Departamento Administrativo Nacional de Estadística

Dirección de Metodología y Producción Estadística - DIMPE

Manual del Sistema Servicios Públicos Domiciliarios del Índice de Precios al Consumidor – IPC – DANE CENTRAL

Mayo 2010

TABLA DE CONTENIDO

1 ESPECIFICACIONES DEL SISTEMA 12

1.1 REQUERIMIENTOS DEL SISTEMA. 12

1.2 DESCRIPCION DEL PROYECTO 12

2 MENU PRINCIPAL (IPC2001_MAIN.MPR) 13

2.1 ARCHIVO 13

2.1.1 Close (Cierra la Ventana Activa) 13
2.1.2 Salir (Sale de Aplicativo) 13

2.2 EDITAR 13

2.2.1 Deshacer (Deshace la ultima accion) 13
2.2.2 Copiar (Copia la seleccion al portapapeles) 13
2.2.3 Pegar (Pegar el contenido del Portapapeles) 13

2.3 RESALTAR 13

2.3.1 Periodos de espera 13
2.3.2 Sustituciones Inmediatas 13
2.3.3 Fuentes Complementarias 13
2.3.4 No continúan entre los más vendidos 13
2.3.5 Cambios de Referencia 13
2.3.6 Posibles precios Fuera de Rango 13
2.3.7 Deshacer Resaltar 13

2.4 FORMULARIOS 13

2.4.1 Pre- Prediligenciamiento (Formulario fprepre.scx) 13
2.4.2 Prediligenciamiento (Formulario prediligencia.scx) 13
2.4.3 Reimpresión 13
2.4.3.1 Formularios (Formulario REIMPRESION.scx) 13
2.4.3.2 Decada (Formulario REIMP_DEC.scx) 13
2.4.3.3 Zona (Formulario REIMP_ZONA.scx) 13
2.4.3.4 Fuentes Prediligenciadas Por Decada (Formulario REIMP_REFUE.scx) 13
2.4.4 Captura NUEVO IPC (Formulario fcaptura.scx) 13
2.4.5 Servicios Públicos (Formulario FSERVPUBI.scx) 13
2.4.6 Transporte Urbano (Formulario FTRANSURB.scx) 13

2.5 MANTENIMIENTO 14

2.5.1 Mantenimiento General (Formulario mantenimiento.scx) 14

2.5.2 Ruta Base de Datos (Formulario pidedir.scx) 14

2.6 HERRAMIENTAS 14

2.6.1 Generar Archivo DANE CENTRAL (Formulario GENERACION.scx) 14
2.6.2 Copia de seguridad 14
2.6.2.1 Generar (Formulario COPIASEG.scx) 14
2.6.2.2 Recuperar (Formulario RECOSEG.scx) 14
2.6.3 Historicos 14
2.6.3.1 Generar Copia(Formulario COPHIST.scx) 14
2.6.3.2 Recuperar Historicos (Formulario REC_HIST.scx) 14
2.6.4 Administración de usuarios 14
2.6.4.1 Ingreso (Formulario INGRESO.scx) 14
2.6.4.2 Modificar (Formulario M_USUARIO.scx) 14
2.6.4.3 Inactivar (Formulario E_USUARIO.scx) 14
2.6.5 Imprimir F.U.R. en Blanco (Formulario FBLANCOS.scx) 14
2.6.6 Reportes Canasta Local (Formulario Finfreg.scx) 14

2.7 ACERCA DE (FORMULARIO ACERCADE.SCX) 14

3 ESTRUCTURA DE LA BASE DE DATOS 15

3.1 TABLAS DE LA BASE DE DATOS IPCCLIENTE.DBC 15

3.2 VISTA LOCALES IPCRED.DBC 51

3.3 VISTA REMOTAS IPCRED.DBC 51

4 PROGRAMAS (PRG) 114

4.1 IPC2001_APP.PRG && Programa Principal, es el primero que se ejecuta 114

4.2 ARTI_PRE.PRG && Devuelve el numero de articulos prediligenciados por fuente para que salga impreso en el formulario 118

4.3 BOOT.PRG && Captura la ruta desde donde se esta ejecutando con la idea que se pueda procesar desde cualquir carpeta o unidad logica 118

4.4 CIUDAD.PRG && Devuelve el nombre de la ciudad según los parámetros que recibe 119

4.5 DISTRIBUCION.PRG && Devuelve la distribución de los fuente – articulo según Grupo Fuente y por Decada 119

4.6 DISTRIBUCION1.PRG && Devuelve la distribución de los fuente – articulo según Grupo Fuente y por Periodicidad 123

4.7 ENVEZ4.PRG && Procesa la información capturada para generar el reporte de Precios Promedio por Clase 130

4.8 FCOM.PRG && Devuelve la información de las fuentes complementarias para cada fuente original al momento de salir prediligenciada 132

4.9 PORCENTAJES.PRG && Devuelve los porcentajes de captura para la fuente actual 132

4.10 REPORTE1.PRG && Procesa la información existente para generar el reporte de Canasta para Cada Ciudad 133

4.11 REPORTE2.PRG && Procesa la información existente para generar el reporte de Canasta para Cada Ciudad 135

4.12 REPORTE3.PRG && Procesa la información existente para generar el Listado de Fuentes para Cada Ciudad 135

4.13 REPORTE5.PRG && Procesa la información existente para generar el reporte de Canasta para Cada Ciudad 137

5 FORMULARIOS 138

5.1 ACTUALIZAR FUENTES EXISTENTES(ACTFTES.SCX) 138

5.1.1 Vistas 138
5.1.2 Botón Aceptar 139
5.1.3 Botón Cancelar 140

5.2 FORMULARIO GENERAR HISTORICOS (genhist.scx) 140

5.2.1 Botón Buscar 141
5.2.2 Botón Salir 142
5.2.3 Botón Iniciar 142
5.2.4 Combo para seleccionar Mes 145
5.2.5 Combo para seleccionar Año 145

5.3 FORMULARIO GENERAR COPIA DE SEGURIDAD (copiaseg.scx) 146

5.3.1 Vistas 146
5.3.2 Botón Buscar 147
5.3.3 Botón Salir 148
5.3.4 Botón Iniciar 148

5.4 FORMULARIO PARA ELIMINAR USUARIOS(e_usuario.scx) 150

5.4.1 Vistas 150
5.4.2 Botón Salir 152
5.4.3 Botón Eliminar 153
5.4.4 Escriba la actual clave de acceso 154
5.4.5 Evento Haga Click sobre Nombre del Usuario que desea eliminar Objeto List 155

5.5 CAMBIO DE ESPECIFICACION (FESPECIFICA.SCX) 155

5.5.1 Vistas 155
5.5.2 Botón Aceptar 157
5.5.3 Botón Cancelar 157
5.5.4 Botón Eliminar 158
5.5.5 Botón Adicionar 158
5.5.6 Entorno de Datos 159

5.6 FORMULARIO INICIAL (FINICIAL.SCX) 160

5.6.1 Vistas 160

5.7 FORMULARIO PARA BARRA DE PROGRESO (FPROGRESO.scx) 161

5.8 FORMULARIO PARA CAPTURAR FUENTES COMPLEMENTARISAS (FTESCOM.SCX) 161

5.8.1 Vistas 161
5.8.2 Boton Seleccionar 163
5.8.3 Boton Cerrar 164

5.9 FORMULARIO PARA CALCULAR SERVICIOS PUBLICOS (Fserpub.SCX) 166

5.9.1 Vistas 166

5.10 FORMULARIO PARA INGRESAR USUARIOS (Iusuario.SCX) 226

5.10.1 Vistas 226
5.10.2 Boton cancelar 228
5.10.3 Boton aceptar 228
5.10.4 Clave de Acceso(Text2) 230
5.10.5 Nivel de Acceso (Spinner1) 231
5.10.6 Codigo (Text5) 231

5.11 FORMULARIO PARA MODIFICAR USUARIOS (musuario.SCX) 231

5.11.1 Vistas 231
5.11.2 Boton Salir 233
5.11.3 Boton Aceptar 233
5.11.4 Digite el nombre de Usuario del usuario a modificar 234
5.11.5 Escriba la actual clave de acceso 235
5.11.6 Escriba la Nueva Clave 237
5.11.7 Nivel de Acceso 237

5.12 FORMULARIO PARA MANTENIMIENTO GENERAL (MANTENIMIENTO.SCX) . 237

5.12.1 Entorno de Datos 237
5.12.2 Vistas 238

5.13 MANTENIMIENTO FUENTE 244

5.13.1 Boton Salir 256
5.13.2 Boton Cancelar 257
5.13.3 Boton Guardar 257
5.13.4 Nombre Fuente(enombfte) 265
5.13.5 Codigo Fuente(ecodfte) 265
5.13.6 Zona(ezona) 272
5.13.7 Grupo Fuente(egrupfte) 272

5.14 MANTENIMIENTO FUENTE ARTICULO 273

5.14.1 Navegador (Navigator1) 273
5.14.2 Nombre Fuente(CNombreFuente) 274
5.14.3 Escriba el Código de la Fuente(EcodFte) 277
5.14.4 Decada(edeca) 285
5.14.5 Periodicidad(eperifte) 285
5.14.6 Boton Continuar 286
5.14.7 Boton Siguiente (Siguiente) 287
5.14.8 Periodicidad(Spinner1) 287
5.14.9 Boton Adicionar(Binsertar) 288
5.14.10 Boton Anterior(Banterior) 289
5.14.11 Boton Eliminar(Command1) 290
5.14.12 Boton Finalizar(Bfinalizar) 290

5.15 MANTENIMIENTO DE ESPECIFICACIONES 291

5.15.1 Seleccione el Nombre de la Fuente que desea Ubicar(CnombreFuente) 292
5.15.2 Boton Siguiente(Bsiguiente) 293
5.15.3 Columna Codigo Especificacion (Column1) 294
5.15.4 Columna Nombre Especificacion (Column2) 294
5.15.5 Columna Detalle Especificacion (Column3) 295
5.15.6 Columna del Codigo Especificacion (Column1) 295
5.15.7 Boton Adicionar(Binsertar) 296
5.15.8 Boton Anterior(Banterior) 297
5.15.9 Boton Eliminar(Command1) 298
5.15.10 Boton Cerrar(Bcancelar) 298
5.15.11 Seleccionar Codigo Articulo (Combo1) 299
5.15.12 Digite el código de la fuente que desea ubicar(EcodFte) 299
5.15.13 Seleccionar codigo Articulo desde la cuadricula (Text1) 301
5.15.14 Seleccione el Código del Articulo(Combo1) 302
5.15.15 Boton Salir 305
5.15.16 Boton Cancelar 305
5.15.17 Boton Guardar 306

5.16 MANTENIMIENTO DE MOVIMIENTO 320

5.16.1 Navegador 320
5.16.2 Seleccione el Nombre de la Fuente(CnombreFuente) 323
5.16.3 Escriba el Código de la Fuente(EcodFte) 326
5.16.4 Seleccione el Nombre de la Fuente(cnombrefuente1) 329
5.16.5 Escriba el Código de la Fuente(ecodfte1) 331
5.16.6 Fuente destino 333
5.16.7 Decada (edeca) 338
5.16.8 Periodicidad (eperifte) 338

5.17 FORMULARIO PARA LIMPIAR BASE DE DATOS (PIDEDIR.SCX) 339

5.17.1 Vistas 339
5.17.2 Boton Ubicar la Base de Datos 339
5.17.3 Boton Aceptar 340
5.17.4 Boton Salir 340
5.17.5 Boton limpiar 340
5.17.6 Boton Reindexar 346

5.18 FORMULARIO PARA EL CIERRE (FSPUBCIERRE.SCX) 352

5.18.1 Entorno de Datos 352
5.18.2 Vistas 352
5.18.3 Boton Cierre 354
5.18.4 Boton salir 356
5.18.5 Boton Prediligenciar 356

5.19 FORMULARIO PARA EL REPORTE FINAL (CIUDAD.SCX) 365

5.19.1 Procedimiento INIT 365
5.19.3 Boton Command4 (Precios consolidados) 366
5.19.4 Boton Salir 366

5.20 FORMULARIO PARA RECUPERAR COPIA DE SEGURIDAD (reccopiaseg.SCX)366

5.20.1 Vistas 366
5.20.2 Boton Buscar 368
5.20.3 Boton Salir 368
5.20.4 Boton Iniciar 368

5.21 FORMULARIO PARA INGRESAR AL APLICATIVO (SEGURO.SCX) 370

5.21.1 Vistas 370
5.21.2 Boton Salir 371
5.21.3 Clave de acceso(Text2) 371
5.21.4 Boton aceptar 371
5.21.4.1.1 PAGEFRAME1.PAGE5.text20 374
5.21.4.1.2 PAGEFRAME1.PAGE5.text21 374
5.21.4.1.3 PAGEFRAME1.PAGE5.text23 374
5.21.4.1.4 PAGEFRAME1.PAGE5.text22 374

5.21.4.1.5 PAGEFRAME1.PAGE5.Navigator2 374
5.21.4.1.6 PAGEFRAME1.PAGE5.ecodfte 375
5.21.4.1.7 PAGEFRAME1.PAGE5.Grid2 376
5.21.5 PAGEFRAME1.PAGE7.pageframe1 378
5.21.5.1 PAGEFRAME1.PAGE7.pageframe1.page1 379
5.21.5.2 PAGEFRAME1.PAGE7.pageframe1.page2 379
5.21.5.3 PAGEFRAME1.PAGE7.pageframe1.page3 381
5.21.5.4 PAGEFRAME1.PAGE7.pageframe1.page4 382
5.21.5.5 PAGEFRAME1.PAGE7.pageframe1.page1.cmdAdd 384
5.21.5.6 PAGEFRAME1.PAGE7.pageframe1.page1.Navigator1 384
5.21.5.7 PAGEFRAME1.PAGE7.pageframe1.page1.text3 385
5.21.5.8 PAGEFRAME1.PAGE7.pageframe1.page1.navigator2 386
5.21.5.9 PAGEFRAME1.PAGE7.pageframe1.page1.text4 387
5.21.5.10 PAGEFRAME1.PAGE7.pageframe1.page2.cmdAdd 388
5.21.5.11 PAGEFRAME1.PAGE7.pageframe1.page2.Navigator1 389
5.21.5.12 PAGEFRAME1.PAGE7.pageframe1.page2.Text3 390
5.21.5.13 PAGEFRAME1.PAGE7.pageframe1.page2.navigator2 391
5.21.5.14 PAGEFRAME1.PAGE7.pageframe1.page2.Ecodfte 391
5.21.5.15 PAGEFRAME1.PAGE7.pageframe1.page2.ListSource 392
5.21.5.16 PAGEFRAME1.PAGE7.pageframe1.page3.CmdAdd 393
5.21.5.17 PAGEFRAME1.PAGE7.pageframe1.page3.Navigator1 394
5.21.5.18 PAGEFRAME1.PAGE7.pageframe1.page3.text3 396
5.21.5.19 PAGEFRAME1.PAGE7.pageframe1.page3.Navigator2 397
5.21.5.19.1 PAGEFRAME1.PAGE7.pageframe1.page3.ecodfte 398
5.21.5.20 PAGEFRAME1.PAGE7.pageframe1.page3.filtrar 398
5.21.5.21 PAGEFRAME1.PAGE7.pageframe1.page4.Navigator1 398
5.21.5.22 PAGEFRAME1.PAGE7.pageframe1.page3.text3 400
5.21.5.23 PAGEFRAME1.PAGE7.pageframe1.page3.Navigator2 401
5.21.5.24 PAGEFRAME1.PAGE7.pageframe1.page3.Filtrar 401
5.21.5.25 PAGEFRAME1.PAGE7.pageframe1.page4.grid1 402
5.21.5.26 PAGEFRAME1.PAGE7.pageframe1.page4.Ecodfte 404
5.21.6 PAGEFRAME1.PAGE8 405
5.21.6.1 PAGEFRAME1.PAGE8.command3 405
5.21.6.2 PAGEFRAME1.PAGE8.command1 405
5.21.6.3 PAGEFRAME1.PAGE8.command2 405
5.21.6.4 PAGEFRAME1.PAGE8.command5 406
5.21.6.5 PAGEFRAME1.PAGE8.command6 406
5.21.6.6 PAGEFRAME1.PAGE8.command4 406
5.21.6.7 PAGEFRAME1.PAGE8.List1 416
5.21.6.8 PAGEFRAME1.PAGE8.List2 416
5.21.6.9 PAGEFRAME1.PAGE8.command7 416
5.21.7 Command2 419
5.21.8 Command1 420
5.21.9 WzBoton4 420

6 INFORMES 420

6.1 LSERVPUB2.FRX - VARIACIONES , PRECIOS PONDERADOS PARA UNA FUENTE Y RESULTADOS FINALES QUE ENTRAN AL CALCULO. 420

6.2 LSERVPUB3.FRX - VARIACIONES Y PRECIOS PONDERADOS PARA LAS FUENTES FINALES QUE ENTRAN AL CALCULO 422

6.3 LSERVPUB.FRX - VARIACIONES Y PRECIOS PONDERADOS PARA UNA FUENTE 423

6.4 LSERVTUR.FRX - RESULTADOS SERVICIO - TRANSPORTE URBANO 423

6.5 REPORTE1.FRX - FUENTE - ARTICULO – ESPECIFICACIONES 425

6.6 REPORTE2.FRX - COTIZACIONES POR ARTICULO 425

6.7 REPORTE3.FRX - DIRECTORIO DE FUENTES 425

6.8 REPORTE4.FRX - PRECIOS PROMEDIO POR CLASE DE FUENTE 426

6.9 RTVARIACION.FRX - VARIACIONES POR ARTÍCULO 426

INTRODUCCION

El presente manual está dirigido a personas con conocimientos en el desarrollo de aplicaciones y programación en ambiente Visual Fox Pro, además, se requiere estar familiarizado con la metodología del Indice de Precios al Consumidor.

Este manual presenta todos los elementos de programación utilizados para el desarrollo de la aplicación, así como la descripción de formularios y procedimientos diseñados con la herramienta **Visual Fox Pro** versión **6.0.**

1 ESPECIFICACIONES DEL SISTEMA

1.1 REQUERIMIENTOS DEL SISTEMA.

En este punto se describe la configuración mínima con la que se puede ejecutar la aplicación para el óptimo funcionamiento de los módulos desarrollados. Es importante tener en cuenta que la aplicación funciona con una configuración menor, pero el tiempo de respuesta en la ejecución de procesos será deficiente.

Velocidad de procesador: 1.8 Mhz
Memoria RAM: 512 Mb.
Espacio disponible en disco duro: 1 Gb.
Configuración de pantalla por Panel de Control: 800 x 600 píxeles.

1.2 DESCRIPCION DEL PROYECTO.

Esta aplicación está desarrollada con el Administrador de Proyectos (herramienta de Visual Fox Pro 6.0), y la descripción que se hace a continuación con respecto al desarrollo del software utiliza el mismo orden propuesto por el módulo administrador.

El proyecto se llama **ipc2001.pjx**, y está ubicado en la carpeta E**:\nuevoipc.**

A continuación se describen los componentes del proyecto:

2 MENU PRINCIPAL (IPC2001_MAIN.MPR)

2.1 ARCHIVO

2.1.1 Close (Cierra la Ventana Activa)
2.1.2 Salir (Sale de Aplicativo)

2.2 EDITAR

2.2.1 Deshacer (Deshace la ultima accion)
2.2.2 Copiar (Copia la seleccion al portapapeles)
2.2.3 Pegar (Pegar el contenido del Portapapeles)

2.3 RESALTAR

2.3.1 Periodos de espera
2.3.2 Sustituciones Inmediatas
2.3.3 Fuentes Complementarias
2.3.4 No continúan entre los más vendidos
2.3.5 Cambios de Referencia
2.3.6 Posibles precios Fuera de Rango
2.3.7 Deshacer Resaltar

2.4 FORMULARIOS

2.4.1 Pre- Prediligenciamiento (Formulario fprepre.scx)
2.4.2 Prediligenciamiento (Formulario prediligencia.scx)
2.4.3 Reimpresión
2.4.3.1 Formularios (Formulario REIMPRESION.scx)
2.4.3.2 Decada (Formulario REIMP_DEC.scx)
2.4.3.3 Zona (Formulario REIMP_ZONA.scx)
2.4.3.4 Fuentes Prediligenciadas Por Decada (Formulario REIMP_REFUE.scx)
2.4.4 Captura NUEVO IPC (Formulario fcaptura.scx)
2.4.5 Servicios Públicos (Formulario FSERVPUBI.scx)
2.4.6 Transporte Urbano (Formulario FTRANSURB.scx)

2.5 MANTENIMIENTO

2.5.1 Mantenimiento General (Formulario mantenimiento.scx)
2.5.2 Ruta Base de Datos (Formulario pidedir.scx)

2.6 HERRAMIENTAS

2.6.1 Generar Archivo DANE CENTRAL (Formulario GENERACION.scx)
2.6.2 Copia de seguridad
2.6.2.1 Generar (Formulario COPIASEG.scx)
2.6.2.2 Recuperar (Formulario RECOSEG.scx)
2.6.3 Historicos
2.6.3.1 Generar Copia(Formulario COPHIST.scx)
2.6.3.2 Recuperar Historicos (Formulario REC_HIST.scx)
2.6.4 Administración de usuarios
2.6.4.1 Ingreso (Formulario INGRESO.scx)
2.6.4.2 Modificar (Formulario M_USUARIO.scx)
2.6.4.3 Inactivar (Formulario E_USUARIO.scx)
2.6.5 Imprimir F.U.R. en Blanco (Formulario FBLANCOS.scx)
2.6.6 Reportes Canasta Local (Formulario Finfreg.scx)

2.7 ACERCA DE (FORMULARIO ACERCADE.SCX)

3 ESTRUCTURA DE LA BASE DE DATOS

3.1 TABLAS DE LA BASE DE DATOS IPCCLIENTE.DBC

☞ **archivos**

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
procod		Numérico	1	0
arccod		Numérico	2	0
variable		Carácter	10	
dbfpocket		Carácter	30	
dbfpc		Carácter	30	
arcbas		Lógico	1	
arcget		Lógico	1	
arcdec		Lógico	1	
arcmvt		Lógico	1	
arcobs		Carácter	30	
bdpocket		Carácter	30	
bdpc		Carácter	30	

☞ **auxiliar:** Contiene informacion de los usuarios que son registrados para poder trabajar sobre el aplicativo.

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
quien	Nombre de usuario	Carácter	20	
conque	Contraseña de Usuario	Carácter	8	
que	Nivel de Acceso al aplicativo (1-3)	Carácter	1	
activo	Indica si el usuario esta activo o no para ingresar al aplicativo	Lógico	1	
codigo	Codigo Identificador de Usuario	Carácter	10	
cargo	Cargo de Usuario	Carácter	25	

☞ **esptem:** Contiene informacion historica de los periodos anteriores que se han procesado, tal como especificaciones de una fuente articulo, para elmomento de consultarse un periodo anterior obtengamos las especificaciones reales de ese periodo.

Indices:

Nombre	Tipo	Expresión
NKCODFTE	Normal	codfte

Nombre	Tipo
PKESP	Normal

Expresión
ALLTRIM(STR(ano,4,0))+ALLTRIM(STR(mes,2,0))+ALLTRIM(STR(decmvt,1,0))+ALLTRIM(STR(impctv,7,0))

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codfte	Codigo Fuente	Carácter	20	
codart	Codigo Articulo	Carácter	7	
codesp	Codigo Especificacion	Carácter	2	
nomesp	Nombre Especificacion	Carácter	40	
nombdet	Descripcion de la Especificacion	Carácter	40	
nombdet1	Almacena el detalle asignado a la especificacion asignada al marcar C.R. o S.I	Carácter	40	
impctv	Consecutivo de Impresion	Numérico	7	0
ano	Año del periodo en que aplico la especificacion para esta fuente Articulo	Numérico	4	0
mes	Mes del periodo en que aplico la especificacion para esta fuente Articulo, en conjunto con el año	Numérico	2	0
decmvt	Decada del periodo en que aplico la especificacion para esta fuente Articulo, en conjunto con el año y el mes	Numérico	1	0
marca	Indica si esta marcado para ser eliminado	Lógico	1	
borra	Indica si el registro esta marcado para ser eliminado.	Lógico	1	
demas	Marca logica	Lógico	1	

☞ **estrato:** Contiene Informacion de Consumos, numero de Usuarios, Cargos Fijos para cada estrato de acuerdo a la ciudad y servicio.

Indices:

Nombre	**Tipo**	**Expresión**
NKRANGO	Normal	coddpto+ano+mes+codart+codfuente

Nombre	**Tipo**	**Expresión**
pkestrato	Principal	coddpto+ano+mes+codart+codfuente+codestrato

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
coddpto	Codigo Departamento	Carácter	2	
ano	Año que indica el periodo donde aplico la informacion	Carácter	4	
mes	Mes que indica el periodo donde aplico la informacion en conjunto con el año	Carácter	2	
codfuente	Consecutivo de la Fuente	Carácter	4	
codart	Codigo Articulo	Carácter	7	
codestrato	Identificacion del Estato (1-6)	Carácter	1	
usuarios	Numero de Usuarios que aplico para elperiodo y Estrato	Numérico	9	0
consumo	Consumo que aplico para el periodo y Estrato	Numérico	6	0
cargofijo	Cargo que se cobra de acuerdo al estrato por tener el servicio	Numérico	12	4
codgri	Codigo de Grupo de Ingreso	Carácter	1	
qo_pond	Poderacion asignada	Numérico	8	4
impuesto	Porcentaje de acuerdo al servico y a la ciudad	Numérico	12	4
fpdepor	Porcentaje facturado de acuerdo al servicio y a la ciudad	Numérico	12	4
marca	Indica si esta marcado para ser eliminado	Lógico	1	

☞ **historia:** Contiene la informacion de la captura correspondiente a periodos anteriores, cada vez que se prediligencia primera decada se descarga lo que se capturo en el mes inmediatamente anterior.

Indices:

Nombre	Tipo	Expresión
NKCODFTE	Normal	codfte

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codfte	Codigo Fuente	Carácter	20	
codart	Codigo Articulo	Carácter	7	
impctv	Consecutivo de Impresion	Numérico	7	0
panart	Precio Anterior, recogido para el periodo año mes decada de la fuente articulo	Numérico	11	2
pactmvt	Precio Actual recogido para el periodo año mes decada de la fuente articulo	Numérico	11	2
cmvmvt	Novedad Tecnica Continua Entre los mas Vendidos, asignado para el periodo año mes decada de la fuente articulo	Lógico	1	
crfmvt	Novedad Tecnica Cambio de Referencia, asignado para el periodo año mes decada de la fuente articulo	Lógico	1	
pesmvt	Novedad Tecnica Periodo de Espera , asignado para el periodo año mes decada de la fuente articulo	Lógico	1	
sinmmvt	Novedad Tecnica Sustitucion Inmediata, asignado para el periodo año mes decada de la fuente articulo	Lógico	1	
fcommvt	Novedad Tecnica Fuente Complementaria (Codigo), asignado para el periodo año mes decada de la fuente articulo	Carácter	1	
canrmvt	Cantidad Recolectada en el periodo año mes decada de la fuente articulo	Numérico	8	2
obsmvt	Codigo de Observacion, asignado para el periodo	Carácter	2	

	año mes decada de la fuente articulo			
decmvt	Decada que identifica el periodo en conjunto con el año y el mes	Numérico	1	0
total	Numero de Dia para el reporte de Captura Diaria	Numérico	2	0
cantant	Cantidad Anterior asignada para el periodo año mes decada de la fuente articulo	Numérico	8	2
ano	Indica el periodo en conjunto con el mes y la decada	Numérico	4	0
mes	Indica el periodo en conjunto con el año y la decada	Numérico	2	0
pesart	Periodo de Espera Anterior	Lógico	1	
variacion	variacion	Carácter	7	
marca	Indica si esta marcado para ser eliminado	Lógico	1	
borra	Indica si el registro esta marcado para ser eliminado.	Lógico	1	

☞ **histpre:** Contiene informacion de los precios historicos para los articulos agrupados por fuentes para consultarlos en el modulo de analisis, con el fin de descongestionar los archivos historisa y esptemp, es decir, no necesitarlos para consultar los precios pasado.

Indices:

Nombre	Tipo	Expresión
NKCODFTE	Normal	codfte
NKFTEART	Normal	codfte+codart

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codfte	Codigo Fuente	Carácter	20	
codart	Codigo Articulo	Carácter	7	
decmvt	Decada Precio Historico	Numérico	1	0
mesmvt	Mes Precio Historico	Numérico	2	0
anomvt	Año Precio Historico	Numérico	2	0
pesart	Periodo de Espera Anterior para la Fuente Articulo	Lógico	1	
panart	Precio Anterior para Fuente Articulo	Numérico	11	2
cantant	Cantidad Anterior para la Fuente Articulo	Numérico	8	2
pactmvt	Precio Actual para la Fuente Articulo	Numérico	11	2

canrmvt	Cantidad Recolectada a la fuente articulo	Numérico	8	2
pesmvt	Indica que se marco Periodo de Espera	Lógico	1	
crfmvt	Indica que se marco Cambio de Referencia	Lógico	1	
sinmmvt	Indica que se marco Sustitucion Inmediata a la fuente articulo en el periodo correspondiente	Lógico	1	
fcommvt	Contiene el codigo de la fuente complementaria asignada a la fuente articulo en el periodo correspondiente	Carácter	1	
marca	Marca Logica	Lógico	1	
borra	Marca Logica queindica si esta marcadopara ser eliminado	Lógico	1	
demas	Marca Logica	Lógico	1	
pci		Lógico	1	

☞ **histpre:** Contiene Informacion de los indicadores de la ciudad.

Indices:

Nombre	Tipo	Expresión
pkindciud	Principal	coddpto+codmpio+ano+mes+decada

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
coddpto	Codigo Departamento	Carácter	2	
codmpio	Codigo Municipio	Carácter	3	
ano	Año para el que aplica el indicador	Carácter	4	
mes	Mes para el que aplica el indicador	Carácter	2	
decada	Decada para el que aplica el indicador	Carácter	1	
proceso	Proceso al cual pertenece el indicador (Recoleccion, Supervision, Captura, Calidad)	Carácter	2	
indciudad	Valor Indicador Ciudad	Carácter	10	
marca	Marca Logica	Lógico	1	
borra	Indica si esta marcado para	Lógico	1	

	ser eliminado			
demas	Marca Logica	Lógico	1	
fechaso	Fecha Computador, actual	Fecha	8	

☞ **ipccargo:** Contiene Informacion de los cargos que se tendran en cuenta para los Indicadores de Calidad.

Indices:

Nombre	Tipo	Expresión
Pkcodcargo	Principal	codcargo

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codcargo	Codigo Cargo	Carácter	1	
nomcargo	Nombre Cargo	Carácter	20	
marca	Marca logica	Lógico	1	
borra	Nos indica si esta marcado para eliminarse	Lógico	1	
demas	Maca logica	Lógico	1	

☞ **ipcconfig:**

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
condatbas		Lógico	1	
condatdec		Lógico	1	
condatmvt		Lógico	1	
condatmin		Numérico	2	0
condatmax		Numérico	2	0
condatmaxp		Numérico	2	0
condatid		Numérico	1	0

☞ **ipcdias:** Contiene Informacion correspondiente a la Planilla de Recoleccion.

Indices:

Nombre	Tipo
NKDIAS	Normal

Expresión
ALLTRIM(STR(anoplarec,4,0))+ALLTRIM(STR(mesplarec,2,0))+ALLTRIM(STR(decplarec))+ALLTRIM(STR(zonafunci,2,0))+codfte

Nombre	Tipo	Expresión
NKCODFTE	Normal	codfte

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
zonafunci	Zona asignada a Funcionario	Numérico	2	0
codfunci	Codigo Funcionario	Carácter	10	
anoplarec	Año de proceso	Numérico	4	0
mesplarec	Mes de proceso	Numérico	2	0
decplarec	Decada de proceso	Numérico	1	0
ctvplarec	Consecutivo Planilla de Recoleccion	Carácter	3	
ctvdia	Consecutivo Dia	Carácter	2	
nrodia	Numero Dia	Carácter	2	
codfte	Codigo Fuente	Carácter	20	
ca	Cantidad Articulos a Recolectar	Numérico	4	0
ar	Cantidad Articulos Recolectados	Numérico	4	0
pendientes	Cantidad Articulos Pendientes	Numérico	4	0
observa	Observaciones Planilla de Recoleccion	Memo	4	
marca	Marca Logica	Lógico	1	
borra	Marca Logica	Lógico	1	
demas	Marca Logica	Lógico	1	
fechaso	Fecha Computador, actual	Fecha	8	

☞ **ipcestr:** Contiene la informacion de los estratos manejados para losprocesosinterno en el aplicativo.

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal

estrato	Identificacion de Estrato	Numérico	1	0
nomestr	Nombre de Estrato	Carácter	25	

☞ **ipcfunci:** Contiene Informacion de los Funcionarios que desempeñan los cargos a los que se le aplica los Indicadores de calidad.

Indices:

Nombre	Tipo	Expresión
NKCODCARG	Normal	codcargo
Pkfunci	Principal	codcargo+ALLTRIM(codfunci)

Nombre	Tipo
NKFUNCICAR	Normal

Expresión
codcargo+ALLTRIM(zonafunci)+ALLTRIM(codfunci)

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codcargo	Codigo de Cargo de Funcionario	Carácter	1	
zonafunci	Zona asignada al funcionario	Carácter	25	
codfunci	Codigo de Funcionario	Carácter	10	
nomfunci	Nombre de Funcionario	Carácter	40	
fecha	Fecha registro	Fecha	8	
marca	Marca Logica	Lógico	1	
borra	Indica si esta marcado para ser eliminado	Lógico	1	
demas	Marca Logica	Lógico	1	

☞ **ipcindzo:** Contiene Informacion de los Indicadores de la Zona.

Indices:

Nombre	Tipo	Expresión
Pkfunci	Principal	coddpto+codmpio+ano+mes+decada+zona

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
coddpto	Codigo Departamento	Carácter	2	
codmpio	Codigo Municipio	Carácter	3	
ano	Año para el que aplica el indicador	Carácter	4	
mes	Mes para el que aplica el	Carácter	2	

	indicador			
decada	Decada para el que aplica el indicador	Carácter	1	
zona	Zona para la que aplica el indicador	Carácter	2	
proceso	Proceso al cual pertenece el indicador (Recoleccion, Supervision, Captura, Calidad)	Carácter	2	
indicazona	sumatoria totales indicadores / sumatoria ponderaciones Indicador de la Zona	Numérico	6	2
marca	Marca Logica	Lógico	1	
borra	Indica si esta marcado para ser eliminado	Lógico	1	
demas	Marca Logica	Lógico	1	
fechaso	Fecha computador, actual	Fecha	8	

☞ **ipcmart:** Contiene la informacion de los articulos que componen la Canasta Nacional, con su informacion base.

Indices:

Nombre	Tipo	Expresión
CODART	Principal	codart
NKART	Normal	codart

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
Codart	Codigo Articulo	Carácter	7	
grpart	Grupo al que pertenece el articulo, ubicado en el primer digito del Codigo Articulo	Numérico	1	0
subgart	Subgrupo al que pertenece el articulo, ubicado en el segundo digito del Codigo Articulo, ademas la suma de Grupo y Subgrupo	Numérico	1	0
clasart	Clase al que pertenece el articulo, ubicado en el tercer digito del Codigo Articulo, ademas la sumade Grupo, Subgrupo y Clase	Numérico	1	0
gasbart	Gasto Basico al que pertenece el articulo,	Carácter	2	

	ubicado en el cuarto y quinto digito del Codigo Articulo			
variart	Variedad que identifica el articulo, ubicado en el los dos ultimos digito del Codigo Articulo, ademas la suma de todos	Carácter	2	
perart	Indica si este articulo Pertenece a la Linea de Pobreza	Numérico	1	0
nomart	Nombre del Articulo	Carácter	44	
canbart	Cantidad Base del Articulo	Numérico	5	0
uniart	Unidad Base del Articulo	Carácter	5	
presart	Presentacion del Articulo	Carácter	10	
calcula	calcula	Carácter	1	
fornal	Indica si forma parte de los Articulos de Formacion Nacional	Lógico	1	
educacion	Indica si forma parte de los Articulos de educacion	Lógico	1	
mensuales	Indica si fomrma parte de los Articulos de Mensuales Observados	Lógico	1	
marca	Marca Logica para uso posterior	Lógico	1	
borra	Marca logica que indica si el registro esta marcado para ser eliminado	Lógico	1	
demas	Marca Logica para evitar tener problemas con las vistas al crear nuevos campos	Lógico	1	
pci	pci	Lógico	1	

☞ **ipcmes:** Contiene Informacion de losmese del año para procesos internos.

Indices:

Nombre	Tipo	Expresión
Pkmes	Principal	cod_mes

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
cod_mes	Identificacion del Mes	Carácter	2	

des_mes	Nombre del Mes	Carácter	10	
ishort	Nombre corto o abreviatura del nombre del Mes	Carácter	3	
totdias	Numero de Dias que tiene el mes	Numérico	2	0
isvariable	Indica si este mes tiene un numero variable de Dias de acuerdo al tipo de año	Lógico	1	

☞ **ipcmfte:** Contiene Informacion de las Fuentes donde se recolecta informacion de la Ciudad.

Indices:

Nombre	Tipo	Expresión
NKESTRATO	Normal	STR(estrfte)
CODFTE	Principal	codfte
NKFTE	Principal	codfte
FTE	Principal	codfte

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codfte	Codigo de Fuente	Carácter	20	
depfte	Departamento al que pertenece la Fuente, corresponde tambien a los dos primeros digitos del Codigo de la Fuente	Carácter	2	
mpiofte	Municipio al que pertenece la Fuente, corresponde tambien a los tres digitos ubicados apartir del tercero del Codigo de la Fuente, es decir, el tres, cuatro y cinco	Carácter	3	
sectfte	Sector donde se encuentra ubicada la fuente en la ciudad, esta informacion es parte de la georeferenciacion, y se compone de cuatro digitos a partir de la posicion 6	Carácter	4	
seccfte	Seccion del Sector donde se encuentra ubicada la fuente en la ciudad, esta informacion es parte de la georeferenciacion, y se compone de dos digitos a partir de la posicion 10	Carácter	2	

manzfte	Manzana de la Seccion del Sector donde se encuentra ubicada la fuente en la ciudad, esta informacion es parte de la georeferenciacion, y se compone de dos digitos a partir de la posicion 12	Carácter	2	
grupfte	Grupo de Ingresos al que pertenece la fuente(1-3),se encuentra en la posicion 14 del Codigo Fuente	Carácter	1	
clasfte	Clase de Fuente (0-9),se encuentra en la posicion 15 del Codigo Fuente, y se compone de dos digitos a partir de la posicion 15	Carácter	2	
consfte	Consecutivo de Fuente,se encuentra en la posicion 17 del Codigo Fuente, y se compone de cuatro digitos a partir de la posicion 17	Carácter	2	
nombfte	Nombre de la Fuente	Carácter	40	
direfte		Carácter	40	
telfte	Telefono de la Fuente	Carácter	40	
estrfte	Identificacion del Estrato de la Fuente	Numérico	1	0
zona	Zona a la que esta asignada la Fuente de acuerdo a la Zonificacion de la ciudad	Numérico	4	0
fecha	Fecha Registro	Fecha	8	
marca	Indica si esta marcada para eliminar	Lógico	1	
borra	Indica si el registro esta marcado para ser eliminado.	Lógico	1	
demas	Marca Logica	Lógico	1	
pci	pci	Lógico	1	

☞ **ipcpdaesp:** Contiene informacion de las fuente articulo especificaciones creadas en terreno.

Indices:

Nombre	**Tipo**	**Expresión**
NKFTEART	Normal	ALLTRIM(codfte)+codart
NKFTE	Normal	codfte

PKPDAESP Normal codfte20

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
zona	Zona a la que pertenece la fuente creada en terreno	Numérico	2	0
codfte	Codigo de fuente creada en terreno	Carácter	3	
codfte20	codigo con veinte digitos inical para luego ser reemplazado por el que defina el usuario	Carácter	20	
codart	Codigo de articulo	Carácter	7	
codesp	Codigo de Especificacion	Carácter	2	
nomesp	Nombre de especificacion	Carácter	25	
nomdet	Detalle de la especificacion	Carácter	40	
marca	Marca logica para posterior uso	Lógico	1	
borra	Marca logica para indicar si el registro esta marcado para ser eliminado	Lógico	1	
demas	Marca logica para evitar tener problemas con las vistas remotas al momento de crear nuevos campos	Lógico	1	

☞ **ipcpdafa:** Contiene informacion de las fuentes nuevas creadas en terreno y los articulos a signados.

Indices:

Nombre	Tipo	Expresión
PKPDAFA	Normal	codfte20

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
zona	Zona a la que pertenece la fuente creada en terreno	Numérico	2	0
codfte	Codigo de fuente creada en terreno	Carácter	3	
codfte20	codigo con veinte digitos	Carácter	20	

	inical para luego ser reemplazado por el que defina el usuario			
codart	Codigo de articulo asignado a la fuente en terreno	Carácter	7	
decart	Decada en la que se recolectaria la fuente articulo	Numérico	1	0
perfap	Periodicidad en la que se recolectaria la fuente articulo	Numérico	1	0
panart	Precio anterior del articulo	Numérico	11	2
cantant	Cantidad anterior del articulo	Numérico	8	2
marca	Marca logica para uso posterior	Lógico	1	
borra	Marca logica que indica si el registro esta marcado para ser eliminado	Lógico	1	
demas	Marca logica para evitar tener problemas con las vistas remotas al momento de crear nuevos campos	Lógico	1	

☞ **ipcpdafte:** Contiene la informacion de las fuentes nuevas creadas en terreno y descargadas de la pocket para posteriormente asignar codigo oficial y habilitarlas.

Indices:

Nombre	**Tipo**	**Expresión**
NKCODFTE	Normal	codfte
pkcodfte20	Principal	ALLTRIM(STR(zona,2,0))+codfte20

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
zona	Zona a la que pertenece la fuente creada en terreno	Numérico	2	0
codfte	Codigo de fuente creada en terreno	Carácter	3	
codfte20	codigo con veinte digitos	Carácter	20	

	inical para luego ser reemplazado por el que defina el usuario			
nombfte	Nombre de la fuente nueva	Carácter	40	
direfte	Direccion de la fuente nueva	Carácter	40	
telfte	Telefono de la fuente nueva	Carácter	15	
grupfte	Contiene el Grupo Fuente al que pertenece	Carácter	1	
claffte	Contiene informacion de la clase fuente a la que pertenece	Carácter	2	
marca	Marca logica para uso posterior	Lógico	1	
borra	Marca logica que indica si el registro esta marcado para ser eliminado	Lógico	1	
demas	Marca logica para evitar tener problemas con las vistas remotas al momento de crear nuevos campos	Lógico	1	

☞ **ipcplact:** Almacenara la informacion de asignacion de Cargas de Trabajo para los diferentes tipos de cargo y sus funcionarios.

Indices:

Nombre	Tipo	Expresión
NKPLACT	Normal	codcargo+zonafunci+anoplarec+mesplarec+decplarec

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codcargo	Codigo Cargo	Carácter	1	
zonafunci	Zona asignada a Funcionario	Carácter	2	
codfunci	Codigo Funcionario	Carácter	10	
anoplarec	Año Planilla Recoleccion	Carácter	4	
mesplarec	Mes Planilla de Recoleccion	Carácter	2	
decplarec	Decada Planilla de Recoleccion	Carácter	1	
ctvdia	Consecutivo Dia	Carácter	2	

nrodia	Numero Dia	Carácter	2	
entra	Numero que entra	Numérico	4	0
sale	Numero que sale	Numérico	4	0
pendientes	Numero Pendientes	Numérico	4	0
totalentra	Total que entran	Numérico	6	0
totalsale	Total que salen	Numérico	6	0
totalpte	Total pendientes	Numérico	6	0
marca	Marca Logica	Lógico	1	
borra	Indica si esta marcado para ser eliminado	Lógico	1	
demas	Marca Logica	Lógico	1	
fechaso	Fecha del computador actual	Fecha	8	

☞ **ipcrango:** Contiene la informacion de los Rangos Maximo y Minimo permitidos para cada Articulo y de acuerdo a los Estratos manejados en los procesos internamente.

Indices:

Nombre	**Tipo**	**Expresión**
CODART	Normal	codart
PKRANGO	Normal	codart+STR(estrfte)+ALLTRIM(clasfte)

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
coddpto	Codigo del departamento	Carácter	2	
codmpio	Codigo del municipio	Carácter	3	
codart	Codigo de articulo	Carácter	7	
estrfte	Identificacion de Estrato	Numérico	1	0
clasfte	Clase fuente	Carácter	2	
maxart	Maximo de Variacion permitida normal para este articulo de acuerdo al Estrato y a la clase	Numérico	7	4
minart	Minimo de Variacion permitida normal para este articulo de acuerdo al Estrato	Numérico	8	4
rango	Rango, diferencia entre el maximo y el minimo	Numérico	8	4

marca	Maraca Logica indica si ha cambiado	Lógico	1	
borra	Indica si esta marcado para ser eliminado	Lógico	1	
demas	Nos sirve para evitar problemas cuado se modifican las tablas originales con respecto a las vistas	Lógico	1	

- **ipcresp:** Contiene informacion de las Fuente con sus articulos y las especificacionescorrespondientes a cada uno, donde se detallan por ejemplo,marcas, tipos, detalles en general.

Indices:

Nombre	**Tipo**	**Expresión**
Pkresp	Principal	codfte+codart+codesp
NKRESP	Normal	codfte+codart
CODFTE	Normal	codfte
CODART	Normal	codart

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codfte	Codigo Fuente	Carácter	20	
codart	Codigo Articulo	Carácter	7	
codesp	Codigo Especificacion	Carácter	2	
nomesp	Nombre Especificacion	Carácter	40	
nombdet	Descripcion del detalle de la especificacion	Carácter	40	
nombdet1	Contiene la especificacion asignada al marcar cambio de referencia o Sustitucion Inmediata	Carácter	40	
tipo	Tipo de Especificacion	Numérico	1	0
valor	valor	Carácter	25	
fecha	Fecha Registro	Fecha	8	
orden	Destinado para que el	Carácter	3	

	usuario ordene las especificaciones escibiendo el numero deseado			
reorden	para reordenar luego de haber limpiado Especifcaciones.	Carácter	2	
marca	Indica si esta marcadopara ser eliminado	Lógico	1	
borra	Indica si el registro esta marcado para ser eliminado.	Lógico	1	
demas	Marca logica	Lógico	1	

☞ **ipcrfap:** Contiene Informacion de lasFuentes con los Articulos que cotizan en ellas, indicando Periodicidad(mes) y Decada para cuando deben ser prediligenciadas y recolectadas.

Indices:

Nombre	Tipo	Expresión
NKART	Normal	codart
Pkfap	Principal	codfte+codart
NKFTE	Normal	codfte

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codfte	Codigo Fuente, proviene de ipcmfte, la relacionamos con los articulos que cotiza	Carácter	20	
codart	Codigo articulo, proviene del ipcmart(canasta)	Carácter	7	
decart	Decada en la cual debe salir prediligenciado la Fuente-Articulo para ser recolectado	Numérico	1	0
panart	Precio Anterior (Ultimo Precio) recolectado de la Fuente Articulo	Numérico	11	2
furart	Fecha Ultima Recoleccion	Numérico	6	0
perfap	Periodicidad asignada a la fuente Articulo, indica el mes en que debe salir prediligenciado, en juego con la decada	Numérico	1	0
cantant	Cantidad Anterior (Ultimacantidad) recolectada para esa fuente Articulo	Numérico	8	2

pesart	Periodo de Espera asignado en el ultimo periodo que se recolecto la Fuente articulo	Lógico	1	
fcommvt	Codigo de Fuente Complementaria asignada a la Fuente Articulo Original en la ultima recoleccion	Carácter	1	
fecha	Fecha Registro	Fecha	8	
marca	Indica si esta marcada para ser eliminado	Lógico	1	
borra	Indica si el registro esta marcado para ser eliminado.	Lógico	1	
demas	Marca Logica	Lógico	1	
pci	pci	Lógico	1	

☞ **ipcrfunc:** Contiene informacion de los Funcionarios con su respectivo Cargo, ademas de la o las zonas que le corresponde de acuerdo al cargo.

Indices:

Nombre	**Tipo**	**Expresión**
NKRFUNC	Normal	codcargo+zonafunci+ctvzona+ALLTRIM(codfunci)

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codcargo	Codigo cargo	Carácter	1	
codfunci	Codigo Funcionario	Carácter	10	
ctvzona	Consecutivo Zona	Carácter	2	
zonafunci	Zona asignada a Funcionario	Carácter	2	
marca	Marca Logica	Lógico	1	
borra	Marca Logica	Lógico	1	
demas	Marca Logica	Lógico	1	

☞ **ipcrobs:** Contiene Informacion de las observaciones asignadas a las Fuentes que cotizan en la ciudad.

Indices:

Nombre	**Tipo**	**Expresión**
NKIPCFTE	Normal	codfte
IMPCTV	Normal	impctv
Pkobsv	Principal	ALLTRIM((STR(impctv,7,0)))+codfte+codobs

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
impctv	Consecutivo de Impresion	Numérico	7	0
codfte	Codigo Fuente	Carácter	20	
codobs	Codigo Observacion	Carácter	2	
observa	Descripcion de la Observacion	Memo	4	
marca	Indica si esta marcado para ser eliminado	Lógico	1	
borra	Indica si el registro esta marcado para ser eliminado.	Lógico	1	
demas	Marca logica	Lógico	1	

☞ **ipcrpart:** Contiene informacion de las periodicidades y los mese a los que corresponde cada una.

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codper	Codigo de Periodicidad Generales	Numérico	1	0
codpart	Codigo de Periodicidad por Articulo	Numérico	1	0
mes	Mes al que corresponde la periodicidad	Numérico	2	0

☞ **ipctctv:** Contiene informacion de los consecutivos de impresion que decada a decada y mes a mes se incrementa y se le asignan a las fuentes para que en captura se puedan ubicar por este numero y para que historicamente se pueda identificar.

Indices:

Nombre	Tipo	Expresión
NKCTV	Normal	finctv

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
inictv	Consecutivo Inicial	Numérico	7	0
finctv	Consecutivo Final	Numérico	7	0

añoctv	Año para llevar el control de la informacion que se va aprediligenciar sin depender de la fecha delcomputador	Numérico	4	0
mesctv	Mes para llevar el control de la informacion que se va aprediligenciar sin depender de la fecha delcomputador	Numérico	2	0
decctv	Decada para llevar el control de la informacion que se va aprediligenciar sin depender de la fecha delcomputador	Numérico	1	0
estpre	Marca para llevar el control de si ya se prediligencio o no un periodo determinado	Lógico	1	
estcap	Marca para llevar el control de si ya se capturo lo que esta en movimiento del periodod actual	Lógico	1	
esttras	Marca para llevar el control de si ya se Transmitio, es decir, se hizo el envio de la informacion a la SEDE CENTRAL	Lógico	1	
estpro	Estado del proceso	Lógico	1	
fecha	Fecha Registro	Lógico	1	

☞ **ipctfcm:** Contiene informacion de las fuentes Complementarias asignadas a las diferentes Fuentes Originales de las ciudaddes.

Indices:

Nombre	**Tipo**	**Expresión**
pkfcm	Principal	ALLTRIM(STR(codfcom))+codfte
NKCODFTE	Normal	codfte

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codfcom	Codigo Fuente Complementaria	Numérico	1	0
nomfte	Nombre Fuente Complementaria	Carácter	40	
direfte	Direccion Fuente Complementaria	Carácter	40	
telfte	Telefono Fuente Complementaria	Carácter	20	
codfte	Codigo Fuente Original, es decir, a la que se le relaciona la Fuente Complementaria	Carácter	20	
fcommvt	Fuente complementaria	Carácter	1	
fecha	Fecha Registro	Fecha	8	
marca	Indica si se va a eliminar esta Fuente Complementaria	Lógico	1	
borra	Indica si el registro esta marcado para ser eliminado.	Lógico	1	
demas	Marca Logica	Lógico	1	

☞ **ipctmvt:** Contiene Informacion de lo que se debe capturar, con datos anteriores, novedades tecnicas, etc.

Indices:

Nombre	Tipo	Expresión
NKIMPCTV	Normal	impctv
WIZARD_1	Normal	STR(impctv,7,0)+codfte+codart
NKMVT	Normal	codfte+codart
Pkmvt	Principal	STR(impctv,7,0)+codfte+codart
MOVI	Normal	codfte

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
ano	Año que corresponde al periodo actual, en conjunto con el mes y la decada	Numérico	4	0
mes	Mes que corresponde al periodo actual, en conjunto con el Año y la decada	Numérico	2	0
decmvt	Decada que corresponde al periodo actual, en conjunto con el mes y el año	Numérico	1	0
zona	Zona a la que esta asignada la Fuente	Numérico	4	0
codfte	Codigo Fuente	Carácter	20	
codart	Codigo Articulo	Carácter	7	
impctv	Consecutivo de Impresion	Numérico	7	0
panart	Precio anterior, corresponde al dato entregado en el ultimo periodo	Numérico	11	2
pactmvt	Precio Actual, corresponde al dato entregado en el preiodo actual	Numérico	11	2
cmvmvt	Novedad Tecnica, Continua Entre Los Mas Vendidos	Lógico	1	
crfmvt	Novedad Tecnica, Cambio de Referencia	Lógico	1	
pesmvt	Novedad Tecnica, Periodo de Espera	Lógico	1	
sinmmvt	Novedad Tecnica, Sustitucion Inmediata	Lógico	1	
fcommvt	Novedad Tecnica, Fuente Complementaria (Codigo)	Carácter	1	
canrmvt	Cantidad Recolectada, corresponde a la cantidad cotizada para el articulo en la fuente	Numérico	8	2
obsmvt	Codigo de la Observacion	Carácter	2	

total	Numero de Dia, utilizado para las consultas en el reporte de Captura Diaria	Numérico	2	0
cantant	Cantidad Recolectada en el Periodo Anterior	Numérico	8	2
pesart	Periodo de Espera asignado en el periodo anterior	Lógico	1	
variacion	Variacion de acuerdo a la informacion capturada del periodo actual contra la del periodo anterior	Carácter	7	
movdiasup	fecha de supervision	Fecha	8	
codcargo	Codigo Cargo, Indicadores de calidad	Carácter	1	
codfunci	Codigo Funcionario, Indicadores de calidad	Carácter	10	
ctvdia	Consecutivo Dia para los formatos de Indicadores de calidad	Carácter	2	
diacri	Dia para el formato de critica, Indicadores de calidad	Carácter	2	
diasup	Dia para el formato de Supervision, Indicadores de calidad	Carácter	2	
diacalicri	Dia para el formato de control calidad Critica, Indicadores de calidad	Carácter	2	
diacalisup	Dia para el formato de Control Calidad Supervision, Indicadores de calidad	Carácter	2	
diacalicap	Dia para el formato control calidad Captura, Indicadores de calidad	Carácter	2	
ctvoruta	Consecutivo de Ruta para la Planilla de Recoleccion	Carácter	6	
ruta	Numero de Ruta	Carácter	3	
plarec	Indica que la informacion de este registro corresponde a la planilla de Recoleccion	Lógico	1	
placri	Indica que la informacion de este registro corresponde a la planilla de Critica	Lógico	1	
plasup	Indica si la fuente articulo efectivamente se superviso, con anterioridad se debio haber marcado para ser	Lógico	1	

	supervisado sobre otro campo supervisado			
calicri	Indica que la informacion de este registro corresponde a la planilla de Calidad de Critica	Lógico	1	
calisup	Indica que la informacion de este registro corresponde a la planilla de Calidad Supervision	Lógico	1	
calicap	Indica que la informacion de este registro corresponde a la planilla de Calidad captura	Lógico	1	
cumr	Novedad, Unidad Mal Recolectada para Critica	Lógico	1	
sumr	Novedad, Unidad Mal Recolectada para Supervision	Lógico	1	
umr	umr	Lógico	1	
cpmr	Novedad, Precio Mal Recolectada para Critica	Lógico	1	
spmr	Novedad, Precio Mal Recolectado para Supervision	Lógico	1	
pmr	pmr	Lógico	1	
ciape	Novedad, Incorrecta aplicacion Periodo de Espera para Critica	Lógico	1	
siape	Novedad, Incorrecta aplicacion Periodo de Espera para Supervision	Lógico	1	
iape	iape	Lógico	1	
ciasi	Novedad, Incorrecta aplicacion Sustitucion Inmediata para Critica	Lógico	1	
siasi	Novedad, Incorrecta aplicacion Sustitucion Inmediata para Supervision	Lógico	1	
iasi	iasi	Lógico	1	
ciacr	Novedad, Incorrecta aplicacion Cambio de Referencia para Critica	Lógico	1	
siacr	Novedad, Incorrecta aplicacion Cambio de Referencia para Supervision	Lógico	1	
iacr	iacr	Lógico	1	
ciafn	Novedad, Incorrecta	Lógico	1	

	aplicacion Fuente Nueva para Critica			
siafn	Novedad, Incorrecta aplicacion Fuente Nueva para Supervision	Lógico	1	
iafn	iafn	Lógico	1	
cadi	Novedad, Ausencia de Informacion para Critica	Lógico	1	
sadi	Novedad, Ausencia de Informacion para Supervision	Lógico	1	
adi	adi	Lógico	1	
conas	Novedad, Observacion no Acorde a la Situacion para Critica	Lógico	1	
sonas	Novedad, Observacion no Acorde a la Situacion para Supervision	Lógico	1	
onas	onas	Lógico	1	
cnae	Novedad, No Actualizo Especificaciones para Critica	Lógico	1	
snae	Novedad, No Actualizo Especificaciones para Supervision	Lógico	1	
nae	nae	Lógico	1	
obsplacri	Observacion Planilla Critica	Memo	4	
totalplasup	Acumulado Planilla de Supervision	Numérico	3	0
totalplacri	Acumulado Planilla de Critica	Numérico	3	0
imsxf	Informacion Mal Suministrada Por la Fuente	Lógico	1	
sdgf	Datos Generales de la Fuente para supervision	Lógico	1	
cadgf	Datos Generales de la Fuente para captura	Lógico	1	
crdgf	Datos Generales de la Fuente para Critica	Lógico	1	
dgf	dgf	Lógico	1	
pe	Periodo de Espera, Indicadores de Calidad	Lógico	1	
si	Sustitucion Inmediata, Indicadores de Calidad	Lógico	1	
cr	Cambio de Referencia, Indicadores de Calidad	Lógico	1	
fn	Fuente Nueva, Indicadores de Calidad	Lógico	1	

obsplasup	Observaciones Planilla Supervision	Memo	4	
caea	Especificaciones del Articulo, Para Captura	Lógico	1	
crea	Especificaciones del Articulo, Para Critica	Lógico	1	
ea	ea	Lógico	1	
caur	Unidad Recolectada, Para Captura	Lógico	1	
crur	Unidad Recolectada, Para Critica	Lógico	1	
ur	ur	Lógico	1	
capr	Precio Recolectado, Para Captura	Lógico	1	
crpr	Precio Recolectado, Para Critica	Lógico	1	
pr	pr	Lógico	1	
caape	Aplicacion Periodo de Espera, Para Captura	Lógico	1	
crape	Aplicacion Periodo de Espera, Para Critica	Lógico	1	
ape	ape	Lógico	1	
caasi	Aplicacion Sustitucion Inmediata, Para Captura	Lógico	1	
crasi	Aplicacion Sustitucion Inmediata, Para Critica	Lógico	1	
asi	asi	Lógico	1	
caacr	Aplicacion Cambio de Referencia, Para Captura	Lógico	1	
cracr	Aplicacion Cambio de Referencia, Para Critica	Lógico	1	
acr	acr	Lógico	1	
caafc	Aplicacion Fuente Complementaria para Captura	Lógico	1	
crafc	Aplicacion Fuente Complementaria para Critica	Lógico	1	
afc	afc	Lógico	1	
caob	Novedad Tecnica, Observaciones para Control calidad Captura	Lógico	1	
crob	Novedad Tecnica, Observaciones para Control calidad Captura	Lógico	1	
ob	ob	Lógico	1	
totalcalicri	Acumulado para Control calidad Critica	Numérico	3	0

obscalicri	Observaciones para Control Calidad Critica	Memo	4	
totalcalicap	Acumulado para Control calidad Captura	Numérico	3	0
obscalicap	Observaciones para Control Calidad Captura	Memo	4	
entra	Formato Cargas de Trabajo entran	Numérico	4	0
sale	Formato Cargas de Trabajo Sale	Numérico	4	0
pendientes	Formato Cargas de Trabajo Pendientes	Numérico	4	0
totalentra	Acumulados de los que Entran	Numérico	6	0
totalsale	Acumulados de los que Salen	Numérico	6	0
totalpte	Acumulados de los Pendientes	Numérico	6	0
fechaso	Fecha del Computador	Fecha	8	
fecharec	Fecha de captura en terreno	Date Time	8	
sianalisis	Indica que la fuente articulo se analizo si esta en verdadero .T.			
diaanalisis	Contiene el numero del dia en que se analizo la fuente articulo	Numérico	2	0
diaasupervisar	Contiene informacion del dia en que se marco la fuente articulo para tenerse en cuenta para ser supervisada	Numérico	2	0
supervisado	Indica que la informacion de este registro corresponde a la planilla de Supervision (que se va asupervisar)	Lógico	1	
obssupersado	Observaciones del por que se marca para ser supervisado	Memo	4	
fechasup	Indica la fecha cuando fue supervisada la fuente articulo	Date Time	8	
marca	Nos indica si esta marcado para ser eliminado	Lógico	1	
borra	Nos indica si esta marcado para ser eliminado	Lógico	1	
demas	Marca Logica	Lógico	1	
tippre	tippre	Carácter	1	
repres	repres	Carácter	1	

canpref	canpref	Numérico	7	0
pci	pci	Lógico	1	
diaarec	diaarec	Carácter	2	
recfur	recfur	Lógico	1	
fechaana	fechaana	Date Time	8	
fechaccc	fechaccc	Date Time	8	

☞ **modelos:**

Indices:

Nombre	Tipo	Expresión
pkmodelo	Principal	codfte+codart+codmod

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codfte		Carácter	20	
codart		Carácter	7	
codmod		Carácter	2	
anoinimod		Carácter	4	
anofinmod		Carácter	4	
nrobuses		Numérico	9	0
tarmedant		Numérico	12	2
tarmed		Numérico	12	2
ano		Carácter	4	
mes		Carácter	2	
marca		Lógico	1	

☞ **observa:** Contiene la informacion de las observaciones predefinidas clasificadas por Alza, Baja o Novedad Tecnica.

Indices:

Nombre	Tipo	Expresión
pkcodobs	Principal	codobs
NKCODOBS	Normal	codobs

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codobs	Contiene el codigo de la observacion estandarizada	Carácter	2	
observa		Memo	4	
alza	Marca logica que indica si pertene a algrupo de las alzas	Lógico	1	
baja	Marca logica que indica si pertene a algrupo de las bajas	Lógico	1	
novpe	Marca logica que indica si pertene a al grupo de novedades tecnicas Periodo de Espera	Lógico	1	
codnove	Clasifica las observaciones por Alza,Baja o Novedad Tecnica	Carácter	2	
marca	Marca Logica	Lógico	1	
borra	Maraca logica que indica si esta marcado para ser eliminado	Lógico	1	
demas	Marca demas para no tener problemas al modificar las vistas	Lógico	1	

☞ **pcimart**

Indices:

Nombre	Tipo	Expresión
pkcodpci	Principal	codpci
FKCODUM	Normal	codum

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codpci		Carácter	12	
codipc		Carácter	7	
codipc7		Carácter	7	
nomart		Carácter	44	
espec		Memo	4	
comipc		Lógico	1	
cant		Numérico	4	0

codum		Carácter	2	
marca		Lógico	1	
borra		Lógico	1	
demas				

☞ **pcirpart**

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codper		Numérico	1	0
codpart		Numérico	1	0
mes		Numérico	2	0
marca		Lógico	1	
borra		Lógico	1	
demas		Lógico	1	

☞ **pcium**

Indices:

Nombre	Tipo	Expresión
pkcodum	Principal	codum

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codum		Carácter	2	
nomum		Carácter	16	
abrev		Carácter	8	
otrum		Carácter	8	
marca		Lógico	1	
borra		Lógico	1	
demas		Lógico	1	

☞ **pondesgc**

Indices:

Nombre	Tipo	Expresión
pkpondera	Principal	coddpto+codmpio+codgri+codart
NKPONDERA	Normal	codgri+codart

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal

coddpto	Codigo Departamento	Carácter	2	
codmpio	Codigo Municipio	Carácter	3	
codart	Codigo Articulo	Carácter	7	
codgri	Codigo Grupo Ingreso	Carácter	1	
vlr_pon	Valor Ponderacion	Numérico	14	8
marca		Lógico	1	
borra		Lógico	1	
demas		Lógico	1	

☞ **procesos**

Indices:

Nombre	**Tipo**	**Expresión**
pkpondera	Principal	coddpto+codmpio+codgri+codart
NKPONDERA	Normal	codgri+codart

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
procod		Numérico	1	0
prousucod		Carácter	15	
prodir		Carácter	50	
pockettopc		Carácter	50	
pctopocket		Carácter	50	
bdinicial		Carácter	50	
proundimp		Carácter	15	
proundexp		Carácter	15	

☞ **rango**

Indices:

Nombre	**Tipo**	**Expresión**
NKRANGO	Normal	coddpto+ano+mes+codart+codfuente
PKRANGO	Principal	coddpto+ano+mes+codart+codfuente+codtarifa

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
coddpto		Carácter	2	

ano		Carácter	4	
mes		Carácter	2	
codfuente		Carácter	4	
codart		Carácter	7	
codtarifa		Carácter	2	
destarifa		Carácter	20	
ran_ini		Numérico	3	0
ran_fin		Numérico	3	0
unidad		Carácter	15	
rani		Numérico	3	0
marca		Lógico	1	

☞ **registro:** Contiene el registro de los ingresos de los Usuarios al aplicativo y adicionalmente a las pantallas que utiliza,lafecha en que lo hace, estolo hacemos para controlar si hay mas de un usuario utilizando el aplicativo y asi poder procesar algunas ordenes.

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
quien	Usuario que ingreso al aplicativo y adicionalmente a alguna pantalla	Carácter	20	
npantalla	Nombre de la Pantalla a la que ingresa el Usuario	Carácter	25	
fechar	Fecha en la que ingresa a la Pantalla	Fecha	8	
act_ina	Indica si esta aun utilizandoesta pantalla o no	Lógico	1	
borra	Marca que indica si esta marcado parra borrar	Lógico	1	
demas	Marca Logica	Lógico	1	

☞ **rutabd:** Contiene la Ruta de la Base de Datos que esta utilizando el aplicativo, esta la debe indicar el usuario puesto que puede estar en el mismo equipo del cliente o en un equipo servidor aparte, la utilizamos para Reindexar, Generar Envio, Copia de Seguridad,

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
rutabd	Ruta donde se encuentra ubicada fisicamente la Base de datos	Carácter	60	
quien	Quien asigno esta informacion	Carácter	20	

fecharuta	Fecha en que se indico esta informacion	Fecha	8	
act_ina	Indica si es activa o no la ruta en este registro, es decir, si es la actual	Lógico	1	

☞ **tarifa**

Indices:

Nombre	**Tipo**	**Expresión**
PKDESC	Normal	coddpto+ano+mes+codart+codfuente+codestrato+codtarifa
Tarifa	Normal	coddpto+ano+mes+codart+codfuente+codtarifa
ESTRATO	Normal	coddpto+ano+mes+codart+codfuente+codestrato
VARIAMES	Normal	coddpto+ano+mes+codart+codfuente
NKTARIFA	Normal	coddpto+ano+mes+codart+codfuente
Pktarifa	Normal	coddpto+ano+mes+codart+codfuente+codestrato+codtarifa

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
coddpto		Carácter	2	
ano		Carácter	4	
mes		Carácter	2	
codfuente		Carácter	4	
codart		Carácter	7	
codestrato		Carácter	1	
codtarifa		Carácter	2	
vlr_tarifa		Numérico	12	4
consum_tar		Numérico	6	0
vlr_consum		Numérico	12	4
vlr_ponde		Numérico	10	2
ran_ini		Numérico	3	0
ran_fin		Numérico	3	0
rani		Numérico	3	0
marca		Lógico	1	

☞ **tarifast**

Indices:

Nombre	**Tipo**	**Expresión**
Pktarifast	Principal	codfte+codart+codmod+codtar

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codfte		Carácter	20	
codart		Carácter	7	
codmod		Carácter	2	
codtar		Carácter	2	
destar		Carácter	15	
desde		Numérico	12	2
hasta		Numérico	12	2
ano		Carácter	4	
mes		Carácter	2	
marca		Lógico	1	

☞ **totaltarifa**

Indices:

Nombre	Tipo	Expresión
Pktotar	Principal	codfte+codart

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
codfte		Carácter	20	
codart		Carácter	7	
tarmedtota		Numérico	12	2
tarmedtot		Numérico	12	2
tarifapro		Numérico	12	2
dia_rige		Numérico	2	0
variames		Numérico	6	2
variamessg		Numérico	6	2
preciomes		Numérico	12	2
preciomess		Numérico	12	2
mes		Carácter	2	
ano		Carácter	4	
marca		Lógico	1	

☞ **variames**

Indices:

Nombre	Tipo	Expresión
pkvariames	Principal	coddpto+ano+mes+codfuente+codart+codgri

Campos:

Nombre	Descripción	Tipo	Ancho	Decimal
coddpto		Carácter	2	
ano		Carácter	4	
mes		Carácter	2	
codfuente		Carácter	4	
codart		Carácter	7	
codgri		Carácter	1	
precio		Numérico	12	2
variacion		Numérico	6	2
iva		Numérico	2	0
Marca		Lógico	1	
variarecal		Numérico	6	2
precioreca		Numérico	12	2
totusers	Contiene el total de Usuarios por Grupo Ingreso	Numérico	9	0

3.2 VISTA LOCALES IPCRED.DBC

☞ **Vcaptura**

```
SELECT DISTINCT V_ipctmvt.ano, V_ipctmvt.mes, V_ipctmvt.impctv,;
  V_ipctmvt.codfte, V_ipctmvt.decmvt, V_ipctmvt.diaarec;
 FROM ipcred!v_ipctmvt;
 WHERE V_ipctmvt.borra = .F.;
   AND VAL(V_ipctmvt.diaarec) > 0;
 ORDER BY V_ipctmvt.impctv
```

Sin Duplicados

3.3 VISTA REMOTAS IPCRED.DBC

☞ **mto_mvto**

```
SELECT DISTINCT Ipctmvt.codart, Ipctmvt.impctv, Ipcmart.nomart;
 FROM ipcmart Ipcmart, ipctmvt Ipctmvt;
 WHERE Ipctmvt.codart = Ipcmart.codart;
   AND (Ipctmvt.borra = .F.;
   AND Ipctmvt.codfte = ?ptrofte);
 ORDER BY Ipctmvt.codart
```

Sin Duplicados

☞ **vesp**

```
SELECT DISTINCT Ipcresp.codfte, Ipcresp.codart, Ipcresp.codesp,;
  Ipcresp.nomesp, Ipcresp.nombdet, Ipcresp.tipo, Ipcresp.valor,;
  Ipcresp.marca, Ipcresp.borra;
 FROM ipcresp Ipcresp;
 WHERE Ipcresp.codfte = ?vespfuente;
   AND Ipcresp.codart = ?vesparticulo;
 ORDER BY Ipcresp.codfte, Ipcresp.codesp
```

Sin Duplicados

☞ **vfap**

```
SELECT DISTINCT Ipcrfap.codfte, Ipcrfap.codart, Ipcrfap.decart,;
  Ipcrfap.panart, Ipcrfap.perfap, Ipcrfap.cantant,;
  Ipcrfap.codfte+Ipcrfap.codart AS exp_5, Ipcrfap.pci;
 FROM ipcrfap Ipcrfap;
 WHERE Ipcrfap.codfte = ?vfapcodfte;
   AND Ipcrfap.codart = ?vfapcodart;
   AND Ipcrfap.borra = .F.
```

Sin Duplicados

☞ **vfuente**

```
SELECT DISTINCT Ipcmfte.codfte, Ipcmfte.nombfte, Ipcmfte.direfte,;
  Ipcmfte.telfte, Ipcmfte.estrfte, Ipcmfte.zona;
 FROM ipcmfte Ipcmfte;
 WHERE Ipcmfte.codfte = ?nofte
```

Sin Duplicados

☞ **vfuentes**

```
SELECT DISTINCT Ipcrfap.codart, Ipcrfap.codfte, Ipcmart.nomart,;
  Ipcmfte.nombfte;
 FROM ipcrfap Ipcrfap, ipcmfte Ipcmfte, ipcmart Ipcmart;
 WHERE Ipcrfap.codart = Ipcmart.codart;
   AND Ipcrfap.codfte = Ipcmfte.codfte;
   AND (Ipcrfap.codart IN ("2210102","2220101","2220201","2220202","2220203","7320101");
   AND Ipcmfte.borra = .F.;
   AND Ipcrfap.borra = .F.);
 ORDER BY Ipcrfap.codart, Ipcrfap.codfte
```

Sin Duplicados

☞ **vfuentest**

```
SELECT Ipcrfap.codfte, Ipcrfap.codart, Ipcmart.nomart, Ipcmfte.nombfte,;
  Ipcrfap.panart, Ipcrfap.cantant;
 FROM ipcrfap Ipcrfap, ipcmfte Ipcmfte, ipcmart Ipcmart;
 WHERE Ipcrfap.codfte = Ipcmfte.codfte;
   AND Ipcrfap.codart = Ipcmart.codart;
   AND ({fn substring(Ipcrfap.codart,1,3)} = "721";
   AND Ipcrfap.borra = .F.);
 ORDER BY Ipcrfap.codfte, Ipcrfap.codart
```

☞ **vobs**

```
SELECT Ipcrobs.impctv, Ipcrobs.codfte, Ipcrobs.codobs, Ipcrobs.observa,;
  Ipcrobs.marca, Ipcrobs.borra;
 FROM ipcrobs Ipcrobs;
 WHERE Ipcrobs.codfte = ?NoFte;
   AND Ipcrobs.impctv = ?NoFormulario;
 ORDER BY Ipcrobs.impctv, Ipcrobs.codfte, Ipcrobs.codobs
```

☞ **vr_diaspda**

```
SELECT Ipcconfig.condatid, Ipcconfig.condatmin, Ipcconfig.condatmax,;
  Ipcconfig.condatmaxp;
 FROM ipcconfig Ipcconfig;
 ORDER BY Ipcconfig.condatid
```

☞ **vr_obsciu**

```
SELECT DISTINCT Observa.codobs, Observa.observa, Observa.alza,;
  Observa.baja, Observa.novpe, Observa.codnove;
 FROM observa Observa;
 WHERE Observa.codobs >= "61";
   AND Observa.codobs <= "70";
 ORDER BY Observa.codobs
```

Sin Duplicados

☞ **vr_obslist**

```
SELECT Observa.codobs, Observa.observa, Observa.alza, Observa.baja,;
  Observa.novpe, Observa.codnove;
 FROM observa Observa;
 WHERE Observa.borra = .F.;
   AND Observa.codnove = ?PTipVaria;
   OR (Observa.borra = .F.;
   AND Observa.codnove = "G";
   AND Observa.codobs NOT BETWEEN "70" AND "80");
 ORDER BY Observa.codobs
```

☞ **v_anafte**

```
SELECT Ipcmfte.nombfte, Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv,;
  Ipctmvt.panart, Ipctmvt.pactmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt,;
  Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.canrmvt,;
  Ipctmvt.obsmvt, Ipctmvt.decmvt, Ipctmvt.cantant, Ipctmvt.ano, Ipctmvt.mes,;
  Ipctmvt.pesart, Ipctmvt.plasup, Ipctmvt.placri, Ipctmvt.calicri,;
  Ipctmvt.sianalisis, Ipctmvt.diaanalisis, Ipctmvt.diaasupervisar,;
  Ipctmvt.zona, Ipctmvt.marca, Ipctmvt.supervisado, Ipctmvt.obssupersado,;
  Ipctmvt.fechasup, {fn substring(Ipctmvt.codfte,14,1)} AS grupo,;
  Ipctmvt.variacion, Ipctmvt.diacri, Ipctmvt.cumr, Ipctmvt.cpmr,;
  Ipctmvt.ciape, Ipctmvt.ciasi, Ipctmvt.ciacr, Ipctmvt.ciafn, Ipctmvt.cadi,;
  Ipctmvt.conas, Ipctmvt.cnae, Ipctmvt.totalplacri, Ipctmvt.obsplacri,;
  Ipctmvt.diacalicri, Ipctmvt.crea, Ipctmvt.crur, Ipctmvt.crpr,;
  Ipctmvt.crape, Ipctmvt.crasi, Ipctmvt.cracr, Ipctmvt.crafc, Ipctmvt.crob,;
  Ipctmvt.totalcalicri, Ipctmvt.crdgf, Ipctmvt.obscalicri, Ipctmvt.total,;
  Ipcmfte.mpiofte, Ipcmart.nomart, SUBSTR(Ipctmvt.codfte,15,2) AS clase,;
  IIF(VAL(Ipctmvt.variacion)#0,STR(ROUND(VAL(Ipctmvt.variacion),2),7,2),'       ') AS variacion1,;
  Ipctmvt.diaarec, Ipctmvt.fechaana, Ipctmvt.fechaccc, Ipctmvt.diasup,;
  Ipctmvt.sumr, Ipctmvt.spmr, Ipctmvt.siape, Ipctmvt.siasi, Ipctmvt.siacr,;
  Ipctmvt.siafn, Ipctmvt.sadi, Ipctmvt.sonas, Ipctmvt.snae,;
  Ipctmvt.totalplasup, Ipctmvt.obsplasup;
 FROM ipcmfte Ipcmfte, ipctmvt Ipctmvt, ipcmart Ipcmart;
 WHERE Ipctmvt.codfte = Ipcmfte.codfte;
   AND Ipcmart.codart = Ipctmvt.codart;
   AND (Ipctmvt.borra = .F.;
   AND Ipctmvt.zona IN
(?z_1,?z_2,?z_3,?z_4,?z_5,?z_6,?z_7,?z_8,?z_9,?z_10,?z_11,?z_12,?z_13,?z_14,?z_15,?z_16,
?z_17,?z_18,?z_19,?z_20,?z_21,?z_22,?z_23);
   AND Ipctmvt.codfte = ?panafte;
   AND ALLTRIM(STR(Ipctmvt.decmvt)) BETWEEN ?pdecmvt AND ?pdecmvt1;
   AND Ipctmvt.total BETWEEN ?pdiaanali1 AND ?pdiaanali2;
   AND VAL(Ipctmvt.diaarec) > 0);
 ORDER BY Ipctmvt.zona, Ipctmvt.codfte, Ipctmvt.codart
```

☞ **v_anaftemvt**

```
SELECT DISTINCT Ipctmvt.impctv, Ipctmvt.codfte, Ipcmfte.nombfte,;
  Ipcmfte.direfte, Ipcmfte.telfte, Ipctmvt.zona, Ipctmvt.diaarec;
 FROM ipcmfte Ipcmfte, ipctmvt Ipctmvt;
 WHERE Ipctmvt.codfte = Ipcmfte.codfte;
   AND (Ipctmvt.borra = .F.;
   AND Ipctmvt.zona IN
(?z_1,?z_2,?z_3,?z_4,?z_5,?z_6,?z_7,?z_8,?z_9,?z_10,?z_11,?z_12,?z_13,?z_14,?z_15,?z_16,
?z_17,?z_18,?z_19,?z_20,?z_21,?z_22,?z_23);
   AND ALLTRIM(STR(Ipctmvt.decmvt)) BETWEEN ?pdecmvt AND ?pdecmvt1;
   AND Ipctmvt.total BETWEEN ?pdiaanali1 AND ?pdiaanali2;
```

```
  AND VAL(Ipctmvt.diaarec) > 0);
 ORDER BY Ipcmfte.nombfte
```

Sin Duplicados

☞ **v_analicri**

```
SELECT Ipctmvt.impctv, Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.plasup,;
  Ipctmvt.placri, Ipctmvt.calicri, Ipctmvt.diaasupervisar,;
  Ipctmvt.supervisado, Ipctmvt.obssupersado, Ipctmvt.fechasup,;
  Ipctmvt.diacri, Ipctmvt.cumr, Ipctmvt.cpmr, Ipctmvt.ciape, Ipctmvt.ciasi,;
  Ipctmvt.ciacr, Ipctmvt.ciafn, Ipctmvt.cadi, Ipctmvt.conas, Ipctmvt.cnae,;
  Ipctmvt.totalplacri, Ipctmvt.obsplacri, Ipctmvt.diacalicri, Ipctmvt.crea,;
  Ipctmvt.crur, Ipctmvt.crpr, Ipctmvt.crape, Ipctmvt.crasi, Ipctmvt.cracr,;
  Ipctmvt.crafc, Ipctmvt.crob, Ipctmvt.totalcalicri, Ipctmvt.crdgf,;
  Ipctmvt.obscalicri, Ipctmvt.sianalisis, Ipctmvt.diaanalisis,;
  Ipctmvt.diaarec, Ipctmvt.fechaana, Ipctmvt.fechaccc;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.borra = .F.;
   AND Ipctmvt.codfte = ?pfteanacri;
   AND Ipctmvt.codart = ?partanacri;
   AND Ipctmvt.impctv = ?pimpctvana;
 ORDER BY Ipctmvt.impctv, Ipctmvt.codfte, Ipctmvt.codart
```

☞ **v_analisis**

```
SELECT Ipcmfte.nombfte, Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv,;
  Ipctmvt.panart, Ipctmvt.pactmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt,;
  Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.canrmvt,;
  Ipctmvt.obsmvt, Ipctmvt.decmvt, Ipctmvt.cantant, Ipctmvt.ano, Ipctmvt.mes,;
  Ipctmvt.pesart, Ipctmvt.plasup, Ipctmvt.placri, Ipctmvt.calicri,;
  Ipctmvt.sianalisis, Ipctmvt.diaanalisis, Ipctmvt.diaasupervisar,;
  Ipctmvt.zona, Ipctmvt.marca, Ipctmvt.supervisado, Ipctmvt.obssupersado,;
  Ipctmvt.fechasup, {fn substring(Ipctmvt.codfte,14,1)} AS grupo,;
  Ipctmvt.variacion, Ipctmvt.diacri, Ipctmvt.cumr, Ipctmvt.cpmr,;
  Ipctmvt.ciape, Ipctmvt.ciasi, Ipctmvt.ciacr, Ipctmvt.ciafn, Ipctmvt.cadi,;
  Ipctmvt.conas, Ipctmvt.cnae, Ipctmvt.totalplacri, Ipctmvt.obsplacri,;
  Ipctmvt.diacalicri, Ipctmvt.crea, Ipctmvt.crur, Ipctmvt.crpr,;
  Ipctmvt.crape, Ipctmvt.crasi, Ipctmvt.cracr, Ipctmvt.crafc, Ipctmvt.crob,;
  Ipctmvt.totalcalicri, Ipctmvt.crdgf, Ipctmvt.obscalicri, Ipctmvt.total,;
  Ipcmfte.mpiofte, SUBSTR(Ipctmvt.codfte,15,2) AS clase,;
  IIF(VAL(Ipctmvt.variacion)#0,STR(ROUND(VAL(Ipctmvt.variacion),2),7,2),'       ') AS variacion1,;
  Ipctmvt.diaarec, Ipctmvt.fechaana, Ipctmvt.fechaccc, Ipctmvt.diasup,;
  Ipctmvt.sumr, Ipctmvt.spmr, Ipctmvt.siape, Ipctmvt.siasi, Ipctmvt.siacr,;
  Ipctmvt.siafn, Ipctmvt.sadi, Ipctmvt.sonas, Ipctmvt.snae,;
  Ipctmvt.totalplasup, Ipctmvt.obsplasup;
 FROM ipcmfte Ipcmfte, ipctmvt Ipctmvt;
```

```
 WHERE Ipctmvt.codfte = Ipcmfte.codfte;
   AND (Ipctmvt.borra = .F.;
   AND Ipctmvt.zona IN
(?z_1,?z_2,?z_3,?z_4,?z_5,?z_6,?z_7,?z_8,?z_9,?z_10,?z_11,?z_12,?z_13,?z_14,?z_15,?z_16,
?z_17,?z_18,?z_19,?z_20,?z_21,?z_22,?z_23);
   AND Ipctmvt.codart = ?pcodart;
   AND ALLTRIM(STR(Ipctmvt.decmvt)) BETWEEN ?pdecmvt AND ?pdecmvt1;
   AND Ipctmvt.total BETWEEN ?pdiaanali1 AND ?pdiaanali2;
   AND VAL(Ipctmvt.diaarec) > 0);
 ORDER BY Ipctmvt.zona, Ipctmvt.codfte
```

☞ **v_analisis_ori**

```
SELECT Ipcmfte.nombfte, Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv,;
  Ipctmvt.panart, Ipctmvt.pactmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt,;
  Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.canrmvt,;
  Ipctmvt.obsmvt, Ipctmvt.decmvt, Ipctmvt.cantant, Ipctmvt.ano, Ipctmvt.mes,;
  Ipctmvt.pesart, Ipctmvt.plasup, Ipctmvt.placri, Ipctmvt.calicri,;
  Ipctmvt.sianalisis, Ipctmvt.diaanalisis, Ipctmvt.diaasupervisar,;
  Ipctmvt.zona, Ipctmvt.marca, Ipctmvt.supervisado, Ipctmvt.obssupersado,;
  Ipctmvt.fechasup, {fn substring(Ipctmvt.codfte,14,1)} AS grupo,;
  IIF(VAL(Ipctmvt.variacion)#0,Ipctmvt.variacion,'       ') AS variacion,;
  Ipctmvt.diacri, Ipctmvt.cumr, Ipctmvt.cpmr, Ipctmvt.ciape, Ipctmvt.ciasi,;
  Ipctmvt.ciacr, Ipctmvt.ciafn, Ipctmvt.cadi, Ipctmvt.conas, Ipctmvt.cnae,;
  Ipctmvt.totalplacri, Ipctmvt.obsplacri, Ipctmvt.diacalicri, Ipctmvt.crea,;
  Ipctmvt.crur, Ipctmvt.crpr, Ipctmvt.crape, Ipctmvt.crasi, Ipctmvt.cracr,;
  Ipctmvt.crafc, Ipctmvt.crob, Ipctmvt.totalcalicri, Ipctmvt.crdgf,;
  Ipctmvt.obscalicri;
 FROM ipcmfte Ipcmfte, ipctmvt Ipctmvt;
 WHERE Ipctmvt.codfte = Ipcmfte.codfte;
   AND (Ipctmvt.borra = .F.;
   AND Ipctmvt.codart = ?pcodart;
   AND Ipctmvt.zona IN (?pzonaanalisis);
   AND ALLTRIM(STR(Ipctmvt.decmvt)) BETWEEN ?pdecmvt AND ?pdecmvt1);
 ORDER BY Ipctmvt.zona, Ipctmvt.codfte
```

☞ **v_anohist**

```
SELECT DISTINCT Historia.ano;
 FROM historia Historia;
 WHERE Historia.borra = .F.;
   AND Historia.ano >= 1998;
 ORDER BY Historia.ano
```

Sin Duplicados

☞ **v_artiesp**

```
SELECT DISTINCT Ipcmart.codart, Ipcmart.nomart;
 FROM ipcmart Ipcmart, ipcrfap Ipcrfap;
 WHERE Ipcrfap.codart = Ipcmart.codart;
   AND (Ipcrfap.codfte = ?orespafte;
   AND Ipcrfap.borra = .F.);
 ORDER BY Ipcmart.nomart
```

Sin Duplicados

- **v_artixdias**

```
SELECT Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.ruta, Ipctmvt.ctvoruta,;
  Ipctmvt.diaarec;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.codfte = ?pftedia;
 ORDER BY Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.diaarec
```

- **v_auxiliar**

```
SELECT *;
 FROM auxiliar Auxiliar
```

- **v_caliobs**

```
SELECT Ipcrobs.codfte, Ipcrobs.codobs, Ipcrobs.observa;
 FROM ipcrobs Ipcrobs;
 WHERE Ipcrobs.impctv = ?pcalictvo;
   AND Ipcrobs.codfte = ?vcalifte5;
   AND Ipcrobs.codobs = ?pcalicodobs
```

- **v_cargadec_c1**

```
SELECT ALLTRIM(Ipcrfunc.codfunci) AS codfunci, Ipctmvt.total,;
  COUNT(Ipctmvt.total) AS entra, Ipcrfunc.zonafunci, Ipcrfunc.codcargo;
 FROM ipctmvt Ipctmvt, ipcrfunc Ipcrfunc;
 WHERE Ipcrfunc.codcargo = 'C';
   AND Ipcrfunc.borra = .F.;
   AND Ipctmvt.borra = .F.;
   AND Ipctmvt.ano = ?pcargaano;
   AND Ipctmvt.mes = ?pcargames;
   AND Ipctmvt.decmvt = ?pcargadec;
   AND Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
   AND Ipctmvt.total > 0;
   AND IIF((Ipctmvt.pactmvt#0.OR.Ipctmvt.pesmvt.OR.Ipctmvt.obsmvt=="88"),.T.,.F.) = .T.;
 GROUP BY Ipcrfunc.codfunci, Ipctmvt.total;
 ORDER BY 1, Ipctmvt.total
```

☞ **v_cargadec_c2**

```
SELECT ALLTRIM(Ipcrfunc.codfunci) AS codfunci,;
  COUNT(Ipctmvt.sianalisis) AS sale, Ipctmvt.diaanalisis AS dia,;
  Ipcrfunc.zonafunci, Ipcrfunc.codcargo;
 FROM ipctmvt Ipctmvt, ipcrfunc Ipcrfunc;
 WHERE Ipctmvt.borra = .F.;
   AND Ipcrfunc.borra = .F.;
   AND Ipctmvt.ano = ?pcargaano;
   AND Ipctmvt.mes = ?pcargames;
   AND Ipctmvt.decmvt = ?pcargadec;
   AND Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
   AND Ipctmvt.sianalisis = .T.;
   AND Ipcrfunc.codcargo = 'C';
   AND Ipctmvt.diaanalisis > 0;
   AND Ipctmvt.total > 0;
   AND IIF((Ipctmvt.pactmvt#0.OR.Ipctmvt.pesmvt.OR.Ipctmvt.obsmvt=="88"),.T.,.F.) = .T.;
 GROUP BY Ipcrfunc.codfunci, Ipctmvt.diaanalisis;
 ORDER BY 1, Ipctmvt.diaanalisis
```

☞ **v_cargadec_r1**

```
SELECT ALLTRIM(Ipcrfunc.codfunci) AS codfunci, Ipctmvt.diaarec,;
  COUNT(Ipctmvt.diaarec) AS entra, Ipcrfunc.zonafunci, Ipcrfunc.codcargo;
 FROM ipctmvt Ipctmvt, ipcrfunc Ipcrfunc;
 WHERE Ipctmvt.borra = .F.;
   AND Ipcrfunc.borra = .F.;
   AND Ipctmvt.ano = ?pcargaano;
   AND Ipctmvt.mes = ?pcargames;
   AND Ipctmvt.decmvt = ?pcargadec;
   AND Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
   AND Ipctmvt.diaarec > "0";
   AND Ipcrfunc.codcargo = 'R';
 GROUP BY Ipcrfunc.codfunci, Ipctmvt.diaarec;
 ORDER BY 1, Ipctmvt.diaarec
```

☞ **v_cargadec_r2**

```
SELECT ALLTRIM(Ipcrfunc.codfunci) AS codfunci,;
  COUNT(Ipctmvt.total) AS sale, Ipctmvt.total AS dia, Ipcrfunc.zonafunci,;
  Ipcrfunc.codcargo;
 FROM ipctmvt Ipctmvt, ipcrfunc Ipcrfunc;
 WHERE Ipctmvt.borra = .F.;
   AND Ipcrfunc.borra = .F.;
```

```
  AND Ipctmvt.ano = ?pcargaano;
  AND Ipctmvt.mes = ?pcargames;
  AND Ipctmvt.decmvt = ?pcargadec;
  AND Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
  AND Ipcrfunc.codcargo = 'R';
  AND Ipctmvt.total > 0;
  AND IIF((Ipctmvt.pactmvt#0.OR.Ipctmvt.pesmvt.OR.Ipctmvt.obsmvt=="88"),.T.,.F.) = .T.;
GROUP BY Ipcrfunc.codfunci, Ipctmvt.total;
ORDER BY 1, Ipctmvt.total
```

☞ **v_cargadec_s1**

```
SELECT Ipcrfunc.codfunci, Ipctmvt.diaasupervisar, COUNT(*) AS entra,;
 Ipcrfunc.zonafunci, Ipcrfunc.codcargo;
FROM ipctmvt Ipctmvt, ipcrfunc Ipcrfunc;
WHERE Ipctmvt.borra = .F.;
  AND Ipcrfunc.borra = .F.;
  AND Ipctmvt.ano = ?pcargaano;
  AND Ipctmvt.mes = ?pcargames;
  AND Ipctmvt.decmvt = ?pcargadec;
  AND Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
  AND Ipctmvt.supervisado = .T.;
  AND Ipcrfunc.codcargo = 'S';
  AND Ipctmvt.total > 0;
  AND IIF((Ipctmvt.pactmvt#0.OR.Ipctmvt.pesmvt.OR.Ipctmvt.obsmvt=="88"),.T.,.F.) = .T.;
GROUP BY Ipcrfunc.codfunci, Ipctmvt.diaasupervisar;
ORDER BY Ipcrfunc.codcargo, Ipcrfunc.codfunci,;
 Ipctmvt.diaasupervisar
```

☞ **v_cargadec_s2**

```
SELECT Ipcrfunc.codfunci, DAY(Ipctmvt.movdiasup) AS dia,;
 COUNT(*) AS sale, Ipcrfunc.zonafunci, Ipcrfunc.codcargo;
FROM ipctmvt Ipctmvt, ipcrfunc Ipcrfunc;
WHERE Ipctmvt.borra = .F.;
  AND Ipcrfunc.borra = .F.;
  AND Ipctmvt.ano = ?pcargaano;
  AND Ipctmvt.mes = ?pcargames;
  AND Ipctmvt.decmvt = ?pcargadec;
  AND Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
  AND Ipctmvt.plasup = .T.;
  AND Ipcrfunc.codcargo = 'S';
  AND Ipctmvt.total > 0;
  AND IIF((Ipctmvt.pactmvt#0.OR.Ipctmvt.pesmvt.OR.Ipctmvt.obsmvt=="88"),.T.,.F.) = .T.;
GROUP BY Ipcrfunc.codfunci, 2;
ORDER BY Ipcrfunc.codcargo, Ipcrfunc.codfunci
```

☞ **v_cargames_c1**

```
SELECT Ipcrfunc.codfunci, Ipctmvt.total, COUNT(*) AS entra,;
  Ipcrfunc.zonafunci, Ipcrfunc.codcargo;
 FROM ipctmvt Ipctmvt, ipcrfunc Ipcrfunc;
 WHERE Ipctmvt.borra = .F.;
   AND Ipcrfunc.borra = .F.;
   AND Ipctmvt.ano = ?pcargaano;
   AND Ipctmvt.mes = ?pcargames;
   AND Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
   AND Ipctmvt.total > 0;
   AND Ipcrfunc.codcargo = 'C';
   AND IIF((Ipctmvt.pactmvt#0.OR.Ipctmvt.pesmvt.OR.Ipctmvt.obsmvt=="88"),.T.,.F.) = .T.;
 GROUP BY Ipcrfunc.codfunci, Ipctmvt.total;
 ORDER BY Ipcrfunc.codcargo, Ipcrfunc.codfunci, Ipctmvt.total
```

☞ **v_cargames_c2**

```
SELECT Ipcrfunc.codfunci, Ipctmvt.diaanalisis AS dia, COUNT(*) AS sale,;
  Ipcrfunc.zonafunci, Ipcrfunc.codcargo;
 FROM ipctmvt Ipctmvt, ipcrfunc Ipcrfunc;
 WHERE Ipctmvt.borra = .F.;
   AND Ipcrfunc.borra = .F.;
   AND Ipctmvt.ano = ?pcargaano;
   AND Ipctmvt.mes = ?pcargames;
   AND Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
   AND Ipctmvt.sianalisis = .T.;
   AND Ipcrfunc.codcargo = 'C';
   AND Ipctmvt.diaanalisis > 0;
   AND Ipctmvt.total > 0;
   AND IIF((Ipctmvt.pactmvt#0.OR.Ipctmvt.pesmvt.OR.Ipctmvt.obsmvt=="88"),.T.,.F.) = .T.;
 GROUP BY Ipcrfunc.codfunci, Ipctmvt.diaanalisis;
 ORDER BY Ipcrfunc.codcargo, Ipcrfunc.codfunci, Ipctmvt.diaanalisis
```

☞ **v_cargames_r1**

```
SELECT ALLTRIM(Ipcrfunc.codfunci) AS codfunci, Ipctmvt.diaarec,;
  COUNT(Ipctmvt.diaarec) AS entra, Ipcrfunc.zonafunci, Ipcrfunc.codcargo;
 FROM ipctmvt Ipctmvt, ipcrfunc Ipcrfunc;
 WHERE Ipctmvt.borra = .F.;
   AND Ipcrfunc.borra = .F.;
   AND Ipctmvt.ano = ?pcargaano;
   AND Ipctmvt.mes = ?pcargames;
   AND Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
   AND Ipctmvt.diaarec > "0";
   AND Ipcrfunc.codcargo = 'R';
```

```
GROUP BY Ipcrfunc.codfunci, Ipctmvt.diaarec;
ORDER BY 1, Ipctmvt.diaarec
```

☞ **v_cargames_r2**

```
SELECT ALLTRIM(Ipcrfunc.codfunci) AS codfunci,;
  COUNT(Ipctmvt.total) AS sale, Ipctmvt.total AS dia, Ipcrfunc.zonafunci,;
  Ipcrfunc.codcargo;
 FROM ipctmvt Ipctmvt, ipcrfunc Ipcrfunc;
 WHERE Ipctmvt.borra = .F.;
   AND Ipcrfunc.borra = .F.;
   AND Ipctmvt.ano = ?pcargaano;
   AND Ipctmvt.mes = ?pcargames;
   AND Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
   AND Ipcrfunc.codcargo = 'R';
   AND Ipctmvt.total > 0;
   AND IIF((Ipctmvt.pactmvt#0.OR.Ipctmvt.pesmvt.OR.Ipctmvt.obsmvt=="88"),.T.,.F.) = .T.;
 GROUP BY Ipcrfunc.codfunci, Ipctmvt.total;
 ORDER BY 1, Ipctmvt.total
```

☞ **v_cargames_s1**

```
SELECT Ipcrfunc.codfunci, Ipctmvt.diaasupervisar, COUNT(*) AS entra,;
  Ipcrfunc.zonafunci, Ipcrfunc.codcargo;
 FROM ipctmvt Ipctmvt, ipcrfunc Ipcrfunc;
 WHERE Ipctmvt.borra = .F.;
   AND Ipcrfunc.borra = .F.;
   AND Ipctmvt.ano = ?pcargaano;
   AND Ipctmvt.mes = ?pcargames;
   AND Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
   AND Ipctmvt.supervisado = .T.;
   AND Ipcrfunc.codcargo = 'S';
   AND Ipctmvt.total > 0;
   AND IIF((Ipctmvt.pactmvt#0.OR.Ipctmvt.pesmvt.OR.Ipctmvt.obsmvt=="88"),.T.,.F.) = .T.;
 GROUP BY Ipcrfunc.codfunci, Ipctmvt.diaasupervisar;
 ORDER BY Ipcrfunc.codcargo, Ipcrfunc.codfunci,;
  Ipctmvt.diaasupervisar
```

☞ **v_cargames_s2**

```
SELECT Ipcrfunc.codfunci, DAY(Ipctmvt.movdiasup) AS dia,;
  COUNT(*) AS sale, Ipcrfunc.zonafunci, Ipcrfunc.codcargo;
 FROM ipctmvt Ipctmvt, ipcrfunc Ipcrfunc;
 WHERE Ipctmvt.borra = .F.;
   AND Ipcrfunc.borra = .F.;
   AND Ipctmvt.ano = ?pcargaano;
   AND Ipctmvt.mes = ?pcargames;
   AND Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
```

```
  AND Ipctmvt.plasup = .T.;
  AND Ipcrfunc.codcargo = 'S';
  AND Ipctmvt.total > 0;
  AND IIF((Ipctmvt.pactmvt#0.OR.Ipctmvt.pesmvt.OR.Ipctmvt.obsmvt=="88"),.T.,.F.) = .T.;
 GROUP BY Ipcrfunc.codfunci, 2;
 ORDER BY Ipcrfunc.codcargo, Ipcrfunc.codfunci
```

☞ **v_ctacalicrixzona**

```
SELECT DISTINCT CNT(1) AS ctaftes, Ipcrfunc.codcargo,;
  Ipcrfunc.codfunci;
 FROM ipctmvt Ipctmvt, ipcrfunc Ipcrfunc;
 WHERE Ipctmvt.borra = .F.;
  AND Ipctmvt.calicri = .T.;
  AND Ipcrfunc.codcargo = 'C';
  AND Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
 GROUP BY Ipcrfunc.codcargo, Ipcrfunc.codfunci
```

Sin Duplicados

☞ **v_ctaplacri**

```
SELECT Ipctmvt.decmvt, COUNT(Ipctmvt.codart) AS ctaftes;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.borra = .F.;
  AND Ipctmvt.placri = .T.;
 GROUP BY Ipctmvt.decmvt
```

☞ **v_ctaplacrixzona**

```
SELECT CNT(1) AS ctaftes, Ipctmvt.zona;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.borra = .F.;
  AND Ipctmvt.placri = .T.;
 GROUP BY Ipctmvt.zona;
 ORDER BY Ipctmvt.zona
```

☞ **v_ctaplasup**

```
SELECT Ipctmvt.decmvt, COUNT(Ipctmvt.codart) AS ctaftes;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.borra = .F.;
  AND Ipctmvt.supervisado = .T.;
```

```
GROUP BY Ipctmvt.decmvt
```

☞ **v_ctaplasupxzona**

```
SELECT CNT(1) AS ctaftes, Ipctmvt.zona;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.borra = .F.;
   AND Ipctmvt.supervisado = .T.;
 GROUP BY Ipctmvt.zona;
 ORDER BY Ipctmvt.zona
```

☞ **v_ctvcaptu**

```
SELECT DISTINCT Ipctmvt.impctv, Ipctmvt.codfte, Ipcmfte.nombfte,;
  Ipcmfte.direfte, Ipctmvt.codart, Ipcmart.nomart,;
  ALLTRIM(STR(Ipcmart.canbart))+ALLTRIM(Ipcmart.uniart) AS unidadbase,;
  Ipctmvt.panart, Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.crfmvt,;
  Ipctmvt.fcommvt, Ipctmvt.cmvmvt, Ipctmvt.canrmvt, Ipctmvt.pactmvt,;
  Ipctmvt.variacion, Ipcmart.canbart, Ipctmvt.decmvt, Ipcmfte.zona,;
  Ipctmvt.cantant;
 FROM ipctmvt Ipctmvt, ipcmfte Ipcmfte, ipcmart Ipcmart;
 WHERE Ipctmvt.codart = Ipcmart.codart;
   AND Ipctmvt.codfte = Ipcmfte.codfte;
   AND (Ipctmvt.borra = .F.;
   AND Ipctmvt.impctv = ?ctvcaptu);
 ORDER BY Ipcmfte.zona, Ipctmvt.impctv, Ipctmvt.codfte, Ipctmvt.codart
```

Sin Duplicados

☞ **v_ctvos_historia**

```
SELECT DISTINCT VAL(SUBSTR(Historia.codfte,17,4)) AS ctvo;
 FROM historia Historia;
 ORDER BY 1
```

Sin Duplicados

☞ **v_ctvos_histpre**

```
SELECT DISTINCT VAL(SUBSTR(Histpre.codfte,17,4)) AS ctvo;
 FROM histpre Histpre;
 ORDER BY 1
```

Sin Duplicados

☞ **v_ctvos_ipcmfte**

```
SELECT DISTINCT VAL(SUBSTR(Ipcmfte.codfte,17,4)) AS ctvo;
 FROM ipcmfte Ipcmfte;
 ORDER BY 1
```

Sin Duplicados

☞ **v_ctvos_no_asignados**

```
SELECT Histpre.codfte, Histpre.codart, Histpre.decmvt, Histpre.mesmvt,;
  Histpre.anomvt, Histpre.pesart, Histpre.panart, Histpre.cantant,;
  Histpre.pactmvt, Histpre.canrmvt, Histpre.pesmvt, Histpre.crfmvt,;
  Histpre.sinmmvt, Histpre.fcommvt, Histpre.marca, Histpre.borra;
 FROM histpre Histpre
```

☞ **v_deccaptu**

```
SELECT DISTINCT Ipctmvt.zona, Ipctmvt.impctv, Ipctmvt.codfte,;
  Ipcmfte.nombfte, Ipcmfte.direfte, Ipctmvt.codart, Ipcmart.nomart,;
  ALLTRIM(STR(Ipcmart.canbart))+ALLTRIM(Ipcmart.uniart) AS unidadbase,;
  Ipctmvt.panart, Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.crfmvt,;
  Ipctmvt.fcommvt, Ipctmvt.cmvmvt, Ipctmvt.canrmvt, Ipctmvt.pactmvt,;
  Ipctmvt.variacion, Ipcmart.canbart, Ipctmvt.decmvt, Ipctmvt.cantant;
 FROM ipctmvt Ipctmvt, ipcmfte Ipcmfte, ipcmart Ipcmart;
 WHERE Ipctmvt.codart = Ipcmart.codart;
   AND Ipctmvt.codfte = Ipcmfte.codfte;
   AND (Ipctmvt.borra = .F.;
   AND Ipctmvt.decmvt = ?deccaptu;
   AND Ipctmvt.zona IN
(?z_1,?z_2,?z_3,?z_4,?z_5,?z_6,?z_7,?z_8,?z_9,?z_10,?z_11,?z_12,?z_13,?z_14,?z_15,?z_16,
?z_17,?z_18,?z_19,?z_20,?z_21,?z_22,?z_23));
 ORDER BY Ipctmvt.zona, Ipctmvt.impctv, Ipctmvt.codfte, Ipctmvt.codart
```

Sin Duplicados

☞ **v_decdispon**

```
SELECT DISTINCT Ipctmvt.decmvt;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.decmvt > 0;
 ORDER BY Ipctmvt.decmvt
```

Sin Duplicados

☞ **v_decespmvt**

```
SELECT Ipcresp.codfte, Ipcresp.codart, Ipcresp.codesp, Ipcresp.nomesp,;
  Ipcresp.nombdet, Ipcresp.nombdet1, Ipcresp.borra;
 FROM  {oj    ;
   ON  Ipctmvt.codfte+Ipctmvt.codart = Ipcresp.codfte+Ipcresp.codart };
 WHERE Ipctmvt.borra = .F.;
   AND Ipctmvt.decmvt = ?pdeccaptu
```

☞ **v_decplarec**

```
SELECT SUM(Ipcdias.ar) AS cacaptu,;
  STR((SUM(Ipcdias.ar)/SUM(Ipcdias.ca))*100,6,2) AS porcentdec,;
  Ipcdias.decplarec;
 FROM ipcdias Ipcdias;
 WHERE Ipcdias.borra = .F.;
   AND Ipcdias.nrodia IS NOT NULL;
   AND Ipcdias.zonafunci = ?pzonafunci1;
   AND Ipcdias.anoplarec = ?panoplarec;
   AND Ipcdias.mesplarec = ?pmesplarec;
   AND Ipcdias.decplarec = ?pdecplarec;
 GROUP BY Ipcdias.decplarec
```

☞ **v_diamvt**

```
SELECT DISTINCT Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv,;
  Ipctmvt.panart, Ipctmvt.pactmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt,;
  Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.canrmvt,;
  Ipctmvt.obsmvt, Ipctmvt.decmvt, Ipctmvt.total, Ipctmvt.cantant,;
  Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.variacion, Ipctmvt.marca;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.total = ?diamvt;
   AND Ipctmvt.borra = .F.;
 ORDER BY Ipctmvt.impctv, Ipctmvt.codart
```

Sin Duplicados

☞ **v_diaplarec**

```
SELECT DISTINCT SUM(Ipcdias.ar) AS cacaptu,;
  STR((SUM(Ipcdias.ar)/SUM(Ipcdias.ca))*100,6,2) AS porcentdia,;
  Ipcdias.nrodia;
 FROM ipcdias Ipcdias;
 WHERE Ipcdias.borra = .F.;
   AND Ipcdias.nrodia IS NOT NULL;
   AND Ipcdias.zonafunci = ?pzonafunci1;
```

```
  AND Ipcdias.anoplarec = ?panoplarec;
  AND Ipcdias.mesplarec = ?pmesplarec;
  AND Ipcdias.decplarec = ?pdecplarec;
  AND Ipcdias.nrodia = ?pdiaplarec;
GROUP BY Ipcdias.nrodia
```

Sin Duplicados

☞ **v_diaporcent1**

```
SELECT Ipcdias.nrodia, SUM(Ipcdias.ar) AS ar, Ipcdias.marca,;
 Ipcdias.borra;
FROM ipcdias Ipcdias;
WHERE Ipcdias.borra = .F.;
  AND Ipcdias.nrodia IS NOT NULL;
  AND Ipcdias.zonafunci = ?pzonafunci1;
  AND Ipcdias.anoplarec = ?panoplarec;
  AND Ipcdias.mesplarec = ?pmesplarec;
  AND Ipcdias.decplarec = ?pdecplarec;
GROUP BY Ipcdias.nrodia;
ORDER BY Ipcdias.nrodia
```

☞ **v_diaporcent2**

```
SELECT MIN(Ipcdias.nrodia) AS nrodia, Ipcdias.codfte, Ipcdias.ca,;
 Ipcdias.marca, Ipcdias.borra;
FROM ipcdias Ipcdias;
WHERE Ipcdias.borra = .F.;
  AND Ipcdias.nrodia IS NOT NULL;
  AND Ipcdias.zonafunci = ?pzonafunci1;
  AND Ipcdias.anoplarec = ?panoplarec;
  AND Ipcdias.mesplarec = ?pmesplarec;
  AND Ipcdias.decplarec = ?pdecplarec;
GROUP BY Ipcdias.codfte;
ORDER BY Ipcdias.codfte
```

☞ **v_diascapxfte**

```
SELECT Ipctmvt.codfte, Ipctmvt.total, COUNT(Ipctmvt.codart) AS ac;
FROM ipctmvt Ipctmvt;
WHERE Ipctmvt.decmvt = ?diasdecada;
  AND Ipctmvt.codfte = ?pftedia;
  AND Ipctmvt.borra = .F.;
  AND Ipctmvt.total > 0;
  AND IIF((Ipctmvt.pactmvt#0.OR.Ipctmvt.pesmvt.OR.Ipctmvt.obsmvt=="88"),.T.,.F.) = .T.;
GROUP BY Ipctmvt.codfte, Ipctmvt.total;
```

```
ORDER BY Ipctmvt.codfte, Ipctmvt.total
```

☞ **v_diasentra**

```
SELECT Ipctmvt.total, COUNT(Ipctmvt.codfte) AS entra;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.borra = .F.;
   AND Ipctmvt.supervisado = .T.;
 GROUP BY Ipctmvt.total
```

☞ **v_diasnocapxfte**

```
SELECT Ipctmvt.decmvt, Ipctmvt.codfte, COUNT(Ipctmvt.codart) AS axc;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.decmvt = ?diasdecada;
   AND Ipctmvt.codfte = ?pftedia;
   AND Ipctmvt.borra = .F.;
 GROUP BY Ipctmvt.decmvt, Ipctmvt.codfte;
 ORDER BY Ipctmvt.decmvt, Ipctmvt.codfte
```

☞ **v_diasuperv**

```
SELECT DISTINCT Ipctmvt.movdiasup;
 FROM ipctmvt Ipctmvt;
 WHERE PADL(SUBSTR(ALLTRIM(DTOC(Ipctmvt.movdiasup)),4,2),2,'0') > "0";
 ORDER BY Ipctmvt.movdiasup
```

Sin Duplicados

☞ **v_dias_analisis**

```
SELECT DISTINCT Ipctmvt.diaanalisis;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.diaanalisis <> 0;
   AND Ipctmvt.borra = .F.;
 ORDER BY Ipctmvt.diaanalisis
```

☞ **v_dias_a_supervisar**

```
SELECT DISTINCT Ipctmvt.diaasupervisar;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.diaasupervisar <> 0;
   AND Ipctmvt.supervisado = .T.;
   AND Ipctmvt.borra = .F.;
 ORDER BY Ipctmvt.diaasupervisar
```

Sin Duplicados

☞ **v_dia_analino**

```
SELECT DISTINCT Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv,;
  Ipctmvt.panart, Ipctmvt.pactmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt,;
  Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.canrmvt,;
  Ipctmvt.obsmvt, Ipctmvt.decmvt, Ipctmvt.total, Ipctmvt.cantant,;
  Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.variacion, Ipctmvt.marca,;
  Ipctmvt.diaanalisis, Ipctmvt.sianalisis;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.total = ?pdia_analisis;
   AND Ipctmvt.sianalisis = .F.;
   AND Ipctmvt.borra = .F.;
 ORDER BY Ipctmvt.impctv, Ipctmvt.codart
Sin Duplicados
```

☞ **v_dia_analisis**

```
SELECT DISTINCT Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv,;
  Ipctmvt.panart, Ipctmvt.pactmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt,;
  Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.canrmvt,;
  Ipctmvt.obsmvt, Ipctmvt.decmvt, Ipctmvt.total, Ipctmvt.cantant,;
  Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.variacion, Ipctmvt.marca,;
  Ipctmvt.diaanalisis;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.diaanalisis = ?pdia_analisis;
   AND Ipctmvt.borra = .F.;
 ORDER BY Ipctmvt.impctv, Ipctmvt.codart

Sin Duplicados
```

☞ **v_dia_a_supervisar**

```
SELECT DISTINCT Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv,;
  Ipctmvt.panart, Ipctmvt.pactmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt,;
  Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.canrmvt,;
  Ipctmvt.obsmvt, Ipctmvt.decmvt, Ipctmvt.total, Ipctmvt.cantant,;
  Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.variacion, Ipctmvt.marca,;
  Ipctmvt.diaanalisis;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.diaasupervisar = ?pdia_asupervisar;
   AND Ipctmvt.borra = .F.;
 ORDER BY Ipctmvt.impctv, Ipctmvt.codart

Sin Duplicados
```

☞ **v_espbase**

```
SELECT DISTINCT Ipcresp.codesp, Ipcresp.nomesp, Ipcresp.nombdet,;
  Ipcresp.nombdet1, Ipcresp.borra;
 FROM ipcresp Ipcresp;
 WHERE Ipcresp.codart = ?respart;
 GROUP BY Ipcresp.codesp;
 ORDER BY Ipcresp.codesp
```

Sin Duplicados

- **v_esptemp**

```
SELECT DISTINCT Esptemp.codfte, Esptemp.codart, Esptemp.codesp,;
  Esptemp.nomesp, Esptemp.nombdet, Esptemp.impctv, Esptemp.ano, Esptemp.mes,;
  Esptemp.decmvt, Esptemp.marca, Esptemp.borra;
 FROM esptemp Esptemp;
 WHERE Esptemp.borra = .F.;
 ORDER BY Esptemp.ano, Esptemp.mes, Esptemp.decmvt, Esptemp.impctv,;
  Esptemp.codart, Esptemp.codesp
```

Sin Duplicados

- **v_estrato**

```
SELECT DISTINCT *;
 FROM estrato Estrato;
 WHERE Estrato.ano = ?mAno;
   AND Estrato.mes = ?mMes;
   AND Estrato.codart = ?mArticulo;
   AND Estrato.codfuente = ?mFuente;
 ORDER BY Estrato.coddpto, Estrato.ano, Estrato.mes, Estrato.codart,;
  Estrato.codfuente
```

Sin Duplicados

- **v_estratoant**

```
SELECT DISTINCT *;
 FROM estrato Estrato;
 WHERE Estrato.ano = ?mAnoAnt;
   AND Estrato.mes = ?mMesAnt;
   AND Estrato.codart = ?mArticulo;
   AND Estrato.codfuente = ?mFuente;
 ORDER BY Estrato.coddpto, Estrato.ano, Estrato.mes, Estrato.codart,;
  Estrato.codfuente, Estrato.codestrato
```

☞ **v_f01acure**

```
* Acumulado diario de recolecccion
SELECT Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt, Ipctmvt.zona,;
  Ipctmvt.total AS dia, COUNT(Ipctmvt.codart) AS ar;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.ano = ?panoh;
   AND Ipctmvt.mes = ?pmesh;
   AND Ipctmvt.decmvt = ?pdech;
   AND Ipctmvt.zona = ?pzonafunci1;
   AND Ipctmvt.borra = .F.;
   AND Ipctmvt.total > 0;
   AND IIF((Ipctmvt.pactmvt#0.OR.Ipctmvt.pesmvt.OR.Ipctmvt.obsmvt=="88"),.T.,.F.) = .T.;
 GROUP BY Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt, Ipctmvt.zona,;
  Ipctmvt.total;
 ORDER BY Ipctmvt.total
```

☞ **v_f01ca**

```
* Acumulado diario de recolecccion
SELECT Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.zona,;
  COUNT(Ipctmvt.codart) AS ca;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.borra = .F.;
   AND Ipctmvt.ano = ?panoplarec;
   AND Ipctmvt.mes = ?pmesplarec;
   AND Ipctmvt.zona = ?pzonafunci1;
 GROUP BY Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.zona
```

☞ **v_fappa**

```
SELECT Ipcrfap.codfte, Ipcrfap.codart, Ipcrfap.decart, Ipcrfap.panart,;
  Ipcrfap.furart, Ipcrfap.perfap, Ipcrfap.cantant, Ipcrfap.pesart,;
  Ipcrfap.fcommvt, Ipcrfap.marca, Ipcrfap.borra, Ipcrfap.pci;
 FROM ipcrfap Ipcrfap;
 WHERE Ipcrfap.codfte = ?fappafte;
 ORDER BY Ipcrfap.codfte, Ipcrfap.codart
```

☞ **v_fappart**

```
SELECT Ipcrfap.codfte, Ipcrfap.codart, Ipcrfap.decart, Ipcrfap.panart,;
  Ipcrfap.furart, Ipcrfap.perfap, Ipcrfap.cantant, Ipcrfap.pesart,;
  Ipcrfap.fcommvt, Ipcrfap.marca, Ipcrfap.borra, Ipcrfap.pci;
 FROM ipcrfap Ipcrfap;
 WHERE Ipcrfap.codfte = ?fappafte;
   AND Ipcrfap.codart = ?fappart;
 ORDER BY Ipcrfap.codfte, Ipcrfap.codart
```

- **v_fechamvt**

```
SELECT DISTINCT Ipctmvt.total;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.total <> 0;
   AND Ipctmvt.borra = .F.;
 ORDER BY Ipctmvt.total

 Sn Duplicados
```

- **v_fteart**

```
SELECT Ipcrfap.codfte, Ipcrfap.codart, Ipcrfap.perfap, Ipcrfap.decart,;
  Ipcmfte.nombfte, Ipcmart.nomart, ALLTRIM(STR(Ipcmart.canbart)),;
  ALLTRIM(Ipcmart.uniart);
 FROM ipcrfap Ipcrfap, ipcmfte Ipcmfte, ipcmart Ipcmart;
 WHERE Ipcrfap.codfte = Ipcmfte.codfte;
   AND Ipcrfap.codart = Ipcmart.codart;
   AND Ipcrfap.codfte = ?codigofuente
```

- **v_fteartmvt**

```
SELECT Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.zona, Ipctmvt.impctv,;
  Ipctmvt.panart, Ipctmvt.pactmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt,;
  Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.canrmvt,;
  Ipctmvt.obsmvt, Ipctmvt.decmvt, Ipctmvt.total, Ipctmvt.cantant,;
  Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.pesart, Ipctmvt.variacion,;
  Ipctmvt.fechaso, Ipctmvt.marca, Ipctmvt.fecharec, Ipctmvt.recfur,;
  Ipctmvt.borra;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.codfte = ?ftemvt;
   AND Ipctmvt.codart = ?artmvt;
 ORDER BY Ipctmvt.codfte, Ipctmvt.codart
```

- **v_ftemvtpa**

```
SELECT Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv, Ipctmvt.panart,;
  Ipctmvt.pactmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt, Ipctmvt.pesmvt,;
  Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.canrmvt, Ipctmvt.obsmvt,;
  Ipctmvt.decmvt, Ipctmvt.total, Ipctmvt.cantant, Ipctmvt.ano, Ipctmvt.mes,;
  Ipctmvt.pesart, Ipctmvt.variacion, Ipctmvt.marca, Ipctmvt.borra;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.codfte = ?ftemvtpa;
 ORDER BY Ipctmvt.codfte, Ipctmvt.codart
```

☞ **v_ftepa**

```
SELECT Ipcmfte.codfte, Ipcmfte.depfte, Ipcmfte.mpiofte, Ipcmfte.sectfte,;
  Ipcmfte.seccfte, Ipcmfte.manzfte, Ipcmfte.grupfte, Ipcmfte.clasfte,;
  Ipcmfte.consfte, Ipcmfte.nombfte, Ipcmfte.direfte, Ipcmfte.telfte,;
  Ipcmfte.estrfte, Ipcmfte.zona, Ipcmfte.marca, Ipcmfte.borra, Ipcmfte.pci,;
  Ipcmfte.tipbar, Ipcmfte.tipptov;
 FROM ipcmfte Ipcmfte;
 WHERE SUBSTR(Ipcmfte.codfte,17,4) = ?ftepafte
```

☞ **v_ftescalicap2**

```
SELECT Ipcmfte.codfte, Ipcmfte.depfte, Ipcmfte.mpiofte, Ipcmfte.sectfte,;
  Ipcmfte.seccfte, Ipcmfte.manzfte, Ipcmfte.grupfte, Ipcmfte.clasfte,;
  Ipcmfte.consfte, Ipcmfte.nombfte, Ipcmfte.direfte, Ipcmfte.telfte,;
  Ipcmfte.estrfte, Ipcmfte.zona, Ipcmfte.marca, Ipcmfte.borra, Ipcmfte.pci,;
  Ipcmfte.tipbar, Ipcmfte.tipptov;
 FROM ipcmfte Ipcmfte;
 WHERE SUBSTR(Ipcmfte.codfte,17,4) = ?ftepafte
```

Sin Duplicados

☞ **v_ftescalicri2**

```
SELECT DISTINCT Ipctmvt.impctv, Ipctmvt.codfte, Ipcmfte.nombfte;
 FROM ipcmfte Ipcmfte, ipctmvt Ipctmvt;
 WHERE Ipctmvt.codfte = Ipcmfte.codfte;
   AND (Ipctmvt.zona = ?pzonacri;
   AND Ipctmvt.ano = ?panoplacri;
   AND Ipctmvt.mes = ?pmesplacri;
   AND Ipctmvt.decmvt = ?pdecplacri;
   AND Ipctmvt.calicri = .T.);
 ORDER BY Ipcmfte.nombfte, Ipctmvt.codfte
```

Sin Duplicados

☞ **v_ftesmvt**

```
SELECT DISTINCT Ipctmvt.impctv, Ipctmvt.codfte, Ipcmfte.nombfte,;
  Ipcmfte.direfte, Ipcmfte.telfte, Ipctmvt.zona, Ipctmvt.diaarec,;
  Ipctmvt.decmvt;
 FROM ipctmvt Ipctmvt, ipcmfte Ipcmfte;
 WHERE Ipcmfte.codfte = Ipctmvt.codfte;
   AND (Ipctmvt.borra = .F.;
   AND VAL(Ipctmvt.diaarec) > 0);
 ORDER BY Ipcmfte.nombfte
```

Sin Duplicados

☞ **v_ftesplare1**

```
SELECT Ipctmvt.ctvoruta, Ipctmvt.ruta, Ipctmvt.diaarec, Ipctmvt.codfte,;
  Ipcmfte.nombfte, SUBSTR(Ipctmvt.codfte,6,4) AS sectorfte,;
  SUBSTR(Ipctmvt.codfte,10,2) AS seccion,;
  SUBSTR(Ipctmvt.codfte,12,2) AS manzafte,;
  SUBSTR(Ipctmvt.codfte,14,1) AS grpfte,;
  SUBSTR(Ipctmvt.codfte,15,2) AS clafte,;
  SUBSTR(Ipctmvt.codfte,17,4) AS ctvfte, COUNT(Ipctmvt.codart) AS ca,;
  Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt, Ipctmvt.zona;
 FROM ipctmvt Ipctmvt, ipcmfte Ipcmfte;
 WHERE Ipctmvt.codfte = Ipcmfte.codfte;
   AND (Ipctmvt.zona = ?pzonafunci1;
   AND Ipctmvt.ano = ?panoh;
   AND Ipctmvt.mes = ?pmesh;
   AND Ipctmvt.decmvt = ?pdech;
   AND Ipcmfte.borra = .F.);
 GROUP BY Ipctmvt.ctvoruta, Ipctmvt.ruta, Ipctmvt.diaarec,;
  Ipctmvt.codfte, Ipcmfte.nombfte, Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt,;
  Ipctmvt.zona;
 ORDER BY Ipctmvt.diaarec, Ipctmvt.ruta, Ipctmvt.ctvoruta,;
  Ipcmfte.nombfte
```

☞ **v_ftesplaredc**

```
SELECT Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt, Ipctmvt.zona,;
  Ipctmvt.codfte, Ipctmvt.codart;
 FROM ipctmvt Ipctmvt, ipcmfte Ipcmfte;
 WHERE Ipctmvt.codfte = Ipcmfte.codfte;
   AND (Ipctmvt.ano = ?panoh;
   AND Ipctmvt.mes = ?pmesh;
   AND Ipctmvt.zona = ?pzonafunci1;
   AND Ipcmfte.borra = .F.);
 ORDER BY Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt, Ipctmvt.zona,;
  Ipctmvt.codfte, Ipctmvt.codart
```

☞ **v_ftesplaredr**

```
SELECT Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt, Ipctmvt.zona,;
  Ipctmvt.codfte, Ipctmvt.codart;
 FROM ipctmvt Ipctmvt, ipcmfte Ipcmfte;
 WHERE Ipctmvt.codfte = Ipcmfte.codfte;
```

```
  AND (Ipctmvt.ano = ?panoh;
  AND Ipctmvt.mes = ?pmesh;
  AND Ipctmvt.zona = ?pzonafunci1;
  AND Ipcmfte.borra = .F.;
  AND Ipctmvt.total > 0;
  AND IIF((Ipctmvt.pactmvt#0.OR.Ipctmvt.pesmvt.OR.Ipctmvt.obsmvt=="88"),.T.,.F.) = .T.);
ORDER BY Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt, Ipctmvt.zona,;
 Ipctmvt.codfte, Ipctmvt.codart
```

☞ **v_ftesplaredr1**

```
SELECT Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt, Ipctmvt.zona,;
 Ipctmvt.codfte, COUNT(Ipctmvt.codart) AS artrec;
FROM ipctmvt Ipctmvt, ipcmfte Ipcmfte;
WHERE Ipctmvt.codfte = Ipcmfte.codfte;
  AND (Ipctmvt.ano = ?panoh;
  AND Ipctmvt.mes = ?pmesh;
  AND Ipctmvt.decmvt = ?pdech;
  AND Ipctmvt.zona = ?pzonafunci1;
  AND Ipctmvt.borra = .F.;
  AND Ipctmvt.total > 0;
  AND IIF((Ipctmvt.pactmvt#0.OR.Ipctmvt.pesmvt.OR.Ipctmvt.obsmvt=="88"),.T.,.F.) = .T.);
GROUP BY Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt, Ipctmvt.zona,;
 Ipctmvt.codfte;
ORDER BY Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt, Ipctmvt.zona,;
 Ipctmvt.codfte
```

☞ **v_ftesplaredr2**

```
SELECT Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt, Ipctmvt.zona,;
 Ipctmvt.codfte, COUNT(Ipctmvt.codart) AS artrec, Ipctmvt.total;
FROM ipctmvt Ipctmvt, ipcmfte Ipcmfte;
WHERE Ipctmvt.codfte = Ipcmfte.codfte;
  AND (Ipctmvt.ano = ?panoh;
  AND Ipctmvt.mes = ?pmesh;
  AND Ipctmvt.decmvt = ?pdech;
  AND Ipctmvt.zona = ?pzonafunci1;
  AND Ipctmvt.borra = .F.;
  AND Ipctmvt.total > 0;
  AND IIF((Ipctmvt.pactmvt#0.OR.Ipctmvt.pesmvt.OR.Ipctmvt.obsmvt=="88"),.T.,.F.) = .T.);
GROUP BY Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt, Ipctmvt.zona,;
 Ipctmvt.codfte, Ipctmvt.total;
ORDER BY Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt, Ipctmvt.zona,;
 Ipctmvt.codfte
```

☞ **v_ftesplasup2**

```
SELECT DISTINCT Ipctmvt.codfte, Ipcmfte.nombfte, Ipcmfte.direfte,;
```

```
 Ipcmfte.telfte;
FROM ipcmfte Ipcmfte, ipctmvt Ipctmvt;
WHERE Ipctmvt.codfte = Ipcmfte.codfte;
  AND (Ipctmvt.zona = ?pzonaplasup;
  AND Ipctmvt.ano = ?panoplarec;
  AND Ipctmvt.mes = ?pmesplarec;
  AND Ipctmvt.decmvt = ?pdecplarec;
  AND Ipctmvt.supervisado = .T.);
ORDER BY Ipcmfte.nombfte, Ipctmvt.codfte
```

Sin Duplicados

☞ **v_funcicap**

```
SELECT DISTINCT Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt,;
 Ipcrfunc.codcargo, Ipcrfunc.codfunci, Ipcfunci.nomfunci;
FROM ipcfunci Ipcfunci, ipcrfunc Ipcrfunc, ipctmvt Ipctmvt;
WHERE Ipcrfunc.codcargo = Ipcfunci.codcargo;
  AND Ipcrfunc.codfunci = Ipcfunci.codfunci;
  AND (VAL(Ipcrfunc.zonafunci) = Ipctmvt.zona;
  AND Ipcrfunc.codcargo = 'T');
GROUP BY Ipcrfunc.codfunci;
ORDER BY Ipctmvt.decmvt, Ipcrfunc.codcargo, Ipcrfunc.codfunci
```

Sin Duplicados

☞ **v_funcicri**

```
SELECT DISTINCT Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt,;
 Ipcrfunc.codcargo, Ipcrfunc.codfunci, Ipcfunci.nomfunci;
FROM ipcfunci Ipcfunci, ipcrfunc Ipcrfunc, ipctmvt Ipctmvt;
WHERE Ipcrfunc.codcargo = Ipcfunci.codcargo;
  AND Ipcrfunc.codfunci = Ipcfunci.codfunci;
  AND (VAL(Ipcrfunc.zonafunci) = Ipctmvt.zona;
  AND Ipcrfunc.codcargo = 'C';
  AND Ipcrfunc.borra = .F.);
GROUP BY Ipcrfunc.codfunci;
ORDER BY Ipctmvt.decmvt, Ipcrfunc.codcargo, Ipcrfunc.codfunci
```

Sin Duplicados

☞ **v_funcidecsupcri**

```
SELECT DISTINCT Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt,;
 Ipcrfunc.codcargo, Ipcrfunc.codfunci, Ipcfunci.nomfunci, Ipctmvt.sumr,;
 Ipctmvt.spmr, Ipctmvt.siape, Ipctmvt.siasi, Ipctmvt.siacr, Ipctmvt.siafn,;
 Ipctmvt.sadi, Ipctmvt.sonas, Ipctmvt.snae, Ipctmvt.cumr, Ipctmvt.cpmr,;
 Ipctmvt.ciape, Ipctmvt.ciasi, Ipctmvt.ciacr, Ipctmvt.ciafn, Ipctmvt.cadi,;
 Ipctmvt.conas, Ipctmvt.cnae;
```

```
FROM ipcfunci Ipcfunci, ipcrfunc Ipcrfunc, ipctmvt Ipctmvt;
WHERE Ipcrfunc.codcargo = Ipcfunci.codcargo;
  AND Ipcrfunc.codfunci = Ipcfunci.codfunci;
  AND (VAL(Ipcrfunc.zonafunci) = Ipctmvt.zona;
  AND Ipcrfunc.codcargo = 'R';
  AND Ipcrfunc.borra = .F.);
ORDER BY Ipctmvt.decmvt, Ipcrfunc.codcargo, Ipcrfunc.codfunci
```

Sin Duplicados

☞ **v_funcimescap**

```
SELECT DISTINCT Ipctmvt.ano, Ipctmvt.mes, Ipcrfunc.codcargo,;
  Ipcrfunc.codfunci, Ipcfunci.nomfunci, Ipctmvt.caea, Ipctmvt.caur,;
  Ipctmvt.capr, Ipctmvt.caape, Ipctmvt.caasi, Ipctmvt.caacr, Ipctmvt.caafc,;
  Ipctmvt.caob;
FROM ipcfunci Ipcfunci, ipcrfunc Ipcrfunc, ipctmvt Ipctmvt;
WHERE Ipcrfunc.codcargo = Ipcfunci.codcargo;
  AND Ipcrfunc.codfunci = Ipcfunci.codfunci;
  AND (VAL(Ipcrfunc.zonafunci) = Ipctmvt.zona;
  AND Ipcrfunc.codcargo = 'T';
  AND Ipcrfunc.borra = .F.);
ORDER BY Ipcrfunc.codcargo, Ipcrfunc.codfunci
```

Sin Duplicados

☞ **v_funcimescri**

```
SELECT DISTINCT Ipctmvt.ano, Ipctmvt.mes, Ipcrfunc.codcargo,;
  Ipcrfunc.codfunci, Ipcfunci.nomfunci, Ipctmvt.crea, Ipctmvt.crur,;
  Ipctmvt.crpr, Ipctmvt.crape, Ipctmvt.crasi, Ipctmvt.cracr, Ipctmvt.crafc,;
  Ipctmvt.crob;
FROM ipcfunci Ipcfunci, ipcrfunc Ipcrfunc, ipctmvt Ipctmvt;
WHERE Ipcrfunc.codcargo = Ipcfunci.codcargo;
  AND Ipcrfunc.codfunci = Ipcfunci.codfunci;
  AND (VAL(Ipcrfunc.zonafunci) = Ipctmvt.zona;
  AND Ipcrfunc.codcargo = 'C';
  AND Ipcrfunc.borra = .F.);
ORDER BY Ipcrfunc.codcargo, Ipcrfunc.codfunci
```

Sin Duplicados

☞ **v_funcimessupcri**

```
SELECT Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.zona, Ipctmvt.decmvt,;
  Ipctmvt.codfte, Ipctmvt.codart, Ipcrfunc.codcargo, Ipcrfunc.codfunci,;
  Ipcfunci.nomfunci, Ipctmvt.sumr, Ipctmvt.spmr, Ipctmvt.siape,;
```

```
 Ipctmvt.siasi, Ipctmvt.siacr, Ipctmvt.siafn, Ipctmvt.sadi, Ipctmvt.sonas,;
 Ipctmvt.snae, Ipctmvt.cumr, Ipctmvt.cpmr, Ipctmvt.ciape, Ipctmvt.ciasi,;
 Ipctmvt.ciacr, Ipctmvt.ciafn, Ipctmvt.cadi, Ipctmvt.conas, Ipctmvt.cnae;
FROM ipcfunci Ipcfunci, ipcrfunc Ipcrfunc, ipctmvt Ipctmvt;
WHERE Ipcrfunc.codcargo = Ipcfunci.codcargo;
  AND Ipcrfunc.codfunci = Ipcfunci.codfunci;
  AND (VAL(Ipcrfunc.zonafunci) = Ipctmvt.zona;
  AND Ipcrfunc.codcargo = 'R';
  AND Ipcrfunc.borra = .F.);
ORDER BY Ipcrfunc.codcargo, Ipctmvt.zona, Ipctmvt.codfte,;
 Ipctmvt.codart, Ipcrfunc.codfunci
```

☞ **v_funcionarios**

```
SELECT Ipcrfunc.codcargo, Ipccargo.nomcargo, Ipcrfunc.codfunci,;
 Ipcfunci.nomfunci, Ipcrfunc.zonafunci, Ipcrfunc.borra;
FROM ipcfunci Ipcfunci, ipcrfunc Ipcrfunc, ipccargo Ipccargo;
WHERE Ipcrfunc.codcargo = Ipcfunci.codcargo;
  AND Ipcrfunc.codfunci = Ipcfunci.codfunci;
  AND Ipcrfunc.codcargo = Ipccargo.codcargo;
  AND Ipcrfunc.borra = .F.;
ORDER BY Ipcrfunc.codcargo, Ipcrfunc.codfunci
```

☞ **v_funcirec**

```
SELECT DISTINCT Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt,;
 Ipcrfunc.codcargo, Ipcrfunc.codfunci, Ipcfunci.nomfunci,;
 Ipcrfunc.zonafunci;
FROM ipcfunci Ipcfunci, ipcrfunc Ipcrfunc, ipctmvt Ipctmvt;
WHERE Ipcrfunc.codcargo = Ipcfunci.codcargo;
  AND Ipcrfunc.codfunci = Ipcfunci.codfunci;
  AND (VAL(Ipcrfunc.zonafunci) = Ipctmvt.zona;
  AND Ipcrfunc.codcargo = 'R';
  AND Ipcrfunc.borra = .F.);
GROUP BY Ipcfunci.codfunci;
ORDER BY Ipcrfunc.codcargo, Ipcrfunc.zonafunci, Ipcrfunc.codfunci,;
 Ipctmvt.decmvt
```

Sin Duplicados

☞ **v_historia**

```
SELECT DISTINCT Historia.codfte, Historia.codart, Historia.impctv,;
 Historia.panart, Historia.pactmvt, Historia.cmvmvt, Historia.crfmvt,;
 Historia.pesmvt, Historia.sinmmvt, Historia.fcommvt, Historia.canrmvt,;
```

```
  Historia.obsmvt, Historia.decmvt, Historia.total, Historia.cantant,;
  Historia.ano, Historia.mes, Historia.pesart, Historia.variacion,;
  Historia.marca, Historia.borra, Historia.pci;
 FROM historia Historia;
 WHERE Historia.borra = .F.;
 ORDER BY Historia.impctv DESC, Historia.codart
Sin Duplicados
```

☞ **v_histpre**

```
SELECT Histpre.codfte, Histpre.codart, Histpre.decmvt, Histpre.mesmvt,;
  Histpre.anomvt, Histpre.pesart, Histpre.panart, Histpre.cantant,;
  Histpre.pactmvt, Histpre.canrmvt, Histpre.pesmvt, Histpre.crfmvt,;
  Histpre.sinmmvt, Histpre.fcommvt, Histpre.marca, Histpre.borra;
 FROM histpre Histpre
```

☞ **v_impctvesp**

```
SELECT DISTINCT Esptemp.codfte, Esptemp.codart, Esptemp.codesp,;
  Esptemp.nomesp, Esptemp.nombdet, Esptemp.impctv, Esptemp.ano, Esptemp.mes,;
  Esptemp.decmvt, Esptemp.marca, Esptemp.borra;
 FROM esptemp Esptemp;
 WHERE Esptemp.borra = .F.;
   AND Esptemp.impctv = ?impctvh;
 ORDER BY Esptemp.codart, Esptemp.codesp
```

☞ **v_indcalicap**

```
SELECT Ipctmvt.decmvt,;
  ALLTRIM(STR(SUM(((100-
Ipctmvt.totalcalicap)*Pondesgc.vlr_pon))/SUM(Pondesgc.vlr_pon),6,2)) AS indicadeca;
 FROM ipctmvt Ipctmvt, Pondesgc Pondesgc;
 WHERE Ipctmvt.borra = .F.;
   AND Ipctmvt.calicap = .T.;
   AND SUBSTR(Ipctmvt.codfte,14,1)+SUBSTR(Ipctmvt.codart,1,5) =
Pondesgc.codgri+SUBSTR(Pondesgc.codart,1,5);
 GROUP BY Ipctmvt.decmvt;
 ORDER BY Ipctmvt.decmvt
```

☞ **v_indcalicapxzona**

```
SELECT ALLTRIM(STR(SUM(((100-
Ipctmvt.totalcalicap)*Pondesgc.vlr_pon))/SUM(Pondesgc.vlr_pon),6,2)) AS indicadeca,;
  Ipcrfunc.codcargo, Ipcrfunc.codfunci;
 FROM ipctmvt Ipctmvt, Pondesgc Pondesgc, ipcrfunc Ipcrfunc;
 WHERE Ipctmvt.borra = .F.;
   AND Ipctmvt.calicap = .T.;
   AND Ipcrfunc.codcargo = 'T';
```

```
  AND SUBSTR(Ipctmvt.codfte,14,1)+SUBSTR(Ipctmvt.codart,1,5) =
Pondesgc.codgri+SUBSTR(Pondesgc.codart,1,5);
  AND Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
  AND Ipcrfunc.borra = .F.;
 GROUP BY Ipcrfunc.codcargo, Ipcrfunc.codfunci;
 ORDER BY Ipcrfunc.codfunci
```

☞ **v_indcalicri**

```
SELECT Ipctmvt.decmvt,;
  ALLTRIM(STR(SUM(((100-
Ipctmvt.totalcalicri)*Pondesgc.vlr_pon))/SUM(Pondesgc.vlr_pon),10,6)) AS indicadeca;
 FROM ipctmvt Ipctmvt, Pondesgc Pondesgc;
 WHERE Ipctmvt.borra = .F.;
  AND Ipctmvt.calicri = .T.;
  AND SUBSTR(Ipctmvt.codfte,14,1)+SUBSTR(Ipctmvt.codart,1,5) =
Pondesgc.codgri+SUBSTR(Pondesgc.codart,1,5);
 GROUP BY Ipctmvt.decmvt;
 ORDER BY Ipctmvt.decmvt
```

☞ **v_indcalicrixzona**

```
SELECT ALLTRIM(STR(SUM(((100-
Ipctmvt.totalcalicri)*Pondesgc.vlr_pon))/SUM(Pondesgc.vlr_pon),6,2)) AS indicadeca,;
  Ipcrfunc.codcargo, Ipcrfunc.codfunci;
 FROM ipctmvt Ipctmvt, Pondesgc Pondesgc, ipcrfunc Ipcrfunc;
 WHERE Ipctmvt.borra = .F.;
  AND Ipctmvt.calicri = .T.;
  AND Ipcrfunc.codcargo = 'C';
  AND SUBSTR(Ipctmvt.codfte,14,1)+SUBSTR(Ipctmvt.codart,1,5) =
Pondesgc.codgri+SUBSTR(Pondesgc.codart,1,5);
  AND Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
  AND Ipcrfunc.borra = .F.;
 GROUP BY Ipcrfunc.codcargo, Ipcrfunc.codfunci;
 ORDER BY Ipcrfunc.codfunci
```

☞ **v_indciuda**

```
SELECT Indciuda.coddpto, Indciuda.codmpio, Indciuda.ano, Indciuda.mes,;
  Indciuda.decada, Indciuda.proceso, Indciuda.indciudad;
 FROM indciuda Indciuda;
 WHERE Indciuda.proceso = ?procesoind;
  AND Indciuda.ano = ?anoind;
  AND Indciuda.mes = ?mesind;
 ORDER BY Indciuda.decada
```

☞ **v_indplacri**

```
SELECT Ipctmvt.decmvt,;
```

```
 ALLTRIM(STR(SUM(((100-
Ipctmvt.totalplacri)*Pondesgc.vlr_pon))/SUM(Pondesgc.vlr_pon),10,6)) AS indicadeca;
 FROM ipctmvt Ipctmvt, Pondesgc Pondesgc;
 WHERE Ipctmvt.borra = .F.;
   AND Ipctmvt.placri = .T.;
   AND SUBSTR(Ipctmvt.codfte,14,1)+SUBSTR(Ipctmvt.codart,1,5) =
Pondesgc.codgri+SUBSTR(Pondesgc.codart,1,5);
 GROUP BY Ipctmvt.decmvt;
 ORDER BY Ipctmvt.decmvt
```

☞ **v_indplacrixzona**

```
SELECT ALLTRIM(STR(SUM(((100-
Ipctmvt.totalplacri)*Pondesgc.vlr_pon))/SUM(Pondesgc.vlr_pon),6,2)) AS indicadeca,;
  Ipctmvt.zona;
 FROM ipctmvt Ipctmvt, Pondesgc Pondesgc;
 WHERE Ipctmvt.borra = .F.;
   AND Ipctmvt.placri = .T.;
   AND SUBSTR(Ipctmvt.codfte,14,1)+SUBSTR(Ipctmvt.codart,1,5) =
Pondesgc.codgri+SUBSTR(Pondesgc.codart,1,5);
 GROUP BY Ipctmvt.zona
```

☞ **v_indplasup**

```
SELECT Ipctmvt.decmvt,;
  ALLTRIM(STR(SUM(((100-
Ipctmvt.totalplasup)*Pondesgc.vlr_pon))/SUM(Pondesgc.vlr_pon),10,6)) AS indicadeca;
 FROM ipctmvt Ipctmvt, Pondesgc Pondesgc;
 WHERE Ipctmvt.borra = .F.;
   AND Ipctmvt.supervisado = .T.;
   AND SUBSTR(Ipctmvt.codfte,14,1)+SUBSTR(Ipctmvt.codart,1,5) =
Pondesgc.codgri+SUBSTR(Pondesgc.codart,1,5);
 GROUP BY Ipctmvt.decmvt;
 ORDER BY Ipctmvt.decmvt
```

☞ **v_indplasupxzona**

```
SELECT ALLTRIM(STR(SUM(((100-
Ipctmvt.totalplasup)*Pondesgc.vlr_pon))/SUM(Pondesgc.vlr_pon),6,2)) AS indicadeca,;
  Ipctmvt.zona;
 FROM ipctmvt Ipctmvt, Pondesgc Pondesgc;
 WHERE Ipctmvt.borra = .F.;
   AND Ipctmvt.supervisado = .T.;
   AND SUBSTR(Ipctmvt.codfte,14,1)+SUBSTR(Ipctmvt.codart,1,5) =
Pondesgc.codgri+SUBSTR(Pondesgc.codart,1,5);
 GROUP BY Ipctmvt.zona
```

- **v_ipccargo**

```
SELECT Ipccargo.codcargo, Ipccargo.nomcargo, Ipccargo.marca,;
  Ipccargo.borra;
 FROM ipccargo Ipccargo;
 ORDER BY Ipccargo.nomcargo
```

- **v_ipcdias**

```
SELECT Ipcdias.zonafunci, Ipcdias.codfunci, Ipcdias.anoplarec,;
  Ipcdias.mesplarec, Ipcdias.decplarec, Ipcdias.ctvplarec, Ipcdias.ctvdia,;
  Ipcdias.nrodia, Ipcdias.codfte, Ipcdias.ca, Ipcdias.ar, Ipcdias.pendientes,;
  Ipcdias.observa, Ipcdias.marca, Ipcdias.borra;
 FROM ipcdias Ipcdias;
 WHERE Ipcdias.zonafunci = ?pzonafunci1;
   AND Ipcdias.anoplarec = ?panoplarec;
   AND Ipcdias.mesplarec = ?pmesplarec;
   AND Ipcdias.decplarec = ?pdecplarec;
   AND Ipcdias.codfte = ?pftedia;
 ORDER BY Ipcdias.ctvplarec, Ipcdias.ctvdia
```

- **v_ipcdias2**

```
SELECT Ipcdias.zonafunci, Ipcdias.codfunci, Ipcdias.anoplarec,;
  Ipcdias.mesplarec, Ipcdias.decplarec, Ipcdias.ctvplarec, Ipcdias.ctvdia,;
  Ipcdias.nrodia, Ipcdias.codfte, Ipcdias.ca, Ipcdias.ar, Ipcdias.pendientes,;
  Ipcdias.observa;
 FROM ipcdias Ipcdias;
 WHERE Ipcdias.borra = .F.;
 ORDER BY Ipcdias.ctvplarec, Ipcdias.ctvdia
```

- **v_ipcestr**

```
SELECT *;
 FROM ipcestr Ipcestr
```

- **v_ipcfunccc**

```
SELECT Ipcfunci.codcargo, Ipcfunci.codfunci, Ipcfunci.nomfunci;
 FROM ipcfunci Ipcfunci;
 WHERE Ipcfunci.codcargo = 'C';
   AND Ipcfunci.borra = .F.;
 ORDER BY Ipcfunci.codcargo, Ipcfunci.codfunci
```

- **v_ipcfunci**

```
SELECT Ipcfunci.codcargo, Ipcfunci.zonafunci, Ipcfunci.codfunci,;
  Ipcfunci.nomfunci, Ipcfunci.marca, Ipcfunci.borra;
 FROM ipcfunci Ipcfunci;
 WHERE Ipcfunci.codcargo = ?pcodcargo;
 ORDER BY Ipcfunci.zonafunci
```

☞ **v_ipcindzo**

```
SELECT Ipcindzo.coddpto, Ipcindzo.codmpio, Ipcindzo.ano, Ipcindzo.mes,;
  Ipcindzo.decada, Ipcindzo.zona, Ipcindzo.proceso, Ipcindzo.indicazona;
 FROM ipcindzo Ipcindzo;
 WHERE Ipcindzo.proceso = ?procesoind;
   AND Ipcindzo.ano = ?anoind;
   AND Ipcindzo.mes = ?mesind;
 ORDER BY Ipcindzo.decada, Ipcindzo.zona
```

☞ **v_ipcmart**

```
SELECT Ipcmart.codart, Ipcmart.grpart, Ipcmart.subgart, Ipcmart.clasart,;
  Ipcmart.gasbart, Ipcmart.variart, Ipcmart.perart, Ipcmart.nomart,;
  Ipcmart.canbart, Ipcmart.uniart, Ipcmart.presart, Ipcmart.calcula,;
  Ipcmart.fornal, Ipcmart.educacion, Ipcmart.mensuales, Ipcmart.marca,;
  Ipcmart.borra;
 FROM ipcmart Ipcmart;
 ORDER BY Ipcmart.codart
```

☞ **v_ipcmes**

```
SELECT *;
 FROM ipcmes Ipcmes;
 ORDER BY Ipcmes.cod_mes
```

☞ **v_ipcmfte**

```
SELECT Ipcmfte.codfte, Ipcmfte.depfte, Ipcmfte.mpiofte, Ipcmfte.sectfte,;
  Ipcmfte.seccfte, Ipcmfte.manzfte, Ipcmfte.grupfte, Ipcmfte.clasfte,;
  Ipcmfte.consfte, Ipcmfte.nombfte, Ipcmfte.direfte, Ipcmfte.telfte,;
  Ipcmfte.estrfte, Ipcmfte.zona, Ipcmfte.marca, Ipcmfte.borra, Ipcmfte.pci,;
  Ipcmfte.tipbar, Ipcmfte.tipptov;
 FROM ipcmfte Ipcmfte;
 ORDER BY Ipcmfte.codfte
```

☞ **v_ipcmfte1**

```
SELECT Ipcmfte.codfte, Ipcmfte.depfte, Ipcmfte.mpiofte, Ipcmfte.sectfte,;
  Ipcmfte.seccfte, Ipcmfte.manzfte, Ipcmfte.grupfte, Ipcmfte.clasfte,;
  Ipcmfte.consfte, Ipcmfte.nombfte, Ipcmfte.direfte, Ipcmfte.telfte,;
```

```
 Ipcmfte.estrfte, Ipcmfte.zona, Ipcmfte.marca, Ipcmfte.borra, Ipcmfte.pci,;
 Ipcmfte.tipbar, Ipcmfte.tipptov;
FROM ipcmfte Ipcmfte;
ORDER BY Ipcmfte.codfte
```

☞ **v_ipcplact**

```
SELECT Ipcplact.codcargo, Ipcplact.zonafunci, Ipcplact.codfunci,;
 Ipcplact.anoplarec, Ipcplact.mesplarec, Ipcplact.decplarec,;
 Ipcplact.ctvdia, Ipcplact.nrodia, Ipcplact.entra, Ipcplact.sale,;
 Ipcplact.pendientes, Ipcplact.totalentra, Ipcplact.totalsale,;
 Ipcplact.totalpte, Ipcplact.marca, Ipcplact.borra;
FROM ipcplact Ipcplact;
ORDER BY Ipcplact.codcargo, Ipcplact.zonafunci, Ipcplact.anoplarec,;
 Ipcplact.mesplarec, Ipcplact.decplarec, Ipcplact.nrodia
```

☞ **v_ipcplact_c**

```
SELECT Ipcplact.codcargo, Ipcplact.codfunci, Ipcplact.anoplarec,;
 Ipcplact.mesplarec, Ipcplact.decplarec, Ipcplact.ctvdia, Ipcplact.nrodia,;
 Ipcplact.entra, Ipcplact.sale, Ipcplact.pendientes, Ipcplact.totalentra,;
 Ipcplact.totalsale, Ipcplact.totalpte, Ipcplact.marca, Ipcplact.borra,;
 Ipcplact.zonafunci,;
 IIF(Ipcplact.sale#0.AND.Ipcplact.entra#0,STR(Ipcplact.sale/Ipcplact.entra,7,2),'') AS indicador;
FROM ipcplact Ipcplact;
WHERE Ipcplact.codcargo = 'C';
  AND ALLTRIM(Ipcplact.codfunci) = ?pcodfunci;
  AND Ipcplact.decplarec = ?pcargadec;
ORDER BY Ipcplact.anoplarec, Ipcplact.mesplarec, Ipcplact.decplarec,;
 Ipcplact.ctvdia, Ipcplact.nrodia
```

☞ **v_ipcplact_s**

```
SELECT Ipcplact.codcargo, Ipcplact.codfunci, Ipcplact.anoplarec,;
 Ipcplact.mesplarec, Ipcplact.decplarec, Ipcplact.ctvdia, Ipcplact.nrodia,;
 Ipcplact.entra, Ipcplact.sale, Ipcplact.pendientes, Ipcplact.totalentra,;
 Ipcplact.totalsale, Ipcplact.totalpte, Ipcplact.marca, Ipcplact.borra,;
 Ipcplact.zonafunci,;
 IIF(Ipcplact.sale#0.AND.Ipcplact.entra#0,STR(Ipcplact.sale/Ipcplact.entra,7,2),'') AS indicador;
FROM ipcplact Ipcplact;
WHERE Ipcplact.codcargo = 'S';
  AND ALLTRIM(Ipcplact.codfunci) = ?pcodfunci;
  AND Ipcplact.decplarec = ?pcargadec;
ORDER BY Ipcplact.anoplarec, Ipcplact.mesplarec, Ipcplact.decplarec,;
 Ipcplact.ctvdia, Ipcplact.nrodia
```

☞ **v_ipcplact_t**

```
SELECT Ipcplact.codcargo, Ipcplact.codfunci, Ipcplact.anoplarec,;
  Ipcplact.mesplarec, Ipcplact.decplarec, Ipcplact.ctvdia, Ipcplact.nrodia,;
  Ipcplact.entra, Ipcplact.sale, Ipcplact.pendientes, Ipcplact.totalentra,;
  Ipcplact.totalsale, Ipcplact.totalpte, Ipcplact.marca, Ipcplact.borra,;
  Ipcplact.zonafunci,;
  IIF(Ipcplact.sale#0.AND.Ipcplact.entra#0,STR(Ipcplact.sale/Ipcplact.entra,7,2),'') AS indicador;
 FROM ipcplact Ipcplact;
 WHERE Ipcplact.codcargo = 'T';
   AND ALLTRIM(Ipcplact.codfunci) = ?pcodfunci;
   AND Ipcplact.decplarec = ?pcargadec;
 ORDER BY Ipcplact.anoplarec, Ipcplact.mesplarec, Ipcplact.decplarec,;
  Ipcplact.ctvdia, Ipcplact.nrodia
```

☞ **v_ipcrango**

```
SELECT DISTINCT Ipcrango.coddpto, Ipcrango.codmpio, Ipcrango.codart,;
  Ipcrango.estrfte, ALLTRIM(STR(Ipcrango.estrfte)) AS grupo,;
  Ipcrango.clasfte, Ipcrango.maxart, Ipcrango.minart, Ipcrango.rango,;
  Ipcrango.marca, Ipcrango.borra;
 FROM ipcrango Ipcrango;
 ORDER BY Ipcrango.codart, Ipcrango.estrfte
```

Sin Duplicados

☞ **v_ipcresp**

```
SELECT Ipcresp.codfte, Ipcresp.codart, Ipcresp.codesp, Ipcresp.nomesp,;
  Ipcresp.nombdet, Ipcresp.nombdet1, Ipcresp.tipo, Ipcresp.valor,;
  Ipcresp.marca, Ipcresp.borra, Ipcresp.orden;
 FROM ipcresp Ipcresp;
 ORDER BY Ipcresp.codfte, Ipcresp.codart, Ipcresp.codesp
```

☞ **v_ipcrespci**

```
SELECT Ipcresp.codfte, Ipcresp.codart, Ipcresp.codesp, Ipcresp.nomesp,;
  Ipcresp.nombdet, Ipcresp.nombdet1, Ipcresp.tipo, Ipcresp.valor,;
  Ipcresp.marca, Ipcresp.borra, Ipcresp.orden;
 FROM ipcresp Ipcresp;
 WHERE SUBSTR(Ipcresp.codfte,17,4) >= ?vt_cvtini;
   AND SUBSTR(Ipcresp.codfte,17,4) <= ?vt_cvtini1
```

☞ **v_ipcrfap**

```
SELECT Ipcrfap.codfte, Ipcrfap.codart, Ipcrfap.decart, Ipcrfap.panart,;
  Ipcrfap.furart, Ipcrfap.perfap, Ipcrfap.cantant, Ipcrfap.pesart,;
  Ipcrfap.fcommvt, Ipcrfap.marca, Ipcrfap.borra, Ipcrfap.pci;
 FROM ipcrfap Ipcrfap;
 ORDER BY Ipcrfap.codfte, Ipcrfap.codart
```

☞ **v_ipcrfunc**

```
SELECT Ipcrfunc.codcargo, Ipcrfunc.codfunci, Ipcrfunc.ctvzona,;
  Ipcrfunc.zonafunci, Ipcrfunc.marca, Ipcrfunc.borra;
 FROM ipcrfunc Ipcrfunc;
 WHERE Ipcrfunc.codcargo = ?pcodcargo;
   AND ALLTRIM(Ipcrfunc.codfunci) = ?pcodfunci
```

☞ **v_ipcrfunc1**

```
SELECT DISTINCT Ipcrfunc.codcargo, Ipcrfunc.codfunci, Ipcrfunc.ctvzona,;
  Ipcrfunc.zonafunci, Ipcrfunc.marca, Ipcrfunc.borra;
 FROM ipcrfunc Ipcrfunc;
 WHERE Ipcrfunc.codcargo = ?pcodcargo;
   AND Ipcrfunc.borra = .F.
```

Sin Duplicados

☞ **v_ipcrfunc2**

```
SELECT DISTINCT Ipcrfunc.codcargo, Ipcrfunc.codfunci, Ipcrfunc.ctvzona,;
  Ipcrfunc.zonafunci, Ipcrfunc.marca, Ipcrfunc.borra;
 FROM ipcrfunc Ipcrfunc;
 WHERE Ipcrfunc.codcargo = ?pcodcargo;
 GROUP BY Ipcrfunc.codcargo, Ipcrfunc.codfunci
```

Sin Duplicados

☞ **v_ipcrobs**

```
SELECT Ipcrobs.impctv, Ipcrobs.codfte, Ipcrobs.codobs, Ipcrobs.observa,;
  Ipcrobs.marca, Ipcrobs.borra;
 FROM ipcrobs Ipcrobs;
 ORDER BY Ipcrobs.impctv, Ipcrobs.codfte
```

☞ **v_ipcrpart**

```
SELECT *;
 FROM ipcrpart Ipcrpart;
 ORDER BY Ipcrpart.codper, Ipcrpart.codpart
```

☞ **v_ipctctv**

```
SELECT Ipctctv.inictv, Ipctctv.finctv, Ipctctv.añoctv, Ipctctv.mesctv,;
  Ipctctv.decctv, Ipctctv.estpre, Ipctctv.estcap, Ipctctv.esttras,;
  Ipctctv.estpro;
 FROM ipctctv Ipctctv;
```

ORDER BY Ipctctv.finctv

☞ **v_ipctctv2**

* utilizado en indicadores de calidad para saber la fecha de prediligenciamiento
SELECT *;
FROM ipctctv Ipctctv;
ORDER BY Ipctctv.finctv

☞ **v_ipctfcm**

SELECT DISTINCT Ipctfcm.codfcom, Ipctfcm.nomfte, Ipctfcm.direfte,;
Ipctfcm.telfte, Ipctfcm.codfte, Ipctfcm.marca, Ipctfcm.borra;
FROM ipctfcm Ipctfcm;
WHERE Ipctfcm.codfte = ?NoFte;
ORDER BY Ipctfcm.codfte, Ipctfcm.codfcom

Sin Duplicados

☞ **v_ipctmvt**

SELECT Ipctmvt.zona, Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv,;
Ipctmvt.panart, Ipctmvt.pactmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt,;
Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.canrmvt,;
Ipctmvt.obsmvt, Ipctmvt.decmvt, Ipctmvt.total, Ipctmvt.cantant,;
Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.pesart, Ipctmvt.variacion, Ipctmvt.marca,;
Ipctmvt.borra, Ipctmvt.pci, Ipctmvt.diaarec, Ipctmvt.ruta,;
Ipctmvt.ctvoruta, Ipctmvt.movdiasup, Ipctmvt.diasup, Ipctmvt.diacri,;
Ipctmvt.diacalicri, Ipctmvt.diacalisup, Ipctmvt.plasup, Ipctmvt.calicri,;
Ipctmvt.sumr, Ipctmvt.spmr, Ipctmvt.siape, Ipctmvt.siasi, Ipctmvt.siacr,;
Ipctmvt.siafn, Ipctmvt.sadi, Ipctmvt.sonas, Ipctmvt.snae, Ipctmvt.imsxf,;
Ipctmvt.sdgf, Ipctmvt.cadgf, Ipctmvt.crdgf, Ipctmvt.dgf, Ipctmvt.pe,;
Ipctmvt.si, Ipctmvt.cr, Ipctmvt.fn, Ipctmvt.obsplasup, Ipctmvt.placri,;
Ipctmvt.umr, Ipctmvt.iafn, Ipctmvt.onas, Ipctmvt.nae, Ipctmvt.totalplasup,;
Ipctmvt.supervisado, Ipctmvt.obssupersado, Ipctmvt.fechasup,;
Ipctmvt.diaasupervisar, Ipctmvt.diaanalisis, Ipctmvt.sianalisis,;
Ipctmvt.fecharec, Ipctmvt.cumr, Ipctmvt.cpmr, Ipctmvt.ciape, Ipctmvt.ciasi,;
Ipctmvt.ciacr, Ipctmvt.ciafn, Ipctmvt.cadi, Ipctmvt.conas, Ipctmvt.cnae,;
Ipctmvt.obsplacri, Ipctmvt.totalplacri, Ipctmvt.crea, Ipctmvt.crur,;
Ipctmvt.crpr, Ipctmvt.crape, Ipctmvt.crasi, Ipctmvt.cracr, Ipctmvt.crafc,;
Ipctmvt.crob, Ipctmvt.totalcalicri, Ipctmvt.obscalicri;
FROM ipctmvt Ipctmvt;
ORDER BY Ipctmvt.impctv, Ipctmvt.codfte, Ipctmvt.codart

☞ **v_ipctmvt7**

SELECT Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv, Ipctmvt.panart,;
Ipctmvt.pactmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt, Ipctmvt.pesmvt,;
Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.canrmvt, Ipctmvt.obsmvt,;

```
 Ipctmvt.decmvt, Ipctmvt.total, Ipctmvt.cantant, Ipctmvt.ano, Ipctmvt.mes,;
 Ipctmvt.pesart, Ipctmvt.variacion, Ipctmvt.marca, Ipctmvt.fecharec;
FROM ipctmvt Ipctmvt;
WHERE {fn substring(Ipctmvt.codart,1,3)} = "721";
ORDER BY Ipctmvt.codfte, Ipctmvt.codart
```

☞ **v_ipctmvt8**

```
SELECT Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv, Ipctmvt.panart,;
 Ipctmvt.pactmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt, Ipctmvt.pesmvt,;
 Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.canrmvt, Ipctmvt.obsmvt,;
 Ipctmvt.decmvt, Ipctmvt.total, Ipctmvt.cantant, Ipctmvt.ano, Ipctmvt.mes,;
 Ipctmvt.pesart, Ipctmvt.variacion, Ipctmvt.marca, Ipctmvt.fecharec;
FROM ipctmvt Ipctmvt;
WHERE Ipctmvt.codart IN ('2210102','2220101','2220201','2220202','2220203','7320101');
  AND Ipctmvt.borra = .F.;
ORDER BY Ipctmvt.codfte, Ipctmvt.codart
```

☞ **v_ipctmvtart**

```
SELECT Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt, Ipctmvt.zona,;
 Ipctmvt.impctv, Ipctmvt.codfte, Ipctmvt.codart, Ipcmart.nomart,;
 ALLTRIM(STR(Ipcmart.canbart))+ALLTRIM(Ipcmart.uniart) AS cantunib,;
 Ipctmvt.cantant, Ipctmvt.panart, Ipctmvt.pesart, Ipctmvt.pesmvt,;
 Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt,;
 Ipctmvt.canrmvt, Ipctmvt.pactmvt, Ipctmvt.obsmvt, Ipctmvt.total,;
 Ipctmvt.variacion, Ipctmvt.fechaso, Ipctmvt.supervisado, Ipctmvt.plasup,;
 Ipctmvt.marca, Ipctmvt.borra, SUBSTR(Ipctmvt.codfte,14,1) AS grupo,;
 SUBSTR(Ipctmvt.codfte,15,2) AS clase, Ipctmvt.recfur,;
 IIF(VAL(Ipctmvt.variacion)#0,STR(ROUND(VAL(Ipctmvt.variacion),2),7,2),'       ') AS variacion1,;
 Ipctmvt.fecharec;
FROM ipctmvt Ipctmvt, ipcmart Ipcmart;
WHERE Ipctmvt.codart = Ipcmart.codart;
  AND (Ipctmvt.borra = .F.;
  AND Ipctmvt.impctv = ?ctvcaptu;
  AND VAL(Ipctmvt.diaarec) > 0);
ORDER BY Ipctmvt.codart
```

☞ **v_ipctmvtpr**

```
SELECT Ipctmvt.zona, Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv,;
 Ipctmvt.panart, Ipctmvt.pactmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt,;
 Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.canrmvt,;
 Ipctmvt.obsmvt, Ipctmvt.decmvt, Ipctmvt.total, Ipctmvt.cantant,;
 Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.pesart, Ipctmvt.variacion, Ipctmvt.marca,;
 Ipctmvt.borra, Ipctmvt.demas, Ipctmvt.pci;
FROM ipctmvt Ipctmvt;
```

```
WHERE SUBSTR(Ipctmvt.codfte,17,4) >= ?vt_cvtini;
  AND SUBSTR(Ipctmvt.codfte,17,4) <= ?vt_cvtini1;
ORDER BY Ipctmvt.impctv, Ipctmvt.codfte, Ipctmvt.codart
```

☞ **v_meshist**

```
SELECT DISTINCT Historia.mes;
 FROM historia Historia;
 WHERE Historia.ano = ?anohist;
   AND Historia.borra = .F.;
 ORDER BY Historia.mes
```

Sin Duplicados

☞ **v_mesplarec**

```
SELECT SUM(Ipcdias.ar) AS cacaptu,;
  STR((SUM(Ipcdias.ar)/SUM(Ipcdias.ca))*100,6,2) AS porcentmes,;
  Ipcdias.mesplarec;
 FROM ipcdias Ipcdias;
 WHERE Ipcdias.borra = .F.;
   AND Ipcdias.nrodia IS NOT NULL;
   AND Ipcdias.zonafunci = ?pzonafunci1;
   AND Ipcdias.anoplarec = ?panoplarec;
   AND Ipcdias.mesplarec = ?pmesplarec;
 GROUP BY Ipcdias.mesplarec
```

☞ **v_modelos**

```
SELECT *;
 FROM modelos Modelos;
 WHERE Modelos.codfte = ?mFuente;
   AND Modelos.codart = ?mArticulo;
 ORDER BY Modelos.codfte, Modelos.codart, Modelos.codmod
```

☞ **v_mvtporcent**

```
SELECT DISTINCT Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv,;
  Ipctmvt.pactmvt, Ipctmvt.pesmvt, Ipctmvt.decmvt;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.impctv = ?ctvcaptu;
   AND Ipctmvt.borra = .F.
```

Sin Duplicados

☞ **v_mvtporcentdec**

```
SELECT DISTINCT Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.pactmvt,;
  Ipctmvt.pesmvt, Ipctmvt.decmvt;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.decmvt = ?pdeccaptu;
   AND Ipctmvt.borra = .F.
```

Sin Duplicados

☞ **v_mvtporcentmes**

```
SELECT DISTINCT Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.pactmvt,;
  Ipctmvt.pesmvt, Ipctmvt.mes;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.mes = ?pmescaptu;
   AND Ipctmvt.borra = .F.
```

Sin Duplicados

☞ **v_obscali**

```
SELECT Ipcrobs.impctv, Ipcrobs.codfte, Ipcrobs.codobs, Ipcrobs.observa,;
  Ipcrobs.marca, Ipcrobs.borra;
 FROM ipcrobs Ipcrobs;
 WHERE Ipcrobs.impctv = ?pobsctvimp;
   AND Ipcrobs.codfte = ?pobscodfte;
   AND Ipcrobs.codobs = ?pobscodobs
```

☞ **v_observa**

```
SELECT Observa.codobs, Observa.observa, Observa.alza, Observa.baja,;
  Observa.novpe, Observa.codnove, Observa.marca, Observa.borra;
 FROM observa Observa;
 ORDER BY Observa.codobs
```

☞ **v_oftepa**

```
SELECT Ipcmfte.codfte, Ipcmfte.depfte, Ipcmfte.mpiofte, Ipcmfte.sectfte,;
  Ipcmfte.seccfte, Ipcmfte.manzfte, Ipcmfte.grupfte, Ipcmfte.clasfte,;
  Ipcmfte.consfte, Ipcmfte.nombfte, Ipcmfte.direfte, Ipcmfte.telfte,;
  Ipcmfte.estrfte, Ipcmfte.zona, Ipcmfte.marca, Ipcmfte.borra;
 FROM ipcmfte Ipcmfte;
 WHERE Ipcmfte.codfte = ?oftepafte
```

☞ **v_oresppa**

```
SELECT Ipcresp.codfte, Ipcresp.codart, Ipcresp.codesp, Ipcresp.nomesp,;
```

```
 Ipcresp.nombdet, Ipcresp.nombdet1, Ipcresp.tipo, Ipcresp.valor,;
 Ipcresp.marca, Ipcresp.borra;
FROM ipcresp Ipcresp;
WHERE Ipcresp.codfte = ?orespafte;
ORDER BY Ipcresp.codfte, Ipcresp.codart, Ipcresp.codesp
```

☞ **v_pcimart**

```
SELECT Pcimart.codipc, Pcimart.codpci, Pcimart.nomart, Pcimart.espec,;
 Pcimart.comipc, Pcimart.cant, Pcimart.codum;
FROM pcimart Pcimart
```

☞ **v_pcirpart**

```
SELECT Pcirpart.codper, Pcirpart.codpart, Pcirpart.mes;
 FROM pcirpart Pcirpart
```

☞ **v_pdaftes**

```
SELECT Ipcpdafte.zona, Ipcpdafte.codfte, Ipcpdafte.codfte20,;
 Ipcpdafte.nombfte, Ipcpdafte.direfte, Ipcpdafte.telfte,;
 Ipcpdafte.grupfte, Ipcpdafte.claffte, Ipcpdafte.marca, Ipcpdafte.borra;
FROM ipcpdafte Ipcpdafte;
WHERE Ipcpdafte.borra = .F.;
ORDER BY Ipcpdafte.zona, Ipcpdafte.codfte
```

☞ **v_pdazonas**

```
SELECT DISTINCT Ipcpdafte.zona;
 FROM ipcpdafte Ipcpdafte;
 WHERE Ipcpdafte.borra = .F.;
 ORDER BY Ipcpdafte.zona
```

☞ **v_pesptemp**

```
SELECT DISTINCT Esptemp.codfte, Esptemp.codart, Esptemp.codesp,;
 Esptemp.nomesp, Esptemp.nombdet, Esptemp.nombdet1, Esptemp.impctv,;
 Esptemp.ano, Esptemp.mes, Esptemp.decmvt, Esptemp.marca, Esptemp.borra;
FROM esptemp Esptemp;
WHERE Esptemp.borra = .F.;
  AND Esptemp.ano = ?anohist;
  AND Esptemp.mes = ?meshist;
```

ORDER BY Esptemp.impctv

Sin Duplicados

☞ **v_phistoria**

```
SELECT DISTINCT Historia.codfte, Historia.codart, Historia.impctv,;
  Historia.panart, Historia.pactmvt, Historia.cmvmvt, Historia.crfmvt,;
  Historia.pesmvt, Historia.sinmmvt, Historia.fcommvt, Historia.canrmvt,;
  Historia.obsmvt, Historia.decmvt, Historia.total, Historia.cantant,;
  Historia.ano, Historia.mes, Historia.pesart, Historia.variacion,;
  Historia.marca, Historia.borra;
 FROM historia Historia;
 WHERE Historia.borra = .F.;
   AND Historia.ano = ?anohist;
   AND Historia.mes = ?meshist;
 ORDER BY Historia.impctv
```

Sin Duplicados

☞ **v_pon_cali**

```
SELECT Pondesgc.coddpto, Pondesgc.codmpio, Pondesgc.codart,;
  Pondesgc.codgri, Pondesgc.vlr_pon, Pondesgc.marca, Pondesgc.borra;
 FROM Pondesgc Pondesgc;
 WHERE Pondesgc.coddpto = ?departamento;
   AND Pondesgc.codmpio = ?municipio;
 ORDER BY Pondesgc.codgri, Pondesgc.codart
```

☞ **v_porcentcapdeczona**

```
SELECT DISTINCT Ipctmvt.zona, COUNT(Ipctmvt.codart), Ipctmvt.pactmvt,;
  Ipctmvt.pesmvt, Ipctmvt.decmvt;
 FROM Ipctmvt Ipctmvt;
 WHERE Ipctmvt.decmvt = ?pdeccaptu;
   AND Ipctmvt.borra = .F.;
   AND Ipctmvt.pactmvt <> 0;
   OR (Ipctmvt.pesmvt = .T.);
 GROUP BY Ipctmvt.zona
```

Sin Duplicados

☞ **v_porcentcapmeszona**

```
SELECT DISTINCT Ipctmvt.zona, COUNT(Ipctmvt.codart), Ipctmvt.pactmvt,;
  Ipctmvt.pesmvt, Ipctmvt.mes;
 FROM Ipctmvt Ipctmvt;
```

```
WHERE Ipctmvt.mes = ?pmescaptu;
  AND Ipctmvt.borra = .F.;
  AND Ipctmvt.pactmvt <> 0;
  OR (Ipctmvt.pesmvt = .T.);
GROUP BY Ipctmvt.zona
```

Sin Duplicados

☞ **v_porcentdeczona**

```
SELECT DISTINCT Ipctmvt.zona, COUNT(Ipctmvt.codart), Ipctmvt.pactmvt,;
  Ipctmvt.pesmvt, Ipctmvt.decmvt;
FROM ipctmvt Ipctmvt;
WHERE Ipctmvt.borra = .F.;
  AND Ipctmvt.decmvt = ?pdeccaptu;
GROUP BY Ipctmvt.zona
```

Sin Duplicados

☞ **v_porcentmeszona**

```
SELECT DISTINCT Ipctmvt.zona, COUNT(Ipctmvt.codart), Ipctmvt.pactmvt,;
  Ipctmvt.pesmvt, Ipctmvt.mes;
FROM ipctmvt Ipctmvt;
WHERE Ipctmvt.borra = .F.;
  AND Ipctmvt.mes = ?pmescaptu;
GROUP BY Ipctmvt.zona
```

Sin Duplicados

☞ **v_prechpredil**

```
SELECT DISTINCT Histpre.codfte, Histpre.codart, Histpre.anomvt,;
  Histpre.mesmvt, Histpre.decmvt, Histpre.pesart, Histpre.panart,;
  Histpre.cantant, Histpre.pactmvt, Histpre.canrmvt, Histpre.pesmvt,;
  Histpre.crfmvt, Histpre.sinmmvt, Histpre.fcommvt, Histpre.marca,;
  Histpre.borra,;

IIF(Histpre.pactmvt#0.AND.Histpre.canrmvt#0,STR(((((Histpre.pactmvt/Histpre.canrmvt)/(Histpre.p
anart/Histpre.cantant))*100)-100),7,2),'') AS varia,;
  Histpre.pci;
FROM histpre Histpre;
WHERE Histpre.borra = .F.;
ORDER BY Histpre.anomvt, Histpre.mesmvt, Histpre.decmvt
```

Sin Duplicados

☞ **v_preciosh**

```
SELECT DISTINCT Histpre.codfte, Histpre.codart, Histpre.anomvt,;
  Histpre.mesmvt, Histpre.decmvt, Histpre.pesart, Histpre.panart,;
  Histpre.cantant, Histpre.pactmvt, Histpre.canrmvt, Histpre.pesmvt,;
  Histpre.crfmvt, Histpre.sinmmvt, Histpre.fcommvt, Histpre.marca,;
  Histpre.borra,;

IIF(Histpre.pactmvt#0.AND.Histpre.canrmvt#0,STR(((((Histpre.pactmvt/Histpre.canrmvt)/(Histpre.p
anart/Histpre.cantant))*100)-100),7,2),'') AS varia;
 FROM histpre Histpre;
 WHERE Histpre.codfte = ?pcodfteh;
   AND Histpre.codart = ?pcodarth;
 ORDER BY Histpre.anomvt, Histpre.mesmvt, Histpre.decmvt
```

Sin Duplicados

☞ **v_prefcm**

```
SELECT DISTINCT Ipctfcm.codfcom, Ipctfcm.nomfte, Ipctfcm.direfte,;
  Ipctfcm.telfte, Ipctfcm.codfte, Ipctfcm.marca;
 FROM ipctfcm Ipctfcm;
 WHERE Ipctfcm.borra = .F.;
 ORDER BY Ipctfcm.codfcom, Ipctfcm.codfte
```

Sin Duplicados

☞ **v_rango**

```
SELECT DISTINCT *;
 FROM rango Rango;
 WHERE Rango.ano = ?mAno;
   AND Rango.mes = ?mMes;
   AND Rango.codart = ?mArticulo;
   AND Rango.codfuente = ?mFuente;
 ORDER BY Rango.coddpto, Rango.ano, Rango.mes, Rango.codart,;
  Rango.codfuente, Rango.codtarifa
```

Sin Duplicados

☞ **v_rangoant**

```
SELECT DISTINCT *;
 FROM rango Rango;
 WHERE Rango.ano = ?mAnoAnt;
   AND Rango.mes = ?mMesAnt;
   AND Rango.codart = ?mArticulo;
   AND Rango.codfuente = ?mFuente;
 ORDER BY Rango.coddpto, Rango.ano, Rango.mes, Rango.codart,;
  Rango.codfuente, Rango.codtarifa
```

Sin Duplicados

☞ **v_rechesp**

```
SELECT DISTINCT Esptemp.codfte, Esptemp.codart, Esptemp.codesp,;
  Esptemp.nomesp, Esptemp.nombdet, Esptemp.nombdet1, Esptemp.impctv,;
  Esptemp.ano, Esptemp.mes, Esptemp.decmvt, Esptemp.marca, Esptemp.borra;
 FROM esptemp Esptemp;
 WHERE Esptemp.ano = ?recano;
   AND Esptemp.mes = ?recmes;
   AND Esptemp.borra = .F.;
 ORDER BY Esptemp.decmvt, Esptemp.impctv, Esptemp.codfte,;
  Esptemp.codart
```

Sin Duplicados

☞ **v_rechist**

```
SELECT DISTINCT Historia.codfte, Historia.codart, Historia.impctv,;
  Historia.panart, Historia.pactmvt, Historia.cmvmvt, Historia.crfmvt,;
  Historia.pesmvt, Historia.sinmmvt, Historia.fcommvt, Historia.canrmvt,;
  Historia.obsmvt, Historia.decmvt, Historia.total, Historia.cantant,;
  Historia.ano, Historia.mes, Historia.pesart, Historia.variacion,;
  Historia.marca, Historia.borra;
 FROM historia Historia;
 WHERE Historia.ano = ?recano;
   AND Historia.mes = ?recmes;
   AND Historia.borra = .F.;
 ORDER BY Historia.decmvt, Historia.impctv, Historia.codfte,;
  Historia.codart
```

Sin Duplicados

☞ **v_registro**

```
SELECT DISTINCT Registro.quien, Registro.npantalla, Registro.fechar,;
  Registro.act_ina, Registro.borra;
 FROM registro Registro;
 WHERE Registro.act_ina = .T.
```

Sin Duplicados

☞ **v_reimpctv**

```
SELECT DISTINCT Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv,;
  Ipctmvt.panart, Ipctmvt.pactmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt,;
  Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.canrmvt,;
  Ipctmvt.obsmvt, Ipctmvt.decmvt, Ipctmvt.cantant, Ipctmvt.ano, Ipctmvt.mes,;
```

```
 Ipctmvt.pesart, Ipctmvt.marca;
FROM ipctmvt Ipctmvt;
WHERE Ipctmvt.borra = .F.;
  AND Ipctmvt.impctv = ?reimpctv;
ORDER BY Ipctmvt.codart
```

Sin Duplicados

☞ **v_reimpctvh**

```
SELECT DISTINCT Historia.codfte, Historia.codart, Historia.impctv,;
  Historia.pactmvt, Historia.panart, Historia.cmvmvt, Historia.crfmvt,;
  Historia.pesmvt, Historia.sinmmvt, Historia.fcommvt, Historia.canrmvt,;
  Historia.obsmvt, Historia.decmvt, Historia.cantant, Historia.ano,;
  Historia.mes, Historia.pesart, Historia.marca;
FROM historia Historia;
WHERE Historia.borra = .F.;
  AND Historia.impctv = ?impctvh;
ORDER BY Historia.codart
```

Sin Duplicados

☞ **v_reimpdecesp**

```
SELECT DISTINCT Esptemp.codfte, Esptemp.codart, Esptemp.codesp,;
  Esptemp.nomesp, Esptemp.nombdet, Esptemp.nombdet1, Esptemp.impctv,;
  Esptemp.ano, Esptemp.mes, Esptemp.decmvt, Esptemp.marca, Esptemp.borra;
FROM esptemp Esptemp;
WHERE Esptemp.borra = .F.;
  AND Esptemp.ano = ?reimpano;
  AND Esptemp.mes = ?reimpmes;
  AND Esptemp.decmvt = ?reimpdec;
ORDER BY Esptemp.impctv, Esptemp.codfte, Esptemp.codart,;
  Esptemp.codesp
```

Sin Duplicados

☞ **v_reimpdechis**

```
SELECT DISTINCT Historia.codfte, Historia.codart, Historia.impctv,;
  Historia.panart, Historia.pactmvt, Historia.cmvmvt, Historia.crfmvt,;
  Historia.pesmvt, Historia.sinmmvt, Historia.fcommvt, Historia.canrmvt,;
  Historia.obsmvt, Historia.decmvt, Historia.cantant, Historia.ano,;
  Historia.mes, Historia.pesart, Historia.marca, Historia.variacion,;
  Ipcmfte.nombfte, Ipcmfte.direfte, Ipcmfte.telfte, Ipcmfte.zona;
FROM historia Historia, ipcmfte Ipcmfte;
WHERE Historia.codfte = Ipcmfte.codfte;
  AND (Historia.borra = .F.;
```

```
  AND Historia.ano = ?reimpano;
  AND Historia.mes = ?reimpmes;
  AND Historia.decmvt = ?reimpdec;
  AND Ipcmfte.zona IN
(?z_1,?z_2,?z_3,?z_4,?z_5,?z_6,?z_7,?z_8,?z_9,?z_10,?z_11,?z_12,?z_13,?z_14,?z_15,?z_16,
?z_17,?z_18,?z_19,?z_20,?z_21,?z_22,?z_23));
 ORDER BY Ipcmfte.zona, Historia.codfte
```

Sin Duplicados

☞ **v_reimpdecmvt**

```
SELECT DISTINCT Ipctmvt.zona, Ipctmvt.codfte, Ipctmvt.codart,;
  Ipctmvt.impctv, Ipctmvt.panart, Ipctmvt.fcommvt, Ipctmvt.decmvt,;
  Ipctmvt.cantant, Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.marca, Ipctmvt.pesart;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.borra = .F.;
  AND Ipctmvt.ano = ?reimpano;
  AND Ipctmvt.mes = ?reimpmes;
  AND Ipctmvt.decmvt = ?reimpdec;
  AND Ipctmvt.zona IN
(?z_1,?z_2,?z_3,?z_4,?z_5,?z_6,?z_7,?z_8,?z_9,?z_10,?z_11,?z_12,?z_13,?z_14,?z_15,?z_16,
?z_17,?z_18,?z_19,?z_20,?z_21,?z_22,?z_23);
 ORDER BY Ipctmvt.zona
```

Sin Duplicados

☞ **v_reimpifesp**

```
SELECT DISTINCT Esptemp.codfte, Esptemp.codart, Esptemp.codesp,;
  Esptemp.nomesp, Esptemp.nombdet, Esptemp.nombdet1, Esptemp.impctv,;
  Esptemp.ano, Esptemp.mes, Esptemp.decmvt, Esptemp.marca, Esptemp.borra;
 FROM esptemp Esptemp;
 WHERE Esptemp.borra = .F.;
  AND Esptemp.impctv >= ?pinicial;
  AND Esptemp.impctv <= ?pfinal;
 ORDER BY Esptemp.impctv, Esptemp.codfte, Esptemp.codart,;
  Esptemp.codesp
```

Sin Duplicados

☞ **v_reimpifh**

```
SELECT DISTINCT Historia.codfte, Historia.codart, Historia.impctv,;
  Historia.panart, Historia.pactmvt, Historia.cmvmvt, Historia.crfmvt,;
  Historia.pesmvt, Historia.sinmmvt, Historia.fcommvt, Historia.canrmvt,;
  Historia.obsmvt, Historia.decmvt, Historia.cantant, Historia.ano,;
```

```
 Historia.mes, Historia.pesart, Historia.marca;
FROM historia Historia;
WHERE Historia.borra = .F.;
  AND Historia.impctv >= ?pinicial;
  AND Historia.impctv <= ?pfinal;
ORDER BY Historia.impctv, Historia.codfte, Historia.codart
```

Sin Duplicados

☞ **v_reimpifmvt**

```
SELECT DISTINCT Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv,;
  Ipctmvt.panart, Ipctmvt.decmvt, Ipctmvt.cantant, Ipctmvt.ano, Ipctmvt.mes,;
  Ipctmvt.pesart, Ipctmvt.marca, Ipctmvt.fcommvt;
FROM ipctmvt Ipctmvt;
WHERE Ipctmvt.borra = .F.;
  AND Ipctmvt.impctv >= ?pinicial;
  AND Ipctmvt.impctv <= ?pfinal;
ORDER BY Ipctmvt.impctv, Ipctmvt.codfte, Ipctmvt.codart
```

Sin Duplicados

☞ **v_reimpzonaesp**

```
SELECT DISTINCT Esptemp.codfte, Esptemp.codart, Esptemp.codesp,;
  Esptemp.nomesp, Esptemp.nombdet, Esptemp.nombdet1, Esptemp.impctv,;
  Esptemp.ano, Esptemp.mes, Esptemp.decmvt, Esptemp.marca, Esptemp.borra;
FROM esptemp Esptemp;
WHERE Esptemp.borra = .F.;
  AND Esptemp.ano = ?zonaano;
  AND Esptemp.mes = ?zonames;
  AND Esptemp.decmvt = ?zonadec;
ORDER BY Esptemp.impctv, Esptemp.codart, Esptemp.codesp
```

Sin Duplicados

☞ **v_reimpzonahis**

```
SELECT DISTINCT Ipcmfte.zona, Historia.codfte, Historia.codart,;
  Historia.impctv, Historia.panart, Historia.pactmvt, Historia.cmvmvt,;
  Historia.crfmvt, Historia.pesmvt, Historia.sinmmvt, Historia.fcommvt,;
  Historia.canrmvt, Historia.obsmvt, Historia.decmvt, Historia.cantant,;
  Historia.ano, Historia.mes, Historia.pesart, Historia.marca,;
  Ipcmfte.nombfte, Ipcmfte.direfte, Ipcmfte.telfte;
FROM ipcmfte Ipcmfte, historia Historia;
WHERE Historia.codfte = Ipcmfte.codfte;
  AND (Historia.borra = .F.;
```

```
  AND Historia.ano = ?zonaano;
  AND Historia.mes = ?zonames;
  AND Ipcmfte.zona = ?pzona);
 ORDER BY Historia.impctv, Historia.codart
```

Sin Duplicados

☞ **v_reimpzonahis1**

```
SELECT DISTINCT Ipcmfte.zona, Historia.codfte, Historia.codart,;
  Historia.impctv, Historia.panart, Historia.pactmvt, Historia.cmvmvt,;
  Historia.crfmvt, Historia.pesmvt, Historia.sinmmvt, Historia.fcommvt,;
  Historia.canrmvt, Historia.obsmvt, Historia.decmvt, Historia.cantant,;
  Historia.ano, Historia.mes, Historia.pesart, Historia.marca,;
  Ipcmfte.nombfte, Ipcmfte.direfte, Ipcmfte.telfte;
 FROM ipcmfte Ipcmfte, historia Historia;
 WHERE Historia.codfte = Ipcmfte.codfte;
  AND (Historia.borra = .F.;
  AND Historia.ano = ?zonaano;
  AND Historia.mes = ?zonames;
  AND Historia.decmvt = ?zonadec;
  AND Ipcmfte.zona IN
(?z_1,?z_2,?z_3,?z_4,?z_5,?z_6,?z_7,?z_8,?z_9,?z_10,?z_11,?z_12,?z_13,?z_14,?z_15,?z_16,
?z_17,?z_18,?z_19,?z_20,?z_21,?z_22,?z_23));
 ORDER BY Ipcmfte.zona, Historia.codfte
```

Sin Duplicados

☞ **v_reimpzonamvt**

```
SELECT DISTINCT Ipcmfte.marca, Ipctmvt.codfte, Ipctmvt.codart,;
  Ipctmvt.impctv, Ipctmvt.panart, Ipctmvt.fcommvt, Ipctmvt.decmvt,;
  Ipctmvt.cantant, Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.pesart, Ipctmvt.marca,;
  Ipcmfte.zona, Ipcmfte.nombfte, Ipcmfte.direfte, Ipcmfte.telfte;
 FROM ipcmfte Ipcmfte, ipctmvt Ipctmvt;
 WHERE Ipctmvt.codfte = Ipcmfte.codfte;
  AND (Ipctmvt.borra = .F.;
  AND Ipctmvt.ano = ?zonaano;
  AND Ipctmvt.mes = ?zonames;
  AND Ipctmvt.decmvt = ?zonadec;
  AND Ipcmfte.zona = ?pzona);
 ORDER BY Ipctmvt.impctv, Ipctmvt.panart
```

Sin Duplicados

☞ **v_reimpzonamvt1**

```
SELECT DISTINCT Ipcmfte.marca, Ipctmvt.codfte, Ipctmvt.codart,;
  Ipctmvt.impctv, Ipctmvt.panart, Ipctmvt.fcommvt, Ipctmvt.decmvt,;
  Ipctmvt.cantant, Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.pesart, Ipcmfte.nombfte,;
  Ipcmfte.direfte, Ipcmfte.telfte, Ipctmvt.zona;
 FROM ipcmfte Ipcmfte, ipctmvt Ipctmvt;
 WHERE Ipctmvt.codfte = Ipcmfte.codfte;
   AND (Ipctmvt.borra = .F.;
   AND Ipctmvt.ano = ?zonaano;
   AND Ipctmvt.mes = ?zonames;
   AND Ipctmvt.decmvt = ?zonadec;
   AND Ipctmvt.zona IN
(?z_1,?z_2,?z_3,?z_4,?z_5,?z_6,?z_7,?z_8,?z_9,?z_10,?z_11,?z_12,?z_13,?z_14,?z_15,?z_16,
?z_17,?z_18,?z_19,?z_20,?z_21,?z_22,?z_23));
 ORDER BY Ipctmvt.zona
Sin Duplicados
```

☞ **v_resppa**

```
SELECT Ipcresp.codfte, Ipcresp.codart, Ipcresp.codesp, Ipcresp.nomesp,;
  Ipcresp.nombdet, Ipcresp.nombdet1, Ipcresp.tipo, Ipcresp.valor,;
  Ipcresp.orden, Ipcresp.marca, Ipcresp.borra;
 FROM ipcresp Ipcresp;
 WHERE Ipcresp.codfte = ?respafte;
   AND Ipcresp.codart = ?respart;
 ORDER BY Ipcresp.codfte, Ipcresp.codart, Ipcresp.orden, Ipcresp.codesp
```

☞ **v_rrctaplacri**

```
SELECT Ipcfunci.codfunci, Ipcfunci.nomfunci,;
  COUNT(Ipctmvt.codart) AS ctaftes, Ipctmvt.decmvt, Ipcfunci.codcargo;
 FROM ipctmvt Ipctmvt, ipcrfunc Ipcrfunc, ipcfunci Ipcfunci;
 WHERE Ipcrfunc.codfunci = Ipcfunci.codfunci;
   AND (Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
   AND Ipcrfunc.codcargo = 'R';
   AND Ipctmvt.borra = .F.;
   AND Ipctmvt.placri = .T.;
   AND Ipctmvt.zona > 0;
   AND Ipctmvt.decmvt = ?prrpdec;
   AND Ipcrfunc.borra = .F.);
 GROUP BY Ipctmvt.decmvt, Ipcfunci.codfunci
```

☞ **v_rrctaplasup**

```
SELECT Ipcfunci.codfunci, Ipcfunci.nomfunci,;
  COUNT(Ipctmvt.codart) AS ctaftes, Ipctmvt.decmvt, Ipcfunci.codcargo;
 FROM ipctmvt Ipctmvt, ipcrfunc Ipcrfunc, ipcfunci Ipcfunci;
 WHERE Ipcrfunc.codfunci = Ipcfunci.codfunci;
   AND (Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
```

```
  AND Ipcrfunc.codcargo = 'R';
  AND Ipctmvt.borra = .F.;
  AND Ipctmvt.supervisado = .T.;
  AND Ipctmvt.zona > 0;
  AND Ipctmvt.decmvt = ?prrpdec;
  AND Ipcrfunc.borra = .F.);
GROUP BY Ipctmvt.decmvt, Ipcfunci.codfunci
```

☞ **v_rrfuncimes**

```
SELECT Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.zona, Ipctmvt.decmvt,;
  Ipctmvt.codfte, Ipctmvt.codart, Ipcrfunc.codcargo, Ipcrfunc.codfunci,;
  Ipcfunci.nomfunci, Ipctmvt.sumr, Ipctmvt.spmr, Ipctmvt.siape,;
  Ipctmvt.siasi, Ipctmvt.siacr, Ipctmvt.siafn, Ipctmvt.sadi, Ipctmvt.sonas,;
  Ipctmvt.snae, Ipctmvt.cumr, Ipctmvt.cpmr, Ipctmvt.ciape, Ipctmvt.ciasi,;
  Ipctmvt.ciacr, Ipctmvt.ciafn, Ipctmvt.cadi, Ipctmvt.conas, Ipctmvt.cnae,;
  Ipctmvt.crea, Ipctmvt.crur, Ipctmvt.crpr, Ipctmvt.crape, Ipctmvt.crasi,;
  Ipctmvt.cracr, Ipctmvt.crafc, Ipctmvt.crob;
 FROM ipcfunci Ipcfunci, ipcrfunc Ipcrfunc, ipctmvt Ipctmvt;
 WHERE Ipcrfunc.codcargo = Ipcfunci.codcargo;
  AND Ipcrfunc.codfunci = Ipcfunci.codfunci;
  AND (VAL(Ipcrfunc.zonafunci) = Ipctmvt.zona;
  AND Ipctmvt.zona > 0;
  AND Ipcrfunc.codcargo = ?prrcargo;
  AND Ipcrfunc.borra = .F.);
 ORDER BY Ipcrfunc.codcargo, Ipctmvt.zona, Ipctmvt.codfte,;
  Ipctmvt.codart, Ipcrfunc.codfunci
```

☞ **v_rrindcalicri**

```
SELECT Ipcfunci.codfunci, Ipcfunci.nomfunci,;
  ALLTRIM(STR(SUM(((100-
Ipctmvt.totalcalicri)*Pondesgc.vlr_pon))/SUM(Pondesgc.vlr_pon),10,6)) AS indicadeca,;
  Ipctmvt.decmvt, Ipcfunci.codcargo;
 FROM ipctmvt Ipctmvt, Pondesgc Pondesgc, ipcrfunc Ipcrfunc,;
  ipcfunci Ipcfunci;
 WHERE Ipcrfunc.codfunci = Ipcfunci.codfunci;
  AND (Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
  AND Ipcrfunc.codcargo = 'C';
  AND Ipctmvt.borra = .F.;
  AND Ipctmvt.calicri = .T.;
  AND SUBSTR(Ipctmvt.codart,1,5) = SUBSTR(Pondesgc.codart,1,5);
  AND Pondesgc.codgri = '9';
  AND Ipctmvt.zona > 0;
  AND Ipctmvt.decmvt = ?prrpdec;
  AND Ipcrfunc.borra = .F.);
 GROUP BY Ipctmvt.decmvt, Ipcfunci.codfunci;
 ORDER BY Ipctmvt.decmvt
```

☞ **v_rrindcalicri_niv_ing**

```
SELECT Ipcfunci.codfunci, Ipcfunci.nomfunci,;
  ALLTRIM(STR(SUM(((100-
Ipctmvt.totalcalicri)*Pondesgc.vlr_pon))/SUM(Pondesgc.vlr_pon),10,6)) AS indicadeca,;
  Ipctmvt.decmvt, Ipcfunci.codcargo;
 FROM ipctmvt Ipctmvt, Pondesgc Pondesgc, ipcrfunc Ipcrfunc,;
  ipcfunci Ipcfunci;
 WHERE Ipcrfunc.codfunci = Ipcfunci.codfunci;
   AND (Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
   AND Ipcrfunc.codcargo = 'C';
   AND Ipctmvt.borra = .F.;
   AND Ipctmvt.calicri = .T.;
   AND SUBSTR(Ipctmvt.codfte,14,1)+SUBSTR(Ipctmvt.codart,1,5) =
Pondesgc.codgri+SUBSTR(Pondesgc.codart,1,5);
   AND Ipctmvt.zona > 0;
   AND Ipctmvt.decmvt = ?prrpdec;
   AND Ipcrfunc.borra = .F.);
 GROUP BY Ipctmvt.decmvt, Ipcfunci.codfunci;
 ORDER BY Ipctmvt.decmvt
```

☞ **v_rrindplacri**

```
SELECT Ipcfunci.codfunci, Ipcfunci.nomfunci,;
  ALLTRIM(STR(SUM(((100-
Ipctmvt.totalplacri)*Pondesgc.vlr_pon))/SUM(Pondesgc.vlr_pon),10,6)) AS indicadeca,;
  Ipctmvt.decmvt, Ipcfunci.codcargo;
 FROM ipctmvt Ipctmvt, Pondesgc Pondesgc, ipcrfunc Ipcrfunc,;
  ipcfunci Ipcfunci;
 WHERE Ipcrfunc.codfunci = Ipcfunci.codfunci;
   AND (Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
   AND Ipcrfunc.codcargo = 'R';
   AND Ipctmvt.borra = .F.;
   AND Ipctmvt.placri = .T.;
   AND SUBSTR(Ipctmvt.codart,1,5) = SUBSTR(Pondesgc.codart,1,5);
   AND Pondesgc.codgri = '9';
   AND Ipctmvt.zona > 0;
   AND Ipctmvt.decmvt = ?prrpdec;
   AND Ipcrfunc.borra = .F.);
 GROUP BY Ipctmvt.decmvt, Ipcfunci.codfunci;
 ORDER BY Ipctmvt.decmvt
```

☞ **v_rrindplacri_niv_ing**

```
SELECT Ipcfunci.codfunci, Ipcfunci.nomfunci,;
  ALLTRIM(STR(SUM(((100-
Ipctmvt.totalplacri)*Pondesgc.vlr_pon))/SUM(Pondesgc.vlr_pon),10,6)) AS indicadeca,;
```

```
 Ipctmvt.decmvt, Ipcfunci.codcargo;
FROM ipctmvt Ipctmvt, Pondesgc Pondesgc, ipcrfunc Ipcrfunc,;
 ipcfunci Ipcfunci;
WHERE Ipcrfunc.codfunci = Ipcfunci.codfunci;
  AND (Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
  AND Ipcrfunc.codcargo = 'R';
  AND Ipctmvt.borra = .F.;
  AND Ipctmvt.placri = .T.;
  AND SUBSTR(Ipctmvt.codfte,14,1)+SUBSTR(Ipctmvt.codart,1,5) =
Pondesgc.codgri+SUBSTR(Pondesgc.codart,1,5);
  AND Ipctmvt.zona > 0;
  AND Ipctmvt.decmvt = ?prrpdec;
  AND Ipcrfunc.borra = .F.);
GROUP BY Ipctmvt.decmvt, Ipcfunci.codfunci;
ORDER BY Ipctmvt.decmvt
```

☞ **v_rrindplasup**

```
SELECT Ipcfunci.codfunci, Ipcfunci.nomfunci,;
 ALLTRIM(STR(SUM(((100-
Ipctmvt.totalplasup)*Pondesgc.vlr_pon))/SUM(Pondesgc.vlr_pon),10,6)) AS indicadeca,;
 Ipctmvt.decmvt, Ipcfunci.codcargo;
FROM ipctmvt Ipctmvt, Pondesgc Pondesgc, ipcrfunc Ipcrfunc,;
 ipcfunci Ipcfunci;
WHERE Ipcrfunc.codfunci = Ipcfunci.codfunci;
  AND (Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
  AND Ipcrfunc.codcargo = 'R';
  AND Ipctmvt.borra = .F.;
  AND Ipctmvt.supervisado = .T.;
  AND SUBSTR(Ipctmvt.codart,1,5) = SUBSTR(Pondesgc.codart,1,5);
  AND Pondesgc.codgri = '9';
  AND Ipctmvt.zona > 0;
  AND Ipctmvt.decmvt = ?prrpdec;
  AND Ipcrfunc.borra = .F.);
GROUP BY Ipctmvt.decmvt, Ipcfunci.codfunci;
ORDER BY Ipctmvt.decmvt
```

☞ **v_rrindplasup_niv_ing**

```
SELECT Ipcfunci.codfunci, Ipcfunci.nomfunci,;
 ALLTRIM(STR(SUM(((100-
Ipctmvt.totalplasup)*Pondesgc.vlr_pon))/SUM(Pondesgc.vlr_pon),10,6)) AS indicadeca,;
 Ipctmvt.decmvt, Ipcfunci.codcargo;
FROM ipctmvt Ipctmvt, Pondesgc Pondesgc, ipcrfunc Ipcrfunc,;
 ipcfunci Ipcfunci;
WHERE Ipcrfunc.codfunci = Ipcfunci.codfunci;
  AND (Ipctmvt.zona = VAL(Ipcrfunc.zonafunci);
  AND Ipcrfunc.codcargo = 'R';
```

```
  AND Ipctmvt.borra = .F.;
  AND Ipctmvt.supervisado = .T.;
  AND SUBSTR(Ipctmvt.codfte,14,1)+SUBSTR(Ipctmvt.codart,1,5) =
Pondesgc.codgri+SUBSTR(Pondesgc.codart,1,5);
  AND Ipctmvt.zona > 0;
  AND Ipctmvt.decmvt = ?prrpdec;
  AND Ipcrfunc.borra = .F.);
 GROUP BY Ipctmvt.decmvt, Ipcfunci.codfunci;
 ORDER BY Ipctmvt.decmvt
```

☞ **v_rutabd**

```
SELECT DISTINCT *;
 FROM rutabd Rutabd;
 WHERE Rutabd.act_ina = .T.
```

Sin Duplicados

☞ **v_supervisado**

```
SELECT DISTINCT Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.impctv,;
  Ipctmvt.panart, Ipctmvt.pactmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt,;
  Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.canrmvt,;
  Ipctmvt.obsmvt, Ipctmvt.decmvt, Ipctmvt.total, Ipctmvt.cantant,;
  Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.variacion, Ipctmvt.marca,;
  Ipctmvt.movdiasup, Ipctmvt.plasup, Ipctmvt.diaasupervisar,;
  Ipctmvt.supervisado;
 FROM ipctmvt Ipctmvt;
 WHERE Ipctmvt.movdiasup BETWEEN ?pdia_superv1 AND ?pdia_superv2;
   AND Ipctmvt.diaasupervisar BETWEEN ?pdia_Asuperv1 AND ?pdia_Asuperv2;
   AND Ipctmvt.borra = .F.;
   AND Ipctmvt.supervisado = .T.;
 ORDER BY Ipctmvt.impctv, Ipctmvt.codart
```

Sin Duplicados

☞ **v_tarifa**

```
SELECT DISTINCT *;
 FROM tarifa Tarifa;
 WHERE Tarifa.ano = ?mAno;
   AND Tarifa.mes = ?mMes;
   AND Tarifa.codart = ?mArticulo;
   AND Tarifa.codfuente = ?mFuente;
 ORDER BY Tarifa.coddpto, Tarifa.ano, Tarifa.mes, Tarifa.codart,;
  Tarifa.codfuente, Tarifa.codestrato, Tarifa.codtarifa
```

Sin Duplicados

- **v_tarifaant**

```
SELECT DISTINCT *;
 FROM tarifa Tarifa;
 WHERE Tarifa.ano = ?mAnoAnt;
   AND Tarifa.mes = ?mMesAnt;
   AND Tarifa.codart = ?mArticulo;
   AND Tarifa.codfuente = ?mFuente;
 ORDER BY Tarifa.coddpto, Tarifa.ano, Tarifa.mes, Tarifa.codart,;
  Tarifa.codfuente, Tarifa.codestrato, Tarifa.codtarifa
```

Sin Duplicados

- **v_tarifast**

```
SELECT *;
 FROM tarifast Tarifast;
 WHERE Tarifast.codfte = ?mFuente;
   AND Tarifast.codart = ?mArticulo;
 ORDER BY Tarifast.codfte, Tarifast.codart, Tarifast.codmod,;
  Tarifast.codtar
```

- **v_tcaptu**

```
SELECT Ipctmvt.zona, Ipctmvt.impctv, Ipctmvt.codfte, Ipcmfte.nombfte,;
  Ipcmfte.direfte, Ipctmvt.codart, Ipcmart.nomart,;
  ALLTRIM(STR(Ipcmart.canbart))+ALLTRIM(Ipcmart.uniart) AS unidadbase,;
  Ipctmvt.panart, Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.crfmvt,;
  Ipctmvt.fcommvt, Ipctmvt.cmvmvt, Ipctmvt.canrmvt, Ipctmvt.pactmvt,;
  Ipctmvt.variacion, Ipcmart.canbart, Ipctmvt.decmvt, Ipctmvt.cantant;
 FROM ipctmvt Ipctmvt, ipcmfte Ipcmfte, ipcmart Ipcmart;
 WHERE Ipctmvt.codart = Ipcmart.codart;
   AND Ipctmvt.codfte = Ipcmfte.codfte;
   AND (Ipctmvt.borra = .F.;
   AND Ipctmvt.zona IN
(?z_1,?z_2,?z_3,?z_4,?z_5,?z_6,?z_7,?z_8,?z_9,?z_10,?z_11,?z_12,?z_13,?z_14,?z_15,?z_16,
?z_17,?z_18,?z_19,?z_20,?z_21,?z_22,?z_23));
 ORDER BY Ipctmvt.zona, Ipctmvt.impctv, Ipctmvt.codfte, Ipctmvt.codart
```

- **v_tododias**

```
SELECT Ipcdias.zonafunci, Ipcdias.codfunci, Ipcdias.anoplarec,;
  Ipcdias.mesplarec, Ipcdias.decplarec, Ipcdias.ctvplarec, Ipcdias.ctvdia,;
```

```
 Ipcdias.nrodia, Ipcdias.codfte, Ipcdias.ca, Ipcdias.ar, Ipcdias.observa,;
 Ipcdias.marca, Ipcdias.borra;
FROM ipcdias Ipcdias;
ORDER BY Ipcdias.anoplarec, Ipcdias.mesplarec, Ipcdias.decplarec,;
 Ipcdias.zonafunci, Ipcdias.ctvdia
```

☞ **v_totaltarifa**

```
SELECT *;
 FROM totaltarifa Totaltarifa;
 WHERE Totaltarifa.codfte = ?mFuente;
   AND Totaltarifa.codart = ?mArticulo;
 ORDER BY Totaltarifa.codfte, Totaltarifa.codart
```

☞ **v_variames**

```
SELECT DISTINCT Variames.coddpto, Variames.ano, Variames.mes,;
  Variames.codart, Variames.codfuente, Variames.codgri, Variames.precio,;
  Variames.variacion, Variames.iva, Variames.marca, Variames.variarecal,;
  Variames.precioreca, Ipcmfte.codfte, Ipcmfte.nombfte, Variames.totusers;
 FROM variames Variames, ipcmfte Ipcmfte;
 WHERE Variames.coddpto = SUBSTR(Ipcmfte.codfte,1,2);
   AND Variames.codgri = SUBSTR(Ipcmfte.codfte,14,1);
   AND Variames.codfuente = SUBSTR(Ipcmfte.codfte,17,4);
   AND Ipcmfte.borra = .F.;
 ORDER BY Variames.coddpto, Variames.ano, Variames.mes, Variames.codart,;
  Variames.codfuente, Variames.codgri
```

Sin Duplicados

☞ **v_zonacalicap2**

```
SELECT Ipctmvt.zona, Ipctmvt.codfunci, Ipctmvt.ano, Ipctmvt.mes,;
  Ipctmvt.decmvt, Ipctmvt.codart, Ipcmart.nomart, Ipctmvt.marca,;
  Ipctmvt.borra, Ipctmvt.calicap, Ipctmvt.cantant, Ipctmvt.panart,;
  Ipctmvt.canrmvt, Ipctmvt.pactmvt, Ipctmvt.pesart, Ipctmvt.pesmvt,;
  Ipctmvt.cmvmvt, Ipctmvt.crfmvt, Ipctmvt.sinmmvt, Ipctmvt.fcommvt,;
  Ipctmvt.obsmvt, Ipctmvt.variacion, Ipctmvt.impctv, Ipctmvt.diacalicap,;
  Ipctmvt.cadgf, Ipctmvt.caea, Ipctmvt.caur, Ipctmvt.capr, Ipctmvt.caape,;
  Ipctmvt.caasi, Ipctmvt.caacr, Ipctmvt.caafc, Ipctmvt.caob,;
  Ipctmvt.totalcalicap, Ipctmvt.obscalicap, Ipctmvt.fecharec;
 FROM ipctmvt Ipctmvt, ipcmart Ipcmart;
 WHERE Ipctmvt.codart = Ipcmart.codart;
   AND (Ipctmvt.zona = ?pzonacap;
   AND Ipctmvt.ano = ?panocap;
   AND Ipctmvt.mes = ?pmescap;
   AND Ipctmvt.decmvt = ?pdeccap;
```

```
  AND Ipctmvt.codfte = ?vcalifte6;
  AND Ipctmvt.calicap = .T.);
 ORDER BY Ipctmvt.codart
```

☞ **v_zonacalicri2**

```
SELECT Ipctmvt.zona, Ipctmvt.codfunci, Ipctmvt.ano, Ipctmvt.mes,;
  Ipctmvt.decmvt, Ipctmvt.codart, Ipctmvt.panart, Ipctmvt.pactmvt,;
  Ipctmvt.cmvmvt, Ipctmvt.crfmvt, Ipctmvt.pesmvt, Ipctmvt.sinmmvt,;
  Ipctmvt.fcommvt, Ipctmvt.canrmvt, Ipctmvt.obsmvt, Ipctmvt.cantant,;
  Ipctmvt.pesart, Ipctmvt.diacalicri, Ipctmvt.calicri, Ipctmvt.crdgf,;
  Ipctmvt.crea, Ipctmvt.crur, Ipctmvt.crpr, Ipctmvt.crape, Ipctmvt.crasi,;
  Ipctmvt.cracr, Ipctmvt.crafc, Ipctmvt.crob, Ipctmvt.totalcalicri,;
  Ipctmvt.obscalicri, Ipctmvt.marca, Ipctmvt.borra,;

IIF(Ipctmvt.panart#0.AND.Ipctmvt.pactmvt#0.AND.Ipctmvt.canrmvt#0.AND.Ipctmvt.cantant#0,ALL
TRIM(STR((((Ipctmvt.pactmvt/Ipctmvt.canrmvt)/(Ipctmvt.panart/Ipctmvt.cantant)*100)-100),7,2)),'')
AS artivaria,;
  Ipcmart.nomart, Ipctmvt.codfte, Ipctmvt.impctv;
 FROM ipctmvt Ipctmvt, ipcmart Ipcmart;
 WHERE Ipcmart.codart = Ipctmvt.codart;
  AND (Ipctmvt.zona = ?pzonacri;
  AND Ipctmvt.ano = ?panoplacri;
  AND Ipctmvt.mes = ?pmesplacri;
  AND Ipctmvt.decmvt = ?pdecplacri;
  AND Ipctmvt.codfte = ?vcalifte5;
  AND Ipctmvt.calicri = .T.);
 ORDER BY Ipctmvt.diacalicri, Ipctmvt.codart
```

☞ **v_zonaplacap2**

```
SELECT Ipctmvt.zona, Ipctmvt.codfunci, Ipctmvt.ano, Ipctmvt.mes,;
  Ipctmvt.decmvt, Ipctmvt.ctvdia, Ipctmvt.diasup, Ipctmvt.plasup,;
  Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.totalplasup, Ipctmvt.imsxf,;
  Ipctmvt.pe, Ipctmvt.si, Ipctmvt.cr, Ipctmvt.fn, Ipctmvt.obsplasup,;
  Ipctmvt.marca, Ipctmvt.borra, Ipctmvt.placri, Ipctmvt.calicri,;
  Ipctmvt.calisup, Ipctmvt.calicap, Ipctmvt.cantant, Ipctmvt.panart,;
  Ipctmvt.canrmvt, Ipctmvt.pactmvt, Ipctmvt.pesart, Ipctmvt.pesmvt,;
  Ipctmvt.cmvmvt, Ipctmvt.crfmvt, Ipctmvt.sinmmvt, Ipctmvt.fcommvt,;
  Ipctmvt.obsmvt,;

IIF(Ipctmvt.panart#0.AND.Ipctmvt.pactmvt#0.AND.Ipctmvt.canrmvt#0.AND.Ipctmvt.cantant#0,ALL
TRIM(STR((((Ipctmvt.pactmvt/Ipctmvt.canrmvt)/(Ipctmvt.panart/Ipctmvt.cantant)*100)-100),7,2)),'')
AS artivaria,;
  Ipctmvt.variacion, Ipctmvt.impctv, Ipctmvt.diacalicap, Ipctmvt.cadgf,;
  Ipctmvt.caea, Ipctmvt.caur, Ipctmvt.capr, Ipctmvt.caape, Ipctmvt.caasi,;
```

```
 Ipctmvt.caacr, Ipctmvt.caafc, Ipctmvt.caob, Ipctmvt.totalcalicap,;
 Ipctmvt.obscalicap, Ipctmvt.supervisado, Ipctmvt.fechasup;
FROM ipctmvt Ipctmvt;
WHERE Ipctmvt.zona = ?pzonacap;
  AND Ipctmvt.ano = ?panocap;
  AND Ipctmvt.mes = ?pmescap;
  AND Ipctmvt.decmvt = ?pdeccap;
ORDER BY Ipctmvt.diasup, Ipctmvt.codfte, Ipctmvt.codart
```

- **v_zonaplacri2**

```
SELECT Ipctmvt.zona, Ipctmvt.codfunci, Ipctmvt.ano, Ipctmvt.mes,;
 Ipctmvt.decmvt, Ipctmvt.ctvdia, Ipctmvt.diasup, Ipctmvt.diacri,;
 Ipctmvt.plasup, Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.totalplasup,;
 Ipctmvt.imsxf, Ipctmvt.pe, Ipctmvt.si, Ipctmvt.cr, Ipctmvt.fn,;
 Ipctmvt.obsplasup, Ipctmvt.marca, Ipctmvt.borra, Ipctmvt.placri,;
 Ipctmvt.calicri, Ipctmvt.calisup, Ipctmvt.variacion, Ipctmvt.calicap,;
 Ipctmvt.cantant, Ipctmvt.panart, Ipctmvt.canrmvt, Ipctmvt.pactmvt,;
 Ipctmvt.pesart, Ipctmvt.pesmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt,;
 Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.obsmvt, Ipctmvt.cumr,;
 Ipctmvt.cpmr, Ipctmvt.ciape, Ipctmvt.ciasi, Ipctmvt.ciacr, Ipctmvt.ciafn,;
 Ipctmvt.cadi, Ipctmvt.conas, Ipctmvt.cnae, Ipctmvt.crdgf,;
 Ipctmvt.totalplacri, Ipctmvt.obsplacri, Ipctmvt.crea, Ipctmvt.crur,;
 Ipctmvt.crpr, Ipctmvt.crape, Ipctmvt.crasi, Ipctmvt.cracr, Ipctmvt.crafc,;
 Ipctmvt.crob, Ipctmvt.impctv, Ipctmvt.supervisado,;
 Ipctmvt.diaasupervisar, Ipctmvt.sianalisis, Ipctmvt.diaanalisis,;
 Ipctmvt.diacalicri;
FROM ipctmvt Ipctmvt;
WHERE Ipctmvt.zona = ?pzonacri;
  AND Ipctmvt.ano = ?panoplacri;
  AND Ipctmvt.mes = ?pmesplacri;
  AND Ipctmvt.decmvt = ?pdecplacri;
ORDER BY Ipctmvt.diasup, Ipctmvt.codfte, Ipctmvt.codart
```

- **v_zonaplacri3**

```
SELECT Ipctmvt.zona, Ipctmvt.codfunci, Ipctmvt.ano, Ipctmvt.mes,;
 Ipctmvt.decmvt, Ipctmvt.ctvdia, Ipctmvt.diasup, Ipctmvt.diacri,;
 Ipctmvt.plasup, Ipctmvt.codart, Ipctmvt.marca, Ipctmvt.borra,;
 Ipctmvt.placri,;

IIF(Ipctmvt.panart#0.AND.Ipctmvt.pactmvt#0.AND.Ipctmvt.canrmvt#0.AND.Ipctmvt.cantant#0,ALL
TRIM(STR((((Ipctmvt.pactmvt/Ipctmvt.canrmvt)/(Ipctmvt.panart/Ipctmvt.cantant)*100)-100),7,2)),'')
AS artivaria,;
 Ipctmvt.cumr, Ipctmvt.cpmr, Ipctmvt.ciape, Ipctmvt.ciasi, Ipctmvt.ciacr,;
 Ipctmvt.ciafn, Ipctmvt.cadi, Ipctmvt.conas, Ipctmvt.cnae, Ipctmvt.crdgf,;
```

```
 Ipctmvt.totalplacri, Ipctmvt.obsplacri, Ipctmvt.supervisado,;
 Ipctmvt.diaasupervisar, Ipctmvt.sianalisis, Ipctmvt.diaanalisis,;
 Ipcmart.nomart, Ipctmvt.codfte, Ipctmvt.impctv, Ipctmvt.spmr, Ipctmvt.pmr,;
 Ipctmvt.sumr, Ipctmvt.siape, Ipctmvt.siasi, Ipctmvt.siacr, Ipctmvt.siafn,;
 Ipctmvt.sadi, Ipctmvt.sonas, Ipctmvt.snae, Ipctmvt.totalplasup,;
 Ipctmvt.sdgf, Ipctmvt.obsplasup, Ipctmvt.fechasup, Ipctmvt.obssupersado;
FROM ipctmvt Ipctmvt, ipcmart Ipcmart;
WHERE Ipcmart.codart = Ipctmvt.codart;
  AND (Ipctmvt.zona = ?pzonacri;
  AND Ipctmvt.ano = ?panoplacri;
  AND Ipctmvt.mes = ?pmesplacri;
  AND Ipctmvt.decmvt = ?pdecplacri;
  AND Ipctmvt.codfte = ?vcalifte4);
ORDER BY Ipctmvt.diasup, Ipctmvt.codart
```

☞ **v_zonaplacriccc**

```
SELECT Ipctmvt.zona, Ipctmvt.codfunci, Ipctmvt.ano, Ipctmvt.mes,;
 Ipctmvt.decmvt, Ipctmvt.ctvdia, Ipctmvt.diasup, Ipctmvt.diacri,;
 Ipctmvt.plasup, Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.totalplasup,;
 Ipctmvt.imsxf, Ipctmvt.pe, Ipctmvt.si, Ipctmvt.cr, Ipctmvt.fn,;
 Ipctmvt.obsplasup, Ipctmvt.marca, Ipctmvt.borra, Ipctmvt.placri,;
 Ipctmvt.calicri, Ipctmvt.calisup, Ipctmvt.variacion, Ipctmvt.calicap,;
 Ipctmvt.cantant, Ipctmvt.panart, Ipctmvt.canrmvt, Ipctmvt.pactmvt,;
 Ipctmvt.pesart, Ipctmvt.pesmvt, Ipctmvt.cmvmvt, Ipctmvt.crfmvt,;
 Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.obsmvt, Ipctmvt.cumr,;
 Ipctmvt.cpmr, Ipctmvt.ciape, Ipctmvt.ciasi, Ipctmvt.ciacr, Ipctmvt.ciafn,;
 Ipctmvt.cadi, Ipctmvt.conas, Ipctmvt.cnae, Ipctmvt.crdgf,;
 Ipctmvt.totalplacri, Ipctmvt.obsplacri, Ipctmvt.crea, Ipctmvt.crur,;
 Ipctmvt.crpr, Ipctmvt.crape, Ipctmvt.crasi, Ipctmvt.cracr, Ipctmvt.crafc,;
 Ipctmvt.crob, Ipctmvt.impctv, Ipctmvt.supervisado,;
 Ipctmvt.diaasupervisar, Ipctmvt.sianalisis, Ipctmvt.diaanalisis,;
 Ipctmvt.diacalicri;
FROM ipctmvt Ipctmvt;
WHERE Ipctmvt.ano = ?panoplacri;
  AND Ipctmvt.mes = ?pmesplacri;
  AND Ipctmvt.calicri = .T.;
ORDER BY Ipctmvt.codfte, Ipctmvt.codart
```

☞ **v_zonaplare2**

```
SELECT Ipctmvt.zona, Ipctmvt.codfunci, Ipctmvt.ano, Ipctmvt.mes,;
 Ipctmvt.decmvt, Ipctmvt.ctvoruta, Ipctmvt.ruta, Ipctmvt.codfte,;
 Ipctmvt.codart, Ipctmvt.marca, Ipctmvt.borra,;
 SUBSTR(Ipctmvt.codfte,6,4) AS sectorfte,;
 SUBSTR(Ipctmvt.codfte,10,2) AS seccion,;
 SUBSTR(Ipctmvt.codfte,12,2) AS manzafte,;
 SUBSTR(Ipctmvt.codfte,14,1) AS grpfte,;
```

```
 SUBSTR(Ipctmvt.codfte,15,2) AS clafte;
FROM ipctmvt Ipctmvt;
WHERE Ipctmvt.zona = ?pzonafunci1;
  AND Ipctmvt.ano = ?panoh;
  AND Ipctmvt.mes = ?pmesh;
  AND Ipctmvt.decmvt = ?pdech;
ORDER BY Ipctmvt.codfte
```

☞ **v_zonaplare3**

```
SELECT Ipctmvt.zona, Ipctmvt.codfunci, Ipctmvt.ano, Ipctmvt.mes,;
  Ipctmvt.decmvt, Ipctmvt.ctvoruta, Ipctmvt.ruta, Ipctmvt.codfte,;
  Ipctmvt.marca, Ipctmvt.borra, SUBSTR(Ipctmvt.codfte,6,4) AS sectorfte,;
  SUBSTR(Ipctmvt.codfte,10,2) AS seccion,;
  SUBSTR(Ipctmvt.codfte,12,2) AS manzafte,;
  SUBSTR(Ipctmvt.codfte,14,1) AS grpfte,;
  SUBSTR(Ipctmvt.codfte,15,2) AS clafte;
FROM ipctmvt Ipctmvt;
WHERE Ipctmvt.zona = ?pzonafunci3;
  AND Ipctmvt.ano = ?panoh;
  AND Ipctmvt.mes = ?pmesh;
  AND Ipctmvt.decmvt = ?pdech;
ORDER BY Ipctmvt.codfte
```

☞ **v_zonaplasup2**

```
SELECT Ipctmvt.zona, Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt,;
  Ipctmvt.diasup, Ipctmvt.codfte, Ipctmvt.codart, Ipctmvt.plasup,;
  Ipctmvt.totalplasup, Ipctmvt.imsxf, Ipctmvt.pe, Ipctmvt.si, Ipctmvt.cr,;
  Ipctmvt.fn, Ipctmvt.obsplasup, Ipctmvt.marca, Ipctmvt.borra, Ipctmvt.sumr,;
  Ipctmvt.spmr, Ipctmvt.siape, Ipctmvt.siasi, Ipctmvt.siacr, Ipctmvt.siafn,;
  Ipctmvt.sadi, Ipctmvt.sonas, Ipctmvt.snae, Ipctmvt.sdgf,;
  Ipctmvt.supervisado;
FROM ipctmvt Ipctmvt;
WHERE Ipctmvt.supervisado = .T.;
  AND Ipctmvt.zona = ?pzonaplasup;
  AND Ipctmvt.ano = ?panoplarec;
  AND Ipctmvt.mes = ?pmesplarec;
  AND Ipctmvt.decmvt = ?pdecplarec
```

☞ **v_zonaplasup3**

```
SELECT Ipctmvt.zona, Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt,;
```

```
 Ipctmvt.diasup, Ipctmvt.codfte, Ipctmvt.codart, Ipcmart.nomart,;
 Ipctmvt.plasup, Ipctmvt.totalplasup, Ipctmvt.imsxf, Ipctmvt.pe, Ipctmvt.si,;
 Ipctmvt.cr, Ipctmvt.fn, Ipctmvt.obsplasup, Ipctmvt.marca, Ipctmvt.borra,;
 Ipctmvt.pesart, Ipctmvt.cantant, Ipctmvt.panart, Ipctmvt.cmvmvt,;
 Ipctmvt.crfmvt, Ipctmvt.pesmvt, Ipctmvt.sinmmvt, Ipctmvt.fcommvt,;
 Ipctmvt.obsmvt, Ipctmvt.canrmvt, Ipctmvt.pactmvt, Ipctmvt.sumr,;
 Ipctmvt.spmr, Ipctmvt.siape, Ipctmvt.siasi, Ipctmvt.siacr, Ipctmvt.siafn,;
 Ipctmvt.sadi, Ipctmvt.sonas, Ipctmvt.snae, Ipctmvt.sdgf,;
 SUBSTR(DTOC(Ipctmvt.movdiasup),4,2) AS diasuper, Ipctmvt.supervisado,;
 Ipctmvt.impctv;
FROM ipctmvt Ipctmvt, ipcmart Ipcmart;
WHERE Ipcmart.codart = Ipctmvt.codart;
  AND (Ipctmvt.supervisado = .T.;
  AND Ipctmvt.zona = ?pzonaplasup;
  AND Ipctmvt.ano = ?panoplarec;
  AND Ipctmvt.mes = ?pmesplarec;
  AND Ipctmvt.decmvt = ?pdecplarec;
  AND Ipctmvt.codfte = ?vcalifte3);
ORDER BY Ipctmvt.codfte, Ipctmvt.codart
```

☞ **v_zonaplasup4**

```
* Utilizado Formulario Supervisión
SELECT Ipctmvt.zona, Ipctmvt.ano, Ipctmvt.mes, Ipctmvt.decmvt,;
 Ipctmvt.diasup, Ipctmvt.codfte, Ipctmvt.codart,;
 ALLTRIM(STR(Ipcmart.canbart))+ALLTRIM(Ipcmart.uniart) AS cantunib,;
 Ipcmart.nomart, Ipctmvt.plasup, Ipctmvt.totalplasup, Ipctmvt.imsxf,;
 Ipctmvt.pe, Ipctmvt.si, Ipctmvt.cr, Ipctmvt.fn, Ipctmvt.obsplasup,;
 Ipctmvt.marca, Ipctmvt.borra, Ipctmvt.pesart, Ipctmvt.cantant,;
 Ipctmvt.panart, Ipctmvt.cmvmvt, Ipctmvt.crfmvt, Ipctmvt.pesmvt,;
 Ipctmvt.sinmmvt, Ipctmvt.fcommvt, Ipctmvt.obsmvt, Ipctmvt.canrmvt,;
 Ipctmvt.pactmvt, Ipctmvt.sumr, Ipctmvt.spmr, Ipctmvt.siape, Ipctmvt.siasi,;
 Ipctmvt.siacr, Ipctmvt.siafn, Ipctmvt.sadi, Ipctmvt.sonas, Ipctmvt.snae,;
 Ipctmvt.sdgf, Ipctmvt.movdiasup, Ipctmvt.supervisado,;
 Ipctmvt.diaasupervisar, Ipctmvt.total, Ipctmvt.fechaso, Ipctmvt.impctv,;
 SUBSTR(Ipctmvt.codfte,14,1) AS grupo,;
 SUBSTR(Ipctmvt.codfte,15,2) AS clase, Ipctmvt.variacion,;
 IIF(.NOT.EMPTY(Ipctmvt.obsplasup).AND..NOT.ISNULL(Ipctmvt.obsplasup),.T.,.F.) AS
lgcobsplasup,;
 Ipctmvt.demas, Ipctmvt.fechasup, Ipctmvt.diaarec;
FROM ipctmvt Ipctmvt, ipcmart Ipcmart;
WHERE Ipcmart.codart = Ipctmvt.codart;
  AND (Ipctmvt.supervisado = .T.;
  AND Ipctmvt.zona = ?pzonaplasup;
  AND Ipctmvt.ano = ?panoplarec;
  AND Ipctmvt.mes = ?pmesplarec;
  AND Ipctmvt.decmvt = ?pdecplarec;
  AND Ipctmvt.codfte = ?vcalifte3;
```

```
  AND Ipctmvt.borra = .F.);
 ORDER BY Ipctmvt.codfte, Ipctmvt.diaasupervisar, Ipctmvt.codart
```

☞ **v_zonas**

```
SELECT DISTINCT Ipcmfte.zona;
 FROM ipcmfte Ipcmfte;
 WHERE Ipcmfte.borra = .F.;
  AND Ipcmfte.zona <> 0;
 ORDER BY Ipcmfte.zona
```

Sin Duplicados

☞ **v_zonas1**

```
SELECT DISTINCT Ipcmfte.zona;
 FROM ipcmfte Ipcmfte;
 WHERE Ipcmfte.borra = .F.;
  AND Ipcmfte.zona <> 0;
 ORDER BY Ipcmfte.zona
```

Sin Duplicados

☞ **v_zonas10**

```
SELECT DISTINCT Ipcmfte.zona;
 FROM ipcmfte Ipcmfte;
 WHERE Ipcmfte.borra = .F.;
  AND Ipcmfte.zona <> 0;
 ORDER BY Ipcmfte.zona
```

Sin Duplicados

☞ **v_zonas2**

```
SELECT DISTINCT Ipcmfte.zona;
 FROM ipcmfte Ipcmfte;
 WHERE Ipcmfte.borra = .F.;
  AND Ipcmfte.zona <> 0;
 ORDER BY Ipcmfte.zona
```

Sin Duplicados

☞ **v_zonas3**

```
SELECT DISTINCT Ipcmfte.zona;
 FROM ipcmfte Ipcmfte;
 WHERE Ipcmfte.borra = .F.;
```

```
  AND Ipcmfte.zona <> 0;
 ORDER BY Ipcmfte.zona
```

Sin Duplicados

☞ **v_zonas4**

```
SELECT DISTINCT Ipcmfte.zona;
 FROM ipcmfte Ipcmfte;
 WHERE Ipcmfte.borra = .F.;
  AND Ipcmfte.zona <> 0;
 ORDER BY Ipcmfte.zona
```

Sin Duplicados

☞ **v_zonas5**

```
SELECT DISTINCT Ipcmfte.zona;
 FROM ipcmfte Ipcmfte;
 WHERE Ipcmfte.borra = .F.;
 ORDER BY Ipcmfte.zona
```

Sin Duplicados

☞ **v_zonas555**

```
SELECT DISTINCT Ipcmfte.zona;
 FROM ipcmfte Ipcmfte;
 WHERE Ipcmfte.borra = .F.;
  AND Ipcmfte.zona <> 0;
 ORDER BY Ipcmfte.zona
```

Sin Duplicados

☞ **v_zonas6**

```
SELECT DISTINCT Ipcmfte.zona;
 FROM ipcmfte Ipcmfte, ipctmvt Ipctmvt;
 WHERE Ipcmfte.codfte = Ipctmvt.codfte;
  AND (Ipcmfte.borra = .F.;
  AND Ipctmvt.decmvt = ?m.pdecplarec;
```

```
  AND Ipctmvt.supervisado = .T.;
  AND Ipctmvt.ano = ?m.VCtvAno;
  AND Ipctmvt.mes = ?m.VCtvMes);
ORDER BY Ipcmfte.zona
```

Sin Duplicados

☞ **v_zonas7**

```
SELECT DISTINCT Ipcmfte.zona;
 FROM ipcmfte Ipcmfte;
 WHERE Ipcmfte.borra = .F.;
   AND Ipcmfte.zona <> 0;
 ORDER BY Ipcmfte.zona
```

Sin Duplicados

☞ **v_zonas8**

```
SELECT DISTINCT Ipcmfte.zona;
 FROM ipcmfte Ipcmfte;
 WHERE Ipcmfte.borra = .F.;
   AND Ipcmfte.zona <> 0;
 ORDER BY Ipcmfte.zona
```

Sin Duplicados

☞ **v_zonas9**

```
SELECT DISTINCT Ipcmfte.zona;
 FROM ipcmfte Ipcmfte;
 WHERE Ipcmfte.borra = .F.;
   AND Ipcmfte.zona <> 0;
 ORDER BY Ipcmfte.zona
```

Sin Duplicados

☞ **v_zonasmvtxdecada**

```
SELECT DISTINCT Ipctmvt.zona;
 FROM ipctmvt Ipctmvt;
```

```
WHERE ALLTRIM(STR(Ipctmvt.decmvt,1,0)) BETWEEN ?pdecmvt AND ?pdecmvt1;
  AND Ipctmvt.borra = .F.;
ORDER BY Ipctmvt.zona
```

Sin Duplicados

4 PROGRAMAS (PRG)

4.1 IPC2001_APP.PRG && Programa Principal, es el primero que se ejecuta

```
&& Definimos variables publicas a utilizar en el aplicativo
&& Encriptamos el nombre de usuario

PUBLIC RAIZ,CONJUNTO,ENVIADODE,LACIUDAD,CODLACIUDAD, PNIVEL, NBLANCOS,
HABILITAR

HABILITAR = .F.

&& Definimos icono a mostrar en la pantalla principal
_SCREEN.ICON="ipc.ico"
&& Definimos variables publicas a utilizar en el aplicativo
&& Encriptamos el nombre de usuario
PUBLIC SISTOPE,ANOPROCESO,MESPROCESO,DEPARTAMENTO,MUNICIPIO,PORCIUDAD

&& Declaramos los valores a enviar a las API de Windows como parametros
&& para los procesos en los cuales apoyan
DECLARE INTEGER WinHelp IN Win32API ;
INTEGER HWND , STRING lpHelpFile , INTEGER wCommand ,ANY dwData

DECLARE INTEGER PostMessage IN Win32API ;
INTEGER nWnd, INTEGER nMsg, INTEGER nParam ,INTEGER nParam

DECLARE INTEGER WinExec IN WIN32API AS WINEXEC;
STRING lpCmdLine,INTEGER nCmdShow

*-- Instrucciones DECLARE DLL para leer/escribir en archivos INI privados
DECLARE INTEGER GetPrivateProfileString IN Win32API  AS GetPrivStr ;
STRING cSection, STRING cKey, STRING cDefault, STRING @cBuffer, ;
INTEGER nBufferSize, STRING cINIFile
```

```
DECLARE INTEGER WritePrivateProfileString IN Win32API AS WritePrivStr ;
STRING cSection, STRING cKey, STRING cValue, STRING cINIFile

*-- Instrucciones DECLARE DLL para leer/escribir en el registro del sistema
DECLARE INTEGER RegOpenKeyEx IN Win32API ;
INTEGER nKey, STRING @cSubKey, INTEGER nReserved,;
INTEGER nAccessMask, INTEGER @nResult

DECLARE INTEGER RegQueryValueEx IN Win32API ;
INTEGER nKey, STRING cValueName, INTEGER nReserved,;
INTEGER @nType, STRING @cBuffer, INTEGER @nBufferSize

DECLARE INTEGER RegCloseKey IN Win32API ;
INTEGER nKey

*-- Instrucción DECLARE DLL para la función de API GetProfileString de Windows 3.1
DECLARE INTEGER GetProfileString IN Win32API AS GetProStr ;
STRING cSection, STRING cKey, STRING cDefault, ;
STRING @cBuffer, INTEGER nBuffer

+-------IF TYPE([APP_GLOBAL.Class]) = "C" AND ;
|                UPPER(APP_GLOBAL.CLASS) == UPPER(APP_CLASSNAME)
|          MESSAGEBOX(APP_ALREADY_RUNNING_LOC,48, ;
|                APP_GLOBAL.CCAPTION )
|     +-------IF VARTYPE(APP_GLOBAL.OFRAME) = "O"
|     |          APP_GLOBAL.OFRAME.SHOW()
|     +-------ENDIF
|          RETURN
|
+-------ENDIF

 RELEASE APP_GLOBAL
 PUBLIC  APP_GLOBAL
&& Ejecutamos el programa que nos devuelve la ruta desde donde se ejecuta el aplicativo
DO BOOT.PRG
LOCAL LCLASTSETTALK, LLAPPRAN, LNSECONDS, LOSPLASH
LOCAL ARRAY LACHECK[1]

LCLASTSETTALK=SET("TALK")
LOSPLASH = .NULL.

&& Configuramos para el manejo de fechas
SET DATE TO DMY
SET TALK OFF
&& Definimos la libreria principal a utilizar para que nos apoye en los proceso internos
SET LIBRARY TO FOXTOOLS.FLL
_SCREEN.WINDOWSTATE = 2
```

```
&& Inicializamos la variable que utilizamos para almacenar el nivel de acceso de && un usuario que
intenta entrar el el aplicativo
PNIVEL = ' '
#IFDEF APP_SPLASHCLASS

        +-------IF NOT EMPTY(APP_SPLASHCLASS)
        |           LOSPLASH = NEWOBJECT(APP_SPLASHCLASS, APP_SPLASHCLASSLIB)
        |     +-------IF VARTYPE(LOSPLASH) = "O"
        |     |           LNSECONDS = SECONDS()
        |     |           LOSPLASH.SHOW()
        |     +-------ENDIF
        +-------ENDIF

+-------#ENDIF

APP_GLOBAL = NEWOBJECT(APP_CLASSNAME, APP_CLASSLIB)

&& Ejecutamos el formulario donde se solicita el nombre y clave de usuario
&& para validar si esta registrado o no y asi mismo permitirle el ingreso
&& o no
      DO FORM SEGURO
&& Si nos devuelve algun valor en la variable, es decir que nivel de acceso el && usuario permitimos
que se ejecute
+-------IF PNIVEL <> ' '
|     +-------IF NOT APP_GLOBAL.SHOW()
|     |
|     |     +-------IF TYPE([APP_GLOBAL.Name]) = "C"
|     |     |           MESSAGEBOX(APP_CANNOT_RUN_LOC,16, ;
|     |     |                 APP_GLOBAL.CCAPTION )
|     |     |           APP_GLOBAL.RELEASE()
|     |     +-------ELSE
|     |     |           MESSAGEBOX(APP_CANNOT_RUN_LOC,16)
|     |     +-------ENDIF
|     |
|     +-------ELSE
|     |           LLAPPRAN = .T.
|     +-------ENDIF
+-------ENDIF
```

4.2 ARTI_PRE.PRG && Devuelve el numero de articulos prediligenciados por fuente para que salga impreso en el formulario

```
LPARAMETERS FUENTES1
&&Al momento de prediligenciamiento recibimos parametros y
&&devolvemos la informacion correspondiente al numero de
&&articulos que salieron prediligenciados para cada fuente

&&Definimos y cargamos variables para almacenar el nombre de los archivos
	LOCAL CALIAS, NARTI
	NARTI=0
	CALIAS=ALIAS()

&&seleccionamos de el archivo de movimiento la informacion
&&correspondiente
		SELECT DISTINCT CODART AS CRESP FROM CHE2 WHERE CHE2.CODFTE =
		FUENTES1 INTO CURSOR N_ARTICULOS
	fuentes1 into cursor n_articulos

	SELE N_ARTICULOS
	STORE RECCOUNT() TO NARTI
&&Segun los parametros que hayamos recibido asi mismo devolvemos
&&los datos
	SELEC (CALIAS)
	RETURN NARTI
```

4.3 BOOT.PRG && Captura la ruta desde donde se esta ejecutando con la idea que se pueda procesar desde cualquir carpeta o unidad logica

```
	SET PATH TO FORMS;  GRAPHICS;  DATA; HELP; INCLUDE; LIBS; MENUS; PROGS;
REPORTS

	LCMAINCLASSLIB="ipc2001_app"
	SET CLASSLIB TO (LCMAINCLASSLIB) ADDITIVE

	SET TALK OFF
	SET NOTIFY OFF
	SET DELETED ON
	SET EXACT ON
	SET CENTURY ON
	SET DATE TO DMY
	SET EXCLUSIVE OFF
	SET MULTILOCKS ON
```

4.4 CIUDAD.PRG && Devuelve el nombre de la ciudad según los parámetros que recibe

```
	LPARAMETERS CFUENTES
&&Recibimos los parametros de departamento y municipio
&&para devolver el nombre de la ciudad del archivo divipola
	LOCAL CRESP
	CRESP=""
+-------IF !USED("divipola")
|		USE DIVIPOLA IN 0
+-------ENDIF
&&Consultamos en el archivo divipola
	SELECT DIVIPOLA
	LOCATE FOR DIVIPOLA.CODDPTO+DIVIPOLA.CODMPIO = CFUENTES
+-------IF FOUND()
|		CRESP = DIVIPOLA.NOMMPIO
+-------ENDIF
&&Retornamo el nombre de la ciudad
	RETURN CRESP
```

4.5 DISTRIBUCION.PRG && Devuelve la distribución de los fuente – articulo según Grupo Fuente y por Decada

```
&&Extraemos la informacion del archivo de fuente articulo
&&de los articulos y un conteo del numero de fuentes que lo
&&cotizan y  lo agrupamos por grupo de fuente
SELE DISTINCT CODART,COUNT(CODFTE), SUBSTR(CODFTE,14,1) AS GRUPO;
		FROM V_IPCRFAP GROUP BY GRUPO,CODART INTO CURSOR UNO ORDER BY
		GRUPO
&&Complementamos la informacion de la consulta anterior para el reporte
SELE DISTINCT UNO.CODART,UNO.GRUPO,UNO.CNT_CODFTE, V_IPCMART.NOMART,
V_IPCRFAP.DECART, COUNT(DECART) AS DECADA;
	FROM UNO, V_IPCMART, V_IPCRFAP;
	WHERE UNO.CODART = V_IPCRFAP.CODART AND UNO.CODART =
		V_IPCMART.CODART GROUP BY UNO.CODART,UNO.GRUPO,
	V_IPCRFAP.DECART;
	INTO CURSOR CUALQUIER ORDER BY UNO.CODART

&&Creamos un cursor donde vamos a ingresar uno por uno la
&&informacion de los articulo segun el grupo de fuente
	CREATE CURSOR DISTRIBUCION;
	(CODIGO C(7), NOMBRE C(25),D1 N(3), D2 N(3), D3 N(3), GB N(3),
		GM N(3), GA N(3))
&&Definimos e inicializamos las variables a utilizar para acumular
&&la informacion
	LOCAL VD1,VD2,VD3,VGB,VGM,VGA
	STORE 0 TO VD1,VD2,VD3,VGB,VGM,VGA
```

```
      SELE CUALQUIER
&&Inicialmnete insertamos el codigo y nombre del articulo
      INSERT INTO DISTRIBUCION(CODIGO, NOMBRE);
            VALUES(CUALQUIER.CODART,CUALQUIER.NOMART)

      CODIGOARTICULO=CUALQUIER.CODART
&&Recorremos el archivo resultado donde estan los articulos
&&y el numero de fuentes
+-------SCAN
|
|     +-------IF CUALQUIER.CODART=CODIGOARTICULO
&&Mientras sea el mismo articulo
&&Vamos acumulando en las variables correspondientes
&&el numero de articulos segun decada y grupo de fuente
&&Cuando es decada 1 y los distinto grupos de fuente
|     |     +-------IF CUALQUIER.DECART=1 AND CUALQUIER.DECADA<>0
|     |     |           STORE CUALQUIER.DECADA TO VD1
|     |     |     +-------IF CUALQUIER.GRUPO="1" AND
                                                !EMPTY(CUALQUIER.CNT_CODFTE)
|     |     |     |       STORE CUALQUIER.CNT_CODFTE TO VGB
|     |     |     +-------ELSE
|     |     |     |     +-------IF CUALQUIER.GRUPO="2" AND
                                                        !EMPTY(CUALQUIER.CNT_CODFTE)
|     |     |     |     |           STORE CUALQUIER.CNT_CODFTE TO VGM
|     |     |     |     +-------ELSE
|     |     |     |     |     +-------IF CUALQUIER.GRUPO="3" AND
                                                                !EMPTY(CUALQUIER.CNT_CODFT
                                                                E)
                                                                STORE
                                                                CUALQUIER.CNT_CODFTE TO
                                                                VGA
|     |     |     |     |     +-------ENDIF
|     |     |     |     +-------ENDIF
|     |     |     +-------ENDIF
|     |     +-------ELSE
&&Cuando es decada 2 y los distinto grupos de fuente
|     |     |     +-------IF CUALQUIER.DECART=2 AND CUALQUIER.DECADA<>0
|     |     |     |           STORE CUALQUIER.DECADA TO VD2
|     |     |     |     +-------IF CUALQUIER.GRUPO="1" AND
                                                        !EMPTY(CUALQUIER.CNT_CODFTE)
|     |     |     |     |           STORE CUALQUIER.CNT_CODFTE TO VGB
|     |     |     |     +-------ELSE
|     |     |     |     |     +-------IF CUALQUIER.GRUPO="2" AND
!EMPTY(CUALQUIER.CNT_CODFTE) STORE CUALQUIER.CNT_CODFTE TO VGM
|     |     |     |     |     +-------ELSE
|     |     |     |     |     |     +-------IF CUALQUIER.GRUPO="3" AND
!EMPTY(CUALQUIER.CNT_CODFTE) STORE CUALQUIER.CNT_CODFTE TO VGA
```

```
|     |     |     |     |     |     +-------ENDIF
|     |     |     |     |     +-------ENDIF
|     |     |     |     +-------ENDIF
|     |     |     +-------ELSE
&&Cuando es decada 3 y los distinto grupos de fuente
|     |     |     |     +-------IF CUALQUIER.DECART=3 AND CUALQUIER.DECADA<>0
|     |     |     |     |           STORE CUALQUIER.DECADA TO VD3
|     |     |     |     |     +-------IF CUALQUIER.GRUPO="1" AND
!EMPTY(CUALQUIER.CNT_CODFTE)
|     |     |     |     |     |         STORE CUALQUIER.CNT_CODFTE    TO VGB
|     |     |     |     |     +-------ELSE
|     |     |     |     |     |     +-------IF CUALQUIER.GRUPO="2" AND
!EMPTY(CUALQUIER.CNT_CODFTE)
|     |     |     |     |     |     |       STORE CUALQUIER.CNT_CODFTE TO VGM
|     |     |     |     |     |     +-------ELSE
|     |     |     |     |     |     |     +-------IF CUALQUIER.GRUPO="3" AND
!EMPTY(CUALQUIER.CNT_CODFTE)
|     |     |     |     |     |     |     |     STORE CUALQUIER.CNT_CODFTE TO VGA
|     |     |     |     |     |     |     +-------ENDIF
|     |     |     |     |     |     +-------ENDIF
|     |     |     |     |     +-------ENDIF
|     |     |     |     +-------ENDIF
|     |     |     +-------ENDIF
|     |     +-------ENDIF
|     |           CODIGOARTICULO=CUALQUIER.CODART
|     |
&&Al cambiar de codigo de articulo tambien
&&Vamos acumulando en las variables correspondientes
&&el numero de articulos segun decada y grupo de fuente
|     +-------ELSE
&&Segun la informacion acumulada en las diferentes variables
&&insertamos e el cursor creado inicialmente con este proposito
|     |           REPLACE DISTRIBUCION.D1 WITH VD1, DISTRIBUCION.D2 WITH  VD2,
DISTRIBUCION.D3 WITH VD3;
|     |           DISTRIBUCION.GB WITH VGB, DISTRIBUCION.GM WITH VGM,
DISTRIBUCION.GA WITH VGA
&&Reinicializamos variables
|     |           STORE 0 TO VD1,VD2,VD3,VGB,VGM,VGA
&&Insertamos inicialmente el codigo y nombre de articulo
|     |           INSERT INTO DISTRIBUCION(CODIGO, NOMBRE);
|     |                 VALUES(CUALQUIER.CODART,CUALQUIER.NOMART)
&&Cuando es decada 1 y los distinto grupos de fuente
|     |     +-------IF CUALQUIER.DECART=1 AND CUALQUIER.DECADA<>0
|     |     |           STORE CUALQUIER.DECADA TO VD1
|     |     |     +-------IF CUALQUIER.GRUPO="1" AND !EMPTY(CUALQUIER.CNT_CODFTE)
|     |     |     |       STORE CUALQUIER.CNT_CODFTE TO VGB
|     |     |     +-------ELSE
|     |     |     |     +-------IF CUALQUIER.GRUPO="2" AND !EMPTY(CUALQUIER.CNT_CODFTE)
```

```
|     |     |     |     |          STORE CUALQUIER.CNT_CODFTE TO VGM
|     |     |     |     +-------ELSE
|     |     |     |     |     +-------IF CUALQUIER.GRUPO="3" AND
!EMPTY(CUALQUIER.CNT_CODFTE)
|     |     |     |     |     |         STORE CUALQUIER.CNT_CODFTE TO VGA
|     |     |     |     |     +-------ENDIF
|     |     |     |     +-------ENDIF
|     |     |     +-------ENDIF
&&Cuando es decada 2 y los distinto grupos de fuente
|     |     +-------ELSE
|     |     |     +-------IF CUALQUIER.DECART=2 AND CUALQUIER.DECADA<>0
|     |     |     |           STORE CUALQUIER.DECADA TO VD2
|     |     |     |     +-------IF CUALQUIER.GRUPO="1" AND  !EMPTY(CUALQUIER.CNT_CODFTE)
|     |     |     |     |      STORE CUALQUIER.CNT_CODFTE TO VGB
|     |     |     |     +-------ELSE
|     |     |     |     |     +-------IF CUALQUIER.GRUPO="2" AND
!EMPTY(CUALQUIER.CNT_CODFTE)
|     |     |     |     |     |         STORE CUALQUIER.CNT_CODFTE TO VGM
|     |     |     |     |     +-------ELSE
|     |     |     |     |     |     +-------IF CUALQUIER.GRUPO="3" AND
!EMPTY(CUALQUIER.CNT_CODFTE)
|     |     |     |     |     |     |       STORE CUALQUIER.CNT_CODFTE TO VGA
|     |     |     |     |     |     +-------ENDIF
|     |     |     |     |     +-------ENDIF
|     |     |     |     +-------ENDIF
&&Cuando es decada 3 y los distinto grupos de fuente
|     |     |     +-------ELSE
|     |     |     |     +-------IF CUALQUIER.DECART=3 AND CUALQUIER.DECADA<>0
|     |     |     |     |          STORE CUALQUIER.DECADA TO VD3
|     |     |     |     |     +-------IF CUALQUIER.GRUPO="1" AND
!EMPTY(CUALQUIER.CNT_CODFTE)
|     |     |     |     |     |         STORE CUALQUIER.CNT_CODFTE TO VGB
|     |     |     |     |     +-------ELSE
|     |     |     |     |     |     +-------IF CUALQUIER.GRUPO="2" AND
!EMPTY(CUALQUIER.CNT_CODFTE)
|     |     |     |     |     |     |       STORE CUALQUIER.CNT_CODFTE TO VGM
|     |     |     |     |     |     +-------ELSE
|     |     |     |     |     |     |     +-------IF CUALQUIER.GRUPO="3" AND
!EMPTY(CUALQUIER.CNT_CODFTE)
|     |     |     |     |     |     |     |      STORE CUALQUIER.CNT_CODFTE TO VGA
|     |     |     |     |     |     |     +-------ENDIF
|     |     |     |     |     |     +-------ENDIF
|     |     |     |     |     +-------ENDIF
|     |     |     |     +-------ENDIF
|     |     |     +-------ENDIF
|     |     +-------ENDIF
|     |           CODIGOARTICULO=CUALQUIER.CODART
|     |           LOOP
```

```
|       +-------ENDIF
+-------ENDSCAN
&&Luego de que el archivo ya contiene la informacion clasificada
&&Llamamos el reporte correspondiente para desplegar la informacion
        DO FORM ('impresion1.scx') WITH REPDISTRI
```

4.6 DISTRIBUCION1.PRG&& Devuelve la distribución de los fuente – articulo según Grupo Fuente y por Periodicidad

```
&&Extractamos la informacion referente a los articulos y
&&a los diferente grupos de fuentes y agrupamos por los mismos
        SELE CODART,COUNT(CODFTE), SUBSTR(CODFTE,14,1) AS GRUPO FROM V_IPCRFAP
GROUP BY GRUPO,CODART INTO CURSOR UNO
&&Complementamos la informacion de la consulta anterior a
&&partir de los archivos maestro de articulo y fuente articulo
        SELE UNO.CODART, V_IPCMART.NOMART, UNO.CNT_CODFTE, UNO.GRUPO,
                V_IPCRFAP.PERFAP, COUNT(PERFAP) AS PERIODICIDAD;
            FROM UNO, V_IPCMART, V_IPCRFAP;
            WHERE UNO.CODART = V_IPCRFAP.CODART AND UNO.CODART =
                        V_IPCMART.CODART;
            GROUP BY V_IPCRFAP.PERFAP, UNO.GRUPO,UNO.CODART;
            INTO CURSOR CUALQUIER ORDER BY UNO.CODART
&&Creamos cursor para almacenar la informacion concerniente
&&a los articulos con el numero de cotizaciones por grupo
&&de fuente y por periodicidades
        CREATE CURSOR DISTRIBUCION1;
            (CODIGO C(7), NOMBRE C(25), P0 N(3), P1 N(3), P2 N(3), P3 N(3),
                        P4 N(3), P5 N(3), P6 N(3), P7 N(3), P8 N(3), P9 N(3), GB N(3), GM N(3), GA
                        N(3))
&&Definimos e inicializamos variables para acumular la informacion
&&de fuentes por periodicidades
        LOCAL VP0,VP1,VP2,VP3,VP4,VP5,VP6,VP7,VP8,VP9,VGB,VGM,VGA
        STORE 0 TO VP0,VP1,VP2,VP3,VP4,VP5,VP6,VP7,VP8,VP9,VGB,VGM,VGA

        SELE CUALQUIER
&&INsertamos el codigo y nombre del primer articulo
        INSERT INTO DISTRIBUCION1(CODIGO, NOMBRE);
            VALUES(CUALQUIER.CODART,CUALQUIER.NOMART)

        CODIGOARTICULO=CUALQUIER.CODART
&&REcorremos el archivo resultado de la consulta donde estan los articulos
&&con sus fuentes y periodicidades
SCAN
&&Mientras estemos en el mismo codigo de articulo empezamos a acumular
&&segun el grupo al que pertenezcan y el numero de cotizaciones
-IF CUALQUIER.CODART=CODIGOARTICULO
```

```
&&Para los articulos de grupo de fuente 1 y su periodicidades
--IF CUALQUIER.GRUPO="1" AND !EMPTY(CUALQUIER.CNT_CODFTE)
        STORE CUALQUIER.CNT_CODFTE TO VGB
---IF CUALQUIER.PERFAP=0 AND
                !EMPTY(CUALQUIER.PERIODICIDAD)
       STORE CUALQUIER.PERIODICIDAD TO VP0
---ELSE
----IF CUALQUIER.PERFAP=1 AND
        !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE CUALQUIER.PERIODICIDAD TO
        VP1
----ELSE
-----IF CUALQUIER.PERFAP=2 AND
                !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE
                CUALQUIER.PERIODICIDAD TO VP2
-----ELSE
------IF CUALQUIER.PERFAP=3 AND !EMPTY(CUALQUIER.PERIODICIDAD)
       STORE
                CUALQUIER.PERIODICIDAD TO VP3
------ELSE
-------IF CUALQUIER.PERFAP=4 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP4
-------ELSE
+-------IF CUALQUIER.PERFAP=5 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP5
+-------ELSE
        +-------IF CUALQUIER.PERFAP=6 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                        STORE CUALQUIER.PERIODICIDAD TO VP6
        +-------ELSE
                +-------IF CUALQUIER.PERFAP=7 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                                STORE CUALQUIER.PERIODICIDAD TO VP7
                +-------ELSE
                        +-------IF CUALQUIER.PERFAP=8 AND
                                        !EMPTY(CUALQUIER.PERIODICIDAD)
                                        STORE CUALQUIER.PERIODICIDAD TO VP8
                        +-------ELSE
                                +-------IF CUALQUIER.PERFAP=9 AND
                                                !EMPTY(CUALQUIER.PERIODICIDAD)
                                                STORE CUALQUIER.PERIODICIDAD TO VP9
                                +-------ENDIF
                        +-------ENDIF
                +-------ENDIF
        +-------ENDIF
+-------ENDIF
-------ENDIF
-----ENDIF
----ENDIF
```

```
---ENDIF
--ENDIF
-ELSE
&&Para los articulos de grupo de fuente 2 y su periodicidades
-IF CUALQUIER.GRUPO="2" AND
        !EMPTY(CUALQUIER.CNT_CODFTE)
        STORE CUALQUIER.CNT_CODFTE TO VGM
--IF CUALQUIER.PERFAP=0 AND
        !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE CUALQUIER.PERIODICIDAD TO
        VP0
--ELSE
---IF CUALQUIER.PERFAP=1 AND
        !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE
        CUALQUIER.PERIODICIDAD TO
        VP1
---ELSE
----IF CUALQUIER.PERFAP=2 AND !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE CUALQUIER.PERIODICIDAD TO VP2
----ELSE
-----IF CUALQUIER.PERFAP=3 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP3
-----ELSE
------IF CUALQUIER.PERFAP=4 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP4
------ELSE
-------IF CUALQUIER.PERFAP=5 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP5
-------ELSE
+-------IF CUALQUIER.PERFAP=6 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP6
+-------ELSE
                +-------IF CUALQUIER.PERFAP=7 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                                STORE CUALQUIER.PERIODICIDAD TO VP7
                +-------ELSE
                        +-------IF CUALQUIER.PERFAP=8 AND
                                        !EMPTY(CUALQUIER.PERIODICIDAD)
                                        STORE CUALQUIER.PERIODICIDAD TO VP8
                        +-------ELSE
                                +-------IF CUALQUIER.PERFAP=9 AND
                                                !EMPTY(CUALQUIER.PERIODICIDAD)
                                                STORE CUALQUIER.PERIODICIDAD TO VP9
                                +-------ENDIF
                        +-------ENDIF
                +-------ENDIF
+-------ENDIF
-------ENDIF
```

```
------ENDIF
-----ENDIF
----ENDIF
---ENDIF
--ENDIF
-ELSE
&&Para los articulos de grupo de fuente 3 y su periodicidades
-IF CUALQUIER.GRUPO="3" AND !EMPTY(CUALQUIER.CNT_CODFTE)
        STORE CUALQUIER.CNT_CODFTE TO VGA
--IF CUALQUIER.PERFAP=0 AND !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE CUALQUIER.PERIODICIDAD TO VP0
--ELSE
---IF CUALQUIER.PERFAP=1 AND !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE CUALQUIER.PERIODICIDAD TO VP1
---ELSE
----IF CUALQUIER.PERFAP=2 AND !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE CUALQUIER.PERIODICIDAD TO VP2
----ELSE
-----IF CUALQUIER.PERFAP=3 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP3
-----ELSE
------IF CUALQUIER.PERFAP=4 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP4
------ELSE
-------IF CUALQUIER.PERFAP=5 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP5
-------ELSE
+-------IF CUALQUIER.PERFAP=6 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP6
+-------ELSE
        +-------IF CUALQUIER.PERFAP=7 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                        STORE CUALQUIER.PERIODICIDAD TO VP7
        +-------ELSE
                +-------IF CUALQUIER.PERFAP=8 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                                STORE CUALQUIER.PERIODICIDAD TO VP8
                +-------ELSE
                        +-------IF CUALQUIER.PERFAP=9 AND
                                        !EMPTY(CUALQUIER.PERIODICIDAD)
                                        STORE CUALQUIER.PERIODICIDAD TO VP9
                        +-------ENDIF
                +-------ENDIF
         +-------ENDIF
        +-------ENDIF
+-------ENDIF
-------ENDIF
------ENDIF
-----ENDIF
----ENDIF
```

```
---ENDIF
--ENDIF
-ENDIF
-ENDIF
&&Cuando cambiamos de codigo de articulo
-ELSE
&&Almacenamos los valores acumulados en las variables correspondientes
&&en el archivo creado para ese proposito
        REPLACE DISTRIBUCION1.P0 WITH VP0, DISTRIBUCION1.P1 WITH VP1,
        DISTRIBUCION1.P2 WITH VP2, DISTRIBUCION1.P3 WITH VP3;
        DISTRIBUCION1.P4 WITH VP4, DISTRIBUCION1.P5 WITH VP5, DISTRIBUCION1.P6 WITH
        VP6;
        DISTRIBUCION1.P7 WITH VP7, DISTRIBUCION1.P8 WITH VP8, DISTRIBUCION1.P9 WITH
        VP9;
        DISTRIBUCION1.GB WITH VGB, DISTRIBUCION1.GM WITH VGM, DISTRIBUCION1.GA
        WITH VGA
&&Reinicializamos variables
        STORE 0 TO VP0,VP1,VP2,VP3,VP4,VP5,VP6,VP7,VP8,VP9,VGB,VGM,VGA
&&Almacenamos la informacion codigo y nombre de articulo
&&para el siguiente articulo y asi sucesivamente
        INSERT INTO DISTRIBUCION1(CODIGO, NOMBRE);
        VALUES(CUALQUIER.CODART,CUALQUIER.NOMART)
&&Para los articulos de grupo de fuente 1 y su periodicidades
-IF CUALQUIER.GRUPO="1" AND !EMPTY(CUALQUIER.CNT_CODFTE)
        STORE CUALQUIER.CNT_CODFTE TO VGB
--IF CUALQUIER.PERFAP=0 AND !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE CUALQUIER.PERIODICIDAD TO VP0
--ELSE
---IF CUALQUIER.PERFAP=1 AND !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE CUALQUIER.PERIODICIDAD TO VP1
---ELSE
----IF CUALQUIER.PERFAP=2 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP2
----ELSE
-----IF CUALQUIER.PERFAP=3 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP3
-----ELSE
------IF CUALQUIER.PERFAP=4 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP4
------ELSE
-------IF CUALQUIER.PERFAP=5 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP5
-------ELSE
+-------IF CUALQUIER.PERFAP=6 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP6
+-------ELSE
        +-------IF CUALQUIER.PERFAP=7 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                        STORE CUALQUIER.PERIODICIDAD TO VP7
```

```
        +-------ELSE
                +-------IF CUALQUIER.PERFAP=8 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                        STORE CUALQUIER.PERIODICIDAD TO VP8
                +-------ELSE
                        +-------IF CUALQUIER.PERFAP=9 AND
                                !EMPTY(CUALQUIER.PERIODICIDAD)
                                STORE CUALQUIER.PERIODICIDAD TO VP9
                        +-------ENDIF
                +-------ENDIF
        +-------ENDIF
+-------ENDIF
-------ENDIF
------ENDIF
-----ENDIF
----ENDIF
---ENDIF
--ENDIF
-ELSE
&&Para los articulos de grupo de fuente 2 y su periodicidades
-IF CUALQUIER.GRUPO="2" AND !EMPTY(CUALQUIER.CNT_CODFTE)
        STORE CUALQUIER.CNT_CODFTE TO VGM
--IF CUALQUIER.PERFAP=0 AND !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE CUALQUIER.PERIODICIDAD TO VP0
--ELSE
---IF CUALQUIER.PERFAP=1 AND !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE CUALQUIER.PERIODICIDAD TO VP1
---ELSE
----IF CUALQUIER.PERFAP=2 AND !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE CUALQUIER.PERIODICIDAD TO VP2
----ELSE
-----IF CUALQUIER.PERFAP=3 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP3
-----ELSE
------IF CUALQUIER.PERFAP=4 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP4
------ELSE
-------IF CUALQUIER.PERFAP=5 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP5
-------ELSE
+-------IF CUALQUIER.PERFAP=6 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP6
+-------ELSE
        +-------IF CUALQUIER.PERFAP=7 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                        STORE CUALQUIER.PERIODICIDAD TO VP7
        +-------ELSE
                +-------IF CUALQUIER.PERFAP=8 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                        STORE CUALQUIER.PERIODICIDAD TO VP8
                +-------ELSE
```

```
                        +-------IF CUALQUIER.PERFAP=9 AND
                                        !EMPTY(CUALQUIER.PERIODICIDAD)
                                        STORE CUALQUIER.PERIODICIDAD TO VP9
                        +-------ENDIF
                +-------ENDIF
        +-------ENDIF
+-------ENDIF
-------ENDIF
------ENDIF
-----ENDIF
----ENDIF
---ENDIF
--ENDIF
-ELSE
&&Para los articulos de grupo de fuente 3 y su periodicidades
-IF CUALQUIER.GRUPO="3" AND !EMPTY(CUALQUIER.CNT_CODFTE)
        STORE CUALQUIER.CNT_CODFTE TO VGA
--IF CUALQUIER.PERFAP=0 AND
        !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE CUALQUIER.PERIODICIDAD TO VP0
--ELSE
---IF CUALQUIER.PERFAP=1 AND !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE CUALQUIER.PERIODICIDAD TO VP1
---ELSE
---IF CUALQUIER.PERFAP=2 AND !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE CUALQUIER.PERIODICIDAD TO VP2
---ELSE
----IF CUALQUIER.PERFAP=3 AND !EMPTY(CUALQUIER.PERIODICIDAD)
        STORE CUALQUIER.PERIODICIDAD TO VP3
----ELSE
-----IF CUALQUIER.PERFAP=4 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP4
-----ELSE
-----IF CUALQUIER.PERFAP=5 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP5
-----ELSE
------IF CUALQUIER.PERFAP=6 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP6
------ELSE
-------IF CUALQUIER.PERFAP=7 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP7
-------ELSE
+-------IF CUALQUIER.PERFAP=8 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP8
+-------ELSE
        +-------IF CUALQUIER.PERFAP=9 AND !EMPTY(CUALQUIER.PERIODICIDAD)
                STORE CUALQUIER.PERIODICIDAD TO VP9
        +-------ENDIF
```

```
+-------ENDIF
-------ENDIF
------ENDIF
-----ENDIF
----ENDIF
---ENDIF
--ENDIF
--ENDIF
        CODIGOARTICULO=CUALQUIER.CODART
        LOOP
-ENDIF
ENDSCAN
&&Luego de que se encuentra cargado el archivo con la informacion
&&procesada llamamos el reporte correspondiente para desplegar
&&la informacion
    DO FORM ('impresion1.scx') WITH REPDISTRI1
```

4.7 ENVEZ4.PRG && Procesa la información capturada para generar el reporte de Precios Promedio por Clase

```
&&Precios promedio
    LOCAL TABLA1, LOCDIR
    SET SAFETY OFF
    LOCDIR=SET('DEFAULT')+SYS(2003)

    TABLA1='&locdir'+'\'+'ppclas'
      SELECT 1 AS NUM, SUBSTR(V_IPCTMVT.CODFTE,15,2) AS CLASE, V_IPCTMVT.*,
      V_IPCMART.CANBART;
      WHERE (V_IPCTMVT.MARCA = .F.) AND V_IPCTMVT.PACTMVT <> 0  AND
      V_IPCTMVT.CODART = V_IPCMART.CODART AND V_IPCTMVT.DECMVT IN (A,B,C) AND
      (ALLTRIM(V_IPCTMVT.CODART) IN (SELECT ALLTRIM(SELEC14.CODART) AS LLAVE
      FROM SELEC14));
      FROM V_IPCTMVT, V_IPCMART;
          ORDER BY 4,2 ;
      INTO TABLE +'&tabla1'

      SELE PPCLAS REPLACE ALL PPCLAS.PACTMVT WITH IIF(CANRMVT > 0 AND PACTMVT
      > 0,(PACTMVT / CANRMVT) * CANBART,(PACTMVT * CANBART) / CANBART)
      REPLACE ALL PPCLAS.PANART WITH IIF(CANRMVT > 0  AND PACTMVT > 0,(PANART /
      CANRMVT) * CANBART,(PANART * CANBART) / CANBART)

      CREATE TABLE CLASES FREE (CODART C(7),CLASE C(2),PPCL N(12,2) NULL ,PACL
      N(12,2) NULL)

    SELE PPCLAS
    GO TOP
```

```
+-------DO WHILE !EOF()
|           SELE PPCLAS
|           M.PACTMVT=1
|           M.PANART=1
|           M.CUANTACT=0
|           M.CUANTANT=0
|           M.CODART=PPCLAS.CODART
|           M.CLASE=SUBSTR(CODFTE,15,2)
|     +-------DO WHILE SUBSTR(CODFTE,15,2)=m.CLASE AND m.CODART=PPCLAS.CODART
                                AND !EOF()
|     |     +-------IF PPCLAS.PACTMVT <> 0
|     |     |           M.PACTMVT=m.PACTMVT*PPCLAS.PACTMVT
|     |     |           M.CUANTACT=m.CUANTACT+1
|     |     +-------ENDIF
|     |
|     |     +-------IF PPCLAS.PANART <> 0
|     |     |           M.PANART=m.PANART*PPCLAS.PANART
|     |     |           M.CUANTANT=m.CUANTANT+1
|     |     +-------ENDIF
|     |
|     |           SKIP
|     +-------ENDDO
|
|     +-------IF m.CUANTACT <> 0
|     |           M.PPCL= (m.PACTMVT**(1/m.CUANTACT))
|     +-------ENDIF
|
|     +-------IF m.CUANTANT <> 0
|     |           M.PACL= (m.PANART**(1/m.CUANTANT))
|     +-------ENDIF
|
|     +-------IF m.CUANTANT <> 0 OR m.CUANTACT <> 0
|     |           SELE CLASES
|     |           APPEND BLANK
|     |           GATHER MEMVAR
|     +-------ENDIF
|
+-------ENDDO

      SELE CLASES
      INDEX ON CODART+CLASE TO PKCLASE

      SELE PPCLAS
      INDEX ON CODART+CLASE+CODFTE TAG CODI
      SET RELATION TO (STR(IMPCTV,7,0))+CODFTE+OBSMVT INTO V_IPCROBS ADDITIVE
      SET RELATION TO CODFTE INTO V_IPCMFTE ADDITIVE
      SET RELATION TO CODART INTO V_IPCMART ADDITIVE
```

```
SET RELATION TO CODART+CLASE INTO CLASES  ADDITIVE
```

4.8 FCOM.PRG && Devuelve la información de las fuentes complementarias para cada fuente original al momento de salir prediligenciada

```
LPARAMETERS CFUENTES,NCOMPLE,CCAMPO
&&Al momento de prediligenciamiento recibimos parametros y
&&devolvemos la informacion correspondiente a la o las
&&fuentes complementarias pertenecientes a las fuentes
&&originales que salen prediligenciadas
NOFTE = CFUENTES
&&Definimos y cargamos variables para almacenar el nombre de los archivos
LOCAL CRESP,CALIAS
CRESP=""
CALIAS=ALIAS()
&&Preparamos el archivo de fuentes complementarias para usarla
&&seleccioonamos el archivo de fuentes complementarias
&&buscamos la informacion de las fuentes
&&complementarias correspondientes a la original
SELECT V_PREFCM
LOCATE FOR STR(V_PREFCM.CODFCOM)+CODFTE = STR(NCOMPLE)+CFUENTES
+-------IF FOUND()
|       +-------DO CASE
&&Segun los parametros que hayamos recibido asi mismo devolvemos
&&los datos
|       +-------CASE CCAMPO = "N"
|       |               CRESP = V_PREFCM.NOMFTE
|       +-------CASE CCAMPO = "D"
|       |               CRESP = V_PREFCM.DIREFTE
|       +-------CASE CCAMPO = "T"
|       |               CRESP = V_PREFCM.TELFTE
|       +-------ENDCASE
+-------ENDIF
SELEC (CALIAS)
RETURN CRESP
```

4.9 PORCENTAJES.PRG && Devuelve los porcentajes de captura para la fuente actual

```
LPARAMETERS CTVACTUAL
&&Recibimo el parametro correspondiente al consecutivo solicitado para mostrar la informacion
&&Capturamos el nombre del archivo que esta siendo utilizado para al terminar de procesar
&&con otros archivos retomarlo
CALIAS = ALIAS()
```

```
      CTVCAPTU = CTVACTUAL
&&Verificamos que tipo de informacion se solicito, si con mensaje o no, es decir, informacion
&&sobre el periodo actual incluyendo numero de fuentes, porcentaje de las mismas capturado
&&de los arcticulos, etc
+-------IF CONMENSAJE = .T.
                =MESSAGEBOX(" TOTALES POR MES:"+CHR(13)+CHR(13)+"
                Fuentes: "+ALLTRIM(STR(NFTES))+CHR(13)+;
|         ARTICULOS: "+ALLTRIM(STR(NART))+CHR(13)+;
|         ARTICULOS CAPTURADOS: "+ALLTRIM(STR(NARTCAP))+CHR(13)+;
|         % ARTICULOS CAPTURADOS: "+ALLTRIM(STR(PARTCAP,5,3))+"
                "+CHR(13)+CHR(13)+;
|         "TOTALES POR DECADA:"+CHR(13)+CHR(13)+"
                Fuentes: "+ALLTRIM(STR(NFTESDEC))+CHR(13)+;
|         "ARTICULOS: "+ALLTRIM(STR(NARTDEC))+CHR(13)+;
|                "ARTICULOS CAPTURADOS: "+ALLTRIM(STR(NARTCAPDEC))+CHR(13)+;
                "% ARTICULOS CAPTURADOS:
                "+ALLTRIM(STR(PARTCAPDEC,5,3))+CHR(13),0,'Información de Captura')
+-------ELSE
&&e lo contrario es solo para el formulario actual
|           PUBLIC NARTFTE, NARTCAPFTEDEC
|           STORE 0 TO NARTFTE, NARTCAPFTEDEC
&&Reconsultamos la informacion de la vista segun el consecutivo actual(fuentes ay articulos)
|           REQUERY('v_mvtporcent')
|           NARTFTE = RECCOUNT()
|
+-------ENDIF
```

4.10 REPORTE1.PRG && Procesa la información existente para generar el reporte de Canasta para Cada Ciudad

```
&&Canasta Nacional (Fuentes-Articulos-Especificaciones)
      LOCAL LOCDIR, TABLA1, TABLA2, TABLA3
      SET SAFETY OFF
      LOCDIR=SET('DEFAULT')+SYS(2003)

      TABLA1='&locdir'+'\'+'rep'
      SELECT SUBSTR(V_IPCRFAP.CODFTE,17,4), V_IPCRFAP.CODFTE,
                V_IPCRFAP.CODART, V_IPCRFAP.DECART, V_IPCRFAP.PERFAP,;
         V_IPCMFTE.NOMBFTE, V_IPCMART.NOMART, V_IPCMFTE.ZONA;
                FROM V_IPCRFAP, V_IPCMFTE, V_IPCMART;
                WHERE (V_IPCRFAP.MARCA = .F.) AND (V_IPCRFAP.CODFTE =
                V_IPCMFTE.CODFTE AND V_IPCRFAP.CODART = V_IPCMART.CODART AND ;
                ALLTRIM(V_IPCRFAP.CODFTE) IN (SELECT ALLTRIM(SELEC23.CODFTE) AS
                LLAVE FROM SELEC23));
         ORDER BY 5,1,3;
         INTO TABLE+'&tabla1'
```

```
&&Para listar solo la decada y mes seleccionado
      TABLA1='&locdir'+'\'+'repo1'
      SELECT * FROM REP ;
                WHERE REP.DECART IN (A,B,C) AND (REP.PERFAP IN (SELECT
                V_IPCRPART.CODPART AS LLAVE FROM V_IPCRPART WHERE
                BETWEEN(MES,MES1,MES2)));
         ORDER BY 5,1,3;
         INTO TABLE+'&tabla1'


      TABLA2='&locdir'+'\'+'repo2'
&&Las que si tienen especificaciones
            SELE
         REPO1.CODFTE,REPO1.NOMBFTE,REPO1.CODART,REPO1.NOMART,REPO1.DECART,
         REPO1.PERFAP,REPO1.ZONA,;
    V_IPCRESP.CODESP, V_IPCRESP.NOMESP, V_IPCRESP.NOMBDET;
    FROM REPO1, V_IPCRESP;
         WHERE ((REPO1.CODFTE + REPO1.CODART) = (V_IPCRESP.CODFTE +
         V_IPCRESP.CODART));
    INTO TABLE+'&tabla2'

    TABLA3='&locdir'+'\'+'repo3'
&&Las que no tienen especificaciones
            SELE REPO1.CODFTE, REPO1.NOMBFTE, REPO1.CODART, REPO1.NOMART,
         REPO1.DECART, REPO1.PERFAP, REPO1.ZONA;
    FROM REPO1;
    WHERE ((REPO1.CODFTE + REPO1.CODART);
         NOT IN (SELECT DISTINCT (V_IPCRESP.CODFTE + V_IPCRESP.CODART) FROM
         V_IPCRESP));
         INTO TABLE+'&tabla3'

             CREATE TABLE REPO FREE (CODFTE C(20), NOMBFTE C(35) NULL,CODART
         C(7),NOMART C(30) NULL,DECART N(1) NULL,;
         PERFAP N(1) NULL,CODESP C(2) NULL,NOMESP C(40) NULL,NOMBDET C(30)
         NULL,ZONA N(4) NULL)

      SELE REPO
         INDEX ON CODFTE + CODART +
         CODESP+ALLTRIM(STR(DECART))+ALLTRIM(STR(PERFAP))+ALLTRIM(STR(ZONA)) TAG
         CODI
    APPEND FROM+'&tabla2'
    APPEND FROM+'&tabla3'
```

4.11 REPORTE2.PRG && Procesa la información existente para generar el reporte de Canasta para Cada Ciudad

```
&&Cotizaciones por articulo
    LOCAL TABLA1
    SET SAFETY OFF
    LOCDIR=SET('DEFAULT')+SYS(2003)

    TABLA1='&locdir'+'\'+'REPOa'
&&Para listar solo la decada y mes seleccionado
    SELECT SUBSTR(V_IPCRFAP.CODFTE,17,4), V_IPCRFAP.CODFTE,
            V_IPCRFAP.CODART, V_IPCRFAP.DECART, V_IPCRFAP.PERFAP,;
      V_IPCMFTE.NOMBFTE, V_IPCMART.NOMART, V_IPCMFTE.TELFTE,
            V_IPCMFTE.DIREFTE, V_IPCMFTE.ZONA;
      FROM V_IPCRFAP, V_IPCMFTE, V_IPCMART;
      WHERE (V_IPCRFAP.MARCA = .F.) AND (V_IPCRFAP.CODFTE =
            V_IPCMFTE.CODFTE AND V_IPCRFAP.CODART = V_IPCMART.CODART AND
            V_IPCRFAP.DECART IN (A,B,C) AND ALLTRIM(V_IPCRFAP.CODART) IN (SELECT
            ALLTRIM(SELEC14.CODART) AS LLAVE FROM SELEC14));
      ORDER BY 1,3;
      INTO TABLE+'&tabla1'

    TABLA1='&locdir'+'\'+'REPOart'
    SELECT DISTINCT * FROM REPOA ;
      WHERE (REPOA.PERFAP IN (SELECT V_IPCRPART.CODPART AS LLAVE FROM
            V_IPCRPART WHERE BETWEEN(MES,MES1,MES2))) ;
      ORDER BY 1,3 ;
        INTO TABLE +'&tabla1'

    SELE REPOART
      INDEX ON
      CODART+CODFTE+ALLTRIM(STR(DECART))+ALLTRIM(STR(PERFAP))+ALLTRIM(STR(Z
      ONA)) TAG CODI
  REPORT FORM REPORTE2 noconsole preview
```

4.12 REPORTE3.PRG && Procesa la información existente para generar el Listado de Fuentes para Cada Ciudad

```
&&Directorio de Fuentes
    SET SAFETY OFF
    LOCAL TABLA1, LOCDIR, PER, PERF, DECA

    LOCDIR=SET('DEFAULT')+SYS(2003)

&&Para listar solo la decada y mes seleccionado
    TABLA1='&locdir'+'\'+'REPOf'
```

```
SELECT SUBSTR(V_IPCRFAP.CODFTE,17,4), V_IPCRFAP.CODFTE,
        V_IPCMFTE.NOMBFTE, V_IPCRFAP.DECART, V_IPCRFAP.PERFAP,;
        V_IPCMFTE.TELFTE, V_IPCMFTE.DIREFTE, V_IPCMFTE.ZONA,
        V_IPCMFTE.ESTRFTE;
  FROM V_IPCRFAP, V_IPCMFTE, V_IPCMART;
  WHERE (V_IPCRFAP.MARCA = .F.) AND (V_IPCRFAP.CODFTE =
        V_IPCMFTE.CODFTE AND V_IPCRFAP.CODART = V_IPCMART.CODART AND ;
  V_IPCRFAP.DECART IN (A,B,C) AND ALLTRIM(V_IPCMFTE.CODFTE) IN (SELECT
        ALLTRIM(SELEC23.CODFTE) AS LLAVE FROM SELEC23));
  ORDER BY 1,3 ;
  INTO TABLE+'&tabla1'

TABLA1='&locdir'+'\'+'repoft'
SELECT DISTINCT * FROM REPOF ;
  WHERE (REPOF.PERFAP IN (SELECT V_IPCRPART.CODPART AS LLAVE FROM
        V_IPCRPART WHERE BETWEEN(MES,MES1,MES2))) ;
  ORDER BY 1,3 ;
  INTO TABLE +'&tabla1'

INDEX ON CODFTE+ALLTRIM(STR(PERFAP)) TO REPOFT

CREATE TABLE REPOFUEN FREE (CODFTE C(20),NOMBFTE C(35),DECART C(7),
        PERFAP C(20),;
  TELFTE C(20),DIREFTE C(40),ESTRFTE C(1),ZONA N(4))

SELE REPOFUEN
INDEX ON CODFTE TO REPOFUEN

SELE REPOFT

+-------SCAN
|       SELE REPOFT
|       SCATTER MEMVAR
|       M.PERFAP=ALLTRIM(STR(m.PERFAP))
|       M.DECART=ALLTRIM(STR(m.DECART))
|
|       SELE REPOFUEN
|   +-------IF SEEK(m.CODFTE)
|   |   +-------IF RIGHT(ALLTRIM(PERF),1) != m.PERFAP
|   |   |       PERF=ALLTRIM(REPOFUEN.PERFAP)+','+m.PERFAP
|   |   |       REPLACE REPOFUEN.PERFAP WITH PERF
|   |   +-------ENDIF
|   |
|   |   +-------IF RIGHT(ALLTRIM(DECA),1)=m.DECART OR
                            LEFT(ALLTRIM(DECA),1)=m.DECART
|   |   |
|   |   +-------ELSE
|   |   |       DECA=ALLTRIM(REPOFUEN.DECART)+','+m.DECART
```

```
|       |       |           REPLACE REPOFUEN.DECART WITH DECA
|       |       +-------ENDIF
|       |
|       +-------ELSE
|       |             PERF=m.PERFAP
|       |             DECA=m.DECART
|       |             APPEND BLANK
|       |             GATHER MEMVAR
|       +-------ENDIF
+-------ENDSCAN
```

4.13 REPORTE5.PRG && Procesa la información existente para generar el reporte de Canasta para Cada Ciudad

```
&&Precios promedio
      LOCAL TABLA1, LOCDIR
      SET SAFETY OFF

      LOCDIR=SET('DEFAULT')+SYS(2003)

      TABLA1='&locdir'+'\'+'mvtt'
      SELECT
         V_IPCTMVT.*,IIF(V_IPCTMVT.PANART>0,ROUND(((V_IPCTMVT.PACTMVT/V_IPCTMVT.P
         ANART)*100)-100,2),000.00) AS VARIA ;
     FROM V_IPCTMVT ;
         WHERE (V_IPCTMVT.MARCA = .F.) AND (ALLTRIM(V_IPCTMVT.CODART) IN (SELECT
         ALLTRIM(SELEC14.CODART) AS LLAVE FROM SELEC14));
     ORDER BY 4,2 ;
     INTO TABLE +'&tabla1'

&&Para listar solo la decada seleccionada
      TABLA1='&locdir'+'\'+'mvt'
      SELECT * ;
            FROM MVTT ;
            WHERE VARIA<>0 AND MVTT.DECMVT IN (A,B,C) ;
            ORDER BY 4,2;
            INTO TABLE+'&tabla1'

      SELE MVT
      INDEX ON CODART+CODFTE+ALLTRIM(STR(DECMVT)) TAG CODI
      SET RELATION TO (STR(IMPCTV,7,0))+CODFTE+OBSMVT INTO V_IPCROBS ADDITIVE
      SET RELATION TO CODFTE INTO V_IPCMFTE ADDITIVE
      SET RELATION TO CODART INTO V_IPCMART ADDITIVE
```

5 FORMULARIOS

5.1 ACTUALIZAR FUENTES EXISTENTES(ACTFTES.SCX)

5.1.1 Vistas

Vfuente, v_registro

```
+-------PROCEDURE ACTIVATE
&& Al entrar al formulario actualizamos el archivo de registro para saber
&& que este usuario abrio este formulario

&& Actualizamos la informacion para tener lo mas reciente de los usuarios
&& que hayan empezado a hacer uso de la informacion por medio del aplicativo
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&& Buscamos la coincidencia del usuario que abrio esta pantalla para saber
&& si ya esta relacionado en el archivo y no volver a registrarlo
                        LOCATE FOR (ALLTRIM(V_REGISTRO.QUIEN) + NPANTALLA =
                        ALLTRIM(PQUIEN) + WONTOP()) AND V_REGISTRO.ACT_INA = .T.
&& si no es encontrado lo registramos y actualizamos el archivo
|     +-------IF !FOUND()
|     |           INSERT INTO V_REGISTRO(QUIEN, NPANTALLA, FECHAR,
                              ACT_INA);
|     |           VALUES(PQUIEN,WONTOP( ), DATE(), .T.)
|     |           TABLEUPDATE(.T.)
|     +-------ENDIF
|
&& Seleccionamos la vista a utilizar
|           SELECT VFUENTE
|
|
+-------ENDPROC

+-------PROCEDURE SHOW
|           LPARAMETERS NSTYLE
|
+-------ENDPROC

+-------PROCEDURE INIT
|           PARAMETERS NCONSECUTIVO,CNOFTE
|           PUBLIC SALIR
&& Recepcion de Parametros para saber de que fuente hay que traer la informacion && a mostrar
&& Iniciar variable
|
|SALIR=.T.
```

```
|
&&Asignacion del contenido de variables a los objetos del formulario
|       +-------WITH THISFORM
|       |             .NNOFORMULARIO=NCONSECUTIVO
|       |             .TNOFORMULARIO.VALUE=.NNOFORMULARIO
|       |             .CCODFTE=CNOFTE
|       |             .CODFTE1.VALUE=CNOFTE
|       +-------ENDWITH
|
&&Dejar que ejecute el codigo original de la clase
|             DODEFAULT()
|
&&Cargar variable con la informacion de los objetos del formulario
&&Actualizar la informacion de la vista segun los parametros recibidos
|             NOFORMULARIO=THIS.NNOFORMULARIO
|             NOFTE=THIS.CCODFTE
|             SELECT VFUENTE
|             REQUERY()
|
+-------ENDPROC


+-------PROCEDURE DESTROY
&&Al salir del formulario actualizamos el archivo de registro para saber
&&que este usuario ya salio de este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|             REQUERY('v_registro')
|             SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para indicar
&&que la cerro en este momento
                          LOCATE FOR
                          ALLTRIM(V_REGISTRO.QUIEN)+ALLTRIM(V_REGISTRO.NPANTALLA) =
                          ALLTRIM(PQUIEN)+WONTOP( )
|       +-------IF FOUND()
&&Actualizamos la informacion y tambien el archivo
|       |             REPLACE V_REGISTRO.ACT_INA WITH .F.
|       |             TABLEUPDATE(.T.)
|       +-------ENDIF
+-------ENDPROC
```

5.1.2 Botón Aceptar

```
+-------PROCEDURE CLICK
&&validamos que hayan ingresado informacion para las diferentes variables
&&y asi poder almacenarla
|       +-------IF EMPTY(ALLTRIM(THISFORM.DIREFTE1.VALUE)) OR
EMPTY(ALLTRIM(THISFORM.NOMFTE1.VALUE)) OR EMPTY((THISFORM.ESTRFTE1.VALUE))
```

```
|       |           MESSAGEBOX('Debe diligenciar todos los datos de la fuente',0,'Precaución')
|       |           THISFORM.NOMFTE1.SETFOCUS
|       +-------ELSE
&&En caso afirmativo actualizamos la tabla con la informacion ingresada y salimos del formulario
|       |           SELECT VFUENTE
|       |           TABLEUPDATE(.T.)
|       |           THISFORM.RELEASE
|       +-------ENDIF
|
|
+-------ENDPROC
```

5.1.3 Botón Cancelar

```
+-------PROCEDURE CLICK
|           TABLEREVERT(.T.)
|           THISFORM.RELEASE
&&Reversamos la actualizacion de la tabla en caso que haya ingresado algo de informacion
&&y salimos del formulario
+-------ENDPROC
```

5.2 FORMULARIO GENERAR HISTORICOS (genhist.scx)

```
+-------PROCEDURE ACTIVATE
&& Al entrar al formulario actualizamos el archivo de registro para saber
&& que este usuario abrio este formulario
&& Actualizamos la informacion para tener lo mas reciente de los usuarios
&& que hayan empezado a hacer uso de la informacion por medio del aplicativo
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&& Buscamos la coincidencia del usuario que abrio esta pantalla para saber
&& si ya esta relacionado en el archivo y no voilver a registrarlo
|           LOCATE FOR (ALLTRIM(V_REGISTRO.QUIEN) + NPANTALLA = ALLTRIM(PQUIEN) +
WONTOP()) AND V_REGISTRO.ACT_INA = .T.
&&si no es encontrado lo registramos y actualizamos el archivo
|       +-------IF !FOUND()
|       |           INSERT INTO V_REGISTRO(QUIEN, NPANTALLA, FECHAR, ACT_INA);
|       |                 VALUES(PQUIEN,WONTOP( ), DATE(), .T.)
|       |           TABLEUPDATE(.T.)
|       +-------ENDIF
|
+-------ENDPROC

+-------PROCEDURE INIT
&&Declaramos e Inicializamos variables
|           PUBLIC UNIDAD, ANOHIST, MESHIST
```

```
|           UNIDAD=''
|           ANOHIST = 0
|           MESHIST = 0
&&Configuramos el ambiente de trabajo para el manejo de fechas
|           SET CENTURY ON
|           SET DATE DMY
|
|           THISFORM.SALIR.SETFOCUS
|
+-------ENDPROC

+-------PROCEDURE DESTROY
&&Al salir del formulario actualizamos el archivo de registro para saber
&&que este usuario ya salio de este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para indicar
&&que la cerro en este momento
|           LOCATE FOR ALLTRIM(V_REGISTRO.QUIEN)+ALLTRIM(V_REGISTRO.NPANTALLA) =
ALLTRIM(PQUIEN)+WONTOP( )
|     +-------IF FOUND()
|     |           REPLACE V_REGISTRO.ACT_INA WITH .F.
&&Actualizamos la informacion y tambien el archivo
|     |           TABLEUPDATE(.T.)
|     +-------ENDIF
+-------ENDPROC
```

5.2.1 Botón Buscar

```
+-------PROCEDURE CLICK
&&Validamos que se haya seleccionado informacion de año y mes
|     +-------IF EMPTY(THISFORM.COMBO1.VALUE) OR ISNULL(THISFORM.COMBO1.VALUE)
                                OR EMPTY(THISFORM.COMBO2.VALUE) OR
                                ISNULL(THISFORM.COMBO2.VALUE)
|     |           MENSAJE=MESSAGEBOX(" Debe seleccionar el año y el mes a
                                los cuales corresponde la informacion del Historico
                                !!",0+64+0,"INFORMACION")
|     |           THISFORM.COMBO1.SETFOCUS
|     +-------ELSE
&&En caso de que hayan seleccionado informacion correctamente
&&Definimos variables a utilizar para almacenar las rutas y carpetas
&&Eliminamos archivos innecesarios
|     |           LOCAL GCFILE,DIR1
|     |           DIR1=SYS(5)
|     |           LOGICA=DIRECTORY(DIR1+"\"+"IpcHist")
```

```
|       |           VFP=DIR1+"\"+"IpcHist"
&&Validamos la existencia de la carpeta predeterminada
|       |       +-------IF LOGICA
&&En caso de que exista eliminamos archivos innecesarios
|       |       |           DELETE FILE DIR1+"\"+"IpcHist\"+"*.DBF"
|       |       +-------ELSE
&&En caso de que no exista la creamos automaticamente
|       |       |           MKDIR DIR1+"\"+"IpcHist"
|       |       +-------ENDIF
|       |
|       |           GCFILE=GETDIR(DIR1+"\"+"IpcHist")
|       |
|       |           UNIDAD=GCFILE
|       |       +-------IF NOT EMPTY(GCFILE)
&&Luego de tener la ruta a donde se van a almacenar los historicos
&&Habilitamos botones necesarios para que se puede iniciar
&&el proceso de generacion de historicos
|       |       |           THISFORM.NOMBREARCHIVO.VALUE = GCFILE
|       |       |           THISFORM.INICIAR.ENABLED=.T.
|       |       +-------ENDIF
|       +-------ENDIF
+-------ENDPROC
```

5.2.2 Botón Salir

```
+-------PROCEDURE CLICK
&&Salir de esta pantalla
|           THISFORM.RELEASE
|
+-------ENDPROC
```

5.2.3 Botón Iniciar

```
+-------PROCEDURE CLICK
|           #DEFINE SW_SHOWMINIMIZED 2
&&Definimos el estado de pantalla para cuando ejecutemos el winzip
&&se ejecute minimizado;  en este caso vamos a trabajar con la API de Windows
|           LOCAL RUTAZIP
&&Cargamos variables con el año y mes seleccionados
|           ANOHIST = THISFORM.COMBO1.VALUE
|           MESHIST = THISFORM.COMBO2.VALUE
|
&&Refrescamos la informacion de la vista parametrizadas de historico
&&la cual nos trae solamente lo concerniente al año y mes seleccionados
|           REQUERY('v_phistoria')
&&Comprobamos que la consulta haya obtenido informacion
|       +-------IF _TALLY <> 0
```

```
&&En caso afirmativo, ubicamos el winzip dependiendo de la version de Windows
&&y cargamos la ruta
|     |     +-------IF FILE('c:\archivos de programa\winzip\winzip32.exe')
|     |     |           RUTAZIP='c:\archivos de
                                        programa\winzip\winzip32.exe'
|     |     +-------ELSE
|     |     |           RUTAZIP='c:\program files\winzip\winzip32.exe'
|     |     +-------ENDIF
&&Cargamos variables con la ruta de la carpeta a utilizar y los nombres de los alias
|     |           RUTA=SYS(5)+"\IpcHist\"
|     |           TABLA1="historia"
|     |           TABLA2="esptemp"
&&Configuramos el ambiente de trabajo para el manejo de creacion y reemplazo
&&de archivos
|     |           SET SAFETY OFF
|     |           SET TALK OFF
&&Definimos el nombre del archivo a generar que consta del año y mes a descargar
&&de historicos
|     |           NOMBRECOPIA
                        =ALLTRIM(DEPARTAMENTO)+PADL(ALLTRIM(STR(MESHIST)),2,'0')+
                        ALLTRIM(STR(ANOHIST))
&&Ejecutamos barra de progreso
|     |           DO FORM FPROGRESO NAME PROGRESO LINKED
                        PROGRESO.LPROGRESO.CAPTION='Generando Copia de Historicos
                        en: '+RUTA+" "+" Espere por favor..."
|     |           PROGRESO.BARRAPROGRESO1.MAXVALOR=3000
|     |           PROGRESO.BARRAPROGRESO1.MINVALOR=1
&&Avanzamos barra de progreso
|     |
PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROGRESO+
1000
|     |           PROGRESO.REFRESH
|     |           PROGRESS()
&&Generamos una copia de la vista obtenida
|     |           SELE V_PHISTORIA
|     |           COPY TO RUTA+TABLA1+".dbf"
&&Avanzamos la barra de progreso
|     |
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
RESO+1000
|     |           PROGRESO.REFRESH
|     |           PROGRESS()
&&Refresdcamos la informacion de las especificaciones correspondientes al periodo
&&seleccionado (Año yMes)
|     |           REQUERY('v_pesptemp')
&&Generamos una copia de la vista obtenida
|     |           SELE V_PESPTEMP
|     |           COPY TO RUTA+TABLA2+".dbf"
```

```
&&Terminamos el recorrido de la barra de progreso y la cerramos
|     |           PROGRESO.BARRAPROGRESO1.LLENE
|     |           PROGRESO.RELEASE
&&Verificamos la unidad adonde se va generar el archivo
|     |     +-------IF UNIDAD=UPPER('A:\') OR UNIDAD=UPPER('B:\')
|     |     |           MENSAJE= MESSAGEBOX("Ya insertó el disquete en
                                    la unidad "+UNIDAD +" ?",4+64+256," Copia para
                                    Reimpresion ")
|     |     |     +-------IF MENSAJE=6 AND (DISKSPACE(UNIDAD)= -1)
|     |     |     |           MESSAGEBOX("Ha ocurrido un error al
                                    intentar leer la unidad de disco.  "+UNIDAD +" ?",0+64+0,"
                                    Precaución IPC-98 ")
|     |     |     |           MENSAJE=7
|     |     |     +-------ENDIF
|     |     +-------ELSE
|     |     |           MENSAJE=6
|     |     +-------ENDIF
|     |     +-------DO CASE
|     |     +-------CASE MENSAJE = 6
&& Cargamos variable con la ruta actual para luego volver a esta
|     |     |           ACTUAL=FULLPATH(SYS(2003))
&& Nos direccionamos a la nueva ruta
|     |     |           SET DEFAULT TO &RUTA
|     |     |           WAIT WIND 'Generando Copia de Historicos en:
                                '+RUTA+" "+" Espere por favor..." NOWAIT
&& Cargamos variable con los nombres de los archivos a comprimir
|     |     |           CUALES="esptemp.* historia.*"
&& Dependiendo de la unidad destino ejecutamos el comando correspondiente
|     |     |     +-------IF UNIDAD=UPPER('A:\') OR UNIDAD=UPPER('B:\')
|     |     |     |           COMANDO=RUTAZIP+" -min -a -ex
                                    "+UNIDAD+ALLTRIM(NOMBRECOPIA)+"
                                    "+RUTA+CUALES
|     |     |     +-------ELSE
|     |     |     |           COMANDO=RUTAZIP+" -min -a -ex
                                    "+UNIDAD+ALLTRIM(NOMBRECOPIA)+"
                                    "+RUTA+CUALES
|     |     |     +-------ENDIF
&& Ejecutamos el comando por medio de la API de Windows
|     |     |           =WINEXEC('&comando',SW_SHOWMINIMIZED)
&& Actualizamos el campo marca tanto en historicos de precios como de especificaciones
&& para saber que esta informacion no esta disponible y debe eliminarse al correr
&& el proceso de limpiar Base de Datos
|     |     |           UPDATE V_PHISTORIA SET MARCA = .T. WHERE MARCA = .F.
|     |     |           UPDATE V_PESPTEMP SET MARCA = .T. WHERE MARCA = .F.
|     |     |
|     |     |           PROGRESO.BARRAPROGRESO1.LLENE
|     |     |           PROGRESO.RELEASE
&&Retornamos a la ruta original desde donde esta el aplicativo trabajando
```

```
|     |     |          SET DEFAULT TO &ACTUAL
|     |     |          WAIT WINDOW('Generación de Copia Historicos
                                    terminada con éxito.'+CHR(13)+'Cierre la ventana de
                                    Winzip si es nesario'+CHR(13)+'cuando termine de
                                    comprimir') TIMEOUT 3
|     |     +-------CASE MENSAJE = 7
|     |     |          MENSAJE=MESSAGEBOX(" Proceso Cancelado !
                                  ",0+64+0," Información ")
|     |     +-------ENDCASE
&&En caso de que las vistas no hayan obtenido informacion lo informamos
|     +-------ELSE
|     |          MENSAJE=MESSAGEBOX(" No existe informacion en los
                              Historicos ! ",0+64+0," Información ")
|     +-------ENDIF
|
+-------ENDPROC
```

5.2.4 Combo para seleccionar Mes

```
+-------PROCEDURE VALID
&&Validamos la consistencia de la informacion seleccionada para habilitar o no
&&el boton de procesar la informacion
|     +-------IF (!EMPTY(THISFORM.COMBO1.VALUE) AND
                        !ISNULL(THISFORM.COMBO1.VALUE)) AND (!EMPTY(THIS.VALUE) AND
                        !ISNULL(THIS.VALUE))
|     |          THISFORM.COMMAND2.ENABLED=.T.
|     +-------ELSE
|     |          THISFORM.COMMAND2.ENABLED=.F.
|     +-------ENDIF
+-------ENDPROC
```

5.2.5 Combo para seleccionar Año

```
+-------PROCEDURE GOTFOCUS
&&Refrescamos la informacion de la vista a partir de su origen
|          REQUERY('v_anohist')
+-------ENDPROC

+-------PROCEDURE VALID
&&Validamos la consistencia de la informacion
|     +-------IF !EMPTY(THIS.VALUE)
&&Asignamos valor a las variables
|     |          ANOHIST=THIS.VALUE
|     |          THISFORM.COMBO2.ENABLED=.T.
&&Al seleccionar un año extractamos informacion correspondiente alos meses de los
&&cuales se tiene informacion para este año
```

```
|       |           REQUERY('V_MESHIST')
|       |           SELECT V_MESHIST
&&Asignamos al origen de datos del combo destinado para desplegar los meses la vista
&&que los contiene
|       |           THISFORM.COMBO2.ROWSOURCE = 'v_meshist.mes'
|       |           THISFORM.COMBO2.ROWSOURCETYPE = 2
|       |           THISFORM.COMBO2.CONTROLSOURCE = 'v_meshist.mes'
|       +-------ENDIF
+-------ENDPROC
```

5.3 FORMULARIO GENERAR COPIA DE SEGURIDAD (copiaseg.scx)

5.3.1 Vistas

v_rutabd, v_registro

```
+-------PROCEDURE SHOW
|           LPARAMETERS NSTYLE
|
+-------ENDPROC

+-------PROCEDURE INIT
&&Definimos e inicializamos variables
|           PUBLIC UNIDAD,CODENVIO
|           UNIDAD=''
&&Configuramos el ambiente de trabajo para el manejo de fechas
|           SET CENTURY ON
|           SET DATE DMY
&&Refrescamos la informacion del archivo de registro para saber si hay usuarios
&&utilizando el aplicativo para podre procesar la copia de seguridad
&&Puesto que ningun archivo debe estarse usando
|           REQUERY('v_registro')

|           SELE V_REGISTRO
&&Verificamos que no haya otra pantalla abierta ya que se estaria usando
&&alguna archivo, en ese caso habilitamos el boton de buscar donde generar
&&la Copia de Seguridad
|       +-------IF EOF()
|       |           THISFORM.COMMAND2.ENABLED = .T.
&&Utilizamos la ruta que se encuentra en el archivo destinado para tal fin
&&de todas maneras solicitamos que verifique dicha ruta, ya que de esta depende
&&que el aplicativo ubique la informacion a comprimir
|       |           SELECT V_RUTABD
|       |       +-------IF !EOF()
|       |       |           THISFORM.RUTABD.VALUE = ALLTRIM(V_RUTABD.RUTABD)
|       |       |           = MESSAGEBOX('Verifique que la Ruta B.D. Origen
```

```
                                                sea la correcta, en caso de no ser valida dirijase
                                                a:'+CHR(13)+CHR(13)+'Herramientas - Asignar Ruta de
                                                Base de Datos !!',0+64+0,'Verificar')
|     |     |           THISFORM.COMMAND2.ENABLED = .T.
|     |     |           THISFORM.COMMAND2.SETFOCUS
|     |     +-------ELSE
&&En caso de que en el archivo de ruta de Base de Datos no se haya definido aun
&&ruta alguna, se solicita para poder trabajar estos procesos
|     |     |           = MESSAGEBOX('Aun no se ha asignado Ruta de la
                                                Base de Datos Origen (Servidor)  !!, dirijase
                                                a:'+CHR(13)+CHR(13)+'Mantenimiento - Asignar Ruta de
                                                Base de Datos',0+64+0,'Corregir')
|     |     |           THISFORM.COMMAND2.ENABLED = .F.
|     |     +-------ENDIF
|     |           USE
|     +-------ELSE
&&En caso de encontrar otro usuario trabajando el aplicativo o que el usuario local
&&tenga abierta otra pantalla, informamos que no se puede procesar la tarea
|     |           LOCATE FOR
                              ALLTRIM(V_REGISTRO.QUIEN)+ALLTRIM(V_REGISTRO.NPANTALL
                              A) != ALLTRIM(PQUIEN)+WONTOP( )
|     |     +-------IF FOUND()
|     |     |           =MESSAGEBOX('NO se puede procesar la orden, esta
                                                siendo usado el aplicativo por
                                                '+ALLTRIM(V_REGISTRO.QUIEN)+' ó no ha cerrado
                                                todas las pantallas!!..',0+64+0,'Informacion')
|     |     |           THISFORM.COMMAND2.ENABLED = .F.
|     |     +-------ENDIF
|     +-------ENDIF
|           SELE V_REGISTRO
|           USE
|
+-------ENDPROC
```

5.3.2 Botón Buscar

```
+-------PROCEDURE CLICK
&&DEclaramos e inicializamos variables
|           LOCAL GCFILE,DIR1
|           DIR1=SYS(5)
|           LOGICA=DIRECTORY(DIR1+"\"+"IpcBack")
&&Verificamos la existencia de la carpeta que se recomienda utilizar
&&En caso de que no exista la carpeta el aplicativo la crea automaticamente
|     +-------IF LOGICA
|     +-------ELSE
|     |           MKDIR DIR1+"\"+"IpcBack"
|     +-------ENDIF
```

```
&&Se captura la ruta seleccionada
|           GCFILE=GETDIR(DIR1+"\"+"IpcBack")
|
|           UNIDAD=GCFILE
&&Si se selecciono alguna carpeta se muestra en el formulario y habilitamos el boton
&&para procesar
|     +-------IF NOT EMPTY(GCFILE)
|     |           THISFORM.NOMBREARCHIVO.VALUE = GCFILE
|     |           THISFORM.INICIAR.ENABLED=.T.
|     +-------ENDIF
+-------ENDPROC
```

5.3.3 Botón Salir

```
+-------PROCEDURE CLICK
|           THISFORM.RELEASE
|
+-------ENDPROC
```

5.3.4 Botón Iniciar

```
+-------PROCEDURE CLICK
&&Definimos el nombre del archivo a generar, dicho nombre incluye
&&departamento y fecha
                NOMBRECOPIA
                =ALLTRIM(DEPARTAMENTO)+PADL(ALLTRIM(STR(DAY(DATE()))),2,'0')+PADL(ALL
                TRIM(STR(MONTH(DATE()))),2,'0')+SUBSTR(STR(YEAR(DATE()),4,0),3,2)
|
&&Configuramos ambiente de trabajo para el manejo de archivos
|           SET SAFETY OFF
|           SET TALK OFF
|
&&DEfinimos variable y la manera como se va a visualizar el winzip al ser ejecutado
|           LOCAL RUTAZIP
|           #DEFINE SW_SHOWMINIMIZED 2
|
&&Dependiendo de la unidad de destino seleccionada
|     +-------IF UNIDAD=UPPER('A:\') OR UNIDAD=UPPER('B:\')
                                MENSAJE= MESSAGEBOX("Ya insertó el disquete en la unidad
                                "+UNIDAD +" ?",4+64+256," Generación de copia de seguridad ")
|     |     +-------IF MENSAJE=6 AND (DISKSPACE(UNIDAD)= -1)
                                MESSAGEBOX("Ha ocurrido un error al intentar leer la unidad de disco.
                                "+UNIDAD +" ?",0+64+0," Precaución ")
|     |     |           MENSAJE=7
|     |     +-------ENDIF
|     +-------ELSE
|     |           MENSAJE=6
```

```
|       +-------ENDIF
|
|       +-------DO CASE
|       +-------CASE MENSAJE = 6
&&Cargamos variable con ruta actual para luego de finalizar la generacion de Copia de Seguridad
&&retornar a esta
|       |               ACTUAL=FULLPATH(SYS(2003))
|       |               &&Ejecutamos la barra de progreso
|       |               DO FORM FPROGRESO NAME PROGRESO LINKED
                                        PROGRESO.LPROGRESO.CAPTION='Generando Copia de Seguridad
                                        en: '+UNIDAD
|       |               PROGRESO.BARRAPROGRESO1.MAXVALOR=3000
|       |               PROGRESO.BARRAPROGRESO1.MINVALOR=1
&&Avanzamos
                                        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRA
                                        PROGRESO1.PROGRESO+200
|       |               PROGRESO.REFRESH
|       |               PROGRESS()
|       |               CUALES = 'iPc*.* '
                                        CUALES1 = 'esptemp.dbf esptemp.cdx historia.dbf historia.cdx '
|       |               CUALES2 = ' rutabd.dbf registro.dbf'
                                        ipctctv.* ipcestr.* ipcmart.* ipcmfte.* ipcrango.* ipctmvt.* ipcrobs.*
                                        ipcresp.*       ipcrfap.* ipcrpart.* ipctfcm.* rutabd.* registro.*'
&&Ubicamos el Winzip seun el idioma de la version de Windows
|       |       +-------IF FILE('c:\archivos de programa\winzip\winzip32.exe')
                                        RUTAZIP='c:\archivos de programa\winzip\winzip32.exe'
|       |       +-------ELSE
|       |       |               RUTAZIP='c:\program files\winzip\winzip32.exe'
|       |       +-------ENDIF
|       |
&&Segun la unidad seleccionada como destino definimos la orden a ejecutar
|       |       +-------IF UNIDAD=UPPER('A:\') OR UNIDAD=UPPER('B:\')
                                        COMANDO='&rutazip'+' -min -a -ex '+UNIDAD+NOMBRECOPIA+'
                                        '+ALLTRIM(THISFORM.RUTABD.VALUE)+'*.*'
|       |       +-------ELSE
                                                COMANDO='&rutazip'+' -min -a -ex
                                                '+UNIDAD+NOMBRECOPIA+'
                                                '+ALLTRIM(THISFORM.RUTABD.VALUE)+'*.*'
|       |       |
|       |       +-------ENDIF
&&Avanzamo barra de progreso
                                        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRA
                                        PROGRESO1.PROGRESO+200
|       |               PROGRESO.REFRESH
|       |               PROGRESS()
&&Avanzamo barra de progreso
                                        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRA
                                        PROGRESO1.PROGRESO+300
```

```
|       |           PROGRESO.REFRESH
|       |           PROGRESS()
&&Ejecutamos la orden definida utilizando la API de Windows
|       |           =WINEXEC('&comando',SW_SHOWMINIMIZED)
|       |
&&Llenamos la barra de progreso y la cerramos
|       |           PROGRESO.BARRAPROGRESO1.LLENE
|       |           PROGRESO.RELEASE
&&Retornamos a la ruta original desde donde se esta ejecutando el aplicativo
|       |           SET DEFAULT TO &ACTUAL
                                MENSAJE6=MESSAGEBOX(' Generación de Copia de Seguridad
                                terminada con éxito,'+CHR(13)+' cierre por favor la ventana de Winzip si
                                es necesario !!.'+CHR(13)+'Cuando termine de Comprimir',0+64+0,"
                                INFORMACION ")
|       |
|       +-------CASE MENSAJE = 7
                        MENSAJE=MESSAGEBOX(" Cancelada la Generación de Copia de Seguridad
                        ! ",0+64+0," Información ")
|       +-------ENDCASE
|
+-------ENDPROC
```

5.4 FORMULARIO PARA ELIMINAR USUARIOS(e_usuario.scx)

5.4.1 Vistas

v_auxiliar, v_registro

```
+-------PROCEDURE ACTIVATE
&&Al entrar al formulario actualizamos el archivo de registro para saber
&&que este usuario abrio este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para saber
&&si ya esta relacionado en el archivo y no voilver a registrarlo
|           LOCATE FOR (ALLTRIM(V_REGISTRO.QUIEN) + NPANTALLA =
                        ALLTRIM(PQUIEN) + WONTOP()) AND V_REGISTRO.ACT_INA = .T.
&&si no es encontrado lo registramos y actualizamos el archivo
|       +-------IF !FOUND()
|       |           INSERT INTO V_REGISTRO(QUIEN, NPANTALLA, FECHAR,
                                    ACT_INA);
|       |                 VALUES(PQUIEN,WONTOP( ), DATE(), .T.)
|       |           TABLEUPDATE(.T.)
|       +-------ENDIF
```

```
|
+-------ENDPROC

+-------PROCEDURE INIT
|           SET SYSMENU OFF
&&Ocultamos el menu del aplicativo
&&Declaramos e Inicializamos variables
                PUBLIC PQUIEN, PCONQUE, PQUE, PQUIENACTUAL, PCONQUEACTUAL,
                PQUEACTUAL
|
|           PQUIEN=" "
|           PCONQUE=" "
|           PQUE=" "
|
&&Configuramos el ambiente de trabajo para el manejo de fechas
|           SET CENTURY ON
|           SET DATE DMY
|
&&Creamos cursor para almacenar las claves de ususarios desencriptadas temporalmente
&&y asi poder eliminar los usuarios sin necesidad de que el administrador conozca la misma
&&de los usuarios registrados en el aplicativo
|           CREATE CURSOR CLAVES ;
                        (QUIEN C(20), QUIENACTUAL C(20), CONQUE C(8), CONQUEACTUAL C(8),
                        QUE C(1),QUEACTUAL C(1))
|
|           THISFORM.TEXT1.SETFOCUS
&&Cargamos variables con las claves desencriptadas
&&Desencriptamos la información de los usuario
|           *************************
|           SELE V_AUXILIAR
|     +-------SCAN FOR ACTIVO = .T.
|     |           PQUIEN = V_AUXILIAR.QUIEN
|     |           PQUIENACTUAL = V_AUXILIAR.QUIEN
|     |           PCONQUE = V_AUXILIAR.CONQUE
|     |           PCONQUEACTUAL = V_AUXILIAR.CONQUE
|     |           PQUE = V_AUXILIAR.QUE
|     |           PQUEACTUAL = V_AUXILIAR.QUE
|     |     +-------FOR I=1 TO 20
|     |     |           LETRA=SUBSTR(PQUIEN,I,1)
|     |     |           NLETRA=CHR(ASC(LETRA)-10)
|     |     |           PQUIEN=STUFF(PQUIEN,I,1,NLETRA)
|     |     |
|     |     +-------ENDFOR
|     |
|     |     +-------FOR I=1 TO 8
|     |     |           LETRA=SUBSTR(PCONQUE,I,1)
|     |     |           NLETRA=CHR(ASC(LETRA)-10)
|     |     |           PCONQUE=STUFF(PCONQUE,I,1,NLETRA)
```

```
|       |       +-------ENDFOR
|       |
|       |       +-------FOR I=1 TO 1
|       |       |               LETRA=SUBSTR(PQUE,I,1)
|       |       |               NLETRA=CHR(ASC(LETRA)-10)
|       |       |               PQUE=STUFF(PQUE,I,1,NLETRA)
|       |       +-------ENDFOR
&&Cargamos el cursor
                                INSERT INTO CLAVES(QUIEN, QUIENACTUAL, CONQUE,
                                CONQUEACTUAL, QUE, QUEACTUAL);
                                VALUES (PQUIEN, PQUIENACTUAL, PCONQUE, PCONQUEACTUAL, PQUE,
                                PQUEACTUAL)
|       |
|       +-------ENDSCAN
|               THISFORM.TEXT1.VALUE=" "
|               THISFORM.TEXT2.VALUE=" "
|               THISFORM.TEXT3.VALUE=" "
+-------ENDPROC

+-------PROCEDURE UNLOAD
&&Desplegamos el menu del aplicativo nuevamente
|               SET SYSMENU ON
+-------ENDPROC

+-------PROCEDURE DESTROY
&&Al salir del formulario actualizamos el archivo de registro para saber
&&que este usuario ya salio de este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|               REQUERY('v_registro')
|               SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para indicar
&&que la cerro en este momento
|               LOCATE FOR
                                ALLTRIM(V_REGISTRO.QUIEN)+ALLTRIM(V_REGISTRO.NPANTALLA) =
                                ALLTRIM(PQUIEN)+WONTOP( )
|       +-------IF FOUND()
|       |               REPLACE V_REGISTRO.ACT_INA WITH .F.
&&Actualizamos la informacion y tambien el archivo
|       |               TABLEUPDATE(.T.)
|       +-------ENDIF
|
          +-------ENDPROC
```

5.4.2 Botón Salir

```
+-------PROCEDURE CLICK
|               THISFORM.RELEASE
```

```
|
+-------ENDPROC
```

5.4.3 Botón Eliminar

```
+-------PROCEDURE CLICK
&&Confirmamos la decision de eliminar el usuario seleccionado
|           MENSAJE=MESSAGEBOX(" Esta seguro de ELIMINAR a
                    "+CHR(13)+CHR(13)+ALLTRIM(PQUIEN)+" ?",4+32+256," Eliminar Usuarios ")
|     +-------DO CASE
&&En caso afirmativo
|     +-------CASE MENSAJE=6
|     |           SELE V_AUXILIAR
&&Localiza el usuario y lo inactivamos
|     |           LOCATE FOR V_AUXILIAR.QUIEN = PQUIENACTUAL AND
                            V_AUXILIAR.CONQUE = PCONQUEACTUAL AND V_AUXILIAR.QUE =
                            PQUEACTUAL
|     |     +-------IF FOUND()
|     |     |           REPLACE ACTIVO WITH .F.
|     |     |           THISFORM.ACEPTAR.ENABLED=.F.
&&Procesamos de nuevo la creacion y desencriptar la informacion de usuarios
&&y cargarla a este cursor temporalmente
|     |     |     CREATE CURSOR CLAVES ;
                            (QUIEN C(20),QUIENACTUAL C(20), CONQUE C(8),CONQUEACTUAL
                            C(8), QUE C(1),QUEACTUAL C(1))
|     |     |
|     |     |           SELE V_AUXILIAR
&&Recorremos los usuarios que se encuentran activos
|     |     |     +-------SCAN FOR ACTIVO = .T.
|     |     |     |           PQUIEN = V_AUXILIAR.QUIEN
|     |     |     |           PQUIENACTUAL = V_AUXILIAR.QUIEN
|     |     |     |           PCONQUE = V_AUXILIAR.CONQUE
|     |     |     |           PCONQUEACTUAL = V_AUXILIAR.CONQUE
|     |     |     |           PQUE = V_AUXILIAR.QUE
|     |     |     |           PQUEACTUAL = V_AUXILIAR.QUE
&&desencriptamos el nombre de usuario
|     |     |     |     +-------FOR I=1 TO 20
|     |     |     |     |           LETRA=SUBSTR(PQUIEN,I,1)
|     |     |     |     |           NLETRA=CHR(ASC(LETRA)-10)
|     |     |     |     |           PQUIEN=STUFF(PQUIEN,I,1,NLETRA)
|     |     |     |     |
|     |     |     |     +-------ENDFOR
&&desencriptamos la clave de usuario
|     |     |     |     +-------FOR I=1 TO 8
|     |     |     |     |           LETRA=SUBSTR(PCONQUE,I,1)
|     |     |     |     |           NLETRA=CHR(ASC(LETRA)-10)
                                                PCONQUE=STUFF(PCONQUE,I,1,NLETR
                                                A)
```

```
|     |     |     |     +-------ENDFOR
&&desencriptamos el nivel de acceso de usuario
|     |     |     |     +-------FOR I=1 TO 1
|     |     |     |     |           LETRA=SUBSTR(PQUE,I,1)
|     |     |     |     |           NLETRA=CHR(ASC(LETRA)-10)
|     |     |     |     |           PQUE=STUFF(PQUE,I,1,NLETRA)
|     |     |     |     +-------ENDFOR
|     |     |     |
                                          INSERT INTO CLAVES (QUIEN, QUIENACTUAL,
                                          CONQUE, CONQUEACTUAL, QUE, QUEACTUAL);
                                          VALUES (PQUIEN, PQUIENACTUAL, PCONQUE,
                                          PCONQUEACTUAL, PQUE, PQUEACTUAL)
|     |     |     |
|     |     |     +-------ENDSCAN
|     |     |           THISFORM.LIST1.REFRESH
|     |     |           THISFORM.TEXT1.VALUE=" "
|     |     |           THISFORM.TEXT2.VALUE=" "
|     |     |           THISFORM.TEXT3.VALUE=" "
|     |     |
|     |     |           THISFORM.CANCELAR.SETFOCUS
|     |     +-------ENDIF
|     +-------CASE MENSAJE=7
                        MENSAJE=MESSAGEBOX(" Usuario no Eliminado ",0+64+0," Eliminar
                        Usuarios ")
|     +-------ENDCASE
|
+-------ENDPROC
```

5.4.4 Escriba la actual clave de acceso

```
+-------PROCEDURE LOSTFOCUS
&&Validamos la consistencia de la información y dependiendo habilitamos el botón
|     +-------IF VAL(m.QUE)<1 OR VAL(m.QUE)>3
                              MENSAJE=MESSAGEBOX(" Nivel fuera de rango, corrija por favor
                              ",0+48+0," Modificar Usuarios ")
|     |           THISFORM.TEXT3.SETFOCUS
|     |           THISFORM.ACEPTAR.ENABLED=.F.
|     +-------ELSE
|     |           THISFORM.ACEPTAR.SETFOCUS
|     |
|     +-------ENDIF
+-------ENDPROC

+-------PROCEDURE VALID
+-------IF LEN(ALLTRIM(m.QUIEN))>=4 AND LEN(ALLTRIM(m.CONQUE))>=4 AND
                VAL(m.QUE)<>0
                THISFORM.ACEPTAR.ENABLED=.T.
|     |
```

```
|       +-------ENDIF
+-------ENDPROC

+-------PROCEDURE GOTFOCUS
|       +-------IF LEN(ALLTRIM(m.CONQUE1))<4
|       |               THISFORM.TEXT4.SETFOCUS
|       |               THISFORM.ACEPTAR.ENABLED=.F.
|       +-------ENDIF
+-------ENDPROC
```

5.4.5 Evento Haga Click sobre Nombre del Usuario que desea eliminar Objeto List

```
+-------PROCEDURE CLICK
&&Al seleccionar uno de los usuarios que aparecen e la lista mostramos su información
&&en las diferentes cajas de texto
|               THISFORM.TEXT1.VALUE = CLAVES.QUIEN
|               THISFORM.TEXT2.VALUE = CLAVES.CONQUE
|               THISFORM.TEXT3.VALUE = CLAVES.QUE
&&Cargamos las variables con los valores correspondientes al usuario
|               PQUIEN = THISFORM.TEXT1.VALUE
|               PQUIENACTUAL = CLAVES.QUIENACTUAL
|               PCONQUEACTUAL = CLAVES.CONQUEACTUAL
|               PQUEACTUAL = CLAVES.QUEACTUAL
|               THISFORM.TEXT1.REFRESH
|               THISFORM.TEXT2.REFRESH
|               THISFORM.TEXT3.REFRESH
|               THISFORM.ACEPTAR.ENABLED=.T.
+-------ENDPROC
```

5.5 CAMBIO DE ESPECIFICACION (FESPECIFICA.SCX)

5.5.1 Vistas

v_resppa
v_registro

```
+-------PROCEDURE ACTIVATE
&&Al entrar al formulario actualizamos el archivo de registro para saber
&&que este usuario abrio este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|               REQUERY('v_registro')
|               SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para saber
&&si ya esta relacionado en el archivo y no voilver a registrarlo
```

```
                    LOCATE FOR (ALLTRIM(V_REGISTRO.QUIEN) + NPANTALLA = ALLTRIM(PQUIEN)
                    + WONTOP()) AND V_REGISTRO.ACT_INA = .T.
&&si no es encontrado lo registramos y actualizamos el archivo
|       +-------IF !FOUND()
                              INSERT INTO V_REGISTRO(QUIEN, NPANTALLA, FECHAR, ACT_INA);
|       |           VALUES(PQUIEN,WONTOP( ), DATE(), .T.)
|       |               TABLEUPDATE(.T.)
|       +-------ENDIF
|
+-------ENDPROC
+-------PROCEDURE SHOW
|               LPARAMETERS NSTYLE
&&Permitimos que ejecute el codigo por defecto de la clase
|               DODEFAULT()
&&Seleccionamos la vista y la filtramos segun el campo marca, este nos indica
&&en caso de que el registro este marcadao para ser boorrado
&&Refrescamos información en pantalla
|               SELECT V_RESPPA
|               SET FILTER TO V_RESPPA.MARCA = .F.
|               THISFORM.GESPCOPIA.REFRESH
|               THISFORM.REFRESH
+-------ENDPROC
+-------PROCEDURE INIT
|               LPARAMETERS NOFTE,NOARTICULO,ARTICULO
&&Recibimos los parametros que nos indican que proceso se va allevar a cabo,
&&es decir, un cambio de referencia o una consulta de especificaciones
|               RESPAFTE = NOFTE
|               RESPART = NOARTICULO
|
&&Actualizamos el archivo que carga la informacion para este proceso
|               REQUERY('v_resppa')
|               SELECT V_RESPPA
&&Actualizamos en pantalla segun los parametros que recibimos
|       +-------WITH THISFORM
|       |               .TCODART.VALUE=NOARTICULO
|       |               .TNOMART.VALUE=ARTICULO
|       |
|       |               .BCANCELAR.VISIBLE=CR
|       |               .BELIMINAR.VISIBLE=CR
|       |               .BINSERTAR.VISIBLE=CR
|       |
|       |       +-------IF CR
&&Segun los parametros mostramos la cuadricula correspondiente
&&ya se para consultar especificaciones o para cambio de referencia
|       |       |               THISFORM.CAPTION=" Cambio de Referencia "
|       |       |               .GESPCOPIA.READONLY=!CR
|       |       |               .BACEPTAR.LEFT=29
|       |       +-------ELSE
```

```
|     |     |          THISFORM.CAPTION=" Consulta de Especificaciones "
|     |     |          .GESPCOPIA.ENABLED=!CR
|     |     |          .GESPCOPIA.READONLY=CR
&&Activamos o inactivamos botones
                                        .BACEPTAR.LEFT=252 &&(thisform.width%2)-
                                        (.BAceptar.width%2)
|     |     +-------ENDIF
|     +-------ENDWITH
|
+-------ENDPROC
+-------PROCEDURE DESTROY
&&Al salir del formulario actualizamos el archivo de registro para saber
&&que este usuario ya salio de este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para indicar
&&que la cerro en este momento
                        LOCATE FOR
                        ALLTRIM(V_REGISTRO.QUIEN)+ALLTRIM(V_REGISTRO.NPANTALLA) =
                        ALLTRIM(PQUIEN)+WONTOP( )
|     +-------IF FOUND()
|     |           REPLACE V_REGISTRO.ACT_INA WITH .F.
&&Actualizamos la informacion y tambien el archivo
|     |           TABLEUPDATE(.T.)
|     +-------ENDIF
+-------ENDPROC
```

5.5.2 Botón Aceptar

```
+-------PROCEDURE CLICK
&&Seleccionamos el archivo y actualizamos la informacion digitada
&&e inactivamos el boton cancelar
|           SELECT V_RESPPA
&&Actualizamos la información hacia la vista
|           TABLEUPDATE(.T.)
|           THISFORM.BCANCELAR.ENABLED=.F.
|
|           THISFORM.RELEASE
+-------ENDPROC
```

5.5.3 Botón Cancelar

```
+-------PROCEDURE CLICK
&&Seleccionamos la vista
|           SELECT V_RESPPA
&&Deshacemos los cambios hechos para que no se almacenen en el archivo
```

```
|           TABLEREVERT(.T.)
&&Refrescamos la información en panatalla
|           THISFORM.GESPCOPIA.REFRESH
|           THISFORM.REFRESH
+-------ENDPROC
```

5.5.4 Botón Eliminar

```
+-------PROCEDURE CLICK
|           THISFORM.GESPCOPIA.SETFOCUS
&&Para el caso de eliminar información simplemente la inactivamos para que no se tenga en cuenta
&&pero queda en el archivo mientras no se procese limpiar B.D.
|           SELECT V_RESPPA
|           REPLACE V_RESPPA.MARCA WITH .T.
&&Si no es el ultimo registro en el archivo, avanza uno
|     +-------IF !EOF()
|     |           SKIP 1 IN 'v_respPA'
|     +-------ENDIF
&&Refrescamos información en pantalla
|           THISFORM.BCANCELAR.ENABLED=.T.
|           THISFORM.GESPCOPIA.REFRESH
|           THISFORM.REFRESH
+-------ENDPROC
```

5.5.5 Botón Adicionar

```
+-------PROCEDURE CLICK
&&Seleccionamos la vista, nos vamos al primer último registro de la misma
|           SELECT V_RESPPA
|           GO BOTTOM
&&Al valor del codigo de especificación mayor le sumamos uno para que sirva de codigo
&&para la nueva especificacion
|           XX = VAL(V_RESPPA.CODESP)+1
|           NUEVOVALOR = ALLTRIM(STR(XX,2,0))
&&Desactivamos el filtro temporalmente para tener acceso a los registros marcados
&&y encaso de requerirse agregar una especificacion y haya un registro con marca .T.
&&reutilizarlo colocando marca a .F. para que lo tome el filtro y nos lo muestre en pantalla
|           SET FILTER TO
&&Validamos que no se creen mas de 99 especificaciones que el limite por ahora
|     +-------IF XX<100
&&Seleccinamos la vista y buscamos la informacion correspondiente a la fuente - articulos -
especfificacion
&&En caso de ser encontrada, (esto sucederia si esta marcada para borrar),
&&procedemos a desmarcarla y que quede activa
&&Limpiamos el contenido de los campos nombre de especificacion y de detalle  select v_resppa
```

```
        LOCATE FOR V_RESPPA.CODFTE+V_RESPPA.CODART+V_RESPPA.CODESP =
        RESPAFTE+RESPART+PADL(NUEVOVALOR, 2, '0')
|       |       +-------IF FOUND()
                                REPLACE V_RESPPA.MARCA WITH .F., V_RESPPA.NOMBDET WITH
                                '', V_RESPPA.NOMESP WITH '' FOR (V_RESPPA.CODFTE =
                                RESPAFTE) AND (V_RESPPA.CODART = RESPART) AND
                                (V_RESPPA.CODESP=PADL(NUEVOVALOR, 2, '0'))
|       |       +-------ELSE
&&En caso de no ser encontrada la informacion, es insertada como nueva
                                        INSERT INTO V_RESPPA(CODFTE, CODART, CODESP,
                                        MARCA) ;
                                                VALUES (RESPAFTE, RESPART, PADL(NUEVOVALOR,
                                                2, '0'), .F.)
|       |       +-------ENDIF
|       +-------ENDIF
|
&&Redefinimos la configuracion de manera que entre la información mostrada no se
&&tengan en cuenta los que estan marcados para eliminar
|               SELEC V_RESPPA
|               SET FILTER TO V_RESPPA.MARCA = .F.
|
|       +-------WITH THISFORM.GESPCOPIA
|       |               .REFRESH
|       |               .SETFOCUS
|       +-------ENDWITH
+-------ENDPROC
```

5.5.6 Entorno de Datos

```
+-------PROCEDURE INIT
&& Creación de Indices
|               SELE V_IPCROBS
|               INDEX ON STR(IMPCTV,7,0)+CODFTE+CODOBS TAG PKOBS
|
|               SELE V_IPCMFTE
|               INDEX ON CODFTE TAG PKCODFTE
|
|               SELE  V_IPCMART
|               INDEX ON CODART TAG PKCODART4
+-------ENDPROC
```

5.6 FORMULARIO INICIAL (FINICIAL.SCX)

5.6.1 Vistas

v_ipcmfte
divipola
v_rutabd

```
+-------PROCEDURE INIT
&&Declaramos variables publicas para almacenar el codigo de departamento y el del municipio
&&y para almacenar la version del sistema operativo en el que se ejecuto el aplicativo
&&Consultamos el nombre de la ciudad donde se esta trabajando
&&segun la informacion que se encuentra en el archivo maestro de fuente
|
|           PUBLIC DEPARTAMENTO, MUNICIPIO, SISTOPE, PORCIUDAD, PRUTABD
&&Almacenamos la version del sistema operativo en la variable
|           SISTOPE=SUBSTR(OS(),9,2)
|
&&Del maestro de fuentes extractamos la información del Departamento y del Municipio propio de la
ciudad
&&para usarlos porteriormente en las diferentes pantallas, reportes, nombres de archivos como
&&envio o Copia de Seguridad y cerramos el archivo de una vez
|           SELE V_IPCMFTE
|           DEPARTAMENTO=SUBSTR(CODFTE,1,2)
|           MUNICIPIO=SUBSTR(CODFTE,3,3)
|           USE
|
&&Cargamos variable con el nombre de la ciudad
        |
        PORCIUDAD=ALLTRIM(CIUDAD(ALLTRIM(DEPARTAMENTO)+ALLTRIM(MUNICIPIO)))
|           SELE DIVIPOLA
|           USE
|
&&Verificamos que ya se haya asignado ruta de la Base de Datos la cual sera utilizada
&&en posteriores procesos como: Copia de Seguridad, Envio, Reindexar, etc..
&&Si aun no la han capturado lo solicitamos
|           SELE V_RUTABD
|     +-------IF !EOF()
|     |           PRUTABD = ALLTRIM(V_RUTABD.RUTABD)
|     +-------ELSE
                        =MESSAGEBOX('Aun no se ha asignado ruta de la Base de
                        Datos,'+CHR(13)+'por favor dirijase a mantenimiento - Ruta de Base de Datos y
                        asignela..',0,'Informacion')
|     +-------ENDIF
|           USE
|
```

```
&&Declaración de Variables
|           PUBLIC FTEPAFTE
|           PUBLIC FAPPAFTE, FAPPART
|           PUBLIC RESPAFTE, RESPART
|           PUBLIC OFTEPAFTE, ORESPAFTE
|
|
+-------ENDPROC

+-------PROCEDURE ACTIVATE
&&Definimos el estado de la ventana a mostrar, es decir, maximizada
&&en este caso
|           THISFORM.WINDOWSTATE=2
+-------ENDPROC
```

5.7 FORMULARIO PARA BARRA DE PROGRESO (FPROGRESO.scx)

```
+-------PROCEDURE INIT
&&Configuramos para que al hacer clic el usuario no genere error o la cierre
&&mientras se esta ejecutando
|           SET ESCAPE OFF
+-------ENDPROC
```

5.8 FORMULARIO PARA CAPTURAR FUENTES COMPLEMENTARISAS (FTESCOM.SCX)

5.8.1 Vistas

v_ipctfcm
v_registro

```
+-------PROCEDURE INIT
|           PARAMETERS NCONSECUTIVO,CNOFTE,CNOMBRE,COBS
&&Recibimos los parametros necesarios para el manejo de
&&las fuentes complementarias
|           PUBLIC SALIR
|           SALIR=.T.
|
&&Refrescamos informacion en pantalla
|     +-------WITH THISFORM
|     |           .CODOBS=COBS
|     |           .NNOFORMULARIO=NCONSECUTIVO
|     |           .TNOFORMULARIO.VALUE=.NNOFORMULARIO
|     |           .NOMFTE=CNOMBRE
|     |           .CCODFTE=CNOFTE
|     |           .ENOMBREFUENTE.VALUE=.NOMFTE
```

```
|       |               .CODFTE1.VALUE=CNOFTE
|       +-------ENDWITH
&&Filtramos la informacion con respecto al campo marca el cual nos indica si esta marcado para
&&eliminar o no, mostramos lo que nos estan marcados
&&Configuramos como se va a mostrar la informacion que provienen de campos que
&&aceptan valores nulos
|               SET FILTER TO V_IPCTFCM.MARCA = .F.
|               SET NULLDISPLAY TO [ ]
|               DODEFAULT()
+-------ENDPROC

+-------PROCEDURE SHOW
|               LPARAMETERS NSTYLE
&&Almacenamos en variables los parametros recibidos
|               NOFORMULARIO=THIS.NNOFORMULARIO
|               NOFTE=THIS.CCODFTE
|               NOARTICULO=THIS.CCODART
&&Reconsultamos la vista de las fuentes complementarias
|
|               REQUERY('V_IPCTFCM')
|               SELECT V_IPCTFCM
&&Validamos que el valor del codigo de observacion digitado
&&no este vacio
|       +-------IF !EMPTY(THIS.CODOBS)
|       |               LOCATE FOR V_IPCTFCM.CODFCOM = THIS.CODOBS
|       |       +-------IF !FOUND()
&&Si no es encontrado en el archivo creamos una nueva observacion
&&con este codigo
|       |       |               THISFORM.NAVIGATOR.CMDADD.CLICK
|       |       |               REPLACE V_IPCTFCM.CODFCOM WITH THIS.CODOBS
|       |       +-------ENDIF
|       +-------ENDIF
|
+-------ENDPROC

+-------PROCEDURE ACTIVATE
&&Al entrar al formulario actualizamos el archivo de registro para saber
&&que este usuario abrio este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|               REQUERY('v_registro')
|               SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para saber
&&si ya esta relacionado en el archivo y no voilver a registrarlo
                                LOCATE FOR (ALLTRIM(V_REGISTRO.QUIEN) + NPANTALLA =
                                ALLTRIM(PQUIEN) + WONTOP()) AND V_REGISTRO.ACT_INA = .T.
|       +-------IF !FOUND()
                                INSERT INTO V_REGISTRO(QUIEN, NPANTALLA, FECHAR, ACT_INA);
```

```
|       |       VALUES(PQUIEN,WONTOP( ), DATE(), .T.)
&&si no es encontrado lo registramos y actualizamos el archivo
|       |               TABLEUPDATE(.T.)
|       +-------ENDIF
|
|               SELECT V_IPCTFCM
|
|               THIS.NAVIGATOR.CMDPREV.ENABLED=.T.
|               THIS.NAVIGATOR.CMDNEXT.ENABLED=.T.
|               THIS.NAVIGATOR.CMDTOP.ENABLED=.T.
|               THIS.NAVIGATOR.CMDEND.ENABLED=.T.
|               THIS.NAVIGATOR.REFRESH()
+-------ENDPROC
+-------PROCEDURE UNLOAD
&&Seleccionamos el archivo de fuentes complementarias
&&le quitamos el filtro
|               SELECT V_IPCTFCM
|               SET FILTER TO
&&Retornamos el valor del codigo de la nueva fuente complementaria
|               RETURN THISFORM.CODOBS
|
+-------ENDPROC
+-------PROCEDURE DESTROY
&&Al salir del formulario actualizamos el archivo de registro para saber
&&que este usuario ya salio de este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|               REQUERY('v_registro')
|               SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para indicar
&&que la cerro en este momento
                                LOCATE FOR
                                ALLTRIM(V_REGISTRO.QUIEN)+ALLTRIM(V_REGISTRO.NPANTALLA) =
                                ALLTRIM(PQUIEN)+WONTOP( )
|       +-------IF FOUND()
|       |               REPLACE V_REGISTRO.ACT_INA WITH .F.
&&Actualizamos la informacion y tambien el archivo
|       |               TABLEUPDATE(.T.)
|       +-------ENDIF
+-------ENDPROC
```

5.8.2 Boton Seleccionar

```
+-------PROCEDURE CLICK
|       +-------IF THISFORM.NAVIGATOR.EDITMODE THEN
|       |               THISFORM.NAVIGATOR.CMDADD.CLICK
|       |               SALIR=.T.
```

```
|       +-------ENDIF
|       +-------IF SALIR
|       |             THISFORM.CODOBS=THISFORM.NOFTECOMPLEMENTARIA.VALUE
|       |             THISFORM.RELEASE
|       +-------ENDIF
+-------ENDPROC
```

5.8.3 Boton Cerrar

```
+-------PROCEDURE CLICK
|             THISFORM.NAVIGATOR.CMDEXIT.CLICK
+-------ENDPROC

+-------PROCEDURE CMDEXIT.CLICK
|       +-------IF THIS.PARENT.EDITMODE
|       |             THIS.PARENT.CMDEDIT.CLICK
|       +-------ENDIF
|             THISFORM.CODOBS=-1
|             THISFORM.RELEASE
+-------ENDPROC

+-------PROCEDURE CMDADD.CLICK
&&Filtramos el archivo de fuentes complementarias segun el codigo de fuente original
&&que viene de captura
                              +-------IF THIS.PARENT.EDITMODE AND
                              !EMPTY(ALLTRIM(THISFORM.NOMFTE1.VALUE)) AND
                              !EMPTY(ALLTRIM(THISFORM.DIREFTE1.VALUE))
|       |         DODEFAULT()
&&Almacenamos la información en la vista
                              REPLACE V_IPCTFCM.CODFCOM WITH
                              THISFORM.NOFTECOMPLEMENTARIA.VALUE,;
|       |                   V_IPCTFCM.CODFTE WITH THISFORM.CODFTE1.VALUE,;
|       |                   V_IPCTFCM.NOMFTE WITH THISFORM.NOMFTE1.VALUE,;
|       |                   V_IPCTFCM.DIREFTE WITH THISFORM.DIREFTE1.VALUE,;
|       |                   V_IPCTFCM.TELFTE WITH THISFORM.TELFTE1.VALUE
&&Confirmamos la actualizacion de la informacion ingresada
|       |             TABLEUPDATE(.T.)
&&Si no esta en modo edicion tomamos el codigode fuente ( encaso de haberse digitado)
|       +-------ELSE
|       |             NOFTE=THISFORM.CCODFTE
&&Reconsultamos el contenido de la vista
|       |             SELECT V_IPCTFCM
&&Inactivamos el filtro temporalmente para poder tener acceso a la informacion
&&marcada para eliminar
|       |             SET FILTER TO
|       |
|       |       +-------IF !THIS.PARENT.EDITMODE
```

```
&&Nos vamos al principio del archivo buscamos el codigo de funte complementaria introducido
|     |     |           GO TOP
|     |     |     +-------FOR I = 1 TO NOMAXFTESCOMP
LOCATE ALL FOR (V_IPCTFCM.CODFCOM = I) AND
(V_IPCTFCM.CODFTE=THISFORM.CCODFTE)
|     |     |     |     +-------IF FOUND()
&&Si se encontrado
|     |     |     |     |     +-------IF V_IPCTFCM.MARCA = .T.
&&Si esta marcado para eliminarse reutilizamos esa información cambiando el valor del campo marca
REPLACE V_IPCTFCM.MARCA WITH .F.
&&Reactivamos el el filtro para accesar la información que acabamos de habilitar
SET FILTER TO V_IPCTFCM.MARCA = .F.
&&Ubicamos la fuente complementaria y limpiamos la informacion que tenia anteriormente
&&para agregarle la nueva ingresada
LOCATE FOR (V_IPCTFCM.CODFCOM = I) AND (V_IPCTFCM.CODFTE=THISFORM.CCODFTE)
THIS.PARENT.CMDEDIT.CLICK
REPLACE V_IPCTFCM.NOMFTE WITH ",;
V_IPCTFCM.DIREFTE WITH ",;
|     |     |     |     |     |         V_IPCTFCM.TELFTE WITH "
|     |     |     |     |     |
|     |     |     |     |     |         EXIT
|     |     |     |     |     +-------ENDIF
|     |     |     |     +-------ELSE
|     |     |     |     |           SELECT V_IPCTFCM
|     |     |     |     |           DODEFAULT()
&&Almacenamos informacion en caso de no hayarse esta fuente complementaria
REPLACE V_IPCTFCM.CODFCOM WITH I, V_IPCTFCM.CODFTE WITH THISFORM.CCODFTE
|     |     |     |     |           EXIT
|     |     |     |     +-------ENDIF
|     |     |     +-------ENDFOR
|     |     |           THISFORM.CODFTE1.REFRESH
|     |     |
|     |     +-------ELSE
&&En caso de que no se diligencia la información completamente lo solicitamos
                                  +-------IF !EMPTY(ALLTRIM(THISFORM.NOMFTE1.VALUE)) AND
                                  !EMPTY(ALLTRIM(THISFORM.DIREFTE1.VALUE))
|     |     |     |           DODEFAULT()
|     |     |     +-------ELSE
MESSAGEBOX('Olvido diligenciar el nombre y/o la dirección de la fuente. ',0,'Precaución')
|     |     |     |           SALIR=.F.
|     |     |     +-------ENDIF
|     |     +-------ENDIF
|     +-------ENDIF
&&Seleccionamos la vista y reactivamos el filtro para no mostrar la informacion marcada para eliminar
|           SELECT V_IPCTFCM
|           SET FILTER TO (V_IPCTFCM.MARCA = .F.)
|           THISFORM.REFRESH
|
```

```
+-------ENDPROC

+-------PROCEDURE CMDDELETE.CLICK
&&Seleccionamos la vista y colocamos el campo marca para indicar que lo marcamos para ser
eliminado
|           SELE V_IPCTFCM
|           REPLACE V_IPCTFCM.MARCA WITH .T.
&&Confirmamos la actualizacion de la inmformacion y filtramos para que no sea mostrada la
&&que esta marcada para eliminar
|           TABLEUPDATE(.T.)
|           SET FILTER TO V_IPCTFCM.MARCA = .F.
|
|           DODEFAULT()
+-------ENDPROC
```

5.9 FORMULARIO PARA CALCULAR SERVICIOS PUBLICOS (Fserpub.SCX)

5.9.1 Vistas

v_ipctctv
v_variames
vfuentes
v_estrato
v_estratoant
v_rango
v_rangoant
v_rangoant

```
+-------PROCEDURE CALQOPOND
&&calcula el promedio ponderado con el dato No. de usuarios
|           SUM USUARIOS FOR CODGRI='1' TO QOGRI1
|           SUM USUARIOS FOR CODGRI='2' TO QOGRI2
|           SUM USUARIOS FOR CODGRI='3' TO QOGRI3
|
|     +-------IF QOGRI1=0
|     |           REPLA ALL QO_POND WITH 100 FOR CODGRI='1'
|     +-------ELSE
|     |           REPLA ALL QO_POND WITH (USUARIOS/QOGRI1*100) FOR CODGRI='1'
|     +-------ENDIF
|
|     +-------IF QOGRI2=0
|     |           REPLA ALL QO_POND WITH 100 FOR CODGRI='2'
|     +-------ELSE
|     |           REPLA ALL QO_POND WITH (USUARIOS/QOGRI2*100) FOR CODGRI='2'
|     +-------ENDIF
```

```
|
|       +-------IF QOGRI3=0
|       |             REPLA ALL QO_POND WITH 100 FOR CODGRI='3'
|       +-------ELSE
|       |             REPLA ALL QO_POND WITH (USUARIOS/QOGRI3*100) FOR CODGRI='3'
|       +-------ENDIF
|             SELE &TABLA
|             TABLEUPDATE(.T.)
|
+-------ENDPROC
+-------PROCEDURE CALVALCONSUMO
|             LOCAL VCTOS, REPLL, VESTR, TABEST, TABRAN, TABTAR, DIFERIMP, VTOTVCO &&,
fuente, mvt1, mvt2, mvt3
|
|             VTOTVCO=0
|
|       +-------IF ANTERIOR
|       |             TABEST='v_estratoant'
|       |             TABRAN='v_rangoant'
|       |             TABTAR='v_tarifaant'
|       +-------ELSE
|       |             TABEST='v_estrato'
|       |             TABRAN='v_rango'
|       |             TABTAR='v_tarifa'
|       +-------ENDIF
|
|             VCTOS=0
|             REPLL=0
|             VLR1=0
|             VLR2=0
|             VLR3=0
|
|             SELE &TABLA
|             SUM VLR_TARIFA TO INFO
|
|       +-------IF INFO!=0
|       |
|       |             SELE MAX(CODTARIFA)AS MAX ;
|       |                   FROM &TABRAN ;
|       |                   WHERE MES=CMES AND ANO=CANO AND CODFTE=MFUENTE AND
CODART=MARTICULO ;
|       |                   INTO CURSOR MAX
|       |             VCTOS = VAL(MAX.MAX)
|       |
&&Coloca como rango final el máximo valor de la variable
|       |             SELE &TABRAN
|       |             *set order to pkrango   &&ano+mes+codart+codfte+codtarifa
```

```
|     |          REPLA RAN_FIN WITH 999 FOR VAL(CODTARIFA)=VCTOS AND (RAN_FIN<>0
AND RAN_FIN<>999) AND MES=CMES AND ANO=CANO ;
|     |                AND CODFTE=MFUENTE AND CODART=MARTICULO
|     |
&&Para el calculo del consum_tar (consumo por tarifa)
|     |          REPLA RANI WITH RAN_INI-1 FOR MES=CMES AND ANO=CANO AND
CODFTE=MFUENTE AND CODART=MARTICULO ;
|     |                AND RAN_INI!= 0
|     |
|     |          REPLA RANI WITH 0 FOR MES=CMES AND ANO=CANO AND CODFTE=MFUENTE
AND CODART=MARTICULO ;
|     |                AND RAN_INI=0
|     |
|     |          SELE &TABRAN
|     |          TABLEUPDATE(.T.)
|     |
|     |          ******************************************************************************************
|     |          SELECT &TABTAR
&&Coloca como rango final el máximo valor de la variable
|     |          REPLA ALL RAN_FIN WITH 999 FOR VAL(CODTARIFA)=VCTOS AND
(RAN_FIN<>0 AND RAN_FIN<>999) AND MES=CMES AND ANO=CANO ;
|     |                AND CODFTE=MFUENTE AND CODART=MARTICULO AND
BETWEEN(CODESTRATO,ESP1,ESP2)
|     |
&&Para el calculo del consum_tar (consumo por tarifa)
|     |          REPLA ALL RANI WITH RAN_INI-1 FOR MES=CMES AND ANO=CANO AND
CODFTE=MFUENTE AND CODART=MARTICULO ;
|     |                AND RAN_INI!= 0 AND BETWEEN(CODESTRATO,ESP1,ESP2)
|     |
|     |          REPLA ALL RANI WITH 0 FOR MES=CMES AND ANO=CANO AND
CODFTE=MFUENTE AND CODART=MARTICULO ;
|     |                AND RAN_INI=0 AND BETWEEN(CODESTRATO,ESP1,ESP2)
|     |
|     |          SELE &TABTAR
|     |          TABLEUPDATE(.T.)
|     |
*******************************************************************************************
|     |
|     |          SELECT &TABTAR
|     |          SET ORDER TO TAG PKTARIFA  &&ano+mes+codart+codfte+codestrato+codtarifa
(candidato)
|     |          SET RELATION TO ANO+MES+CODART+CODFTE+CODESTRATO INTO &TABEST
ADDITIVE
|     |          SET RELATION TO ANO+MES+CODART+CODFTE+CODTARIFA INTO &TABRAN
ADDITIVE
|     |
|     |          SET EXACT OFF
|     |          *ano+mes+codart+codfte+codestrato+codtarifa
```

```
|     |           SEEK CANO+CMES+MARTICULO+MFUENTE
&&Distribución del consumo x para las tarifas de acuerdo a los rangos
&&Teniendo en cuenta el a que estratos se aplican los rango
|     |     +-------DO WHILE !EOF() AND ANO=CANO AND MES=CMES AND
CODART=MARTICULO AND CODFTE=MFUENTE
|     |     |           VEST=CODESTRATO
|     |     |           CAMPO=TABEST+'.consumo'
|     |     |           VRESTOCONS=&CAMPO  &&resto del consumo para repartir (inicialmente es
todo el consumo de este estrato para repartir)
|     |     |     +-------DO WHILE !EOF() AND ANO=CANO AND MES=CMES AND
CODART=MARTICULO AND CODFTE=MFUENTE AND CODESTRATO=VEST
|     |     |     |           CAMPO1=TABTAR+'.ran_fin'
|     |     |     |           CAMPO2=TABTAR+'.rani'
|     |     |     |     +-------IF VRESTOCONS-(&CAMPO1-&CAMPO2)>0
|     |     |     |     |           M.CONSUM_TAR=&CAMPO1-&CAMPO2
|     |     |     |     |           VRESTOCONS=VRESTOCONS-(&CAMPO1-&CAMPO2)   &&PARA
REPARTIR
|     |     |     |     +-------ELSE
|     |     |     |     |           M.CONSUM_TAR=VRESTOCONS
|     |     |     |     |           VRESTOCONS=0
|     |     |     |     +-------ENDIF
|     |     |     |           REPLACE CONSUM_TAR WITH m.CONSUM_TAR
|     |     |     |           SKIP
|     |     |     +-------ENDDO &&MIENTRAS SEA EL MISMO ESTRATO
|     |     +-------ENDDO
|     |
|     |           SELECT &TABTAR
|     |
&&para los estratos que no aplican los rangos  (p.ej energía solo aplica a los 3 1ros estratos)
|     |           REPLA ALL VLR_TARIFA WITH 0 FOR ANO=CANO AND MES=CMES AND
CODFTE=MFUENTE ;
|     |                 AND CODART=MARTICULO AND !BETWEEN(CODESTRATO,ESP1,ESP2)
AND CODTARIFA!='01'
|     |
|     |           CAMPO=TABEST+'.consumo'
|     |           REPLA ALL CONSUM_TAR WITH &CAMPO FOR ANO=CANO AND MES=CMES
AND ;
|     |                 CODFTE=MFUENTE AND CODART=MARTICULO AND
!BETWEEN(CODESTRATO,ESP1,ESP2)
|     |           ** fin distribucion
|     |
|     |           ******* CALCULO
|     |           REPLA ALL VLR_CONSUM WITH VLR_TARIFA*CONSUM_TAR FOR ANO=CANO
AND MES=CMES ;
|     |                 AND CODFTE=MFUENTE AND CODART=MARTICULO
|     |           SELE &TABTAR
|     |           TABLEUPDATE(.T.)
|     |
```

```
|     |          SET RELATION TO
|     |          SET FILTER TO MARCA=.F.
|     |          GO TOP
&&Suma por estrato el campo vlr_consum (vlr_tarifa*consum_tar*iva)  **Opcional el iva
|     |          TABLA='&locdir'+'\temp.dbf'
|     |          TOTAL TO &TABLA ON ANO+MES+CODFTE+CODART+CODESTRATO FIELDS
VLR_CONSUM ;
|     |                FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO
|     |          USE &TABLA IN 0
|     |
|     |          SELECT A.ANO, A.MES, A.CODART, A.CODFTE, A.VLR_TARIFA, A.VLR_CONSUM,
A.VLR_PONDE, A.CODESTRATO, B.CODGRI, B.USUARIOS,;
|     |                B.QO_POND, B.CARGOFIJO, B.IMPUESTO, B.CONSUMO, B.FPDEPOR,
ROUND(A.VLR_CONSUM+B.CARGOFIJO,2) AS TOTVLRCONS,;
|     |                000 AS IMPULRESTO;
|     |                FROM TEMP A, &TABEST B;
|     |                WHERE A.ANO=CANO AND A.MES=CMES AND A.CODART=MARTICULO
AND A.CODFTE=MFUENTE AND ;
|     |                A.ANO=B.ANO AND A.MES=B.MES AND A.CODFTE=B.CODFTE AND ;
|     |                A.CODART=B.CODART AND A.CODESTRATO= B.CODESTRATO;
|     |                INTO TABLE TEMP1
|     |
|     |          SELECT TEMP
|     |          USE  &&lo cierra para ejecutarlo de nuevo si es necesario
|     |
&&Agrega Cargo fijo e impuestos si existieren (adiciona el iva al cargo fijo)
|     |          SELECT TEMP1
|     |     +-------IF !EMPTY(IVA) AND MARTICULO="7320101" &&se aplica a telefono local y aqui no
redondean a 2 en total vlr_consumo
|     |     |
|     |     |          SELECT TEMP1
|     |     |          GO TOP
|     |     |     +-------SCAN
|     |     |     |          DIFERIMP=CONSUMO-V_LIMIMP
|     |     |     |     +-------IF DIFERIMP>0
|     |     |     |     |          REPLACE IMPULRESTO WITH DIFERIMP
|     |     |     |     +-------ENDIF
|     |     |     |          VEST=CODESTRATO
|     |     |     |          VFIELD='vtar'+VEST
|     |     |     |          &VFIELD=VLR_TARIFA
|     |     |     +-------ENDSCAN
|     |     |
|     |     |          SELECT TEMP1
|     |     |     +-------DO CASE
|     |     |     +-------CASE INLIST(SUBSTR(MFUENTE,1,2),"11","52","66") &&para Bogotá, Pasto,
Pereira
```

```
|     |     |     |           REPLA ALL IMPUESTO WITH (CARGOFIJO+((CONSUMO-
V_LIMIMP)*VTAR4))*IVA FOR (CONSUMO-V_LIMIMP)>0 AND;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND BETWEEN(CODESTRATO,'1','2')
|     |     |     |
|     |     |     |           REPLA ALL IMPUESTO WITH (CARGOFIJO)*IVA FOR (CONSUMO-
V_LIMIMP)<=0 AND;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO  AND BETWEEN(CODESTRATO,'1','2')
|     |     |     |
|     |     |     |           REPLA ALL IMPUESTO WITH
(CARGOFIJO+(CONSUMO*VLR_TARIFA))*IVA FOR;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND CODESTRATO='3'
|     |     |     |
|     |     |     |           REPLA ALL IMPUESTO WITH (TOTVLRCONS)*IVA FOR;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND BETWEEN(CODESTRATO,'4','6')
|     |     |     |
|     |     |     |           *** V A L O R   P O N D E R A D O
|     |     |     |           REPLA ALL VLR_PONDE WITH
((TOTVLRCONS+IMPUESTO+FPDEPOR)*QO_POND)/100 ;
|     |     |     |                 FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO
|     |     |     |
|     |     |     |
|     |     |     +-------CASE SUBSTR(MFUENTE,1,2)="50"&&para VILLAVICENCIO
|     |     |     |           GO TOP
|     |     |     |     +-------SCAN FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE
AND CODART=MARTICULO
|     |     |     |     |     +-------DO CASE
|     |     |     |     |     +-------CASE  BETWEEN(CODESTRATO,'1','3')
|     |     |     |     |     |           REPLACE TOTVLRCONS WITH
(V_LIMIMP*VLR_TARIFA+CARGOFIJO)
|     |     |     |     |     |           REPLACE IMPUESTO WITH (IMPULRESTO*VTAR4)*(1+IVA)
|     |     |     |     |     |           REPLACE VLR_PONDE WITH
((TOTVLRCONS+IMPUESTO+FPDEPOR)*QO_POND)/100
|     |     |     |     |     |
|     |     |     |     |     +-------CASE  BETWEEN(CODESTRATO,'4','6')
|     |     |     |     |     |           REPLACE TOTVLRCONS WITH
(CONSUMO*VLR_TARIFA+CARGOFIJO)
|     |     |     |     |     |           REPLA IMPUESTO WITH (TOTVLRCONS)*(1+IVA)
|     |     |     |     |     |           REPLACE VLR_PONDE WITH
((IMPUESTO+FPDEPOR)*QO_POND)/100
|     |     |     |     |     |
|     |     |     |     |     +-------ENDCASE
|     |     |     |     +-------ENDSCAN
|     |     |     |
```

```
|     |     |     |
|     |     |     +-------CASE SUBSTR(MFUENTE,1,2)="23"&&para MONTERIA
|     |     |     |           GO TOP
|     |     |     |     +-------SCAN FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE
AND CODART=MARTICULO
|     |     |     |     |     +-------DO CASE
|     |     |     |     |     +-------CASE  BETWEEN(CODESTRATO,'1','2')
|     |     |     |     |     |           REPLACE TOTVLRCONS WITH (V_LIMIMP*VLR_TARIFA)
|     |     |     |     |     |           REPLACE IMPUESTO WITH (IMPULRESTO*VTAR4)*(1+IVA)
|     |     |     |     |     |           REPLACE VLR_PONDE WITH
((TOTVLRCONS+IMPUESTO+FPDEPOR)*QO_POND)/100
|     |     |     |     |     +-------CASE  CODESTRATO='3'
|     |     |     |     |     |           REPLACE TOTVLRCONS WITH
(CONSUMO*VLR_TARIFA+CARGOFIJO)
|     |     |     |     |     |           REPLA IMPUESTO WITH (IMPULRESTO*VTAR4)*(1+IVA)
|     |     |     |     |     |           REPLACE VLR_PONDE WITH
((TOTVLRCONS+IMPUESTO+FPDEPOR)*QO_POND)/100
|     |     |     |     |     +-------CASE  BETWEEN(CODESTRATO,'4','6')
|     |     |     |     |     |           REPLACE TOTVLRCONS WITH
(CONSUMO*VLR_TARIFA+CARGOFIJO)
|     |     |     |     |     |           REPLA IMPUESTO WITH (TOTVLRCONS)*(1+IVA)
|     |     |     |     |     |           REPLACE VLR_PONDE WITH
((IMPUESTO+FPDEPOR)*QO_POND)/100
|     |     |     |     |     |
|     |     |     |     |     |
|     |     |     |     |     +-------ENDCASE
|     |     |     |     +-------ENDSCAN
|     |     |     |
|     |     |     |
|     |     |     +-------CASE SUBSTR(MFUENTE,1,2)="13"&&para CARTAGENA
|     |     |     |           GO TOP
|     |     |     |     +-------SCAN FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE
AND CODART=MARTICULO
|     |     |     |     |     +-------DO CASE
|     |     |     |     |     +-------CASE  BETWEEN(CODESTRATO,'1','2')
|     |     |     |     |     |           REPLACE TOTVLRCONS WITH
(V_LIMIMP*VLR_TARIFA)+(IMPULRESTO*VTAR3)
|     |     |     |     |     |           REPLACE IMPUESTO WITH
((IMPULRESTO*VLR_TARIFA)+CARGOFIJO)*(1+IVA)
|     |     |     |     |     |           REPLACE VLR_PONDE WITH
((TOTVLRCONS+IMPUESTO+FPDEPOR)*QO_POND)/100
|     |     |     |     |     +-------CASE  BETWEEN(CODESTRATO,'3','6')
|     |     |     |     |     |           REPLA IMPUESTO WITH (TOTVLRCONS)*(1+IVA)
|     |     |     |     |     |           REPLACE VLR_PONDE WITH
((IMPUESTO+FPDEPOR)*QO_POND)/100
|     |     |     |     |     +-------ENDCASE
|     |     |     |     +-------ENDSCAN
|     |     |     |
```

```
|     |     |     |
|     |     |     +-------CASE SUBSTR(MFUENTE,1,2)="41"&&para NEIVA
|     |     |     |           REPLA ALL IMPUESTO WITH
((IMPULRESTO*VLR_TARIFA)+CARGOFIJO)*IVA FOR ;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND BETWEEN(CODESTRATO,'1','2')
|     |     |     |
|     |     |     |           REPLA ALL IMPUESTO WITH (TOTVLRCONS)*IVA FOR;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND BETWEEN(CODESTRATO,'3','6')
|     |     |     |
|     |     |     |           *** V A L O R   P O N D E R A D O
|     |     |     |           REPLA ALL VLR_PONDE WITH
((TOTVLRCONS+IMPUESTO+FPDEPOR)*QO_POND)/100 ;
|     |     |     |                 FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO
|     |     |     |
|     |     |     |
|     |     |     +-------CASE SUBSTR(MFUENTE,1,2)="54" &&para CUCUTA

|     |     |     |           REPLA ALL IMPUESTO WITH
((IMPULRESTO*VLR_TARIFA)+CARGOFIJO)*IVA FOR ;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND CODGRI="1"
|     |     |     |
|     |     |     |           REPLA ALL IMPUESTO WITH (TOTVLRCONS)*IVA FOR;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND BETWEEN(CODGRI,'2','3')
|     |     |     |
|     |     |     |           *** V A L O R   P O N D E R A D O
|     |     |     |           REPLA ALL VLR_PONDE WITH
((TOTVLRCONS+IMPUESTO+FPDEPOR)*QO_POND)/100 ;
|     |     |     |                 FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO
|     |     |     |
|     |     |     |
|     |     |     +-------CASE  SUBSTR(MFUENTE,1,2)="08" &&para BQUILLA
|     |     |     |           REPLA ALL TOTVLRCONS WITH (V_LIMIMP*VLR_TARIFA) FOR
(CONSUMO-V_LIMIMP)>0 AND;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND BETWEEN(CODESTRATO,'1','2')
|     |     |     |           REPLA ALL TOTVLRCONS WITH
(CONSUMO*VLR_TARIFA)+CARGOFIJO FOR (CONSUMO-V_LIMIMP)>0 AND;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND BETWEEN(CODESTRATO,'3','6')
|     |     |     |
|     |     |     |           REPLA ALL TOTVLRCONS WITH
CARGOFIJO+(CONSUMO*VLR_TARIFA) FOR (CONSUMO-V_LIMIMP)<=0 AND;
```

```
|     |     |     |                ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO
|     |     |     |
|     |     |     |
|     |     |     |          REPLA ALL IMPUESTO WITH ((((CONSUMO-
V_LIMIMP)*VLR_TARIFA)+CARGOFIJO)*(1+IVA))+TOTVLRCONS FOR (CONSUMO-V_LIMIMP)>0
AND;
|     |     |     |                ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND BETWEEN(CODESTRATO,'1','2')
|     |     |     |
|     |     |     |
|     |     |     |          REPLA ALL IMPUESTO WITH
(((CONSUMO*VLR_TARIFA)+CARGOFIJO)*(1+IVA))+4.6 FOR (CONSUMO-V_LIMIMP)>0 AND;
|     |     |     |                ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND BETWEEN(CODESTRATO,'3','6')
|     |     |     |
|     |     |     |          *** V A L O R   P O N D E R A D O
|     |     |     |          REPLA ALL VLR_PONDE WITH ((IMPUESTO+FPDEPOR)*QO_POND)/100
;
|     |     |     |                FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO
|     |     |     |
|     |     |     |
|     |     |     +-------CASE  SUBSTR(MFUENTE,1,2)="68" &&para BMANGA
|     |     |     |          REPLA ALL TOTVLRCONS WITH
CARGOFIJO+(V_LIMIMP*VLR_TARIFA)+((CONSUMO-V_LIMIMP)*VTAR4) FOR (CONSUMO-
V_LIMIMP)>0 AND;
|     |     |     |                ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND CODGRI='1'
|     |     |     |
|     |     |     |          REPLA ALL TOTVLRCONS WITH
CARGOFIJO+(CONSUMO*VLR_TARIFA) FOR (CONSUMO-V_LIMIMP)<=0 AND;
|     |     |     |                ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO  AND CODGRI='1'
|     |     |     |
*caso especial de ciertos impulsos constantes a los que se les aplica el IVA
|     |     |     |          REPLA ALL IMPUESTO WITH (((CONSUMO-
V_LIMIMP)*VTAR4)+CARGOFIJO)*IVA FOR;
|     |     |     |                ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND CODGRI='1' AND CODESTRATO="1"
|     |     |     |          REPLA ALL IMPUESTO WITH (((CONSUMO-
V_LIMIMP)*VTAR4)+CARGOFIJO)*IVA FOR;
|     |     |     |                ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND CODGRI='1' AND CODESTRATO="2"
|     |     |     |          REPLA ALL IMPUESTO WITH (((CONSUMO-
V_LIMIMP)*VTAR4)+CARGOFIJO)*IVA FOR;
|     |     |     |                ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND CODGRI='1' AND CODESTRATO="3
```

```
"**fin de caso especiales de númeo x de impulsos a los que se les aplica el iva
|     |     |     |
|     |     |     |           REPLA ALL IMPUESTO WITH (TOTVLRCONS)*IVA FOR;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND BETWEEN(CODGRI,'2','3')
|     |     |     |
*** V A L O R   P O N D E R A D O
|     |     |     |           REPLA ALL VLR_PONDE WITH
((TOTVLRCONS+IMPUESTO+FPDEPOR)*QO_POND)/100 ;
|     |     |     |                 FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO
|     |     |     |
|     |     |     +-------CASE SUBSTR(MFUENTE,1,2)="17" &&para Manizales
|     |     |     |           REPLA ALL TOTVLRCONS WITH
CARGOFIJO+(V_LIMIMP*VLR_TARIFA)+((CONSUMO-V_LIMIMP)*VTAR4) FOR (CONSUMO-
V_LIMIMP)>0 AND;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND CODGRI='1'
|     |     |     |
|     |     |     |           REPLA ALL TOTVLRCONS WITH
CARGOFIJO+(CONSUMO*VLR_TARIFA) FOR (CONSUMO-V_LIMIMP)<=0 AND;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO  AND CODGRI='1'
|     |     |     |
|     |     |     |           REPLA ALL IMPUESTO WITH (((CONSUMO-
V_LIMIMP)*VTAR4)+CARGOFIJO)*IVA FOR;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND CODGRI='1'
|     |     |     |
|     |     |     |           REPLA ALL IMPUESTO WITH (TOTVLRCONS)*IVA FOR;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND BETWEEN(CODGRI,'2','3')
|     |     |     |
|     |     |     |           *** V A L O R   P O N D E R A D O
|     |     |     |
|     |     |     |           REPLA ALL VLR_PONDE WITH
((TOTVLRCONS+IMPUESTO+FPDEPOR)*QO_POND)/100 ;
|     |     |     |                 FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO
|     |     |     |
|     |     |     |
|     |     |     +-------CASE SUBSTR(MFUENTE,1,2)="05" &&para Medellín,
|     |     |     |           REPLA ALL IMPUESTO WITH (CARGOFIJO+((CONSUMO-
V_LIMIMP)*VTAR4))*(1+IVA) FOR (CONSUMO-V_LIMIMP)>0 AND;
|     |     |     |                 ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND BETWEEN(CODESTRATO,'1','3')
|     |     |     |
```

```
|     |     |     |          REPLA ALL IMPUESTO WITH (CARGOFIJO)*IVA FOR (CONSUMO-
V_LIMIMP)<=0 AND;
|     |     |     |               ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO  AND BETWEEN(CODESTRATO,'1','3')
|     |     |     |
|     |     |     |          REPLA ALL IMPUESTO WITH (TOTVLRCONS)*IVA FOR;
|     |     |     |               ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND BETWEEN(CODESTRATO,'4','6')
|     |     |     |
|     |     |     |          *** V A L O R   P O N D E R A D O para Medellín
|     |     |     |          REPLA ALL VLR_PONDE WITH
((VLR_CONSUM+IMPUESTO+FPDEPOR)*QO_POND)/100 ;
|     |     |     |               FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND CODGRI="1"
|     |     |     |          REPLA ALL VLR_PONDE WITH
((TOTVLRCONS+IMPUESTO+FPDEPOR)*QO_POND)/100 ;
|     |     |     |               FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND CODGRI<>"1"
|     |     |     |
|     |     |     |
|     |     |     +-------CASE SUBSTR(ALLTRIM(MFUENTE),1,2)="76" &&para Cali
|     |     |     |          REPLA ALL IMPUESTO WITH (CARGOFIJO+((CONSUMO-
V_LIMIMP)*VLR_TARIFA))*IVA FOR (CONSUMO-V_LIMIMP)>0 AND;
|     |     |     |               ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND BETWEEN(CODESTRATO,'1','3')
|     |     |     |
|     |     |     |          REPLA ALL IMPUESTO WITH (CARGOFIJO)*IVA FOR (CONSUMO-
V_LIMIMP)<=0 AND;
|     |     |     |               ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO  AND BETWEEN(CODESTRATO,'1','3')
|     |     |     |
|     |     |     |          REPLA ALL IMPUESTO WITH (TOTVLRCONS)*IVA FOR;
|     |     |     |               ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO AND BETWEEN(CODESTRATO,'4','6')
|     |     |     |
|     |     |     |          *** V A L O R   P O N D E R A D O
|     |     |     |          REPLA ALL VLR_PONDE WITH
((TOTVLRCONS+IMPUESTO+FPDEPOR)*QO_POND)/100 ;
|     |     |     |               FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO
|     |     |     +-------ENDCASE
|     |     |
|     |     +-------ELSE &&sin iva y diferentes de telefono
|     |     |          ** el impuesto puede ser para el caso de alumbrado público, colocar este valor en
esta casilla para los estrato que se aplique
|     |     |
|     |     |          SELECT TEMP1
|     |     |          GO TOP
```

```
|     |     |
|     |     |
|     |     |     +-------IF MARTICULO="2220101" AND SUBSTR(ALLTRIM(MFUENTE),1,2)="50"
&&para VILLA &&CONTRIBUCIOS FONDOS DE SOLIDARIDAD EN ACUEDUCTO
|     |     |     |     +-------SCAN FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE
AND CODART=MARTICULO
|     |     |     |     |     +-------DO CASE
|     |     |     |     |     +-------CASE BETWEEN(CODESTRATO,'1','4')
|     |     |     |     |     |           *ROUND(a.vlr_consum+b.cargofijo,2) AS totvlrCons,;
STORE totvlrCons*Qo_pond TO Vtotvco
|     |     |     |     |     |           VTOTVCO=TOTVLRCONS*QO_POND
|     |     |     |     |     |           REPLACE VLR_PONDE WITH VTOTVCO/100
|     |     |     |     |     |
|     |     |     |     |     +-------CASE BETWEEN(CODESTRATO,'5','6')
|     |     |     |     |     |           REPLACE TOTVLRCONS WITH
(CONSUMO*VLR_TARIFA+CONSUMO*IMPUESTO)+CARGOFIJO
|     |     |     |     |     |           REPLACE VLR_PONDE WITH (TOTVLRCONS*QO_POND)/100
|     |     |     |     |     +-------ENDCASE
|     |     |     |     +-------ENDSCAN
|     |     |     |
|     |     |     |
|     |     |     +-------ELSE
|     |     |     |
|     |     |     |     +-------IF  MARTICULO="2220101" AND
(SUBSTR(ALLTRIM(MFUENTE),1,2)="23" OR SUBSTR(ALLTRIM(MFUENTE),1,2)="68") &&PARA
ENERGIA EN MONTERIA y en BMANGA, donde Alumbrado es el 10% del vlrconsumo (posteror/se
parametrizara)
|     |     |     |     |           REPLA ALL IMPUESTO WITH TOTVLRCONS*0.10 FOR ANO=CANO
AND MES=CMES AND CODFTE=MFUENTE AND CODART=MARTICULO
|     |     |     |     |           REPLA ALL VLR_PONDE WITH
((TOTVLRCONS+IMPUESTO)*QO_POND)/100 FOR ANO=CANO AND MES=CMES AND
CODFTE=MFUENTE AND CODART=MARTICULO
|     |     |     |     +-------ELSE
&&OTRAS CIUDADES Y OTROS SERVICIOS EXEPTO ACUEDUCTO EN VILLAVICENCIO
|     |     |     |     |           REPLA ALL VLR_PONDE WITH
((TOTVLRCONS+IMPUESTO)*QO_POND)/100 FOR ANO=CANO AND MES=CMES AND
CODFTE=MFUENTE AND CODART=MARTICULO

|     |     |     |     +-------ENDIF
|     |     |     |
|     |     |     |
|     |     |     +-------ENDIF
|     |     |
|     |     |
|     |     +-------ENDIF
|     |
|     |           SELECT TEMP1
|     |           GO TOP
```

```
|       |
&&Suma el valor ponderado por grupo de ingreso
|       |       +-------SCAN FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO
|       |       |       +-------DO CASE
|       |       |       +-------CASE VAL(TEMP1.CODGRI)=1
|       |       |       |               VLR1=VLR1+VLR_PONDE
|       |       |       |
|       |       |       +-------CASE VAL(TEMP1.CODGRI)=2
|       |       |       |               VLR2=VLR2+VLR_PONDE
|       |       |       |
|       |       |       +-------CASE VAL(TEMP1.CODGRI)=3
|       |       |       |               VLR3=VLR3+VLR_PONDE
|       |       |       +-------ENDCASE
|       |       +-------ENDSCAN
|       |
|       |               ** reemplazar en tabla de estrato los valores de temp1
|       |               SELECT TEMP1
|       |               GO TOP
|       |       +-------SCAN FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO
|       |       |               VESTR=CODESTRATO
|       |       |               SELECT &TABEST
|       |       |       +-------IF SEEK(CANO+CMES+MARTICULO+MFUENTE+VESTR)
|       |       |       |               REPLACE IMPUESTO WITH TEMP1.IMPUESTO
|       |       |       +-------ENDIF
|       |       |               SELECT TEMP1
|       |       +-------ENDSCAN

|       |               SELECT &TABEST
|       |               TABLEUPDATE(.T.)
|       |               ************************************************************************
&&Graba la informacion en las tablas y visualiza resultados
|       |               THISFORM.GRABA
|       +-------ELSE
|       |               NANSWER=MESSAGEBOX(CMESSAGETEXT, NDIALOGTYPE, CMESSAGETITLE)
|       +-------ENDIF
|
+-------ENDPROC
+-------PROCEDURE NVOMES
|               LOCAL TABLA
&&Agrega nuevos registros a las tablas para la captura de la informacion mes actual Se ejecuta
cuando va a calcular el mes actual en caso que haya habido algun cambio en los datos del mes
anterior Actualiza tabla de estratos
|               TABLA='&locdir'+'\Estr'
|               SELE
ANO,MES,CODART,CODFTE,CODESTRATO,USUARIOS,CARGOFIJO,CONSUMO,CODGRI,IMPU
ESTO,FPDEPOR,QO_POND,MARCA;
```

```
|                FROM V_ESTRATOANT ;
|                INTO TABLE &TABLA ;
|                ORDER BY ANO,MES,CODART,CODFTE,CODESTRATO
|
|          SELECT V_ESTRATO
|          REQUERY('V_estrato')
|          SELECT V_ESTRATO
|          SET FILTER TO
|          GO TOP
|          SET ORDER TO TAG PKESTRATO &&ano+mes+codart+codfte+codestrato tag pkestrato
|
|          SELE ESTR
|     +-------SCAN
|     |          SCATTER MEMVAR
|     |          M.MES=MMES
|     |          M.ANO=MANO
|     |          M.QO_POND=0
|     |          SELE V_ESTRATO
|     |          SEEK(m.ANO+m.MES+m.CODART+m.CODFTE+m.CODESTRATO)
|     |     +-------IF !FOUND()
|     |     |          INSERT INTO V_ESTRATO FROM MEMVAR
|     |     +-------ELSE
|     |     |     +-------IF MARCA &&DELETED()
|     |     |     |          REPLACE MARCA WITH .F.
|     |     |     +-------ENDIF
|     |     |          REPLACE USUARIOS WITH m.USUARIOS, CONSUMO WITH m.CONSUMO
|     |     +-------ENDIF
|     |          SELE ESTR
|     +-------ENDSCAN
|          SELECT V_ESTRATO
|          SET FILTER TO MARCA =.F.
|          GO TOP
|
&&Actualiza tabla de rangos
|          TABLA='&locdir'+'\ran'
|          SELE * FROM V_RANGOANT ;
|                INTO TABLE &TABLA ;
|                ORDER BY ANO,MES,CODART,CODFTE,CODTARIFA
|
|          SELECT V_RANGO
|          REQUERY('V_RANGO')
|          SELECT V_RANGO
|          SET FILTER TO
|          GO TOP
|          SET ORDER TO TAG PKRANGO  &&ano+mes+codart+codfte+codtarifa tag pkrango
|          SELE RAN
|     +-------SCAN
|     |          SCATTER MEMVAR
```

```
|       |           M.MES=MMES
|       |           M.ANO=MANO
|       |           SELE V_RANGO
|       |           SEEK(m.ANO+m.MES+m.CODART+m.CODFTE+m.CODTARIFA)
|       |     +-------IF !FOUND()
|       |     |           INSERT INTO V_RANGO FROM MEMVAR
|       |     |           *else
|       |     |           * gather memvar
|       |     +-------ENDIF
|       |           SELE RAN
|       +-------ENDSCAN
|             SELECT V_RANGO
|             SET FILTER TO MARCA=.F.
|             GO TOP
|
&&Actualiza tabla de tarifas
|             TABLA='&locdir'+'\tar'
|             SELE
ANO,MES,CODART,CODFTE,CODESTRATO,CODTARIFA,VLR_TARIFA,RAN_INI,RAN_FIN,RANI,
CONSUM_TAR,VLR_CONSUM,VLR_PONDE,MARCA ;
|                   FROM V_TARIFAANT ;
|                   INTO TABLE &TABLA ;
|                   ORDER BY ANO,MES,CODART,CODFTE,CODESTRATO,CODTARIFA
|
|             SELECT V_TARIFA
|             REQUERY('V_tarifa')
|             SELECT V_TARIFA
|             SET FILTER TO
|             GO TOP
|             SET ORDER TO TAG PKTARIFA &&ano+mes+codart+codfte+codestrato+codtarifa tag
pktarifa
|
|             SELE TAR
|       +-------SCAN
|       |           SCATTER MEMVAR
|       |           M.MES=MMES
|       |           M.ANO=MANO
|       |           M.CONSUM_TAR=0
|       |           M.VLR_CONSUM=0
|       |           M.VLR_PONDE=0
|       |           SELECT V_TARIFA
|       |           SEEK(m.ANO+m.MES+m.CODART+m.CODFTE+m.CODESTRATO+m.CODTARIFA)
|       |     +-------IF !FOUND()
|       |     |           INSERT INTO V_TARIFA FROM MEMVAR
|       |     |           *else
|       |     |           * gather memvar
|       |     +-------ENDIF
|       |           SELE TAR
```

```
|       +-------ENDSCAN
|
|               SELECT V_TARIFA
|               SET FILTER TO MARCA=.F.
|               GO TOP
|
&&Consulta información mes actual
|               SELE V_TARIFA
|               TABLEUPDATE(.T.)
|               REQUERY('v_tarifa')
|               SET FILTER TO MARCA=.F.
|               GO TOP
|
|
|               SELE V_ESTRATO
|               TABLEUPDATE(.T.)
|               REQUERY('v_estrato')
|               SET FILTER TO MARCA=.F.
|               GO TOP
|
|
|               SELE V_RANGO
|               TABLEUPDATE(.T.)
|               REQUERY('v_rango')
|               SET FILTER TO MARCA=.F.
|               GO TOP
|
|
|               SELE DISTINCT * ;
|                     FROM V_TARIFA ;
|                     WHERE CODESTRATO+CODTARIFA NOT IN (SELE CODESTRATO+CODTARIFA
AS LLAVE FROM V_TARIFAANT);
|                     INTO CURSOR BORRANG
|
|               SELE BORRANG
|       +-------IF !EOF()
|       |       +-------SCAN
|       |       |               SCATTER MEMVAR
|       |       |               SELE V_TARIFA
|       |       |       +-------IF
SEEK(m.ANO+m.MES+m.CODART+m.CODFTE+m.CODESTRATO+m.CODTARIFA)
|       |       |       |       +-------IF !MARCA &&!DELETED()
|       |       |       |       |               REPLACE MARCA WITH .T. FOR ANO=m.ANO AND MES=m.MES
AND CODART=m.CODART AND CODESTRATO=m.CODESTRATO AND
CODTARIFA=m.CODTARIFA
|       |       |       |       |
|       |       |       |       +-------ENDIF
|       |       |       +-------ENDIF
```

```
|     |     |           SELE BORRANG
|     |     +-------ENDSCAN
|     |           SELE V_TARIFA
|     |           *pack
|     +-------ENDIF
|           SELE V_TARIFA
|           TABLEUPDATE(.T.)
|
|
|           SELE DISTINCT * ;
|                 FROM V_RANGO ;
|                 WHERE CODTARIFA NOT IN (SELE CODTARIFA AS LLAVE FROM V_RANGOANT);
|                 INTO CURSOR BORRANG
|
|           SELE BORRANG
|     +-------IF !EOF()
|     |     +-------SCAN
|     |     |           SCATTER MEMVAR
|     |     |           SELE V_RANGO
|     |     |     +-------IF SEEK(m.ANO+m.MES+m.CODART+m.CODFTE+m.CODTARIFA)
|     |     |     |     +-------IF !MARCA &&!DELETED()
|     |     |     |     |           REPLACE MARCA WITH .T. FOR ANO=m.ANO AND MES=m.MES
AND CODART=m.CODART AND CODTARIFA=m.CODTARIFA
|     |     |     |     +-------ENDIF
|     |     |     +-------ENDIF
|     |     |           SELE BORRANG
|     |     +-------ENDSCAN
|     |           SELE V_RANGO
|     |           *pack
|     +-------ENDIF
|
|           SELE V_RANGO
|           TABLEUPDATE(.T.)
|
|           SELE DISTINCT * ;
|                 FROM V_ESTRATO ;
|                 WHERE CODESTRATO NOT IN (SELE CODESTRATO AS LLAVE FROM
V_ESTRATOANT);
|                 INTO CURSOR BORRANG
|
|           SELE BORRANG
|     +-------IF !EOF()
|     |     +-------SCAN
|     |     |           SCATTER MEMVAR
|     |     |           SELE V_ESTRATO
|     |     |     +-------IF SEEK(m.ANO+m.MES+m.CODART+m.CODFTE+m.CODESTRATO)
|     |     |     |     +-------IF !MARCA &&!DELETED()
```

```
|       |       |       |       |               REPLACE MARCA WITH .T. FOR ANO=m.ANO AND MES=m.MES
AND CODART=m.CODART AND CODESTRATO=m.CODESTRATO
|       |       |       |       +-------ENDIF
|       |       |       +-------ENDIF
|       |       |               SELE BORRANG
|       |       +-------ENDSCAN
|       |               SELE V_ESTRATO
|       |               *pack
|       +-------ENDIF
|
|               SELE V_ESTRATO
|               TABLEUPDATE(.T.)
|
&&Consulta información mes actual
|               REQUERY('v_tarifa')
|               REQUERY('v_estrato')
|               REQUERY('v_rango')
|
|
|
+-------ENDPROC
+-------PROCEDURE VALIRANGOS
|               LOCAL ERROR,ERROR1,NROANT,ESTRATO,TARIFA,FIN
|               THISFORM.CORRECTO=.T.
|               FIN=0
|               ERROR=.F.
|               ERROR1=.F.
|               ERROR2=.F.
|               TARIFA=''
|
|               SELE &TABLA
|               TABLEUPDATE(.T.)
|
|               SELE &TABLA
|               GO BOTTOM
&&para tener el dato del rango final de la última rango-tarifa
|               REPLA RAN_FIN WITH 999
|
|               GO TOP
|       +-------SCAN FOR !MARCA &&!deleted()
|       |               SCATTER FIELDS RAN_INI MEMVAR
|       |       +-------IF RAN_INI=0 AND RECNO()=1
|       |       |               LOOP
|       |       +-------ENDIF
|       |
&&para tener el rango completo y calcular consum_tar en la tabla de tarifas
|       |       +-------IF (RAN_INI - RANI) != 1
|       |       |               REPLA RANI WITH (RAN_INI - 1)
```

```
|       |       +-------ENDIF
|       |
|       |       +-------IF RAN_INI=0  OR  RAN_FIN=0
|       |       |               ERROR=.T.
|       |       +-------ENDIF
|       |
|       |       +-------IF RAN_INI > RAN_FIN
|       |       |               ERROR1=.T.
|       |       +-------ENDIF
|       |
|       |               * controlar que el rango inicial sea continuación del rango anterior
|       |       +-------IF RECNO()!= 1
|       |       |               NROANT=VAL(CODTARIFA) - 1
|       |       |               NROANT=PADL(ALLTRIM(STR(NROANT)),2,'0')
|       |       |               SELE RAN_FIN FROM &TABLA WHERE CODTARIFA=NROANT INTO
CURSOR CUALANT
|       |       |               FIN=CUALANT.RAN_FIN
|       |       |       +-------IF m.RAN_INI != FIN+1
|       |       |       |               ERROR2=.T.
|       |       |       +-------ENDIF
|       |       +-------ENDIF
|       +-------ENDSCAN
|
|
|       +-------IF ERROR OR ERROR1 OR ERROR2
|       |       +-------IF ERROR
|       |       |               NANSWER=MESSAGEBOX(CMESSAGETEXT, NDIALOGTYPE,
CMESSAGETITLE)
|       |       +-------ENDIF
|       |
|       |       +-------IF ERROR1
|       |       |               NANSWER=MESSAGEBOX('El rango final es mayor que el inicial. Revise',
NDIALOGTYPE, CMESSAGETITLE)
|       |       +-------ENDIF
|       |
|       |       +-------IF ERROR2
|       |       |               NANSWER=MESSAGEBOX('Los rangos no son continuos. Revise',
NDIALOGTYPE, CMESSAGETITLE)
|       |       +-------ENDIF
|       |               THISFORM.CORRECTO=.F.
|       +-------ELSE
&&Revisa si se borro un rango de la vista v_rangoant, que ya existía
|       |               SELE DISTINCT CODTARIFA ;
|       |                       FROM V_TARIFAANT ;
|       |                       WHERE ALLTRIM(CODTARIFA) NOT IN (SELE ALLTRIM(CODTARIFA) AS
LLAVE FROM V_RANGOANT);
|       |                       INTO CURSOR BORRANG
|       |
```

```
|       |           SELE BORRANG
|       |       +-------IF !EOF()
|       |       |           SELE V_TARIFAANT
|       |       |           REPLACE ALL MARCA WITH .T. FOR CODTARIFA=BORRANG.CODTARIFA
|       |       |           SELE V_TARIFAANT
|       |       |           TABLEUPDATE(.T.)
|       |       |           *pack
|       |       +-------ENDIF
&&Revisa si se ingresaron rangos a la vista v_rangoant, que no existian
|       |           SELE DISTINCT CODTARIFA, RAN_INI, RAN_FIN, RANI;
|       |                FROM V_RANGOANT ;
|       |                WHERE ALLTRIM(CODTARIFA) NOT IN (SELE DISTINCT
ALLTRIM(CODTARIFA) AS LLAVE FROM V_TARIFAANT);
|       |                INTO CURSOR INSRANG
|       |
|       |           SELE INSRANG
|       |       +-------IF !EOF()
|       |       |           GO TOP
&&Revisa si se ingreso algun codigo de tarifa nuevo
|       |       |       +-------SCAN
|       |       |       |           TARIFA=INSRANG.CODTARIFA
|       |       |       |           RAINI=INSRANG.RAN_INI
|       |       |       |           RAFIN=INSRANG.RAN_FIN
|       |       |       |           RI=INSRANG.RANI
|       |       |       |           SELE V_TARIFAANT
|       |       |       |           LOCATE FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE
AND ;
|       |       |       |                CODART=MARTICULO AND CODTARIFA=TARIFA
|       |       |       |       +-------IF !FOUND()
&&crea los registros en la tabla de tarifas
|       |       |       |       |       +-------FOR ESTRATO=1 TO CTOSEST
|       |       |       |       |       |           INSERT INTO V_TARIFAANT
(ANO,MES,CODFTE,CODART,CODESTRATO,CODTARIFA,RAN_INI,RAN_FIN,VLR_TARIFA,CON
SUM_TAR,VLR_PONDE,RANI) ;
|       |       |       |       |       |               VALUES
(CANO,CMES,MFUENTE,MARTICULO,ALLTRIM(STR(ESTRATO)),TARIFA,RAINI,RAFIN,0,0,0,0)
|       |       |       |       |       +-------ENDFOR
|       |       |       |       +-------ENDIF
|       |       |       |           SELE INSRANG
|       |       |       +-------ENDSCAN
|       |       +-------ENDIF
|       +-------ENDIF
|           SELECT V_TARIFAANT
|           TABLEUPDATE(.T.)
|
+-------ENDPROC
+-------PROCEDURE REVERT
```

```
|           NANSWER = MESSAGEBOX('Se eliminaran todos los cambios realizados, esta
seguro?',1+32+0,CMESSAGETITLE)
|     +-------DO CASE
|     +-------CASE NANSWER = 1
|     |           SELE &TABLA
|     |           TABLEREVERT(.T.)
|     |           THISFORM.REFRESH()
|     +-------ENDCASE
|
+-------ENDPROC
+-------PROCEDURE CALASEO
&&Calculo para ASEO  (No tiene cargo firjo ni rango de tarifas)
|           LOCAL VCTOS, REPLL, TABEST, TABRAN, TABTAR &&, Vlr1, Vlr2, Vlr3
|     +-------IF ANTERIOR
|     |           TABEST='v_estratoant'
|     |           TABRAN='v_rangoant'
|     |           TABTAR='v_tarifaant'
|     +-------ELSE
|     |           TABEST='v_estrato'
|     |           TABRAN='v_rango'
|     |           TABTAR='v_tarifa'
|     +-------ENDIF
|
|           VCTOS=0
|           REPLL=0
|           VLR1=0
|           VLR2=0
|           VLR3=0
|
|           SELE &TABLA
|           SUM VLR_TARIFA TO INFO
|
|           *if info!=0
|
|           SELE MAX(CODTARIFA)AS MAX ;
|                 FROM &TABRAN ;
|                 WHERE MES=CMES AND ANO=CANO AND CODFTE=MFUENTE AND
CODART=MARTICULO ;
|                 INTO CURSOR MAX
|           VCTOS = VAL(MAX.MAX)
|
|           SELE &TABRAN
&&Coloca como rango final el máximo valor de la variable
|           REPLA RAN_FIN WITH 999 FOR VAL(CODTARIFA)=VCTOS AND MES=CMES AND
ANO=CANO ;
|                 AND CODFTE=MFUENTE AND CODART=MARTICULO
&&Para el calculo del consum_tar (consumo por tarifa)
```

```
|           REPLA RANI WITH RAN_INI-1 FOR MES=CMES AND ANO=CANO AND
CODFTE=MFUENTE AND CODART=MARTICULO ;
|                 AND RAN_INI!= 0
|
|           REPLA RANI WITH 0 FOR MES=CMES AND ANO=CANO AND CODFTE=MFUENTE AND
CODART=MARTICULO ;
|                 AND RAN_INI=0
|
|           SELE &TABRAN
|           TABLEUPDATE(.T.)
|
|           SELECT &TABTAR
|           SET ORDER TO PKDESC
|           SET RELA TO
|           SET RELATION TO ANO+MES+CODART+CODFTE+CODESTRATO INTO &TABEST
ADDITIVE
|           SET RELATION TO ANO+MES+CODART+CODFTE+CODTARIFA INTO &TABRAN
ADDITIVE
|
|           ******* CALCULO
|
|           REPLA ALL VLR_CONSUM WITH VLR_TARIFA FOR ANO=CANO AND MES=CMES;
|                 AND CODFTE=MFUENTE AND CODART=MARTICULO
|
|           SELE &TABTAR
|           TABLEUPDATE(.T.)
|
|           GO TOP
|
|           TABLA='&locdir'+'\temp.dbf'
|           TOTAL TO &TABLA ON ANO+MES+CODFTE+CODART+CODESTRATO FIELDS
VLR_CONSUM ;
|                 FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO
|
|           USE &TABLA IN 0 EXCLU
|
|           SELE A.ANO, A.MES, A.CODART, A.CODFTE, A.VLR_TARIFA, A.VLR_CONSUM,
A.VLR_PONDE, ;
|                 A.CODESTRATO, B.CODGRI, B.USUARIOS, B.QO_POND, B.CARGOFIJO,
B.IMPUESTO, B.CONSUMO, B.FPDEPOR, ROUND(A.VLR_CONSUM+B.CARGOFIJO,2) AS
TOTVLRCONS;
|                 FROM TEMP A, &TABEST B;
|                 WHERE A.ANO=CANO AND A.MES=CMES AND A.CODART=MARTICULO AND
A.CODFTE=MFUENTE AND ;
|                 A.ANO=B.ANO AND A.MES=B.MES AND A.CODFTE=B.CODFTE ;
|                 AND A.CODART=B.CODART AND A.CODESTRATO= B.CODESTRATO;
|                 INTO TABLE TEMP1
```

```
|
|           SELE TEMP
|           USE  &&lo cierra para ejecutarlo de nuevo si es necesario
|
&&Agrega Cargo fijo e impuestos si existieren (adiciona el iva al cargo fijo)
|           SELECT TEMP1
|     +-------IF !EMPTY(IVA)
|     |           REPLA VLR_PONDE WITH
((VLR_CONSUM+IMPUESTO+(CARGOFIJO*IVA))*QO_POND)/100 ;
|     |                 FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO ;
|     |                 IN TEMP1
|     +-------ELSE
|     |           REPLA ALL VLR_PONDE WITH (TOTVLRCONS*QO_POND)/100 ;
|     |                 FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO;
|     |                 IN TEMP1
|     +-------ENDIF
|
|           SELE TEMP1
|           GO TOP
|
&&Suma el valor ponderado por grupo de ingreso
|     +-------SCAN FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO
|     |     +-------DO CASE
|     |     +-------CASE VAL(TEMP1.CODGRI)=1
|     |     |           VLR1=VLR1+VLR_PONDE
|     |     |
|     |     +-------CASE VAL(TEMP1.CODGRI)=2
|     |     |           VLR2=VLR2+VLR_PONDE
|     |     |
|     |     +-------CASE VAL(TEMP1.CODGRI)=3
|     |     |           VLR3=VLR3+VLR_PONDE
|     |     +-------ENDCASE
|     +-------ENDSCAN
&&Graba la informacion en las tablas y visualiza resultados
|           THISFORM.GRABA
+-------ENDPROC
+-------PROCEDURE VALIESTRATO
|           LOCAL ERROR, CUANTOS
|           ERROR=.F.
|           CUANTOS=0
|           THISFORM.CORRECTO=.T.
|
|           SELE &TABLA
|     +-------IF SUBSTR(MFUENTE,1,2)="54"
|     |     +-------SCAN
```

```
|       |       |       +-------IF !BETWEEN(CODGRI,'1','3') OR !BETWEEN(CODESTRATO,'1','9') &&or
consumo=0 or cargofijo=0 or usuarios=0
|       |       |       |               ERROR=.T.
|       |       |       +-------ENDIF
|       |       +-------ENDSCAN
|       +-------ELSE
|       |       +-------SCAN
|       |       |       +-------IF !BETWEEN(CODGRI,'1','3') OR !BETWEEN(CODESTRATO,'1','6') &&or
consumo=0 or cargofijo=0 or usuarios=0
|       |       |       |               ERROR=.T.
|       |       |       +-------ENDIF
|       |       +-------ENDSCAN
|       +-------ENDIF
|
|
&&Para validar que haya al menos 1 estrato
|               COUNT TO CTOSEST FOR !MARCA AND ANO=CANO AND MES=CMES AND
CODFTE=MFUENTE AND CODART=MARTICULO
|
|       +-------IF MARTICULO="2220101"  &&ENERGIA
|       |               CTOSEST=3
|       +-------ENDIF
|       +-------IF SUBSTR(MFUENTE,1,2)="54" AND  (MARTICULO="2220201" OR
MARTICULO="2220202" OR MARTICULO="2220203")&&solo para Cúcuta  acueducto y
alcantarillado y aseo
|       |               CTOSEST=9
|       +-------ENDIF
|
|       +-------IF ERROR OR CTOSEST=0
|       |               ANSWER = MESSAGEBOX('Datos incompletos ó mal diligenciados', NDIALOGTYPE,
CMESSAGETITLE)
|       |               THISFORM.CORRECTO=.F.
|       +-------ELSE
|       |               THISFORM.CALQOPOND
|       |               SELE &TABLA
|       |               TABLEUPDATE(.T.)
|       |               THISFORM.REFRESH()
|       +-------ENDIF
|
|
+-------ENDPROC
+-------PROCEDURE ADICIONE
&&Adiciona rangos ó estratos dependiendo de donde se hace el llamado
|               LOCAL NVONRO, GRI,DESRANG, TABEST, TABRAN, TABTAR, VCODTAR,
ANTNRO&&fuente, mvt1, mvt2, mvt3
|
|       +-------IF ANTERIOR
|       |               TABEST='v_estratoant'
```

```
|       |           TABRAN='v_rangoant'
|       |           TABTAR='v_tarifaant'
|       +-------ELSE
|       |           TABEST='v_estrato'
|       |           TABRAN='v_rango'
|       |           TABTAR='v_tarifa'
|       +-------ENDIF
|
|           SELE &TABLA
|
|           CALCULATE MAX(&CAMPO) ;
|                FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO ;
|                TO NVONRO
|
|           NVONRO=VAL(NVONRO)+1
|
|       +-------IF CAMPO='codestrato' &&Adiciona a la tabla de estrato
|       |           NVONRO=ALLTRIM(STR(NVONRO))
&&Controla que el rango del estrato estre entre 1 y 6
|       |       +-------IF NVONRO > '6'
|       |       |           NANSWER=MESSAGEBOX('No deben exitir mas de 6 Estratos.',
NDIALOGTYPE, CMESSAGETITLE)
|       |       +-------ELSE
|       |       |       +-------DO CASE
|       |       |       +-------CASE BETWEEN(NVONRO,'1','3')
|       |       |       |           GRI='1'
|       |       |       +-------CASE BETWEEN(NVONRO,'4','5')
|       |       |       |           GRI='2'
|       |       |       +-------CASE NVONRO='6'
|       |       |       |           GRI='3'
|       |       |       +-------ENDCASE
|       |       |
|       |       |           SELECT &TABLA
|       |       |           SET FILTER TO
|       |       |       +-------IF SEEK(CANO+CMES+MARTICULO+MFUENTE+NVONRO)
|       |       |       |       +-------IF MARCA
|       |       |       |       |           REPLACE MARCA WITH .F.
|       |       |       |       +-------ENDIF
|       |       |       +-------ELSE
|       |       |       |           INSERT INTO &TABLA
(&CAMPO,ANO,MES,CODFTE,CODART,CODGRI,USUARIOS,CONSUMO,CARGOFIJO,QO_POND
,IMPUESTO,FPDEPOR,MARCA) ;
|       |       |       |                   VALUES
(NVONRO,CANO,CMES,MFUENTE,MARTICULO,GRI,0,0,0,0,0,0,.F.)
|       |       |       |
|       |       |       +-------ENDIF
|       |       |
```

```
|     |     |           SELECT &TABLA
|     |     |           TABLEUPDATE(.T.)
|     |     |           SET FILTER TO MARCA=.F.
|     |     |           GO TOP
|     |     |           ** si adiciona un estrato entonces, con los rangos disponibles debe crear rangos
de tarifas para dicho estrato
|     |     |           ** en la tabla de tarifas
|     |     |           SELECT &TABTAR
|     |     |           SET ORDER TO TAG PKTARIFA
&&ano+mes+codart+codfte+codestrato+codtarifa
|     |     |           SELECT &TABRAN
|     |     |           SET ORDER TO TAG PKRANGO   &&ano+mes+codart+codfte+codtarifa
|     |     |           GO TOP
|     |     |     +-------DO WHILE !EOF()
|     |     |     |           VCODTAR=CODTARIFA
|     |     |     |           VRAN_INI=RAN_INI
|     |     |     |           VRAN_FIN=RAN_FIN
|     |     |     |           VRANI=RANI
|     |     |     |           SELECT &TABTAR
|     |     |     |           SET FILTER TO
|     |     |     |           GO TOP
|     |     |     |     +-------IF SEEK(CANO+CMES+MARTICULO+MFUENTE+NVONRO+VCODTAR)
|     |     |     |     |           REPLA RAN_INI WITH VRAN_INI, RAN_FIN WITH VRAN_FIN, RANI
WITH VRANI, MARCA WITH .F.
|     |     |     |     +-------ELSE
|     |     |     |     |           INSERT INTO &TABTAR
(ANO,MES,CODFTE,CODART,CODESTRATO,CODTARIFA,VLR_TARIFA,CONSUM_TAR,VLR_CO
NSUM,VLR_PONDE, RAN_INI,RAN_FIN,RANI,MARCA);
|     |     |     |     |                 VALUES
(CANO,CMES,MFUENTE,MARTICULO,NVONRO,VCODTAR,0,0,0,0,VRAN_INI,VRAN_FIN,VRANI,.
F.)
|     |     |     |     +-------ENDIF
|     |     |     |
|     |     |     |           SELECT &TABRAN
|     |     |     |           SKIP
|     |     |     +-------ENDDO
|     |     |           SELECT &TABTAR
|     |     |           TABLEUPDATE(.T.)
|     |     |           SET FILTER TO MARCA=.F.
|     |     |           GO TOP
|     |     +-------ENDIF
|     |
|     +-------ELSE  &&Adiciona a la tabla de rangos  campo = 'codtarifa'
|     |           CALCULATE MAX(RAN_FIN) ;
|     |                 FOR ANO=CANO AND MES=CMES AND CODFTE=MFUENTE AND
CODART=MARTICULO ;
|     |                 TO NVOINI
|     |           NVOINI=NVOINI+1
```

```
|       |       +-------IF NVOINI>999
|       |       |               ANSWER = MESSAGEBOX('Rango inicial no puede ser mayor a 999. Corrija el
rango final del registro anterior', NDIALOGTYPE, CMESSAGETITLE)
|       |       |               *RETURN
|       |       |               *NVoIni=NvoIni-1
|       |       +-------ELSE
|       |       |
|       |       |               DESRANG='Rango'+ALLTRIM(STR(NVOINI))
|       |       |               NVONRO=ALLTRIM(STR(NVONRO))
|       |       |               NVONRO=PADL(NVONRO,2,'0')
|       |       |               **LA UNIDAD
|       |       |               SELECT &TABLA
|       |       |               SET FILTER TO
|       |       |       +-------IF SEEK(CANO+CMES+MARTICULO+MFUENTE+NVONRO)
|       |       |       |       +-------IF MARCA
|       |       |       |       |               REPLACE MARCA WITH .F., UNIDAD WITH MUNID
|       |       |       |       +-------ENDIF
|       |       |       +-------ELSE
|       |       |       |               INSERT INTO &TABLA
(&CAMPO,ANO,MES,CODFTE,CODART,UNIDAD,RAN_INI,DESTARIFA,RAN_FIN,UNIDAD,RANI,M
ARCA) ;
|       |       |       |                       VALUES
(NVONRO,CANO,CMES,MFUENTE,MARTICULO,MUNID,NVOINI,DESRANG,999,MUNID,0,.F. )
|       |       |       +-------ENDIF
|       |       |               SELECT &TABLA
|       |       |               SET FILTER TO MARCA=.F.
|       |       |               GO TOP
|       |       |
|       |       |               ** se ajusta el valor del rango inicial del registro nuevo que se inserto a la tabla de
rangos
|       |       |
|       |       |               SELECT CODTARIFA, RAN_FIN FROM &TABLA WHERE
CODTARIFA<>NVONRO ORDER BY CODTARIFA DESC INTO CURSOR TMPRAN
|       |       |
|       |       |               SELECT TMPRAN
|       |       |               GO TOP
|       |       |               NVOINIRANNVO=RAN_FIN+1
|       |       |               SELECT &TABLA
|       |       |               GO BOTT
|       |       |       +-------IF !EOF()
|       |       |       |               REPLACE RAN_INI WITH NVOINIRANNVO
|       |       |       +-------ENDIF
|       |       |               GO TOP
|       |       |               * TABLEUPDATE(.T)
|       |       |
|       |       |               * si adiciona un RANGO enonces, PARA LOS diferentes estratos se crea un
rango de tarifas
|       |       |               ** en la tabla de tarifas
```

```
|     |     |           SELECT &TABTAR
|     |     |           SET ORDER TO TAG PKTARIFA
&&ano+mes+codart+codfte+codestrato+codtarifa
|     |     |
|     |     |           SELECT &TABEST
|     |     |           SET ORDER TO TAG PKESTRATO  &&ano+mes+codart+codfte+codestrato
|     |     |           GO TOP
|     |     |     +-------DO WHILE !EOF()
|     |     |     |           VCODESTRATO=CODESTRATO
|     |     |     |           SELECT &TABTAR
|     |     |     |           SET FILTER TO
|     |     |     |           GO TOP
|     |     |     |     +-------IF
SEEK(CANO+CMES+MARTICULO+MFUENTE+VCODESTRATO+NVONRO)
|     |     |     |     |     +-------IF MARCA
|     |     |     |     |     |           REPLACE MARCA WITH .F., RAN_INI WITH NVOINI, RAN_FIN
WITH 999
|     |     |     |     |     +-------ENDIF
|     |     |     |     +-------ELSE
|     |     |     |     |           INSERT INTO &TABTAR
(ANO,MES,CODFTE,CODART,CODESTRATO,CODTARIFA,VLR_TARIFA,CONSUM_TAR,VLR_CO
NSUM,VLR_PONDE, RAN_INI,RAN_FIN,RANI,MARCA);
|     |     |     |     |                 VALUES
(CANO,CMES,MFUENTE,MARTICULO,VCODESTRATO,NVONRO,0,0,0,0,NVOINI,999,0,.F.)
|     |     |     |     +-------ENDIF
|     |     |     |
|     |     |     |           SELECT &TABEST
|     |     |     |           SKIP
|     |     |     +-------ENDDO
|     |     |           SELECT &TABTAR
|     |     |           TABLEUPDATE(.T.)
|     |     |           SET FILTER TO MARCA=.F.
|     |     |           GO TOP
|     |     |           ** SE HACE UN BARRIDO SOBRE LA TABLA DE TARIFAS Y SE CUADRA EL
VALOR DEL RANGO ANTERIOR
|     |     |     +-------IF NVONRO>="02"  &&SI EL RANGO QUE ADICINO ES AL MENOS EL
SEGUNDO, ENTONCES EXISTE AL MENOS UN ANTERIOR
|     |     |     |           ** CUYOS VALORES DE RANGO FINAL DE DEBE AJUSTAR
|     |     |     |           SELECT DISTINCT CODESTRATO FROM &TABTAR ORDER BY
CODESTRATO INTO CURSOR CUALESTR
|     |     |     |           ANTNRO=VAL(NVONRO)-1
|     |     |     |           ANTNRO=ALLTRIM(STR(ANTNRO))
|     |     |     |           ANTNRO=PADL(ANTNRO,2,'0')
|     |     |     |           NVOFIN=NVOINI-1
|     |     |     |           SELECT CUALESTR
|     |     |     |           GO TOP
|     |     |     |     +-------SCAN
|     |     |     |     |           VCODESTRATO=CODESTRATO
```

```
|     |     |     |     |           SELECT &TABTAR
|     |     |     |     |     +-------IF
SEEK(CANO+CMES+MARTICULO+MFUENTE+VCODESTRATO+ANTNRO)  &&buscar el anterior
rango para verificar el valor del rango final
|     |     |     |     |     |           * debe encontrarlo
|     |     |     |     |     |           REPLACE RAN_FIN WITH NVOFIN
|     |     |     |     |     +-------ENDIF
|     |     |     |     |           SELECT CUALESTR
|     |     |     |     +-------ENDSCAN
|     |     |     +-------ENDIF
|     |     +-------ENDIF
|     +-------ENDIF
|           SELECT &TABLA
|           TABLEUPDATE(.T.)
|           GO BOTTOM
|           THISFORM.REFRESH()
|
|
+-------ENDPROC
+-------PROCEDURE ELIMINE
|           NANSWER = MESSAGEBOX('Está seguro de eliminar el rango
seleccionado?',1+32+0,CMESSAGETITLE)
|     +-------DO CASE
|     +-------CASE NANSWER = 1
|     |     +-------IF ANTERIOR
|     |     |           TABEST='v_estratoant'
|     |     |           TABRAN='v_rangoant'
|     |     |           TABTAR='v_tarifaant'
|     |     +-------ELSE
|     |     |           TABEST='v_estrato'
|     |     |           TABRAN='v_rango'
|     |     |           TABTAR='v_tarifa'
|     |     +-------ENDIF
|     |
|     |     +-------IF CAMPO='codestrato' &&eliminara de la tabla de estratos
|     |     |           SELECT &TABLA
|     |     |           REPLACE MARCA WITH .T.
|     |     |           VCODESTS=CODESTRATO  &&codigo del estrato que sale
|     |     |           ** se eliminaran tambien las tarifas correspondientes a este estrato que salio
|     |     |           SELECT &TABTAR
|     |     |           SET FILTER TO
|     |     |           GO TOP
|     |     |           REPLACE ALL MARCA WITH .T. FOR ANO=CANO AND MES=CMES AND
CODFTE=MFUENTE AND CODART=MARTICULO AND CODESTRATO=VCODESTS
|     |     |           SET FILTER TO MARCA=.F.
|     |     |           GO TOP
|     |     |
|     |     +-------ELSE &&ELIMINARA a la tabla de rangos
```

```
|     |     |
|     |     |           SELECT &TABLA
|     |     |           VCODRANS=CODTARIFA  &&codigo del del rango que sale
|     |     |           REPLACE MARCA WITH .T.
|     |     |           SET FILTER TO MARCA=.F.
|     |     |           GO BOTT
|     |     |     +-------IF !EOF()
|     |     |     |           REPLACE RAN_FIN WITH 999
|     |     |     +-------ENDIF
|     |     |
|     |     |           ** se eliminaran tambien las tarifas correspondientes a este RANGO que salio
|     |     |           SELECT &TABTAR
|     |     |           SET FILTER TO
|     |     |           GO TOP
|     |     |           REPLACE ALL MARCA WITH .T. FOR ANO=CANO AND MES=CMES AND
CODFTE=MFUENTE AND CODART=MARTICULO AND CODTARIFA=VCODRANS
|     |     |           SET FILTER TO MARCA=.F.
|     |     |           GO TOP
|     |     |           *aqui quede
|     |     |           ** SE HACE UN BARRIDO SOBRE LA TABLA DE TARIFAS Y SE CUADRA EL
VALOR DEL RANGO FINAL en 999
|     |     |
|     |     |           ANTNRO=VAL(VCODRANS)-1
|     |     |           ANTNRO=ALLTRIM(STR(ANTNRO))
|     |     |           ANTNRO=PADL(ANTNRO,2,'0')
|     |     |     +-------IF ANTNRO>'00' &&SI ES CERO O MENOR ES OR QUE NO QUEDAN MAS
RANGOS
|     |     |     |           SELECT DISTINCT CODESTRATO FROM &TABTAR ORDER BY
CODESTRATO INTO CURSOR CUALESTR
|     |     |     |           SELECT CUALESTR
|     |     |     |           GO TOP
|     |     |     |     +-------SCAN
|     |     |     |     |           VCODESTRATO=CODESTRATO
|     |     |     |     |           SELECT &TABTAR
|     |     |     |     |     +-------IF
SEEK(CANO+CMES+MARTICULO+MFUENTE+VCODESTRATO+ANTNRO)  &&buscar el anterior
rango para verificar el valor del rango final
|     |     |     |     |     |           * debe encontrarlo
|     |     |     |     |     |           REPLACE RAN_FIN WITH 999
|     |     |     |     |     +-------ENDIF
|     |     |     |     |           SELECT CUALESTR
|     |     |     |     +-------ENDSCAN
|     |     |     +-------ENDIF
|     |     |
|     |     +-------ENDIF
|     |
|     |           SELECT &TABLA
|     |           SET FILTER TO MARCA=.F.
```

```
|       |           GO TOP
|       |
|       |           THISFORM.REFRESH()
|       |
|       +-------ENDCASE
+-------ENDPROC
+-------PROCEDURE GRABA
|           LOCAL FUENTE, MVT1, MVT2, MVT3
|           FUENTE=MFUENTE
|           MVT1=''
|           MVT2=''
|           MVT3=''
|       +-------IF ANTERIOR &&cálculo del mes anterior
|       |       +-------IF !EMPTY(PREANTT1) AND !EMPTY(PREANTT2) AND !EMPTY(PREANTT3)
|       |       |           THISFORM.BXTAR.PAGE1.TVG1ANT.VALUE=((VLR1/PREANTT1)*100)-100
|       |       |           THISFORM.BXTAR.PAGE1.TVG2ANT.VALUE=((VLR2/PREANTT2)*100)-100
|       |       |           THISFORM.BXTAR.PAGE1.TVG3ANT.VALUE=((VLR3/PREANTT3)*100)-100
|       |       |
&&Actualiza tabla V_variames
|       |       |           SELE V_VARIAMES
|       |       |       +-------IF SEEK(CANO+CMES+MFUENTE+MARTICULO+'1')
|       |       |       |           REPLA PRECIO WITH VLR1, VARIACION WITH ((VLR1/PREANTT1)*100)-
100, IVA WITH IVA, MARCA WITH .F.
|       |       |       +-------ELSE
|       |       |       |           INSERT INTO V_VARIAMES
(ANO,MES,CODFTE,CODART,CODGRI,PRECIO,VARIACION,IVA,MARCA) VALUES
(CANO,CMES,MFUENTE,MARTICULO,'1',VLR1,(((VLR1/PREANTT1)*100)-100),IVA,MARCA)
|       |       |       +-------ENDIF
|       |       |
|       |       |       +-------IF SEEK(CANO+CMES+MFUENTE+MARTICULO+'2')
|       |       |       |           REPLA PRECIO WITH VLR2, VARIACION WITH ((VLR2/PREANTT2)*100)-
100, IVA WITH IVA, MARCA WITH .F.
|       |       |       +-------ELSE
|       |       |       |           INSERT INTO V_VARIAMES
(ANO,MES,CODFTE,CODART,CODGRI,PRECIO,VARIACION,IVA,MARCA) VALUES
(CANO,CMES,MFUENTE,MARTICULO,'2',VLR2,(((VLR2/PREANTT2)*100)-100),IVA,.F.)
|       |       |       +-------ENDIF
|       |       |
|       |       |       +-------IF SEEK(CANO+CMES+MFUENTE+MARTICULO+'3')
|       |       |       |           REPLA PRECIO WITH VLR3, VARIACION WITH ((VLR3/PREANTT3)*100)-
100, IVA WITH IVA, MARCA WITH .F.
|       |       |       +-------ELSE
|       |       |       |           INSERT INTO V_VARIAMES
(ANO,MES,CODFTE,CODART,CODGRI,PRECIO,VARIACION,IVA,MARCA) VALUES
(CANO,CMES,MFUENTE,MARTICULO,'3',VLR3,(((VLR3/PREANTT3)*100)-100),IVA,.F.)
|       |       |       +-------ENDIF
|       |       |
|       |       +-------ELSE  &&si no tiene valores preantt1, preantt2, preantt3
```

```
|     |     |
&&Actualiza tabla V_variames
|     |     |           SELE V_VARIAMES
|     |     |     +-------IF SEEK(CANO+CMES+MFUENTE+MARTICULO+'1')
|     |     |     |           *repla variacion with ((Vlr1/precio)*100)-100
|     |     |     |           REPLA VARIARECAL WITH ((VLR1/PRECIO)*100)-100
|     |     |     |
|     |     |     |           *repla precio with Vlr1, iva with iva, marca WITH .F.
|     |     |     |           REPLA PRECIORECA WITH VLR1, IVA WITH IVA, MARCA WITH .F.
|     |     |     |
|     |     |     |           *ThisForm.BxTar.Page1.TVG1ant.value=V_variames.variacion
|     |     |     |
THISFORM.BXTAR.PAGE1.TVG1ANT.VALUE=V_VARIAMES.VARIARECAL
|     |     |     |
|     |     |     +-------ELSE
|     |     |     |           INSERT INTO V_VARIAMES
(ANO,MES,CODFTE,CODART,CODGRI,PRECIO,VARIACION,IVA,MARCA,PRECIORECA,VARIAR
ECAL) VALUES (CANO,CMES,MFUENTE,MARTICULO,'1',VLR1,0,IVA,.F.,VLR1,0)
|     |     |     +-------ENDIF
|     |     |
|     |     |     +-------IF SEEK(CANO+CMES+MFUENTE+MARTICULO+'2')
|     |     |     |           *repla variacion with ((Vlr2/precio)*100)-100
|     |     |     |           REPLA VARIARECAL WITH ((VLR2/PRECIO)*100)-100
|     |     |     |
|     |     |     |           *repla precio with Vlr2, iva with iva, marca WITH .F.
|     |     |     |           REPLA PRECIORECA WITH VLR2, IVA WITH IVA, MARCA WITH .F.
|     |     |     |
|     |     |     |           *ThisForm.BxTar.Page1.TVG2ant.value=V_variames.variacion
|     |     |     |
THISFORM.BXTAR.PAGE1.TVG2ANT.VALUE=V_VARIAMES.VARIARECAL
|     |     |     |
|     |     |     +-------ELSE
|     |     |     |           INSERT INTO V_VARIAMES
(ANO,MES,CODFTE,CODART,CODGRI,PRECIO,VARIACION,IVA,MARCA,PRECIORECA,VARIAR
ECAL) VALUES (CANO,CMES,MFUENTE,MARTICULO,'2',VLR2,0,IVA,.F.,VLR2,0)
|     |     |     +-------ENDIF
|     |     |
|     |     |     +-------IF SEEK(CANO+CMES+MFUENTE+MARTICULO+'3')
|     |     |     |           *repla variacion with ((Vlr3/precio)*100)-100
|     |     |     |           REPLA VARIARECAL WITH ((VLR3/PRECIO)*100)-100
|     |     |     |
|     |     |     |           *repla precio with Vlr3, iva with iva, marca WITH .F.
|     |     |     |           REPLA PRECIORECA WITH VLR3, IVA WITH IVA, MARCA WITH .F.
|     |     |     |
|     |     |     |           *ThisForm.BxTar.Page1.TVG3ant.value=V_variames.variacion
|     |     |     |
THISFORM.BXTAR.PAGE1.TVG3ANT.VALUE=V_VARIAMES.VARIARECAL
|     |     |     |
```

```
|       |       |       +-------ELSE
|       |       |       |               INSERT INTO V_VARIAMES
(ANO,MES,CODFTE,CODART,CODGRI,PRECIO,VARIACION,IVA,MARCA,PRECIORECA,VARIAR
ECAL) VALUES (CANO,CMES,MFUENTE,MARTICULO,'3',VLR3,0,IVA,.F.,VLR3,0)
|       |       |       +-------ENDIF
|       |       +-------ENDIF
|       |
|       |               THISFORM.BXTAR.PAGE1.TG1ANT.VALUE=VLR1
|       |               THISFORM.BXTAR.PAGE1.TG2ANT.VALUE=VLR2
|       |               THISFORM.BXTAR.PAGE1.TG3ANT.VALUE=VLR3
|       |
|       |               SELE V_ESTRATOANT
|       |               *tableupdate(.t.)
|       |               REQUERY('v_estratoANT')
|       |
|       |               SELECT V_VARIAMES
|       |               TABLEUPDATE(.T.)
|       |               *requery('V_variames')
|       |
|       +-------ELSE  &&cálculo del mes actual
|       |               * 1 ////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
&&Actualiza tabla V_variames
|       |               SELE V_VARIAMES
|       |       +-------IF SEEK(CANO+CMES+MFUENTE+MARTICULO+'1')
|       |       |               REPLA PRECIO WITH VLR1, VARIACION WITH ((VLR1/PREANT1)*100)-100,
IVA WITH IVA, MARCA WITH .F.,;
|       |       |                       PRECIORECA WITH VLR1, VARIARECAL WITH ((VLR1/PREANT1)*100)-
100
|       |       +-------ELSE
|       |       |               INSERT INTO V_VARIAMES
(ANO,MES,CODFTE,CODART,CODGRI,PRECIO,VARIACION,IVA,MARCA,PRECIORECA,VARIAR
ECAL) VALUES (CANO,CMES,MFUENTE,MARTICULO,'1',VLR1,(((VLR1/PREANT1)*100)-
100),IVA,.F.,VLR1,(((VLR1/PREANT1)*100)-100))
|       |       +-------ENDIF
&&Remplaza en la tabla de movimientos por grupo de ingresos y fuente-articulo
|       |               SELE V_IPCTMVT8
|       |               FUENTE=STUFF(MFUENTE,14,1,'1')
|       |       +-------IF SEEK(FUENTE+MARTICULO)
|       |       |               V_PACTR=V_VARIAMES.PRECIO &&precio actual real
|       |       |               V_PANTR=PREANT1         &&precio anterior real
|       |       |               M.PACTMVT=(((100+V_VARIAMES.VARIACION)*PANART)/100)+0.01
|       |       |               REPLA PACTMVT WITH m.PACTMVT, CANRMVT WITH 1
|       |       +-------ELSE
|       |       |               MVT1='1'
|       |       +-------ENDIF
|       |
|       |               * 2 ////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
&&Actualiza tabla V_variames
```

```
|       |           SELE V_VARIAMES
|       |       +-------IF SEEK(CANO+CMES+MFUENTE+MARTICULO+'2')
|       |       |           REPLA PRECIO WITH VLR2, VARIACION WITH ((VLR2/PREANT2)*100)-100,
IVA WITH IVA, MARCA WITH .F.,;
|       |       |                   PRECIORECA WITH VLR2, VARIARECAL WITH ((VLR2/PREANT2)*100)-
100
|       |       +-------ELSE
|       |       |           INSERT INTO V_VARIAMES
(ANO,MES,CODFTE,CODART,CODGRI,PRECIO,VARIACION,IVA,MARCA,PRECIORECA,VARIAR
ECAL) VALUES (CANO,CMES,MFUENTE,MARTICULO,'2',VLR2,(((VLR2/PREANT2)*100)-
100),IVA,.F.,VLR2,(((VLR2/PREANT2)*100)-100))
|       |       +-------ENDIF
&&Remplaza en la tabla de movimientos por grupo de ingresos y fuente-articulo
|       |           SELE V_IPCTMVT8
|       |           FUENTE=STUFF(MFUENTE,14,1,'2')
|       |       +-------IF SEEK(FUENTE+MARTICULO)
|       |       |           V_PACTR=V_VARIAMES.PRECIO &&precio actual real
|       |       |           V_PANTR=PREANT2       &&precio anterior real
|       |       |           M.PACTMVT=(((100+V_VARIAMES.VARIACION)*PANART)/100)+0.01
|       |       |           REPLA PACTMVT WITH m.PACTMVT, CANRMVT WITH 1
|       |       +-------ELSE
|       |       |           MVT2='2'
|       |       +-------ENDIF
|       |
|       |           * 3
//////////////
```

```
|     |     |     |           REPLA PACTMVT WITH m.PACTMVT, CANRMVT WITH 1

|     |     |     +-------ELSE
|     |     |     |           MVT3='3'
|     |     |     +-------ENDIF
|     |     +-------ENDIF
|     |
|     |
|     |     +-------IF!EMPTY(MVT1) OR !EMPTY(MVT2) OR !EMPTY(MVT3)
|     |     |           ANSWER = MESSAGEBOX('Sólo se guardo el dato calculado para el grupo:
'+SUBSTR(MFUENTE,14,1)+ ',  ya que la fuente seleccionada es: '+ MFUENTE+'.  Debe agregar las
fuentes para los grupos: '+MVT1+', '+MVT2+', '+MVT3+',  a la tabla de movimientos', NDIALOGTYPE,
CMESSAGETITLE)
|     |     +-------ENDIF
|     |
&&Deja visualizar los resultados
|     |         THISFORM.BXTAR.PAGE2.TG1ACT.VALUE=VLR1
|     |         THISFORM.BXTAR.PAGE2.TG2ACT.VALUE=VLR2
|     |         THISFORM.BXTAR.PAGE2.TG3ACT.VALUE=VLR3
|     |         THISFORM.BXTAR.PAGE2.TVG1ACT.VALUE=((VLR1/PREANT1)*100)-100
|     |         THISFORM.BXTAR.PAGE2.TVG2ACT.VALUE=((VLR2/PREANT2)*100)-100
|     |         THISFORM.BXTAR.PAGE2.TVG3ACT.VALUE=((VLR3/PREANT3)*100)-100
|     |
|     |           SELE V_ESTRATO
|     |           *tableupdate(.t.)
|     |           REQUERY('v_estrato')
|     |
|     |           ***  OBSERVAR SI HUBO CAMBIOS EN COSNUMOS MEDIOS O EN USUARIOS
ENTRE LOS DOS ULTIMOS PERIODOS:
|     |           ** SI HUBO CAMBIOS ENTONCES SE DEBE INFORMAR AL USUARIO QUE DEBE
RECALCULAR PASANDOSE A LA CARPETA DEL MES ANTERIOR
|     |
|     |           SELECT V_ESTRATO.CODART, V_ESTRATO.USUARIOS AS USACT,
V_ESTRATOANT.USUARIOS AS USANT,;
|     |                 V_ESTRATO.CONSUMO  AS CONSUMOACT, V_ESTRATOANT.CONSUMO AS
CONSUMOANT,;
|     |                 IIF(V_ESTRATO.USUARIOS<>V_ESTRATOANT.USUARIOS,1,0) AS DIFERUS,;
|     |                 IIF(V_ESTRATO.CONSUMO<>V_ESTRATOANT.CONSUMO,1,0) AS
DIFERCON;
|     |                 FROM V_ESTRATO, V_ESTRATOANT;
|     |                 WHERE V_ESTRATO.CODART=V_ESTRATOANT.CODART AND
V_ESTRATO.CODESTRATO=V_ESTRATOANT.CODESTRATO;
|     |                 ORDER BY V_ESTRATO.CODART, V_ESTRATO.CODESTRATO;
|     |                 INTO CURSOR TMPDIFER
|     |
|     |           SELECT CODART, SUM(DIFERUS) AS TOTDIFERUS, SUM(DIFERCON) AS
TOTDIFERCON;
|     |                 FROM TMPDIFER;
```

```
|     |                 GROUP BY CODART;
|     |                 ORDER BY CODART;
|     |                 INTO CURSOR TMPDIFDEF
|     |
|     |     +-------IF TMPDIFDEF.TOTDIFERUS>0 OR TMPDIFDEF.TOTDIFERCON>0
|     |     |           ANSWER = MESSAGEBOX('Hubo cambio en usuarios o consumo medio. Debe
RECALCULAR el mes anterior y luego si CALCULAR el Actual!', NDIALOGTYPE, CMESSAGETITLE)
|     |     |
*****************************************************************************************
|     |     +-------ELSE
|     |     |
|     |     |           ********** imprimir  o vista previa: ************************************
|     |     |           ANSWER=MESSAGEBOX('Desea imprimir ?',4+32+256,'Resultados Mes:')
|     |     |     +-------DO CASE
|     |     |     +-------CASE ANSWER = 6
|     |     |     |           SELECT TEMP1
|     |     |     |           SET RELATION TO ANO+MES+CODFTE+CODART+CODGRI INTO
V_VARIAMES ADDITIVE
|     |     |     |           REPORT FORM LSERVPUB.FRX NOCONSOLE TO PRINTER PROMPT
|     |     |     +-------CASE ANSWER = 7
|     |     |     |           SELECT TEMP1
|     |     |     |           SET RELATION TO ANO+MES+CODFTE+CODART+CODGRI INTO
V_VARIAMES ADDITIVE
|     |     |     |           REPORT FORM LSERVPUB.FRX NOCONSOLE PREVIEW
|     |     |     +-------ENDCASE
|     |     |           ***************************************************************************
|     |     +-------ENDIF
|     +-------ENDIF
|           THISFORM.REQUERIES
|           THISFORM.REFRESH()
+-------ENDPROC
+-------PROCEDURE REQUERIES
|           LOCAL MMESANTT, MANOANTT, TAR, EST
|
|           MUNID=''
&&Da valor a la variable unidad
|
|
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN6.HEADER1.CAPTION='Imp
uesto'

|           THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN6.ENABLED=.F.
|
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN6.TEXT1.ENABLED=.F.
|           ** lo sgte para los datos del mes actual
|
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN6.HEADER1.CAPTION='
Impuesto'
```

```
|
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN6.ENABLED=.F.
|
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN6.TEXT1.ENABLED=.F.
|           ** lo sgte para los datos del mes anterior
|           THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN7.ENABLED=.F.
|
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN7.TEXT1.ENABLED=.F.
|           ** lo sgte para los datos del mes actual
|
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN7.ENABLED=.F.
|
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN7.TEXT1.ENABLED=.F.
|
|           ** como aqui aplica el solo al telefono, entonces, por defecto para el mes anterior se tiene :
|           THISFORM.BXTAR.PAGE1.EMBIVAANT.ENABLED=.F.
|           THISFORM.BXTAR.PAGE1.EMBIVAANT.VALUE=0
|           THISFORM.BXTAR.PAGE1.VIVAANT.VALUE=0
|           IVA=THISFORM.BXTAR.PAGE1.VIVAANT.VALUE
|           THISFORM.BXTAR.PAGE1.VIVAANT.ENABLED=.F.
|           THISFORM.BXTAR.PAGE1.V_LIMIMPANT.ENABLED=.F.
|
|           ** como aqui aplica el iva, entonces: para el mes actual
|           THISFORM.BXTAR.PAGE2.EMBIVAANT.ENABLED=.F.
|           THISFORM.BXTAR.PAGE2.EMBIVAANT.VALUE=0
|           THISFORM.BXTAR.PAGE2.VIVAANT.VALUE=0
|           IVA=THISFORM.BXTAR.PAGE2.VIVAANT.VALUE
|           THISFORM.BXTAR.PAGE2.VIVAANT.ENABLED=.F.
|           THISFORM.BXTAR.PAGE2.V_LIMIMP.ENABLED=.F.
|
|           ** lo sgte para los datos del mes anterior
|
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE3.GTARIFA.COLUMN3.HEADER1.CAPTION='Valor
Tarifa'
|           ** lo sgte para los datos del mes actual
|
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE3.GRID1.COLUMN3.HEADER1.CAPTION='Valor
Tarifa'
|
|           MUNID='kwt'
|           ESP2='6'
|           THISFORM.BXTAR.PAGE1.SPANT2.VALUE=6
|           THISFORM.BXTAR.PAGE2.SPANT2.VALUE=6
|
|           THISFORM.REFRESH()
|     +-------DO CASE
|     +-------CASE MARTICULO='2220201' &&Acueducto
|     |           MUNID='m3'
```

```
|       |
|       |       +-------IF SUBSTR(MFUENTE,1,2)="54" &&solo para Cúcuta
|       |       |               ESP2='9'
|       |       |               THISFORM.BXTAR.PAGE1.SPANT2.VALUE=9
|       |       |               THISFORM.BXTAR.PAGE2.SPANT2.VALUE=9
|       |       +-------ENDIF
|       |
|       |               ** lo sgte para los datos del mes anterior
|       |
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN6.HEADER1.CAPTION='Alu
mbradoP.'
|       |
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN6.ENABLED=.T.
|       |
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN6.TEXT1.ENABLED=.T.
|       |               ** lo sgte para los datos del mes actual
|       |
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN6.HEADER1.CAPTION='
AlumbradoP.'
|       |
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN6.ENABLED=.T.
|       |
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN6.TEXT1.ENABLED=.T.
|       |               THISFORM.REFRESH()
|       |
|       +-------CASE MARTICULO='2220202'  &&alcantarillado
|       |               MUNID='m3'
|       |       +-------IF SUBSTR(MFUENTE,1,2)="54" &&solo para Cúcuta
|       |       |               ESP2='9'
|       |       |               THISFORM.BXTAR.PAGE1.SPANT2.VALUE=9
|       |       |               THISFORM.BXTAR.PAGE2.SPANT2.VALUE=9
|       |       +-------ENDIF
|       |
|       +-------CASE MARTICULO='2210102'  &&Gas
|       |               MUNID='m3'
|       |
|       |       +-------IF SUBSTR(MFUENTE,1,2)="68" &&solo para BMANGA
|       |       |               ** lo sgte para los datos del mes anterior
|       |       |
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN6.HEADER1.CAPTION='Imp
uesto'
|       |       |
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN6.ENABLED=.T.
|       |       |
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN6.TEXT1.ENABLED=.T.
|       |       |               ** lo sgte para los datos del mes actual
```

```
|       |       |
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN6.HEADER1.CAPTION='
Impuesto'
|       |       |
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN6.ENABLED=.T.
|       |       |
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN6.TEXT1.ENABLED=.T.
|       |       |               THISFORM.REFRESH()
|       |       +-------ENDIF
|       |
|       |
|       |
|       +-------CASE MARTICULO='2220203' &&Aseo
|       |               MUNID='m3'
|       |       +-------IF SUBSTR(MFUENTE,1,2)="54" &&solo para Cúcuta
|       |       |               ESP2='9'
|       |       |               THISFORM.BXTAR.PAGE1.SPANT2.VALUE=9
|       |       |               THISFORM.BXTAR.PAGE2.SPANT2.VALUE=9
|       |       +-------ENDIF
|       |
|       |               ** lo sgte para los datos del mes anterior
|       |
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE3.GTARIFA.COLUMN3.HEADER1.CAPTION='Vlr.Ase
o'
|       |               ** lo sgte para los datos del mes actual
|       |
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE3.GRID1.COLUMN3.HEADER1.CAPTION='Vlr.Aseo'
|       |
|       +-------CASE MARTICULO='2220101' &&Energia
|       |               MUNID='kwt'
|       |               ESP2='3'
|       |               THISFORM.BXTAR.PAGE1.SPANT2.VALUE=3
|       |               THISFORM.BXTAR.PAGE2.SPANT2.VALUE=3
|       |               ** lo sgte para los datos del mes anterior
|       |
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN6.HEADER1.CAPTION='Alu
mbradoP.'
|       |
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN6.ENABLED=.T.
|       |
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN6.TEXT1.ENABLED=.T.
|       |               ** lo sgte para los datos del mes actual
|       |
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN6.HEADER1.CAPTION='
AlumbradoP.'
|       |
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN6.ENABLED=.T.
```

```
|       |
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN6.TEXT1.ENABLED=.T.
|       |           THISFORM.REFRESH()
|       |
|       +-------CASE MARTICULO='7320101' &&Telefónía
|       |           MUNID='Imp'
|       |           ** lo sgte para los datos del mes anterior
|       |
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN7.ENABLED=.T.
|       |
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.COLUMN7.TEXT1.ENABLED=.T.
|       |
|       |           ** como aqui aplica el iva, entonces: para el mes anterior
|       |           THISFORM.BXTAR.PAGE1.EMBIVAANT.ENABLED=.T.
|       |           THISFORM.BXTAR.PAGE1.EMBIVAANT.VALUE=1
|       |           THISFORM.BXTAR.PAGE1.VIVAANT.VALUE=16
|       |           IVA=THISFORM.BXTAR.PAGE1.VIVAANT.VALUE/100
|       |           THISFORM.BXTAR.PAGE1.VIVAANT.ENABLED=.T.
|       |           THISFORM.BXTAR.PAGE1.V_LIMIMPANT.ENABLED=.T.
|       |
|       |           ** como aqui aplica el iva, entonces: para el mes actual
|       |           THISFORM.BXTAR.PAGE2.EMBIVAANT.ENABLED=.T.
|       |           THISFORM.BXTAR.PAGE2.EMBIVAANT.VALUE=1
|       |           THISFORM.BXTAR.PAGE2.VIVAANT.VALUE=16
|       |           IVA=THISFORM.BXTAR.PAGE2.VIVAANT.VALUE/100
|       |           THISFORM.BXTAR.PAGE2.VIVAANT.ENABLED=.T.
|       |           THISFORM.BXTAR.PAGE2.V_LIMIMP.ENABLED=.T.
|       |
|       |           ** lo sgte para los datos del mes actual
|       |
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN7.ENABLED=.T.
|       |
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.COLUMN7.TEXT1.ENABLED=.T.
|       |
|       |           ** lo sgte para los datos del mes anterior
|       |
THISFORM.BXTAR.PAGE1.BXTARANT.PAGE3.GTARIFA.COLUMN3.HEADER1.CAPTION='Vlr.Imp
ulso'
|       |           ** lo sgte para los datos del mes actual
|       |
THISFORM.BXTAR.PAGE2.BXTARACT.PAGE3.GRID1.COLUMN3.HEADER1.CAPTION='Vlr.Impuls
o'
|       |
|       |           THISFORM.REFRESH()
|       +-------ENDCASE
|               THISFORM.REFRESH()
&&Para encontrar la informacion de el mes anterior al anterior
|       +-------IF MMESANT='01'
```

```
|       |           MMESANTT = '12'
|       |           MANOANTT = VAL(MANO)-1
|       |           MANOANTT = ALLTRIM(STR(MANOANTT))
|       |           *       mAnoAntt= alltrim(str(V_IPCTCTV.AÑOCTV-1))
|       +-------ELSE
|       |           MMESANTT = VAL(MMESANT)-1
|       |           MMESANTT = PADL(ALLTRIM(STR(MMESANTT)),2,'0')
|       |           MANOANTT = MANO
|       +-------ENDIF
&&Muestra el dato del mes anterior
|           SELE V_VARIAMES
|       +-------IF SEEK(CANO+CMES+MFUENTE+MARTICULO+'1')
|       |           * 1)
|       |           *preant1=V_variames.precio &&Mes inmediatamente anterior al periodo actual
|       |       +-------IF EMPTY(V_VARIAMES.PRECIORECA)
|       |       |           REPLACE V_VARIAMES.PRECIORECA WITH V_VARIAMES.PRECIO,
V_VARIAMES.VARIARECAL WITH V_VARIAMES.VARIACION
|       |       +-------ENDIF
|       |           PREANT1=V_VARIAMES.PRECIORECA &&Mes inmediatamente anterior al periodo
actual
|       |           THISFORM.BXTAR.PAGE1.TG1ANT.VALUE=PREANT1
|       |           *ThisForm.BxTar.Page1.TVG1Ant.value=V_variames.variacion
|       |           THISFORM.BXTAR.PAGE1.TVG1ANT.VALUE=V_VARIAMES.VARIARECAL
|       |
|       |           * 2)
|       |           SEEK(CANO+CMES+MFUENTE+MARTICULO+'2')
|       |           *preant2=V_variames.precio &&Mes inmediatamente anterior al periodo actual
|       |       +-------IF FOUND()
|       |       |       +-------IF EMPTY(V_VARIAMES.PRECIORECA)
|       |       |       |           REPLACE V_VARIAMES.PRECIORECA WITH V_VARIAMES.PRECIO,
V_VARIAMES.VARIARECAL WITH V_VARIAMES.VARIACION
|       |       |       +-------ENDIF
|       |       +-------ENDIF
|       |           PREANT2=V_VARIAMES.PRECIORECA &&Mes inmediatamente anterior al periodo
actual
|       |           THISFORM.BXTAR.PAGE1.TG2ANT.VALUE=PREANT2
|       |           *ThisForm.BxTar.Page1.TVG2Ant.value=V_variames.variacion
|       |           THISFORM.BXTAR.PAGE1.TVG2ANT.VALUE=V_VARIAMES.VARIARECAL
|       |
|       |           * 3)
|       |           SEEK(CANO+CMES+MFUENTE+MARTICULO+'3')
|       |           *preant3=V_variames.precio &&Mes inmediatamente anterior al periodo actual
|       |       +-------IF FOUND()
|       |       |       +-------IF EMPTY(V_VARIAMES.PRECIORECA)
|       |       |       |           REPLACE V_VARIAMES.PRECIORECA WITH V_VARIAMES.PRECIO,
V_VARIAMES.VARIARECAL WITH V_VARIAMES.VARIACION
|       |       |       +-------ENDIF
|       |       +-------ENDIF
```

```
|       |           PREANT3=V_VARIAMES.PRECIORECA &&Mes inmediatamente anterior al periodo
actual
|       |           THISFORM.BXTAR.PAGE1.TG3ANT.VALUE=PREANT3
|       |           *ThisForm.BxTar.Page1.TVG3Ant.value=V_variames.variacion
|       |           THISFORM.BXTAR.PAGE1.TVG3ANT.VALUE=V_VARIAMES.VARIARECAL
|       |
|       |
|       |
&&Si el calculo anterior tuvo IVA
|       |     +-------IF !EMPTY(V_VARIAMES.IVA)
|       |     |           THISFORM.BXTAR.PAGE1.EMBIVAANT.VALUE=.T.
|       |     |           THISFORM.BXTAR.PAGE1.VIVAANT=V_VARIAMES
|       |     |           IVA=V_VARIAMES.IVA
|       |     +-------ENDIF
|       +-------ENDIF
|
|       +-------IF SEEK(MANOANTT+MMESANTT+MFUENTE+MARTICULO+'1')
|       |           * 1)
|       |           *preantt1=V_variames.precio &&Mes anterior al periodo anterior
|       |     +-------IF EMPTY(V_VARIAMES.PRECIORECA)
|       |     |           REPLACE V_VARIAMES.PRECIORECA WITH V_VARIAMES.PRECIO,
V_VARIAMES.VARIARECAL WITH V_VARIAMES.VARIACION
|       |     +-------ENDIF
|       |           PREANTT1=V_VARIAMES.PRECIORECA &&Mes anterior al periodo anterior
|       |
|       |           * 2)
|       |           SEEK(MANOANTT+MMESANTT+MFUENTE+MARTICULO+'2')
|       |     +-------IF FOUND()
|       |     |     +-------IF EMPTY(V_VARIAMES.PRECIORECA)
|       |     |     |           REPLACE V_VARIAMES.PRECIORECA WITH V_VARIAMES.PRECIO,
V_VARIAMES.VARIARECAL WITH V_VARIAMES.VARIACION
|       |     |     +-------ENDIF
|       |     +-------ENDIF
|       |
|       |           *preantt2=V_variames.precio &&Mes anterior al periodo anterior
|       |           PREANTT2=V_VARIAMES.PRECIORECA &&Mes anterior al periodo anterior
|       |
|       |           * 3)
|       |           SEEK(MANOANTT+MMESANTT+MFUENTE+MARTICULO+'3')
|       |     +-------IF FOUND()
|       |     |     +-------IF EMPTY(V_VARIAMES.PRECIORECA)
|       |     |     |           REPLACE V_VARIAMES.PRECIORECA WITH V_VARIAMES.PRECIO,
V_VARIAMES.VARIARECAL WITH V_VARIAMES.VARIACION
|       |     |     +-------ENDIF
|       |     +-------ENDIF
|       |
|       |           *preantt=V_variames.precio &&Mes anterior al periodo anterior
|       |           PREANTT=V_VARIAMES.PRECIORECA &&Mes anterior al periodo anterior
```

```
|       |
|       +-------ENDIF
|
|
&&Deshace lo que se haya grabado en caso de que quiera calcular otra fuente-articulo
|               TABLEREVERT(.T.,'v_tarifaant')
|               TABLEREVERT(.T.,'v_estratoant')
|               TABLEREVERT(.T.,'v_rangoant')
|               TABLEREVERT(.T.,'v_tarifa')
|               TABLEREVERT(.T.,'v_estrato')
|               TABLEREVERT(.T.,'v_rango')
&&Consulta información mes anterior
|               REQUERY('v_tarifaant')
|               REQUERY('v_estratoant')
|               REQUERY('v_rangoant')
|               SELE V_RANGOANT
|               REPLA RANI WITH 0 FOR RECNO()=1
|
&&Consulta información mes actual
|               REQUERY('v_tarifa')
|               REQUERY('v_estrato')
|               REQUERY('v_rango')
|
|               THISFORM.BXTAR.PAGE1.BXTARANT.ENABLED=.T.
|               THISFORM.CFUENTE.ENABLED=.F.
|               THISFORM.CARTICULO.ENABLED=.F.
|
|               THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.SETFOCUS()
|
|               SELECT V_RANGOANT
|               SET FILTER TO MARCA=.F.
|               GO TOP
|
|               SELE V_TARIFAANT
|               SET FILTER TO MARCA=.F.
|               GO TOP
|               COUNT TO TAR
|
|               TABLA='v_estratoant'
|               SELE &TABLA
|               SET FILTER TO MARCA=.F.
|               GO TOP
|               COUNT TO EST
|
|       +-------IF EMPTY(TAR) OR EMPTY(EST)
|       |               ANSWER = MESSAGEBOX('No existe informacion para esa fuente, debe digitar todos
los datos del mes anterior y mes actual para ejecutar el cálculo', NDIALOGTYPE, CMESSAGETITLE)
|       |               THISFORM.BXTAR.PAGE1.BXTARANT.ENABLED=.T.
```

```
|       |           THISFORM.CFUENTE.ENABLED=.T.
|       |           THISFORM.CARTICULO.ENABLED=.T.
|       +-------ELSE
|       |           THISFORM.BXTAR.PAGE1.BXTARANT.ENABLED=.T.
|       |           THISFORM.CFUENTE.ENABLED=.F.
|       |           THISFORM.CARTICULO.ENABLED=.F.
|       +-------ENDIF
|           SELE &TABLA
|           THISFORM.REFRESH()
|
+-------ENDPROC
+-------PROCEDURE INIT
|           *LPARAMETERS NoArticulo,NoFuente,Fuente,Artículo
|           PUBLIC
MANO,MMES,MANOANT,MMESANT,TABLA,CMESSAGETITLE,CMESSAGETEXT,NDIALOGTYPE,
MARTICULO,;
|                   MFUENTE,MUNID LOCDIR,CANO,CMES,CAMPO, CTOSEST,
IVA,ESP1,ESP2,PREANT1,PREANT2,PREANT3, ;
|                   ANTERIOR, PREANTT1, PREANTT2, PREANTT3, VLR1, VLR2, VLR3, V_LIMIMP,
V_LIMIMPANT, V_PERIODO, ESENTER &&mDec
|           SET CENTURY ON
|           SET DATE DMY
|           SET DELETED ON
|           SET MULTILOCKS ON
|           SET TALK OFF
|           SET NOTIFY OFF
|           SET SAFETY OFF
|           ESENTER=.F.
|           SELECT V_IPCTMVT8
|           REQUERY('v_ipctmvt8')
|           SET FILTER TO MARCA=.F.
|           INDEX ON CODFTE+CODART TAG NKMVT
|
|           SELECT V_VARIAMES
|           SET FILTER TO MARCA=.F.
|           INDEX ON ANO+MES+CODFTE+CODART+CODGRI TAG PKVARIAMES
|
|           SELECT V_ESTRATO
|           SET FILTER TO MARCA=.F.
|           INDEX ON ANO+MES+CODART+CODFTE+CODESTRATO TAG PKESTRATO
|
|           SELECT V_ESTRATOANT
|           SET FILTER TO MARCA=.F.
|           INDEX ON ANO+MES+CODART+CODFTE+CODESTRATO TAG PKESTRATO
|
|           SELECT V_RANGO
|           SET FILTER TO MARCA=.F.
|           INDEX ON ANO+MES+CODART+CODFTE+CODTARIFA TAG PKRANGO
```

```
|
|           SELECT V_RANGOANT
|           SET FILTER TO MARCA=.F.
|           INDEX ON ANO+MES+CODART+CODFTE+CODTARIFA TAG PKRANGO
|
|           SELECT V_TARIFA
|           SET FILTER TO MARCA=.F.
|           INDEX ON ANO+MES+CODART+CODFTE+CODESTRATO+CODTARIFA TAG PKDESC
DESC
|           INDEX ON ANO+MES+CODART+CODFTE+CODESTRATO+CODTARIFA TAG
PKTARIFA
|
|           SELECT V_TARIFAANT
|           SET FILTER TO MARCA=.F.
|           INDEX ON ANO+MES+CODART+CODFTE+CODESTRATO+CODTARIFA TAG PKDESC
DESC
|           INDEX ON ANO+MES+CODART+CODFTE+CODESTRATO+CODTARIFA TAG
PKTARIFA
|
|
|           THISFORM.ACEPTO=.F.
|           IVA=16/100
|           V_LIMIMPANT=250
|           V_LIMIMP=250
|
&&Directorio inicial
|           LOCDIR=SET('DEFAULT')+SYS(2003)
|
&&Asigna valores a el rango de aplicacion de los rangos por estrato
|           ESP1='1'
|           ESP2='6'
|           THISFORM.BXTAR.PAGE1.SPANT1.VALUE=1
|           THISFORM.BXTAR.PAGE1.SPANT2.VALUE=6
|           THISFORM.BXTAR.PAGE2.SPANT1.VALUE=1
|           THISFORM.BXTAR.PAGE2.SPANT2.VALUE=6
|
|           THISFORM.BXTAR.PAGE1.V_LIMIMPANT.VALUE=250
|           THISFORM.BXTAR.PAGE2.V_LIMIMP.VALUE=250
|
|
&&Para los mensajes de error
|           CMESSAGETITLE = 'IPC - Servicios Públicos'
|           CMESSAGETEXT = 'Datos Incompletos, Revise.'
|           NDIALOGTYPE = 0 + 48 + 0
|
&&lee la fecha de proceso
|           SELEC V_IPCTCTV
|           GO BOTTOM
```

```
|           SKIP -1
|           MANO= ALLTRIM(STR(V_IPCTCTV.AÑOCTV))
|           MMES= PADL(ALLTRIM(STR(V_IPCTCTV.MESCTV)),2,'0')
|           *     mDec= alltrim(str(V_IPCTCTV.DECCTV))
|           THISFORM.V_PERIODO.VALUE=CMONTH(CTOD('27/'+MMES+'/'+MANO))+"/"+MANO
|     +-------IF MMES='01'
|     |           MMESANT = '12'
|     |           *       mAnoAnt = alltrim(str(V_IPCTCTV.AÑOCTV-1))
|     |           MANOANT = VAL(MANO)-1
|     |           MANOANT = ALLTRIM(STR(MANOANT))
|     +-------ELSE
|     |           *       mMesAnt = padl(alltrim(str(V_IPCTCTV.MESCTV-1)),2,'0')
|     |           *       mAnoAnt = alltrim(str(V_IPCTCTV.AÑOCTV))
|     |           MMESANT = VAL(MMES)-1
|     |           MMESANT = PADL(ALLTRIM(STR(MMESANT)),2,'0')
|     |           MANOANT = MANO
|     +-------ENDIF
|
|           *cAno=mAnoAnt
|           *cMes=mMesAnt
|           CANO=MANO
|           CMES=MMES
|
|           THISFORM.CFUENTE.VALUE=''
|           THISFORM.CARTICULO.VALUE=''
|
|
+-------ENDPROC
+-------PROCEDURE UNLOAD
|     +-------IF THISFORM.ACEPTO
|     |           RETURN THISFORM.PROMEDIO
|     +-------ELSE
|     |           RETURN 'N'
|     +-------ENDIF
+-------ENDPROC
+-------PROCEDURE DESTROY
&&Al salir del formulario actualizamos el archivo de registro para saber
&&que este usuario ya salio de este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para indicar
&&que la cerro en este momento
|           LOCATE FOR ALLTRIM(V_REGISTRO.QUIEN)+ALLTRIM(V_REGISTRO.NPANTALLA) =
ALLTRIM(PQUIEN)+WONTOP( )
|     +-------IF FOUND()
&&Actualizamos la informacion y tambien el archivo
```

```
|       |           REPLACE V_REGISTRO.ACT_INA WITH .F.
|       |           TABLEUPDATE(.T.)
|       +-------ENDIF
+-------ENDPROC
+-------PROCEDURE ACTIVATE
&&Al entrar al formulario actualizamos el archivo de registro para saber
&&que este usuario abrio este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para saber
&&si ya esta relacionado en el archivo y no voilver a registrarlo
|           LOCATE FOR (ALLTRIM(V_REGISTRO.QUIEN) + NPANTALLA = ALLTRIM(PQUIEN) +
WONTOP()) AND V_REGISTRO.ACT_INA = .T.
&&si no es encontrado lo registramos y actualizamos el archivo
|       +-------IF !FOUND()
|       |           INSERT INTO V_REGISTRO(QUIEN, NPANTALLA, FECHAR, ACT_INA);
|       |                 VALUES(PQUIEN,WONTOP( ), DATE(), .T.)
|       |           TABLEUPDATE(.T.)
|       +-------ENDIF
|
+-------ENDPROC
        TOP = 0
        LEFT = 0
        WIDTH = 624
        HEIGHT = 70
        BORDERWIDTH = 1
        SPECIALEFFECT = 0
        TABINDEX = 5
        NAME = "Container1"
        ERASEPAGE = .T.
        TABSTYLE = 0
        ACTIVEPAGE = 1
        TOP = 72
        LEFT = 0
        WIDTH = 625
        HEIGHT = 377
        TABINDEX = 2
        NAME = "BxTar"
        PAGE1.FONTBOLD = .T.
        PAGE1.FONTSIZE = 12
        PAGE1.CAPTION = "Datos Mes Anterior"
        PAGE1.FORECOLOR = 0,0,255
        PAGE1.NAME = "Page1"
        PAGE2.FONTBOLD = .T.
        PAGE2.FONTITALIC = .F.
        PAGE2.FONTSIZE = 12
```

```
      PAGE2.CAPTION = "Datos Mes Actual"
      PAGE2.FORECOLOR = 0,0,255
      PAGE2.NAME = "Page2"
+-------PROCEDURE PAGE1.CLICK
|             CANO=MANOANT
|             CMES=MMESANT
|
|             THIS.BXTARANT.ACTIVEPAGE=1
|             TABLA='v_estratoant'
|             SELE &TABLA
|             THIS.BXTARANT.PAGE1.GESTRATO.SETFOCUS()
|             THISFORM.REFRESH()
|
+-------ENDPROC
+-------PROCEDURE PAGE2.CLICK
|             PREANT1=THISFORM.BXTAR.PAGE1.TG1ANT.VALUE
|             PREANT2=THISFORM.BXTAR.PAGE1.TG2ANT.VALUE
|             PREANT3=THISFORM.BXTAR.PAGE1.TG3ANT.VALUE
|
&&      Toma los valores del IVA y rango de estratos para el mes actual
|
THISFORM.BXTAR.PAGE2.EMBIVAANT.VALUE=THISFORM.BXTAR.PAGE1.EMBIVAANT.VALUE
|
THISFORM.BXTAR.PAGE2.VIVAANT.VALUE=THISFORM.BXTAR.PAGE1.VIVAANT.VALUE
|
THISFORM.BXTAR.PAGE2.SPANT1.VALUE=THISFORM.BXTAR.PAGE1.SPANT1.VALUE
|
THISFORM.BXTAR.PAGE2.SPANT2.VALUE=THISFORM.BXTAR.PAGE1.SPANT2.VALUE
|
|             CANO=MANO
|             CMES=MMES
|
&&Consulta informacion nuevo mes
|             THISFORM.NVOMES
|
|             THIS.BXTARACT.ACTIVEPAGE=1
|             THIS.BXTARACT.PAGE1.GESTRATOACT.SETFOCUS()
|
|             TABLA='v_estrato'
|             SELE &TABLA
|             THISFORM.REFRESH()
+-------ENDPROC

+-------PROCEDURE REFRESH
|
|
+-------ENDPROC
```

```
+-------PROCEDURE REFRESH
|
|
+-------ENDPROC


+-------PROCEDURE PAGE1.CLICK
|           TABLA='v_estratoant'
|           SELE &TABLA
|
|
+-------ENDPROC
+-------PROCEDURE PAGE2.CLICK
|
|
+-------ENDPROC
+-------PROCEDURE PAGE3.CLICK
|           TABLA='v_tarifaant'
|           SELE &TABLA
+-------ENDPROC


+-------PROCEDURE DELETED
|           LPARAMETERS NRECNO
|           SET DELETED ON
+-------ENDPROC

+-------PROCEDURE VALID
|     +-------IF !BETWEEN(THIS.VALUE,'1','3')
|     |           NANSWER=MESSAGEBOX('El Grupo de Ingreso debe estar entre 1 y 3. Corrija',
NDIALOGTYPE, CMESSAGETITLE)
|     +-------ENDIF
+-------ENDPROC
+-------PROCEDURE KEYPRESS
|           LPARAMETERS NKEYCODE, NSHIFTALTCTRL
|     +-------IF NKEYCODE=13
|     |           SELECT V_ESTRATOANT
|     |     +-------IF !EOF()
|     |     |           SKIP 1
|     |     |           THISFORM.BXTAR.PAGE1.BXTARANT.PAGE1.GESTRATO.REFRESH
|     |     +-------ENDIF
|     +-------ENDIF
|
+-------ENDPROC

+-------PROCEDURE CLICK
|           THISFORM.VALIESTRATO
|
```

```
|       +-------IF  THISFORM.CORRECTO=.T.
|       |           THISFORM.BXTAR.PAGE1.BXTARANT.ACTIVEPAGE=2
|       |           TABLA='v_rangoant'
|       |           SELE &TABLA
|       |           THISFORM.BXTAR.PAGE1.SPANT2.KEYBOARDHIGHVALUE=CTOSEST
|       |           THISFORM.BXTAR.PAGE1.SPANT2.VALUE=CTOSEST
|       |           EST2=ALLTRIM(STR(CTOSEST))
|       |           THISFORM.REFRESH()
|       +-------ENDIF
+-------ENDPROC
        TOP = 176
        LEFT = 19

+-------PROCEDURE CLICK
|           ANTERIOR=.T.
|           CAMPO='codestrato'
|           THISFORM.ADICIONE
|
|
+-------ENDPROC

+-------PROCEDURE CLICK
|           ANTERIOR=.T.
|           CAMPO='codestrato'
|           SELE &TABLA
|           THISFORM.ELIMINE
|
+-------ENDPROC

+-------PROCEDURE CLICK
|           THISFORM.REVERT
|
+-------ENDPROC


+-------PROCEDURE KEYPRESS
|           LPARAMETERS NKEYCODE, NSHIFTALTCTRL
|
|       +-------IF NKEYCODE=13
|       |           SELECT &TABLA
|       |       +-------IF !EOF()
|       |       |           SKIP 1
|       |       |           THISFORM.BXTAR.PAGE1.BXTARANT.PAGE2.GRANGO.REFRESH
|       |       +-------ENDIF

|       +-------ENDIF
|
+-------ENDPROC
```

```
+-------PROCEDURE KEYPRESS
|             LPARAMETERS NKEYCODE, NSHIFTALTCTRL
|
+-------ENDPROC

+-------PROCEDURE CLICK
|             THISFORM.VALIRANGOS
|      +-------IF THISFORM.CORRECTO=.T.
|      |             THISFORM.BXTAR.PAGE1.BXTARANT.ACTIVEPAGE=3
|      |             THISFORM.BXTAR.PAGE1.REFRESH()
|      |             TABLA='v_tarifaant'
|      |             SELE &TABLA
|      |             REQUERY('&tabla')
|      |             SET FILTER TO MARCA=.F.
|      |             GO TOP
|      +-------ENDIF
+-------ENDPROC

+-------PROCEDURE CLICK
|             ANTERIOR=.T.
|             CAMPO='codtarifa'
|             THISFORM.ADICIONE
|
|
+-------ENDPROC

+-------PROCEDURE CLICK
|             ANTERIOR=.T.
|             CAMPO='codtarifa'
|             SELE &TABLA
|             THISFORM.ELIMINE
|
|
+-------ENDPROC

+-------PROCEDURE CLICK
|             THISFORM.REVERT
|
+-------ENDPROC

+-------PROCEDURE CLICK
|             THISFORM.VALIRANGOS
|
|      +-------IF THISFORM.CORRECTO=.T.
|      |             TABLA='v_estratoant'
|      |             SELE &TABLA
|      |             THISFORM.BXTAR.PAGE1.BXTARANT.ACTIVEPAGE=1
```

```
|       |               THISFORM.BXTAR.PAGE1.BXTARANT.REFRESH()
|       +-------ENDIF
|
|
+-------ENDPROC


+-------PROCEDURE KEYPRESS
|               LPARAMETERS NKEYCODE, NSHIFTALTCTRL
|
|       +-------IF NKEYCODE=13
|       |               SELECT &TABLA
|       |       +-------IF !EOF()
|       |       |               SKIP 1
|       |       |               THISFORM.BXTAR.PAGE1.BXTARANT.PAGE3.GTARIFA.REFRESH
|       |       +-------ENDIF
|       +-------ENDIF
|
+-------ENDPROC

+-------PROCEDURE CLICK
|               SELE &TABLA
|               TABLEUPDATE(.T.)
|
|               TABLA='v_rangoant'
|               SELE &TABLA
|               THISFORM.BXTAR.PAGE1.BXTARANT.ACTIVEPAGE=2
|               THISFORM.BXTAR.PAGE1.BXTARANT.REFRESH()
+-------ENDPROC

+-------PROCEDURE CLICK
|               SELE &TABLA
|               TABLEUPDATE(.T.)
|
|               ANTERIOR=.T.
|
&&DEPENDIENDO DEL ARTICULO SE EJECUTA SI ES ASEO ES DIFERENTE....
|       +-------IF MARTICULO='2220203'
|       |               THISFORM.CALASEO
|       +-------ELSE
|       |               THISFORM.CALVALCONSUMO
|       +-------ENDIF
|               TABLA='v_tarifaant'
|
|
+-------ENDPROC

+-------PROCEDURE REFRESH
```

```
|
|
+-------ENDPROC

+-------PROCEDURE REFRESH
|
|
+-------ENDPROC

+-------PROCEDURE REFRESH
|
|
+-------ENDPROC


+-------PROCEDURE VALID
|           ESP1=ALLTRIM(STR(THIS.VALUE))
+-------ENDPROC

+-------PROCEDURE VALID
|           ESP2=ALLTRIM(STR(THIS.VALUE))
+-------ENDPROC

+-------PROCEDURE CLICK
|     +-------IF THIS.VALUE=1
|     |           THISFORM.BXTAR.PAGE1.VIVAANT.ENABLED=.T.
|     |           THISFORM.BXTAR.PAGE1.V_LIMIMPANT.ENABLED=.T.
|     |           THIS.TOOLTIPTEXT='Si aplica el IVA para este cálculo'
|     |           THISFORM.BXTAR.PAGE1.VIVAANT.SETFOCUS
|     +-------ELSE
|     |           THISFORM.BXTAR.PAGE1.VIVAANT.ENABLED=.F.
|     |           THISFORM.BXTAR.PAGE1.V_LIMIMPANT.ENABLED=.F.
|     |           THISFORM.BXTAR.PAGE1.VIVAANT.VALUE=0
|     |           THISFORM.BXTAR.PAGE1.V_LIMIMPANT.VALUE=0
|     |           THIS.TOOLTIPTEXT='No aplica el IVA para este cálculo'
|     +-------ENDIF
+-------ENDPROC

+-------PROCEDURE VALID
|     +-------IF EMPTY(THIS.VALUE)
|     |           THISFORM.BXTAR.PAGE1.EMBIVAANT.VALUE=0
|     |           THISFORM.BXTAR.PAGE1.EMBIVAANT.TOOLTIPTEXT='No aplica el IVA para este
cálculo'
|     +-------ELSE
|     |           IVA=THIS.VALUE/100
|     +-------ENDIF
|
+-------ENDPROC
```

```
+-------PROCEDURE CLICK
|             THISFORM.BXTAR.ACTIVEPAGE=2
|             THISFORM.BXTAR.PAGE2.CLICK()
|
+-------ENDPROC

+-------PROCEDURE CLICK
|             THISFORM.RELEASE
|
|
|
+-------ENDPROC


+-------PROCEDURE VALID
|       +-------IF EMPTY(THIS.VALUE)
|       |             THISFORM.BXTAR.PAGE1.EMBIVAANT.VALUE=1
|       |             THISFORM.BXTAR.PAGE1.EMBIVAANT.TOOLTIPTEXT='No aplica el Nro. de
Impulsos para este cálculo'
|       +-------ELSE
|       |             V_LIMIMPANT=THISFORM.BXTAR.PAGE1.V_LIMIMPANT.VALUE
|       +-------ENDIF
|
+-------ENDPROC


+-------PROCEDURE PAGE1.CLICK
|             TABLA='v_estrato'
|             SELE &TABLA
|
+-------ENDPROC
+-------PROCEDURE PAGE2.CLICK
|             TABLA='v_rango'
|             SELE &TABLA
|
|             LOCAL ERROR,ERROR1,NROANT,ESTRATO,TARIFA
|             ERROR=.F.
|             ERROR1=.F.
|             ERROR2=.F.
|             TARIFA=''
|
|             SELE &TABLA
|       +-------SCAN
|       |             SCATTER FIELDS RAN_INI MEMVAR
|       |
|       |       +-------IF RAN_INI=0 AND RECNO()=1
```

```
|       |       |               LOOP
|       |       +-------ENDIF
|       |
|       |       +-------IF RAN_INI=0 OR RAN_FIN=0
|       |       |               ERROR=.T.
|       |       +-------ENDIF
|       |
|       |       +-------IF RAN_INI>RAN_FIN
|       |       |               ERROR1=.T.
|       |       +-------ENDIF
|       |
|       |               * controlar que el rango inicial sea continuación del rango anterior
|       |       +-------IF RECNO()!= 1
|       |       |               NROANT=VAL(CODTARIFA)-1
|       |       |               NROANT=PADL(ALLTRIM(STR(NROANT)),2,'0')
|       |       |               SELE RAN_FIN FROM &TABLA WHERE CODTARIFA=NROANT INTO
CURSOR CUALANT
|       |       |
|       |       |       +-------IF m.RAN_INI != CUALANT.RAN_FIN+1
|       |       |       |               ERROR2=.T.
|       |       |       +-------ENDIF
|       |       +-------ENDIF
|       +-------ENDSCAN
|
|       +-------IF ERROR OR ERROR1 OR ERROR2
|       |       +-------IF ERROR
|       |       |               NANSWER=MESSAGEBOX(CMESSAGETEXT, NDIALOGTYPE,
CMESSAGETITLE)
|       |       |               THIS.PARENT.GRANGO.SETFOCUS
|       |       +-------ENDIF
|       |       +-------IF ERROR1
|       |       |               NANSWER=MESSAGEBOX('El rango final es mayor que el inicial. Revise',
NDIALOGTYPE, CMESSAGETITLE)
|       |       |               THIS.PARENT.GRANGO.SETFOCUS
|       |       +-------ENDIF
|       |
|       |       +-------IF ERROR2
|       |       |               CMESSAGETEXT='Los rangos no son continuos. Revise'
|       |       |               NANSWER=MESSAGEBOX(CMESSAGETEXT, NDIALOGTYPE,
CMESSAGETITLE)
|       |       |               THIS.PARENT.GRANGO.SETFOCUS
|       |       +-------ENDIF
|       |
|       +-------ELSE
|       |               SELE &TABLA
|       |               TABLEUPDATE(.T.)
|       |
&&Revisa si se borro un rango de la vista v_rango, que ya existia
```

```
|       |               SELE DISTINCT CODTARIFA FROM V_TARIFA WHERE
ALLTRIM(CODTARIFA)NOT IN (SELE ALLTRIM(CODTARIFA) AS LLAVE FROM V_RANGO) INTO
CURSOR BORRANG
|       |               SELE BORRANG
|       |       +-------IF !EOF()
|       |       |               SELE V_TARIFA
|       |       |               REPLACE ALL MARCA WITH .T. FOR CODTARIFA=LLAVE
|       |       +-------ENDIF
|       |
&&Revisa si se ingresaron rangos a la vista v_rango, que no existian
|       |               SELE DISTINCT CODTARIFA FROM V_RANGO WHERE
ALLTRIM(CODTARIFA)NOT IN (SELE DISTINCT ALLTRIM(CODTARIFA) AS LLAVE FROM
V_TARIFA) INTO CURSOR INSRANG
|       |               SELE INSRANG
|       |       +-------IF !EOF()
|       |       |               GO TOP
&&Revisa si se ingreso algun codigo de tarifa nuevo
|       |       |       +-------SCAN
|       |       |       |               TARIFA=CODTARIFA
|       |       |       |               SELE V_TARIFA
|       |       |       |               LOCATE FOR CODTARIFA=TARIFA
|       |       |       |       +-------IF !FOUND()
|       |       |       |       |       +-------FOR ESTRATO=1 TO 6
|       |       |       |       |       |               INSERT INTO V_TARIFA
(ANO,MES,CODFTE,CODART,CODESTRATO,CODTARIFA) ;
|       |       |       |       |       |                       VALUES
(MANO,MMES,MFUENTE,MARTICULO,ALLTRIM(STR(ESTRATO)),TARIFA)
|       |       |       |       |       +-------ENDFOR
|       |       |       |       +-------ENDIF
|       |       |       |               SELE INSRANG
|       |       |       +-------ENDSCAN
|       |       +-------ENDIF
|       |
|       |               TABLA='v_tarifa'
|       |               SELE &TABLA
|       |               TABLEUPDATE(.T.)
|       |               REQUERY('v_tarifa')
|       +-------ENDIF
+-------ENDPROC


+-------PROCEDURE VALID
|       +-------IF ESENTER
|       |               ESENTER=.F.
|       |
|       |               SELECT &TABLA
|       |       +-------IF !EOF()
|       |       |               SKIP 1
```

```
|       |       +-------ENDIF
|       +-------ENDIF
+-------ENDPROC
+-------PROCEDURE KEYPRESS
|               LPARAMETERS NKEYCODE, NSHIFTALTCTRL
|       +-------IF NKEYCODE=13 &&enter
|       |               ESENTER=.T.
|       +-------ELSE
|       |               ESENTER=.F.
|       +-------ENDIF
+-------ENDPROC

+-------PROCEDURE KEYPRESS
|               LPARAMETERS NKEYCODE, NSHIFTALTCTRL
|       +-------IF NKEYCODE=13
|       |               SELECT V_ESTRATO
|       |       +-------IF !EOF()
|       |       |               SKIP 1
|       |       |               THISFORM.BXTAR.PAGE2.BXTARACT.PAGE1.GESTRATOACT.REFRESH
|       |       +-------ENDIF
|       +-------ENDIF
|
|
|
+-------ENDPROC

+-------PROCEDURE CLICK
|               THISFORM.VALIESTRATO
|
|       +-------IF  THISFORM.CORRECTO=.T.
|       |               THISFORM.BXTAR.PAGE2.BXTARACT.ACTIVEPAGE=2
|       |               TABLA='v_rango'
|       |               SELE &TABLA
|       |               THISFORM.BXTAR.PAGE2.SPANT2.KEYBOARDHIGHVALUE=CTOSEST
|       |               THISFORM.BXTAR.PAGE2.SPANT2.VALUE=CTOSEST
|       |               EST2=ALLTRIM(STR(CTOSEST))
|       |               THISFORM.REFRESH()
|       +-------ENDIF
|
|
+-------ENDPROC

+-------PROCEDURE CLICK
|               ANTERIOR=.F.
|               CAMPO='codestrato'
|               THISFORM.ADICIONE

|
```

```
+-------ENDPROC

+-------PROCEDURE CLICK
|             ANTERIOR=.F.
|             CAMPO='codestrato'
|             SELE &TABLA
|             THISFORM.ELIMINE
|
+-------ENDPROC

+-------PROCEDURE CLICK
|             THISFORM.REVERT
|
+-------ENDPROC

+-------PROCEDURE COLUMN3.ADDOBJECT
|             LPARAMETERS CNAME, CCLASS
|
|
+-------ENDPROC



+-------PROCEDURE KEYPRESS
|             LPARAMETERS NKEYCODE, NSHIFTALTCTRL
|
|      +-------IF NKEYCODE=13
|      |             SELECT &TABLA
|      |      +-------IF !EOF()
|      |      |             SKIP 1
|      |      |             THISFORM.BXTAR.PAGE2.BXTARANT.PAGE2.GRANGOACT.REFRESH
|      |      +-------ENDIF
|      +-------ENDIF
+-------ENDPROC

+-------PROCEDURE CLICK
|             THISFORM.VALIRANGOS
|
|      +-------IF THISFORM.CORRECTO=.T.
|      |             THISFORM.BXTAR.PAGE2.BXTARACT.ACTIVEPAGE=3
|      |             THISFORM.BXTAR.PAGE2.REFRESH()
|      |             TABLA='v_tarifa'
|      |             SELE &TABLA
|      |             REQUERY('&tabla')
|      +-------ENDIF
|
+-------ENDPROC
```

```
+-------PROCEDURE CLICK
|             ANTERIOR=.F.
|             CAMPO='codtarifa'
|             THISFORM.ADICIONE
|
+-------ENDPROC

+-------PROCEDURE CLICK
|             ANTERIOR=.F.
|             CAMPO='codtarifa'
|             SELE &TABLA
|             THISFORM.ELIMINE
|
+-------ENDPROC


+-------PROCEDURE CLICK
|             THISFORM.REVERT
+-------ENDPROC


+-------PROCEDURE CLICK
|             THISFORM.VALIRANGOS
|
|      +-------IF THISFORM.CORRECTO=.T.
|      |             TABLA='v_estrato'
|      |             SELE &TABLA
|      |             THISFORM.BXTAR.PAGE2.BXTARACT.ACTIVEPAGE=1
|      |             THISFORM.BXTAR.PAGE2.BXTARACT.REFRESH()
|      +-------ENDIF
|
+-------ENDPROC

+-------PROCEDURE VALID
|             SELECT &TABLA
|      +-------IF !EOF()
|      |             SKIP 1
|      +-------ENDIF
+-------ENDPROC

+-------PROCEDURE CLICK

|             SELE &TABLA
|             TABLEUPDATE(.T.)
|
|             TABLA='v_rango'
|             SELE &TABLA
|             THISFORM.BXTAR.PAGE2.BXTARACT.ACTIVEPAGE=2
```

```
|           THISFORM.BXTAR.PAGE2.BXTARACT.REFRESH()
|
+-------ENDPROC


+-------PROCEDURE CLICK
|           SELE &TABLA
|           TABLEUPDATE(.T.)
|
|           ANTERIOR=.F.
|
&&DEPENDIENDO DEL ARTICULO SE EJECUTA SI ES ASEO ES DIFERENTE....
|     +-------IF MARTICULO='2220203'
|     |           THISFORM.CALASEO
|     +-------ELSE
|     |           THISFORM.CALVALCONSUMO
|     +-------ENDIF
|           TABLA='v_tarifa'
|           THISFORM.BXTAR.PAGE1.COMMAND2.ENABLED=.T.
|           THISFORM.BXTAR.PAGE1.COMMAND2.SETFOCUS
|           THISFORM.REFRESH
+-------ENDPROC


+-------PROCEDURE CLICK
|           THISFORM.ACEPTO=.T.
|           *ThisForm.Release
|
|
|
|           THISFORM.BXTAR.PAGE2.COMMAND2.SETFOCUS
+-------ENDPROC

+-------PROCEDURE CLICK
|           THISFORM.RELEASE
+-------ENDPROC

+-------PROCEDURE VALID
|           ESP1=ALLTRIM(STR(THIS.VALUE))
+-------ENDPROC


+-------PROCEDURE VALID
|           ESP2=ALLTRIM(STR(THIS.VALUE))
+-------ENDPROC


+-------PROCEDURE CLICK
```

```
|       +-------IF THIS.VALUE=1
|       |             THISFORM.BXTAR.PAGE2.VIVAANT.ENABLED=.T.
|       |             THISFORM.BXTAR.PAGE2.V_LIMIMP.ENABLED=.T.
|       |             THIS.TOOLTIPTEXT='Si aplica el IVA para este cálculo'
|       |             THISFORM.BXTAR.PAGE2.VIVAANT.SETFOCUS
|       +-------ELSE
|       |             THISFORM.BXTAR.PAGE2.VIVAANT.ENABLED=.F.
|       |             THISFORM.BXTAR.PAGE2.V_LIMIMP.ENABLED=.F.
|       |             THISFORM.BXTAR.PAGE2.VIVAANT.VALUE=0
|       |             THISFORM.BXTAR.PAGE2.V_LIMIMP.VALUE=0
|       |             THIS.TOOLTIPTEXT='No aplica el IVA para este cálculo'
|       +-------ENDIF
|
|
+-------ENDPROC

+-------PROCEDURE VALID
|       +-------IF EMPTY(THIS.VALUE)
|       |             THISFORM.BXTAR.PAGE2.EMBIVAANT.VALUE=.F.
|       |             THISFORM.BXTAR.PAGE2.EMBIVAANT.TOOLTIPTEXT='No aplica el IVA para este
cálculo'
|       +-------ELSE
|       |             IVA=THIS.VALUE/100
|       +-------ENDIF
|
+-------ENDPROC

+-------PROCEDURE VALID
|       +-------IF EMPTY(THIS.VALUE)
|       |             THISFORM.BXTAR.PAGE2.EMBIVAANT.VALUE=1
|       |             THISFORM.BXTAR.PAGE2.EMBIVAANT.TOOLTIPTEXT='No aplica el Nro. de
Impulsos para este cálculo'
|       +-------ELSE
|       |             V_LIMIMP=THIS.VALUE
|       +-------ENDIF
|
+-------ENDPROC
```

5.10 FORMULARIO PARA INGRESAR USUARIOS (Iusuario.SCX)

5.10.1 Vistas

```
                v_registro
                v_auxiliar

+-------PROCEDURE ACTIVATE
```

```
&&Al entrar al formulario actualizamos el archivo de registro para saber
&&que este usuario abrio este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para saber
&&si ya esta relacionado en el archivo y no voilver a registrarlo
                        LOCATE FOR (ALLTRIM(V_REGISTRO.QUIEN) + NPANTALLA =
                        ALLTRIM(PQUIEN) + WONTOP()) AND V_REGISTRO.ACT_INA = .T.
&&si no es encontrado lo registramos y actualizamos el archivo
|     +-------IF !FOUND()
                        INSERT INTO V_REGISTRO(QUIEN, NPANTALLA, FECHAR, ACT_INA);
|     |           VALUES(PQUIEN,WONTOP( ), DATE(), .T.)
|     |           TABLEUPDATE(.T.)
|     +-------ENDIF
|
+-------ENDPROC

+-------PROCEDURE INIT
&&Ocultamos el menu y declaramos variables a utilizar
|           SET SYSMENU OFF
|           PUBLIC PQUIEN, PCONQUE, PQUE, PCARGO, PCODIGO
|
|           PQUIEN=" "
|           PCONQUE=" "
|           PQUE=1
|           PCARGO = " "
|           PCODIGO = " "
&&Inicializamos el objeto spinner
|           THISFORM.SPINNER1.VALUE = 1
|
&&Configuramos el ambiente trabajo para el manejo de fechas
|           SET CENTURY ON
|           SET DATE DMY
|
|           THISFORM.TEXT1.SETFOCUS
+-------ENDPROC

+-------PROCEDURE UNLOAD
&&Volvemos a mostrar el menu
|           SET SYSMENU ON
+-------ENDPROC

+-------PROCEDURE DESTROY
&&Al salir del formulario actualizamos el archivo de registro para saber
&&que este usuario ya salio de este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
```

```
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para indicar
&&que la cerro en este momento
                        LOCATE FOR
                        ALLTRIM(V_REGISTRO.QUIEN)+ALLTRIM(V_REGISTRO.NPANTALLA) =
                        ALLTRIM(PQUIEN)+WONTOP( )
|     +-------IF FOUND()
|     |           REPLACE V_REGISTRO.ACT_INA WITH .F.
&&Actualizamos la informacion y tambien el archivo
|     |           TABLEUPDATE(.T.)
|     +-------ENDIF
+-------ENDPROC
```

5.10.2 Boton cancelar

```
+-------PROCEDURE CLICK
&&Salir de esta pantalla
|           THISFORM.RELEASE
|
+-------ENDPROC
```

5.10.3 Boton aceptar

```
+-------PROCEDURE CLICK
&&Solicitamos nos confirme la selección de registrar al usuario en cuestión
                MENSAJE = MESSAGEBOX(" Esta seguro ? ",4+32+256," Ingreso de Usuarios ")
&&Si nos confirman
|     +-------DO CASE
|     +-------CASE MENSAJE = 6
&&Almacenamos en una variable el nombre de usuario que se escribieron
|     |           PQUIEN = THISFORM.TEXT1.VALUE
&&A cada información igresada le asignamos una variable auxiliar
&&NOmbre usuario
|     |           QQUIEN = PQUIEN
&&Calve de Usuario
|     |           PCONQUE = THISFORM.TEXT2.VALUE

&&Nivel de acceso
|     |           PQUE = ALLTRIM(STR(THISFORM.SPINNER1.VALUE,1,0))
&&Cargo de Usuario
|     |           PCARGO = THISFORM.TEXT4.VALUE
&&Codigo de Usuario
|     |           PCODIGO = THISFORM.TEXT5.VALUE
&&Reconsultamos la infrmación del archivo donde esta almacenada la informacion
&&de los usuarios que estan registrados para manejar el aplicativo
|     |           REQUERY('v_auxiliar')
```

```
|       |           SELE V_AUXILIAR
&&Encriptamos la informacion basica
&&NIvel de Acceso
|       |       +-------FOR I = 1 TO 1
|       |       |           LETRA = SUBSTR(PQUE,I,1)
|       |       |           NLETRA = CHR(ASC(LETRA)+10)
|       |       |           PQUE = STUFF(PQUE,I,1,NLETRA)
|       |       +-------ENDFOR
&&Clave de acceso
|       |       +-------FOR I = 1 TO 8
|       |       |           LETRA = SUBSTR(PCONQUE,I,1)
|       |       |           NLETRA = CHR(ASC(LETRA)+10)
|       |       |           PCONQUE = STUFF(PCONQUE,I,1,NLETRA)
|       |       +-------ENDFOR
&&Nombre de Usuario
|       |       +-------FOR I = 1 TO 20
|       |       |           LETRA = SUBSTR(PQUIEN,I,1)
|       |       |           NLETRA = CHR(ASC(LETRA)+10)
|       |       |           PQUIEN = STUFF(PQUIEN,I,1,NLETRA)
|       |       +-------ENDFOR
&&Si escribieron algo para el codigo hacemos lo mismo(Encriptamos)
|       |       +-------IF !EMPTY(PCODIGO)
|       |       |       +-------FOR I = 1 TO 10
|       |       |       |           LETRA = SUBSTR(PCODIGO,I,1)
|       |       |       |           NLETRA = CHR(ASC(LETRA)+10)
|       |       |       |           PCODIGO = STUFF(PCODIGO,I,1,NLETRA)
|       |       |       +-------ENDFOR
|       |       +-------ENDIF
&&Si escribieron algo para el cargo hacemos lo mismo(Encriptamos)
|       |       +-------IF !EMPTY(PCARGO)
|       |       |       +-------FOR I = 1 TO 25
|       |       |       |           LETRA = SUBSTR(PCARGO,I,1)
|       |       |       |           NLETRA = CHR(ASC(LETRA)+10)
|       |       |       |           PCARGO = STUFF(PCARGO,I,1,NLETRA)
|       |       |       +-------ENDFOR
|       |       +-------ENDIF
&&Buscamos la informacion del usuario segun el nombre que estan tratando de ingresar a ver si ya
existe
&&para que no se vuelva a utilizar el mismo nombre de usuario para mas de uno de estos
|       |           LOCATE FOR V_AUXILIAR.QUIEN = PQUIEN
|       |       +-------IF FOUND()
MENSAJE=MESSAGEBOX(" Ya existe usuario con este nombre corrija por favor !!! ",0+48+0,"
Ingreso de Usuarios ")
|       |       |           THISFORM.TEXT1.VALUE=" "
|       |       |           THISFORM.TEXT2.VALUE=" "
|       |       |           THISFORM.SPINNER1.VALUE = 3
|       |       |
|       |       |           THISFORM.TEXT4.VALUE = " "
```

```
|     |     |           THISFORM.TEXT5.VALUE = " "
|     |     |
|     |     |           THISFORM.TEXT1.SETFOCUS
&&Si no es encontrado se almacena la información
|     |     +-------ELSE
                          INSERT INTO V_AUXILIAR(QUIEN, CONQUE, QUE, CODIGO, CARGO);
                          VALUES(PQUIEN, PCONQUE, PQUE, PCODIGO, PCARGO)
|     |     |           TABLEUPDATE(.T.)
|     |     |           THISFORM.TEXT1.VALUE=" "
|     |     |           THISFORM.TEXT2.VALUE=" "
|     |     |           THISFORM.SPINNER1.VALUE = 3
|     |     |
|     |     |           THISFORM.TEXT4.VALUE = " "
|     |     |           THISFORM.TEXT5.VALUE = " "
|     |     |
MENSAJE=MESSAGEBOX(" La clave de"+" "+ALLTRIM(QQUIEN)+" "+" ya fue reconocida como
clave del sistema ",0+64+0," Ingreso de Usuarios ")
|     |     |
|     |     |           THISFORM.CANCELAR.SETFOCUS
|     |     +-------ENDIF
&&Procedimiento cancelado
|     +-------CASE MENSAJE=7
MENSAJE=MESSAGEBOX(" Usuario no almacenado ",0+64+0," Ingreso de Usuarios ")
|     +-------ENDCASE
|
+-------ENDPROC

+-------PROCEDURE GOTFOCUS
&&valñidamos la informacion del usuario a ingresar(Nombre y clave)
        +-------IF LEN(ALLTRIM(THISFORM.TEXT1.VALUE))<4 OR
        LEN(ALLTRIM(THISFORM.TEXT2.VALUE))<4
                          MENSAJE=MESSAGEBOX(" Verifique que el nombre o la clave del usuario que
                          esta registrando tenga 4 o mas caracteres ",0+48+0," Ingreso de Usuarios ")
|     |       THISFORM.TEXT1.SETFOCUS
|     |       THISFORM.ACEPTAR.ENABLED=.F.
|     +-------ENDIF
+-------ENDPROC
```

5.10.4 Clave de Acceso(Text2)

```
+-------PROCEDURE GOTFOCUS
|     +-------IF LEN(ALLTRIM(THISFORM.TEXT1.VALUE))<4
                          MENSAJE=MESSAGEBOX(" Verifique que el nombre del usuario que esta
                          registrando tenga 4 o mas caracteres ",0+48+0," Ingreso de Usuarios ")
|     |           THISFORM.TEXT1.SETFOCUS
|     +-------ENDIF
+-------ENDPROC
```

5.10.5 Nivel de Acceso (Spinner1)

```
+-------PROCEDURE VALID
&&Validamos la consistencia de la informacion para el caso del nombre y clave de usuario
&&Dejamos como  requeridos el nombre, clave y nivel de acceso, los demas son opcionales
        +-------IF LEN(ALLTRIM(THISFORM.TEXT1.VALUE)) >= 4 AND
        LEN(ALLTRIM(THISFORM.TEXT2.VALUE)) >= 4 AND THIS.VALUE <> 0
|     |           THISFORM.ACEPTAR.ENABLED=.T.
|     +-------ENDIF
+-------ENDPROC
```

5.10.6 Codigo (Text5)

```
+-------PROCEDURE GOTFOCUS
|     +-------IF LEN(ALLTRIM(THISFORM.TEXT1.VALUE))<4
                        MENSAJE=MESSAGEBOX(" Verifique que el nombre del usuario que esta
                        registrando tenga 4 o mas caracteres ",0+48+0," Ingreso de Usuarios ")
|     |       THISFORM.TEXT1.SETFOCUS
|     +-------ENDIF
+-------ENDPROC
```

5.11 FORMULARIO PARA MODIFICAR USUARIOS (musuario.SCX)

5.11.1 Vistas

```
                v_auxiliar
                v_registro

+-------PROCEDURE DESTROY
&&Al salir del formulario actualizamos el archivo de registro para saber
&&que este usuario ya salio de este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para indicar
&&que la cerro en este momento
                        LOCATE FOR
                        ALLTRIM(V_REGISTRO.QUIEN)+ALLTRIM(V_REGISTRO.NPANTALLA) =
                        ALLTRIM(PQUIEN)+WONTOP( )
|     +-------IF FOUND()
|     |           REPLACE V_REGISTRO.ACT_INA WITH .F.
&&Actualizamos la informacion y tambien el archivo
|     |           TABLEUPDATE(.T.)
|     +-------ENDIF
+-------ENDPROC
```

```
+-------PROCEDURE INIT
&&Ocultamos el menu del aplicativo
|           SET SYSMENU OFF
&&Definimos e inicializamos variables para manejar los nombre, claves y
&&niveles de acceso de los usuarios
PUBLIC PQUIEN, PCONQUE, PCONQUE1, PQUE, PMARCA1,  PMARCA2, PMARCA3,
PCONQUEAUX, PQUEAUX, PCHEQUIEN, PCHECONQUE
PUBLIC PCARGO, PCODIGO, PCARGOAUX, PCODIGOAUX , PACTIVO
|
|           PQUIEN=" "
|           PCONQUE=" "
|           CONQUE1=" "
|           QUE=" "
|
|           THISFORM.SPINNER1.VALUE=3
|           STORE .F. TO m.MARCA1,m.MARCA2,m.MARCA3
&&Configuramos el ambiente de trabajo para el manejo de formatos de fechas
|           SET CENTURY ON
|           SET DATE DMY
|
|           THISFORM.TEXT1.SETFOCUS
+-------ENDPROC

+-------PROCEDURE UNLOAD
&&Volvemos a mostrar el menu del aplicativo
|           SET SYSMENU ON
+-------ENDPROC

+-------PROCEDURE ACTIVATE
&&Al entrar al formulario actualizamos el archivo de registro para saber
&&que este usuario abrio este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para saber
&&si ya esta relacionado en el archivo y no voilver a registrarlo
                  LOCATE FOR (ALLTRIM(V_REGISTRO.QUIEN) + NPANTALLA = ALLTRIM(PQUIEN)
                  + WONTOP()) AND V_REGISTRO.ACT_INA = .T.
&&si no es encontrado lo registramos y actualizamos el archivo
|      +-------IF !FOUND()
                        INSERT INTO V_REGISTRO(QUIEN, NPANTALLA, FECHAR, ACT_INA);
|      |           VALUES(PQUIEN,WONTOP( ), DATE(), .T.)
|      |            TABLEUPDATE(.T.)
|      +-------ENDIF
+-------ENDPROC
```

5.11.2 Boton Salir

```
+-------PROCEDURE CLICK
&&Salir de esta pantalla
|           THISFORM.RELEASE
+-------ENDPROC
```

5.11.3 Boton Aceptar

```
+-------PROCEDURE CLICK
&&Solicitamos confirmar la modificacion del usuario
                    MENSAJE=MESSAGEBOX(" Esta seguro ? ",4+32+256," Modificar Usuarios ")
|     +-------DO CASE
|     +-------CASE MENSAJE=6
&&En caso afirmativo cargamos en variables lo digitado por el usuario
|     |
|     |           QQUIEN = THISFORM.TEXT1.VALUE
|     |           PQUIEN = THISFORM.TEXT1.VALUE
|     |           PQUE = ALLTRIM(STR(THISFORM.SPINNER1.VALUE,1,0))
|     |           PCONQUE1 = THISFORM.TEXT4.VALUE
|     |           PCODIGO = THISFORM.TEXT6.VALUE
|     |           PCARGO = THISFORM.TEXT5.VALUE
|     |           PACTIVO = THISFORM.CHECK1.VALUE
&&Se encripta la informacion ingresada por le usuario
|     |           SELE V_AUXILIAR
&&Nivel de Acceso
|     |     +-------FOR I = 1 TO 1
|     |     |           LETRA=SUBSTR(PQUE,I,1)
|     |     |           NLETRA=CHR(ASC(LETRA)+10)
|     |     |           PQUE=STUFF(PQUE,I,1,NLETRA)
|     |     +-------ENDFOR
&&Clave de Acceso
|     |     +-------FOR I=1 TO 8
|     |     |           LETRA=SUBSTR(PCONQUE1,I,1)
|     |     |           NLETRA=CHR(ASC(LETRA)+10)
|     |     |           PCONQUE1=STUFF(PCONQUE1,I,1,NLETRA)
|     |     +-------ENDFOR
&&Nombre de Usuario
|     |     +-------FOR I=1 TO 20
|     |     |           LETRA=SUBSTR(PQUIEN,I,1)
|     |     |           NLETRA=CHR(ASC(LETRA)+10)
|     |     |           PQUIEN=STUFF(PQUIEN,I,1,NLETRA)
|     |     +-------ENDFOR
&&Cargo en caso de ingresarlo
|     |     +-------FOR I=1 TO 25
|     |     |           LETRA=SUBSTR(PCARGO,I,1)
|     |     |           NLETRA=CHR(ASC(LETRA)+10)
|     |     |           PCARGO=STUFF(PCARGO,I,1,NLETRA)
```

```
|       |       +-------ENDFOR
&&Codigo en caso de ingresarlo
|       |       +-------FOR I=1 TO 10
|       |       |               LETRA=SUBSTR(PCODIGO,I,1)
|       |       |               NLETRA=CHR(ASC(LETRA)+10)
|       |       |               PCODIGO=STUFF(PCODIGO,I,1,NLETRA)
|       |       +-------ENDFOR
|       |
&&Se ubica el usuario en la informacion existente y se ereemplaza con la nueva informacion
                                LOCATE FOR V_AUXILIAR.QUIEN = PCHEQUIEN AND
                                V_AUXILIAR.CONQUE = PCHECONQUE
                                REPLACE V_AUXILIAR.QUIEN WITH PQUIEN, V_AUXILIAR.CONQUE WITH
                                PCONQUE1, V_AUXILIAR.QUE WITH PQUE, V_AUXILIAR.ACTIVO WITH
                                PACTIVO
&&Confrimamos la actualizacion de la vista
|       |               TABLEUPDATE(.T.)
|       |               THISFORM.TEXT1.VALUE = " "
|       |               THISFORM.TEXT2.VALUE = " "
|       |               THISFORM.TEXT4.VALUE = " "
|       |               THISFORM.SPINNER1.VALUE = 3
|       |
                                        MENSAJE = MESSAGEBOX(" La clave de"+" "+ALLTRIM(QQUIEN)+"
                                        "+" fue cambiada con éxito ",0+64+0," Modificar Usuarios ")
|       |               THISFORM.TEXT1.READONLY = .F.
|       |               THISFORM.CANCELAR.SETFOCUS
|       |
|       +-------CASE MENSAJE = 7
|       |
                                MENSAJE = MESSAGEBOX(" Usuario no Modificado ",0+64+0," Modificar
                                Usuarios ")
|       +-------ENDCASE
|
+-------ENDPROC

+-------PROCEDURE GOTFOCUS
&&Validamos que la informacion ingresada sea consistente
                        +-------IF LEN(ALLTRIM(THISFORM.TEXT1.VALUE))<4 OR
                        LEN(ALLTRIM(THISFORM.TEXT4.VALUE))<4
                        MENSAJE=MESSAGEBOX(" Verifique que el nombre o la clave del usuario que esta
                        registrando tenga 4 o mas caracteres ",0+48+0," Modificar Usuarios ")
|       |               THISFORM.TEXT1.SETFOCUS
|       |               THISFORM.ACEPTAR.ENABLED=.F.
|       +-------ENDIF
+-------ENDPROC
```

5.11.4 Digite el nombre de Usuario del usuario a modificar

```
+-------PROCEDURE VALID
```

```
&&Capturamos lo ingresado por el usuario en este objeto
|           PCLAVEACTUAL = THIS.VALUE
&&Si es diferente de vacio procedemos a encriptar el nombre de usuario
&&ingresado para realizar la busqueda a partir de esta informacion
|     +-------IF !EMPTY(PCLAVEACTUAL)
|     |
|     |     +-------FOR I=1 TO 20
|     |     |           LETRA = SUBSTR(PCLAVEACTUAL,I,1)
|     |     |           NLETRA = CHR(ASC(LETRA)+10)
|     |     |           PCLAVEACTUAL = STUFF(PCLAVEACTUAL,I,1,NLETRA)
|     |     +-------ENDFOR
|     |           PCHEQUIEN = PCLAVEACTUAL
|     |           REQUERY('v_auxiliar')
|     |           SELE V_AUXILIAR
&&Buscamos la existencia de la informacion en el archivo de usuarios
|     |           LOCATE FOR V_AUXILIAR.QUIEN = PCLAVEACTUAL
|     |     +-------IF !FOUND()
&&Si no la encuentra lo informamos
MENSAJE=MESSAGEBOX(" Nombre de usuario NO EXISTENTE ",0+64+0," INFORMACION
NUEVO IPC ")
|     |     |           RETURN 4
|     |     |           PMARCA1=.F.
|     |     +-------ELSE
&&En caso de encontrarse pero inactivo se ebrinda la alternativa de volverlo a activar
|     |     |     +-------IF V_AUXILIAR.ACTIVO = .F.
= MESSAGEBOX('El usuario solicitado se encuentra inactivo,'+CHR(13)+'tiene la alternativa de
Activarlo'+CHR(13)+'debe conocer la clave anterior para poder actualizar los
datos',0+64+0,'Informacion')
|     |     |     +-------ENDIF
&&En caso de encontrarse la informacion, cargamos variables con el resto de la informacion
&&encriptada
|     |     |           PCONQUEAUX = V_AUXILIAR.CONQUE
|     |     |           PCHECONQUE = V_AUXILIAR.CONQUE
|     |     |           PQUEAUX = V_AUXILIAR.QUE
|     |     |           PCARGOAUX = V_AUXILIAR.CARGO
|     |     |           PCODIGOAUX = V_AUXILIAR.CODIGO
|     |     |           PMARCA1 = .T.
|     |     |           PACTIVO = V_AUXILIAR.ACTIVO
|     |     |           THISFORM.TEXT1.READONLY = .T.
|     |     +-------ENDIF
|     +-------ENDIF
+-------ENDPROC
```

5.11.5 Escriba la actual clave de acceso

```
+-------PROCEDURE VALID
&&Cargamos en una variable el contenido en este objeto
|           PCONQUE = THIS.VALUE
```

```
&&Desencriptamos el contenido de la variable que contiene la clave de usuario
|       +-------FOR I = 1 TO 8
|       |               LETRA = SUBSTR(PCONQUEAUX,I,1)
|       |               NLETRA = CHR(ASC(LETRA)-10)
|       |               PCONQUEAUX = STUFF(PCONQUEAUX,I,1,NLETRA)
|       +-------ENDFOR
&&Desencriptamos el contenido de la variable que contiene el nivel de acceso
&&de usuario
|       +-------IF PCONQUE = PCONQUEAUX
|       |       +-------FOR I = 1 TO 1
|       |       |               LETRA = SUBSTR(PQUEAUX,I,1)
|       |       |               NLETRA = CHR(ASC(LETRA)-10)
|       |       |               PQUEAUX = STUFF(PQUEAUX,I,1,NLETRA)
|       |       +-------ENDFOR
|       |
|       |       +-------FOR I = 1 TO 25
|       |       |               LETRA = SUBSTR(PCARGOAUX,I,1)
|       |       |               NLETRA = CHR(ASC(LETRA)-10)
|       |       |               PCARGOAUX = STUFF(PCARGOAUX,I,1,NLETRA)
|       |       +-------ENDFOR
|       |
|       |       +-------FOR I = 1 TO 10
|       |       |               LETRA = SUBSTR(PCODIGOAUX,I,1)
|       |       |               NLETRA = CHR(ASC(LETRA)-10)
|       |       |               PCODIGOAUX = STUFF(PCODIGOAUX,I,1,NLETRA)
|       |       +-------ENDFOR
|       |
|       |               THISFORM.TEXT6.VALUE = PCODIGOAUX
|       |               THISFORM.TEXT5.VALUE = PCARGOAUX
|       |               THISFORM.SPINNER1.VALUE = VAL(PQUEAUX)
|       |               THISFORM.CHECK1.VALUE = PACTIVO
|       |               THISFORM.TEXT4.ENABLED=.T.
&&Si la clave ingresada por el usuario no corresponde al nombre de usuario
&&recien ingresado, lo informamos
|       +-------ELSE
                                MENSAJE = MESSAGEBOX(" La clave escrita no corresponde al Nombre de
                                Usuario!!",0+64+0," Información")
|       |
|       |               THISFORM.TEXT2.VALUE = " "
|       |               THISFORM.TEXT4.VALUE = " "
|       |               THISFORM.TEXT1.READONLY=.F.
|       |               RETURN 3
|       +-------ENDIF
+-------ENDPROC

+-------PROCEDURE GOTFOCUS
&&Segun los valores de las variables validamos que la informacion ingresada sea
&&consistente
```

```
|       +-------IF PMARCA1 = .F.
                        MENSAJE = MESSAGEBOX(" Información Incompleta !! ",0+48+0," Modificar
                        Usuarios ")
|       +-------ENDIF
|       +-------IF LEN(ALLTRIM(THISFORM.TEXT1.VALUE))<4
                        MENSAJE = MESSAGEBOX(" Verifique que el nombre del usuario que esta
                        modificando tenga 4 o mas caracteres ",0+48+0," Modificar Usuarios ")
|       |           THISFORM.TEXT1.SETFOCUS
|       +-------ENDIF
+-------ENDPROC
```

5.11.6 Escriba la Nueva Clave

```
+-------PROCEDURE LOSTFOCUS
&&Verificamos la informacion ingresada correspondiente a clave de usuario
|       +-------IF LEN(ALLTRIM(THIS.VALUE))<4
                        MENSAJE=MESSAGEBOX(" Verifique que la clave del usuario que esta
                        registrando tenga 4 o mas caracteres ",0+48+0," Modificar Usuarios ")
|       |           RETURN 5
|       |           THISFORM.ACEPTAR.ENABLED=.F.
|       +-------ENDIF
+-------ENDPROC
```

5.11.7 Nivel de Acceso

```
+-------PROCEDURE VALID
&&Evaluamos todos los datos ingresados del usuario a modificar
|       +-------IF LEN(ALLTRIM(THISFORM.TEXT1.VALUE))>=4 AND
LEN(ALLTRIM(THISFORM.TEXT2.VALUE))>=4 AND THIS.VALUE<>0
|       |           THISFORM.ACEPTAR.ENABLED=.T.
|       |
|       +-------ENDIF
+-------ENDPROC

+-------PROCEDURE GOTFOCUS
&&Validamos el dato ingresado para el nivel de acceso
|       +-------IF LEN(ALLTRIM(THISFORM.TEXT4.VALUE))<4
|       |           THISFORM.TEXT4.SETFOCUS
|       |           THISFORM.ACEPTAR.ENABLED=.F.
|       +-------ENDIF
+-------ENDPROC
```

5.12 FORMULARIO PARA MANTENIMIENTO GENERAL (MANTENIMIENTO.SCX)

5.12.1 Entorno de Datos

```
+-------PROCEDURE INIT
&&Configuramos el ambiente de trabajo para el manejo de archivos
|           SET SAFETY OFF
|
&&Creamos las indices y relaciones necesarias para que la captura se pueda hacer mas
optimamente
|           SELE VFAP
|           INDEX ON EXP_5 TAG FTEART
|
|           SELE V_IPCMART
|           INDEX ON CODART TAG CODART
|
|           SELE VFAP
|           SET RELA TO VFAP.CODART INTO V_IPCMART ADDITIVE
|
|           SELE V_RESPPA
|           INDEX ON CODFTE+CODART+CODESP TAG PKRESP
|
|           SELE V_IPCRESP
|           INDEX ON CODFTE+CODART+CODESP TAG PKRESP
|
|           SELE V_IPCMFTE
|           INDEX ON CODFTE TAG PKCODFTE
|
|           SELE V_IPCMFTE1
|           INDEX ON CODFTE TAG PKCODFTE
|
|           SELE VFAP
|
|           SELE V_RESPPA
|           INDEX ON CODFTE+CODART TAG NKRESP
|
|           SELE VFAP
|           SET RELA TO VFAP.CODFTE+VFAP.CODART INTO V_RESPPA ADDITIVE
|
+-------ENDPROC
```

5.12.2 Vistas

v_ipcmfte
v_ipcmart
v_ipcresp
v_ipcrfap
v_ipcrpart
vfap
v_ftepa
v_fappa

```
                v_fappart
                v_resppa
                v_fteartmvt
                v_ipctctv
                v_ftemvtpa
                v_oftepa
                v_ipcmfte1
                mto_mvto
                v_ipctmvt
                v_registro

+-------PROCEDURE ACTUALICELISTAS
&&Cargamos variables con el contenido de las listas
&&que contienen los articulos de la canasta y
&&que no estan asignado a la fuente actualpor un lado
&&y los articulos asignado a la fuente actual por otro lado
                LISTASELECCIONADA =
                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELE
                CTED
                LISTAFUENTE =
                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSOU
                RCE
&&Seleccionamos el archivo de fuentes y actualizamos el nombre de la fuente
                FAPPAFTE =
                ALLTRIM(THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALUE)
|           REQUERY('v_fappa')
|           SELECT V_FAPPA
|
|           SELE V_IPCMFTE
|           LOCATE FOR V_IPCMFTE.CODFTE = FAPPAFTE
                        THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.CNOMBREFUENTE.CONT
                        ROLSOURCE = 'v_ipcmfte.nombfte'
&&Consultamos los codigos y nombres del maestro de articulo
&&los articulos del archivo Fuente Articulos que corresponden a la
&&fuente solicitada
                SELECT V_IPCMART.CODART,V_IPCMART.NOMART FROM V_IPCMART,
                V_FAPPA WHERE ((V_FAPPA.CODART = V_IPCMART.CODART) AND
                (V_FAPPA.CODFTE = FAPPAFTE)) AND V_FAPPA.MARCA = .F. INTO CURSOR
                CANASTAFUENTE
|
|           SELECT CANASTAFUENTE
|           LISTASELECCIONADA.CLEAR
&&Los cargamos en la lista de articulos asignados
|     +-------SCAN
                        LISTASELECCIONADA.ADDITEM(CANASTAFUENTE.CODART+'
                        '+PROPER(CANASTAFUENTE.NOMART))
|     +-------ENDSCAN
|
```

```
|       +-------IF BETWEEN(SUBSTR(FAPPAFTE,17,4),'8887','8890')
&&Seleccionamos los articulos que no estan asignados a la fuente actual
&&para alimentar la otra lista
                        SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM V_IPCMART;
                        WHERE
                        (INLIST(V_IPCMART.CODART,'2210102','2220101','2220201','2220202','22202
                        03','7320101','7320102') AND V_IPCMART.CODART NOT IN (SELECT
                        DISTINCT CODART FROM CANASTAFUENTE)) INTO CURSOR
                        CANASTANACIONAL ORDER BY CODART
|       |        SELECT CANASTANACIONAL
|       |
|       |            LISTAFUENTE.CLEAR
|       |       +-------SCAN
                                    LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
                                    '+PROPER(CANASTANACIONAL.NOMART))
|       |       +-------ENDSCAN
|       +-------ELSE
                        +-------IF (BETWEEN(SUBSTR(FAPPAFTE,17,4),'8850','8886') OR
                        BETWEEN(SUBSTR(FAPPAFTE,17,4),'8891','8950')) AND;
                                SUBSTR(FAPPAFTE,1,2) = "11" AND SUBSTR(FAPPAFTE,1,2) =
                                ALLTRIM(DEPARTAMENTO)
&&Seleccionamos los articulos que no estan asignados a la fuente actual
&&para alimentar la otra lista
                                SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM
                                V_IPCMART;
                                WHERE (V_IPCMART.FORNAL = .T. AND V_IPCMART.CODART NOT
                                IN (SELECT DISTINCT CODART FROM CANASTAFUENTE)) INTO
                                CURSOR CANASTANACIONAL ORDER BY CODART
|       |       |       SELECT CANASTANACIONAL
|       |       |
|       |       |       LISTAFUENTE.CLEAR
|       |       |       +-------SCAN
                                    LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
                                    '+PROPER(CANASTANACIONAL.NOMART))
|       |       |       +-------ENDSCAN
|       |       +-------ELSE
&&Consultamos los codigos y nombres de los articulos que no estan asignado a la fuente actual
SELECT V_IPCMART.CODART,V_IPCMART.NOMART FROM V_IPCMART;
WHERE (V_IPCMART.CODART NOT IN (SELECT DISTINCT CODART FROM CANASTAFUENTE)
AND (V_IPCMART.FORNAL = .F. AND
!INLIST(V_IPCMART.CODART,'2210102','2220101','2220201','2220202','2220203','7320101','732010
2')));
INTO CURSOR CANASTANACIONAL
|       |       |
|       |       |               SELECT CANASTANACIONAL
|       |       |               LISTAFUENTE.CLEAR
&&y los cargamos en la lista de articulos disponibles, es decir, el resto de la canasta
|       |       |       +-------SCAN
```

```
LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
'+PROPER(CANASTANACIONAL.NOMART))
|       |       |       +-------ENDSCAN
|       |       +-------ENDIF
|       +-------ENDIF
|
|               USE
|               SELECT CANASTAFUENTE
|               USE
|
THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.CNOMBREFUENTE.CONTROLSOURCE="
+-------ENDPROC

+-------PROCEDURE ACTUALICEARTICULOS
&&Almacenamos en la variable el contenido de la lista donde se encuentran
&&los articulos que se recolectan en la fuente actual
|
LISTASELECCIONADA=THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1
.LSTSELECTED
&&Validamos la variable
|       +-------IF CAMBIO
|       |
                                ****Insertar los nuevos articulos que se han asociado a la fuente
|       |
&&Almacenamos en la variable el total de articulos seleccionados
&&en la lista
|       |               CUENTA=LISTASELECCIONADA.LISTCOUNT
&&Tomamos uno por uno de los articulos seleccionados para asignarle
&&alguna de las periodicidades permitidas segun el articulo
|       |       +-------FOR I=1 TO CUENTA
&&Tomamos el codigo del articulo
|       |       |               ELARTICULO=SUBSTR(LISTASELECCIONADA.LIST(I),1,7)
|       |       |
|       |       |               *_____________________________________________________
|       |       |               * Agrega articulos a la Fuente articulo *
|       |       |               *_____________________________________________________
&&Buscamos inicialmente si ya existe en fuente articulo la llave
&&(Fuente - articulo) actual
|       |       |               FAPPAFTE = VFAPCODFTE
|       |       |               FAPPART = ELARTICULO
|       |       |               REQUERY('v_fappa')
|       |       |               SELECT V_FAPPA
&&Si no lo encontramos consultamos las periodicidades asignadas
&&para este articulo en la maestro de articulos
|       |       |
|       |       |       +-------IF RECCOUNT()=0
|       |       |       |
```

```
SELECT CODPER,CODPART FROM V_IPCRPART, V_IPCMART WHERE (V_IPCRPART.CODPER
= V_IPCMART.PERART AND V_IPCMART.CODART = ELARTICULO) INTO CURSOR
PERIODICIDADES
|     |     |     |           SELE PERIODICIDADES
|     |     |     |           GO TOP
|     |     |     |           PERIODO=PERIODICIDADES.CODPART
|     |     |     |
&&Almacenamos en el archivo la informacion actual
INSERT INTO V_IPCRFAP (CODFTE,CODART,PERFAP);
VALUES(VFAPCODFTE,ELARTICULO,PERIODO)
|     |     |     |           TABLEUPDATE=(.T.)
|     |     |     +-------ELSE
&&En caso de encontrarse la informacion aun en el archivo pero con el valor del campo marca en .T.
&&habilitamos esta informacion para que se tenga en cuenta cambiandole el valor a .F.
|     |     |     |           SELE V_FAPPA
|     |     |     |           REPLACE V_FAPPA.MARCA WITH .F.
|     |     |     |           TABLEUPDATE=(.T.)
|     |     |     +-------ENDIF
|     |     |
|     |     |           *_____________________________________________________________
_________
|     |     |           * Agrega especificaciones a los articulos *
|     |     |           *_____________________________________________________________
_________
&&En el archivo de especificaciones buscamos tambien la coidncidencia
&&de la llave fuente - articulo
|     |     |           RESPAFTE = VFAPCODFTE
|     |     |           RESPART = ELARTICULO
|     |     |           REQUERY('v_resppa')
|     |     |           SELECT V_RESPPA
&&Si no los encontramos consultamos los diversos fuente - articulo que
&&coincidan con el nuevo fuente - articulo para tomar sus
&&especificaciones y asignarselas al nuevo
|     |     |     +-------IF RECCOUNT()=0
SELECT DISTINCT CODESP, NOMESP FROM V_IPCRESP WHERE (CODART = ELARTICULO)
GROUP BY CODESP INTO CURSOR ESPBASE
|     |     |     |           SELECT ESPBASE
&&Insertamos la informacion de las nueva fuente - articulo - especificacion
|     |     |     |     +-------SCAN
INSERT INTO V_RESPPA (CODFTE, CODART, CODESP, NOMESP, MARCA);
VALUES(VFAPCODFTE, ELARTICULO, ESPBASE.CODESP, ESPBASE.NOMESP, .F.)
|     |     |     |     +-------ENDSCAN
|     |     |     |           USE
|     |     |     +-------ELSE
&&En caso de encontrarse la informacion aun en el archivo pero con el valor del campo marca en .T.
&&habilitamos esta informacion para que se tenga en cuenta cambiandole el valor a .F.
REPLACE V_RESPPA.MARCA WITH .F. FOR V_RESPPA.CODFTE = VFAPCODFTE AND
V_RESPPA.CODART = ELARTICULO
```

```
|       |       |       +-------ENDIF
&&Confirmamos los cambios para que se almacenen
|       |       |               SELE V_RESPPA
|       |       |               TABLEUPDATE(.T.)
|       |       |               SELE V_IPCRESP
|       |       |               TABLEUPDATE(.T.)
|       |       +-------ENDFOR
|       |
|       |               CAMBIO=.F.
|       +-------ENDIF
+-------ENDPROC
+-------PROCEDURE INIT
&&Inactivamos el menu del aplicativo, definimos variables
|               SET SYSMENU OFF
|
&&Declaramos variables a utilizar
|               PUBLIC UNAVEZ, TVEZ, REGUNAVEZ, NUEVOVEZ, REGNUEVO
|               PUBLIC BORRAFAP, BORRAESP, PTROFTE, PASOFTE, CODMAX
|
|               STORE .F. TO UNAVEZ, BORRAFAP, BORRAESP
|               STORE 0 TO TVEZ, NUEVOVEZ, REGUNAVEZ, REGNUEVO
|               STORE " " TO PASOFTE, PTROFTE
|
|               PUBLIC CAMBIO
|
|               PUBLIC FTEPAFTE
|               PUBLIC FAPPAFTE, FAPPART
|               PUBLIC RESPAFTE, RESPART
|               PUBLIC OFTEPAFTE, ORESPAFTE
                        STOR '' TO FTEPAFTE, FAPPAFTE, FAPPART, RESPAFTE, RESPART,
                        OFTEPAFTE, ORESPAFTE
+-------ENDPROC

+-------PROCEDURE DESTROY
&&Al salir del formulario actualizamos el archivo de registro para saber
&&que este usuario ya salio de este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|               REQUERY('v_registro')
|               SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para indicar
&&que la cerro en este momento
                        LOCATE FOR
                        ALLTRIM(V_REGISTRO.QUIEN)+ALLTRIM(V_REGISTRO.NPANTALLA) =
                        ALLTRIM(PQUIEN)+WONTOP( )
|       +-------IF FOUND()
|       |               REPLACE V_REGISTRO.ACT_INA WITH .F.
&&Actualizamos la informacion y tambien el archivo
```

```
|       |           TABLEUPDATE(.T.)
|       +-------ENDIF
+-------ENDPROC

+-------PROCEDURE UNLOAD
&&Habilitamos nuevamente el menu del aplicativo
|           SET SYSMENU ON
+-------ENDPROC

+-------PROCEDURE ACTIVATE
&&Al entrar al formulario actualizamos el archivo de registro para saber
&&que este usuario abrio este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para saber
&&si ya esta relacionado en el archivo y no voilver a registrarlo
                        LOCATE FOR (ALLTRIM(V_REGISTRO.QUIEN) + NPANTALLA =
                        ALLTRIM(PQUIEN) + WONTOP()) AND V_REGISTRO.ACT_INA = .T.
&&si no es encontrado lo registramos y actualizamos el archivo
|       +-------IF !FOUND()
                        INSERT INTO V_REGISTRO(QUIEN, NPANTALLA, FECHAR, ACT_INA);
|       |                 VALUES(PQUIEN,WONTOP( ), DATE(), .T.)
|       |           TABLEUPDATE(.T.)
|       +-------ENDIF
+-------ENDPROC
```

5.13 MANTENIMIENTO FUENTE

```
+-------PROCEDURE PAGE1.DEACTIVATE
&&Validamos si la informacion del codigo fuente que se estaba trabajando
&&sea consistente, de lo contrario limpiamos la variable
|       +-------IF LEN(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE))=20
                        PASOFTE = ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE)
|       +-------ELSE
|       |           PASOFTE = ''
|       +-------ENDIF
+-------ENDPROC

+-------PROCEDURE PAGE1.ACTIVATE
&&Verificamos el valor de la variable pasofte, puesto que ésta almacenara
&&el codigo de fuente que se este trabajando en otra de las pagina de esta pantalla
&&para traer automaticamente el codigo de la fuente y no tener que escribirlo cada vez
|       +-------IF EMPTY(PASOFTE)
&&Si no trae nada mostramos los 5 primeros difgitos del departamento con el municipio
```

```
|       |
THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE=ALLTRIM(DEPARTAMENTO)+ALLTRIM(MU
NICIPIO)
|       |               THISFORM.PAGEFRAME1.PAGE1.ECODFTE.REFRESH
|       |               THISFORM.PAGEFRAME1.PAGE1.ECODFTE.SETFOCUS
&&Limpiamos los demas campos
|       |
|       |               THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE=" "
|       |               THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
|       |               THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE=" "
|       |               THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
|       |               THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE=" "
|       |               THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
|       |               THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
|       |               THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
|       |               THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
|       |               THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
|       |
|       +-------ELSE
&&Si trae informacionla mostramos en el campo correspondiente
|       |               THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE=PASOFTE
|       |               THISFORM.PAGEFRAME1.PAGE1.ECODFTE.REFRESH
|       |               THISFORM.PAGEFRAME1.PAGE1.ECODFTE.SETFOCUS
|       +-------ENDIF
|
|
|
|
+-------ENDPROC

+-------PROCEDURE PAGE2.DEACTIVATE
&&Validamos las periodicidades asignadas a los nuevos articulos
&&para ver si son las permitidas para los mismos
|       +-------IF !EMPTY(VFAPCODART1)
|       |               LOCAL PER1,PER2,PER3,PER4,m.PERFAP
|       |               STORE " " TO PER1,PER2,PER3,PER4
M.PERFAP =
THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFTE.VALUE
SELECT CODPER, CODPART FROM V_IPCRPART, V_IPCMART WHERE
(V_IPCRPART.CODPER = V_IPCMART.PERART AND V_IPCMART.CODART = VFAPCODART1)
INTO CURSOR PERIODICIDADES
|       |               SELE PERIODICIDADES
|       |               GO TOP
|       |       +-------IF PERIODICIDADES.CODPER=0
|       |       |               PER1="0"
|       |       +-------ELSE
|       |       |       +-------IF PERIODICIDADES.CODPER=1
|       |       |       |               PER1="1"
```

```
|     |     |     |           PER2="2"
|     |     |     +-------ELSE
|     |     |     |     +-------IF PERIODICIDADES.CODPER=2
|     |     |     |     |           PER1="3"
|     |     |     |     |           PER2="4"
|     |     |     |     |           PER3="5"
|     |     |     |     +-------ELSE
|     |     |     |     |     +-------IF PERIODICIDADES.CODPER=3
|     |     |     |     |     |           PER1="6"
|     |     |     |     |     |           PER2="7"
|     |     |     |     |     |           PER3="8"
|     |     |     |     |     |           PER4="9"
|     |     |     |     |     +-------ENDIF
|     |     |     |     +-------ENDIF
|     |     |     +-------ENDIF
|     |     +-------ENDIF
|     |
|     |
|     |           SELE PERIODICIDADES
|     |     +-------SCAN
|     |     |           LOCATE FOR PERIODICIDADES.CODPART = m.PERFAP
|     |     |     +-------IF !FOUND()
MENSAJE=MESSAGEBOX(" Periodicidades permitidas:
"+ALLTRIM(PER1)+','+ALLTRIM(PER2)+','+ALLTRIM(PER3)+','+ALLTRIM(PER4)+', Corrija por
favor... ',0+64+0," INFORMACION ")

THISFORM.PAGEFRAME1.PAGE2.BFINALIZAR.ENABLED=.F.
|     |     |     |           THISFORM.PAGEFRAME1.TABS = .F.
|     |     |     |           EXIT
|     |     |     +-------ELSE
THISFORM.PAGEFRAME1.PAGE2.BFINALIZAR.ENABLED=.T.
|     |     |     |           THISFORM.PAGEFRAME1.TABS = .T.
                                          +-------IF
                                          LEN(ALLTRIM(THISFORM.PAGEFRAME1.PAGE2.CONT
                                          AINER2.ECODFTE.VALUE))=20
PASOFTE = ALLTRIM(THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALUE)
|     |     |     |       +-------ELSE
|     |     |     |     |           PASOFTE = ''
|     |     |     |       +-------ENDIF
|     |     |     |       EXIT
|     |     |     +-------ENDIF
|     |     +-------ENDSCAN
|     +-------ENDIF
|
+-------ENDPROC

+-------PROCEDURE PAGE2.ACTIVATE
&&Definimos e inicializamos variables
```

```
|           PUBLIC VFAPCODART1
|           VFAPCODART1=" "
|           CAMBIO=.T.
|           REQUERY('v_ipcmfte1')
|           SELECT V_IPCMFTE1
&&Verificamos el valor de la variable pasofte, puesto que ésta almacenara
&&el codigo de fuente que se este trabajando en otra de las pagina de esta pantalla
&&para traer automaticamente el codigo de la fuente y no tener que escribirlo cada vez
|      +-------IF EMPTY(PASOFTE)
&&Al desplegable de nombres de fuentes lo ubicamos en la primera fuente
                THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.CNOMBREFUENTE.LISTITEMID=
                1
|      |
|      |          THISFORM.LOCKSCREEN=.T.
|      |
                LISTASELECCIONADA=THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.
                MOVERLISTA1.LSTSELECTED
                |      |
                LISTAFUENTE=THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLI
                STA1.LSTSOURCE
&&Cargamos las variable con los contenidos de las listas tanto de
&&articulos asignados a la fuente actual como los disponibles, es decir,
&&el resto de la canasta
|      |
&&Consultamos los articulos que corresponden a la fuente actual
&&y los insertamos en la lista de articulos seleccionados
                FAPPAFTE =
                ALLTRIM(THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALUE)
|      |          REQUERY('v_fappa')
|      |
                SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM V_IPCMART,
                V_FAPPA WHERE ((V_FAPPA.CODART = V_IPCMART.CODART) AND
                (V_FAPPA.CODFTE = FAPPAFTE)) AND  V_FAPPA.MARCA = .F. INTO CURSOR
                CANASTAFUENTE
|      |  SELECT CANASTAFUENTE
|      |  LISTASELECCIONADA.CLEAR
&&Insertamos uno por uno los articulos
|      |      +-------SCAN
                                        LISTASELECCIONADA.ADDITEM(CANASTAFUENTE.CODART
                                        +' '+PROPER(CANASTAFUENTE.NOMART))
|      |      +-------ENDSCAN
|      |
&&Ordenamos la lista
|      |          LISTASELECCIONADA.LISTITEMID=1
|      |
&&Seleccionamos los articulos que no estan asignados a la fuente actual
&&para alimentar la otra lista
```

```
                                SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM
                                V_IPCMART;
                                WHERE (V_IPCMART.CODART NOT IN (SELECT DISTINCT CODART
                                FROM CANASTAFUENTE)) INTO CURSOR CANASTANACIONAL
                                ORDER BY CODART
|     |           SELECT CANASTANACIONAL
|     |
|     |           LISTAFUENTE.CLEAR
|     |     +-------SCAN
                                        LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
                                        '+PROPER(CANASTANACIONAL.NOMART))
|     |     +-------ENDSCAN
|     |           USE
|     |
|     |           SELECT V_IPCMFTE1
                                THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.NAVIGATOR1.ENAB
                                LEDISABLEBUTTONS
|     |
&&Actualizamos la informacion en pantalla
                          +-------WITH
                          THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1
|     |     |           .NOSELECCIONADOS=LISTASELECCIONADA.LISTCOUNT
|     |     |           .NODISPONIBLES=LISTAFUENTE.LISTCOUNT
|     |     |           .LDISPONIBLES.CAPTION=STR(.NODISPONIBLES,5,0)
                                        .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0
                                )
|     |     +-------ENDWITH
|     |
|     |           THISFORM.LOCKSCREEN=.F.
|     +-------ELSE
&&Si trae informacion la variable, reconsultamos la informacion correspondiente a la fuente
&&a ver si tiene articulos relacionados
|     |           OFTEPAFTE = PASOFTE
|     |           REQUERY('v_oftepa')
|     |           SELE V_OFTEPA
&&Si tiene articulos asignados mostramos la informacion basica de esta fuente
|     |     +-------IF RECCOUNT() > 0
|     |     |           THISFORM.LOCKSCREEN=.T.
|     |     |           =SEEK(OFTEPAFTE,'v_ipcmfte1','pkcodfte')
THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALUE = V_IPCMFTE1.CODFTE
THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.CNOMBREFUENTE.CONTROLSOURCE =
'v_ipcmfte1.nombfte'
&&Cargamos las variable con los contenidos de las listas tanto de
&&articulos asignados a la fuente actual como los disponibles, es decir,
&&el resto de la canasta
        LISTASELECCIONADA=THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVER
LISTA1.LSTSELECTED
```

```
        LISTAFUENTE=THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.
        LSTSOURCE
&&Consultamos los articulos que corresponden a la fuente actual
&&y los insertamos en la lista de articulos seleccionados

&&Traemos los articulos asignados a la fuente en cuestión y que tengan el campo marca en falso, es
decir,
&&no esten marcados para eliminar
FAPPAFTE = ALLTRIM(THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALUE)
REQUERY('v_fappa')
SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM V_IPCMART, V_FAPPA WHERE
((V_FAPPA.CODART = V_IPCMART.CODART)AND(V_FAPPA.CODFTE = FAPPAFTE)) AND
V_FAPPA.MARCA = .F. INTO CURSOR CANASTAFUENTE
|     |     |
|     |     |           SELECT CANASTAFUENTE
|     |     |           LISTASELECCIONADA.CLEAR
&&Insertamos uno por uno los articulos para mostrarlos en la lista correspondiente
|     |     |     +-------SCAN
                              LISTASELECCIONADA.ADDITEM(CANASTAFUENTE.CODART+'
                        '+PROPER(CANASTAFUENTE.NOMART))
|     |     |     +-------ENDSCAN
|     |     |
&&Ordenamos la lista
|     |     |           LISTASELECCIONADA.LISTITEMID=1
&&Validamos para el caso de los codigo de fuentes reservados para  los Servicios Publicos
&&solamente se puedan ver los articulos prpios para este grupo, unicamente, es decir, que no vean
alimentos, etc
&&para que no se les puedan a signar a estas fuentes
                                          +-------IF
                                          BETWEEN(SUBSTR(ALLTRIM(PASOFTE),17,4),'8887','8
                                          890')
&&Seleccionamos los articulos que no estan asignados a la fuente actual
&&para alimentar la otra lista
        SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM V_IPCMART;
        WHERE
(INLIST(V_IPCMART.CODART,'2210102','2220101','2220201','2220202','2220203','7320101','732010
2') AND V_IPCMART.CODART NOT IN (SELECT DISTINCT CODART FROM CANASTAFUENTE))
INTO CURSOR CANASTANACIONAL ORDER BY CODART
|     |     |     |       SELECT CANASTANACIONAL
|     |     |     |
|     |     |     |       LISTAFUENTE.CLEAR
|     |     |     |     +-------SCAN
                              LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
                        '+PROPER(CANASTANACIONAL.NOMART))
|     |     |     |     +-------ENDSCAN
|     |     |     +-------ELSE
&&Validamos pa los codigos de fuentes reservados para manejar Formacion Nacional
&&solamente las puedan trabajar en bogota, esto se hace verificando el contenido de la variable
```

```
&&departamento y los dos primeros digitos de las fuente existentes en el archivo maestro de fuentes
&&Si es bogota permitimos que las trabajen, de lo contrario no
                                        +-------IF
                                        (BETWEEN(SUBSTR(ALLTRIM(PASOFTE),17,4),'8850','
                                        8886') OR
                                        BETWEEN(SUBSTR(ALLTRIM(PASOFTE),17,4),'8891','8
                                        950')) AND;
                                        SUBSTR(FAPPAFTE,1,2) = "11" AND
                                        SUBSTR(FAPPAFTE,1,2) = ALLTRIM(DEPARTAMENTO)
&&Seleccionamos los articulos que no estan asignados a la fuente actual
&&para alimentar la otra lista
SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM V_IPCMART;
WHERE (V_IPCMART.FORNAL = .T. AND V_IPCMART.CODART NOT IN (SELECT DISTINCT
CODART FROM CANASTAFUENTE)) INTO CURSOR CANASTANACIONAL ORDER BY CODART
|     |     |     |     | SELECT CANASTANACIONAL
|     |     |     |     |
|     |     |     |     | LISTAFUENTE.CLEAR
|     |     |     |     | +-------SCAN
                                LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
                        '+PROPER(CANASTANACIONAL.NOMART))
|     |     |     |     | +-------ENDSCAN
|     |     |     |     +-------ELSE

&&Seleccionamos los articulos que no estan asignados a la fuente actual
&&para alimentar la otra lista
        SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM V_IPCMART;
        WHERE (V_IPCMART.CODART NOT IN (SELECT DISTINCT CODART FROM
CANASTAFUENTE) AND (V_IPCMART.FORNAL = .F. AND;
!INLIST(V_IPCMART.CODART,'2210102','2220101','2220201','2220202','2220203','7320101','732010
2')));
 INTO CURSOR CANASTANACIONAL ORDER BY CODART
|     |     |     |     | SELECT CANASTANACIONAL
|     |     |     |     |
|     |     |     |     | LISTAFUENTE.CLEAR
|     |     |     |     | +-------SCAN
                        LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
                  '+PROPER(CANASTANACIONAL.NOMART))
|     |     |     |     |     +-------ENDSCAN
|     |     |     |     +-------ENDIF
|     |     |     +-------ENDIF
|     |     +-------ENDIF
|     |
|     |           SELECT V_IPCMFTE1
|     |
THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.NAVIGATOR1.ENABLEDISABLEBUTTONS
&&Actualizamos la informacion en pantalla
```

```
                                +-------WITH
                                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLIS
                                TA1
                                +-------ENDWITH
|       |             THISFORM.LOCKSCREEN=.F.
|       +-------ENDIF
|
&&Refrescamos informacion en pantalla
                    THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFT
                    E.FORECOLOR=RGB(255,255,255)
                    THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EDECA.F
                    ORECOLOR=RGB(255,255,255)
                    THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT2.F
                    ORECOLOR=RGB(255,255,255)
                    THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT3.F
                    ORECOLOR=RGB(255,255,255)
|
&&Refrescamos pantalla
                    THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.CNOMBREFUENTE.CONTROLSO
                    URCE = ''
|
|         THISFORM.REFRESH
+-------ENDPROC

+-------PROCEDURE PAGE3.DEACTIVATE
&&Validamos la consistencia de l valor de la variable que lleva el codigo
&&de fuente actualmente trabajado
|       +-------IF LEN(ALLTRIM(THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE))=20
PASOFTE = ALLTRIM(THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE)
|       +-------ELSE
|       |             PASOFTE = ''
|       +-------ENDIF
|
+-------ENDPROC

+-------PROCEDURE PAGE3.ACTIVATE
&&Reconsultamos la información
|             REQUERY('v_ipcmfte1')
|             SELECT V_IPCMFTE1
&&Validamosa el contenido de la variable para mostrar la información correspondiente
|       +-------IF EMPTY(PASOFTE)
                            THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE =
                            V_IPCMFTE1.CODFTE
|       |         THISFORM.PAGEFRAME1.PAGE3.ECODFTE.REFRESH
|       |         FAPPAFTE = V_IPCMFTE1.CODFTE
|       |         REQUERY('v_fappa')
|       |         SELE V_FAPPA
&&A una variable le cargamos el contenido del combo donde estan los codigos
```

```
&&de los articulos
|     |          LISTASELECCIONADA = THISFORM.PAGEFRAME1.PAGE3.COMBO1
&&Extraemos los articulos correspondientes a la fuente actual y los cargamos en el combo
|     |          SELECT CODART;
|     |                FROM V_FAPPA;
                         WHERE (V_FAPPA.CODFTE =
                 THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE AND V_FAPPA.MARCA = .F.);
|     |               INTO CURSOR ARTICULOS
|     |
|     |          SELECT ARTICULOS
|     |          LISTASELECCIONADA.CLEAR
|     |          THISFORM.PAGEFRAME1.PAGE3.COMBO1.VALUE = " "
&&Cargamos la informacion correspondiente a los articulos que se cotizan en esta fuente
|     |     +-------SCAN
|     |     |          LISTASELECCIONADA.ADDITEM(ARTICULOS.CODART)
|     |     +-------ENDSCAN
|     +-------ELSE
&&reconsultamos la información
|     |          OFTEPAFTE = PASOFTE
|     |          REQUERY('v_oftepa')
|     |          SELE V_OFTEPA
|     |     +-------IF RECCOUNT() > 0
&&Si la variable tiene informacion (codigo fuente) mostramos la informacion de la misma
THISFORM.PAGEFRAME1.PAGE3.CONTAINER2.CNOMBREFUENTE.VALUE =
V_OFTEPA.NOMBFTE
THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE = V_OFTEPA.CODFTE
&&Cargamos a un variable el valor del comobo para luego cargarlo con los articulos
&&q eue se cotizan en esta fuente
LISTASELECCIONADA = THISFORM.PAGEFRAME1.PAGE3.COMBO1
|     |     |          FAPPAFTE = PASOFTE
|     |     |          REQUERY('v_fappa')
|     |     |          SELE V_FAPPA
&&Llevamos a cabo el cargue de la información, seleccionando los que tiene marca = .F., es decir,
que
&&no estan marcados para eliminar
|     |     |          SELECT CODART;
|     |     |            FROM V_FAPPA;
                 WHERE (V_FAPPA.CODFTE =
        THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE AND V_FAPPA.MARCA = .F.);
|     |     |            INTO CURSOR ARTICULOS
|     |     |
|     |     |          SELECT ARTICULOS
|     |     |          LISTASELECCIONADA.CLEAR
|     |     |          THISFORM.PAGEFRAME1.PAGE3.COMBO1.VALUE=" "
|     |     |     +-------SCAN
                              LISTASELECCIONADA.ADDITEM(ARTICULOS.CODART)
|     |     |     +-------ENDSCAN
|     |     +-------ENDIF
```

```
|       +-------ENDIF
&&Configuramos como se mostrara la informacion que proviene de campos que aceptan valores
nulos
|               SET NULLDISPLAY TO [ ]
|               SELECT V_IPCMFTE1
                        THISFORM.PAGEFRAME1.PAGE3.CONTAINER2.NAVIGATOR1.ENABLEDIS
                        ABLEBUTTONS
|               THISFORM.PAGEFRAME1.PAGE3.ECODFTE.SETFOCUS
|
+-------ENDPROC

+-------PROCEDURE PAGE4.DEACTIVATE
&&Verificamos la consistencia de la informacion de la variable
|       +-------IF LEN(ALLTRIM(THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE))=20
|       |               PASOFTE = ALLTRIM(THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE)
|       +-------ELSE
|       |               PASOFTE = ''
|       +-------ENDIF
|
+-------ENDPROC

+-------PROCEDURE PAGE4.ACTIVATE
&&Habilitamos e inhabilitamos objetos segun sea el caso
|               THISFORM.PAGEFRAME1.PAGE4.COMBO1.VISIBLE=.T.
&&Validamos el valor de la variable a para consultarle
|       +-------IF EMPTY(PASOFTE)
&&Si esta en blanco mostramos la primera fuente del maestro de las mismas
|       |               SELECT V_IPCMFTE
|       |               THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE = V_IPCMFTE.CODFTE
THISFORM.PAGEFRAME1.PAGE4.TEXT9.VALUE = V_IPCMFTE.NOMBFTE
|       +-------ELSE
&&Si trae algun valor mostramos su información en pantalla
|       |               OFTEPAFTE = PASOFTE
|       |               REQUERY('v_oftepa')
|       |               SELE V_OFTEPA
|       |       +-------IF RECCOUNT() > 0
THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE = V_OFTEPA.CODFTE
THISFORM.PAGEFRAME1.PAGE4.TEXT9.VALUE = V_OFTEPA.NOMBFTE
|       |       +-------ENDIF
|       +-------ENDIF
&&Traemos la informacion que tiene (si es que la tiene) en captura
|               PTROFTE = THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE
|               REQUERY('mto_mvto')
|               THISFORM.PAGEFRAME1.PAGE4.GRID1.REFRESH
|               THISFORM.PAGEFRAME1.PAGE4.TEXT1.SETFOCUS
+-------ENDPROC

+-------PROCEDURE PAGE5.ACTIVATE
```

```
&&Definimos e inicializamos variables
|           CAMBIO=.T.
&&Cargamos las variables con el contenido de las listas
LISTAFUENTE =
THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSOURCE
LISTASELECCIONADA =
THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELECTED
&&Reconsultamnos la informacion de las vistas para obtener la informacioon mas actualizada
|           REQUERY('v_ipcmfte1')
|           REQUERY('v_ipcmfte')
|           SELECT V_IPCMFTE
&&Bloqueamos la pantalla mientras se carga la informacion
|           THISFORM.LOCKSCREEN=.T.
|      +-------IF EMPTY(PASOFTE)
&&Al desplegable de nombres de fuentes lo ubicamos en la primera fuente
         THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.CNOMBREFUENTE1.LISTITEMID=1
|      |
&&Consultamos los articulos que corresponden a la fuente actual
&&y los insertamos en la lista de articulos seleccionados
                 FAPPAFTE =
         ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VALUE)
|      |           REQUERY('v_fappa')
|      |
|      |           SELECT V_IPCMART.CODART, V_IPCMART.NOMART;
|      |                FROM V_IPCMART, V_FAPPA;
|                         WHERE ((V_FAPPA.CODART = V_IPCMART.CODART) AND
                 (V_FAPPA.CODFTE = FAPPAFTE));
|      |                AND  V_FAPPA.MARCA = .F.;
|      |                INTO CURSOR CANASTAFUENTE
|      |
|      |           SELECT CANASTAFUENTE
|      |           LISTASELECCIONADA.CLEAR
&&Insertamos uno por uno los articulos
|      |      +-------SCAN
                         LISTASELECCIONADA.ADDITEM(CANASTAFUENTE.CODART+'
                 '+PROPER(CANASTAFUENTE.NOMART))
|      |      +-------ENDSCAN
|      |
&&Ordenamos la lista
|      |           LISTASELECCIONADA.LISTITEMID=1
&&Cargamos la variable para utilizarla como parametro para la vista
&&poderla reconsultar
                                 FAPPAFTE =
                                 ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE
                                 .VALUE)
|      |           REQUERY('v_fappa')
&&A partir del resultado traemos los datos de los articulos que cotizan en esta fuente
&&y que no esten marcados para eliminar
```

```
|     |          SELECT V_IPCMART.CODART, V_IPCMART.NOMART;
|     |               FROM V_IPCMART, V_FAPPA;
                         WHERE ((V_FAPPA.CODART = V_IPCMART.CODART) AND
                   (V_FAPPA.CODFTE = FAPPAFTE));
|     |               AND  V_FAPPA.MARCA = .F.;
|     |               INTO CURSOR CANASTASELECCIONADA
|     |
|     |          SELECT CANASTASELECCIONADA
|     |          LISTAFUENTE.CLEAR
&&Insertamos uno por uno los articulos
|     |     +-------SCAN
                         LISTAFUENTE.ADDITEM(CANASTASELECCIONADA.CODART+'
                   '+PROPER(CANASTASELECCIONADA.NOMART))
|     |     +-------ENDSCAN
|     |
&&Ordenamos la lista
|     |          LISTAFUENTE.LISTITEMID=1
|     +-------ELSE
&&Si la variable trae un codigo de fuente lo ultilizamos como parametro para reconsultar
&&la informacion de ella misma
|     |          OFTEPAFTE = PASOFTE
|     |          REQUERY('v_oftepa')
|     |          SELE V_OFTEPA
&&Si obtuvimos resultados mostramos la informacion concerniente
|     |     +-------IF RECCOUNT() > 0
|     |     |          SELE V_IPCMFTE
|     |     |          LOCATE FOR V_IPCMFTE.CODFTE = OFTEPAFTE
THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VALUE = V_IPCMFTE.CODFTE
THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.CNOMBREFUENTE1.CONTROLSOURCE =
'v_ipcmfte.nombfte'

&&Consultamos los articulos que corresponden a la fuente actual
&&y los insertamos en la lista de articulos seleccionados
|     |     |
                         FAPPAFTE =
                   ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VALUE)
|     |     |          REQUERY('v_fappa')
|     |     |          SELECT V_IPCMART.CODART, V_IPCMART.NOMART;
|     |     |          FROM V_IPCMART, V_FAPPA;
                         WHERE ((V_FAPPA.CODART =
                   V_IPCMART.CODART)AND(V_FAPPA.CODFTE = FAPPAFTE));
|     |     |               AND V_FAPPA.MARCA = .F.;
|     |     |               INTO CURSOR CANASTASELECCIONADA
|     |     |
|     |     |          SELECT CANASTASELECCIONADA
|     |     |          LISTASELECCIONADA.CLEAR
&&Insertamos uno por uno los articulos
|     |     |     +-------SCAN
```

```
LISTASELECCIONADA.ADDITEM(CANASTASELECCIONADA.CODART+'
'+PROPER(CANASTASELECCIONADA.NOMART))
|     |     |     +-------ENDSCAN
|     |     |
&&Ordenamos la lista
|     |     |           LISTASELECCIONADA.LISTITEMID=1
|     |     +-------ENDIF
|     +-------ENDIF
&&Habilitamos las barras de botones cada una a partir de una vista
|           SELECT V_IPCMFTE
                        THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.NAVIGATOR1.ENABLEDIS
                        ABLEBBUTTONS
|           SELECT V_IPCMFTE1
|
THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.NAVIGATOR2.ENABLEDISABLEBUTTONS
|
&&Actualizamos la informacion en pantalla
|     +-------WITH THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1
|     |           .NOSELECCIONADOS=LISTASELECCIONADA.LISTCOUNT
|     |           .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0)
|     +-------ENDWITH
|
&&Refrescamos pantalla
|           LISTASELECCIONADA.SETFOCUS
|
        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFTE.FOR
        ECOLOR=RGB(255,255,255)

        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EDECA.FOREC
        OLOR=RGB(255,255,255)

        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT2.FORECO
        LOR=RGB(255,255,255)

        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT3.FORECO
        LOR=RGB(255,255,255)
&&Desbloquemos pantalla al terminar de procesar
|           THISFORM.LOCKSCREEN=.F.
|           THISFORM.REFRESH
+-------ENDPROC
```

5.13.1 Boton Salir

```
+-------PROCEDURE CLICK
&&Salir de esta pantalla
|           THISFORM.RELEASE
+-------ENDPROC
```

5.13.2 Boton Cancelar

```
+-------PROCEDURE CLICK
&&Reinicializamos los valores de los objetos en caso de decidir ingresar de nuevo
&&la informacion de una fuente
                        THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE =
                        ALLTRIM(DEPARTAMENTO)+ALLTRIM(MUNICIPIO)
|       THISFORM.PAGEFRAME1.PAGE1.ECODFTE.REFRESH
|       THISFORM.PAGEFRAME1.PAGE1.ECODFTE.SETFOCUS
|
|       THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE=''
|       THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
|
|       THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE=''
|       THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
|
|       THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE=''
|       THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
|
|       THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
|       THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
|
|       THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
|       THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
|
|       THISFORM.PAGEFRAME1.PAGE1.GUARDAR.ENABLED=.F.
+-------ENDPROC
```

5.13.3 Boton Guardar

```
+-------PROCEDURE CLICK
&&Validamos el codigo de la fuente original digitado, que el departamento digitado pertenezca a los
ya definidos
|       SELE V_FTEPA
                        +-------IF
                !INLIST(SUBSTR(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE,1,2),'05','08','
                11','13','17','23','41','50','52','54','66','68','76')
                        MESSAGEBOX(" El código del Departamento no es válido, corrija por favor..
                        ",0+48+0,"ADVERTENCIA")

&&Volvemos de solo lectura los objetos
|   |       THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE=''
|   |       THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE=''
|   |       THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE=''
|   |       THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
|   |       THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
|   |
|   |       THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.T.
```

```
|     |          THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.T.
|     |          THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.T.
|     |          THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.T.
|     |          THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.T.
|     |
|     |          THISFORM.PAGEFRAME1.PAGE1.GUARDAR.ENABLED=.F.
&&Volvemos a dejar el objeto donde debe ir codigo de la fuente
&&original con los primero 5 digitos
               THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE=ALLTRIM(DEPARTAMENTO)
               +ALLTRIM(MUNICIPIO)
|     | THISFORM.PAGEFRAME1.PAGE1.ECODFTE.SETFOCUS
|     |
|     +-------ELSE
|     |
                      +-------IF
                      LEN(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE))<20
&&En caso de no ser aceptable el codigo digitado
                              WAIT WIND " Recuerde que el codigo de la fuente debe tener 20 digitos
                              " NOWAIT
&&Volvemos de solo lectura los objetos
|     |     |          THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE=''
|     |     |          THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE=''
|     |     |          THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE=''
|     |     |          THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
|     |     |          THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
|     |     |
|     |     |          THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.T.
|     |     |          THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.T.
|     |     |          THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.T.
|     |     |          THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.T.
|     |     |          THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.T.
|     |     |
|     |     |          THISFORM.PAGEFRAME1.PAGE1.GUARDAR.ENABLED=.F.
&&Volvemos a dejar el objeto donde debe ir codigo de la fuente
&&original con los primero 5 digitos
               THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE=ALLTRIM(DEPARTAMENTO)
               +ALLTRIM(MUNICIPIO)
|     | THISFORM.PAGEFRAME1.PAGE1.ECODFTE.SETFOCUS
|     | +-------ELSE
&&Si es correcto el codigo digitado, que el codigo de fuente tenga los 20 digitos
                              +-------IF
                              LEN(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE))
                              =20
&&que el consecutivo digitado se diferente de cero
                                          +-------IF
                                          SUBSTR(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.E
                                          CODFTE.VALUE),17,4)<>"0000"
&&Si son consecutivos reservados verificamos que sea en bogota donde los estan trabajando
```

```
                                                        +-------IF
                                                        SUBSTR(ALLTRIM(THISFORM.PAGEFRAME1.P
                                                        AGE1.ECODFTE.VALUE),1,2)="11" AND
                                                        SUBSTR(ALLTRIM(THISFORM.PAGEFRAME1.P
                                                        AGE1.ECODFTE.VALUE),1,2)=ALLTRIM(DEPAR
                                                        TAMENTO)
&&Verificamos si el consecutivo pertenecece al renago de reservados, ya sea para Servicios Publicos
&&o para Formación Nacional
                                                        +-------IF
                                                        BETWEEN(SUBSTR(ALLTRIM(THISFORM.PAG
                                                        EFRAME1.PAGE1.ECODFTE.VALUE),17,4),'885
                                                        0','8950')
= MESSAGEBOX('Consecutivo de Fuente reservado para asignarle Articulos de Formación Nacional
!!!',0+64+0,'Precaución')
                                                        +-------IF
                                                        BETWEEN(SUBSTR(ALLTRIM(THISFORM.PAG
                                                        EFRAME1.PAGE1.ECODFTE.VALUE),17,4),'888
                                                        7','8890')
= MESSAGEBOX('Consecutivo de Fuente reservado para asignarle Servicios Publicos
!!!',0+64+0,'Precaución')
&&Si esta reservado para servicios publicos se permite ingresar la información asi no sea bogota
|     |     |     |     |     |     |       +-------ENDIF
                                                        +-------IF
                                                        BETWEEN(SUBSTR(ALLTRIM(THISFORM.PAG
                                                        EFRAME1.PAGE1.ECODFTE.VALUE),17,4),'885
                                                        0','8886') OR
                                                        BETWEEN(SUBSTR(ALLTRIM(THISFORM.PAG
                                                        EFRAME1.PAGE1.ECODFTE.VALUE),17,4),'889
                                                        1','8950')
&&En caso de pertenecer al rango de Formación Nacional y no ser bogota no permitimos trabajar
estas fuentes
= MESSAGEBOX('CONSECUTIVO DE FUENTE RESERVADO PARA UTILIZARLOS BOGOTA
UNICAMENTE !!!',0+64+0,'Información')
|     |     |     |     |     |     |     +-------ENDIF
|     |     |     |     |     |     +-------ENDIF
&&Buscamos la fuente
&&Cargamos el parametro con el codigo de fuente para traer la informacion correspondiente
                                        FTEPAFTE =
                                        SUBSTR(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODF
                                        TE.VALUE),17,4)
|     |     |     |   REQUERY('v_ftepa')
|     |     |     |   SELE V_FTEPA
|     |     |     |   +-------IF RECCOUNT() > 0
|     |     |     |               SELE V_FTEPA
&&Actualizamos la informacion digitada en el archivo
          REPLACE V_FTEPA.CODFTE WITH
          ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE);
```

```
        V_FTEPA.NOMBFTE WITH
        ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE);
        V_FTEPA.DIREFTE WITH
        ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE);
        V_FTEPA.TELFTE WITH ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE);
        V_FTEPA.ESTRFTE WITH THISFO       RM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE;
        V_FTEPA.ZONA WITH THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE;
        V_FTEPA.MARCA WITH .F.
|     |     |     |           TABLEUPDATE(.T.)
&&Confirmamos la actualización
        WAIT WIND "Información reemplazada" NOWAIT

                                                PASOFTE = V_FTEPA.CODFTE
|     |     |     |     +-------ELSE
&&Si la fuente no es encontrada, ingresamos la informacon
|     |     |     |     |     |     SELE V_FTEPA
        INSERT INTO V_FTEPA(CODFTE, NOMBFTE, DIREFTE, TELFTE, ESTRFTE, ZONA,
        MARCA);
        VALUES(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE),;
        ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE),;
        ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE),;
        ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE),;
        THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE,;
        THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE, .F.)
&&Reiniciamos valores para los objetos para recibir nueva información
                THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE="
                THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE="
                THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE="
                THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
                THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
                THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
&&Los dejamos de solo lectura mientras digitan nueva fuente
                THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.T.
                THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.T.
                THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.T.
                THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.T.
                THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.T.
|     |     |     |     |           TABLEUPDATE(.T.)
                WAIT WIND "Información Ingresada" NOWAIT
|     |     |     |     |     |     PASOFTE = V_FTEPA.CODFTE
|     |     |     |     |     |      +-------ENDIF
|     |     |     |     |     +-------ELSE
&&Validamos que el consecutivo no este reservado Ya sea para Servicios Publicos o
&&para Formación Nacional
```

```
                                                    +-------IF
                                                    BETWEEN(SUBSTR(ALLTRIM(THISFORM.PAG
                                                    EFRAME1.PAGE1.ECODFTE.VALUE),17,4),'885
                                                    0','8950')
= MESSAGEBOX('Consecutivo de Fuente reservado para asignarle Articulos de Formación Nacional
!!!',0+64+0,'Precaución')
                                                    +-------IF
                                                    BETWEEN(SUBSTR(ALLTRIM(THISFORM.PAG
                                                    EFRAME1.PAGE1.ECODFTE.VALUE),17,4),'888
                                                    7','8890')

= MESSAGEBOX('Consecutivo de Fuente reservado para asignarle Servicios Publicos
!!!',0+64+0,'Precaución')

&&Buscamos la fuente
&&Si esta reservado para servicios publicos se permite ingresar la información asi no sea bogota
FTEPAFTE = SUBSTR(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE),17,4)

                                                    REQUERY('v_ftepa')
                                                    SELE V_FTEPA
&&Si la fuente ya existia se reemplaza con la nueva informacion digitada
                                                            +-------IF RECCOUNT() > 0
                                                                        SELE V_FTEPA
&&Actualizamos la informacion digitada en el archivo
        REPLACE V_FTEPA.CODFTE WITH
        ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE);
V_FTEPA.NOMBFTE WITH ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE);
V_FTEPA.DIREFTE WITH ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE);
V_FTEPA.TELFTE WITH ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE);
V_FTEPA.ESTRFTE WITH THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE;
V_FTEPA.ZONA WITH THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE;
V_FTEPA.MARCA WITH .F.
&&Reiniciamos valores a los objetos para recibir nueva información
                THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE=''
                THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE=''

                THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE=''
                THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
                THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
                THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.T.
                THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.T.
                THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.T.
                THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.T.
```

```
                THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.T.
&&Confirmamos la actualización
TABLEUPDATE(.T.)
WAIT WIND "Información reemplazada" NOWAIT
PASOFTE = V_FTEPA.CODFTE
                                                +-------ELSE
&&Si la fuente no es encontrada, ingresamos la informacon
SELE V_FTEPA
INSERT INTO V_FTEPA(CODFTE,NOMBFTE,DIREFTE,TELFTE,ESTRFTE,ZONA, MARCA);
VALUES(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE),;
ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE),;
ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE),;
ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE),;
THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE,;
THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE, .F.)
&&Reiniciamos valores a los objetos
                THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE="
                THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE="
                THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE="
                THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
                THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
                THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
&&Habilitamos e inhabilitamos objetos segun el caso
                THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.T.
                THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.T.
                THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.T.
                THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.T.
                THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.T.
&&Confirmamos la actualización
                                                TABLEUPDATE(.T.)

                                                WAIT WIND "Información Ingresada" NOWAIT
&&Cargamos la variable con la fuente trabajada para no tener
&&que digitarla en caso de pasara a otra de las pantallas
                                                PASOFTE = V_FTEPA.CODFTE
|     |     |     |     | +-------ENDIF
|     |     |     |       +-------ENDIF
&&Validamos que el consecutivo no este reservado Ya sea para Servicios Publicos o
&&para Formación Nacional, para asi mismo permitir trabajarlos o no segun sea Bogota o no
                                                +-------IF
                                                BETWEEN(SUBSTR(ALLTRIM(THISFORM.PAGEFRAME
                                                1.PAGE1.ECODFTE.VALUE),17,4),'8850','8886') OR
                                                BETWEEN(SUBSTR(ALLTRIM(THISFORM.PAGEFRAME
                                                1.PAGE1.ECODFTE.VALUE),17,4),'8891','8950')
```

```
                                        = MESSAGEBOX('CONSECUTIVO DE FUENTE
                                        RESERVADO PARA UTILIZARLOS BOGOTA
                                        UNICAMENTE !!!',0+64+0,'Información')

                                        THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE =
                                        SUBSTR(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.E
                                        CODFTE.VALUE),1,17)
                        |     |     +-------ENDIF
|     |     |           +-------ELSE
&&REconsultamos la información para verificar si la informacion existe o no
FTEPAFTE = SUBSTR(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE),17,4)

REQUERY('v_ftepa')
SELE V_FTEPA
|     |     |     |     | +-------IF RECCOUNT() > 0
                                                    SELE V_FTEPA
&&Actualizamos la informacion digitada en el archivo
REPLACE V_FTEPA.CODFTE WITH
ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE);
V_FTEPA.NOMBFTE WITH ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE);
V_FTEPA.DIREFTE WITH ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE);
V_FTEPA.TELFTE WITH ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE);
V_FTEPA.ESTRFTE WITH THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE;
V_FTEPA.ZONA WITH THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE;
V_FTEPA.MARCA WITH .F.
&&Reinicializar valores para los objetos del formulario
                THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE="
                THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE="
                THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE="
                THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
                THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
                THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
                THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
&&Habilitar e inhabilitar objetos segun sea el caso
                THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.T.
                THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.T.
                THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.T.
                THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.T.
                THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.T.

                TABLEUPDATE(.T.)
                WAIT WIND "Información reemplazada" NOWAIT

                                        PASOFTE = V_FTEPA.CODFTE
                                        +-------ELSE
```

```
&&Si la fuente no es encontrada, ingresamos la informacon
SELE V_FTEPA
INSERT INTO V_FTEPA(CODFTE,NOMBFTE,DIREFTE,TELFTE,ESTRFTE,ZONA, MARCA);
VALUES(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE),;

        ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE),;
        ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE),;
        ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE),;
        THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE,;
        THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE, .F.)
&&Reinicializamos valores para los objetos del formulario
        THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE="
        THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
        THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE="
        THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
        THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE="
        THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
        THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
        THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
        THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
        THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
&&Habilitar e inhabilitar objetos segun sea el caso
        THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.T.
        THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.T.
        THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.T.
        THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.T.
        THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.T.
&&Actualizar los cambios hechos
                                TABLEUPDATE(.T.)
                                WAIT WIND "Información Ingresada" NOWAIT
&&Cargamos la variable con el codigo de fuente trabajado
                                PASOFTE=V_FTEPA.CODFTE
                                +-------ENDIF
                        +-------ENDIF
|       |       +-------ENDIF
|               +-------ELSE
|                   = MESSAGEBOX('Consecutivo NO Valido',0+64,'Advertencia')                |       +-------
ENDIF
|       +-------ELSE
                        = MESSAGEBOX('Codigo NO Valido',0+64,'Advertencia')
                    +-------ENDIF
|       +-------ENDIF
+-------ENDIF
+-------ENDPROC

+-------PROCEDURE GOTFOCUS
&&Validamos la consistencia de la informacion digitada para la fuente
|       +-------IF LEN(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE))=20 AND;
```

```
!EMPTY(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE)) AND;
|      |      |!EMPTY(THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE) AND;
|      |      |!EMPTY(THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE)
|      |       THISFORM.PAGEFRAME1.PAGE1.GUARDAR.ENABLED =.T.
|      +-------ELSE
|      |       = MESSAGEBOX('Información Incompleta',0,'Información')
|      |       THISFORM.PAGEFRAME1.PAGE1.GUARDAR.ENABLED =.F.
|      |       THISFORM.PAGEFRAME1.PAGE1.ECODFTE.SETFOCUS
|      +-------ENDIF
+-------ENDPROC
```

5.13.4 Nombre Fuente(enombfte)

```
+-------PROCEDURE GOTFOCUS
&&Validamos que el codigo de la fuente ingresado sea de 20 caracteres
|      +-------IF LEN(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE))<20
                        MENSAJE=MESSAGEBOX(" Recuerde que el codigo de la fuente debe tener
                        20 digitos,corrija por favor",0+32+0,"Actualizar Fuentes")
&&En caso de no ser asi devolvemos el foco para que lo vuelvan a digitar
|      |            THISFORM.PAGEFRAME1.PAGE1.ECODFTE.SETFOCUS
|      +-------ENDIF
|
+-------ENDPROC
```

5.13.5 Codigo Fuente(ecodfte)

```
+-------PROCEDURE LOSTFOCUS
&&Cargamos el contenido de la lista de fuentes complementarias a la variable
&&Validamos el codigo de la fuente original digitado
         +-------IF
         !INLIST(SUBSTR(THIS.VALUE,1,2),'05','08','11','13','17','23','41','50','52','54','66','68','76')
|                MESSAGEBOX(" El código del Departamento no es válido, corrija por favor..
                ",0+48+0,"ADVERTENCIA")
|      |
&&Volvemos de solo lectura los objetos
|      |            THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE="
|      |            THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE="
|      |            THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE="
|      |            THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
|      |            THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
|      |
|      |            THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.T.
|      |            THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.T.
|      |            THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.T.
|      |            THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.T.
|      |            THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.T.
|      |
|      |            THISFORM.PAGEFRAME1.PAGE1.GUARDAR.ENABLED=.F.
```

```
&&Volvemos a dejar el objeto donde debe ir codigo de la fuente
&&original con los primero 5 digitos
        THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE=ALLTRIM(DEPARTAMENTO)+ALLTR
        IM(MUNICIPIO)
|     |           THISFORM.PAGEFRAME1.PAGE1.ECODFTE.SETFOCUS
|     |
|     +-------ELSE
|     |
                        +-------IF
                        LEN(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE))<20
&&En caso de no ser aceptable el codigo digitado
                                WAIT WIND " Recuerde que el codigo de la fuente debe tener 20 digitos
                                " NOWAIT
&&Volvemos de solo lectura los objetos
|     |     |           THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE=''
|     |     |           THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE=''
|     |     |           THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE=''
|     |     |           THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
|     |     |           THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
|     |     |
|     |     |           THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.T.
|     |     |           THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.T.
|     |     |           THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.T.
|     |     |           THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.T.
|     |     |           THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.T.
|     |     |
|     |     |           THISFORM.PAGEFRAME1.PAGE1.GUARDAR.ENABLED=.F.
&&Limpiamos la lista
&&Volvemos a dejar el objeto donde debe ir codigo de la fuente
&&original con los primero 5 digitos
        THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE=ALLTRIM(DEPARTAMENTO)+ALLTR
        IM(MUNICIPIO)
|     |     |           THISFORM.PAGEFRAME1.PAGE1.ECODFTE.SETFOCUS
|     |     +-------ELSE
&&Si es correcto el codigo digitado
|     |     |     +-------IF LEN(ALLTRIM(THIS.VALUE))=20
                                    +-------IF
                                    SUBSTR(ALLTRIM(THISFORM.PAGEFRAME1.PAGE1.ECODF
                                    TE.VALUE),17,4)<>"0000"
                                          +-------IF SUBSTR(ALLTRIM(THIS.VALUE),1,2)="11" AND
                                          SUBSTR(ALLTRIM(THIS.VALUE),1,2)=ALLTRIM(DEPAR
                                          TAMENTO)
                                                +-------IF
                                                BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4
                                                ),'8850','8950')
                                                = MESSAGEBOX('Consecutivo de Fuente
                                                reservado para asignarle Articulos de Formación
                                                Nacional !!!',0+64+0,'Precaución')
```

```
                                                        +-------IF
                                                        BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4
                                                        ),'8887','8890')
= MESSAGEBOX('Consecutivo de Fuente reservado para asignarle Servicios Publicos
!!!',0+64+0,'Precaución')
|     |     |     |     |     +-------ENDIF
                                        |               +-------IF
                                                        BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4
                                                        ),'8850','8886') OR
                                                        BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4
                                                        ),'8891','8950')
= MESSAGEBOX('Consecutivo de Fuente reservado para utilizarlos Bogota !!!',0+64+0,'Información')
|     |     |     |     |     |+-------ENDIF
|     |     |     |     |     |     |
|     |     |     |         +-------ENDIF
&&Buscamos la fuente
|     |     |           FTEPAFTE = SUBSTR(ALLTRIM(THIS.VALUE),17,4)
|     |     |     |           REQUERY('v_ftepa')
|     |     |                     SELE V_FTEPA
|     |     |     |           +-------IF RECCOUNT() > 0
|     |     |     |                 +-------IF V_FTEPA.MARCA = .F.
|     |     |     |     |     |             PASOFTE = V_FTEPA.CODFTE
&&Si encontramos la fuente que corresponda al codigo digitado
&&Extraemos las fuentes complementarias correspondientes

&&Refrescamos pantalla y habilitamos los objetos para que puedan
&&modificar la informacion
                        THISFORM.PAGEFRAME1.PAGE1.recuadro1.LIST1.REFRESH
                        THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.F.
&&Cargamos los objetos con la informacion de la fuente
THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE = V_FTEPA.NOMBFTE
THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE = V_FTEPA.DIREFTE
THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE = V_FTEPA.TELFTE
THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE = V_FTEPA.ESTRFTE
THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE = V_FTEPA.ZONA
THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
|     |     +-------ELSE
= MESSAGEBOX(' La Fuente no se encuentra en el Maestro de Fuentes, Ingrese la informacion por
favor ',0+64+0,'Actualizar Fuentes')
&&Habilitamos los objetos para que se pueda digitar informacion
```

```
&&Nombre, direccion, telefono, etc
                        THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE=''
                        THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
                        THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE=''
                        THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
                        THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE=''
                        THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
                        THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
                        THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
                        THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
                        THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
|     |     +-------ENDIF
|           +-------ELSE
&&Si la fuente no es encontrada, lo informamos, y solicitamos
&&que ingrese la informacion de la nueva fuente
                        = MESSAGEBOX(' La Fuente no se encuentra en el Maestro de Fuentes,
                        Ingrese la informacion por favor ',0+64+0,'Actualizar Fuentes')
&&Habilitamos los objetos para que se pueda digitar informacion
&&Nombre, direccion, telefono, etc
                        THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE=''
                        THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
                        THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE=''
                        THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
                        THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE=''
                        THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
                        THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
                        THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
                        THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
                        THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
|     |     +-------ENDIF
|           +-------ELSE
&&Validamos el consecutivo de la fuente actualemnet digitado para saber
&&si estan reservados para utilizarlos para articulos de formacion nacional o para servicios publicos
                        +-------IF BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4),'8850','8950')
                                = MESSAGEBOX('Consecutivo de Fuente reservado para asignarle
                                Articulos de Formación Nacional !!!',0+64+0,'Precaución')
                                +-------IF
                                BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4),'8887','8890')
```

```
= MESSAGEBOX('Consecutivo de Fuente reservado para asignarle Servicios Publicos
!!!',0+64+0,'Precaución')
&&Buscamos la fuente
|      |      |            FTEPAFTE = SUBSTR(ALLTRIM(THIS.VALUE),17,4)
&&Reconsultamos la informacion origen para verificar la existencia del consecutivo digitado
                                        REQUERY('v_ftepa')
                                        SELE V_FTEPA
&&Si se encuentra en el maestro de fuentes, verificamos si esta marcado para eliminarse
                                        +-------IF RECCOUNT() > 0
|      |      |      |      +-------IF V_FTEPA.MARCA = .F.
&&Cargamos el codigo de fuente a la variable
                                                PASOFTE = V_FTEPA.CODFTE
&&Si encontramos la fuente que corresponda al codigo digitado
&&Extraemos las fuentes complementarias correspondientes

&&Refrescamos pantalla y habilitamos los objetos para que puedan
&&modificar la informacion
&&Habilitamos e inhabilitamos objetos segun sea el caso
                        THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.F.
&&Cargamos los objetos con la informacion de la fuente
THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE = V_FTEPA.NOMBFTE
THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE = V_FTEPA.DIREFTE
THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE = V_FTEPA.TELFTE
THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE = V_FTEPA.ESTRFTE
THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE = V_FTEPA.ZONA
THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
                                +-------ELSE
= MESSAGEBOX(' La Fuente no se encuentra en el Maestro de Fuentes, Ingrese la informacion por
favor ',0+64+0,'Actualizar Fuentes')
&&Habilitamos los objetos para que se pueda digitar informacion
&&Nombre, direccion, telefono, etc
                        THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.F.
                        THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE=''
                        THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
                        THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE=''
                        THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
```

```
                    THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE=''
                    THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
                    THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
                    THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
                    THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
                    THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
                              +-------ENDIF
                         +-------ELSE
&&Si la fuente no es encontrada, lo informamos, y solicitamos
&&que ingrese la informacion de la nueva fuente
                              = MESSAGEBOX(' La Fuente no se encuentra en el Maestro de
                              Fuentes, Ingrese la informacion por favor ',0+64+0,'Actualizar
                              Fuentes')
&&Habilitamos los objetos para que se pueda digitar informacion
&&Nombre, direccion, telefono, etc
                    THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.F.
                    THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.F.
                    THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.F.
                    THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.F.
                    THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.F.
                    THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE=''
                    THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
                    THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE=''
                    THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
                    THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE=''
                    THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
                    THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
                    THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
                    THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
                    THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
|     |     |     +-------ENDIF
|           |     +-------ENDIF
&&validamos que el consecutivos de fuente no sea reservado para Formación Nacional
              ||     +-------IF BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4),'8850','8886') OR
                         BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4),'8891','8950')
                         = MESSAGEBOX('Consecutivo de Fuente reservado para utilizarlos
                         Bogota !!!',0+64+0,'Información')
                         THIS.VALUE=SUBSTR(ALLTRIM(THIS.VALUE),1,17)
|     |     +-------ENDIF
|           +-------ELSE
&&Buscamos la fuente
|           FTEPAFTE = SUBSTR(ALLTRIM(THIS.VALUE),17,4)
                    REQUERY('v_ftepa')
|           SELE V_FTEPA
|                   +-------IF RECCOUNT() > 0
|     |     |     +-------IF V_FTEPA.MARCA = .F.
                                   PASOFTE = V_FTEPA.CODFTE
&&Si encontramos la fuente que corresponda al codigo digitado
```

```
&&Extraemos las fuentes complementarias correspondientes
&&La insertamos una a una en la lista para fuentes complementarias

&&Refrescamos pantalla y habilitamos los objetos para que puedan
&&modificar la informacion

THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.F.
&&Cargamos los objetos con la informacion de la fuente
THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE = V_FTEPA.NOMBFTE
THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE = V_FTEPA.DIREFTE
THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE = V_FTEPA.TELFTE
THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE = V_FTEPA.ESTRFTE
THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE = V_FTEPA.ZONA
THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
|      |      |      +-------ELSE
= MESSAGEBOX(' La Fuente no se encuentra en el Maestro de Fuentes, Ingrese la informacion por
favor ',0+64+0,'Actualizar Fuentes')
&&Habilitamos los objetos para que se pueda digitar informacion
&&Nombre, direccion, telefono, etc
THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE=''
THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE=''
THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE=''
THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
|      |      |      +-------ENDIF
|              |      +-------ELSE
&&Si la fuente no es encontrada, lo informamos, y solicitamos
&&que ingrese la informacion de la nueva fuente
                                = MESSAGEBOX(' La Fuente no se encuentra en el Maestro de
                                Fuentes, Ingrese la informacion por favor ',0+64+0,'Actualizar Fuentes')
```

```
&&Habilitamos los objetos para que se pueda digitar informacion
&&Nombre, direccion, telefono, etc
THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE1.ETELFTE.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE1.EZONA.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.VALUE=''
THISFORM.PAGEFRAME1.PAGE1.ENOMBFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.VALUE=''
THISFORM.PAGEFRAME1.PAGE1.EDIREFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.ETELFTE.VALUE=''
THISFORM.PAGEFRAME1.PAGE1.ETELFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.VALUE=1
THISFORM.PAGEFRAME1.PAGE1.EGRUPFTE.REFRESH
THISFORM.PAGEFRAME1.PAGE1.EZONA.VALUE=1
THISFORM.PAGEFRAME1.PAGE1.EZONA.REFRESH
|      |      |      +-------ENDIF
|               |      +-------ENDIF
|      |      +-------ENDIF
|               +-------ELSE
|      |      = MESSAGEBOX('Consecutivo NO Valido',0+64,'Advertencia')
|      | +-------ENDIF
|    +-------ELSE
|         = MESSAGEBOX('Codigo NO Valido',0+64,'Advertencia')
   +-------ENDIF
| +-------ENDIF
 +-------ENDIF
|
+-------ENDPROC
```

5.13.6 Zona(ezona)

```
+-------PROCEDURE VALID
&&Validamos el valor digitado para la zona de la fuente,
&&para permitir guardar o no
|      +-------IF (!EMPTY(THIS.VALUE) OR (THIS.VALUE = 0))
|      |             THISFORM.PAGEFRAME1.PAGE1.GUARDAR.ENABLED=.T.
|      +-------ENDIF
+-------ENDPROC
```

5.13.7 Grupo Fuente(egrupfte)

```
+-------PROCEDURE VALID
&&Validamos el valor digitado para la zona de la fuente,
&&para permitir guardar o no
|      +-------IF (!EMPTY(THIS.VALUE) OR (THIS.VALUE = 0))
|      |             THISFORM.PAGEFRAME1.PAGE1.GUARDAR.ENABLED=.T.
```

```
|       +-------ENDIF
+-------ENDPROC
```

5.14 MANTENIMIENTO FUENTE ARTICULO

5.14.1 Navegador (Navigator1)

```
+-------PROCEDURE RECORDPOINTERMOVED
&&Definimos y cargamos variable con el nombre del archivo
|           LOCAL CALIAS
|           CALIAS=ALIAS()
|
&&Asignamos el contenido de las listas a variables
                LISTASELECCIONADA=THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.
                MOVERLISTA1.LSTSELECTED
                LISTAFUENTE =
                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSOU
                RCE
|
&&Cargamos variables con los codigo de fuente y articulo actuales
|           ACTUALINDEX = LISTASELECCIONADA.LISTITEMID
&&cargamos las variables requeridas para reconsultar la información del archivo
&&Para la fuente
|           VFAPCODFTE = THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALUE
&&para el articulo
                        VFAPCODART =
                        SUBSTR(LISTASELECCIONADA.LIST(LISTASELECCIONADA.LISTITEMID),1,
                        7)
|           LISTASELECCIONADA.LISTITEMID = ACTUALINDEX
|
                        THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALUE =
                        V_IPCMFTE1.CODFTE
&&Ejecutamos el evento para que actualice la informacion de las listas en pantalla
|           THISFORM.ACTUALICELISTAS
&&Cargamos la variable con el codigo de fuente trabajado
|           PASOFTE = THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALUE
|
                        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EP
                        ERIFTE.FORECOLOR=RGB(255,255,255)

                        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.ED
                        ECA.FORECOLOR=RGB(255,255,255)

                        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TE
                        XT2.FORECOLOR=RGB(255,255,255)
```

```
                        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TE
                        XT3.FORECOLOR=RGB(255,255,255)
|
&&Actualizamos informacion en pantalla referente al numero de articulos
&&disponibles para asignar y los ya asignados a la fuente en cuestion
|       +-------WITH THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1
|       |             .NOSELECCIONADOS=LISTASELECCIONADA.LISTCOUNT
|       |             .NODISPONIBLES=LISTAFUENTE.LISTCOUNT
|       |             .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0)
|       |             .LDISPONIBLES.CAPTION=STR(.NODISPONIBLES,5,0)
|       +-------ENDWITH
|
|               SELECT(CALIAS)
+-------ENDPROC
```

5.14.2 Nombre Fuente(CNombreFuente)

```
+-------PROCEDURE INTERACTIVECHANGE
&&Cargamos a variables los contenidos de las listas
&&de articulos asignados a la fuente y los no asignados
                LISTASELECCIONADA =
                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELE
                CTED
                LISTAFUENTE =
                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSOU
                RCE
&&Cargamos variables con los codigos de fuente y articulo para reconsultar
&&la informacion de la vista en base a los parametros de fuente y articulo
|               ACTUALINDEX = LISTASELECCIONADA.LISTITEMID
                        VFAPCODFTE =
                        ALLTRIM(THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALU
                        E)
                        VFAPCODART =
                        SUBSTR(LISTASELECCIONADA.LIST(LISTASELECCIONADA.LISTITEMID),1,
                        7)
&&Ordenamos la lista
|               LISTASELECCIONADA.LISTITEMID = ACTUALINDEX
|
|               SELECT V_IPCMFTE1
&&MOstramos el codigo de fuente en panatlla en el objeto correspondiente
&&y habilitamos la rra de botones de recorrer las fuentes
                        THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALUE =
                        V_IPCMFTE1.CODFTE

                        THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.NAVIGATOR1.ENABLEDIS
                        ABLEBUTTONS
|
&&Cargamos variable para utilizarla como parametro y reconsultar otra vista que
```

```
&&trae la informacion correspondiente a la fuente y al articulo
                        FAPPAFTE =
                        ALLTRIM(THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALU
                        E)
|           REQUERY('v_fappa')
|           SELECT V_FAPPA
&&Consultamos los articulos que se recolectan en la fuente actual
                        SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM V_IPCMART,
                        V_FAPPA WHERE ((V_FAPPA.CODART = V_IPCMART.CODART) AND
                        (V_FAPPA.CODFTE = FAPPAFTE)) AND V_FAPPA.MARCA = .F. INTO
                        CURSOR CANASTAFUENTE
|
|           SELECT CANASTAFUENTE
|           LISTASELECCIONADA.CLEAR
|     +-------SCAN
                        LISTASELECCIONADA.ADDITEM(CANASTAFUENTE.CODART+'
                        '+PROPER(CANASTAFUENTE.NOMART))
|     +-------ENDSCAN
|     +-------IF BETWEEN(SUBSTR(FAPPAFTE,17,4),'8887','8890')
&&En caso que el consecutivo de fuente este en el rango de los reservados para Servicios Publicos
&&solo cargamos los articulos de sErvicios Publicos para que no pueda asignarle articulos de otros
&&grupos
&&Seleccionamos los articulos que no estan asignados a la fuente actual
&&para alimentar la otra lista
SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM V_IPCMART;
WHERE
(INLIST(V_IPCMART.CODART,'2210102','2220101','2220201','2220202','2220203','7320101','732010
2') AND V_IPCMART.CODART NOT IN (SELECT DISTINCT CODART FROM CANASTAFUENTE))
INTO CURSOR CANASTANACIONAL ORDER BY CODART
|     |  SELECT CANASTANACIONAL
|     |
|     |  LISTAFUENTE.CLEAR
|     |  +-------SCAN
                        LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
                        '+PROPER(CANASTANACIONAL.NOMART))
|     |     +-------ENDSCAN
|     +-------ELSE
&&En caso que el consecutivo este en el rango de los resrevados para articulos de formacion
nacional
&&Verificamos que la ciudad sea bogota para permitirles trabajarla o no;  en caso afirmativo
                        +-------IF (BETWEEN(SUBSTR(FAPPAFTE,17,4),'8850','8886') OR
                        BETWEEN(SUBSTR(FAPPAFTE,17,4),'8891','8950')) AND;
                                SUBSTR(FAPPAFTE,1,2) = "11" AND SUBSTR(FAPPAFTE,1,2) =
                                ALLTRIM(DEPARTAMENTO)
&&Seleccionamos los articulos que no estan asignados a la fuente actual para alimentar la otra lista
SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM V_IPCMART;
WHERE (V_IPCMART.FORNAL = .T. AND V_IPCMART.CODART NOT IN (SELECT DISTINCT
CODART FROM CANASTAFUENTE)) INTO CURSOR CANASTANACIONAL ORDER BY CODART
```

```
|     |     |     SELECT CANASTANACIONAL
|     |     |
|     |     |     LISTAFUENTE.CLEAR
|     |     |     +-------SCAN
                                LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
                                '+PROPER(CANASTANACIONAL.NOMART))
|     |     |     +-------ENDSCAN
|     |     +-------ELSE
|     |     |
&&Consultamos los articulos que no se recolectan en la fuente actual y que no
&&sean de Servicios Publicos
SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM V_IPCMART;
WHERE (V_IPCMART.CODART NOT IN (SELECT DISTINCT CODART FROM CANASTAFUENTE)
AND;
|     |     |           (V_IPCMART.FORNAL = .F. AND;
!INLIST(V_IPCMART.CODART,'2210102','2220101','2220201','2220202','2220203','7320101','732010
2')));
|     |     |         INTO CURSOR CANASTANACIONAL
|     |     |         SELECT CANASTANACIONAL
|     |     |         LISTAFUENTE.CLEAR
|     |     |     +-------SCAN
LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
'+PROPER(CANASTANACIONAL.NOMART))
|     |     |     +-------ENDSCAN
|     |     +-------ENDIF
|     +-------ENDIF
|           USE
|           SELECT CANASTAFUENTE
|           USE
&&Refrescamos informacion en pantalla
                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFT
                E.FORECOLOR=RGB(255,255,255)
                |
                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EDECA.F
                ORECOLOR=RGB(255,255,255)
                |
                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT2.F
                ORECOLOR=RGB(255,255,255)
                |
                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT3.F
                ORECOLOR=RGB(255,255,255)
|
&&Mostramos el numero de articulos asignados y los no asignadoas a la fuente actual
|     +-------WITH THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1
|     |           .NOSELECCIONADOS=LISTASELECCIONADA.LISTCOUNT
|     |           .NODISPONIBLES=LISTAFUENTE.LISTCOUNT
|     |           .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0)
|     |           .LDISPONIBLES.CAPTION=STR(.NODISPONIBLES,5,0)
```

```
|       +-------ENDWITH
|
+-------ENDPROC
```

5.14.3 Escriba el Código de la Fuente(EcodFte)

```
+-------PROCEDURE INIT
&&Capturamos el valor del campo del archivo
|           THIS.VALUE=V_IPCMFTE1.CODFTE
+-------ENDPROC

+-------PROCEDURE LOSTFOCUS
&&Almacenamos el contenido de las listas a las variables
                LISTASELECCIONADA=THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.
                MOVERLISTA1.LSTSELECTED

                LISTAFUENTE=THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLI
                STA1.LSTSOURCE
|
&&Almacenamos los codigos de fuente y de articulo en variables para reconsultar
&&la información correspondiente a estos parametros
|           ACTUALINDEX = LISTASELECCIONADA.LISTITEMID
|           VFAPCODFTE = V_IPCMFTE1.CODFTE
                        VFAPCODART =
                        SUBSTR(LISTASELECCIONADA.LIST(LISTASELECCIONADA.LISTITEMID),1,
                        7)
&&Ordenamos la lista
|           LISTASELECCIONADA.LISTITEMID = ACTUALINDEX
|
|           SELECT V_IPCMFTE1
&&Seleccionamos el archivo de fuentes, almacenamos el numero del registro
&&actual en una variable para podernos devolver a el en caso de que la busqueda
&&no obtenga resultados
|           MARCAREGISTRO = RECNO()
|           FUENTECOD = V_IPCMFTE1.CODFTE
|           OFTEPAFTE = ALLTRIM(THIS.VALUE)
&&Reconsultamos la informacion de la vista
|           REQUERY('v_oftepa')
|           SELE V_OFTEPA
&&Comprobamos si la vista trajo informacion es por que la fuente ya existe, de lo
&&contrario es por que obviamente no existe
|       +-------IF RECCOUNT() = 0
&&En caso de no encontrarlo lo informamos y consultamos si desea ingresarla
&&Verificamos que el consecutivo del codigo de fuente digitado sea consistente
|       |       +-------IF SUBSTR(ALLTRIM(THIS.VALUE),17,4)<>"0000"
                                        INGRESAR=MESSAGEBOX('Fuente no encontrada, desea
                                        ingresarla ??',4+32+0,'Información')
|       |       |       +-------DO CASE
```

```
&&En caso afirmativo lo enviamos a la pantalla por donde se puede ingresar la nueva fuente
|     |     |     +-------CASE INGRESAR=6
|     |     |     |           THISFORM.PAGEFRAME1.PAGES(1).SETFOCUS
&&Enviamos el codigo de fuente digitado y lo cargamos a una variable
                                                THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VA
                                                LUE = THIS.VALUE
|     |     |     |           PASOFTE = THIS.VALUE
&&Enviamos el foco al objeto donde va el codigo de fuente
                                                THISFORM.PAGEFRAME1.PAGE1.ECODFTE.SE
                                                TFOCUS

                                                THISFORM.PAGEFRAME1.PAGE1.ECODFTE.RE
                                                FRESH

                                                THISFORM.PAGEFRAME1.PAGE1.ECODFTE.LO
                                                STFOCUS
&&NOs devolvemos al registro que estabamos inicialmente antes de iniciar la busqueda
&&de la fuente digitada
|     |     |     |           SELECT V_IPCMFTE1
|     |     |     |           GO MARCAREGISTRO
|     |     |     |           THIS.VALUE = V_IPCMFTE1.CODFTE
|     |     |     +-------CASE INGRESAR=7
&&En caso de seleccionar no ingresar la fuente nos devolvemos al registro
&&que estabamaos inicialmente
|     |     |     |           SELECT V_IPCMFTE1
|     |     |     |           GO MARCAREGISTRO
|     |     |     |           THIS.VALUE = V_IPCMFTE1.CODFTE
|     |     |     +-------ENDCASE
|     |     +-------ELSE
&&nos devolvemos al registro que estabamaos inicialmente
|     |     |           SELECT V_IPCMFTE1
|     |     |           GO MARCAREGISTRO
|     |     |           THIS.VALUE = V_IPCMFTE1.CODFTE
|     |     +-------ENDIF
|     +-------ELSE
&&cargamos la variable con el codigo de fuente y ejecutamos el metodo que actualiza
&&las listas en pantalla segun los articulos que tiene asignados la fuente y
&&los que no
|     |           PASOFTE = V_IPCMFTE1.CODFTE
|     |           THISFORM.ACTUALICELISTAS
|     +-------ENDIF
&&Habilitamos la barra de botones y refrescamos pantalla
|           SELECT V_IPCMFTE1

                        THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.NAVIGATOR1.ENABLEDIS
                        ABLEBUTTONS
|           THISFORM.REFRESH
&&Cambiamos de color a la letra donde se muestra precio y cantidad, periodicidad, etc
```

```
&&para cambiar de fuente y seleccionen articulo
                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFT
                E.FORECOLOR=RGB(255,255,255)

                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EDECA.F
                ORECOLOR=RGB(255,255,255)
                |
                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT2.F
                ORECOLOR=RGB(255,255,255)
                |
                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT3.F
                ORECOLOR=RGB(255,255,255)
|
                &&Refrescar pantalla el numero de articulos asignado y el
                &&numero de no asignados a la fuente en cuestion
|       +-------WITH THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1
|       |            .NOSELECCIONADOS=LISTASELECCIONADA.LISTCOUNT
|       |            .NODISPONIBLES=LISTAFUENTE.LISTCOUNT
|       |            .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0)
|       |            .LDISPONIBLES.CAPTION=STR(.NODISPONIBLES,5,0)
|       +-------ENDWITH
|
+-------ENDPROC

+-------PROCEDURE LSTSELECTED.CLICK
&&Capturamos los codigo tanto de fuente como de articulo actual
|           VFAPCODFTE = THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALUE
|           VFAPCODART=SUBSTR(THIS.VALUE,1,7)
|           VFAPCODART1=SUBSTR(THIS.VALUE,1,7)
&&Cambiamos de color a la letra para que se pueda ver la informacion correspondiente
&&a la fuente articulos ( Precio, cantidad, Per, etc.)
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFTE.FOR
        ECOLOR=RGB(0,0,0)
        |
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EDECA.FOREC
        OLOR=RGB(0,0,0)
        |
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT2.FORECO
        LOR=RGB(0,0,0)
        |
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT3.FORECO
        LOR=RGB(0,0,0)
|
&&Enviamos los parametros necesarios a la vista para que nos
&&actualice la informacion
|           SELECT 'vfap'
|           REQUERY('vfap')
```

```
        |
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFTE.REF
        RESH
        |
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EDECA.REFRES
        H
        |
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT2.REFRES
        H
        |
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT3.REFRES
        H
|
|
+-------ENDPROC

+-------PROCEDURE CMDADD.CLICK
&&Almacenamos el contenido de la lista de articulo no asignados
                LISTASELECCIONADA =
                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSOU
                RCE
|
|           THISFORM.LOCKSCREEN=.T.
|     +-------IF CAMBIO
VFAPCODFTE = THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALUE
|     |           FAPPAFTE = ALLTRIM(VFAPCODFTE)
|     |           RESPAFTE = ALLTRIM(VFAPCODFTE)
&&Insertar los nuevos articulos que se han asociado a la fuente
&& A partir del numero de articulos seleccionados
&&insertamos uno a uno los articulos
|     |           CUENTA = LISTASELECCIONADA.LISTCOUNT
|     |     +-------FOR I = 1 TO CUENTA
|     |     |     +-------IF LISTASELECCIONADA.SELECTED(I)
ELARTICULO = SUBSTR(LISTASELECCIONADA.LIST(I),1,7)
|     |     |     |
&&Agrega articulos a la Fuente articulo
&&Buscamos en el archivo fuente - articulo la coincidencia
&&de la fuente - articulo actual
|     |     |     |           FAPPART = ELARTICULO
|     |     |     |           REQUERY('v_fappart')
|     |     |     |           SELECT V_FAPPART
|     |     |     |     +-------IF RECCOUNT() = 0
&&Si no lo encontramos, consultamos las periodicidades  permitidas
&&para el articulo actual
SELECT CODPER, CODPART FROM V_IPCRPART, V_IPCMART WHERE
(V_IPCRPART.CODPER = V_IPCMART.PERART AND V_IPCMART.CODART = ELARTICULO)
INTO CURSOR PERIODICIDADES
|     |     |     |     |           SELE PERIODICIDADES
```

```
|     |     |     |     |           GO TOP
 &&Insertamos la informacion en el archivo de fuente articulo
|     |     |     |     |           PERIODO = PERIODICIDADES.CODPART
|     |     |     |     |           SELECT V_FAPPART
|     |     |     |     |           APPEND BLAN
REPLACE V_FAPPART.CODFTE WITH VFAPCODFTE, V_FAPPART.CODART WITH
ELARTICULO,;
V_FAPPART.PERFAP WITH PERIODO, V_FAPPART.MARCA WITH .F.
|     |     |     |     |           TABLEUPDATE(.T.)
|     |     |     |     +-------ELSE
|     |     |     |     |          SELECT V_FAPPART
REPLACE V_FAPPART.MARCA WITH .F. FOR ((V_FAPPART.CODFTE = VFAPCODFTE) AND
(V_FAPPART.CODART = ELARTICULO))
|     |     |     |     |           TABLEUPDATE(.T.)
|     |     |     |     +-------ENDIF
|     |     |     |           SELE V_IPCRFAP
|     |     |     |           TABLEUPDATE(.T.)
|     |     |     |
&&Realizamos la busqueda en el archivo de especificaciones
&&Si no se encuentra
&&Agrega la informacion en el archivo
|     |     |     |           RESPART = ELARTICULO
|     |     |     |           REQUERY('v_resppa')
|     |     |     |           SELECT V_RESPPA
|     |     |     |     +-------IF RECCOUNT() = 0
SELECT DISTINCT CODESP, NOMESP FROM V_IPCRESP WHERE (CODART = ELARTICULO)
GROUP BY CODESP INTO CURSOR ESPBASE
|     |     |     |     |           SELECT ESPBASE
|     |     |     |     |     +-------SCAN
INSERT INTO V_RESPPA (CODFTE, CODART, CODESP, NOMESP, MARCA);

VALUES(VFAPCODFTE, ELARTICULO, ESPBASE.CODESP, ESPBASE.NOMESP, .F.)
|     |     |     |     |     +-------ENDSCAN
|     |     |     |     |           USE
|     |     |     |     +-------ELSE
REPLACE V_RESPPA.MARCA WITH .F. FOR ((V_RESPPA.CODFTE = VFAPCODFTE) AND
(V_RESPPA.CODART = ELARTICULO))
|     |     |     |     +-------ENDIF
|     |     |     |           SELE V_RESPPA
|     |     |     |           TABLEUPDATE(.T.)
|     |     |     +-------ENDIF
|     |     +-------ENDFOR
|     |           CAMBIO=.F.
|     +-------ENDIF
|
|           DODEFAULT()
&&Refrescamos pantalla
```

```
        |
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFTE.FOR
        ECOLOR=RGB(255,255,255)
        |
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EDECA.FOREC
        OLOR=RGB(255,255,255)
        |
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT2.FORECO
        LOR=RGB(255,255,255)
        |
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT3.FORECO
        LOR=RGB(255,255,255)
|
|       +-------WITH THIS.PARENT
|       |             .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0)
|       |             .LDISPONIBLES.CAPTION=STR(.NODISPONIBLES,5,0)
|       +-------ENDWITH
+-------ENDPROC

+-------PROCEDURE CMDADDALL.CLICK
                    MENSAJE=MESSAGEBOX(" ESTA SEGURO DE AGREGARLE TODOS LOS
                    ARTICULO DE LA CANASTA NACIONAL A ESTA FUENTE ?",4+32+256,"
                    ADVERTENCIA ")
|       +-------DO CASE
|       +-------CASE MENSAJE=6
                            LISTASELECCIONADA=THISFORM.PAGEFRAME1.PAGE1.MOVERLISTA1.L
                            STSELECTED
|       |       +-------IF CAMBIO
                            ****Insertar los nuevos articulos que se han asociado a la fuente
|       |       |           VFAPCODFTE=THISFORM.CONTAINER2.ECODFTE.VALUE
|       |       |           CUENTA=LISTASELECCIONADA.LISTCOUNT
|       |       |       +-------FOR I=1 TO CUENTA
                                                ELARTICULO=SUBSTR(LISTASELECCIONADA.LIST(I),
                                                1,7)
|       |       |       |           *_____________________________________________
|       |       |       |           * Agrega articulos a la Fuente articulo *
|       |       |       |           *_____________________________________________
|       |       |       |           SELECT IPCRFAP
LOCATE FOR (IPCRFAP.CODFTE+IPCRFAP.CODART)=(VFAPCODFTE+ELARTICULO)
|       |       |       |       +-------IF !FOUND()
SELECT CODPER,CODPART FROM IPCRPART,IPCMART WHERE
(IPCRPART.CODPER=IPCMART.PERART AND IPCMART.CODART=ELARTICULO) INTO
CURSOR PERIODICIDADES
|       |       |       |       |           SELE PERIODICIDADES
|       |       |       |       |           GO TOP
|       |       |       |       |           PERIODO=PERIODICIDADES.CODPART
INSERT INTO IPCRFAP (CODFTE,CODART,PERFAP,MARCA);
```

```
                                                        VALUES(VFAPCODFTE,ELARTICULO,PE
                                                        RIODO,.T.)
|     |     |     |     +-------ENDIF
* Agrega especificaciones a los articulos *
|     |     |     |           SELECT IPCRESP
LOCATE FOR (IPCRESP.CODFTE+IPCRESP.CODART)=(VFAPCODFTE+ELARTICULO)
|     |     |     |     +-------IF !FOUND()
SELECT DISTINCT CODESP,NOMESP FROM IPCRESP WHERE (CODART=ELARTICULO)
GROUP BY CODESP INTO CURSOR ESPBASE
|     |     |     |     |           SELECT ESPBASE
|     |     |     |     |     +-------SCAN
INSERT INTO IPCRESP (CODFTE,CODART,CODESP,NOMESP,MARCA);

VALUES(VFAPCODFTE,ELARTICULO,ESPBASE.CODESP,ESPBASE.NOMESP,.T.)
|     |     |     |     |     |
|     |     |     |     |     +-------ENDSCAN
|     |     |     |     |           USE
|     |     |     |     +-------ENDIF
|     |     |     +-------ENDFOR
|     |     |           CAMBIO=.F.
|     |     +-------ENDIF
|     |
|     |           DODEFAULT()
|     |     +-------WITH THIS.PARENT
                                |     |     |
                                .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0)
|     |     |           .LDISPONIBLES.CAPTION=STR(.NODISPONIBLES,5,0)
|     |     +-------ENDWITH
|     +-------ENDCASE
+-------ENDPROC

+-------PROCEDURE CMDREMOVE.CLICK
&&Almacenamos el contenido de la lista de articulos seleccionados en una variable
                LISTASELECCIONADA =
                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELE
                CTED
|     +-------IF CAMBIO
&&Cargamos variables
                        VFAPCODFTE =
                        THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALUE
|     |       FAPPAFTE = ALLTRIM(VFAPCODFTE)
|     |       RESPAFTE = ALLTRIM(VFAPCODFTE)
&&Segun el total de articulos seleccionados
|     |           CUENTA = LISTASELECCIONADA.LISTCOUNT
|     |     +-------FOR I = 1 TO CUENTA
|     |     |     +-------IF LISTASELECCIONADA.SELECTED(I)
&&Capturamos el articulo actual
```

```
                                        ELARTICULO =
                                        SUBSTR(LISTASELECCIONADA.LIST(I),1,7)
|     |     |     |
&&Eliminar articulos que se han desasociado a la fuente
&&Elimina articulos de la Fuente articulo
|     |     |     |           ******
|     |     |     |           FAPPART = ELARTICULO
|     |     |     |           REQUERY('v_fappart')
|     |     |     |           SELECT V_FAPPART
|     |     |     |     +-------IF RECCOUNT() <> 0
|     |     |     |     |
|     |     |     |     |           SELE V_FAPPART
|     |     |     |     |           REPLACE V_FAPPART.MARCA WITH .T.
|     |     |     |     |           TABLEUPDATE(.T.)
|     |     |     |     +-------ENDIF
&&Eliminar especificaciones a los articulos
|     |     |     |           RESPART = ELARTICULO
|     |     |     |           REQUERY('v_resppa')
|     |     |     |           SELECT V_RESPPA
|     |     |     |     +-------IF RECCOUNT() <> 0
|     |     |     |     |           SELE V_RESPPA
|     |     |     |     |           REPLACE V_RESPPA.MARCA WITH .T.
|     |     |     |     |           TABLEUPDATE(.T.)
|     |     |     |     +-------ENDIF
|     |     |     |
|     |     |     +-------ENDIF
|     |     +-------ENDFOR
|     |
|     |           CAMBIO=.F.
|     +-------ENDIF
|
|           DODEFAULT()
|
        |
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFTE.FOR
        ECOLOR=RGB(255,255,255)
        |
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EDECA.FOREC
        OLOR=RGB(255,255,255)
        |
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT2.FORECO
        LOR=RGB(255,255,255)
        |
        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT3.FORECO
        LOR=RGB(255,255,255)
&&Refrescamos pantalla
|     +-------WITH THIS.PARENT
|     |           .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0)
```

```
|       |           .LDISPONIBLES.CAPTION=STR(.NODISPONIBLES,5,0)
|       +-------ENDWITH
|
+-------ENDPROC
```

5.14.4 Decada(edeca)

```
+-------PROCEDURE VALID
&&Validamos el valor digitado para la decada de la fuente - articulo
|       +-------IF EMPTY(THIS.VALUE)
|       |           MESSAGEBOX("Decada no permitida !!",0,"Informacion")
|       |           THISFORM.PAGEFRAME1.PAGE2.BFINALIZAR.ENABLED=.F.
|       +-------ELSE
|       |           THISFORM.PAGEFRAME1.PAGE2.BFINALIZAR.ENABLED=.T.
|       +-------ENDIF
+-------ENDPROC
```

5.14.5 Periodicidad(eperifte)

```
+-------PROCEDURE LOSTFOCUS
&&Validamos las periodicidades asignadas a los articulos a ver si corresponden a los mismos
|       +-------IF !EMPTY(VFAPCODART1)
|       |           LOCAL PER1,PER2,PER3,PER4,m.PERFAP
|       |           STORE " " TO PER1,PER2,PER3,PER4
|       |           M.PERFAP = THIS.VALUE
                                SELECT CODPER, CODPART FROM V_IPCRPART, V_IPCMART
                                WHERE (V_IPCRPART.CODPER = V_IPCMART.PERART AND
                                V_IPCMART.CODART = VFAPCODART1) INTO CURSOR
                                PERIODICIDADES
|       |           SELE PERIODICIDADES
|       |           GO TOP
|       |       +-------IF PERIODICIDADES.CODPER=0
|       |       |           PER1="0"
|       |       +-------ELSE
|       |       |       +-------IF PERIODICIDADES.CODPER=1
|       |       |       |           PER1="1"
|       |       |       |           PER2="2"
|       |       |       +-------ELSE
|       |       |       |       +-------IF PERIODICIDADES.CODPER=2
|       |       |       |       |           PER1="3"
|       |       |       |       |           PER2="4"
|       |       |       |       |           PER3="5"
|       |       |       |       +-------ELSE
|       |       |       |       |       +-------IF PERIODICIDADES.CODPER=3
|       |       |       |       |       |           PER1="6"
|       |       |       |       |       |           PER2="7"
|       |       |       |       |       |           PER3="8"
|       |       |       |       |       |           PER4="9"
```

```
|     |     |     |     |     +-------ENDIF
|     |     |     |     +-------ENDIF
|     |     |     +-------ENDIF
|     |     +-------ENDIF
|     |
|     |           SELE PERIODICIDADES
|     |     +-------SCAN
|     |     |           LOCATE FOR PERIODICIDADES.CODPART = m.PERFAP
|     |     |     +-------IF !FOUND()
                                          MENSAJE=MESSAGEBOX(" Periodicidades permitidas:
                                          "+ALLTRIM(PER1)+','+ALLTRIM(PER2)+','+ALLTRIM(PE
                                          R3)+','+ALLTRIM(PER4)+', Corrija por favor... ',0+64+0,"
                                          INFORMACION ")

                                          THISFORM.PAGEFRAME1.PAGE2.BFINALIZAR.ENABL
                                          ED=.F.
|     |     |     |           THISFORM.PAGEFRAME1.PAGES(2).SETFOCUS
|     |     |     |           EXIT
|     |     |     +-------ELSE
                                     THISFORM.PAGEFRAME1.PAGE2.BFINALIZAR.ENABLED=.T.
|     |     |     |           EXIT
|     |     |     +-------ENDIF
|     |     +-------ENDSCAN
|     +-------ENDIF
+-------ENDPROC
```

5.14.6 Boton Continuar

```
+-------PROCEDURE CLICK
&&Cargamos la variable con el contendifo de la lista de
&&articulos seleccionados
        |
        LISTASELECCIONADA=THISFORM.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELECTED
|
|           ACTUALINDEX=LISTASELECCIONADA.LISTITEMID
|     +-------WITH THISFORM
&&Actualizamos valor de las variables con codigo de fuente y articulo
|     |           VFAPCODFTE=.CONTAINER2.ECODFTE.VALUE
                  VFAPCODART=SUBSTR(LISTASELECCIONADA.LIST(LISTASELECCIONADA.LISTI
                  TEMID),1,7)
|     +-------ENDWITH
|
|           LISTASELECCIONADA.LISTITEMID=ACTUALINDEX
                      TEXTOARTICULO=LISTASELECCIONADA.LIST(LISTASELECCIONADA.LISTI
                      TEMID)
|           THISFORM.HIDE
|        +-------ENDPROC
```

5.14.7 Boton Siguiente (Siguiente)

```
+-------PROCEDURE CLICK
        LISTASELECCIONADA=THISFORM.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELECTED
|
|     +-------IF LISTASELECCIONADA.LISTITEMID<LISTASELECCIONADA.LISTCOUNT

                        LISTASELECCIONADA.LISTITEMID=LISTASELECCIONADA.LISTITEMID+1
|     |       VESPFUENTE=THISFORM.CONTAINER2.ECODFTE.VALUE
VESPARTICULO=SUBSTR(THISFORM.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELECTED.LIS
T(LISTASELECCIONADA.LISTITEMID),1,7)

TEXTOARTICULO=THISFORM.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELECTED.LIST(LISTA
SELECCIONADA.LISTITEMID)

THISFORM.PAGEFRAME1.PAGE2.TNOMART.VALUE=SUBSTR(TEXTOARTICULO,8,LEN(TEXTO
ARTICULO)-8)
|     |
|     |           VFAPCODFTE=VESPFUENTE
|     |           VFAPCODART=VESPARTICULO
|     |           SELECT 'VFap'
|     |           REQUERY('VFap')
|     |
|     |           THISFORM.REFRESH
|     +-------ELSE
|     |           LISTASELECCIONADA.LISTITEMID=LISTASELECCIONADA.LISTCOUNT
|     |           THISFORM.BFINALIZAR.ENABLED=.T.
|     +-------ENDIF
|
|
+-------ENDPROC
```

5.14.8 Periodicidad(Spinner1)

```
+-------PROCEDURE VALID
|           LOCAL PER1,PER2,PER3,PER4,m.PERFAP
|           STORE " " TO PER1,PER2,PER3,PER4
|           M.PERFAP=THISFORM.PAGEFRAME1.PAGE2.SPINNER1.VALUE
|           M.CODART=THISFORM.PAGEFRAME1.PAGE2.TCODART.VALUE

SELECT CODPER,CODPART FROM IPCRPART,IPCMART WHERE
(IPCRPART.CODPER=IPCMART.PERART AND IPCMART.CODART=m.CODART) INTO CURSOR
PERIODICIDADES
|           SELE PERIODICIDADES
|           GO TOP
|     +-------IF PERIODICIDADES.CODPER=0
|     |           PER1="0"
|     +-------ELSE
```

```
|       |       +-------IF PERIODICIDADES.CODPER=1
|       |       |               PER1="1"
|       |       |               PER2="2"
|       |       +-------ELSE
|       |       |       +-------IF PERIODICIDADES.CODPER=2
|       |       |       |               PER1="3"
|       |       |       |               PER2="4"
|       |       |       |               PER3="5"
|       |       |       +-------ELSE
|       |       |       |       +-------IF PERIODICIDADES.CODPER=3
|       |       |       |       |               PER1="6"
|       |       |       |       |               PER2="7"
|       |       |       |       |               PER3="8"
|       |       |       |       |               PER4="9"
|       |       |       |       +-------ENDIF
|       |       |       +-------ENDIF
|       |       +-------ENDIF
|       +-------ENDIF
|
|
|               SELE PERIODICIDADES
|       +-------SCAN
|       |               LOCATE FOR PERIODICIDADES.CODPART=m.PERFAP
|       |       +-------IF !FOUND()
MENSAJE=MESSAGEBOX(" Periodicidades permitidas:
"+ALLTRIM(PER1)+','+ALLTRIM(PER2)+','+ALLTRIM(PER3)+','+ALLTRIM(PER4)+', Corrija por
favor... ',0+64+0," INFORMACION ")
|       |       |               THISFORM.BFINALIZAR.ENABLED=.F.
|       |       |               EXIT
|       |       +-------ELSE
|       |       |               THISFORM.BFINALIZAR.ENABLED=.T.
|       |       |               EXIT
|       |       +-------ENDIF
|       +-------ENDSCAN
|
+-------ENDPROC
```

5.14.9 Boton Adicionar(Binsertar)

```
+-------PROCEDURE CLICK
|               SELECT IPCRESP

SELECT MAX(CODESP) AS MAXESP FROM IPCRESP WHERE (CODFTE = VFAP.CODFTE) AND
(CODART=VFAP.CODART) INTO CURSOR MAXIMO
|               SELECT MAXIMO
|       +-------IF RECCOUNT() >0
|       |               XX=VAL(MAXESP)+1
|       +-------ELSE
```

```
|       |           XX=1
|       +-------ENDIF
|           USE
|           NUEVOVALOR=ALLTRIM(STR(XX,2,0))
|
|           SET DELETE OFF
|           SELECT IPCRESP
                        =SEEK(VFAP.CODFTE+VFAP.CODART+PADL(NUEVOVALOR, 2,
                        '0'),'ipcresp','PKRESP')
|       +-------IF FOUND()
                        RECALL FOR (CODFTE=VFAP.CODFTE) AND (CODART=VFAP.CODART)
                        AND (CODESP=PADL(NUEVOVALOR, 2, '0'))
                        REPLACE NOMBDET WITH '',NOMESP WITH '' FOR
                        (CODFTE=VFAP.CODFTE) AND (CODART=VFAP.CODART) AND
                        (CODESP=PADL(NUEVOVALOR, 2, '0'))
|       +-------ELSE
|       |           INSERT INTO IPCRESP (CODFTE,CODART,CODESP) ;
                                VALUES (VFAP.CODFTE, VFAP.CODART, PADL(NUEVOVALOR, 2,
                                '0'))
|       |
|       +-------ENDIF
|
|           SET DELETED ON
|
|       +-------WITH THISFORM.PAGEFRAME1.PAGE2.GRID1
|       |           .SETFOCUS
|       |           .REFRESH
|       +-------ENDWITH
+-------ENDPROC
```

5.14.10 Boton Anterior(Banterior)

```
+-------PROCEDURE CLICK
          |
          LISTASELECCIONADA=THISFORM.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELECTED
|
|
|       +-------IF LISTASELECCIONADA.LISTITEMID>1
                        LISTASELECCIONADA.LISTITEMID=LISTASELECCIONADA.LISTITEMID-1
|       +-------ELSE
|       |           LISTASELECCIONADA.LISTITEMID=1
|       +-------ENDIF
|
|           VFAPCODFTE=THISFORM.CONTAINER2.ECODFTE.VALUE
          |
          VFAPCODART=SUBSTR(LISTASELECCIONADA.LIST(LISTASELECCIONADA.LISTITEMID)
          ,1,7)
```

```
        |
        TEXTOARTICULO=THISFORM.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELECTED.LIST
        (LISTASELECCIONADA.LISTITEMID)
        |
        THISFORM.PAGEFRAME1.PAGE2.TNOMART.VALUE=SUBSTR(TEXTOARTICULO,8,LEN(T
        EXTOARTICULO)-8)
|
|           SELECT 'VFap'
|           REQUERY('VFap')
|           THISFORM.REFRESH
+-------ENDPROC
```

5.14.11 Boton Eliminar(Command1)

```
+-------PROCEDURE CLICK
|           SELECT IPCRESP
|           DELETE
|           THISFORM.PAGEFRAME1.PAGE2.GRID1.SETFOCUS
|           THISFORM.PAGEFRAME1.PAGE2.GRID1.REFRESH
+-------ENDPROC
```

5.14.12 Boton Finalizar(Bfinalizar)

```
+-------PROCEDURE CLICK
&&Reactivamos el menu del aplicativo y salimos de esta pantalla
|           SET SYSMENU ON
|           THISFORM.RELEASE
+-------ENDPROC

+-------PROCEDURE GOTFOCUS
&&Validamos las periodicidades asignadas a los nuevos articulos
|     +-------IF !EMPTY(VFAPCODART1)
|     |           LOCAL PER1,PER2,PER3,PER4,m.PERFAP
|     |           STORE " " TO PER1,PER2,PER3,PER4
M.PERFAP =
THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFTE.VALUE
SELECT CODPER, CODPART FROM V_IPCRPART, V_IPCMART WHERE
(V_IPCRPART.CODPER = V_IPCMART.PERART AND V_IPCMART.CODART = VFAPCODART1)
INTO CURSOR PERIODICIDADES
|     |           SELE PERIODICIDADES
|     |           GO TOP
|     |     +-------IF PERIODICIDADES.CODPER=0
|     |     |           PER1="0"
|     |     +-------ELSE
|     |     |     +-------IF PERIODICIDADES.CODPER=1
|     |     |     |           PER1="1"
|     |     |     |           PER2="2"
|     |     |     +-------ELSE
```

```
|     |     |     |     +-------IF PERIODICIDADES.CODPER=2
|     |     |     |     |           PER1="3"
|     |     |     |     |           PER2="4"
|     |     |     |     |           PER3="5"
|     |     |     |     +-------ELSE
|     |     |     |     |     +-------IF PERIODICIDADES.CODPER=3
|     |     |     |     |     |           PER1="6"
|     |     |     |     |     |           PER2="7"
|     |     |     |     |     |           PER3="8"
|     |     |     |     |     |           PER4="9"
|     |     |     |     |     +-------ENDIF
|     |     |     |     +-------ENDIF
|     |     |     +-------ENDIF
|     |     +-------ENDIF
|     |
|     |           SELE PERIODICIDADES
|     |     +-------SCAN
|     |     |           LOCATE FOR PERIODICIDADES.CODPART = m.PERFAP
|     |     |     +-------IF !FOUND()
MENSAJE=MESSAGEBOX(" Periodicidades permitidas:
"+ALLTRIM(PER1)+','+ALLTRIM(PER2)+','+ALLTRIM(PER3)+','+ALLTRIM(PER4)+', Corrija por
favor... ',0+64+0," INFORMACION ")

                        THISFORM.PAGEFRAME1.PAGE2.BFINALIZAR.ENABLED=.F.
|     |     |     |     THISFORM.PAGEFRAME1.TABS = .F.
|     |     |     |     EXIT
|     |     |     +-------ELSE
|     |     |     |     THISFORM.PAGEFRAME1.TABS = .T.

                        THISFORM.PAGEFRAME1.PAGE2.BFINALIZAR.ENABLED=.T.
|     |     |     |         EXIT
|     |     |     +-------ENDIF
|     |     +-------ENDSCAN
|     +-------ENDIF
|
|
+-------ENDPROC
```

5.15 MANTENIMIENTO DE ESPECIFICACIONES

```
+-------PROCEDURE RECORDPOINTERMOVED
&&Definimnos variables para almacenar nombres de tablas actuales
|           LOCAL CALIAS
|           CALIAS=ALIAS()
|
&&Mostramos el codigo y nombre de fuente a partir del archivo
|           THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE = V_IPCMFTE1.CODFTE
```

```
                        THISFORM.PAGEFRAME1.PAGE3.CONTAINER2.CNOMBREFUENTE.VALUE
                        = V_IPCMFTE1.NOMBFTE
|
|           PASOFTE = V_IPCMFTE1.CODFTE
&&Cargamos en una variable el contenido del combo desplegable donde se cargaran
&&los articulos correspondientes al afuente seleccionada
|           LISTASELECCIONADA = THISFORM.PAGEFRAME1.PAGE3.COMBO1
|           FAPPAFTE = PASOFTE
&&Basados en los parametros reconsultamos la informacion de la vista
|           REQUERY('v_fappa')
|           SELECT V_FAPPA
|
&&Extractamos los articulos que tiene asignados la fuente requerida con la condicion
&&que no esten para eliminar y los cargamos en el combo desplegable
                        SELECT CODART FROM V_FAPPA WHERE (V_FAPPA.CODFTE =
                        THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE AND V_FAPPA.MARCA
                        = .F.) INTO CURSOR ARTICULOS
|           SELECT ARTICULOS
|           LISTASELECCIONADA.CLEAR
|           THISFORM.PAGEFRAME1.PAGE3.COMBO1.VALUE=" "
|     +-------SCAN
|     |       LISTASELECCIONADA.ADDITEM(ARTICULOS.CODART)
|     +-------ENDSCAN
|           USE
|           THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.TNOMART.VALUE=" "
|           THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.GRID1.REFRESH
|
|           THISFORM.PAGEFRAME1.PAGE3.COMBO1.REFRESH
|
&&Retomamos el alias que se estaba utilizando inicialmente
|           SELECT(CALIAS)
+-------ENDPROC
```

5.15.1 Seleccione el Nombre de la Fuente que desea Ubicar(CnombreFuente)

```
+-------PROCEDURE VALID
&&Asignamos el contenido del objeto combo donde quedaran cargados los articulos
&&correspondientes a la fuente
|           LISTASELECCIONADA = THISFORM.PAGEFRAME1.PAGE3.COMBO1
|
&&Seleccionamos el archivo de fuentes
&&Desplegamos en pantalla el codigo de la fuente actual
|           SELECT V_IPCMFTE1
|           THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE = V_IPCMFTE1.CODFTE
|           PASOFTE = V_IPCMFTE1.CODFTE
|
&&Reconsultamos la informacion de la vista para traer los articulos que le corresponden a la fuente
actual
```

```
|           FAPPAFTE = PASOFTE
|           REQUERY('v_fappa')
|           SELE V_FAPPA
&&Cargamos los codigos de los articulos correspondiente a la fuente actual siempre y cuando
&&no esten marcados para eliminar
|           SELECT CODART;
|                 FROM V_FAPPA;
                            WHERE (V_FAPPA.CODFTE =
                            THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE AND
                            V_FAPPA.MARCA = .F.) INTO CURSOR ARTICULOS
|           SELECT ARTICULOS
|           LISTASELECCIONADA.CLEAR
|           THISFORM.PAGEFRAME1.PAGE3.COMBO1.VALUE=" "
&&Actualizamos informacion en pantalla
|           THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.TNOMART.VALUE=" "
|           THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.GRID1.REFRESH
|     +-------SCAN
|     |           LISTASELECCIONADA.ADDITEM(ARTICULOS.CODART)
|     +-------ENDSCAN
|
|           USE
|           THISFORM.REFRESH
+-------ENDPROC
```

5.15.2 Boton Siguiente(Bsiguiente)

```
+-------PROCEDURE CLICK
&&A la variable le asignamos el contenido del objeto
|           LISTASELECCIONADA = THISFORM.PAGEFRAME1.PAGE3.COMBO1
|
&&Verificamos si la posicion del articulo actual en el archivo
&&es menor que el total de articulos de esta fuente
|     +-------IF LISTASELECCIONADA.LISTITEMID < LISTASELECCIONADA.LISTCOUNT
&&En caso afirmativo avanzamo 1 registro en el archivo
                        LISTASELECCIONADA.LISTITEMID = LISTASELECCIONADA.LISTITEMID+1
|     |
&&Asignamos el codigo del articulo actual a la variable
                        VESPARTICULO =
                        SUBSTR(THISFORM.PAGEFRAME1.PAGE3.COMBO1.LIST(LISTASELECCIO
                        NADA.LISTITEMID),1,7)
|     |
&&Asignamos a la variable el codigo de la fuente actual
|     |           VESPFUENTE = THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE
&&Actualizar el codigo del articulo en pantalla
|     |           THISFORM.PAGEFRAME1.PAGE3.COMBO1.VALUE = VESPARTICULO
|     |
&&Seleccionamos el archivo de articulos y ubicamos
&&el codigo actual para mostrar incorrespondiente a este
```

```
|     |          SELECT V_IPCMART
|     |          LOCATE FOR V_IPCMART.CODART = VESPARTICULO

                         THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.TNOMART.
                         VALUE = V_IPCMART.NOMART
                         THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.TNOMART.
                         REFRESH
|     |
&&Asignamos codigo de fuente y de articulo a variables
|     |          RESPAFTE = VESPFUENTE
|     |          RESPART = VESPARTICULO
|     |
|     |          VFAPCODFTE = VESPFUENTE
|     |          VFAPCODART = VESPARTICULO
&&Enviamos parametros para que actualice informacion del articulo actual
|     |          SELECT 'VFap'
|     |          REQUERY('VFap')
&&Refrescamos informacion en pantalla
|     |          SELECT 'v_resppa'
|     |          REQUERY('v_resppa')
|     |          SET FILTER TO V_RESPPA.MARCA = .F.
|     |          THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.GRID1.REFRESH
|     +-------ELSE
&&En caso de que la posicion del articulo en el archivo
&&sea mayor no se mueve al siguiente registro
|     |          LISTASELECCIONADA.LISTITEMID=LISTASELECCIONADA.LISTCOUNT
|     |          THISFORM.PAGEFRAME1.PAGE3.BFINALIZAR.ENABLED=.T.
|     +-------ENDIF
+-------ENDPROC
```

5.15.3 Columna Codigo Especificacion (Column1)

```
+-------PROCEDURE COLUMN1.INIT
&&En caso de no haber artriculo seleccionado mostramos un label con capti ''
&&para no ir a mostrar informacion que no correspponda a informacion que no es
                THIS.DYNAMICCURRENTCONTROL =
                "IIF(thisform.pageframe1.page3.combo1.value=' ','embossedlabel1', 'text1')"
+-------ENDPROC
```

5.15.4 Columna Nombre Especificacion (Column2)

```
+-------PROCEDURE COLUMN2.INIT
                THIS.DYNAMICCURRENTCONTROL =
                "IIF(thisform.pageframe1.page3.combo1.value=' ','embossedlabel1', 'text1')"
&&Validamos que se haya seleccionado codigo de articulo para mostrar o
&&no mostrar sus especificaciones
+-------ENDPROC
```

5.15.5 Columna Detalle Especificacion (Column3)

```
+-------PROCEDURE COLUMN3.INIT
&&En caso de no haber artriculo seleccionado mostramos un label con capti ''
&&para no ir a mostrar informacion que no correspponda a informacion que no es
                THIS.DYNAMICCURRENTCONTROL =
                "IIF(thisform.pageframe1.page3.combo1.value=' ','embossedlabel1', 'text1')"
+-------ENDPROC
```

5.15.6 Columna del Codigo Especificacion (Column1)

```
+-------PROCEDURE RIGHTCLICK
&&Procedimiento para intercambiar codigos de espcificacion
&&La variable publica unavez se coloca en verdadero cuando
&&le hacen clic derecho sobre alguno de los codigo de
&&espcificaciones perteneciente al articulo actual
|     +-------IF UNAVEZ
&&Seleccionamos el archivo
|     |         SELE V_RESPPA
&&Cargamos a la variable el codigo de especificacion que se le ha hecho clic derecho
|     |         NUEVOVEZ=THIS.VALUE
&&y  almacenamos el numero de registro
|     |         REGNUEVO=RECNO()
&&NOs vamos al registro donde inicalmente se hizo clic derecho
&&y temporalmente le asignamos codigo de especificacion = 99
&&para que nos presente conflicto al utilizar el mismo codigo
&&de especificacion para el mismo articulos mas de una vez
|     |         GO REGUNAVEZ
|     |         REPLACE V_RESPPA.CODESP WITH "99"
&&Luego vamos al registro que acabamos de tomar el codigo de
&&especificacion y le reemplazamos el codigo de esp con el de
&&la especificacion inicalmente seleccionada
|     |         GO REGNUEVO
|     |         REPLACE V_RESPPA.CODESP WITH TVEZ
&&Y por ultimo vamos al que le habiamos colocado 99 como codigo de esp
&&para colocarle el codigo de la otra especificacion seleccionada para intercambiar
|     |         GO REGUNAVEZ
|     |         REPLACE V_RESPPA.CODESP WITH NUEVOVEZ
&&Reinicializamos el valor de las variables para que empiece de nuevo
|     |         UNAVEZ=.F.
|     |         STORE 0 TO NUEVOVEZ, REGNUEVO
|     +-------ELSE
&&Esta es la primer vez que se hace clic derecho sobre algun codigo de especificacion
&&y cargamos el codigo de espcificacion y el numero de registro
|     |         SELE V_RESPPA
|     |         UNAVEZ=.T.
|     |         TVEZ=THIS.VALUE
|     |         REGUNAVEZ=RECNO()
```

```
|       +-------ENDIF
&&Confirmamos la actualizacion
|               TABLEUPDATE(.T.)
|
+-------ENDPROC
```

5.15.7 Boton Adicionar(Binsertar)

```
+-------PROCEDURE CLICK
&&Seleccionamos el archivo de especificaciones Consultamos el maximo
&&codigo de especificaciones correspondiente a la llave fuente - articulo actual
|               RESPAFTE = THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE
|               RESPART = THISFORM.PAGEFRAME1.PAGE3.COMBO1.VALUE
|
|               SELECT V_RESPPA
|               SET FILTER TO
|
                                SELECT MAX(CODESP) AS MAXESP FROM V_RESPPA WHERE ((CODFTE
                                = VFAP.CODFTE) AND (CODART = VFAP.CODART)) AND
                                V_RESPPA.MARCA = .F. INTO CURSOR MAXIMO
|               SELECT MAXIMO
&&Revisamos que el resultado de la consulta sea mayor que cero
|       +-------IF RECCOUNT() >0
&&En caso afirmativo le sumamos 1 al valor resultante para el nuevo
&&codigo de especificacion
|       |               XX=VAL(MAXIMO.MAXESP)+1
|       +-------ELSE
&&En caso contrario le asignamos 1 al nuevo codigo de especificaciones
|       |               XX=1
|       +-------ENDIF
|               SELE MAXIMO
|               USE
&&Cargamos en la variable el nuevo codigo de especificacion convertido a caracter
|               NUEVOVALOR=ALLTRIM(STR(XX,2,0))
|               SELECT V_RESPPA
&&Configuramos el ambiente de trabajo para que los registros marcados para borrar
&&sean tenidos en cuenta en la busqueda
|
&&Seleccionamos el archivo de especificaciones y localizamos la informacion
&&correspondiente a la llave fuente - articulo - especificacion
                                =SEEK(VFAP.CODFTE+VFAP.CODART+PADL(NUEVOVALOR, 2,
                                '0'),'v_resppa','pkresp')
|       +-------IF FOUND()
&&En caso de encontrarlo desmarca el registro para que no sea borrado
REPLACE V_RESPPA.MARCA WITH .F., V_RESPPA.NOMBDET WITH '', V_RESPPA.NOMESP
WITH '' FOR (V_RESPPA.CODFTE = THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE) AND
(V_RESPPA.CODART = THISFORM.PAGEFRAME1.PAGE3.COMBO1.VALUE) AND
(V_RESPPA.CODESP = PADL(NUEVOVALOR, 2, '0'))
```

```
&&Deja en blanco el contenido de los nombres de especificacion y de detalle
|       |
|       +-------ELSE
&&En caso de no encontrarlo inserta la informacion al archivo
|       |               INSERT INTO V_IPCRESP(CODFTE, CODART, CODESP, MARCA) ;
VALUES (THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE,
THISFORM.PAGEFRAME1.PAGE3.COMBO1.VALUE, PADL(NUEVOVALOR, 2, '0'), .F.)
|       |
|       +-------ENDIF
&&Configuramos de nuevo el ambiente de trabajo para que los registros marcados
&&para eliminar no sean tenidos en cuenta al realizar una busqueda
|               SELE V_IPCRESP
|               TABLEUPDATE(.T.)
|               REQUERY('v_resppa')
|               SELEC V_RESPPA
|               SET FILTER TO V_RESPPA.MARCA = .F.
|
&&Refrescamos la informacion en pantalla
|       +-------WITH THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.GRID1
|       |               .SETFOCUS
|       |               .REFRESH
|       +-------ENDWITH
+-------ENDPROC
```

5.15.8 Boton Anterior(Banterior)

```
+-------PROCEDURE CLICK
&&Cargamos a la variable el contenido del objeto
|               LISTASELECCIONADA = THISFORM.PAGEFRAME1.PAGE3.COMBO1
|
&&Verificamos si la posicion del articulo actual es mayor que uno
|       +-------IF LISTASELECCIONADA.LISTITEMID>1
&&En caso afirmativo retrocedemos 1 registro en el archivo
                                LISTASELECCIONADA.LISTITEMID = LISTASELECCIONADA.LISTITEMID-1
&&A la variable le asignamos el codigo del articulo
                                VESPARTICULO =
                                SUBSTR(THISFORM.PAGEFRAME1.PAGE3.COMBO1.LIST(LISTASELECCIO
                                NADA.LISTITEMID),1,7)
&&A la variable le asignamos  el codigo de la fuente actual
|       |               VESPFUENTE = THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE
&&Actualizamos la informacion en pantalla del codigo del articulo
|       |               THISFORM.PAGEFRAME1.PAGE3.COMBO1.VALUE = VESPARTICULO
|       |
&&Seleccionamos el archivo de articulos, ubicamos el actual y mostramos
&&en pantalla la informacion correspondiente a el mismo
|       |               SELECT V_IPCMART
|       |               LOCATE FOR V_IPCMART.CODART = VESPARTICULO
```

```
THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.TNOMART.VALUE =
V_IPCMART.NOMART

THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.TNOMART.REFRESH
|       |
&&Asignamos los valores de codigo de fuente y de articulo
&&a las variables
|       |
|       |           RESPAFTE = VESPFUENTE
|       |           RESPART = VESPARTICULO
|       |
|       |           VFAPCODFTE = VESPFUENTE
|       |           VFAPCODART = VESPARTICULO
|       |
&&Enviamos parametros a la vista parametrizada para que nos traiga la
&&informacion correspondiente a la llave fuente - articulo
|       |           SELECT 'VFap'
|       |           REQUERY('VFap')
&&Refrescamos la informacion en pantalla
|       |           SELECT 'v_resppa'
|       |           REQUERY('v_resppa')
|       |           SET FILTER TO V_RESPPA.MARCA = .F.
|       |           THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.GRID1.REFRESH
|       |
|       +-------ENDIF
|
+-------ENDPROC
```

5.15.9 Boton Eliminar(Command1)

```
+-------PROCEDURE CLICK
|           SELECT V_RESPPA
|           REPLACE V_RESPPA.MARCA WITH .T.
|           TABLEUPDATE(.T.)
&&Marcamos para Eliminarel registro actual por medio del campo marca
&&Refrescamos pantalla
|           THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.GRID1.REFRESH
|           THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.GRID1.SETFOCUS
+-------ENDPROC
```

5.15.10 Boton Cerrar(Bcancelar)

```
+-------PROCEDURE CLICK
|           SET SYSMENU ON
|           THISFORM.RELEASE
+-------ENDPROC
```

5.15.11 Seleccionar Codigo Articulo (Combo1)

```
+-------PROCEDURE INTERACTIVECHANGE
|           SELECT V_IPCMART
&&Seleccionamos el archivo de articulos y localizamos el codigo
&&seleccionado por el usuario y mostramos informacion correspondiente a este
|           LOCATE FOR V_IPCMART.CODART = THIS.VALUE
                        THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.TNOMART.VALUE
                        = V_IPCMART.NOMART
|           THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.TNOMART.REFRESH
|
&&Cargamos las variables con los codigo de fuente y de articulo para enviarlos como parametro
&&y consultar la informacion correspondiente a estos
|           RESPAFTE = THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE
|           RESPART = THIS.VALUE
|           REQUERY('v_resppa')
|           SELECT V_RESPPA
&&Luego de reconsultar la informacion aplicamos el filtro para que solo nos muestre la informacion
&&que no esta marcada para eliminar
|           SET FILTER TO V_RESPPA.MARCA = .F.
&&Asignamos a variables el codigo de la fuente  y del articulo
&&los enviamos como parametros a la vista parametrizada
&&para que nos despliegue la informaciion correspondiente
|           VFAPCODFTE = THISFORM.PAGEFRAME1.PAGE3.ECODFTE.VALUE
|           VFAPCODART = THIS.VALUE
|           SELECT 'vfap'
|           REQUERY('vfap')
|
|           SELECT V_RESPPA
|           THISFORM.REFRESH
|
+-------ENDPROC
```

5.15.12 Digite el código de la fuente que desea ubicar(EcodFte)

```
+-------PROCEDURE LOSTFOCUS
&&Asignamos a una variable el contenido del objeto combo desplegable que contendra
&&los articulos correspondientes a la fuente
|           LISTASELECCIONADA = THISFORM.PAGEFRAME1.PAGE3.COMBO1
|
&&Seleccionamos el archivo de fuentes y almacenamos en una variable
&&el numero del registro actual
|
|           SELECT V_IPCMFTE1
|           MARCAREGISTRO=RECNO()
|
|           OFTEPAFTE = THIS.VALUE
&&Basados en el parametro consultamos la informacion correspondiente
```

```
|           REQUERY('v_oftepa')
|           SELECT V_OFTEPA
&&Dependiendo del resultado de la consulta sabemos si la fuente exite o no
&&Si no existe
|     +-------IF RECCOUNT() = 0
&&Verificamos que el consecutivo digitado sea consistente, segun el resultado
&&informamos si existe o no en caso de que no exista preguntamos si desea ingresarla
|     |     +-------IF SUBSTR(ALLTRIM(THIS.VALUE),17,4) <> "0000"
INGRESAR=MESSAGEBOX('Fuente no encontrada, desea ingresarla ??',4+32+0,'Información')
|     |     |     +-------DO CASE
&&En caso de aceptar ingresarla lo llevamos a la pantalla correspondiente donde pueda
&&ingresar la fuente
|     |     |     +-------CASE INGRESAR = 6
|     |     |     |       THISFORM.PAGEFRAME1.PAGES(1).SETFOCUS
                  THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE=THIS.VALUE
|     |     |     |       PASOFTE = THIS.VALUE

                  THISFORM.PAGEFRAME1.PAGE1.ECODFTE.SETFOCUS

                  THISFORM.PAGEFRAME1.PAGE1.ECODFTE.REFRESH

                  THISFORM.PAGEFRAME1.PAGE1.ECODFTE.LOSTFOCUS
&&Nos devolvemos al registro almacenado inicialmente
|     |     |     |           SELECT V_IPCMFTE1
|     |     |     |           GO MARCAREGISTRO
|     |     |     |           THIS.VALUE = V_IPCMFTE1.CODFTE
|     |     |     +-------CASE INGRESAR = 7
&&En caso de no aceptar ingresarla Nos devolvemos al registro almacenado inicialmente
|     |     |     |           SELECT V_IPCMFTE1
|     |     |     |           GO MARCAREGISTRO
|     |     |     |           THIS.VALUE = V_IPCMFTE1.CODFTE
|     |     |     +-------ENDCASE
|     |     +-------ELSE
&&En caso de no ser encontrado retornamos al registro donde se estaba inicialmente
&&y actualizamos el codigo de la fuente en pantalla
|     |     |           SELECT V_IPCMFTE1
|     |     |           GO MARCAREGISTRO
|     |     |           THIS.VALUE = V_IPCMFTE1.CODFTE
|     |     +-------ENDIF
|     +-------ELSE
|     |           SELECT V_OFTEPA
|     |           PASOFTE = V_OFTEPA.CODFTE
THISFORM.PAGEFRAME1.PAGE3.CONTAINER2.CNOMBREFUENTE.VALUE =
V_OFTEPA.NOMBFTE

THISFORM.PAGEFRAME1.PAGE3.CONTAINER2.CNOMBREFUENTE.REFRESH
|     |
&&Seleccionamos los articulos que corresponden a la fuente seleccionada
```

```
|       |           FAPPAFTE = PASOFTE
|       |           REQUERY('v_fappa')
|       |           SELE V_FAPPA
|       |           SELECT CODART;
|       |                 FROM V_FAPPA;
WHERE (V_FAPPA.CODFTE = THIS.VALUE AND V_FAPPA.MARCA = .F.) INTO CURSOR
ARTICULOS

|       |           SELECT ARTICULOS
|       |           LISTASELECCIONADA.CLEAR
|       |
|       |           THISFORM.PAGEFRAME1.PAGE3.COMBO1.VALUE=" "
&&Cargamos la lista con el resultado de la seleccion anterior
|       |     +-------SCAN
|       |     |           LISTASELECCIONADA.ADDITEM(ARTICULOS.CODART)
|       |     +-------ENDSCAN
|       |
|       |           USE
|       +-------ENDIF
&&Refrescamos la informacion en pantalla
|           THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.TNOMART.VALUE=" "
|           THISFORM.PAGEFRAME1.PAGE3.PAGEFRAME1.PAGE2.GRID1.REFRESH
|
|           THISFORM.REFRESH
|
+-------ENDPROC

+-------PROCEDURE INIT
&&Asignamos a este campos el codigo de la fuente
|           THIS.VALUE = V_IPCMFTE1.CODFTE
+-------ENDPROC
```

5.15.13 Seleccionar codigo Articulo desde la cuadricula (Text1)

```
+-------PROCEDURE RIGHTCLICK
&&Coloca en el combo el codigo de articulo seleccionado en esta cuadricula
&&y ejecuta el evento lostfocus para que muestre la informacion de precio
&&y cantidad anterior
|           THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE=THIS.VALUE
|           THISFORM.PAGEFRAME1.PAGE4.COMBO1.LOSTFOCUS
+-------ENDPROC

+-------PROCEDURE VALID
&&Validamos que se haya digitado toda la informacion para habilitar
&&o inhabilitar el boton de almacenar informacion
|       +-------IF THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE<>0 AND;
|       |                 THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE<>0 AND;
|       |                 THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE<>0 AND;
```

```
|       |                 THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE<>0
|       |
|       |         THISFORM.PAGEFRAME1.PAGE4.GUARDAR.ENABLED=.T.
|       +-------ELSE
|       |         THISFORM.PAGEFRAME1.PAGE4.GUARDAR.ENABLED=.F.
|       +-------ENDIF
|
|
+-------ENDPROC
```

5.15.14 Seleccione el Código del Articulo(Combo1)

```
+-------PROCEDURE INTERACTIVECHANGE
|           SELE V_IPCMART
&&Si es encontrado traemos el nombre del mismo
|           LOCATE FOR V_IPCMART.CODART = THIS.VALUE
THISFORM.PAGEFRAME1.PAGE4.TEXT10.CONTROLSOURCE = 'v_ipcmart.nomart'
|           THISFORM.PAGEFRAME1.PAGE4.TEXT10.REFRESH
|           THISFORM.PAGEFRAME1.PAGE4.TEXT4.SETFOCUS
|
|           THISFORM.PAGEFRAME1.PAGE4.TEXT10.CONTROLSOURCE = ' '
|
+-------ENDPROC

+-------PROCEDURE GOTFOCUS
&&Validamos la informacion de codigos fuente y articulo ingresada
                +-------IF LEN(ALLTRIM(THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE)) < 20 OR
                SUBSTR(ALLTRIM(THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE),17,4) = "0000"
MENSAJE=MESSAGEBOX(" Recuerde que el codigo de la fuente debe tener 20 digitos,corrija por
favor",0+32+0," Actualizar Fuentes ")
|       |         THISFORM.PAGEFRAME1.PAGE4.TEXT1.SETFOCUS
|       +-------ENDIF
+-------ENDPROC

+-------PROCEDURE LOSTFOCUS
&&Validamos que la informacion digitada sea valida
|       +-------IF !EMPTY(THIS.VALUE)
&&Buscamos el codigo de fuente y el codigo de articulo digitados
&&en el archivo de fuente - articulo
|       |         FAPPAFTE = THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE
|       |         FAPPART = THIS.VALUE
|       |         REQUERY('v_fappart')
|       |         SELE V_FAPPART
|       |     +-------IF RECCOUNT() > 0
&&Si encontramos la coincidencia fuente - articulo buscamos en movimiento
|       |     |         THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY = .F.
|       |     |         THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY = .F.
|       |     |
```

```
THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE = V_FAPPART.PANART
|     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT4.REFRESH
|     |     |
THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE = V_FAPPART.CANTANT
|     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT6.REFRESH
|     |     |
|     |     |          SELE V_IPCMART
&&Si es encontrado traemos el nombre del mismo
|     |     |          LOCATE FOR V_IPCMART.CODART = THIS.VALUE

                         THISFORM.PAGEFRAME1.PAGE4.TEXT10.CONTROLSOURCE =
                         'v_ipcmart.nomart'

|     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT10.REFRESH
|     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT4.SETFOCUS
|     |     |
|     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT10.CONTROLSOURCE = ' '
|     |     |
|     |     |
&&Buscamos la fuente articulo en movimiento
|     |     |          FTEMVT = THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE
|     |     |          ARTMVT = THIS.VALUE
|     |     |          REQUERY('v_fteartmvt')
&&Si se encuentra la informacion habilitamos los objetos donde debemos mostrar la informacion
&&que se le ha capturado y actualizamos la informacion en pantalla
|     |     |     +-------IF RECCOUNT() > 0
THISFORM.PAGEFRAME1.PAGE4.TEXT2.READONLY = .F.

THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY = .F.

THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY = .F.

THISFORM.PAGEFRAME1.PAGE4.TEXT7.READONLY = .F.
&&Precio y cantidad anterior
THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE = V_FTEARTMVT.PANART
|     |     |     |
THISFORM.PAGEFRAME1.PAGE4.TEXT4.REFRESH
THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE = V_FTEARTMVT.CANTANT
|     |     |     |
THISFORM.PAGEFRAME1.PAGE4.TEXT6.REFRESH
&&Precio y cantidad actual
THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE = V_FTEARTMVT.PACTMVT
|     |     |     |
THISFORM.PAGEFRAME1.PAGE4.TEXT2.REFRESH
|     |     |     |
THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE = V_FTEARTMVT.CANRMVT
|     |     |     |
THISFORM.PAGEFRAME1.PAGE4.TEXT7.REFRESH
```

```
|     |     |     |
&&Si la informacion no se encuentra en movimiento, se solicita sea ingresada
&&Esta informacion primero se valida que exista la fuente, luego que exista
&&la llave fuente articulos para que se pueda ingresar en movimiento
|     |     |     +-------ELSE
= MESSAGEBOX(" La fuente con su articulo no existen en el MOVIMIENTO, Ingrese la informacion
por favor ",0+64+0," Actualizar Movimiento")

THISFORM.PAGEFRAME1.PAGE4.TEXT2.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY=.F.
THISFORM.PAGEFRAME1.PAGE4.TEXT7.READONLY=.F.
|     |     |     |
THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE=0
|     |     |     |       THISFORM.PAGEFRAME1.PAGE4.TEXT2.REFRESH
|     |     |     |       THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE=0
|     |     |     |       THISFORM.PAGEFRAME1.PAGE4.TEXT7.REFRESH
|     |     |     |
|     |     |     +-------ENDIF
|     |     |
&&Realizamos la validacion que exista en fuente articulo
&&en caso de no existir retornamos automaticamente a la 'page'
&&es decir, donde debe ingresar la fuente articulo
|     |     +-------ELSE
INGRESAR = MESSAGEBOX('" La fuente con su articulo no fueron encontrados en FUENTE -
ARTICULO, desea ingresarla ??',4+32+0,'Información')
|     |     |     +-------DO CASE
|     |     |     +-------CASE INGRESAR = 6
                        PASOFTE = THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE
|     |     |     |
                        THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE = 0
|     |     |     |       THISFORM.PAGEFRAME1.PAGE4.TEXT4.REFRESH

                        THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE = 0
|     |     |     |       THISFORM.PAGEFRAME1.PAGE4.TEXT6.REFRESH
&&Precio y cantidad actual
|     |     |     |       THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE = 0
|     |     |     |       THISFORM.PAGEFRAME1.PAGE4.TEXT2.REFRESH
|     |     |     |
                        THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE = 0
|     |     |     |       THISFORM.PAGEFRAME1.PAGE4.TEXT7.REFRESH
|     |     |     |
                        THISFORM.PAGEFRAME1.PAGE4.COMBO1.VISIBLE = .F.
|     |     |     |
|     |     |     |       THISFORM.PAGEFRAME1.PAGES(2).SETFOCUS
|     |     |     |       THISFORM.PAGEFRAME1.PAGES(2).ACTIVATE
```

```
                    THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALUE =
                    THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE

                    THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.REFRESH

THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.LOSTFOCUS
|     |     |     |
|     |     |     +-------CASE INGRESAR=7

                    THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE=ALLTRIM(DEPARTAMENT
                    O)+ALLTRIM(MUNICIPIO)
|     |     |     |     THISFORM.PAGEFRAME1.PAGE4.TEXT10.VALUE=" "
|     |     |     |     THISFORM.PAGEFRAME1.PAGE4.TEXT10.REFRESH

                    THISFORM.PAGEFRAME1.PAGE4.TEXT2.READONLY=.T.

                    THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY=.T.

                    THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY=.T.

                    THISFORM.PAGEFRAME1.PAGE4.TEXT7.READONLY=.T.

                    THISFORM.PAGEFRAME1.PAGE4.COMBO1.SETFOCUS

                    THISFORM.PAGEFRAME1.PAGE4.GUARDAR.ENABLED=.F.

|     |     |     +-------ENDCASE
|     |     +-------ENDIF
|     +-------ENDIF
+-------ENDPROC
```

5.15.15 Boton Salir

```
+-------PROCEDURE CLICK
&&Volver a mostrar el menu y salir de esta pantalla
|           SET SYSMENU ON
|           THISFORM.RELEASE
|
+-------ENDPROC
```

5.15.16 Boton Cancelar

```
+-------PROCEDURE CLICK
&&Limpiamos pantalla y habilitamos e inhabilitamos objetos
|           THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE=" "
|           THISFORM.PAGEFRAME1.PAGE4.COMBO1.REFRESH
|           THISFORM.PAGEFRAME1.PAGE4.COMBO1.ENABLED=.F.
|
```

```
|           THISFORM.PAGEFRAME1.PAGE4.TEXT10.VALUE=" "
|           THISFORM.PAGEFRAME1.PAGE4.TEXT10.REFRESH
|
|           THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE=0
|           THISFORM.PAGEFRAME1.PAGE4.TEXT4.REFRESH
|
|           THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE=0
|           THISFORM.PAGEFRAME1.PAGE4.TEXT6.REFRESH
|
|           THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE=0
|           THISFORM.PAGEFRAME1.PAGE4.TEXT2.REFRESH
|
|           THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE=0
|           THISFORM.PAGEFRAME1.PAGE4.TEXT7.REFRESH
|
|
|           THISFORM.PAGEFRAME1.PAGE4.TEXT2.READONLY=.T.
|           THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY=.T.
|           THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY=.T.
|           THISFORM.PAGEFRAME1.PAGE4.TEXT7.READONLY=.T.
&&MOstramos los 5 primeros digitos de la fuente segun la ciudad
        |
        THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE=ALLTRIM(DEPARTAMENTO)+ALLTRIM(
        MUNICIPIO)
|           THISFORM.PAGEFRAME1.PAGE4.TEXT1.REFRESH
|           THISFORM.PAGEFRAME1.PAGE4.TEXT1.SETFOCUS
+-------ENDPROC
```

5.15.17 Boton Guardar

```
+-------PROCEDURE CLICK
&&Validamos la informacion ingresada tanto de codigo fuente y articulo
                +-------IF LEN(ALLTRIM(THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE)) < 20 OR
                LEN(ALLTRIM(THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE)) < 7
                        = MESSAGEBOX(" Informacion Incompleta.... ",0+48+0," Corrija por favor ")
|     |       THISFORM.PAGEFRAME1.PAGE4.TEXT1.SETFOCUS
&&En caso de ser valida
|     +-------ELSE
|     |       LOCAL CANO, CMES, CDEC, CONSECUTIVO
|     |
                        FTEMVTPA = ALLTRIM(THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE)
|     |       REQUERY('v_ftemvtpa')
|     |       SELE V_FTEMVTPA
&&Buscamos en movimiento para ver si existe la llave fuente - articulo
|     |       +-------IF RECCOUNT() > 0
|     |       |                       CONSECUTIVO = V_FTEMVTPA.IMPCTV
|     |       |           CANO = V_FTEMVTPA.ANO
|     |       |           CMES = V_FTEMVTPA.MES
```

```
|     |     |         CDEC = V_FTEMVTPA.DECMVT
|     |     |
FTEMVT = ALLTRIM(THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE)
ARTMVT = ALLTRIM(THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE)
|     |     |         REQUERY('v_fteartmvt')
|     |     |         SELE V_FTEARTMVT
|     |     |     +-------IF RECCOUNT() > 0
&&Si se encuentra reemplazamos la informacion que tenia con la nueva ingresada
REPLACE V_FTEARTMVT.PANART WITH THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE;
V_FTEARTMVT.CANTANT WITH THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE;
V_FTEARTMVT.CANRMVT WITH THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE;
V_FTEARTMVT.PACTMVT WITH THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE

&&Validamos que la cantidad recolectada y la antidad anterior sean ambas diferentes de 0
                |                         +-------IF THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE <>
                                          0 AND THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE <> 0
                                          AND;
                                                THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE <> 0
                                                AND THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE
                                                <> 0
&&Cargamos el campo total con el numero del dia actual
&&para utilizarlo para el reporte de informacion capturada diaria
|     |     |     |     |     NFECHA=DTOC(DATE())
|     |     |     |     |     NFECHA=SUBSTR(NFECHA,1,2)
|     |     |     |     |     NTOTAL=VAL(NFECHA)
REPLACE V_FTEARTMVT.TOTAL WITH NTOTAL
|     |     |     |     +-------ENDIF
|     |     |     |
|     |     |     |         TABLEUPDATE(.T.)
|     |     |     |         SELE V_IPCTMVT
|     |     |     |         TABLEUPDATE(.T.)
&&Limpiamos variables
|     |     |     |
|     |     |     |         THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE=" "
|     |     |     |         THISFORM.PAGEFRAME1.PAGE4.COMBO1.REFRESH
                        THISFORM.PAGEFRAME1.PAGE4.COMBO1.ENABLED=.F.
|     |     |     |
|     |     |     |         THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE=0
|     |     |     |         THISFORM.PAGEFRAME1.PAGE4.TEXT4.REFRESH
|     |     |     |         THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE=0
|     |     |     |         THISFORM.PAGEFRAME1.PAGE4.TEXT6.REFRESH
|     |     |     |         THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE=0
|     |     |     |         THISFORM.PAGEFRAME1.PAGE4.TEXT2.REFRESH
|     |     |     |         THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE=0

|     |     |     |         THISFORM.PAGEFRAME1.PAGE4.TEXT7.REFRESH
|     |     |     |
                  THISFORM.PAGEFRAME1.PAGE4.TEXT2.READONLY=.T.
```

```
            THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY=.T.

            THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY=.T.

            THISFORM.PAGEFRAME1.PAGE4.TEXT7.READONLY=.T.

&&Informamos que se hizo con la informacion ingresada
WAIT WIND " Información Reemplazada " AT 200,400 TIMEOUT 1
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT1.SETFOCUS
|     |     |     |
CAMBIO = MESSAGEBOX(" Cambia de Fuente ?",4+32+0," Mantenimiento")
&&Preguntamos si cambia de fuente
|     |     |     |     +-------DO CASE
|     |     |     |     +-------CASE CAMBIO = 6
&&Si si cambia dejamos los 4 primeros digitos de las fuentes
THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE=" "

THISFORM.PAGEFRAME1.PAGE4.COMBO1.REFRESH
THISFORM.PAGEFRAME1.PAGE4.COMBO1.ENABLED=.F.

THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE=0
THISFORM.PAGEFRAME1.PAGE4.TEXT4.REFRESH
THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE=0
THISFORM.PAGEFRAME1.PAGE4.TEXT6.REFRESH
THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE=0
THISFORM.PAGEFRAME1.PAGE4.TEXT2.REFRESH
THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE=0
THISFORM.PAGEFRAME1.PAGE4.TEXT7.REFRESH
THISFORM.PAGEFRAME1.PAGE4.TEXT2.READONLY=.T.
THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY=.T.
THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY=.T.
THISFORM.PAGEFRAME1.PAGE4.TEXT7.READONLY=.T.
THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE =
ALLTRIM(DEPARTAMENTO)+ALLTRIM(MUNICIPIO)

THISFORM.PAGEFRAME1.PAGE4.TEXT1.REFRESH
THISFORM.PAGEFRAME1.PAGE4.TEXT1.SETFOCUS
|     |     |     |     +-------CASE CAMBIO = 7
&&Si no, mantenemos el codigo de la fuente que se estaba trabajando
THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE=" "
THISFORM.PAGEFRAME1.PAGE4.COMBO1.REFRESH
THISFORM.PAGEFRAME1.PAGE4.COMBO1.ENABLED=.F.

THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE=0
THISFORM.PAGEFRAME1.PAGE4.TEXT4.REFRESH

THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE=0
```

```
THISFORM.PAGEFRAME1.PAGE4.TEXT6.REFRESH

THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE=0
THISFORM.PAGEFRAME1.PAGE4.TEXT2.REFRESH

THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE=0
THISFORM.PAGEFRAME1.PAGE4.TEXT7.REFRESH

THISFORM.PAGEFRAME1.PAGE4.TEXT2.READONLY=.T.
THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY=.T.
THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY=.T.
THISFORM.PAGEFRAME1.PAGE4.TEXT7.READONLY=.T.

THISFORM.PAGEFRAME1.PAGE4.TEXT1.SETFOCUS
|     |     |     |     +-------ENDCASE
|     |     |     +-------ELSE
&&Si no se encuentra la informacion en el archivo de movimiento
&&la ingresamos como nueva
INSERT INTO V_FTEARTMVT(CODFTE, CODART, PANART, CANTANT, PACTMVT, CANRMVT,
IMPCTV, ANO, MES, DECMVT);

VALUE(THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE,
THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE,
THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE,
THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE,
THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE,
THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE, CONSECUTIVO, CANO, CMES, CDEC)

                                        +-------IF
                                        THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE <> 0
                                        AND
                                                THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALU
                                        E <> 0 AND;

THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE <> 0 AND
THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE <> 0
&&Cargamos el campo total con el numero del dia actual
&&para utilizarlo para el reporte de informacion capturada diaria
|     |     |     |     |           NFECHA=DTOC(DATE())
|     |     |     |     |           NFECHA=SUBSTR(NFECHA,1,2)
|     |     |     |     |           NTOTAL=VAL(NFECHA)
REPLACE V_FTEARTMVT.TOTAL WITH NTOTAL
|     |     |     |     +-------ENDIF
|     |     |     |
|     |     |     |           TABLEUPDATE(.T.)
|     |     |     |           SELE V_IPCTMVT
|     |     |     |           TABLEUPDATE(.T.)
&&Limpiamos variables y refrescamos pantalla
```

```
|     |     |     |
|     |     |     |          THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE=" "
|     |     |     |          THISFORM.PAGEFRAME1.PAGE4.COMBO1.REFRESH

                                THISFORM.PAGEFRAME1.PAGE4.COMBO1.ENABLED=.F.
|     |     |     |
|     |     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE=0
|     |     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT4.REFRESH
|     |     |     |
|     |     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE=0
|     |     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT6.REFRESH
|     |     |     |
|     |     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE=0
|     |     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT2.REFRESH
|     |     |     |
|     |     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE=0
|     |     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT7.REFRESH
|     |     |     |
THISFORM.PAGEFRAME1.PAGE4.TEXT2.READONLY=.T.
THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY=.T.
THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY=.T.
THISFORM.PAGEFRAME1.PAGE4.TEXT7.READONLY=.T.

&&Informamos que se hizo con la informacion
WAIT WIND " Información Almacenada " AT 200,400 TIMEOUT 1
&&Preguntamos si cambia de fuente
CAMBIO = MESSAGEBOX(" Cambia de Fuente ?",4+32+0," Mantenimiento")
|     |     |     |     +-------DO CASE
|     |     |     |     +-------CASE CAMBIO = 6
&&Si si cambia dejamos los 5 primeros digitos de las fuentes
THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE=" "
THISFORM.PAGEFRAME1.PAGE4.COMBO1.REFRESH

THISFORM.PAGEFRAME1.PAGE4.COMBO1.ENABLED=.F.

THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE=0
THISFORM.PAGEFRAME1.PAGE4.TEXT4.REFRESH

THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE=0
THISFORM.PAGEFRAME1.PAGE4.TEXT6.REFRESH

THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE=0
THISFORM.PAGEFRAME1.PAGE4.TEXT2.REFRESH

THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE=0
THISFORM.PAGEFRAME1.PAGE4.TEXT7.REFRESH

THISFORM.PAGEFRAME1.PAGE4.TEXT2.READONLY=.T.
```

```
THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY=.T.

THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY=.T.

THISFORM.PAGEFRAME1.PAGE4.TEXT7.READONLY=.T.

THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE =
ALLTRIM(DEPARTAMENTO)+ALLTRIM(MUNICIPIO)
THISFORM.PAGEFRAME1.PAGE4.TEXT1.REFRESH

THISFORM.PAGEFRAME1.PAGE4.TEXT1.SETFOCUS

|     |     |     |     +-------CASE CAMBIO = 7

THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE=" "
THISFORM.PAGEFRAME1.PAGE4.COMBO1.REFRESH

THISFORM.PAGEFRAME1.PAGE4.COMBO1.ENABLED=.F.

THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE=0
THISFORM.PAGEFRAME1.PAGE4.TEXT4.REFRESH

THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE=0
THISFORM.PAGEFRAME1.PAGE4.TEXT6.REFRESH

THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE=0
THISFORM.PAGEFRAME1.PAGE4.TEXT2.REFRESH

THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE=0
THISFORM.PAGEFRAME1.PAGE4.TEXT7.REFRESH

THISFORM.PAGEFRAME1.PAGE4.TEXT2.READONLY=.T.
THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY=.T.
THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY=.T.
THISFORM.PAGEFRAME1.PAGE4.TEXT7.READONLY=.T.

&&Si no, mantenemos el codigo de la fuente que se estaba trabajando
THISFORM.PAGEFRAME1.PAGE4.TEXT1.SETFOCUS
|     |     |     |     +-------ENDCASE
|     |     |     +-------ENDIF
|     |     +-------ELSE
|     |     |           REQUERY('v_ipctctv')
|     |     |           SELE V_IPCTCTV
|     |     |           GO BOTTOM
|     |     |           CONSECUTIVO = V_IPCTCTV.FINCTV + 1
|     |     |           SKIP -1
|     |     |           CANO = V_IPCTCTV.AÑOCTV
```

```
|     |     |          CMES = V_IPCTCTV.MESCTV
|     |     |          CDEC = V_IPCTCTV.DECCTV
|     |     |
INSERT INTO V_FTEARTMVT(CODFTE, CODART, PANART, CANTANT, PACTMVT, CANRMVT,
IMPCTV, ANO, MES, DECMVT);

VALUE(THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE,
THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE,
THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE,
THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE,
THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE,
THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE, CONSECUTIVO, CANO, CMES, CDEC)
|     |     |
                                    +-------IF THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE <>
                                    0 AND THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE <> 0
                                    AND;
                                          THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE <> 0
                                          AND THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE
                                          <> 0
&&Cargamos el campo total con el numero del dia actual
&&para utilizarlo para el reporte de informacion capturada diaria
|     |     |     |          NFECHA=DTOC(DATE())
|     |     |     |          NFECHA=SUBSTR(NFECHA,1,2)
|     |     |     |          NTOTAL=VAL(NFECHA)
|     |     |     |          REPLACE V_FTEARTMVT.TOTAL WITH NTOTAL
|     |     |     +-------ENDIF
|     |     |
|     |     |          TABLEUPDATE(.T.)
|     |     |          SELE V_IPCTMVT
|     |     |          TABLEUPDATE(.T.)
|     |     |
|     |     |          REQUERY('v_ipctctv')
|     |     |          SELE V_IPCTCTV
|     |     |          GO BOTTOM
|     |     |          REPLACE V_IPCTCTV.FINCTV WITH CONSECUTIVO
|     |     |          TABLEUPDATE(.T.)
|     |     |
&&Limpiamos variables y refrescamos pantalla
|     |     |
|     |     |          THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE=" "
|     |     |          THISFORM.PAGEFRAME1.PAGE4.COMBO1.REFRESH
|     |     |          THISFORM.PAGEFRAME1.PAGE4.COMBO1.ENABLED=.F.
|     |     |
|     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE=0
|     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT4.REFRESH
|     |     |
|     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE=0
|     |     |          THISFORM.PAGEFRAME1.PAGE4.TEXT6.REFRESH
```

```
|     |     |
|     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE=0
|     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT2.REFRESH
|     |     |
|     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE=0
|     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT7.REFRESH
|     |     |
|     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT2.READONLY=.T.
|     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY=.T.
|     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY=.T.
|     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT7.READONLY=.T.
|     |     |
&&Informamos que se hizo con la informacion
                                    WAIT WIND " Información Almacenada " AT 200,400 TIMEOUT 1
&&Preguntamos si cambia de fuente
                                    CAMBIO = MESSAGEBOX(" Cambia de Fuente ?",4+32+0,"
                                    Mantenimiento")
|     |     |     +-------DO CASE
|     |     |     +-------CASE CAMBIO = 6
&&Si si cambia dejamos los 5 primeros digitos de las fuentes
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE=" "
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.COMBO1.REFRESH

THISFORM.PAGEFRAME1.PAGE4.COMBO1.ENABLED=.F.
|     |     |     |
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE=0
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT4.REFRESH
|     |     |     |
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE=0
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT6.REFRESH
|     |     |     |
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE=0
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT2.REFRESH
|     |     |     |
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE=0
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT7.REFRESH
|     |     |     |
THISFORM.PAGEFRAME1.PAGE4.TEXT2.READONLY=.T.

THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY=.T.

THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY=.T.

THISFORM.PAGEFRAME1.PAGE4.TEXT7.READONLY=.T.
|     |     |     |
THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE=ALLTRIM(DEPARTAMENTO)+ALLTRIM(MUNIC
IPIO)
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT1.REFRESH
```

```
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT1.SETFOCUS

|     |     |     +-------CASE CAMBIO = 7
&&Si no, mantenemos el codigo de la fuente que se estaba trabajando
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE=" "
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.COMBO1.REFRESH

THISFORM.PAGEFRAME1.PAGE4.COMBO1.ENABLED=.F.
|     |     |     |
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT4.VALUE=0
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT4.REFRESH
|     |     |     |
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT6.VALUE=0
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT6.REFRESH
|     |     |     |
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT2.VALUE=0
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT2.REFRESH
|     |     |     |
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT7.VALUE=0
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT7.REFRESH
|     |     |     |
THISFORM.PAGEFRAME1.PAGE4.TEXT2.READONLY=.T.

THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY=.T.

THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY=.T.

THISFORM.PAGEFRAME1.PAGE4.TEXT7.READONLY=.T.
|     |     |     |           THISFORM.PAGEFRAME1.PAGE4.TEXT1.SETFOCUS
|     |     |     +-------ENDCASE
|     |     |
|     |     +-------ENDIF
|     +-------ENDIF
|           PTROFTE = ALLTRIM(THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE)
|           REQUERY('mto_mvto')
|           THISFORM.PAGEFRAME1.PAGE4.GRID1.REFRESH
|           THISFORM.PAGEFRAME1.PAGE4.GUARDAR.ENABLED=.F.
+-------ENDPROC
```

Codigo Fuente(Text1)

```
+-------PROCEDURE LOSTFOCUS
&&Validamos el codigo de fuente digitado
|     +-------IF LEN(ALLTRIM(THIS.VALUE)) < 20
&&Si no es valido, lo informamos
                                WAIT WIND " Recuerde que el codigo de la fuente debe tener 20 digitos
                                " NOWAIT
&&Inhabilitamos los objetos
```

```
|     |           THISFORM.PAGEFRAME1.PAGE4.COMBO1.ENABLED = .F.
|     |           THISFORM.PAGEFRAME1.PAGE4.TEXT2.READONLY = .T.
|     |           THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY = .T.
|     |           THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY = .T.
|     |           THISFORM.PAGEFRAME1.PAGE4.TEXT7.READONLY = .T.
|     |
|     |           THISFORM.PAGEFRAME1.PAGE4.TEXT1.SETFOCUS

|     |           THISFORM.PAGEFRAME1.PAGE4.GUARDAR.ENABLED = .F.
&&Mostramos los digitos de la fuente
|     |
                                THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE =
                                ALLTRIM(DEPARTAMENTO)+ALLTRIM(MUNICIPIO)
|     |           THISFORM.PAGEFRAME1.PAGE4.TEXT1.REFRESH
|     |
|     +-------ELSE
&&Si es valida la informacion
|     |     +-------IF LEN(ALLTRIM(THIS.VALUE)) = 20
|     |     |           THISFORM.PAGEFRAME1.PAGE4.COMBO1.ENABLED = .T.
                                          ARTICULOSPERMITIDOS =
                                          THISFORM.PAGEFRAME1.PAGE4.COMBO1
&&Primero buscamos para ver si el codigo de fuente existe
&&ya en el archivo maestro
&&Verificamos que el consecutivo digitado sea consistente
|     |     |     +-------IF SUBSTR(ALLTRIM(THIS.VALUE),17,4)<>"0000"
&&En caso afrirmativo, capturamos el nuemro de registro actual
|     |     |     |           SELECT V_IPCMFTE
|     |     |     |           MARCAREGISTRO = RECNO()
&&Capturamos parametros para enviarselos a la vista y reconsultar la informacion
&&de la vista
|     |     |     |           OFTEPAFTE = ALLTRIM(THIS.VALUE)
|     |     |     |           REQUERY('v_oftepa')
|     |     |     |           SELE V_OFTEPA
|     |     |     |     +-------IF RECCOUNT() > 0
&&Si la fuente articulo existe verificamos que no este marcado para eliminar
|     |     |     |     |     +-------IF V_OFTEPA.MARCA = .F.
&&Si es asi mostramos el nombre de la fuente y validamos si se esta trabajando en bogota
&&para el manejo de de los consecutivos de fuente que estan en el rango reservado
&&para servicios Publicos o de Informacion Nacional
        THISFORM.PAGEFRAME1.PAGE4.TEXT9.VALUE = V_OFTEPA.NOMBFTE
        +-------IF SUBSTR(ALLTRIM(THIS.VALUE),1,2) = "11" AND
        SUBSTR(ALLTRIM(THIS.VALUE),1,2) = ALLTRIM(DEPARTAMENTO)
                +-------IF BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4),'8850','8950')
                        MESSAGEBOX('Consecutivo de Fuente reservado para asignarle Articulos de
                        Formación Nacional !!!',0+64+0,'Precaución')
                        +-------IF BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4),'8887','8890')
                                = MESSAGEBOX('Consecutivo de Fuente reservado para asignarle
                                Servicios Publicos !!!',0+64+0,'Precaución')
```

```
&&Si esta en el rango de Servicios Publicos cargamos los articulos pertenecientes a este grupo
                                SELECT V_IPCMART.CODART FROM V_IPCMART;
                                WHERE
                                INLIST(V_IPCMART.CODART,'2210102','2220101','2220201','2220202','
                                2220203','7320101','7320102');
                                INTO CURSOR VALIDOS ORDER BY CODART

                                SELECT VALIDOS
                                ARTICULOSPERMITIDOS.CLEAR

                                THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE=" "
&&Cargamos la lista con el resultado de la seleccion anterior
                                +-------SCAN
                                        ARTICULOSPERMITIDOS.ADDITEM(VALIDOS.CODART)
                                +-------ENDSCAN
                                        USE
|               |       +-------ENDIF
&&Si pertenecen al rango de Formacion Nacional lo informamos y cargamos los articulos
&&que estan marcados de formacion Nacional en el Maestro de Articulos(campo fornal = .T.)
                        +-------IF BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4),'8850','8886') OR
                        BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4),'8891','8950')
                                = MESSAGEBOX('Consecutivo de Fuente reservado para utilizarlos
                                Bogota !!!',0+64+0,'Información')

                                SELECT CODART;
                                FROM V_IPCMART;
                                WHERE (V_IPCMART.FORNAL = .T.) INTO CURSOR ARTIFORNAL

                                SELECT ARTIFORNAL
                                ARTICULOSPERMITIDOS.CLEAR

                                THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE=" "
&&Cargamos la lista con el resultado de la seleccion anterior
                                +-------SCAN
                                        ARTICULOSPERMITIDOS.ADDITEM(ARTIFORNAL.CODART)
                                +-------ENDSCAN
                                USE
|               |       +-------ENDIF
|       |       +-------ELSE
&&      En caso de no estar en el rango cargamos todos los articulos
|       |           SELECT V_IPCMART.CODART FROM V_IPCMART;
                                INTO CURSOR VALIDOS ORDER BY CODART

                                SELECT VALIDOS
                                ARTICULOSPERMITIDOS.CLEAR

                                THISFORM.PAGEFRAME1.PAGE3.COMBO1.VALUE=" "
&&Cargamos la lista con el resultado de la seleccion anterior
```

```
|     |     |     +-------SCAN
                                ARTICULOSPERMITIDOS.ADDITEM(VALIDOS.CODART)
|     |     |     +-------ENDSCAN
|     |           USE
|             |     +-------ENDIF
|     |     +-------ELSE
&&En caso de que la ciudad que esta tratando de ingresar fuentes
&&validamos si el consecutivo correponde a los reservados para articulos
&&de Formacion Nacional si asi lo fuere informamos que no pueden hacerlo y no
&&se cargan articulos
|                     +-------IF BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4),'8850','8950')
                              = MESSAGEBOX('Consecutivo de Fuente reservado para asignarle
                              Articulos de Formación Nacional !!!',0+64+0,'Precaución')
&&En caso de estar en el rango de reservados para Servicios Publicos cargamos los articulos
&&pertencientes a este grupo
|                     |+-------IF BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4),'8887','8890')
                              = MESSAGEBOX('Consecutivo de Fuente reservado para asignarle
                              Servicios Publicos !!!',0+64+0,'Precaución')

                              SELECT V_IPCMART.CODART FROM V_IPCMART;
                              WHERE
                              INLIST(V_IPCMART.CODART,'2210102','2220101','2220201','2220202','
                              2220203','7320101','7320102');
                              INTO CURSOR VALIDOS ORDER BY CODART

                              SELECT VALIDOS
                              ARTICULOSPERMITIDOS.CLEAR

                              THISFORM.PAGEFRAME1.PAGE3.COMBO1.VALUE=" "
&&Cargamos la lista con el resultado de la seleccion anterior
|     |     |     +-------SCAN
                                ARTICULOSPERMITIDOS.ADDITEM(VALIDOS.CODART)
|     |     |     +-------ENDSCAN
                                USE
                     +-------ENDIF
                  +-------IF BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4),'8850','8886') OR
                  BETWEEN(SUBSTR(ALLTRIM(THIS.VALUE),17,4),'8891','8950')

                     = MESSAGEBOX('Consecutivo de Fuente reservado para utilizarlos Bogota
                     !!!',0+64+0,'Información')
|     +-------ENDIF
|   +-------ELSE
&&En caso que no pertenezca a lguno de estpos rangos cargamos toda la canasta
|           SELECT V_IPCMART.CODART FROM V_IPCMART;
                     INTO CURSOR VALIDOS ORDER BY CODART

                     SELECT VALIDOS
                     ARTICULOSPERMITIDOS.CLEAR
```

```
                        THISFORM.PAGEFRAME1.PAGE3.COMBO1.VALUE=" "
&&Cargamos la lista con el resultado de la seleccion anterior
|     |     +-------SCAN
                        ARTICULOSPERMITIDOS.ADDITEM(VALIDOS.CODART)
|     |     +-------ENDSCAN
|           USE
|     +-------ENDIF
        +-------ENDIF
|  +-------ELSE
&&Si el codigo de fuente ingresado no es encontrado preguntamos si desea iongresarlo
&&si acepta lo llevamos automaticamente a la pantalla deonde se puede hacer tal tarea
        INGRESAR=MESSAGEBOX('Fuente no encontrada, desea ingresarla
        ??',4+32+0,'Información')
|     +-------DO CASE
|     +-------CASE INGRESAR=6

                THISFORM.PAGEFRAME1.PAGES(1).SETFOCUS
&&Enviamos el codigo de la fuente digitado
                THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE = THIS.VALUE
|           PASOFTE = THIS.VALUE
                        THISFORM.PAGEFRAME1.PAGE1.ECODFTE.SETFOCUS

                        THISFORM.PAGEFRAME1.PAGE1.ECODFTE.REFRESH

                        THISFORM.PAGEFRAME1.PAGE1.ECODFTE.LOSTFOCUS
&&Nos devolvemos al registro capturado inicialmente en caso de que la busqueda no diera
&&resultados
|           SELECT V_IPCMFTE
|           GO MARCAREGISTRO
|           THIS.VALUE = V_IPCMFTE.CODFTE
|     +-------CASE INGRESAR=7
&&En caso de no aceptar ingresar la fuente nos devolvemos al registro capturado inicialmente
&&en caso de que la busqueda no diera resultados
|           SELECT V_IPCMFTE
|           GO MARCAREGISTRO
|           THIS.VALUE = V_IPCMFTE.CODFTE
|     +-------ENDCASE
|  +-------ENDIF
|           PTROFTE=ALLTRIM(THIS.VALUE)
|           REQUERY("mto_mvto")
|           SELE MTO_MVTO
|  +-------IF RECCOUNT() = 0
                MENSAJE=MESSAGEBOX(" La fuente no aparece en movimiento ",0+48+0,"Actualizar
                Movimiento ")
|     +-------ELSE
                THISFORM.PAGEFRAME1.PAGE4.GRID1.REFRESH
|     +-------ENDIF
```

```
 +-------ELSE
                        INGRESAR=MESSAGEBOX('Fuente no encontrada, desea ingresarla
                        ??',4+32+0,'Información')
        +-------DO CASE
|      +-------CASE INGRESAR=6
|            THISFORM.PAGEFRAME1.PAGES(1).SETFOCUS

                        THISFORM.PAGEFRAME1.PAGE1.ECODFTE.VALUE = THIS.VALUE
|            PASOFTE = THIS.VALUE
                        THISFORM.PAGEFRAME1.PAGE1.ECODFTE.SETFOCUS

                        THISFORM.PAGEFRAME1.PAGE1.ECODFTE.REFRESH

                        THISFORM.PAGEFRAME1.PAGE1.ECODFTE.LOSTFOCUS
                        SELECT V_IPCMFTE
|            GO MARCAREGISTRO
|            THIS.VALUE = V_IPCMFTE.CODFTE
|      +-------CASE INGRESAR=7
|      |            SELECT V_IPCMFTE
|      |            GO MARCAREGISTRO
|      |            THIS.VALUE = V_IPCMFTE.CODFTE
|      +-------ENDCASE
|      +-------ENDIF
|    +-------ELSE
|                =MESSAGEBOX(" El consecutivo no es valido !! ",0+48+0,"Actualizar Movimiento ")
                 THISFORM.PAGEFRAME1.PAGE4.COMBO1.VALUE= ' '
                 THISFORM.PAGEFRAME1.PAGE4.COMBO1.ENABLED=.F.
                 THISFORM.PAGEFRAME1.PAGE4.TEXT2.READONLY=.T.
                 THISFORM.PAGEFRAME1.PAGE4.TEXT4.READONLY=.T.
                 THISFORM.PAGEFRAME1.PAGE4.TEXT6.READONLY=.T.
                 THISFORM.PAGEFRAME1.PAGE4.TEXT7.READONLY=.T.
&&Mostramos los digitos de la fuente
                 THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE=ALLTRIM(DEPARTAMENTO)+AL
                 LTRIM(MUNICIPIO)
                 THISFORM.PAGEFRAME1.PAGE4.TEXT1.REFRESH
                 THISFORM.PAGEFRAME1.PAGE4.TEXT1.SETFOCUS

                 THISFORM.PAGEFRAME1.PAGE4.GUARDAR.ENABLED=.F.
|   +-------ENDIF
| +-------ELSE
                 = MESSAGEBOX('Recuerde que el consecutivo debe tener 20
                 digitos....',0+48+0,'Informacion'))
                 |
                 THISFORM.PAGEFRAME1.PAGE4.TEXT1.VALUE=ALLTRIM(DEPARTAMENTO)+AL
                 LTRIM(MUNICIPIO)
|      |   THISFORM.PAGEFRAME1.PAGE4.TEXT1.REFRESH
|      |   THISFORM.PAGEFRAME1.PAGE4.TEXT1.SETFOCUS
|      | +-------ENDIF
```

```
|       +-------ENDIF
|
|
+-------ENDPROC
```

5.16 MANTENIMIENTO DE MOVIMIENTO

5.16.1 Navegador

```
+-------PROCEDURE RECORDPOINTERMOVED
&&Definimos y cargamos variable con el nombre del archivo
|           LOCAL CALIAS
|           CALIAS=ALIAS()
|
&&Cargamos a variables los contenidos de las listas
&&de articulos asignados a la fuente y los no asignados
                LISTASELECCIONADA =
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELE
                CTED
                LISTAFUENTE =
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSOU
                RCE
&&Cargamos variables con los codigos de fuente y articulo
|           SELECT V_IPCMFTE1
|
                        THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE.VALUE =
                        V_IPCMFTE1.CODFTE
|           PASOFTE = THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE.VALUE

                        THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.CNOMBREFUENTE.CONT
                        ROLSOURCE = 'v_Ipcmfte1.nombfte'
|
                        FAPPAFTE =
                        ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VAL
                        UE)
|           REQUERY('v_fappa')
|           SELECT V_FAPPA
&&Consultamos los articulos que se recolectan en la fuente destino
|           SELECT V_IPCMART.CODART, V_IPCMART.NOMART;
|                FROM V_IPCMART, V_FAPPA;
                        WHERE ((V_FAPPA.CODART = V_IPCMART.CODART) AND
                        (V_FAPPA.CODFTE = FAPPAFTE));
|                AND V_FAPPA.MARCA = .F.;
|                INTO CURSOR CANASTAFUENTE
&&Verificamos si el consecutivo de fuente ingresado pertenece al rango reservado
&&para Servicios Publicos
```

```
|       +-------IF BETWEEN(SUBSTR(FAPPAFTE,17,4),'8887','8890')
&&Tomamos el codigo fuente destino para consultar los articulos que tiene
&&para solo cargar en esta   lista los articulos que no tenga de la fuente origen
                                FAPPAFTE =
                                ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE.VALU
                                E)
|       |               REQUERY('v_fappa')
|       |               SELECT V_FAPPA
&&Seleccionamos los articulos que no estan asignados a la fuente actual para alimentar la otra lista
SELECT V_IPCMART.CODART, V_IPCMART.NOMART;
FROM V_IPCMART, V_FAPPA;
WHERE
(INLIST(V_IPCMART.CODART,'2210102','2220101','2220201','2220202','2220203','7320101','732010
2') AND ((V_FAPPA.CODART = V_IPCMART.CODART) AND (V_FAPPA.CODFTE =
ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE.VALUE))) AND;
V_FAPPA.MARCA = .F. AND V_IPCMART.CODART NOT IN (SELECT DISTINCT CODART FROM
CANASTAFUENTE));
|       |                   INTO CURSOR CANASTANACIONAL;
|       |                   ORDER BY V_IPCMART.CODART
|       |
|       |               SELECT CANASTANACIONAL
|       |
|       |               LISTAFUENTE.CLEAR
|       |       +-------SCAN
LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
'+PROPER(CANASTANACIONAL.NOMART))
|       |       +-------ENDSCAN
|       +-------ELSE
&&En caso de no ser de Servicios Publicos verificamos  si pertenece a los reservados
&&para Formación Nacional
                                +-------IF BETWEEN(SUBSTR(FAPPAFTE,17,4),'8850','8886') OR
                                BETWEEN(SUBSTR(FAPPAFTE,17,4),'8891','8950')
                                        FAPPAFTE =
                                        ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.EC
                                        ODFTE.VALUE)
&&Reconsulatmos la vista para obtener los articulos que cotizan en la fuente
|       |       |               REQUERY('v_fappa')
|       |       |               SELECT V_FAPPA
|       |       |
&&Seleccionamos los articulos que no estan asignados a la fuente origen para alimentar la otra lista
SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM V_IPCMART, V_FAPPA;
WHERE (V_IPCMART.FORNAL = .T. AND;
 ((V_FAPPA.CODART = V_IPCMART.CODART) AND (V_FAPPA.CODFTE =
ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE.VALUE))) AND;
V_FAPPA.MARCA = .F. AND V_IPCMART.CODART NOT IN (SELECT DISTINCT CODART FROM
CANASTAFUENTE));
INTO CURSOR CANASTANACIONAL;
ORDER BY V_IPCMART.CODART
```

```
|     |     |
|     |     |          SELECT CANASTANACIONAL
|     |     |
|     |     |          LISTAFUENTE.CLEAR
|     |     |     +-------SCAN
LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
'+PROPER(CANASTANACIONAL.NOMART))
|     |     |     +-------ENDSCAN
|     |     +-------ELSE
&&Consultamos los articulos que no se recolectan en la fuente actual
                                    FAPPAFTE =
                                    ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.EC
                                    ODFTE.VALUE)
|     |     |          REQUERY('v_fappa')
|     |     |          SELECT V_FAPPA
                                    SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM
                                    V_IPCMART, V_FAPPA;
|     |     |        WHERE ((V_FAPPA.CODART = V_IPCMART.CODART) AND;
                                     (V_FAPPA.CODFTE =
                                    ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.EC
                                    ODFTE.VALUE))) AND;
|     |     |        V_FAPPA.MARCA = .F. AND;
                                     (V_IPCMART.CODART NOT IN (SELECT DISTINCT CODART
                                    FROM CANASTAFUENTE));
|     |     |        INTO CURSOR CANASTANACIONAL
|     |     |
|     |     |        SELECT CANASTANACIONAL
|     |     |
|     |     |        LISTAFUENTE.CLEAR
|     |     |     +-------SCAN
                                    LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
                                    '+PROPER(CANASTANACIONAL.NOMART))
|     |     |     +-------ENDSCAN
|     |     +-------ENDIF
|     +-------ENDIF
|           USE
|           SELECT CANASTAFUENTE
|           USE
|
                        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.C
                        MDADD.ENABLED = .T.

                        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LS
                        TSOURCE.ENABLED = .T.
|
&&Refrescamos informacion en pantalla
                        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EP
                        ERIFTE.FORECOLOR=RGB(255,255,255)
```

```
                    THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.ED
                    ECA.FORECOLOR=RGB(255,255,255)

                    THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TE
                    XT2.FORECOLOR=RGB(255,255,255)

                    THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TE
                    XT3.FORECOLOR=RGB(255,255,255)
|
|
&&Actualizams informacion en pantalla
|       +-------WITH THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1
|       |           .NOSELECCIONADOS=LISTASELECCIONADA.LISTCOUNT
|       |           .NODISPONIBLES=LISTAFUENTE.LISTCOUNT
|       |           .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0)
|       |           .LDISPONIBLES.CAPTION=STR(.NODISPONIBLES,5,0)
|       +-------ENDWITH
|
|           SELECT(CALIAS)
+-------ENDPROC
```

5.16.2 Seleccione el Nombre de la Fuente(CnombreFuente)

```
+-------PROCEDURE INTERACTIVECHANGE
&&Cargamos a variables los contenidos de las listas
&&de articulos asignados a la fuente y los no asignados
                LISTASELECCIONADA =
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELE
                CTED
                LISTAFUENTE =
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSOU
                RCE
&&Cargamos variables con los codigos de fuente y articulo
|           SELECT V_IPCMFTE1
|
                    THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE.VALUE =
                    V_IPCMFTE1.CODFTE

                    THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.NAVIGATOR1.ENABLEDIS
                    ABLEBUTTONS
|
                    FAPPAFTE =
                    ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VAL
                    UE)
|           REQUERY('v_fappa')
|           SELECT V_FAPPA
&&Consultamos los articulos que se recolectan en la fuente actual
```

```
|           SELECT V_IPCMART.CODART, V_IPCMART.NOMART;
|                 FROM V_IPCMART, V_FAPPA;
                                WHERE ((V_FAPPA.CODART = V_IPCMART.CODART) AND
                                (V_FAPPA.CODFTE = FAPPAFTE));
|                 AND V_FAPPA.MARCA = .F.;
|                 INTO CURSOR CANASTAFUENTE
|
|     +-------IF BETWEEN(SUBSTR(FAPPAFTE,17,4),'8887','8890')
                                FAPPAFTE =
                                ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE
                                .VALUE)
|     |           REQUERY('v_fappa')
|     |           SELECT V_FAPPA
&&Seleccionamos los articulos que no estan asignados a la fuente actual
&&para alimentar la otra lista
SELECT V_IPCMART.CODART, V_IPCMART.NOMART;
FROM V_IPCMART, V_FAPPA;
WHERE
(INLIST(V_IPCMART.CODART,'2210102','2220101','2220201','2220202','2220203','7320101','732010
2') AND  ((V_FAPPA.CODART = V_IPCMART.CODART) AND;
 (V_FAPPA.CODFTE =
ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE.VALUE))) AND;
V_FAPPA.MARCA = .F. AND V_IPCMART.CODART NOT IN (SELECT DISTINCT CODART FROM
CANASTAFUENTE));
|     |                 INTO CURSOR CANASTANACIONAL;
|     |                 ORDER BY V_IPCMART.CODART
|     |
|     |           SELECT CANASTANACIONAL
|     |
|     |           LISTAFUENTE.CLEAR
|     |     +-------SCAN
LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
'+PROPER(CANASTANACIONAL.NOMART))
|     |     +-------ENDSCAN
|     +-------ELSE
                        +-------IF BETWEEN(SUBSTR(FAPPAFTE,17,4),'8850','8886') OR
                        BETWEEN(SUBSTR(FAPPAFTE,17,4),'8891','8950')
                                FAPPAFTE =
                                ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE
                                .VALUE)
|     |     |     REQUERY('v_fappa')
|     |     |     SELECT V_FAPPA
&&Seleccionamos los articulos que no estan asignados a la fuente actual para alimentar la otra lista
SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM V_IPCMART, V_FAPPA;
WHERE (V_IPCMART.FORNAL = .T. AND  ((V_FAPPA.CODART = V_IPCMART.CODART) AND
(V_FAPPA.CODFTE = ALLTRIM(THISFORM. PAGEFRAME1. PAGE5.CONTAINER2.
ECODFTE.VALUE))) AND;
```

```
V_FAPPA.MARCA = .F. AND V_IPCMART.CODART NOT IN (SELECT DISTINCT CODART FROM
CANASTAFUENTE));
INTO CURSOR CANASTANACIONAL ORDER BY V_IPCMART.CODART
|     |     |
|     |     |     SELECT CANASTANACIONAL
|     |     |
|     |     |     LISTAFUENTE.CLEAR
|     |     |     +-------SCAN
                                LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
                                '+PROPER(CANASTANACIONAL.NOMART))
|     |     |     +-------ENDSCAN
|     |     +-------ELSE
&&Consultamos los articulos que no se recolectan en la fuente actual
                                FAPPAFTE =
                                ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.EC
                                ODFTE.VALUE)
|     |     |           REQUERY('v_fappa')
|     |     |           SELECT V_FAPPA
SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM V_IPCMART, V_FAPPA;
WHERE ((V_FAPPA.CODART = V_IPCMART.CODART) AND;
 (V_FAPPA.CODFTE = ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.
CONTAINER2.ECODFTE.VALUE))) AND V_FAPPA.MARCA = .F. AND;
 (V_IPCMART.CODART NOT IN (SELECT DISTINCT CODART FROM CANASTAFUENTE));
INTO CURSOR CANASTANACIONAL
|     |     |
|     |     |         SELECT CANASTANACIONAL
|     |     |
|     |     |         LISTAFUENTE.CLEAR
|     |     |     +-------SCAN
                                LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
                                '+PROPER(CANASTANACIONAL.NOMART))
|     |     |     +-------ENDSCAN
|     |     +-------ENDIF
|     +-------ENDIF
|           USE
|           SELECT CANASTAFUENTE
|           USE

                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.CMDADD
                .ENABLED = .T.

                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSOU
                RCE.ENABLED = .T.
&&Refrescamos informacion en pantalla
                |
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFT
                E.FORECOLOR=RGB(255,255,255)
```

```
                |
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EDECA.F
                ORECOLOR=RGB(255,255,255)
                |
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT2.F
                ORECOLOR=RGB(255,255,255)
                |
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT3.F
                ORECOLOR=RGB(255,255,255)
|
|       +-------WITH THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1
|       |             .NOSELECCIONADOS=LISTASELECCIONADA.LISTCOUNT
|       |             .NODISPONIBLES=LISTAFUENTE.LISTCOUNT
|       |             .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0)
|       |             .LDISPONIBLES.CAPTION=STR(.NODISPONIBLES,5,0)
|       +-------ENDWITH
|
+-------ENDPROC
```

5.16.3 Escriba el Código de la Fuente(EcodFte)

```
+-------PROCEDURE LOSTFOCUS
|       +-------IF
          LEN(ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VALUE)) = 20
AND;
          SUBSTR(ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VALUE),1
          7,4) <> "0000"
&&Cargamos a variables los contenidos de las listas
&&de articulos asignados a la fuente origen y destino
LISTASELECCIONADA =
THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELECTED
LISTAFUENTE =
THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSOURCE
&&Cargamos variables con los codigos de fuente y articulo
|       |             SELECT V_IPCMFTE1
|       |             MARCAREGISTRO = RECNO()
|       |             OFTEPAFTE = ALLTRIM(THIS.VALUE)
|       |             REQUERY('v_oftepa')
|       |             SELE V_OFTEPA
|       |       +-------IF RECCOUNT() > 0
|       |       |               =SEEK(OFTEPAFTE,'v_ipcmfte1','pkcodfte')
|       |       |
THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE.VALUE = V_IPCMFTE1.CODFTE

THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.CNOMBREFUENTE.CONTROLSOURCE =
'v_ipcmfte1.nombfte'
|       |       |
```

```
                                        FAPPAFTE =
                                        ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.EC
                                        ODFTE1.VALUE)
|     |     |          REQUERY('v_fappa')
|     |     |          SELECT V_FAPPA
&&Consultamos los articulos que se recolectan en la fuente destino
SELECT V_IPCMART.CODART, V_IPCMART.NOMART;
FROM V_IPCMART, V_FAPPA;
WHERE ((V_FAPPA.CODART = V_IPCMART.CODART) AND (V_FAPPA.CODFTE = FAPPAFTE))
AND V_FAPPA.MARCA = .F.;
INTO CURSOR CANASTAFUENTE
|     |     |
|     |     |     +-------IF BETWEEN(SUBSTR(FAPPAFTE,17,4),'8887','8890')
|     |     |     |           FAPPAFTE = ALLTRIM(THIS.VALUE)
|     |     |     |           REQUERY('v_fappa')
|     |     |     |           SELECT V_FAPPA
&&Seleccionamos los articulos que no estan asignados a la fuente actual para alimentar la otra lista
SELECT V_IPCMART.CODART, V_IPCMART.NOMART;
FROM V_IPCMART, V_FAPPA;
WHERE (INLIST(V_IPCMART.CODART,'2210102','2220101','2220201','2220202','2220203',
'7320101','7320102') AND  ((V_FAPPA.CODART = V_IPCMART.CODART) AND
(V_FAPPA.CODFTE = ALLTRIM(THIS.VALUE))) AND V_FAPPA.MARCA = .F. AND;
V_IPCMART.CODART NOT IN (SELECT DISTINCT CODART FROM CANASTAFUENTE));
 INTO CURSOR CANASTANACIONAL;
ORDER BY V_IPCMART.CODART
|     |     |     |
|     |     |     |           SELECT CANASTANACIONAL
|     |     |     |
|     |     |     |           LISTAFUENTE.CLEAR
|     |     |     |     +-------SCAN
LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
'+PROPER(CANASTANACIONAL.NOMART))
|     |     |     |     +-------ENDSCAN
|     |     |     +-------ELSE
                                          +-------IF
                                          BETWEEN(SUBSTR(FAPPAFTE,17,4),'8850','8886') OR
                                          BETWEEN(SUBSTR(FAPPAFTE,17,4),'8891','8950')
|     |     |     |     |           FAPPAFTE = ALLTRIM(THIS.VALUE)
|     |     |     |     |           REQUERY('v_fappa')
|     |     |     |     |           SELECT V_FAPPA
&&Seleccionamos los articulos que no estan asignados a la fuente origen para alimentar la otra lista
SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM V_IPCMART, V_FAPPA;
|     |     |     |     | WHERE (V_IPCMART.FORNAL = .T. AND;
 ((V_FAPPA.CODART = V_IPCMART.CODART) AND (V_FAPPA.CODFTE =
ALLTRIM(THIS.VALUE))) AND;
|     |     |     |     | V_FAPPA.MARCA = .F. AND;
V_IPCMART.CODART NOT IN (SELECT DISTINCT CODART FROM CANASTAFUENTE));
|     |     |     |     | INTO CURSOR CANASTANACIONAL;
```

```
|     |     |     |     | ORDER BY V_IPCMART.CODART
|     |     |     |     |
|     |     |     |     | SELECT CANASTANACIONAL
|     |     |     |     |
|     |     |     |     |           LISTAFUENTE.CLEAR
|     |     |     |     |     +-------SCAN
LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
'+PROPER(CANASTANACIONAL.NOMART))
|     |     |     |     |     +-------ENDSCAN
|     |     |     |     +-------ELSE
&&Consultamos los articulos que no se recolectan en la fuente actual
|     |     |     |     |           FAPPAFTE = ALLTRIM(THIS.VALUE)
|     |     |     |     |           REQUERY('v_fappa')
|     |     |     |     |           SELECT V_FAPPA
|     |     |     |     |
SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM V_IPCMART, V_FAPPA;
WHERE ((V_FAPPA.CODART = V_IPCMART.CODART) AND;
 (V_FAPPA.CODFTE = FAPPAFTE)) AND;
|     |     |     |     |           V_FAPPA.MARCA = .F. AND;
 (V_IPCMART.CODART NOT IN (SELECT DISTINCT CODART FROM CANASTAFUENTE));
|     |     |     |     |           INTO CURSOR CANASTANACIONAL
|     |     |     |     |
|     |     |     |     |           SELECT CANASTANACIONAL
|     |     |     |     |
|     |     |     |     |           LISTAFUENTE.CLEAR
|     |     |     |     |     +-------SCAN
LISTAFUENTE.ADDITEM(CANASTANACIONAL.CODART+'
'+PROPER(CANASTANACIONAL.NOMART))
|     |     |     |     |     +-------ENDSCAN
|     |     |     |     |
|     |     |     |     |           USE
|     |     |     |     |           SELECT CANASTAFUENTE
|     |     |     |     |           USE
|     |     |     |     +-------ENDIF
|     |     |     +-------ENDIF
|     |     |
        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.CMDADD.ENAB
        LED = .T.
        |     |     |
        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSOURCE.EN
        ABLED = .T.
|     |     |
|     |     |
|     |     +-------ELSE
                                    MESSAGEBOX('Fuente no encontrada',0+64+0,'Informacion')
|     |     |           SELECT V_IPCMFTE1
|     |     |           GO MARCAREGISTRO
|     |     |           THIS.VALUE = V_IPCMFTE1.CODFTE
```

```
|     |     |
        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.CMDADD.ENAB
        LED = .F.
        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSOURCE.EN
        ABLED = .F.
|     |     +-------ENDIF
|     |
|     |           SELE V_IPCMFTE1
        THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.NAVIGATOR1.ENABLEDISABLEBUTTON
        S
|     |
&&Refrescamos informacion en pantalla
        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFTE.FOR
        ECOLOR=RGB(255,255,255)

        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EDECA.FOREC
        OLOR=RGB(255,255,255)

        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT2.FORECO
        LOR=RGB(255,255,255)

        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT3.FORECO
        LOR=RGB(255,255,255)
|     |
                        +-------WITH
                        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1
|     |     |     .NOSELECCIONADOS=LISTASELECCIONADA.LISTCOUNT
|     |     |.NODISPONIBLES=LISTAFUENTE.LISTCOUNT
                        .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0)
|     |     |.LDISPONIBLES.CAPTION=STR(.NODISPONIBLES,5,0)
|     |     +-------ENDWITH
|     |
|     |       THISFORM.REFRESH
|     +-------ELSE
|     |       = MESSAGEBOX('Informacion no Valida...',0+64+0,'Verificar')
|     +-------ENDIF
+-------ENDPROC

+-------PROCEDURE INIT
&&Capturamos el valor del campo del archivo
|           THIS.VALUE=V_IPCMFTE1.CODFTE
+-------ENDPROC
```

5.16.4 Seleccione el Nombre de la Fuente(cnombrefuente1)

```
+-------PROCEDURE INTERACTIVECHANGE
&&Cargamos a variables los contenidos de las listas
&&de articulos asignados a la fuente y los no asignados
```

```
                LISTASELECCIONADA =
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELE
                CTED
&&Cargamos variables con los codigos de fuente y articulo
|
|           SELECT V_IPCMFTE
                        THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VALUE =
                        V_IPCMFTE.CODFTE
                        THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.NAVIGATOR2.ENABLEDIS
                        ABLEBUTTONS
|
                        FAPPAFTE =
                        ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VAL
                        UE)
|           REQUERY('v_fappa')
|           SELECT V_FAPPA

&&Consultamos los articulos que se recolectan en la fuente actual
|           SELECT V_IPCMART.CODART, V_IPCMART.NOMART;
|                FROM V_IPCMART, V_FAPPA;
                        WHERE ((V_FAPPA.CODART = V_IPCMART.CODART) AND
                        (V_FAPPA.CODFTE = FAPPAFTE));
|             AND V_FAPPA.MARCA = .F.;
|             INTO CURSOR CANASTAFUENTE
|
|           SELECT CANASTAFUENTE
|           LISTASELECCIONADA.CLEAR
|      +-------SCAN
                        LISTASELECCIONADA.ADDITEM(CANASTAFUENTE.CODART+'
                        '+PROPER(CANASTAFUENTE.NOMART))
|      +-------ENDSCAN
|
&&Refrescamos informacion en pantalla

        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFTE.FOR
        ECOLOR=RGB(255,255,255)
        |
        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EDECA.FOREC
        OLOR=RGB(255,255,255)
        |
        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT2.FORECO
        LOR=RGB(255,255,255)
        |
        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT3.FORECO
        LOR=RGB(255,255,255)
|
|      +-------WITH THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1
|      |            .NOSELECCIONADOS=LISTASELECCIONADA.LISTCOUNT
```

```
|       |             *.NoDisponibles=ListaFuente.listCount
|       |             .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0)
|       |             * .LDisponibles.Caption=Str(.NoDisponibles,5,0)
|       +-------ENDWITH
|
+-------ENDPROC
```

5.16.5 Escriba el Código de la Fuente(ecodfte1)

```
+-------PROCEDURE INIT
&&Capturamos el valor del campo del archivo
|             THIS.VALUE=V_IPCMFTE.CODFTE
+-------ENDPROC

+-------PROCEDURE LOSTFOCUS
&&Almacenamos el contenido de las listas a las variables
          |
          LISTASELECCIONADA=THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVER
          LISTA1.LSTSELECTED
&&Seleccionamos el archivo de fuentes, almacenamos el numero del registro
&&actual en una variable, buscamos el codigo en el archivo
|             SELECT V_IPCMFTE
|             MARCAREGISTRO = RECNO()
|             OFTEPAFTE = ALLTRIM(THIS.VALUE)
|             REQUERY('v_oftepa')
|             SELE V_OFTEPA
|       +-------IF RECCOUNT() > 0
                        THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VALUE =
                        V_OFTEPA.CODFTE

                        THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.CNOMBREFUENTE1.VALU
                        E = V_OFTEPA.NOMBFTE

                        THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.CNOMBREFUENTE1.REFR
                  ESH
&&Cargamos las variable con los contenidos de las listas tanto de
&&articulos asignados a la fuente actual como los disponibles, es decir,
&&el resto de la canasta
                        LISTASELECCIONADA =
                        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LS
                        TSELECTED
|       |
&&Consultamos los articulos que corresponden a la fuente actual
&&y los insertamos en la lista de articulos seleccionados
|       |
                        FAPPAFTE =
                        ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VAL
                        UE)
```

```
|     |       REQUERY('v_fappa')
                                SELECT V_IPCMART.CODART, V_IPCMART.NOMART FROM
                                V_IPCMART, V_FAPPA WHERE ((V_FAPPA.CODART =
                                V_IPCMART.CODART)AND(V_FAPPA.CODFTE = FAPPAFTE)) AND
                                V_FAPPA.MARCA = .F. INTO CURSOR CANASTAFUENTE
|     |
|     |           SELECT CANASTAFUENTE
|     |           LISTASELECCIONADA.CLEAR
&&Insertamos uno por uno los articulos
|     |     +-------SCAN
                                        LISTASELECCIONADA.ADDITEM(CANASTAFUENTE.CODART
                                        +' '+PROPER(CANASTAFUENTE.NOMART))
|     |     +-------ENDSCAN
|     |
&&Ordenamos la lista
|     |           LISTASELECCIONADA.LISTITEMID=1
&&En caso de no encontrarlo lo informamos y regresamos al registro inicial
|     +-------ELSE
|     |           MESSAGEBOX('Fuente no encontrada',0+64+0,'Informacion')
|     |           SELECT V_IPCMFTE
|     |           GO MARCAREGISTRO
|     |           THIS.VALUE = V_IPCMFTE.CODFTE
|     +-------ENDIF
|
|           SELECT V_IPCMFTE
                THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.NAVIGATOR2.ENABLEDISABLEB
                UTTONS
|
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFT
                E.FORECOLOR=RGB(255,255,255)
                |
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EDECA.F
                ORECOLOR=RGB(255,255,255)
                |
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT2.F
                ORECOLOR=RGB(255,255,255)
                |
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT3.F
                ORECOLOR=RGB(255,255,255)
|
&&Refrescar pantalla el numero de articulos asignado y el
&&numero de no asignados
|     +-------WITH THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1
|     |           .NOSELECCIONADOS=LISTASELECCIONADA.LISTCOUNT
|     |           .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0)
|     +-------ENDWITH
|
|           THISFORM.REFRESH
```

```
+-------ENDPROC
```

5.16.6 Fuente destino

```
+-------PROCEDURE RECORDPOINTERMOVED
&&Definimos y cargamos variable con el nombre del archivo
|           LOCAL CALIAS
|           CALIAS=ALIAS()
|
|           SELE V_IPCMFTE
&&Asignamos el contenido de las listas a variables
                LISTASELECCIONADA =
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELE
                CTED
|
                THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VALUE =
                V_IPCMFTE.CODFTE
|
                PASOFTE = THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VALUE
                |
                THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.CNOMBREFUENTE1.CONTROLS
                OURCE = 'v_ipcmfte.nombfte'
|
&&Consultamos los articulos que corresponden a la fuente actual
&&y los insertamos en la lista de articulos seleccionados
|
FAPPAFTE = ALLTRIM(THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VALUE)
|           REQUERY('v_fappa')
|           SELECT V_IPCMART.CODART, V_IPCMART.NOMART;
|             FROM V_IPCMART, V_FAPPA;
WHERE ((V_FAPPA.CODART = V_IPCMART.CODART)AND(V_FAPPA.CODFTE = FAPPAFTE))
AND V_FAPPA.MARCA = .F.;
|             INTO CURSOR CANASTAFUENTE
|
|           SELECT CANASTAFUENTE
|           LISTASELECCIONADA.CLEAR
&&Insertamos uno por uno los articulos
|     +-------SCAN
                        LISTASELECCIONADA.ADDITEM(CANASTAFUENTE.CODART+'
                        '+PROPER(CANASTAFUENTE.NOMART))
|     +-------ENDSCAN
|
&&Ordenamos la lista
|           LISTASELECCIONADA.LISTITEMID=1
                        |
                        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EP
                        ERIFTE.FORECOLOR=RGB(255,255,255)
```

```
                        |
                        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.ED
                        ECA.FORECOLOR=RGB(255,255,255)
                        |
                        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TE
                        XT2.FORECOLOR=RGB(255,255,255)
                        |
                        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TE
                        XT3.FORECOLOR=RGB(255,255,255)
|
&&Actualizams informacion en pantalla
|       +-------WITH THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1
|       |             .NOSELECCIONADOS=LISTASELECCIONADA.LISTCOUNT
|       |             .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0)
|       +-------ENDWITH
|
|               SELECT(CALIAS)
+-------ENDPROC

+-------PROCEDURE LSTSELECTED.CLICK
&&Capturamos los codigo tanto de fuente como de articulo actual
|               VFAPCODFTE = THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VALUE
|               VFAPCODART=SUBSTR(THIS.VALUE,1,7)
|
                        |
                        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EP
                        ERIFTE.FORECOLOR=RGB(0,0,0)
                        |
                        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.ED
                        ECA.FORECOLOR=RGB(0,0,0)
                        |
                        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TE
                        XT2.FORECOLOR=RGB(0,0,0)
                        |
                        THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TE
                        XT3.FORECOLOR=RGB(0,0,0)
|
&&Enviamos los parametros necesarios a la vista para que nos
&&actualice la informacion
|               SELECT 'vfap'
|               REQUERY('vfap')
                    |
                    THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFT
                    E.REFRESH
                    |
                    THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EDECA.R
                    EFRESH
```

```
                |
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT2.R
                EFRESH
                |
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT3.R
                EFRESH
+-------ENDPROC

+-------PROCEDURE CMDADD.CLICK
&&Almacenamos el contenido de la lista de articulo no asignados
                LISTAFUENTE =
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSOU
                RCE
|
|       +-------IF CAMBIO
                        VFAPCODFTE =
                        THISFORM.PAGEFRAME1.PAGE5.CONTAINER2.ECODFTE1.VALUE
|       |               FAPPAFTE = ALLTRIM(VFAPCODFTE)
|       |               RESPAFTE = ALLTRIM(VFAPCODFTE)
&&Insertar los nuevos articulos que se han asociado a la fuente
&& A partir del numero de articulos seleccionados
&&insertamos uno a uno los articulos
|       |               CUENTA = LISTAFUENTE.LISTCOUNT
|       |       +-------FOR I = 1 TO CUENTA
|       |       |       +-------IF LISTAFUENTE.SELECTED(I)
ELARTICULO = SUBSTR(LISTAFUENTE.LIST(I),1,7)
|       |       |       |
&&Agrega articulos a la Fuente articulo
&&Buscamos en el archivo fuente - articulo la coincidencia
&&de la fuente - articulo actual
|       |       |       |               FAPPART = ELARTICULO
|       |       |       |               REQUERY('v_fappart')
|       |       |       |               SELECT V_FAPPART
|       |       |       |       +-------IF RECCOUNT() = 0
&&Si no lo encontramos, consultamos las periodicidades  permitidas
&&para el articulo actual
SELECT CODPER, CODPART FROM V_IPCRPART, V_IPCMART WHERE
(V_IPCRPART.CODPER = V_IPCMART.PERART AND V_IPCMART.CODART = ELARTICULO)
INTO CURSOR PERIODICIDADES
|       |       |       |       |               SELE PERIODICIDADES
|       |       |       |       |               GO TOP
&&Insertamos la informacion en el archivo de fuente articulo
|       |       |       |       |               PERIODO = PERIODICIDADES.CODPART
|       |       |       |       |               SELECT V_FAPPART
|       |       |       |       |               APPEND BLANK
REPLACE V_FAPPART.CODFTE WITH VFAPCODFTE, V_FAPPART.CODART WITH
ELARTICULO,;
V_FAPPART.PERFAP WITH PERIODO, V_FAPPART.MARCA WITH .F.
```

```
|     |     |     |     |           TABLEUPDATE(.T.)
|     |     |     |     +-------ELSE
|     |     |     |     |           SELECT V_FAPPART
REPLACE ALL V_FAPPART.MARCA WITH .F. FOR ((V_FAPPART.CODFTE = VFAPCODFTE) AND
(V_FAPPART.CODART = ELARTICULO))
|     |     |     |     |           TABLEUPDATE(.T.)
|     |     |     |     +-------ENDIF
|     |     |     |           SELE V_IPCRFAP
|     |     |     |           TABLEUPDATE(.T.)
|     |     |     |
&&Realizamos la busqueda en el archivo de especificaciones
&&Si no se encuentra Agrega la informacion en el archivo
|     |     |     |           RESPART = ELARTICULO
|     |     |     |           REQUERY('v_resppa')
|     |     |     |           SELECT V_RESPPA
|     |     |     |     +-------IF RECCOUNT() = 0
|     |     |     |     |           SELECT DISTINCT CODESP, NOMESP;
|     |     |     |     |                 FROM V_IPCRESP;
                                            WHERE (CODART = ELARTICULO);
|     |     |     |     |                 GROUP BY CODESP;
|     |     |     |     |                 INTO CURSOR ESPBASE
|     |     |     |     |
|     |     |     |     |           SELECT ESPBASE
|     |     |     |     |     +-------SCAN
INSERT INTO V_RESPPA(CODFTE, CODART, CODESP, NOMESP, MARCA);

VALUES(VFAPCODFTE, ELARTICULO, ESPBASE.CODESP, ESPBASE.NOMESP, .F.)
|     |     |     |     |     +-------ENDSCAN
|     |     |     |     |           USE
|     |     |     |     +-------ELSE
REPLACE V_RESPPA.MARCA WITH .F. FOR ((V_RESPPA.CODFTE = VFAPCODFTE) AND
(V_RESPPA.CODART = ELARTICULO))
|     |     |     |     +-------ENDIF
|     |     |     |           SELE V_RESPPA
|     |     |     |           TABLEUPDATE(.T.)
|     |     |     |           SELE V_IPCRESP
|     |     |     |           TABLEUPDATE(.T.)
|     |     |     +-------ENDIF
|     |     +-------ENDFOR
|     |           CAMBIO=.F.
|     +-------ENDIF
|
|           DODEFAULT()
&&Refrescamos pantalla
                |
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFT
                E.FORECOLOR=RGB(255,255,255)
```

```
                |
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.EDECA.F
                ORECOLOR=RGB(255,255,255)
                |
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT2.F
                ORECOLOR=RGB(255,255,255)
                |
                THISFORM.PAGEFRAME1.PAGE5.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT3.F
                ORECOLOR=RGB(255,255,255)
|
|       +-------WITH THIS.PARENT
|       |             .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0)
|       |             .LDISPONIBLES.CAPTION=STR(.NODISPONIBLES,5,0)
|       +-------ENDWITH
|
|
+-------ENDPROC
+-------PROCEDURE CMDADD.GOTFOCUS
|
|
+-------ENDPROC

+-------PROCEDURE CMDREMOVE.CLICK
&&Almacenamos el contenido de la lista de articulos seleccionados en una variable
                LISTASELECCIONADA =
                THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.LSTSELE
                CTED
|       +-------IF CAMBIO
&&Cargamos variables
VFAPCODFTE = THISFORM.PAGEFRAME1.PAGE2.CONTAINER2.ECODFTE.VALUE
&&Segun el total de articulos seleccionados
|       |             CUENTA = LISTASELECCIONADA.LISTCOUNT
|       |       +-------FOR I = 1 TO CUENTA
|       |       |       +-------IF LISTASELECCIONADA.SELECTED(I)
&&Capturamos el articulo actual
ELARTICULO = SUBSTR(LISTASELECCIONADA.LIST(I),1,7)
|       |       |       |
&&Eliminar articulos que se han desasociado a la fuente
&&Elimina articulos de la Fuente articulo
|       |       |       |
|       |       |       |               SELECT V_IPCRFAP
REPLACE ALL WITH .T. FOR ((V_IPCRFAP.CODFTE = VFAPCODFTE) AND
(V_IPCRFAP.CODART = ELARTICULO))
|       |       |       |               TABLEUPDATE(.T.)
&&Eliminar especificaciones a los articulos
|       |       |       |               SELECT V_IPCRESP
REPLACE ALL WITH .T. FOR ((V_IPCRESP.CODFTE = VFAPCODFTE) AND
(V_IPCRESP.CODART = ELARTICULO))
```

```
|       |       |       |               TABLEUPDATE(.T.)
|       |       |       |
|       |       |       +-------ENDIF
|       |       +-------ENDFOR
|       |               CAMBIO=.F.
|       +-------ENDIF
|
|               DODEFAULT()
|
                        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EPERIFT
                        E.FORECOLOR=RGB(255,255,255)
                        |
                        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.EDECA.F
                        ORECOLOR=RGB(255,255,255)
                        |
                        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT2.F
                        ORECOLOR=RGB(255,255,255)
                        |
                        THISFORM.PAGEFRAME1.PAGE2.PAGEFRAME1.PAGE1.MOVERLISTA1.TEXT3.F
                        ORECOLOR=RGB(255,255,255)
&&Refrescamos pantalla
|       +-------WITH THIS.PARENT
|       |               .LSELECCIONADOS.CAPTION=STR(.NOSELECCIONADOS,5,0)
|       |               .LDISPONIBLES.CAPTION=STR(.NODISPONIBLES,5,0)
|       +-------ENDWITH
|
+-------ENDPROC
```

5.16.7 Decada (edeca)

```
+-------PROCEDURE VALID
&&Validamos el valor digitado para la decada de la fuente - articulo
|       +-------IF EMPTY(THIS.VALUE)
|       |               MESSAGEBOX("Decada no permitida !!",0,"Informacion")
|       |               THISFORM.PAGEFRAME1.PAGE2.BFINALIZAR.ENABLED=.F.
|       +-------ELSE
|       |               THISFORM.PAGEFRAME1.PAGE2.BFINALIZAR.ENABLED=.T.
|       +-------ENDIF
+-------ENDPROC
```

5.16.8 Periodicidad (eperifte)

```
+-------PROCEDURE VALID
&&Validamos el valor digitado para la periodicidad
|       +-------IF EMPTY(THIS.VALUE)
MESSAGEBOX("Periodicidad no permitida !!",0,"Informacion")
|       |               THISFORM.PAGEFRAME1.PAGE2.BFINALIZAR.ENABLED=.F.
|       +-------ELSE
```

```
|       |           THISFORM.PAGEFRAME1.PAGE2.BFINALIZAR.ENABLED=.T.
|       +-------ENDIF
+-------ENDPROC
```

5.17 FORMULARIO PARA LIMPIAR BASE DE DATOS (PIDEDIR.SCX)

5.17.1 Vistas

```
                v_rutabd
                v_registro

+-------PROCEDURE INIT
&&Revisamos la informacion del archivo de Ruta de Base de Datos
|           SELE V_RUTABD
&&Si trae informacion es por que tiene alguna ruta para la B.D.
|       +-------IF !EOF()
&&lo mostramos en pantalla
|       |           THISFORM.MITEXTBOX1.VALUE = V_RUTABD.RUTABD
|       |           PRUTABD = ALLTRIM(V_RUTABD.RUTABD)
|       |           THISFORM.LIMPIAR.ENABLED = .T.
|       |           THISFORM.WZBOTON4.ENABLED = .T.
&&En todo caso sugerimos que se revise si s la ruta actualizada de la B.D.
=MESSAGEBOX('Verifique que la ruta de la Base de Datos sea la actual..',0,'Confirmacion')
|       +-------ELSE
&&Si no trae informacion es por que aun no han asignado ruta a la B.D., por lo tanto solicitamos
&&que lo hagan
=MESSAGEBOX('Aun no se ha asignado ruta de la Base de Datos, por favor
asignela..',0,'Informacion')
|       |       THISFORM.WZBOTON1.SETFOCUS
|       +-------ENDIF
+-------ENDPROC
```

5.17.2 Boton Ubicar la Base de Datos

```
+-------PROCEDURE CLICK
|           LOCAL CDIR
&&Declaramos y cargamos variable con la ruta donde este un archivo
&&Se despliega la ventana de seleccionar archivo a partir de las unidades
&&disponibles en el equipo
                        CDIR = GETFILE('DBC', 'Selecciona la Base de Datos del Servidor:',
                        'Examinar',0)
&&Si no esta vacia la cadena resultante almacenamos la ruta
&&y la mostramos en la caja de texto
|       +-------IF NOT EMPTY(CDIR)
|       |           CDIR = JUSTPATH(CDIR) + '\'
|       |           THISFORM.MITEXTBOX1.VALUE = CDIR
```

```
|       |           *=WriteIni("DATABASE", "DataBasePath", cDir)
|       +-------ENDIF
|
+-------ENDPROC
```

5.17.3 Boton Aceptar

```
+-------PROCEDURE CLICK
&&Validamos que haya ruta asignada de la B.D.
|       +-------IF !EMPTY(ALLTRIM(THISFORM.MITEXTBOX1.VALUE))
|       |           SELE V_RUTABD
&&Si trae informacion buscamos la que esta activa hasta ese momento, la inactivamos
&&y activamos la nueva ruta de la B.D.
|       |       +-------IF RECCOUNT() <> 0
|       |       |           LOCATE FOR V_RUTABD.ACT_INA = .T.
|       |       |           REPLACE V_RUTABD.ACT_INA WITH .F.
&&Insertamos la Rura de la B.D.
INSERT INTO V_RUTABD(RUTABD, QUIEN, FECHARUTA, ACT_INA);

VALUES(ALLTRIM(THISFORM.MITEXTBOX1.VALUE), PQUIEN, DATE(), .T.)
|       |       +-------ELSE
&&Si no trae informacion es por que no se ha asignado ruta de la B.D. por lo tanto
&&insertamos la informacion
INSERT INTO V_RUTABD(RUTABD, QUIEN, FECHARUTA, ACT_INA);

VALUES(ALLTRIM(THISFORM.MITEXTBOX1.VALUE), PQUIEN, DATE(), .T.)
|       |       |           PRUTABD = ALLTRIM(THISFORM.MITEXTBOX1.VALUE)
|       |       +-------ENDIF
&&Confirmamos laactualizacion de la informacion en el archivo
|       |           TABLEUPDATE(.T.)
|       |           THISFORM.LIMPIAR.ENABLED = .T.
|       |           THISFORM.WZBOTON4.ENABLED = .T.
|       +-------ENDIF
|
|
+-------ENDPROC
```

5.17.4 Boton Salir

```
+-------PROCEDURE CLICK
&&salir de esta pantalla
|           THISFORM.RELEASE
+-------ENDPROC
```

5.17.5 Boton limpiar

```
+-------PROCEDURE CLICK
&&Reconsultamos la informacion del archivo de registro, el cual nos indica los usuarios
```

```
&&que estan trabajando el aplicativo en ese momento
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Si trae informacion es por que alguein esta trabajando
|     +-------IF !EOF()
&&En este caso revissamos si alguno de los usuarios que esta trabajando el aplicativo es
&&diferente al que esta tratando de procesar la orden
                        LOCATE FOR
                        ALLTRIM(V_REGISTRO.QUIEN)+ALLTRIM(V_REGISTRO.NPANTALLA) !=
                        ALLTRIM(PQUIEN)+WONTOP( )
|     |     +-------IF FOUND()
&&Si se encuentra a alguien mas no permitimos procesar la orden
=MESSAGEBOX('NO se puede procesar la orden, esta siendo usado el aplicativo ó no ha cerrado
todas las pantallas!!..',0+64+0,'Informacion')
|     |     +-------ELSE
&&En caso que no encuentre un usuario     diferente al que esta tratando de procesar la orden
&&procedemos a cerrar los archivo abiertos para que no se presenten problemas al tratar de
&&de usarlos en exclusivo
|     |     |           SELE V_REGISTRO
|     |     |           USE
|     |     |
|     |     |           SELE V_RUTABD
|     |     |           USE
&&Ejecutamos la barra de progreso, le asignamos valores minimos y maximos de avance
|     |     |           DO FORM FPROGRESO NAME PROGRESO LINKED
PROGRESO.LPROGRESO.CAPTION='Procesando limpieza de Base de Datos, espere por favor'
|     |     |         PROGRESO.BARRAPROGRESO1.MAXVALOR=2000
|     |     |         PROGRESO.BARRAPROGRESO1.MINVALOR=1
|     |     |
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
RESO+200
|     |     |           PROGRESO.REFRESH
|     |     |           PROGRESS()
&&Cerramos las bases de dato activas
|     |     |           CLOSE DATA ALL
&&Configuramos para poder acceder de manera exclusiva los archivos
|     |     |           SET EXCLUSIVE ON
&&Abrimos la Base de Datso que vamos a aplicarle el proceso
                OPEN DATABASE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'ipc80.dbc'
&&Empezamos a usar cada archivo para eliminar la informacion que esta marcada para tal proposito
&&sabemos cual esta marcada para borrar si el campo marca esta en .T.
                        USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'ipcmfte.dbf'
|     |     |           DELETE FROM IPCMFTE WHERE IPCMFTE.MARCA = .T.
|     |     |           PACK
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+200
```

```
|     |     |          PROGRESO.REFRESH
|     |     |          PROGRESS()
|     |     |
USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'ipcrfap.dbf'
|     |     |          DELETE FROM IPCRFAP WHERE IPCRFAP.MARCA = .T.
|     |     |          PACK
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+200
|     |     |          PROGRESO.REFRESH
|     |     |          PROGRESS()
|     |     |
        USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'ipcresp.dbf'
|     |     |          DELETE FROM IPCRESP WHERE IPCRESP.MARCA = .T.
|     |     |          PACK
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+100
|     |     |          PROGRESO.REFRESH
|     |     |          PROGRESS()
|     |     |
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+200
|     |     |          PROGRESO.REFRESH
|     |     |          PROGRESS()
|     |     |
        USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'ipctmvt.dbf'
|     |     |          DELETE FROM IPCTMVT WHERE IPCTMVT.MARCA = .T.
|     |     |          PACK
|     |     |          USE
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+200
|     |     |          PROGRESO.REFRESH
|     |     |          PROGRESS()
|     |     |
        USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'ipctmvt.dbf'

        USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'IPCTFCM.dbf'
|     |     |          DELETE FROM IPCTFCM WHERE IPCTFCM.MARCA = .T.
|     |     |          PACK
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+200
|     |     |          PROGRESO.REFRESH
|     |     |          PROGRESS()
|     |     |
```

```
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+200
|     |     |           PROGRESO.REFRESH
|     |     |           PROGRESS()
|     |     |
        USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'esptemp.dbf'
|     |     |           DELETE FROM ESPTEMP WHERE ESPTEMP.MARCA = .T.
|     |     |           PACK
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+200
|     |     |           PROGRESO.REFRESH
|     |     |           PROGRESS()
|     |     |
        USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'historia.dbf'
|     |     |           DELETE FROM HISTORIA WHERE HISTORIA.MARCA = .T.
|     |     |           PACK
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+200
|     |     |           PROGRESO.REFRESH
|     |     |           PROGRESS()
|     |     |
|     |     |           USE
|     |     |           CLOSE DATABASE ALL
|     |     |           PROGRESO.RELEASE
|     |     |           THISFORM.WZBOTON4.ENABLED = .F.
|     |     |           THISFORM.WZBOTON2.ENABLED = .F.
|     |     |           THIS.ENABLED = .F.
=MESSAGEBOX('Por favor salga del aplicativo y vuelva a entrar para realizar la
actualizacion..',0,'Informacion')
|     |     |
|     |     +-------ENDIF
|     +-------ELSE
&&Encaso que no haya usuario alguno trabajando simplementa procesamos lo mismo que en el paso
anterior
|     |           SELE V_REGISTRO
|     |           USE
|     |
|     |           SELE V_RUTABD
|     |           USE
|     |
|     |           DO FORM FPROGRESO NAME PROGRESO LINKED
PROGRESO.LPROGRESO.CAPTION='Procesando limpieza de Base de Datos, espere por favor'
|     |           PROGRESO.BARRAPROGRESO1.MAXVALOR=2000
|     |           PROGRESO.BARRAPROGRESO1.MINVALOR=1
|     |
```

```
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+200
|     |         PROGRESO.REFRESH
|     |         PROGRESS()
|     |
|     |         CLOSE DATA ALL
|     |         SET EXCLUSIVE ON
|     |
        OPEN DATABASE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'ipc80.dbc'
|     |
|     |         USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'ipcmfte.dbf'
|     |         DELETE FROM IPCMFTE WHERE IPCMFTE.MARCA = .T.
|     |         PACK
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+200
|     |         PROGRESO.REFRESH
|     |         PROGRESS()
|     |
|     |         USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'ipcrfap.dbf'
|     |         DELETE FROM IPCRFAP WHERE IPCRFAP.MARCA = .T.
|     |         PACK
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+200
|     |         PROGRESO.REFRESH
|     |         PROGRESS()
|     |
|     |         USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'ipcresp.dbf'
|     |         DELETE FROM IPCRESP WHERE IPCRESP.MARCA = .T.
|     |         PACK
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+100
|     |         PROGRESO.REFRESH
|     |         PROGRESS()
|     |
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+200
|     |         PROGRESO.REFRESH
|     |         PROGRESS()
|     |
|     |         USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'ipctmvt.dbf'
|     |         DELETE FROM IPCTMVT WHERE IPCTMVT.MARCA = .T.
|     |         PACK
|     |         USE
```

```
&&Avanzamos la barra de progreso
PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROGRESO+
200
|     |         PROGRESO.REFRESH
|     |         PROGRESS()
|     |
|     |         USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'ipctmvt.dbf'
|     |         USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'IPCTFCM.dbf'
|     |         DELETE FROM IPCTFCM WHERE IPCTFCM.MARCA = .T.
|     |         PACK
&&Avanzamos la barra de progreso
PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROGRESO+
200
|     |         PROGRESO.REFRESH
|     |         PROGRESS()
&&Avanzamos la barra de progreso
PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROGRESO+
200
|     |         PROGRESO.REFRESH
|     |         PROGRESS()
|     |
|     |         USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'esptemp.dbf'
|     |         DELETE FROM ESPTEMP WHERE ESPTEMP.MARCA = .T.
|     |         PACK
&&Avanzamos la barra de progreso
PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROGRESO+
200
|     |         PROGRESO.REFRESH
|     |         PROGRESS()
|     |
|     |         USE ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'historia.dbf'
|     |         DELETE FROM HISTORIA WHERE HISTORIA.MARCA = .T.
|     |         PACK
&&Avanzamos la barra de progreso
PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROGRESO+
200
|     |         PROGRESO.REFRESH
|     |         PROGRESS()
|     |
|     |         USE
|     |         CLOSE DATABASE ALL
|     |         PROGRESO.RELEASE
|     |         THISFORM.WZBOTON4.ENABLED = .F.
|     |         THISFORM.WZBOTON2.ENABLED = .F.
|     |         THIS.ENABLED = .F.
=MESSAGEBOX('Por favor salga del aplicativo y vuelva a entrar para realizar la
actualizacion..',0,'Informacion')
|     +-------ENDIF
```

```
+-------ENDPROC
+-------PROCEDURE GOTFOCUS
&&Validamos que el usuario que esta tratando de procesar esta orden tenga nivel de acceso
&&para hacerlo, apartir de esto lo permitimos o no
|       +-------IF PNIVEL != "1"
=MESSAGEBOX('No tiene nivel de acceso para procesar esta orden..',0+64+0,'Informacion')
|       |               THIS.ENABLED = .F.
|       |               THISFORM.WZBOTON4.ENABLED = .F.
|       |               THISFORM.WZBOTON3.SETFOCUS
|       +-------ENDIF
|
+-------ENDPROC
```

5.17.6 Boton Reindexar

```
+-------PROCEDURE CLICK
&&Reconsultamos la informacion del archivo de registro, el cual nos indica los usuarios
&&que estan trabajando el aplicativo en ese momento
|               REQUERY('v_registro')
|               SELE V_REGISTRO
&&Si trae informacion es por que alguein esta trabajando
|       +-------IF !EOF()
&&En este caso revissamos si alguno de los usuarios que esta trabajando el aplicativo es
&&diferente al que esta tratando de procesar la orden
                                LOCATE FOR
                                ALLTRIM(V_REGISTRO.QUIEN)+ALLTRIM(V_REGISTRO.NPANTALLA) !=
                                ALLTRIM(PQUIEN)+WONTOP( )
|       |       +-------IF FOUND()
&&Si se encuentra a alguien mas no permitimos procesar la orden
=MESSAGEBOX('NO se puede procesar la orden, esta siendo usado el aplicativo ó no ha cerrado
todas las pantallas!!..',0+64+0,'Informacion')
|       |       +-------ELSE
&&En caso que no encuentre un usuario     diferente al que esta tratando de procesar la orden
&&procedemos a cerrar los archivo abiertos para que no se presenten problemas al tratar de
&&de usarlos en exclusivo
|       |       |               SELE V_REGISTRO
|       |       |               USE
|       |       |
|       |       |               SELE V_RUTABD
|       |       |               USE
PRUTABD = ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'ipc80.dbc'
|       |       |
|       |       |
&&Solicitamos que confirmen requerir ejecutar el proceso de reindexar
MENSAJE=MESSAGEBOX(" Está seguro de requerir REINDEXAR ?",4+64+256," INFORMACION ")
|       |       |       +-------DO CASE
&&En caso afirmativo ejecutamos la barra de progreso y asignamos
&&valores inicial y final de avance
```

```
|     |     |     +-------CASE MENSAJE=6
|     |     |     |           DO FORM FPROGRESO NAME PROGRESO LINKED
PROGRESO.LPROGRESO.CAPTION=' Reindexando Base de Datos, Espere por favor ......'
|     |     |     |           PROGRESO.BARRAPROGRESO1.MAXVALOR=2000
|     |     |     |           PROGRESO.BARRAPROGRESO1.MINVALOR=1
|     |     |     |
&&Ejecutamos el proceso de eliminado de inconsistencias
&&de los diferentes archivos de la base de datos
|     |     |     |
|     |           ***** Configuración de tabla para IPCMFTE *****
&&Enviamos mensaje a la barra de progreso el cual se muestra
&&mientras esta avanzando la misma
|     |     |     |
&&El siguiente procedimiento es el mismo para todos los archivo
&&Activamos la rutina que nos devolveria un mensaje de advertencia
&&para cuando estan siendo utilizados los archivos o no existe
&&la llave
|     |     |     |
&&Eliminamos la llave principal del archivo y los demas indices
PROGRESO.LPROGRESO.CAPTION=' Reindexando Fuentes, Espere por favor ......'
|     |     |     |
|     |     | ***** Cambiar propiedades para IPCMFTE *****
&&Avanzamos la barra de progreso
|     |     |     |           PROGRESS()
|     |     |     |           PROGRESO.REFRESH
|     |     |     |
|     |           ***** Configuración de tabla para IPCMART *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Maestro de Articulos, Espere por favor ......'
|     |     |     |
|
|           ***** Cambiar propiedades para IPCMART *****
|     |     |     |           PROGRESS()
|     |     |     |           PROGRESO.REFRESH
|     |     |     |
|     |           ***** Configuración de tabla para IPCRFAP *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Fuente - Articulo, Espere por favor ......'
|     |     |     |
|
|           ***** Cambiar propiedades para IPCRFAP *****
|     |     |     |           PROGRESS()
|     |     |     |           PROGRESO.REFRESH
|     |     |     |
|     |           ***** Configuración de tabla para IPCRESP *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Especificaciones, Espere por favor ......'
|     |     |     |
|     |             ***** Cambiar propiedades para IPCRESP *****
|     |     |     |           PROGRESS()
|     |     |     |           PROGRESO.REFRESH
```

```
|     |     |     |
|     |           ***** Configuración de tabla para IPCTMVT *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Movimiento, Espere por favor ......'
|     |     |     |
|     |     |     |           PROGRESS()
|     |     |     |           PROGRESO.REFRESH
|     |     |     |
|     |            ***** Configuración de tabla para IPCTCTV *****
|     |           ***** Crear cada índice para IPCTCTV *****
|     |           ***** Cambiar propiedades para IPCTCTV *****
|     |     |     |           PROGRESS()
|     |     |     |           PROGRESO.REFRESH
|     |     |     |
|     |           ***** Configuración de tabla para IPCTFCM *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Fuentes Complementarias, Espere por favor
......'
|     |     |     |
|     |     |     |           PROGRESS()
|     |     |     |           PROGRESO.REFRESH
|     |     |     |
|     |           ***** Configuración de tabla para IPCROBS *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Observaciones, Espere por favor ......'
|     |     |     |
|     |     |     |           PROGRESS()
|     |     |     |           PROGRESO.REFRESH
|     |     |     |
|     |           ***** Configuración de tabla para IPCESTR *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Estrato, Espere por favor ......'
|     |     |     |
|     |           ***** Cambiar propiedades para IPCESTR *****
|     |     |     |           PROGRESS()
|     |     |     |           PROGRESO.REFRESH
|     |     |     |
|     |           ***** Configuración de tabla para IPCRANGO *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Rango, Espere por favor ......'
|     |     |     |
||     |            ***** Cambiar propiedades para IPCRANGO *****
|     |     |     |           PROGRESS()
|     |     |     |           PROGRESO.REFRESH
|     |     |     |
|     |           ***** Configuración de tabla para IPCTESP *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Esp. Temporales, Espere por favor ......'
|     |     |     |           PROGRESS()
|     |     |     |           PROGRESO.REFRESH
|     |     |     |
|     |           ***** Configuración de tabla para IPCRPART *****
|     |           ***** Crear cada índice para IPCRPART *****
|     |     |     |           PROGRESS()
```

```
|     |     |     |           PROGRESO.REFRESH
|     |     |     |
|     |           ***** Configuración de tabla para ESPTEMP *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Historico de Especificaciones, Espere por favor
......'
|     |     |     |           PROGRESS()
|     |     |     |           PROGRESO.REFRESH
|     |     |     |
|     |     |     |           PROGRESO.BARRAPROGRESO1.LLENE
|     |     |     |           PROGRESO.RELEASE
|     |     |     |
|     |     |     |           THISFORM.LIMPIAR.ENABLED = .F.
|     |     |     |           THISFORM.WZBOTON2.ENABLED = .F.
|     |     |     |           THIS.ENABLED = .F.

= MESSAGEBOX( " Proceso Terminado.... "+CHR(13)+CHR(13)+" Es recomendable salir de la
aplicacion y volver a entrar "+CHR(13)+CHR(13)+" Para que tenga efecto el proceso de Reindexar
!!",0+64+0," INFORMACION ")
|     |     |     +-------CASE MENSAJE=7
= MESSAGEBOX(" Proceso Cancelado por el Usuario.... ",0+64+0," INFORMACION ")
|     |     |     +-------ENDCASE
|     |     |
|     |     +-------ENDIF
|     +-------ELSE
|     |           SELE V_REGISTRO
|     |           USE
|     |
|     |           SELE V_RUTABD
|     |           USE
|     |           PRUTABD = ALLTRIM(THISFORM.MITEXTBOX1.VALUE)+'ipc80.dbc'
|     |
&&Solicitamos que confirmen requerir ejecutar el proceso de reindexar
MENSAJE=MESSAGEBOX(" Está seguro de requerir REINDEXAR ?",4+64+256," INFORMACION ")
|     |     +-------DO CASE
&&En caso afirmativo ejecutamos la barra de progreso y asignamos
&&valores inicial y final de avance
|     |     +-------CASE MENSAJE=6
|     |     |           DO FORM FPROGRESO NAME PROGRESO LINKED
PROGRESO.LPROGRESO.CAPTION=' Reindexando Base de Datos, Espere por favor ......'
|     |     |           PROGRESO.BARRAPROGRESO1.MAXVALOR=2000
|     |     |           PROGRESO.BARRAPROGRESO1.MINVALOR=1
|     |     |
&&Ejecutamos el proceso de eliminado de inconsistencias
&&de los diferentes archivos de la base de datos
|     |     |
|     |     |           ***** Configuración de tabla para IPCMFTE *****
&&Enviamos mensaje a la barra de progreso el cual se muestra
&&mientras esta avanzando la misma
```

```
|     |     |
&&El siguiente procedimiento es el mismo para todos los archivo
&&Activamos la rutina que nos devolveria un mensaje de advertencia
&&para cuando estan siendo utilizados los archivos o no existe
&&la llave
|     |     |
&&Eliminamos la llave principal del archivo y los demas indices
PROGRESO.LPROGRESO.CAPTION=' Reindexando Fuentes, Espere por favor .....'
|     |     |
|     |     |           ***** Cambiar propiedades para IPCMFTE *****
&&Avanzamos la barra de progreso
|     |     |           PROGRESS()
|     |     |           PROGRESO.REFRESH
|     |     |
|     |     |           ***** Configuración de tabla para IPCMART *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Maestro de Articulos, Espere por favor .....'
|     |     |
|     |     |           ***** Cambiar propiedades para IPCMART *****
|     |     |           PROGRESS()
|     |     |           PROGRESO.REFRESH
|     |     |
|     |     |           ***** Configuración de tabla para IPCRFAP *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Fuente - Articulo, Espere por favor .....'
|     |     |
|     |     |           ***** Cambiar propiedades para IPCRFAP *****
|     |     |           PROGRESS()
|     |     |           PROGRESO.REFRESH
|     |     |
|     |     |           ***** Configuración de tabla para IPCRESP *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Especificaciones, Espere por favor .....'
|     |     |
|     |     |           ***** Cambiar propiedades para IPCRESP *****
|     |     |           PROGRESS()
|     |     |           PROGRESO.REFRESH
|     |     |
|     |     |           ***** Configuración de tabla para IPCTMVT *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Movimiento, Espere por favor .....'
|     |     |
|     |     |           PROGRESS()
|     |     |           PROGRESO.REFRESH
|     |     |
|     |     |           ***** Configuración de tabla para IPCTCTV *****
|     |     |           ***** Crear cada índice para IPCTCTV *****
|     |     |           ***** Cambiar propiedades para IPCTCTV *****
|     |     |           PROGRESS()
|     |     |           PROGRESO.REFRESH
|     |     |
|     |     |           ***** Configuración de tabla para IPCTFCM *****
```

```
PROGRESO.LPROGRESO.CAPTION=' Reindexando Fuentes Complementarias, Espere por favor
......'
|     |     |
|     |     |           PROGRESS()
|     |     |           PROGRESO.REFRESH
|     |     |
|     |     |           ***** Configuración de tabla para IPCROBS *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Observaciones, Espere por favor ......'
|     |     |
|     |     |           PROGRESS()
|     |     |           PROGRESO.REFRESH
|     |     |
|     |     |           ***** Configuración de tabla para IPCESTR *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Estrato, Espere por favor ......'
|     |     |
|     |     |           ***** Cambiar propiedades para IPCESTR *****
|     |     |           PROGRESS()
|     |     |           PROGRESO.REFRESH
|     |     |
|     |     |           ***** Configuración de tabla para IPCRANGO *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Rango, Espere por favor ......'
|     |     |
|     |     |           ***** Cambiar propiedades para IPCRANGO *****
|     |     |           PROGRESS()
|     |     |           PROGRESO.REFRESH
|     |     |
|     |     |           ***** Configuración de tabla para IPCTESP *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Esp. Temporales, Espere por favor ......'
|     |     |           PROGRESS()
|     |     |           PROGRESO.REFRESH
|     |     |
|     |     |           ***** Configuración de tabla para IPCRPART *****
|     |     |           ***** Crear cada índice para IPCRPART *****
|     |     |           PROGRESS()
|     |     |           PROGRESO.REFRESH
|     |     |
|     |     |           ***** Configuración de tabla para ESPTEMP *****
PROGRESO.LPROGRESO.CAPTION=' Reindexando Historico de Especificaciones, Espere por favor
......'
|     |     |           PROGRESS()
|     |     |           PROGRESO.REFRESH
|     |     |
|     |     |           PROGRESO.BARRAPROGRESO1.LLENE
|     |     |           PROGRESO.RELEASE
|     |     |
|     |     |           THISFORM.LIMPIAR.ENABLED = .F.
|     |     |           THISFORM.WZBOTON2.ENABLED = .F.
|     |     |           THIS.ENABLED = .F.
```

```
= MESSAGEBOX( " Proceso Terminado.... "+CHR(13)+CHR(13)+" Es recomendable salir de la
aplicacion y volver a entrar "+CHR(13)+CHR(13)+" Para que tenga efecto el proceso de Reindexar
!!",0+64+0," INFORMACION ")
|       |       +-------CASE MENSAJE=7
= MESSAGEBOX(" Proceso Cancelado por el Usuario.... ",0+64+0," INFORMACION ")
|       |       +-------ENDCASE
|       +-------ENDIF
+-------ENDPROC

+-------PROCEDURE GOTFOCUS
|       +-------IF PNIVEL != "1"
=MESSAGEBOX('No tiene nivel de acceso para procesar esta orden..',0+64+0,'Informacion')
|       |               THIS.ENABLED = .F.
|       |               THISFORM.LIMPIAR.ENABLED = .F.
|       |               THISFORM.WZBOTON3.SETFOCUS
|       +-------ENDIF
|
+-------ENDPROC
```

5.18 FORMULARIO PARA EL CIERRE (FSPUBCIERRE.SCX)

5.18.1 Entorno de Datos

```
+-------PROCEDURE INIT
|               SELE V_IPCTCTV
|               INDEX ON FINCTV TAG PKFINCTV
|
|               SELE V_IPCTMVT
|               INDEX ON CODFTE+CODART TAG NKMVT
|
|               SELECT V_PREFCM
|               INDEX ON STR(CODFCOM) + CODFTE TAG PKFCM
|
|               SELE V_IPCRFAP
|               INDEX ON CODFTE+CODART TAG NKFAP
+-------ENDPROC
```

5.18.2 Vistas

v_ipctctv
v_ipctmvt
v_ipcrobs
v_ipcrfap
v_ipcresp
v_ipcmart

```
                v_ipcmfte
                v_ipcrpart
                v_prefcm
                v_registro
                v_historia
                v_esptemp

+-------PROCEDURE ACTIVATE
&&Al entrar al formulario actualizamos el archivo de registro para saber
&&que este usuario abrio este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo requery('v_registro')
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para saber
&&si ya esta relacionado en el archivo y no volver a registrarlo
LOCATE FOR (ALLTRIM(V_REGISTRO.QUIEN) + NPANTALLA = ALLTRIM(PQUIEN) + WONTOP())
AND V_REGISTRO.ACT_INA = .T.
&&si no es encontrado lo registramos y actualizamos el archivo
|     +-------IF !FOUND()
|     |       INSERT INTO V_REGISTRO(QUIEN, NPANTALLA, FECHAR, ACT_INA);
|     |       VALUES(PQUIEN,WONTOP( ), DATE(), .T.)
|     |           TABLEUPDATE(.T.)
|     +-------ENDIF
|
+-------ENDPROC

+-------PROCEDURE DESTROY
&&Al salir del formulario actualizamos el archivo de registro para saber
&&que este usuario ya salio de este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para indicar
&&que la cerro en este momento
LOCATE FOR ALLTRIM(V_REGISTRO.QUIEN)+ALLTRIM(V_REGISTRO.NPANTALLA) =
ALLTRIM(PQUIEN)+WONTOP( )
|     +-------IF FOUND()
|     |           REPLACE V_REGISTRO.ACT_INA WITH .F.
&&Actualizamos la informacion y tambien el archivo
|     |           TABLEUPDATE(.T.)
|     +-------ENDIF
+-------ENDPROC

+-------PROCEDURE INIT
&&Configuramos para el manejo de formatos de fecha
|           SET CENTURY ON
```

```
|           SET DATE DMY
|           SET TALK OFF
&&Seleccionamos el archivo de consecutivos, tomamos los datos del
&&ultimo registro para tomarlos como base para el prediligenciamiento
&&Tomamos el ultimo consecutivo utilizado para empezar el nuevo
&&prediligenciamiento
|           SELE V_IPCTCTV
|           SET ORDER TO PKFINCTV
&&vamos al ultimo registro y mostramos esa informacion en pantalla
&&para indicar que ese es el periodo a procesar
|           GO BOTTOM
|           THISFORM.TEXT4.VALUE = ALLTRIM(STR(V_IPCTCTV.AÑOCTV))
|           THISFORM.TEXT5.VALUE = ALLTRIM(STR(V_IPCTCTV.MESCTV))
|           THISFORM.TEXT6.VALUE = ALLTRIM(STR(V_IPCTCTV.DECCTV))
|
&&Declaramos y cargamos variables
                        PUBLIC X, FECHAANO, FECHAMES, FECHADECADA, CONTAR,
                        REPFUENTES, REPREPORTE
|           STORE 'reportefTes.frx' TO REPFUENTES
|           STORE 'reporte.frx' TO REPREPORTE
|
|           SELE V_IPCTCTV
|           GO BOTTOM
&&Cargamos la variable con el consecutivo final utilizado en el periodo
&&inmediatamente anterior y lo incrementamos en uno
|           X = V_IPCTCTV.FINCTV
|           X = X+1
|
|           FECHAANO = VAL(THISFORM.TEXT4.VALUE)
|           FECHAMES = VAL(THISFORM.TEXT5.VALUE)
|           FECHADECADA = VAL(THISFORM.TEXT6.VALUE)
|
|           THISFORM.CIERRE.SETFOCUS
|
+-------ENDPROC
```

5.18.3 Boton Cierre

```
+-------PROCEDURE CLICK
&&Pedimos confirmar la decision de hacer cierre
                MENSAJE=MESSAGEBOX(" Esta seguro de hacer cierre ? ",4+32+256," Cierre ! ")
|       +-------DO CASE
|       +-------CASE MENSAJE=6
&&Si confirmam positivamente cerramos el archivo de movimiento
|       |
&&Ejecutamos la barra de progreso indicando el periodo de proceso,: dia, mes, año
|       |           DO FORM FPROGRESO NAME PROGRESO LINKED
```

```
PROGRESO.LPROGRESO.CAPTION=" Procesando cierre "+" Decada:
"+ALLTRIM(STR(FECHADECADA))+" Mes: "+ALLTRIM(STR(FECHAMES))+" Año:
"+ALLTRIM(STR(FECHAANO))+" espere por favor"
|     |
|     |           REQUERY('v_ipcrfap')
&&Usamos el archivo de movimiento, lo ordenamos, creamos una relacion con
&&el archivo fuente articulo segun la llave fuente-articulo
|     |           SELE V_IPCTMVT
|     |           SET ORDE TO TAG NKMVT
|     |           SET RELA TO (CODFTE+CODART) INTO V_IPCRFAP
&&Configuramos los valores inicial y final de avance de la barra de progreso
|     |           PROGRESO.BARRAPROGRESO1.MAXVALOR=(RECCOUNT()+500)
|     |           PROGRESO.BARRAPROGRESO1.MINVALOR=1
|     |
|     |           GO TOP
&&Recorremos en el archivo de movimiento los que esten capturados, es decir,
&&que tenga precio actual o periodo de espera
|     |     +-------SCAN  FOR (V_IPCTMVT.MARCA=.F.)
&&Avanzamos la barra de progreso
|     |     |           PROGRESO.BARRAPROGRESO1.PROGRESE
&&Si trae marcado periodo de espera, actualizamos en fuente articulo esta novedad
&&en la fuente articulo correspondiente
|     |     |     +-------IF V_IPCTMVT.PESMVT=.T.
REPLACE V_IPCRFAP.PESART WITH V_IPCTMVT.PESMVT;
|     |     |     |                 IN V_IPCRFAP
|     |     |     |           PROGRESO.BARRAPROGRESO1.PROGRESE
|     |     |     +-------ELSE
&&Si no trae periodo de espera, suponemos que trae precio actual, por lo tanto
&&actualizamos en fuente articulo los datos correspondientes, es decir,
&&precio actual pasa a ser precio anterior y asi con todas las novedades, pasan a ser
&&anteriores
REPLACE V_IPCRFAP.PANART WITH V_IPCTMVT.PACTMVT;
V_IPCRFAP.CANTANT WITH V_IPCTMVT.CANRMVT;
V_IPCRFAP.PESART WITH V_IPCTMVT.PESMVT;
V_IPCRFAP.FCOMMVT WITH V_IPCTMVT.FCOMMVT;
IN V_IPCRFAP
|     |     |     |           PROGRESO.BARRAPROGRESO1.PROGRESE
|     |     |     +-------ENDIF
&&Termina el recorreido del archivo
|     |     +-------ENDSCAN
&&Deshacemos la relacion entre movimiento y fuente articulo
|     |           SET RELA TO
|     |
|     |           SELE V_IPCRFAP
|     |           TABLEUPDATE(.T.)
|     |
&&Evaluamos el estado del archivo de fuentes, si esta siendo usada
&&la cerramos, para luego abrirla en modo exclusivo
```

```
|       |           SELE V_IPCMFTE
&&Marcamos para eliminar del maestro de fuentes las fuentes que no esten
&&en el archivo de fuente articulo, es decir, las fuentes a las que se le
&&hayan retirado todos los articulos y no van volver a salir prediligenciadas
UPDATE V_IPCMFTE SET MARCA = .T. WHERE V_IPCMFTE.CODFTE NOT IN(SELECT
DISTINCT V_IPCRFAP.CODFTE FROM V_IPCRFAP)
&&Eliminamos fisicamente los que esten marcados para eliminar
|       |           TABLEUPDATE(.T.)
&&Hacemos lo mismo con el archivo de Especificaciones, eliminamos las
&&especificaciones correspondientes a fuente articulos que ya no existan
&&en el archivo fuente articulo
|       |           SELE V_IPCRESP
UPDATE V_IPCRESP SET MARCA = .T. WHERE (V_IPCRESP.CODFTE + V_IPCRESP.CODART)
NOT IN(SELECT DISTINCT (V_IPCRFAP.CODFTE + V_IPCRFAP.CODART) FROM V_IPCRFAP)
|       |           TABLEUPDATE(.T.)
&&Cerramos la barra de progreso
|       |           PROGRESO.RELEASE
&&Habilitamos los botones para que puedan continuar con el proceso
|       |           THISFORM.CIERRE.ENABLED=.F.
|       |           THISFORM.DANE.ENABLED=.T.
&&Cancelado el proceso de cierre
|       +-------CASE MENSAJE=7
MENSAJE=MESSAGEBOX(" Cierre Cancelado ",0+64+0," Cierre ! ")
|       |           THISFORM.CIERRE.ENABLED=.T.
|       |           THISFORM.DANE.ENABLED=.F.
|       |           THISFORM.CONFIRMAR.ENABLED=.F.
|       +-------ENDCASE
+-------ENDPROC
```

5.18.4 Boton salir

```
+-------PROCEDURE CLICK
&&Salir de esta pantalla
|           THISFORM.RELEASE
+-------ENDPROC
```

5.18.5 Boton Prediligenciar

```
+-------PROCEDURE CLICK
|           SET SAFETY OFF
&&Configuramos el ambiente de trabajo, definimos e inicializamos variables
|           LOCAL SEMES,ANUAL
|
|           SEMES = " "
|           ANUAL = " "
&&Cargamos la informacion del periodo actual en variables
|           NANO = VAL(THISFORM.TEXT4.VALUE)
```

```
|           DECADA = VAL(THISFORM.TEXT6.VALUE)
|           NMES = VAL(THISFORM.TEXT5.VALUE)
|
|&&Solicitamos confirmar procesar prediligenciamiento
                        MENSAJE = MESSAGEBOX(" Esta seguro de generar Prediligenciamiento ?
                        ",4+32+256," Generar Prediligenciamiento ! ")
|     +-------DO CASE
&&Si responde afirmativamente
|     +-------CASE MENSAJE=6
&&Preguntamos si desea incluir los articulos semestrales al prediligenciamiento
                        MENSAJE1=MESSAGEBOX(" Incluir SEMESTRALES al Prediligenciamiento ?
                        ",4+32+256," Generar Prediligenciamiento ! ")
|     |     +-------DO CASE
&&En caso afirmativo, preparamos la informacion para cambiarle la decada
&&y asi salgan prediligenciada
|     |     +-------CASE MENSAJE1=6
|     |     |           SEMES="SI"
|     |     |           UPDATE V_IPCRFAP;
|     |     |                 SET DECART = DECADA;
|     |     |                 WHERE SUBSTR(V_IPCRFAP.CODART,1,3)="512"
&&Preguntamos si requiere incluir los articulos anuales al prediligenciamiento
MENSAJE2=MESSAGEBOX(" Incluir ANUALES al Prediligenciamiento ? ",4+32+256," Generar
Prediligenciamiento ! ")
|     |     |     +-------DO CASE
&&En caso afirmativo, preparamos la informacion para cambiarle la decada y
&&asi salgan prediligenciada
|     |     |     +-------CASE MENSAJE2=6
|     |     |     |           ANUAL="SI"
|     |     |     |           UPDATE V_IPCRFAP;
|     |     |     |                 SET DECART = DECADA;
WHERE (SUBSTR(V_IPCRFAP.CODART,1,4) = "5110" OR SUBSTR(V_IPCRFAP.CODART,1,4) =
"5130" OR SUBSTR(V_IPCRFAP.CODART,1,5) = "52101")
&&En caso negativo, modificamos la decada que tengan actualmente los articulos anuales
&&para que no salgan prediligenciadas
&&En caso de tener decada 1 incrementamos en uno la decada para que no salgan
|     |     |     +-------CASE MENSAJE2 = 7
|     |     |     |           ANUAL = "NO"
|     |     |     |     +-------IF DECADA = 1
|     |     |     |     |           UPDATE V_IPCRFAP;
|     |     |     |     |                 SET DECART = DECADA+1;
WHERE (SUBSTR(V_IPCRFAP.CODART,1,4) = "5110" OR SUBSTR(V_IPCRFAP.CODART,1,4) =
"5130" OR SUBSTR(V_IPCRFAP.CODART,1,5) = "52101")
&&En caso de tener decada 2 incrementamos en uno la decada para que no salgan
|     |     |     |     +-------ELSE
|     |     |     |     |     +-------IF DECADA = 2
|     |     |     |     |     |           UPDATE V_IPCRFAP;
SET DECART = DECADA + 1;
```

```
WHERE (SUBSTR(V_IPCRFAP.CODART,1,4) = "5110" OR SUBSTR(V_IPCRFAP.CODART,1,4) =
"5130" OR SUBSTR(V_IPCRFAP.CODART,1,5) = "52101")
&&En caso de tener decada 3 decrementamos en dos la decada para que no salgan
|     |     |     |     |     +-------ELSE
|     |     |     |     |     |     +-------IF DECADA = 3
UPDATE V_IPCRFAP;
SET DECART = DECADA - 2;
WHERE (SUBSTR(V_IPCRFAP.CODART,1,4) = "5110" OR SUBSTR(V_IPCRFAP.CODART,1,4) =
"5130" OR SUBSTR(V_IPCRFAP.CODART,1,5) = "52101")
|     |     |     |     |     |     |
|     |     |     |     |     |     +-------ENDIF
|     |     |     |     |     +-------ENDIF
|     |     |     |     +-------ENDIF
|     |     |     +-------ENDCASE
&&Lo mismo para los semestrales
&&Si tienen decada 1, lo incermentamos en uno la decada para que no salgan
|     |     +-------CASE MENSAJE1 = 7
|     |     |           SEMES = "NO"
|     |     |     +-------IF DECADA = 1
|     |     |     |           UPDATE V_IPCRFAP;
|     |     |     |                 SET DECART = DECADA + 1;
WHERE SUBSTR(V_IPCRFAP.CODART,1,3) = "512"
&&Si tienen decada 2, lo incermentamos en uno la decada para que no salgan
|     |     |     +-------ELSE
|     |     |     |     +-------IF DECADA = 2
|     |     |     |     |           UPDATE V_IPCRFAP;
|     |     |     |     |                 SET DECART = DECADA + 1;
WHERE SUBSTR(V_IPCRFAP.CODART,1,3) = "512"
&&Si tienen decada 3, lo decrementamos en dos la decada para que no salgan
|     |     |     |     +-------ELSE
|     |     |     |     |     +-------IF DECADA=3
|     |     |     |     |     |           UPDATE V_IPCRFAP;
SET DECART = DECADA - 2;
WHERE SUBSTR(V_IPCRFAP.CODART,1,3) = "512"
|     |     |     |     |     |
|     |     |     |     |     +-------ENDIF
|     |     |     |     +-------ENDIF
|     |     |     +-------ENDIF
&&En caso que hayan seleccionado no incluir los semestrales al prediligenciamiento
&&averiguamos de todas maneras por los anuales
MENSAJE2=MESSAGEBOX(" Incluir ANUALES al Prediligenciamiento ? ",4+32+256," Generar
Prediligenciamiento ! ")
|     |     |     +-------DO CASE
&&En caso afirmativo modificamos la decada para que sean incluidos en el prediligenciamiento
|     |     |     +-------CASE MENSAJE2 = 6
|     |     |     |           ANUAL = "SI"
|     |     |     |           UPDATE V_IPCRFAP;
|     |     |     |                 SET DECART = DECADA;
```

```
WHERE (SUBSTR(V_IPCRFAP.CODART,1,4) = "5110" OR SUBSTR(V_IPCRFAP.CODART,1,4) =
"5130" OR SUBSTR(V_IPCRFAP.CODART,1,5) = "52101")
&&De lo contrario modificamos la decada para que no sean incluidos en el prediligenciamiento
|     |     |     +-------CASE MENSAJE2 = 7
|     |     |     |           ANUAL = "NO"
&&Realizamos las mismas modificaciones en las decadas para que no salgan prediligenciados
|     |     |     |     +-------IF DECADA = 1
|     |     |     |     |           UPDATE V_IPCRFAP;
|     |     |     |     |                 SET DECART = DECADA + 1;
WHERE (SUBSTR(V_IPCRFAP.CODART,1,4) = "5110" OR SUBSTR(V_IPCRFAP.CODART,1,4) =
"5130" OR SUBSTR(V_IPCRFAP.CODART,1,5) = "52101")
|     |     |     |     +-------ELSE
|     |     |     |     |     +-------IF DECADA = 2
|     |     |     |     |     |           UPDATE V_IPCRFAP;
SET DECART = DECADA + 1;
WHERE (SUBSTR(V_IPCRFAP.CODART,1,4) = "5110" OR SUBSTR(V_IPCRFAP.CODART,1,4) =
"5130" OR SUBSTR(V_IPCRFAP.CODART,1,5) = "52101")
|     |     |     |     |     |
|     |     |     |     |     +-------ELSE
|     |     |     |     |     |     +-------IF DECADA = 3
UPDATE V_IPCRFAP;
SET DECART = DECADA - 2;
WHERE (SUBSTR(V_IPCRFAP.CODART,1,4) = "5110" OR SUBSTR(V_IPCRFAP.CODART,1,4) =
"5130" OR SUBSTR(V_IPCRFAP.CODART,1,5) = "52101")
|     |     |     |     |     |     |
|     |     |     |     |     |     +-------ENDIF
|     |     |     |     |     +-------ENDIF
|     |     |     |     +-------ENDIF
|     |     |     +-------ENDCASE
|     |     +-------ENDCASE
&&Damos la oportunidad de devolverse a corregir lo seleccionado anteriormente
MENSAJE3=MESSAGEBOX(" USTED HA SELECCIONADO IMPRIMIR: "+CHR(13)+CHR(13)+"
SEMESTRALES: "+ALLTRIM(SEMES)+CHR(13)+CHR(13)+"       ANUALES:
"+ALLTRIM(ANUAL)+CHR(13)+CHR(13)+" CONFIRMA LA SELECCION ? ",4+32+0,"
ADVERTENCIA ")
|     |     +-------DO CASE
&&Si confirma lo seleccionado
|     |     +-------CASE MENSAJE3=6
&&Ejecutamos la barra de progreso mostrando los datos del periodo en proceso
&&mes, decada, año
|     |     |           SELE V_IPCRFAP
|     |     |           TABLEUPDATE(.T.)
|     |     |           REQUERY('v_ipcrfap')
|     |     |
|     |     |           DO FORM FPROGRESO NAME PROGRESO LINKED
                                    PROGRESO.LPROGRESO.CAPTION=" Prediligenciando
                                    década "+ALLTRIM(STR(DECADA))+" Mes:
```

```
                                        "+ALLTRIM(STR(NMES))+" Año:
                                        "+ALLTRIM(STR(NANO))
|     |     |           PROGRESO.BARRAPROGRESO1.MAXVALOR=1000
|     |     |           PROGRESO.BARRAPROGRESO1.MINVALOR=1
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+100
|     |     |           PROGRESO.REFRESH
|     |     |           PROGRESS()
&&Evaluamos la decada en proceso, si es decada 1 descargamos la captura, es decir,
&&el contenido del archivo de movimiento en el archivo de historicos para mantener
&&la informacion almacenada en caso de requerirse consultar o reimprimir
|     |     |
|     |     |           SELE V_HISTORIA
|     |     |     +-------IF DECADA=1
|     |     |     |           SELE V_IPCTMVT
|     |     |     |           GO TOP
|     |     |     |     +-------SCAN FOR V_IPCTMVT.MARCA = .F.
|     |     |     |     |           SCATTER MEMVAR
INSER INTO V_HISTORIA(CODFTE, CODART, IMPCTV, PANART, PACTMVT, CMVMVT,
CRFMVT, PESMVT, SINMMVT, FCOMMVT, CANRMVT, OBSMVT, DECMVT, CANTANT, ANO,
MES, PESART, VARIACION, MARCA);
VALUES(V_IPCTMVT.CODFTE, V_IPCTMVT.CODART, V_IPCTMVT.IMPCTV,
V_IPCTMVT.PANART, V_IPCTMVT.PACTMVT, V_IPCTMVT.CMVMVT, V_IPCTMVT.CRFMVT,
V_IPCTMVT.PESMVT, V_IPCTMVT.SINMMVT, V_IPCTMVT.FCOMMVT, V_IPCTMVT.CANRMVT,
V_IPCTMVT.OBSMVT, V_IPCTMVT.DECMVT, V_IPCTMVT.CANTANT, V_IPCTMVT.ANO,
V_IPCTMVT.MES, V_IPCTMVT.PESART, V_IPCTMVT.VARIACION, V_IPCTMVT.MARCA)
|     |     |     |     +-------ENDSCAN
|     |     |     |           SELE V_HISTORIA
|     |     |     |           TABLEUPDATE(.T.)
|     |     |     |
UPDATE V_IPCTMVT SET MARCA = .T. WHERE MARCA = .F.
|     |     |     |           SELE V_IPCTMVT
|     |     |     |           TABLEUPDATE(.T.)
&&Luego limpiamos el contenido del archivo de observaciones para
&&empezar un mes nuevo, puesto  que las observaciones corresponden a cada mes
|     |     |     |           SELE V_IPCROBS
|     |     |     |           GO TOP
UPDATE V_IPCROBS SET MARCA = .T. WHERE MARCA = .F.
|     |     |     |           TABLEUPDATE(.T.)
|     |     |     |           USE
|     |     |     +-------ENDIF
&&Avanzamos en la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+100
|     |     |           PROGRESO.REFRESH
|     |     |           PROGRESS()
&&extractamos la informacion de fuente articulo de acuerdo a la mes y a la decada procesada
```

```
                    SELECT V_IPCRFAP.CODFTE, V_IPCRFAP.CODART,
          V_IPCRFAP.DECART,V_IPCRFAP.PANART,;
V_IPCRFAP.CANTANT, V_IPCRFAP.PESART, V_IPCRFAP.FCOMMVT;
|     |     |     FROM V_IPCRFAP, V_IPCMFTE, V_IPCMART;
WHERE V_IPCRFAP.MARCA = .F. AND V_IPCMART.CODART = V_IPCRFAP.CODART AND
_IPCRFAP.DECART = DECADA AND V_IPCRFAP.CODFTE = V_IPCMFTE.CODFTE AND
(V_IPCRFAP.PERFAP IN (SELECT V_IPCRPART.CODPART;
|     |     |     FROM V_IPCRPART;
|     |     |     WHERE V_IPCRPART.MES = NMES));
|     |     |     ORDER BY V_IPCRFAP.CODFTE, V_IPCRFAP.CODART;
|     |     |     INTO CURSOR CHE1
&&Avanzamos en la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+100
|     |     |           PROGRESO.REFRESH
|     |     |           PROGRESS()
|     |     |
|     |     |           SELECT CHE1
|     |     |           FUENTE=CHE1.CODFTE
&&Recorremos el archivo resultante de la consulta anterior para asignarle
&&el consecutivo de impresion
|     |     |     +-------SCAN
|     |     |     |           SCATTER MEMVAR
&&A la primera fuente le asignamos el consecutivo inicial, es decir, el siguiente
&&al consecutivo final del prediligenciamiento anterior
&&Empezamos a alimentar el archivo de movimiento para cargar la informacion en captura
|     |     |     |     +-------IF (CHE1.CODFTE) = FUENTE
INSERT INTO V_IPCTMVT(CODFTE, CODART, PANART, CANTANT, PESART, FCOMMVT,
IMPCTV, DECMVT, ANO, MES);
VALUES(CHE1.CODFTE, CHE1.CODART, CHE1.PANART, CHE1.CANTANT, CHE1.PESART,
CHE1.FCOMMVT, X, DECADA, NANO, NMES)
|     |     |     |     |      FUENTE = (CHE1.CODFTE)
|     |     |     |     +-------ELSE
&&Seguimos incrementando el consecutivo de impresion a medida que cambiamos
&&de codigo de fuente y seguimos alimentando el archivo de movimiento
|     |     |     |     |           X = X + 1
INSERT INTO V_IPCTMVT(CODFTE, CODART, PANART, CANTANT, PESART, FCOMMVT,
IMPCTV, DECMVT, ANO, MES);
VALUES(CHE1.CODFTE, CHE1.CODART, CHE1.PANART, CHE1.CANTANT, CHE1.PESART,
CHE1.FCOMMVT, X, DECADA, NANO, NMES)
|     |     |     |     |           FUENTE = (CHE1.CODFTE)
|     |     |     |     |           LOOP
|     |     |     |     +-------ENDIF
|     |     |     |
|     |     |     +-------ENDSCAN
|     |     |           SELE V_IPCTMVT
|     |     |           TABLEUPDATE(.T.)
&&Avanzamos en la barra de progreso
```

```
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+100
|     |     |           PROGRESO.REFRESH
|     |     |           PROGRESS()
&&Evaluamos el estado de el archivo de consecutivo(usado o no)
&&Si esta usado lo seleccionamos
|     |     |           SELECT V_IPCTCTV
&&Recorremos el archivo de consecutivos el registro que tenga esta de
&&Prediligenciamiento en falso, este seria periodo en proceso
|     |     |     +-------SCAN FOR V_IPCTCTV.ESTPRE=.F.
&&Actualizamos las decadas segun la actual
&&Si es decada 3 la actual, para el siguiente periodo sera decada 1
&&lo mismo incrementar el mes
|     |     |     |     +-------IF FECHADECADA = 3
|     |     |     |     |           SELECT V_IPCTMVT
|     |     |     |     |           FECHADECADA = 1
|     |     |     |     |           FECHAMES = FECHAMES + 1
&&Actualizamos las decadas segun la actual
&&actualizamos el mes, si es mes 12 actualizamos a 1
&&lo mismo incrementamos el año
|     |     |     |     |     +-------IF FECHAMES > 12
|     |     |     |     |     |           FECHAMES = 1
|     |     |     |     |     |           FECHAANO = FECHAANO + 1
|     |     |     |     |     +-------ENDIF
|     |     |     |     +-------ELSE
|     |     |     |     |           FECHADECADA = FECHADECADA + 1
|     |     |     |     |           FECHAMES = FECHAMES
|     |     |     |     +-------ENDIF
&&Insertamos la informacion ya actualizada en el archivo de consecutivos
INSERT INTO V_IPCTCTV(AÑOCTV,MESCTV,DECCTV,INICTV,FINCTV);
                                    VALUES(FECHAANO,FECHAMES,FECHADECADA,0,X)
|     |     |     +-------ENDSCAN
&&Actualizamos el estado de prediligenciamiento y demas parametros
|     |     |           SELECT V_IPCTCTV
|     |     |     +-------SCAN
|     |     |     |           REPLACE V_IPCTCTV.ESTPRE WITH .T.
|     |     |     +-------ENDSCAN
|     |     |           SELECT V_IPCTCTV
|     |     |           GO BOTTOM
|     |     |           REPLACE V_IPCTCTV.ESTPRE WITH .F.
|     |     |           THISFORM.CONFIRMAR.ENABLED=.F.
|     |     |           THISFORM.REPORTE.ENABLED=.T.
|     |     |           TABLEUPDATE(.T.)
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+100
|     |     |           PROGRESO.REFRESH
|     |     |           PROGRESS()
```

```
|       |       |
&&Completamos la informacion del prediligenciamiento con las especificaciones
&&En esta primera consulta estractamos la informacion que coincidan estando en |movimiento
&&(fuente articulo) y en el archivo de especificaciones y que ademaa tengan
&&especificaciones   los articulos
                     SELECT V_IPCTMVT.MARCA, V_IPCTMVT.CODFTE, V_IPCTMVT.CODART,
                     V_IPCTMVT.PANART,;
V_IPCTMVT.IMPCTV, V_IPCTMVT.DECMVT, V_IPCTMVT.FCOMMVT, V_IPCTMVT.ANO,
V_IPCTMVT.MES,;
V_IPCMFTE.NOMBFTE, V_IPCMFTE.DIREFTE, V_IPCMFTE.TELFTE,;
V_IPCMFTE.ZONA, V_IPCMART.NOMART, V_IPCMART.CANBART,;
V_IPCMART.UNIART, V_IPCRFAP.CANTANT, V_IPCRFAP.PESART,;
V_IPCRESP.CODESP, V_IPCRESP.NOMESP, V_IPCRESP.NOMBDET;
FROM V_IPCTMVT, V_IPCMFTE, V_IPCMART, V_IPCRFAP, V_IPCRESP;
WHERE (V_IPCTMVT.MARCA = .F.) AND (V_IPCTMVT.DECMVT = m.DECADA) AND
(V_IPCTMVT.CODFTE = V_IPCRESP.CODFTE) AND;
 (V_IPCTMVT.CODART = V_IPCRESP.CODART) AND (V_IPCMART.CODART =
V_IPCTMVT.CODART) AND;
 (V_IPCTMVT.CODFTE = V_IPCMFTE.CODFTE) AND (V_IPCRFAP.CODART =
V_IPCTMVT.CODART) AND;
|       |       | (V_IPCTMVT.CODFTE = V_IPCRFAP.CODFTE);
|       |       |INTO CURSOR ESTA
&&En esta segunda extraccion de informacion, traemos los articulos que no tienen
&&especificaciones, por que no todos los articulos tienen especificaciones
&&Luego unimos los dos resultados de estas        dos consultas
|       |       |             SELECT ESTA.* FROM ESTA;
UNION SELECT V_IPCTMVT.MARCA, V_IPCTMVT.CODFTE, V_IPCTMVT.CODART,
V_IPCTMVT.PANART, V_IPCTMVT.IMPCTV, V_IPCTMVT.DECMVT,;
V_IPCTMVT.FCOMMVT, V_IPCTMVT.ANO, V_IPCTMVT.MES, V_IPCMFTE.NOMBFTE,
V_IPCMFTE.DIREFTE, V_IPCMFTE.TELFTE,;
V_IPCMFTE.ZONA, V_IPCMART.NOMART, V_IPCMART.CANBART, V_IPCMART.UNIART,;
V_IPCRFAP.CANTANT, V_IPCRFAP.PESART,",",";
FROM V_IPCTMVT, V_IPCMFTE, V_IPCMART, V_IPCRFAP;
WHERE (V_IPCTMVT.MARCA = .F.) AND (V_IPCTMVT.DECMVT = m.DECADA) AND
(V_IPCMART.CODART = V_IPCTMVT.CODART) AND (V_IPCTMVT.CODFTE =
V_IPCMFTE.CODFTE)  AND;
 (V_IPCRFAP.CODART = V_IPCTMVT.CODART) AND (V_IPCTMVT.CODFTE =
V_IPCRFAP.CODFTE) AND;
(ALLTRIM(V_IPCTMVT.CODFTE)+ALLTRIM(V_IPCTMVT.CODART) NOT IN (SELECT DISTINCT
ALLTRIM(ESTA.CODFTE)+ALLTRIM(ESTA.CODART) FROM ESTA));
ORDER BY 13, 5, 3;
INTO CURSOR CHE2
&&Avanzamos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+100
|       |       |             PROGRESO.REFRESH
|       |       |             PROGRESS()
&&Anotacion: el historico de donde se reimprime o se consulta informacion de
```

```
&&periodos anteriores se compone de dos archivo: v_historia y v_esptemp, antes
&&ya vimos donde utilizamos v_historia, ahora vemos donde utilizamos v_esptemp
&&el cual es donde almacenamos las especificaciones de los articulos tal y como
&&las tenian en su momento dependiendo de la decada y el mes y el año
&&en que se procesaron
&&Aqui alimentamos el archivo de historico de especificaciones
|     |     |           SELE V_ESPTEMP
|     |     |           SELE CHE2
|     |     |     +-------SCAN FOR CHE2.MARCA = .F.
INSERT INTO V_ESPTEMP(CODFTE,CODART,IMPCTV,CODESP,NOMESP,NOMBDET, ANO,
MES, DECMVT);
VALUES(CHE2.CODFTE,CHE2.CODART,CHE2.IMPCTV,CHE2.CODESP,CHE2.NOMESP,CHE2.N
OMBDET, NANO, NMES, DECADA)
|     |     |     +-------ENDSCAN
|     |     |           SELE V_ESPTEMP
|     |     |           TABLEUPDATE(.T.)
&&Avanzamos la barra de proceso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+200
|     |     |           PROGRESO.REFRESH
|     |     |           PROGRESS()
|     |     |
|     |     |           SELEC V_IPCTCTV
|     |     |           REPLACE ESTTRAS WITH .T.
|     |     |           THISFORM.TEXT6.READONLY=.T.
|     |     |           TABLEUPDATE(.T.)
|     |     |           PROGRESO.BARRAPROGRESO1.LLENE
|     |     |           PROGRESO.RELEASE
|     |     |
&&Validamos el contenido de la variable que nos informa la version
&&de sistema operativo estamos trabajando
&&Dependiendo del mismo llamamos el reporte correspondiente
&&por que se comprobo que de una version a otra se desplazaba
&&la informacion en la impresion
|     |     |           SELE CHE2
|     |     |     +-------IF SISTOPE="NT"
REPORT FORM "nuevo_predi" NOCONSOLE PREVIEW
|     |     |     +-------ELSE
REPORT FORM "prediligenciamiento" NOCONSOLE PREVIEW
|     |     |     +-------ENDIF
|     |     +-------CASE MENSAJE3=7
MENSAJE4=MESSAGEBOX( " Selección cancelada por el usuario !!!!!",0+48+0," INFORMACION ")
|     |     +-------ENDCASE
|     +-------ENDCASE
|
+-------ENDPROC
```

5.19 FORMULARIO PARA EL REPORTE FINAL (CIUDAD.SCX)

5.19.1 Procedimiento INIT

```
+-------PROCEDURE INIT
THISFORM.CHECK05.VALUE = 0
THISFORM.CHECK08.VALUE = 0
THISFORM.CHECK11.VALUE = 0
THISFORM.CHECK13.VALUE = 0
THISFORM.CHECK17.VALUE = 0
THISFORM.CHECK23.VALUE = 0
THISFORM.CHECK41.VALUE = 0
THISFORM.CHECK50.VALUE = 0
THISFORM.CHECK52.VALUE = 0
THISFORM.CHECK54.VALUE = 0
THISFORM.CHECK66.VALUE = 0
THISFORM.CHECK68.VALUE = 0
THISFORM.CHECK76.VALUE = 0
THISFORM.CHECK15.VALUE = 0
THISFORM.CHECK18.VALUE = 0
THISFORM.CHECK19.VALUE = 0
THISFORM.CHECK20.VALUE = 0
THISFORM.CHECK27.VALUE = 0
THISFORM.CHECK44.VALUE = 0
THISFORM.CHECK47.VALUE = 0
THISFORM.CHECK63.VALUE = 0
THISFORM.CHECK70.VALUE = 0
THISFORM.CHECK73.VALUE = 0
THISFORM.CHECK88.VALUE = 0
THISFORM.CHECK90.VALUE = 0

SELECT CIUDAD_CHECK
GO TOP
REPLACE ALL RUTA WITH ''
REPLACE ALL VALOR_CHEC WITH ''
REPLACE ALL REPORTE WITH ''
*!*	REPLACE ALL DB WITH ''
INDEX ON CODDPTO TO M
GO TOP


DO WHILE !EOF()
	REPLACE RUTA WITH  tcFolder+CODDPTO+CODMPIO
 SKIP
ENDDO

SELECT CIUDAD_CHECK
GO TOP
```

```
I = 1
DO WHILE !EOF()
        IF DIRECTORY (ALLTRIM(RUTA))
                REPLACE VALOR_CHEC WITH
'THISFORM.CHECK'+SUBSTR(ALLTRIM(RUTA),LEN(ALLTRIM(RUTA))-4,2)+'.VALUE = 1'
                REPLACE REPORTE WITH '1'
                oFolder = 'THISFORM.CHECK'+SUBSTR(ALLTRIM(RUTA),LEN(ALLTRIM(RUTA))-
4,2)+'.FORECOLOR = 255'
                &oFolder
        ELSE
                REPLACE VALOR_CHEC WITH
'THISFORM.CHECK'+SUBSTR(ALLTRIM(RUTA),LEN(ALLTRIM(RUTA))-4,2)+'.VALUE = 0'
                REPLACE REPORTE WITH '0'
                oFolder = 'THISFORM.CHECK'+SUBSTR(ALLTRIM(RUTA),LEN(ALLTRIM(RUTA))-
4,2)+'.ENABLED = .F.'
                &oFolder
        ENDIF
 oFolder = ALLTRIM(VALOR_CHEC)
 &oFolder
 SKIP
 I = I + 1
ENDDO
SELECT CIUDAD_CHECK
GO TOP
```

5.19.2 Boton Command2 (Consolidado)

```
|       do reporteconsolidado
```

5.19.3 Boton Command4 (Precios consolidados)

```
|       do precisspconsolidados
```

5.19.4 Boton Salir

```
|       THISFORM.Release
```

5.20 FORMULARIO PARA RECUPERAR COPIA DE SEGURIDAD (reccopiaseg.SCX)

5.20.1 Vistas

v_registro

```
+-------PROCEDURE ACTIVATE
&&Al entrar al formulario actualizamos el archivo de registro para saber
&&que este usuario abrio este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
```

```
        requery('v_registro')
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para saber
&&si ya esta relacionado en el archivo y no voilver a registrarlo
                        LOCATE FOR (ALLTRIM(V_REGISTRO.QUIEN) + NPANTALLA =
                        ALLTRIM(PQUIEN) + WONTOP()) AND V_REGISTRO.ACT_INA = .T.
&&si no es encontrado lo registramos y actualizamos el archivo
|     +-------IF !FOUND()
                        INSERT INTO V_REGISTRO(QUIEN, NPANTALLA, FECHAR, ACT_INA);
|     |       VALUES(PQUIEN,WONTOP( ), DATE(), .T.)
|     |       TABLEUPDATE(.T.)
|     +-------ENDIF
|
+-------ENDPROC

+-------PROCEDURE INIT
&&Declaramos variable para almacenar mas adelante la unidad de donde se recuperara
&&la Copia de Seguridad
|           PUBLIC UNIDAD
|           UNIDAD=''
&&Cinfguramos para el manejo de fechas
|           SET CENTURY ON
|           SET DATE DMY
|
|           THISFORM.COMMAND2.SETFOCUS
+-------ENDPROC

+-------PROCEDURE DESTROY
&&Al salir del formulario actualizamos el archivo de registro para saber
&&que este usuario ya salio de este formulario
&&Actualizamos la informacion para tener lo mas reciente de los usuarios
&&que hayan empezado a hacer uso de la informacion por medio del aplicativo
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Buscamos la coincidencia del usuario que abrio esta pantalla para indicar
&&que la cerro en este momento
                        LOCATE FOR
                        ALLTRIM(V_REGISTRO.QUIEN)+ALLTRIM(V_REGISTRO.NPANTALLA) =
                        ALLTRIM(PQUIEN)+WONTOP( )
|     +-------IF FOUND()
|     |           REPLACE V_REGISTRO.ACT_INA WITH .F.
&&Actualizamos la informacion y tambien el archivo
|     |           TABLEUPDATE(.T.)
|     +-------ENDIF
+-------ENDPROC
```

5.20.2 Boton Buscar ...

```
+-------PROCEDURE CLICK
&&Configuramos para el manejo de archivos
|           SET SAFETY ON
&&Declaramos variables para el manejo rutas y nombres de carpetas
|           PUBLIC ARCHIVO,DIRP
|           LOCAL GCFILE,LOGICA,VARIABLE
&&Tomamos del sistema la unidad desde donde se esta ejecutando
&&el aplicativo
|           DIRP=SYS(5)
&&Verificamos si la carpeta que se sugiere utilizar ya esta creada
|           LOGICA=DIRECTORY(DIRP+"\"+"IPCRBACK")
|     +-------IF LOGICA
&&Si no esta creada se crea automaticamente
|     +-------ELSE
|     |           MKDIR DIRP+"\"+"IPCRBACK"
|     +-------ENDIF
&&Ejecutamos la orden para que el usuario seleccione una carpeta o
&&una unidad de disquete y la almacenamos en una variable
|           GCFILE=GETFILE('ZIP', 'Abrir un .ZIP:', 'seleccionar',0)
|
|           POSICION=RAT("\", GCFILE)
|           UNIDAD=SUBSTR(GCFILE,1,POSICION)
&&Si se selecciono alguna ruta de donde tomar el archivo comprimido
&&se procede al siguiente paso
|     +-------IF NOT EMPTY(GCFILE)
|     |           ARCHIVO=ALLTRIM(SUBSTR(GCFILE,POSICION+1))
|     |           THISFORM.NOMBREARCHIVO.VALUE = UNIDAD+ARCHIVO
|     |           THISFORM.INICIAR.ENABLED=.T.
|     +-------ELSE
&&Si no seleccionaro archivo alguno
                        MENSAJE1=MESSAGEBOX(" No se encuentra el archivo .ZIP, incluyalo por
                        favor ",0+48+0," INFORMACION ")
|     |           THISFORM.INICIAR.ENABLED=.F.
|     +-------ENDIF
+-------ENDPROC
```

5.20.3 Boton Salir

```
+-------PROCEDURE CLICK
&&Salir de esta pantalla
|           THISFORM.RELEASE
|
+-------ENDPROC
```

5.20.4 Boton Iniciar

```
+-------PROCEDURE CLICK
&&Configuramos el ambiente de trabajo para el manejo de archivos
|           SET SAFETY OFF
|           SET TALK OFF
&&DEclaramos variable para almacenar la ruta de donde se va aejcutar el winzip
|           LOCAL RUTAZIP
&&Definimos la ruta del winzip dependiendo de lenguaje de la version de windows
&&si es en español o en ingles
|     +-------IF FILE('c:\archivos de programa\winzip\winzip32.exe')
|     |           RUTAZIP='c:\archivos de programa\winzip\winzip32.exe'
|     +-------ELSE
|     |           RUTAZIP='c:\program files\winzip\winzip32.exe'
|     +-------ENDIF
&&Definmos la manera en que se ejecutaria el winzip, para este caso configuramos para
|&&que lo haga minimizado utilizando las API de Windows
|           #DEFINE SW_SHOWMINIMIZED 2
&&Predeterminamos ruta donde se descomprimirian los archivos
|           RUTAPREDE=DIRP+"\"+"IPCRBACK"
|           NOMBRECOPIA =ARCHIVO
&&Verificamos la unidad seleccionada donde se encuentra el archivo a descomprimir
&&Si es en una unidad de disquete
|     +-------IF UNIDAD=UPPER('A:\') OR UNIDAD=UPPER('B:\')
                                MENSAJE= MESSAGEBOX("Ya insertó el disquete en la unidad
                                "+UNIDAD +" ?",4+64+256," Recuperar copia de seguridad ")
|     +-------ELSE
|     |           MENSAJE=6
|     +-------ENDIF
|           *****
&&En caso afrimativo
|     +-------DO CASE
|     +-------CASE MENSAJE = 6
&&Alamacenamoa la ruta      actual para al terminar este procesa volver a ella
|     |           ACTUAL=FULLPATH(SYS(2003))
&&Ejecutamos la barra de progreso
|     |           DO FORM FPROGRESO NAME PROGRESO LINKED
                                PROGRESO.LPROGRESO.CAPTION='Recuperando Copia de
                                Seguridad en: '+" "+"IPCRBACK"+" "+' Espere por favor.... '
|     |           PROGRESO.BARRAPROGRESO1.MAXVALOR=2500
|     |           PROGRESO.BARRAPROGRESO1.MINVALOR=1
&&Avanzanmos la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+500
|     |           PROGRESO.REFRESH
|     |           PROGRESS()
|     |
&&Segun la unidad seleccionada de donde se encuentra el archivo a descomprimir
&&Definimos el comando
&&Si es en una unidad de disquete
```

```
|     |     +-------IF UNIDAD=UPPER('A:\') OR UNIDAD=UPPER('B:\')
                                    COMANDO=RUTAZIP+" -min -e "+UNIDAD+NOMBRECOPIA+"
                                    "+RUTAPREDE
|     |     +-------ELSE
&&Si es en una unida de Disco Duro
                                    COMANDO=RUTAZIP+" -min -e "+UNIDAD+NOMBRECOPIA+"
                                    "+RUTAPREDE
|     |     +-------ENDIF
&&Primero limpiamos el contenido de la carpeta predeterminada
|     |           DELETE FILE RUTAPREDE+"\"+"*.*"
&&Avanzamoa la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+500
|     |           PROGRESO.REFRESH
|     |           PROGRESS()
&&Avanzamoa la barra de progreso
        PROGRESO.BARRAPROGRESO1.PROGRESO=PROGRESO.BARRAPROGRESO1.PROG
        RESO+500
|     |           PROGRESO.REFRESH
|     |           PROGRESS()
&&Ejecutamos la orden con las API de Windows
|     |           =WINEXEC('&comando', SW_SHOWMINIMIZED)
&&Avanzamoa la barra de progreso
|     |           PROGRESO.REFRESH
&&Avanzamoa la barra de progreso
|     |           PROGRESO.BARRAPROGRESO1.LLENE
|     |           PROGRESO.RELEASE
&&Retornamos a la ruta inicial
|     |           SET DEFAULT TO &ACTUAL
                        MENSAJE7=MESSAGEBOX(" Recuperacion de Copia de Seguridad terminada
                        con éxito.",0+64+0," INFORMACION ")
|     |
|     +-------CASE MENSAJE = 7
                              MENSAJE=MESSAGEBOX(" Cancelada la Recuperacion de Copia de
                              Seguridad ! ",0+64+0," Información ")
|     +-------ENDCASE
|
+-------ENDPROC
```

5.21 FORMULARIO PARA INGRESAR AL APLICATIVO (SEGURO.SCX)

5.21.1 Vistas

v_auxiliar
v_registro

```
+-------PROCEDURE INIT
&&Ocultamos el menu del aplicativos
|           SET SYSMENU OFF
&&configuramos para el manejo de fechas
|           SET CENTURY ON
|           SET DATE DMY
&&Declaramos variables para almacenar la informacion del usuario
&&que esta ingresando para trabajar el aplicativo y el numero de intentos fallidos
&&que realiza para sacarla completamente
|           PUBLIC PQUIEN, PCONQUE, ENTRADAS
&&Inicializamos las variables
|           PQUIEN=" "
|           THISFORM.TEXT1.REFRESH
|           PCONQUE=" "
|           THISFORM.TEXT2.REFRESH
|           ENTRADAS=0
|
|           THISFORM.TEXT1.SETFOCUS
+-------ENDPROC
```

5.21.2 Boton Salir

```
+-------PROCEDURE CLICK
|         MENSAJE=MESSAGEBOX(" Esta seguro ? ",4+32+256," Acceso a NUEVO
                IPC ")
|     +-------DO CASE
|     +-------CASE MENSAJE=6
|     |           PNIVEL=" "
|     |           THISFORM.RELEASE
|     +-------ENDCASE
|
+-------ENDPROC
```

5.21.3 Clave de acceso(Text2)

```
+-------PROCEDURE LOSTFOCUS
&&Validamos que hayan ingresado informacion necesaria(Nombre y Clave)
|     +-------IF !EMPTY(THISFORM.TEXT1.VALUE) AND !EMPTY(THIS.VALUE)
|     |           THISFORM.ACEPTAR.ENABLED=.T.
|     |           THISFORM.ACEPTAR.SETFOCUS
|     +-------ELSE
|     |           THISFORM.ACEPTAR.ENABLED=.F.
|     |
|     +-------ENDIF
+-------ENDPROC
```

5.21.4 Boton aceptar

```
+-------PROCEDURE CLICK
&&Configuramos para el manejo de archivos
|           SET SAFETY OFF
&&LLevamos el control de las veces que intenta ingresar el usuario
|           ENTRADAS=ENTRADAS+1
|           PQUIEN = THISFORM.TEXT1.VALUE
|           PCONQUE = THISFORM.TEXT2.VALUE
&&Obtenemos la informacion mas actualizada del archivo de registro, par verificar
&&que algun otro usuario no haya ingresado ya con el nombre y contraseña que este
&&usuario esta utilizando
|           REQUERY('v_registro')
|           SELE V_REGISTRO
&&Buscamos para en el archivo para verificar que ademas este activo el usuario
|           LOCATE FOR ALLTRIM(V_REGISTRO.QUIEN) = ALLTRIM(PQUIEN) AND
                        V_REGISTRO.ACT_INA = .T.
&&Si es encontrada informacion en el archivo lo informamos y no se permite ingresar
&&a este ultimo
|     +-------IF FOUND()
|     |           =MESSAGEBOX('Otro usuario ingreso con este nombre, por
                              favor intente con otro nombre de Usuario...',0+64+0,'Informacion')
&&Si al tercer intento de ingresar al aplicativo por cualquier motivo no lo logro
&&lo sacamos
|     |     +-------IF ENTRADAS >= 3
|     |     |           MENSAJE=MESSAGEBOX(" Lo siento el sistema lo ha
                                  rechazado ",0+64+0," Acceso a NUEVO IPC ")
|     |     |           PNIVEL=" "
|     |     |           THISFORM.RELEASE
|     |     +-------ENDIF
|     +-------ELSE
&&Si no se encontro usuario alguno trabajando ya con este usuario en el aplicativo
&&procedemos a comprimir la informacion suministrada por el usuario entrante para
&&realizar la comparacion
&&Encriptamos la clave de acceso
|     |     +-------FOR I=1 TO 8
|     |     |           LETRA=SUBSTR(PCONQUE,I,1)
|     |     |           NLETRA=CHR(ASC(LETRA)+10)
|     |     |           PCONQUE=STUFF(PCONQUE,I,1,NLETRA)
|     |     +-------ENDFOR
&&Encriptamos el nombre de usuario
|     |     +-------FOR I=1 TO 20
|     |     |           LETRA=SUBSTR(PQUIEN,I,1)
|     |     |           NLETRA=CHR(ASC(LETRA)+10)
|     |     |           PQUIEN=STUFF(PQUIEN,I,1,NLETRA)
|     |     +-------ENDFOR
&&Seleccionamos el archivo y realizamos la busqueda del usuario entrante
|     |           SELE V_AUXILIAR
|     |           LOCATE FOR V_AUXILIAR.QUIEN = PQUIEN AND
                              V_AUXILIAR.CONQUE = PCONQUE
```

```
&&En caso de no encontrarse se le informa y se anota para llevar el control
&&de los intentos hechos
|     |     +-------IF !FOUND()
|     |     |           MENSAJE=MESSAGEBOX(" Usted no esta autorizado
                                        para trabajar en esta aplicación ",0+64+0," Acceso a
                                        NUEVO IPC ")
|     |     |           THISFORM.TEXT1.VALUE = " "
|     |     |           THISFORM.TEXT2.VALUE = " "
|     |     |
|     |     |           THISFORM.TEXT1.SETFOCUS
&&Si acumula tres intento fallidos lo sacamos
|     |     |     +-------IF ENTRADAS >= 3
|     |     |     |           MENSAJE=MESSAGEBOX(" Lo siento el
                                        sistema lo ha rechazado ",0+64+0," Acceso a
                                        NUEVO IPC ")
|     |     |     |           PNIVEL=" "
|     |     |     |           THISFORM.RELEASE
|     |     |     |
|     |     |     +-------ENDIF
&&Verificamos que esta activo el usuario registrado
&&En caso afirmativo desencripatamos la informacion correspondiente a el nivel
&&de acceso que tiene asignado para tenerlo en cuenta en las opciones a las que
&&tendra acceso y a los proceso que podra ejecutar
|     |     +-------ELSE
|     |     |     +-------IF V_AUXILIAR.ACTIVO = .T.
|     |     |     |           PQUIEN = V_AUXILIAR.QUIEN
|     |     |     |           PQUE = V_AUXILIAR.QUE
|     |     |     |     +-------FOR I=1 TO 1
|     |     |     |     |           LETRA=SUBSTR(PQUE,I,1)
|     |     |     |     |           NLETRA=CHR(ASC(LETRA)-10)
|     |     |     |     |           PQUE=STUFF(PQUE,I,1,NLETRA)
|     |     |     |     +-------ENDFOR
&&Almacenamos el nivel de acceso en una variable
|     |     |     |           PNIVEL = PQUE
&&Desencriptamos el nombre de usuario y lo almacenamos en una variable
|     |     |     |     +-------FOR I=1 TO 20
|     |     |     |     |           LETRA=SUBSTR(PQUIEN,I,1)
|     |     |     |     |           NLETRA=CHR(ASC(LETRA)-10)
|     |     |     |     |           PQUIEN=STUFF(PQUIEN,I,1,NLETRA)
|     |     |     |     +-------ENDFOR
|     |     |     |
|     |     |     |           PQUIEN = PQUIEN
&&Activamos el menu del aplicativo
|     |     |     |           SET SYSMENU ON
|     |     |     |           THISFORM.RELEASE
|     |     |     |           WAIT WIND 'es '+PQUIEN NOWAIT
|     |     |     +-------ELSE
&&Informamos que no se encuntra registrada la informacion suministrada
```

```
|     |     |     |          = MESSAGEBOX('El usuario correspondiente
                                a la informacion ingresada se encuentra
                                inactivo'+CHR(13)+'para trabajar en la aplicacion
                                ...',0+64+0,'Informacion')
|     |     |     |          THISFORM.TEXT1.VALUE = " "
|     |     |     |          THISFORM.TEXT2.VALUE = " "
|     |     |     |
|     |     |     |          THISFORM.TEXT1.SETFOCUS
|     |     |     +-------ENDIF
|     |     +-------ENDIF
|     +-------ENDIF
|
|
+-------ENDPROC
```

```
-----ENDPROC
```

5.21.4.1.1 PAGEFRAME1.PAGE5.text20

```
      NAME = "Text20"
+-------PROCEDURE INIT
|           THIS.VALUE = 0
+-------ENDPROC
```

5.21.4.1.2 PAGEFRAME1.PAGE5.text21

```
      NAME = "Text21"
+-------PROCEDURE INIT
|           THIS.VALUE = 0
+-------ENDPROC
```

5.21.4.1.3 PAGEFRAME1.PAGE5.text23

```
      NAME = "Text23"
+-------PROCEDURE INIT
|           THIS.VALUE = 0
+-------ENDPROC
```

5.21.4.1.4 PAGEFRAME1.PAGE5.text22

```
      NAME = "Text22"
+-------PROCEDURE INIT
|           THIS.VALUE = 0
+-------ENDPROC
```

5.21.4.1.5 PAGEFRAME1.PAGE5.Navigator2

```
      skiptable = v_ftescalicap2
      NAME = "Navigator2"
      cmdTop.NAME = "cmdTop"
      cmdPrior.NAME = "cmdPrior"
      cmdNext.NAME = "cmdNext"
      cmdBottom.NAME = "cmdBottom"
      Datachecker1.NAME = "Datachecker1"
```

```
+-------PROCEDURE recordpointermoved
&& Definimos y cargamos variable con el nombre del archivo
|       LOCAL calias
|       calias=ALIAS()
|
|       THISFORM.LOCKSCREEN = .T.
|
|       THISFORM.Pageframe1.page6.ecaliobs.VALUE = ''
|       THISFORM.Pageframe1.page6.ECodFte.VALUE = v_ftescalicap2.codfte
|       THISFORM.Pageframe1.page6.cnombrefuente.VALUE = v_ftescalicap2.NombFte
|       THISFORM.Pageframe1.page6.ETelefono.VALUE = v_ftescalicap2.telfte
|       THISFORM.Pageframe1.page6.EDireccion.VALUE = v_ftescalicap2.direfte
|       pcalictvo = v_ftescalicap2.IMPctv
|       vcalifte6 = THISFORM.Pageframe1.page6.ECodFte.VALUE
&& Ejecutamos el evento para que actualice la informacion de las listas en pantalla
|       THISFORM.articalicaptura
|
|       THISFORM.LOCKSCREEN = .F.
|
|       SELECT(calias)
+-------ENDPROC
```

5.21.4.1.6 PAGEFRAME1.PAGE5.ecodfte

```
    NAME = "ECodFte"
+-------PROCEDURE LOSTFOCUS
|   +-------IF !EMPTY(THIS.VALUE)
&& Definimos y cargamos variable con el nombre del archivo
|   |       LOCAL calias
|   |       calias=ALIAS()
|   |
|   |       THISFORM.LOCKSCREEN = .T.
|   |
|   |       SELECT v_ftescalicap2
&& Almacenamos el numero del registro
&& actual en una variable para podernos devolver a el en caso de que la busqueda
&& no obtenga resultados
|   |       MarcaRegistro = RECNO()
|   |       SELECT v_ftescalicap2
|   |       LOCATE FOR v_ftescalicap2.codfte = THIS.VALUE
|   |   +-------IF !FOUND()
|   |   |       GO MarcaRegistro
|   |   |       THIS.VALUE = v_ftescalicap2.codfte
|   |   |       pcalictvo = v_ftescalicap2.IMPctv
|   |   +-------ELSE
&& nos devolvemos al registro que estabamaos inicialmente
|   |   |       vcalifte6 = THIS.VALUE
|   |   |       pcalictvo = v_ftescalicap2.IMPctv
&& cargamos la variable con el codigo de fuente y ejecutamos el metodo que actualiza
&& las listas en pantalla segun los articulos que tiene asignados la fuente y
&& los que no
|   |   |       THISFORM.articalicaptura
|   |   +-------ENDIF
```

```
&& Definimos y cargamos variable con el nombre del archivo
|       |
|       |               THISFORM.LOCKSCREEN = .F.
|       |
|       |               SELE(calias)
|       +-------ENDIF
+-------ENDPROC
```

5.21.4.1.7 PAGEFRAME1.PAGE5.Grid2

```
RECORDSOURCE = "v_zonacalicap2"
NAME = "Grid2"

Column1.CONTROLSOURCE = "v_zonacalicap2.codart"
Column1.NAME = "Column3"
Column2.CONTROLSOURCE = "v_zonacalicap2.pesart"
Column2.NAME = "Column4"
Column3.CONTROLSOURCE = "v_zonacalicap2.cantant"
Column3.NAME = "Column5"
column4.CONTROLSOURCE = "v_zonacalicap2.panart"
column4.NAME = "Column6"
column5.CONTROLSOURCE = "v_zonacalicap2.cmvmvt"
column5.NAME = "Column7"
column6.CONTROLSOURCE = "v_zonacalicap2.crfmvt"
column6.NAME = "Column8"
column7.CONTROLSOURCE = "v_zonacalicap2.pesmvt"
column7.NAME = "Column9"
column8.CONTROLSOURCE = "v_zonacalicap2.sinmmvt"
column8.NAME = "Column10"
column9.CONTROLSOURCE = "v_zonacalicap2.fcommvt"
column9.NAME = "Column11"
column10.CONTROLSOURCE = "v_zonacalicap2.canrmvt"
column10.NAME = "Column13"
column11.CONTROLSOURCE = "v_zonacalicap2.pactmvt"
column11.NAME = "Column14"
column12.CONTROLSOURCE = "v_zonacalicap2.obsmvt"
column12.NAME = "Column1"
column13.CONTROLSOURCE = "v_zonacalicap2.variacion"
column13.NAME = "Column2"
column14.CONTROLSOURCE = "v_zonacalicap2.fecharec"
column14.NAME = "Column12"
Column15.CONTROLSOURCE = "v_zonacalicap2.nomart"
Column15.NAME = "Column15"
```

Column15

```
CAPTION = "Artículo"
```

Column4

```
CAPTION = "P.E."
check1.CONTROLSOURCE = "v_zonacalicap2.pesart"
check1.NAME = "Check1"
+-------PROCEDURE check1.INIT
|
|
```

+-------ENDPROC

Column5
CAPTION = "Canti."
CONTROLSOURCE = "v_zonacalicap2.cantant"

Column6
CAPTION = "Precio"
CONTROLSOURCE = "v_zonacalicap2.panart"

Column7

CAPTION = "C.M.V."
check1.CONTROLSOURCE = "v_zonacalicap2.cmvmvt"

Column8
CAPTION = "C.R."
check1.CONTROLSOURCE = "v_zonacalicap2.crfmvt"
check1.NAME = "Check1"

Column9
CAPTION = "P.E."
check1.CONTROLSOURCE = "v_zonacalicap2.pesmvt"
check1.NAME = "Check1"

Column10
CAPTION = "S.I."
check1.CONTROLSOURCE = "v_zonacalicap2.sinmmvt"
check1.NAME = "Check1"

Column11
CAPTION = "F.C."
CONTROLSOURCE = "v_zonacalicap2.fcommvt"

Column13
CAPTION = "Canti."
CONTROLSOURCE = "v_zonacalicap2.canrmvt"
NAME = "Text1"

Column14
CAPTION = "Precio"
CONTROLSOURCE = "v_zonacalicap2.pactmvt"
NAME = "Text1"

Column1
CAPTION = "Obs."
CONTROLSOURCE = "v_zonacalicap2.obsmvt"
NAME = "Text1"
+-------PROCEDURE CLICK
|
+-------ENDPROC
+-------PROCEDURE GOTFOCUS
| +-------IF !EMPTY(THIS.VALUE)

```
|       |               pcalicodobs = PADL(ALLTRIM(THIS.VALUE),2,'0')
|       |               vcalifte5 = THISFORM.Pageframe1.page6.ECodFte.VALUE
|       |               vcalifte6 = THISFORM.Pageframe1.page6.ECodFte.VALUE
|       |
|       |               REQUERY('v_caliobs')
|       |               SELE v_caliobs
|       |
|       |               THISFORM.Pageframe1.page6.ecaliobs.VALUE = v_caliobs.observa
|       +-------ELSE
|       |               THISFORM.Pageframe1.page6.ecaliobs.VALUE = ''
|       +-------ENDIF
+-------ENDPROC
```

Column2

```
        CAPTION = "Varia."
        CONTROLSOURCE = "v_zonacalicap2.variacion"
        NAME = "Text1"
```

Column12

```
        CAPTION = "Fecha - Hora"
        CONTROLSOURCE = "v_zonacalicap2.fecharec"
        NAME = "Text1"
```

Column3

```
        CAPTION = "Artículo"
        TEXT1.CONTROLSOURCE = "v_zonacalicap2.nomart"
        TEXT1.NAME = "Text1"
        Command1.NAME = "Command1"
+-------PROCEDURE TEXT1.DBLCLICK
&& Al hacer doble click ejecuta el evento click del boton
|               THIS.PARENT.Command1.CLICK
+-------ENDPROC
+-------PROCEDURE Command1.CLICK
|       +-------WITH THISFORM.Pageframe1.page6
&& Carga a la variable un valor que determina que se va hacer
&& al ejecutar la panta 'fespecifica', si solo consulta o Cambio de especificaciones
|       |               CR = .F.
&& Ejecuta el formulario para consultar especificaciones del articulo actual
|       |               DO FORM  "forms\FEspecifica.scx" WITH .ECodFte.VALUE, v_zonacalicap2.codart, ;
|       |                       v_zonacalicap2.nomart
|       +-------ENDWITH
+-------ENDPROC
```

5.21.5 PAGEFRAME1.PAGE7.pageframe1

```
        NAME = "Pageframe1"
        Page1.CAPTION = "Asignar para Recolección"
        Page1.NAME = "Page1"
        Page2.CAPTION = "Asignar para Supervisión"
        Page2.NAME = "Page2"
        Page3.CAPTION = "Control Calidad Critica"
        Page3.NAME = "Page3"
```

```
Page4.CAPTION = "Control Calidad Captura"
Page4.NAME = "Page4"
```

5.21.5.1 PAGEFRAME1.PAGE7.pageframe1.page1

```
+-------PROCEDURE Page1.ACTIVATE
|           THISFORM.LOCKSCREEN = .T.
|     +-------IF EMPTY(THISFORM.Pageframe1.page7.Pageframe1.Page1.Text3.VALUE)
&& Tomamos de la vista de zonas el nuemro de la zona
|     |           SELE v_zonas
|     |           GO TOP
|     |           THISFORM.Pageframe1.page7.Pageframe1.Page1.Text3.VALUE =
PADL(ALLTRIM(STR(v_zonas.zona,2,0)),2,'0')
&& Segun elparametro de Cargo asignado traemos los funcionarios correspondientes
|     |           pcodcargo = 'R'
|     |           REQUERY('v_ipcrfunc1')
&& Traemos la informacion de los funcionarios, codigo y nombre
|     |           SELE v_ipcrfunc1
|     |           LOCATE FOR v_ipcrfunc1.zonafunci = PADL(ALLTRIM(STR(v_zonas.zona,2,0)),2,'0')
|     |     +-------IF FOUND()
|     |     |           THISFORM.Pageframe1.page7.Pageframe1.Page1.TEXT1.VALUE = v_ipcrfunc1.codfunci
|     |     +-------ELSE
|     |     |           THISFORM.Pageframe1.page7.Pageframe1.Page1.TEXT1.VALUE = ''
|     |     +-------ENDIF
|     |
|     |           REQUERY('v_ipcfunci')
|     |
|     |           SELE v_ipcfunci
|     |           LOCATE FOR ALLTRIM(v_ipcfunci.codfunci) ==
ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page1.TEXT1.VALUE)
|     |     +-------IF FOUND()
|     |     |           THISFORM.Pageframe1.page7.Pageframe1.Page1.text9.VALUE = v_ipcfunci.nomfunci
|     |     +-------ELSE
|     |     |           THISFORM.Pageframe1.page7.Pageframe1.Page1.text9.VALUE = ''
|     |     +-------ENDIF
&& Ejecutamos los procedimientos
|     |           THISFORM.actualizalistas
|     |           THISFORM.conteo
|     +-------ENDIF
&& para traer las fuentes que corresponden a la zona seleccionada y el conteo de las mismas
|           THISFORM.LOCKSCREEN = .F.
+-------ENDPROC
```

5.21.5.2 PAGEFRAME1.PAGE7.pageframe1.page2

```
+-------PROCEDURE Page2.ACTIVATE
|           THISFORM.LOCKSCREEN = .T.
|     +-------IF EMPTY(THISFORM.Pageframe1.page7.Pageframe1.Page2.Text3.VALUE)
&& Recorremos las zonas, asignamos la seleccionada a la caja de texto
|     |           SELE v_zonas2
|     |           GO TOP
|     |           THISFORM.Pageframe1.page7.Pageframe1.Page2.Text3.VALUE =
PADL(ALLTRIM(STR(v_zonas2.zona,2,0)),2,'0')
&& Limpiamos el origen de datos de la barra de navegar fuentes (<<, <, >,>>) mientras se trae la informacion
```

```
&& de la nueva fuente
|       |       THISFORM.Pageframe1.page7.Pageframe1.Page2.navigator2.skiptable = ''
&& Y lo mismo para el objeto que contiene el nombre de la fuente
|       |       THISFORM.Pageframe1.page7.Pageframe1.Page2.cnombrefuente.VALUE = ''
&& Cargamos variables para traer las fuentes de la zona seleccionada
|       |       pzonacri = VAL(THISFORM.Pageframe1.page7.Pageframe1.Page2.Text3.VALUE)
|       |       panoplacri = VAL(THISFORM.ano.VALUE)
|       |       pmesplacri = VAL(THISFORM.mes.VALUE)
|       |       pdecplacri = VAL(THISFORM.combo1.VALUE)
&& Codigo de fuentes de la zona
|       |       REQUERY('v_zonaplacri2')
|       |       SELE v_zonaplacri2
&& Nombres de fuentes de acuerdo al anterior resultado
|       |       REQUERY('v_ftesplacri2')
|       |       SELE v_ftesplacri2
&& Reasignamos el origen de datos a la barra navegadora para recorrer las fuentes que corresponden
&& ala zona seleccionada
|       |       THISFORM.Pageframe1.page7.Pageframe1.Page2.navigator2.skiptable = 'v_ftesplacri2'
|       |       GO TOP
|       |
|       |       THISFORM.Pageframe1.page7.Pageframe1.Page2.cnombrefuente.VALUE =
v_ftesplacri2.NombFte
|       |       THISFORM.Pageframe1.page7.Pageframe1.Page2.ECodFte.VALUE = v_ftesplacri2.codfte
&& Cargamos variable con el codigo de la fuente seleccionada para traer la informacion de esta
|       |       vcalifte = THISFORM.Pageframe1.page7.Pageframe1.Page2.ECodFte.VALUE
&& Ejecutamos el metodo que muestra la informacion en laslistas
|       |       THISFORM.actualicelistas
&& De acuerdo alcargo traemos la informacion de losfuncionarios
|       |       pcodcargo = 'S'
|       |       REQUERY('v_ipcrfunc1')
|       |
|       |       SELE v_ipcrfunc1
|       |       LOCATE FOR v_ipcrfunc1.zonafunci =
PADL(ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page2.Text3.VALUE),2,'0')
|       |   +-------IF FOUND()
|       |   |       THISFORM.Pageframe1.page7.Pageframe1.Page2.TEXT1.VALUE = v_ipcrfunc1.codfunci
|       |   +-------ELSE
|       |   |       THISFORM.Pageframe1.page7.Pageframe1.Page2.TEXT1.VALUE = ''
|       |   +-------ENDIF
|       |
|       |       REQUERY('v_ipcfunci')
|       |
|       |       SELE v_ipcfunci
|       |       LOCATE FOR ALLTRIM(v_ipcfunci.codfunci) ==
ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page2.TEXT1.VALUE)
|       |   +-------IF FOUND()
|       |   |       THISFORM.Pageframe1.page7.Pageframe1.Page2.text9.VALUE = v_ipcfunci.nomfunci
|       |   +-------ELSE
|       |   |       THISFORM.Pageframe1.page7.Pageframe1.Page2.text9.VALUE = ''
|       |   +-------ENDIF
|     +-------ENDIF
|       THISFORM.LOCKSCREEN = .F.
+-------ENDPROC
```

5.21.5.3 PAGEFRAME1.PAGE7.pageframe1.page3

```
+-------PROCEDURE Page3.DEACTIVATE
|             SELE v_ftesplacri2
|             SET FILTER TO
+-------ENDPROC
+-------PROCEDURE Page3.ACTIVATE
|             THISFORM.LOCKSCREEN = .T.
|       +-------IF EMPTY(THISFORM.Pageframe1.page7.Pageframe1.Page3.Text3.VALUE)
|       |
|       |             SELE v_zonas3
|       |             GO TOP
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page3.Text3.VALUE =
PADL(ALLTRIM(STR(v_zonas3.zona,2,0)),2,'0')
|       |
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page3.navigator2.skiptable = ''
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page3.cnombrefuente.VALUE = ''
|       |
|       |             pzonacri = VAL(THISFORM.Pageframe1.page7.Pageframe1.Page3.Text3.VALUE)
|       |             panoplacri = VAL(THISFORM.ano.VALUE)
|       |             pmesplacri = VAL(THISFORM.mes.VALUE)
|       |             pdecplacri = VAL(THISFORM.combo1.VALUE)
|       |
|       |             REQUERY('v_zonaplacri2')
|       |             SELE v_zonaplacri2
|       |
|       |             REQUERY('v_ftesplacri2')
|       |             SELE v_ftesplacri2
|       |
|       |       +-------IF THISFORM.Pageframe1.page7.Pageframe1.Page3.filtrar.VALUE = .T.
|       |       |             SELE FLOOR(AVG(cuentaarti)) AS vpromedio FROM v_ftesplacri2 INTO CURSO
cpromedio
|       |       |             SELE cpromedio
|       |       |
|       |       |             SELE v_ftesplacri2
|       |       |             SET FILTER TO v_ftesplacri2.cuentaarti >= cpromedio.vpromedio
|       |       +-------ELSE
|       |       |             SELE v_ftesplacri2
|       |       |             SET FILTER TO
|       |       +-------ENDIF
|       |
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page3.navigator2.skiptable = 'v_ftesplacri2'
|       |             GO TOP
|       |
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page3.cnombrefuente.VALUE =
v_ftesplacri2.NombFte
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page3.ECodFte.VALUE = v_ftesplacri2.codfte
|       |             pobsctvimp = v_ftesplacri2.IMPctv
|       |             pobscodfte = v_ftesplacri2.codfte
|       |             vcalifte1 = THISFORM.Pageframe1.page7.Pageframe1.Page3.ECodFte.VALUE
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  ''
|       |
|       |             THISFORM.listascalicritica
```

```
|       |
|       |
|       |
|       |               pcodcargo = 'C'
|       |               REQUERY('v_ipcrfunc1')
|       |
|       |               SELE v_ipcrfunc1
|       |               LOCATE FOR v_ipcrfunc1.zonafunci =
PADL(ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page3.Text3.VALUE),2,'0')
|       |       +-------IF FOUND()
|       |       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.TEXT1.VALUE = v_ipcrfunc1.codfunci
|       |       +-------ELSE
|       |       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.TEXT1.VALUE = ''
|       |       +-------ENDIF
|       |
|       |               REQUERY('v_ipcfunci')
|       |
|       |               SELE v_ipcfunci
|       |               LOCATE FOR ALLTRIM(v_ipcfunci.codfunci) ==
ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page3.TEXT1.VALUE)
|       |       +-------IF FOUND()
|       |       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.text9.VALUE = v_ipcfunci.nomfunci
|       |       +-------ELSE
|       |       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.text9.VALUE = ''
|       |       +-------ENDIF
|       |
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  ''
|       +-------ENDIF
|               THISFORM.LOCKSCREEN = .F.
+-------ENDPROC
```

5.21.5.4 PAGEFRAME1.PAGE7.pageframe1.page4

```
+-------PROCEDURE Page4.DEACTIVATE
|               SELE v_ftesplacap2
|               SET FILTER TO
+-------ENDPROC
+-------PROCEDURE Page4.ACTIVATE
|               THISFORM.LOCKSCREEN = .T.
|       +-------IF EMPTY(THISFORM.Pageframe1.page7.Pageframe1.Page4.Text3.VALUE)
|       |               SELE v_zonas4
|       |               GO TOP
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page4.Text3.VALUE =
PADL(ALLTRIM(STR(v_zonas4.zona,2,0)),2,'0')
|       |
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page4.navigator2.skiptable = ''
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page4.cnombrefuente.VALUE = ''
|       |
|       |               pzonacap = VAL(THISFORM.Pageframe1.page7.Pageframe1.Page4.Text3.VALUE)
|       |               panocap = VAL(THISFORM.ano.VALUE)
|       |               pmescap = VAL(THISFORM.mes.VALUE)
|       |               pdeccap = VAL(THISFORM.combo1.VALUE)
|       |
|       |               REQUERY('v_zonaplacap2')
```

```
|       |           SELE v_zonaplacap2
|       |
|       |           REQUERY('v_ftesplacap2')
|       |           SELE v_ftesplacap2
|       |
|       |     +-------IF THISFORM.Pageframe1.page7.Pageframe1.Page4.filtrar.VALUE = .T.
|       |     |           SELE FLOOR(AVG(cuentaarti)) AS vpromedio FROM v_ftesplacap2 INTO CURSOR
cpromedio
|       |     |           SELE cpromedio
|       |     |
|       |     |           SELE v_ftesplacap2
|       |     |           SET FILTER TO v_ftesplacap2.cuentaarti >= cpromedio.vpromedio
|       |     +-------ELSE
|       |     |           SELE v_ftesplacap2
|       |     |           SET FILTER TO
|       |     +-------ENDIF
|       |
|       |           THISFORM.Pageframe1.page7.Pageframe1.Page4.navigator2.skiptable = 'v_ftesplacap2'
|       |           GO TOP
|       |
|       |           THISFORM.Pageframe1.page7.Pageframe1.Page4.cnombrefuente.VALUE =
v_ftesplacap2.NombFte
|       |           THISFORM.Pageframe1.page7.Pageframe1.Page4.ECodFte.VALUE = v_ftesplacap2.codfte
|       |           pcalictvo = v_ftesplacap2.IMPctv
|       |           vcalifte2 = THISFORM.Pageframe1.page7.Pageframe1.Page4.ECodFte.VALUE
|       |
|       |           THISFORM.listascalicaptura
|       |
|       |           pcodcargo = 'T'
|       |           REQUERY('v_ipcrfunc1')
|       |
|       |           SELE v_ipcrfunc1
|       |           LOCATE FOR v_ipcrfunc1.zonafunci =
PADL(ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page4.Text3.VALUE),2,'0')
|       |     +-------IF FOUND()
|       |     |           THISFORM.Pageframe1.page7.Pageframe1.Page4.TEXT1.VALUE = v_ipcrfunc1.codfunci
|       |     +-------ELSE
|       |     |           THISFORM.Pageframe1.page7.Pageframe1.Page4.TEXT1.VALUE = ''
|       |     +-------ENDIF
|       |
|       |           REQUERY('v_ipcfunci')
|       |           SELE v_ipcfunci
|       |
|       |           LOCATE FOR ALLTRIM(v_ipcfunci.codfunci) ==
ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page4.TEXT1.VALUE)
|       |     +-------IF FOUND()
|       |     |           THISFORM.Pageframe1.page7.Pageframe1.Page4.text9.VALUE = v_ipcfunci.nomfunci
|       |     +-------ELSE
|       |     |           THISFORM.Pageframe1.page7.Pageframe1.Page4.text9.VALUE = ''
|       |     +-------ENDIF
|       +-------ENDIF
|             THISFORM.LOCKSCREEN = .F.
|             THISFORM.Pageframe1.page7.Pageframe1.Page4.ECodFte.SETFOCUS
```

```
+-------ENDPROC
```

5.21.5.5 PAGEFRAME1.PAGE7.pageframe1.page1.cmdAdd

```
+-------PROCEDURE cmdAdd.CLICK
&& Validamos que haya seleccionado zona destino, es decir, a la que se le va  asignar la
&& fuente o las fuentes seleccionadas
|       +-------IF !EMPTY(THISFORM.Pageframe1.page7.Pageframe1.Page1.text4.VALUE) AND
!EMPTY(THISFORM.Pageframe1.page7.Pageframe1.Page1.Text3.VALUE)
&& Almacenamos el contenido de la lista de fuentes no asignados
&& Consultamos la vista de acuerdo a la zona origen para ubicar la fuente que se quiere reasignar
&& zona
|       |               pzonafunci1 = VAL(THISFORM.Pageframe1.page7.Pageframe1.Page1.Text3.VALUE)
|       |               panoh = VAL(THISFORM.ano.VALUE)
|       |               pmesh = VAL(THISFORM.mes.VALUE)
|       |               pdech = VAL(THISFORM.combo1.VALUE)
|       |
|       |               ListaSeleccionada = THISFORM.Pageframe1.page7.Pageframe1.Page1.Moverlista1.lstSource
|       |               ListaFuente = THISFORM.Pageframe1.page7.Pageframe1.Page1.Moverlista1.lstSelected
|       |               THISFORM.LOCKSCREEN=.T.
|       |       +-------IF cambio
&& Procedemos a cambiarl el numero de zona temporalmente para hacer la asignacion
&& al otro recolector
|       |       |               Cuenta = ListaSeleccionada.LISTCOUNT
|       |       |       +-------FOR i = 1 TO Cuenta
|       |       |       |       +-------IF ListaSeleccionada.SELECTED(i)
|       |       |       |       |               LaFuente = SUBSTR(ListaSeleccionada.LIST(i),1,20)
|       |       |       |       |               SELE v_zonaplare2
|       |       |       |       |               REPLACE v_zonaplare2.zona WITH
VAL(THISFORM.Pageframe1.page7.Pageframe1.Page1.text4.VALUE) FOR codfte = LaFuente
|       |       |       |       |               TABLEUPDATE(.T.)
|       |       |       |       +-------ENDIF
|       |       |       +-------ENDFOR
|       |       |               cambio=.F.
|       |       +-------ENDIF
&& Ejecutamos el procedimiento que hace la reasignacion de la zona a la o las fuentes seleccionadas
|       |               THISFORM.asignazona
|       |               THISFORM.LOCKSCREEN=.F.
&& Ejecutamos elprocedimiento que hace conteo del contenido de las listas
|       |               THISFORM.conteo
|       +-------ELSE
|       |               NODEFAULT
|       |               MESSAGEBOX('Debe especificar Zona de Destino para la fuente a trasladar',0+64,'Informacion')
|       +-------ENDIF
+-------ENDPROC
```

5.21.5.6 PAGEFRAME1.PAGE7.pageframe1.page1.Navigator1

```
skiptable = v_zonas
NAME = "Navigator1"
cmdTop.NAME = "cmdTop"
cmdPrior.NAME = "cmdPrior"
```

```
        cmdNext.NAME = "cmdNext"
        cmdBottom.NAME = "cmdBottom"
        Datachecker1.NAME = "Datachecker1"
+-------PROCEDURE recordpointermoved
|               LOCAL calias
|               calias=ALIAS()
&& Tomamos de la vista de zonas el nuemro de la zona
|               SELE v_zonas
|               THISFORM.Pageframe1.page7.Pageframe1.Page1.Text3.VALUE =
PADL(ALLTRIM(STR(v_zonas.zona,2,0)),2,'0')
&& Segun elparametro de Cargo asignado traemos los funcionarios correspondientes
|               pcodcargo = 'R'
|               REQUERY('v_ipcrfunc1')
&& Traemos la informacion de los funcionarios, codigo y nombre
|               SELE v_ipcrfunc1
|               LOCATE FOR v_ipcrfunc1.zonafunci = PADL(ALLTRIM(STR(v_zonas.zona,2,0)),2,'0')
|       +-------IF FOUND()
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page1.TEXT1.VALUE = v_ipcrfunc1.codfunci
|       +-------ELSE
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page1.TEXT1.VALUE = ''
|       +-------ENDIF
|
|               REQUERY('v_ipcfunci')
|
|               SELE v_ipcfunci
|               LOCATE FOR ALLTRIM(v_ipcfunci.codfunci) ==
ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page1.TEXT1.VALUE)
|       +-------IF FOUND()
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page1.text9.VALUE = v_ipcfunci.nomfunci
|       +-------ELSE
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page1.text9.VALUE = ''
|       +-------ENDIF
&& Ejecutamos los procedimientos
|               THISFORM.actualizalistas
|               THISFORM.conteo
&& para traer las fuentes que corresponden a la zona seleccionada y el conteo de las mismas
|               SELECT(calias)
+-------ENDPROC
```

5.21.5.7 PAGEFRAME1.PAGE7.pageframe1.page1.text3

```
        STATUSBARTEXT = "Escriba la Zona"
        NAME = "Text3"
+-------PROCEDURE VALID
|               LOCAL calias
|               calias=ALIAS()
&& Validamos que existan valores para año, mes y zona
|       +-------IF !EMPTY(THISFORM.ano.VALUE) AND !EMPTY(THISFORM.mes.VALUE) AND
!EMPTY(THIS.VALUE)
&& Dea cuerdo al cargo R = Recolector, traemos el funcionario correspondiente
|       |               pcodcargo = 'R'
|       |               REQUER('v_ipcrfunc1')
|       |
|       |               SELE v_ipcrfunc1
```

```
|       |               LOCATE FOR ALLTRIM(v_ipcrfunc1.zonafunci) = PADL(ALLTRIM(THIS.VALUE),2,'0')
|       |       +-------IF FOUND()
|       |       |               THIS.VALUE = PADL(ALLTRIM(THIS.VALUE),2,'0')
|       |       |               THISFORM.Pageframe1.page7.Pageframe1.Page1.TEXT1.VALUE = v_ipcrfunc1.codfunci
&& Traemos informacion (Codigo y Nombre del funcionario)
|       |       |               REQUERY('v_ipcfunci')
|       |       |
|       |       |               SELE v_ipcfunci
|       |       |               LOCATE FOR ALLTRIM(v_ipcfunci.codfunci) = ALLTRIM(v_ipcrfunc1.codfunci)
|       |       |       +-------IF FOUND()
|       |       |       |               THISFORM.Pageframe1.page7.Pageframe1.Page1.text9.VALUE =
v_ipcfunci.nomfunci
|       |       |       +-------ENDIF
&& Ejecutamos los metodos
|       |       |               THISFORM.actualizalistas
|       |       |               THISFORM.conteo
&& para actualizar las listas que contendrian las fuentes para esa zona y el conteo correspondiente
|       |       +-------ELSE
|       |       |               = MESSAGEBOX ('Zona NO Encontrada para este tipo de Funcionarios',0+64,'Informacion')
|       |       +-------ENDIF
|       +-------ENDIF
|
|               SELECT(calias)
+-------ENDPROC
```

5.21.5.8 PAGEFRAME1.PAGE7.pageframe1.page1.navigator2

```
        skiptable = v_zonas1
        NAME = "Navigator2"
        cmdTop.NAME = "cmdTop"
        cmdPrior.NAME = "cmdPrior"
        cmdNext.NAME = "cmdNext"
        cmdBottom.NAME = "cmdBottom"
        Datachecker1.NAME = "Datachecker1"

+-------PROCEDURE recordpointermoved
|               LOCAL calias
|               calias=ALIAS()
&& Tomamos de la vista de zonas el nuemro de la zona
|               SELE v_zonas1
|               THISFORM.Pageframe1.page7.Pageframe1.Page1.text4.VALUE =
PADL(ALLTRIM(STR(v_zonas1.zona,2,0)),2,'0')
&& Segun elparametro de Cargo asignado traemos los funcionarios correspondientes
|               pcodcargo = 'R'
|               REQUER('v_ipcrfunc1')
&& Traemos la informacion de los funcionarios, codigo y nombre
|               SELE v_ipcrfunc1
|               LOCATE FOR ALLTRIM(v_ipcrfunc1.zonafunci) = PADL(ALLTRIM(STR(v_zonas1.zona,2,0)),2,'0')
|       +-------IF FOUND()
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page1.text2.VALUE = v_ipcrfunc1.codfunci
|       +-------ELSE
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page1.text2.VALUE = ''
|       +-------ENDIF
|
|               REQUERY('v_ipcfunci')
```

```
|
|           SELE v_ipcfunci
|           LOCATE FOR ALLTRIM(v_ipcfunci.codfunci) ==
ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page1.text2.VALUE)
|     +-------IF FOUND()
|     |           THISFORM.Pageframe1.page7.Pageframe1.Page1.text10.VALUE = v_ipcfunci.nomfunci
|     +-------ELSE
|     |           THISFORM.Pageframe1.page7.Pageframe1.Page1.text10.VALUE = ''
|     +-------ENDIF
&& Ejecutamos los procedimientos
|           THISFORM.actualizalistas1
|           THISFORM.conteo
&& para traer las fuentes que corresponden a la zona seleccionada y el conteo de las mismas
|           SELECT(calias)
|
+-------ENDPROC
```

5.21.5.9 PAGEFRAME1.PAGE7.pageframe1.page1.text4

```
      STATUSBARTEXT = "Escriba la Zona"
      NAME = "Text4"
+-------PROCEDURE VALID
|           LOCAL calias
|           calias=ALIAS()
&& Validamos que existan valores para año, mes y zona
|     +-------IF !EMPTY(THISFORM.ano.VALUE) AND !EMPTY(THISFORM.mes.VALUE) AND
!EMPTY(THIS.VALUE)
&& Dea cuerdo al cargo R = Recolector, traemos el funcionario correspondiente
|     |           pcodcargo = 'R'
|     |           REQUER('v_ipcrfunc1')
|     |
|     |           SELE v_ipcrfunc1
|     |           LOCATE FOR ALLTRIM(v_ipcrfunc1.zonafunci) = PADL(ALLTRIM(THIS.VALUE),2,'0')
|     |     +-------IF FOUND()
|     |     |           THIS.VALUE = PADL(ALLTRIM(THIS.VALUE),2,'0')
|     |     |           THISFORM.Pageframe1.page7.Pageframe1.Page1.text2.VALUE = v_ipcrfunc1.codfunci
&& Traemos informacion (Codigo y Nombre del funcionario)
|     |     |           REQUERY('v_ipcfunci')
|     |     |
|     |     |           SELE v_ipcfunci
|     |     |           LOCATE FOR ALLTRIM(v_ipcfunci.codfunci) = ALLTRIM(v_ipcrfunc1.codfunci)
|     |     |     +-------IF FOUND()
|     |     |     |           THISFORM.Pageframe1.page7.Pageframe1.Page1.text10.VALUE =
v_ipcfunci.nomfunci
|     |     |     +-------ENDIF
&& Ejecutamos los metodos
|     |     |           THISFORM.actualizalistas1
|     |     |           THISFORM.conteo
&& para actualizar las listas que contendrian las fuentes para esa zona y el conteo correspondiente
|     |     +-------ELSE
|     |     |           = MESSAGEBOX ('Zona NO Encontrada para este tipo de Funcionarios',0+64,'Informacion')
|     |     +-------ENDIF
|     +-------ENDIF
|           SELECT(calias)
```

```
|
|
+-------ENDPROC
```

5.21.5.10 PAGEFRAME1.PAGE7.pageframe1.page2.cmdAdd

```
    cmdAdd.TOOLTIPTEXT = "Agrega Articulos de la Fuente Seleccionada Al reporte de Supervisión"
    cmdAdd.NAME = "cmdAdd"
+-------PROCEDURE cmdAdd.CLICK
&& Validamos que haya seleccionado zona y fuente a la que se le van asignar articulos
&& para supervisar
|   +-------IF !EMPTY(THISFORM.Pageframe1.page7.Pageframe1.Page2.Text3.VALUE) AND ;
|   |   |       !EMPTY(THISFORM.Pageframe1.page7.Pageframe1.Page2.ECodFte.VALUE)
&& Almacenamos el contenido de la lista de articulo no asignados
&& hacemos la asignacion de articulo o articulos para supervisar
|   |       ListaSeleccionada = THISFORM.Pageframe1.page7.Pageframe1.Page2.Moverlista1.lstSource
|   |       THISFORM.LOCKSCREEN=.T.
|   |   +-------IF cambio
|   |   |       supervi_fecha = DTOC(DATE())
|   |   |       supervi_fecha = SUBSTR(supervi_fecha, 1, 2)
|   |   |       dia_supervi = VAL(supervi_fecha)
&& Los articulos seleccionados se los asignamos para ser supervisados, marcando el campo con .T.
|   |   |       Cuenta = ListaSeleccionada.LISTCOUNT
|   |   |   +-------FOR i = 1 TO Cuenta
|   |   |   |   +-------IF ListaSeleccionada.SELECTED(i)
|   |   |   |   |       LaFuente = THISFORM.Pageframe1.page7.Pageframe1.Page2.ECodFte.VALUE
|   |   |   |   |       ElArticulo = SUBSTR(ListaSeleccionada.LIST(i), 1, 7)
|   |   |   |   |       SELE v_zonaplacri2
|   |   |   |   |       REPLACE v_zonaplacri2.supervisado WITH .T., ;
|   |   |   |   |           v_zonaplacri2.placri WITH .F., ;
|   |   |   |   |           v_zonaplacri2.diaasupervisar WITH dia_supervi, ;
|   |   |   |   |           v_zonaplacri2.diacri WITH SUBSTR(DTOC(DATE()),1,2), ;
|   |   |   |   |           v_zonaplacri2.sianalisis WITH .T., ;
|   |   |   |   |           v_zonaplacri2.diaanalisis WITH VAL(SUBSTR(DTOC(DATE()),1,2)) ;
|   |   |   |   |           FOR v_zonaplacri2.codfte = ;
|   |   |   |   |           LaFuente AND v_zonaplacri2.codart = ElArticulo
|   |   |   |   |       TABLEUPDATE(.T.)
|   |   |   |   +-------ENDIF
|   |   |   +-------ENDFOR
|   |   |       cambio=.F.
|   |   +-------ENDIF
&& Ejecutamos el metodo que actualiza las listas de acuerdo a los articulos que se asignaron
&& y los que no
|   |       THISFORM.actualicelistas
|   |       THISFORM.LOCKSCREEN=.F.
|   +-------ELSE
|   |       NODEFAULT
|   |       MESSAGEBOX('Debe especificar Zona de Destino y Codigo de Fuente para asignarle el articulo
a Supervisar',0+64,'Informacion')
|   +-------ENDIF
+-------ENDPROC
```

5.21.5.11 PAGEFRAME1.PAGE7.pageframe1.page2.Navigator1

```
	skiptable = v_zonas2
	NAME = "Navigator1"
	cmdTop.NAME = "cmdTop"
	cmdPrior.NAME = "cmdPrior"
	cmdNext.NAME = "cmdNext"
	cmdBottom.NAME = "cmdBottom"
	Datachecker1.NAME = "Datachecker1"
+-------PROCEDURE recordpointermoved
|		LOCAL calias
|		calias=ALIAS()
&& Recorremos las zonas, asignamos la seleccionada a la caja de texto
|		SELE v_zonas2
|		THISFORM.Pageframe1.page7.Pageframe1.Page2.Text3.VALUE =
PADL(ALLTRIM(STR(v_zonas2.zona,2,0)),2,'0')
&& Limpiamos el origen de datos de la barra de navegar fuentes (<<, <, >,>>) mientras se trae la informacion
&& de la nueva fuente
|		THISFORM.Pageframe1.page7.Pageframe1.Page2.navigator2.skiptable = ''
&& Y lo mismo para el objeto que contiene el nombre de la fuente
|		THISFORM.Pageframe1.page7.Pageframe1.Page2.cnombrefuente.VALUE = ''
&& Cargamos variables para traer las fuentes de la zona seleccionada
|		pzonacri = VAL(THISFORM.Pageframe1.page7.Pageframe1.Page2.Text3.VALUE)
|		panoplacri = VAL(THISFORM.ano.VALUE)
|		pmesplacri = VAL(THISFORM.mes.VALUE)
|		pdecplacri = VAL(THISFORM.combo1.VALUE)
&& Codigo de fuentes de la zona
|		REQUERY('v_zonaplacri2')
|		SELE v_zonaplacri2
&& Nombres de fuentes de acuerdo al anterior resultado
|		REQUERY('v_ftesplacri2')
|		SELE v_ftesplacri2
&& Reasignamos el origen de datos a la barra navegadora para recorrer las fuentes que corresponden
&& ala zona seleccionada
|		THISFORM.Pageframe1.page7.Pageframe1.Page2.navigator2.skiptable = 'v_ftesplacri2'
|		GO TOP
|
|		THISFORM.Pageframe1.page7.Pageframe1.Page2.cnombrefuente.VALUE = v_ftesplacri2.NombFte
|		THISFORM.Pageframe1.page7.Pageframe1.Page2.ECodFte.VALUE = v_ftesplacri2.codfte
&& Cargamos variable con el codigo de la fuente seleccionada para traer la informacion de esta
|		vcalifte = THISFORM.Pageframe1.page7.Pageframe1.Page2.ECodFte.VALUE
&& Ejecutamos el metodo que muestra la informacion en laslistas
|		THISFORM.actualicelistas
&& De acuerdo alcargo traemos la informacion de losfuncionarios
|		pcodcargo = 'S'
|		REQUERY('v_ipcrfunc1')
|
|		SELE v_ipcrfunc1
|		LOCATE FOR v_ipcrfunc1.zonafunci =
PADL(ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page2.Text3.VALUE),2,'0')
|	+-------IF FOUND()
|	|		THISFORM.Pageframe1.page7.Pageframe1.Page2.TEXT1.VALUE = v_ipcrfunc1.codfunci
|	+-------ELSE
|	|		THISFORM.Pageframe1.page7.Pageframe1.Page2.TEXT1.VALUE = ''
```

```
|       +-------ENDIF
|
|               REQUERY('v_ipcfunci')
|
|               SELE v_ipcfunci
|               LOCATE FOR ALLTRIM(v_ipcfunci.codfunci) ==
ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page2.TEXT1.VALUE)
|       +-------IF FOUND()
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page2.text9.VALUE = v_ipcfunci.nomfunci
|       +-------ELSE
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page2.text9.VALUE = ''
|       +-------ENDIF
|
|               SELECT(calias)
+-------ENDPROC
```

5.21.5.12 PAGEFRAME1.PAGE7.pageframe1.page2.Text3

```
        NAME = "Text3"
+-------PROCEDURE VALID
|               LOCAL calias
|               calias=ALIAS()
&& Validamos que existan valores para año,mes y  zona
|       +-------IF !EMPTY(THISFORM.ano.VALUE) AND !EMPTY(THISFORM.mes.VALUE) AND
!EMPTY(THIS.VALUE)
|       |               THIS.VALUE = PADL(ALLTRIM(THIS.VALUE),2,'0')
&& Limpiamoslosorigenes de datos de la barra navegadora y del objeto que contiene el nombre de lafuente
&& mientras reconsultamos la informacion de la zona que sea seleccionada y volvemos a asignar origen de
datos
&& a la barra navegadora de fuentes y en las listas los articulos de cada fuente
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page2.navigator2.skiptable = ''
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page2.cnombrefuente.VALUE = ''
&& Cargamos parametros para traer la informacion con la vista
|       |               pzonacri = VAL(THISFORM.Pageframe1.page7.Pageframe1.Page2.Text3.VALUE)
|       |               panoplacri = VAL(THISFORM.ano.VALUE)
|       |               pmesplacri = VAL(THISFORM.mes.VALUE)
|       |               pdecplacri = VAL(THISFORM.combo1.VALUE)
&& Consultamos las fuentes (codigo)
|       |               REQUERY('v_zonaplacri2')
|       |               SELE v_zonaplacri2
&& En base al resultado anterior traemos los nombres de las fuentes
|       |               REQUERY('v_ftesplacri2')
|       |               SELE v_ftesplacri2
&& Asignamos origen de datos a la barra navegadora
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page2.navigator2.skiptable = 'v_ftesplacri2'
|       |               GO TOP
|       |
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page2.cnombrefuente.VALUE =
v_ftesplacri2.NombFte
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page2.ECodFte.VALUE = v_ftesplacri2.codfte
|       |               vcalifte = THISFORM.Pageframe1.page7.Pageframe1.Page2.ECodFte.VALUE
&& Ejecutamos elmetodo que actualiza la infrmacion de laslistas con losarticulos de la fuente seleccionada
|       |               THISFORM.actualicelistas
&& De acuerdo al cargo que se esta trabajando traemos la informacion de los funcionarios
```

```
|       |           pcodcargo = 'S'
|       |           REQUERY('v_ipcrfunc1')
|       |
|       |           SELE v_ipcrfunc1
|       |           LOCATE FOR v_ipcrfunc1.zonafunci =
PADL(ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page2.Text3.VALUE),2,'0')
|       |     +-------IF FOUND()
|       |     |           THISFORM.Pageframe1.page7.Pageframe1.Page2.TEXT1.VALUE = v_ipcrfunc1.codfunci
|       |     |
|       |     |           REQUERY('v_ipcfunci')
|       |     |
|       |     |           SELE v_ipcfunci
|       |     |           LOCATE FOR ALLTRIM(v_ipcfunci.codfunci) = ALLTRIM(v_ipcrfunc1.codfunci)
|       |     |     +-------IF FOUND()
|       |     |     |           THISFORM.Pageframe1.page7.Pageframe1.Page2.text9.VALUE =
v_ipcfunci.nomfunci
|       |     |     +-------ENDIF
|       |     +-------ELSE
|       |     |           = MESSAGEBOX ('Zona NO Encontrada para este tipo de Funcionarios',0+64,'Informacion')
|       |     +-------ENDIF
|       |
|       +-------ENDIF
|           SELECT(calias)
+-------ENDPROC
```

5.21.5.13 PAGEFRAME1.PAGE7.pageframe1.page2.navigator2

```
    skiptable = v_ftesplacri2
    NAME = "Navigator2"
    cmdTop.NAME = "cmdTop"
    cmdPrior.NAME = "cmdPrior"
    cmdNext.NAME = "cmdNext"
    cmdBottom.NAME = "cmdBottom"
    Datachecker1.NAME = "Datachecker1"
+-------PROCEDURE recordpointermoved
&& Definimos y cargamos variable con el nombre del archivo
|           LOCAL calias
|           calias=ALIAS()
&& Luego de que con la otra barra navegadora asignamos las fuentes correpondientes dependiendo de la zona
&& podemos recorres una a una lasfuentes pertenecientes
|           THISFORM.Pageframe1.page7.Pageframe1.Page2.ECodFte.VALUE = v_ftesplacri2.codfte
|           THISFORM.Pageframe1.page7.Pageframe1.Page2.cnombrefuente.VALUE = v_ftesplacri2.NombFte
|           vcalifte = THISFORM.Pageframe1.page7.Pageframe1.Page2.ECodFte.VALUE
&& Ejecutamos el evento para que actualice la informacion de las listas en pantalla
|           THISFORM.actualicelistas
|
|           SELECT(calias)
+-------ENDPROC
```

5.21.5.14 PAGEFRAME1.PAGE7.pageframe1.page2.Ecodfte

```
    TOOLTIPTEXT = "Escriba el Código de la Fuente"
```

```
        NAME = "ECodFte"
+-------PROCEDURE LOSTFOCUS
|       +-------IF !EMPTY(THIS.VALUE)
|       |               LOCAL calias
|       |               calias = ALIAS()
|       |               THISFORM.LOCKSCREEN = .T.
|       |               SELECT v_ftesplacri2
&& Almacenamos el numero del registro
&& actual en una variable para podernos devolver a el en caso de que la busqueda
&& no obtenga resultados
|       |               MarcaRegistro = RECNO()
|       |               SELECT v_ftesplacri2
|       |               LOCATE FOR v_ftesplacri2.codfte = THIS.VALUE
|       |       +-------IF !FOUND()
|       |       |               GO MarcaRegistro
|       |       |               THIS.VALUE = v_ftesplacri2.codfte
|       |       +-------ELSE
&& nos devolvemos al registro que estabamaos inicialmente
|       |       |               vcalifte = THIS.VALUE
&& cargamos la variable con el codigo de fuente y ejecutamos el metodo que actualiza
&& las listas en pantalla segun los articulos que tiene asignados la fuente y
&& los que no
|       |       |               THISFORM.actualicelistas
|       |       +-------ENDIF
|       |               THISFORM.LOCKSCREEN = .F.
|       |               SELE(calias)
|       +-------ENDIF
+-------ENDPROC
```

5.21.5.15 PAGEFRAME1.PAGE7.pageframe1.page2.ListSource

```
+-------PROCEDURE lstSource.ADDITEM
|               LPARAMETERS cItem
|
|               cambio = .T.
+-------ENDPROC
+-------PROCEDURE lstSource.REMOVEITEM
|               LPARAMETERS nIndex
|
|               cambio = .T.
+-------ENDPROC
+-------PROCEDURE lstSource.CLICK
|
|               pobscodobs = SUBSTR(RIGHT(THIS.VALUE,3),1,2)
|       +-------IF !EMPTY(pobscodobs)
|       |               REQUERY('v_obscali')
|       |               SELE v_obscali
|       |       +-------IF !EOF()
|       |       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  v_obscali.observa
|       |       +-------ELSE
|       |       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  ''
|       |       +-------ENDIF
|       +-------ELSE
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  ''
```

```
|       +-------ENDIF
+-------ENDPROC
+-------PROCEDURE lstSource.GOTFOCUS
|
|               pobscodobs = SUBSTR(RIGHT(THIS.VALUE,3),1,2)
|       +-------IF !EMPTY(pobscodobs)
|       |               REQUERY('v_obscali')
|       |               SELE v_obscali
|       |       +-------IF !EOF()
|       |       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  v_obscali.observa
|       |       +-------ELSE
|       |       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  "
|       |       +-------ENDIF
|       +-------ELSE
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  "
|       +-------ENDIF
+-------ENDPROC
+-------PROCEDURE lstSelected.CLICK
|               pobscodobs = SUBSTR(RIGHT(THIS.VALUE,3),1,2)
|       +-------IF !EMPTY(pobscodobs)
|       |               REQUERY('v_obscali')
|       |               SELE v_obscali
|       |       +-------IF !EOF()
|       |       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  v_obscali.observa
|       |       +-------ELSE
|       |       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  "
|       |       +-------ENDIF
|       +-------ELSE
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  "
|       +-------ENDIF
+-------ENDPROC
+-------PROCEDURE lstSelected.GOTFOCUS
|               pobscodobs = SUBSTR(RIGHT(THIS.VALUE,3),1,2)
|       +-------IF !EMPTY(pobscodobs)
|       |               REQUERY('v_obscali')
|       |               SELE v_obscali
|       |       +-------IF !EOF()
|       |       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  v_obscali.observa
|       |       +-------ELSE
|       |       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  "
|       |       +-------ENDIF
|       +-------ELSE
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  "
|       +-------ENDIF
+-------ENDPROC
```

5.21.5.16 PAGEFRAME1.PAGE7.pageframe1.page3.CmdAdd

```
+-------PROCEDURE cmdAdd.CLICK
&& Validamos que haya seleccionado zona destino, es decir, a la que se le va  asignar la
&& fuente o las fuentes seleccionadas
|       +-------IF !EMPTY(THISFORM.Pageframe1.page7.Pageframe1.Page3.Text3.VALUE) AND
!EMPTY(THISFORM.Pageframe1.page7.Pageframe1.Page3.ECodFte.VALUE)
```

```
&& Almacenamos el contenido de la lista de articulo no asignados
&& Consultamos la vista de acuerdo a la zona origen para ubicar la fuente que se queire reasignar
&& zona
|       |               ListaSeleccionada = THISFORM.Pageframe1.page7.Pageframe1.Page3.Moverlista1.lstSource
|       |               THISFORM.LOCKSCREEN=.T.
|       |       +-------IF cambio
&& Procedemos a cambiarl el numero de zona temporalmente para hacer la asignacion
&& al otro recolector
|       |       |               Cuenta = ListaSeleccionada.LISTCOUNT
|       |       |       +-------FOR i = 1 TO Cuenta
|       |       |       |       +-------IF ListaSeleccionada.SELECTED(i)
|       |       |       |       |               LaFuente = THISFORM.Pageframe1.page7.Pageframe1.Page3.ECodFte.VALUE
|       |       |       |       |               ElArticulo = SUBSTR(ListaSeleccionada.LIST(i), 1, 7)
|       |       |       |       |               SELE v_zonaplacri2
|       |       |       |       |               REPLACE v_zonaplacri2.calicri WITH .T., ;
|       |       |       |       |                       v_zonaplacri2.diacalicri WITH SUBSTR(DTOC(DATE()),1,2) ;
|       |       |       |       |                       FOR v_zonaplacri2.codfte = ;
|       |       |       |       |                       LaFuente AND v_zonaplacri2.codart = ElArticulo
|       |       |       |       |               TABLEUPDATE(.T.)
|       |       |       |       +-------ENDIF
|       |       |       +-------ENDFOR
|       |       |               cambio=.F.
|       |       +-------ENDIF
|       |
|       |               THISFORM.listascalicritica
|       |               THISFORM.LOCKSCREEN=.F.
|       +-------ELSE
|       |               NODEFAULT
|       |               MESSAGEBOX('Debe especificar Zona de Destino y Codigo de Fuente para asignarle el articulo
a Supervisar',0+64,'Informacion')
|       +-------ENDIF
+-------ENDPROC
```

5.21.5.17 PAGEFRAME1.PAGE7.pageframe1.page3.Navigator1

```
        skiptable = v_zonas3
        NAME = "Navigator1"
        cmdTop.NAME = "cmdTop"
        cmdPrior.NAME = "cmdPrior"
        cmdNext.NAME = "cmdNext"
        cmdBottom.NAME = "cmdBottom"
        Datachecker1.NAME = "Datachecker1"
+-------PROCEDURE recordpointermoved
|               LOCAL calias
|               calias=ALIAS()
|
|               SELE v_zonas3
|               THISFORM.Pageframe1.page7.Pageframe1.Page3.Text3.VALUE =
PADL(ALLTRIM(STR(v_zonas3.zona,2,0)),2,'0')
|
|               THISFORM.Pageframe1.page7.Pageframe1.Page3.navigator2.skiptable = ''
|               THISFORM.Pageframe1.page7.Pageframe1.Page3.cnombrefuente.VALUE = ''
|
```

```
|           pzonacri = VAL(THISFORM.Pageframe1.page7.Pageframe1.Page3.Text3.VALUE)
|           panoplacri = VAL(THISFORM.ano.VALUE)
|           pmesplacri = VAL(THISFORM.mes.VALUE)
|           pdecplacri = VAL(THISFORM.combo1.VALUE)
|
|           REQUERY('v_zonaplacri2')
|           SELE v_zonaplacri2
|
|           REQUERY('v_ftesplacri2')
|           SELE v_ftesplacri2
|
|     +-------IF THISFORM.Pageframe1.page7.Pageframe1.Page3.filtrar.VALUE = .T.
|     |           SELE FLOOR(AVG(cuentaarti)) AS vpromedio FROM v_ftesplacri2 INTO CURSO cpromedio
|     |           SELE cpromedio
|     |
|     |           SELE v_ftesplacri2
|     |           SET FILTER TO v_ftesplacri2.cuentaarti >= cpromedio.vpromedio
|     +-------ELSE
|     |           SELE v_ftesplacri2
|     |           SET FILTER TO
|     +-------ENDIF
|
|           THISFORM.Pageframe1.page7.Pageframe1.Page3.navigator2.skiptable = 'v_ftesplacri2'
|           GO TOP
|
|           THISFORM.Pageframe1.page7.Pageframe1.Page3.cnombrefuente.VALUE = v_ftesplacri2.NombFte
|           THISFORM.Pageframe1.page7.Pageframe1.Page3.ECodFte.VALUE = v_ftesplacri2.codfte
|           pobsctvimp = v_ftesplacri2.IMPctv
|           pobscodfte = v_ftesplacri2.codfte
|           THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  ''
|           vcalifte1 = THISFORM.Pageframe1.page7.Pageframe1.Page3.ECodFte.VALUE
|
|           THISFORM.listascalicritica
|
|           pcodcargo = 'C'
|           REQUERY('v_ipcrfunc1')
|
|           SELE v_ipcrfunc1
|           LOCATE FOR v_ipcrfunc1.zonafunci =
PADL(ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page3.Text3.VALUE),2,'0')
|     +-------IF FOUND()
|     |           THISFORM.Pageframe1.page7.Pageframe1.Page3.TEXT1.VALUE = v_ipcrfunc1.codfunci
|     +-------ELSE
|     |           THISFORM.Pageframe1.page7.Pageframe1.Page3.TEXT1.VALUE = ''
|     +-------ENDIF
|
|           REQUERY('v_ipcfunci')
|
|           SELE v_ipcfunci
|           LOCATE FOR ALLTRIM(v_ipcfunci.codfunci) ==
ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page3.TEXT1.VALUE)
|     +-------IF FOUND()
|     |           THISFORM.Pageframe1.page7.Pageframe1.Page3.text9.VALUE = v_ipcfunci.nomfunci
```

```
|       +-------ELSE
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page3.text9.VALUE = ''
|       +-------ENDIF
|
|             THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  ''
|             SELECT(calias)
+-------ENDPROC
```

5.21.5.18 PAGEFRAME1.PAGE7.pageframe1.page3.text3

```
        NAME = "Text3"
+-------PROCEDURE VALID
|             LOCAL calias
|             calias=ALIAS()
|
|       +-------IF !EMPTY(THISFORM.ano.VALUE) AND !EMPTY(THISFORM.mes.VALUE) AND
!EMPTY(THIS.VALUE)
|       |             THIS.VALUE = PADL(ALLTRIM(THIS.VALUE),2,'0')
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page3.navigator2.skiptable = ''
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page3.cnombrefuente.VALUE = ''
|       |
|       |             pzonacri = VAL(THISFORM.Pageframe1.page7.Pageframe1.Page3.Text3.VALUE)
|       |             panoplacri = VAL(THISFORM.ano.VALUE)
|       |             pmesplacri = VAL(THISFORM.mes.VALUE)
|       |             pdecplacri = VAL(THISFORM.combo1.VALUE)
|       |
|       |             REQUERY('v_zonaplacri2')
|       |             SELE v_zonaplacri2
|       |
|       |             REQUERY('v_ftesplacri2')
|       |             SELE v_ftesplacri2
|       |
|       |       +-------IF THISFORM.Pageframe1.page7.Pageframe1.Page3.filtrar.VALUE = .T.
|       |       |             SELE FLOOR(AVG(cuentaarti)) AS vpromedio FROM v_ftesplacri2 INTO CURSO
cpromedio
|       |       |             SELE cpromedio
|       |       |
|       |       |             SELE v_ftesplacri2
|       |       |             SET FILTER TO v_ftesplacri2.cuentaarti >= cpromedio.vpromedio
|       |       +-------ELSE
|       |       |             SELE v_ftesplacri2
|       |       |             SET FILTER TO
|       |       +-------ENDIF
|       |
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page3.navigator2.skiptable = 'v_ftesplacri2'
|       |             GO TOP
|       |
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page3.cnombrefuente.VALUE =
v_ftesplacri2.NombFte
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page3.ECodFte.VALUE = v_ftesplacri2.codfte
|       |             vcalifte1 = THISFORM.Pageframe1.page7.Pageframe1.Page3.ECodFte.VALUE
|       |
|       |             THISFORM.listascalicritica
|       |
```

```
|       |           pcodcargo = 'C'
|       |           REQUERY('v_ipcrfunc1')
|       |
|       |           SELE v_ipcrfunc1
|       |           LOCATE FOR v_ipcrfunc1.zonafunci =
PADL(ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page3.Text3.VALUE),2,'0')
|       |     +-------IF FOUND()
|       |     |           THISFORM.Pageframe1.page7.Pageframe1.Page3.TEXT1.VALUE = v_ipcrfunc1.codfunci
|       |     |
|       |     |           REQUERY('v_ipcfunci')
|       |     |
|       |     |           SELE v_ipcfunci
|       |     |           LOCATE FOR ALLTRIM(v_ipcfunci.codfunci) = ALLTRIM(v_ipcrfunc1.codfunci)
|       |     |     +-------IF FOUND()
|       |     |     |           THISFORM.Pageframe1.page7.Pageframe1.Page3.text9.VALUE =
v_ipcfunci.nomfunci
|       |     |     +-------ENDIF
|       |     +-------ELSE
|       |     |           = MESSAGEBOX ('Zona NO Encontrada para este tipo de Funcionarios',0+64,'Informacion')
|       |     +-------ENDIF
|       |
|       +-------ENDIF
|
|           THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  ''
|           SELECT(calias)
+-------ENDPROC
```

5.21.5.19 PAGEFRAME1.PAGE7.pageframe1.page3.Navigator2

```
    skiptable = v_ftesplacri2
    NAME = "Navigator2"
    cmdTop.NAME = "cmdTop"
    cmdPrior.NAME = "cmdPrior"
    cmdNext.NAME = "cmdNext"
    cmdBottom.NAME = "cmdBottom"
    Datachecker1.NAME = "Datachecker1"
+-------PROCEDURE recordpointermoved
&& Definimos y cargamos variable con el nombre del archivo
|           LOCAL calias
|           calias=ALIAS()
|
|           THISFORM.Pageframe1.page7.Pageframe1.Page3.ECodFte.VALUE = v_ftesplacri2.codfte
|           THISFORM.Pageframe1.page7.Pageframe1.Page3.cnombrefuente.VALUE = v_ftesplacri2.NombFte
|           pobsctvimp = v_ftesplacri2.IMPctv
|           pobscodfte = v_ftesplacri2.codfte
|           vcalifte1 = THISFORM.Pageframe1.page7.Pageframe1.Page3.ECodFte.VALUE
|           THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  ''
&& Ejecutamos el evento para que actualice la informacion de las listas en pantalla
|           THISFORM.listascalicritica
|
|
|           SELECT(calias)
+-------ENDPROC
```

5.21.5.19.1 PAGEFRAME1.PAGE7.pageframe1.page3.ecodfte

```
        NAME = "ECodFte"
+-------PROCEDURE LOSTFOCUS
|       +-------IF !EMPTY(THIS.VALUE)
|       |               LOCAL calias
|       |               calias = ALIAS()
|       |
|       |               THISFORM.LOCKSCREEN = .T.
|       |
|       |               SELECT v_ftesplacri2
&& Almacenamos el numero del registro
&& actual en una variable para podernos devolver a el en caso de que la busqueda
&& no obtenga resultados
|       |               MarcaRegistro = RECNO()
|       |               SELECT v_ftesplacri2
|       |               LOCATE FOR v_ftesplacri2.codfte = THIS.VALUE
|       |       +-------IF !FOUND()
|       |       |               GO MarcaRegistro
|       |       |               THIS.VALUE = v_ftesplacri2.codfte
|       |       |               pobsctvimp = v_ftesplacri2.IMPctv
|       |       |               pobscodfte = v_ftesplacri2.codfte
|       |       +-------ELSE
&& nos devolvemos al registro que estabamaos inicialmente
|       |       |               vcalifte1 = THIS.VALUE
&& cargamos la variable con el codigo de fuente y ejecutamos el metodo que actualiza
&& las listas en pantalla segun los articulos que tiene asignados la fuente y
&& los que no
|       |       |               THISFORM.listascalicritica
|       |       +-------ENDIF
|       |
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page3.ecaliobs.VALUE =  ''
|       |               THISFORM.LOCKSCREEN = .F.
|       |               SELE(calias)
|       +-------ENDIF
+-------ENDPROC
```

5.21.5.20 PAGEFRAME1.PAGE7.pageframe1.page3.filtrar

```
        NAME = "filtrar"
+-------PROCEDURE INTERACTIVECHANGE
|       +-------IF THIS.VALUE = .T.
|       |               THIS.CAPTION = 'Fuentes Grandes'
|       +-------ELSE
|       |               THIS.CAPTION = 'Todas'
|       +-------ENDIF
+-------ENDPROC
```

5.21.5.21 PAGEFRAME1.PAGE7.pageframe1.page4.Navigator1

```
        skiptable = v_zonas4
        NAME = "Navigator1"
        cmdTop.NAME = "cmdTop"
        cmdPrior.NAME = "cmdPrior"
        cmdNext.NAME = "cmdNext"
```

```
      cmdBottom.NAME = "cmdBottom"
      Datachecker1.NAME = "Datachecker1"
+-------PROCEDURE recordpointermoved
|             LOCAL calias
|             calias=ALIAS()
|             SELE v_zonas4
|             THISFORM.Pageframe1.page7.Pageframe1.Page4.Text3.VALUE =
PADL(ALLTRIM(STR(v_zonas4.zona,2,0)),2,'0')
|
|             THISFORM.Pageframe1.page7.Pageframe1.Page4.navigator2.skiptable = ''
|             THISFORM.Pageframe1.page7.Pageframe1.Page4.cnombrefuente.VALUE = ''
|
|             pzonacap = VAL(THISFORM.Pageframe1.page7.Pageframe1.Page4.Text3.VALUE)
|             panocap = VAL(THISFORM.ano.VALUE)
|             pmescap = VAL(THISFORM.mes.VALUE)
|             pdeccap = VAL(THISFORM.combo1.VALUE)
|
|             REQUERY('v_zonaplacap2')
|             SELE v_zonaplacap2
|
|             REQUERY('v_ftesplacap2')
|             SELE v_ftesplacap2
|
|      +-------IF THISFORM.Pageframe1.page7.Pageframe1.Page4.filtrar.VALUE = .T.
|      |             SELE FLOOR(AVG(cuentaarti)) AS vpromedio FROM v_ftesplacap2 INTO CURSOR cpromedio
|      |             SELE cpromedio
|      |
|      |             SELE v_ftesplacap2
|      |             SET FILTER TO v_ftesplacap2.cuentaarti >= cpromedio.vpromedio
|      +-------ELSE
|      |             SELE v_ftesplacap2
|      |             SET FILTER TO
|      +-------ENDIF
|
|             THISFORM.Pageframe1.page7.Pageframe1.Page4.navigator2.skiptable = 'v_ftesplacap2'
|             GO TOP
|
|             THISFORM.Pageframe1.page7.Pageframe1.Page4.cnombrefuente.VALUE = v_ftesplacap2.NombFte
|             THISFORM.Pageframe1.page7.Pageframe1.Page4.ECodFte.VALUE = v_ftesplacap2.codfte
|             pcalictvo = v_ftesplacap2.IMPctv
|             vcalifte2 = THISFORM.Pageframe1.page7.Pageframe1.Page4.ECodFte.VALUE
|
|             THISFORM.listascalicaptura
|
|             pcodcargo = 'T'
|             REQUERY('v_ipcrfunc1')
|
|             SELE v_ipcrfunc1
|             LOCATE FOR v_ipcrfunc1.zonafunci =
PADL(ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page4.Text3.VALUE),2,'0')
|      +-------IF FOUND()
|      |             THISFORM.Pageframe1.page7.Pageframe1.Page4.TEXT1.VALUE = v_ipcrfunc1.codfunci
|      +-------ELSE
```

```
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page4.TEXT1.VALUE = ''
|       +-------ENDIF
|
|             REQUERY('v_ipcfunci')
|             SELE v_ipcfunci
|
|             LOCATE FOR ALLTRIM(v_ipcfunci.codfunci) ==
ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page4.TEXT1.VALUE)
|       +-------IF FOUND()
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page4.text9.VALUE = v_ipcfunci.nomfunci
|       +-------ELSE
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page4.text9.VALUE = ''
|       +-------ENDIF
|
|             SELECT(calias)
|             THISFORM.Pageframe1.page7.Pageframe1.Page4.ECodFte.SETFOCUS
+-------ENDPROC
```

5.21.5.22 PAGEFRAME1.PAGE7.pageframe1.page3.text3

```
        NAME = "Text3"
+-------PROCEDURE VALID
|             LOCAL calias
|             calias=ALIAS()
|
|       +-------IF !EMPTY(THISFORM.ano.VALUE) AND !EMPTY(THISFORM.mes.VALUE) AND
!EMPTY(THIS.VALUE)
|       |
|       |             THIS.VALUE = PADL(ALLTRIM(THIS.VALUE),2,'0')
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page4.navigator2.skiptable = ''
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page4.cnombrefuente.VALUE = ''
|       |
|       |             pzonacap = VAL(THISFORM.Pageframe1.page7.Pageframe1.Page4.Text3.VALUE)
|       |             panocap = VAL(THISFORM.ano.VALUE)
|       |             pmescap = VAL(THISFORM.mes.VALUE)
|       |             pdeccap = VAL(THISFORM.combo1.VALUE)
|       |
|       |             REQUERY('v_zonaplacap2')
|       |             SELE v_zonaplacap2
|       |
|       |             REQUERY('v_ftesplacap2')
|       |             SELE v_ftesplacap2
|       |
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page3.navigator2.skiptable = 'v_ftesplacap2'
|       |             GO TOP
|       |
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page4.cnombrefuente.VALUE =
v_ftesplacap2.NombFte
|       |             THISFORM.Pageframe1.page7.Pageframe1.Page4.ECodFte.VALUE = v_ftesplacap2.codfte
|       |             pcalictvo = v_ftesplacap2.IMPctv
|       |             vcalifte2 = THISFORM.Pageframe1.page7.Pageframe1.Page4.ECodFte.VALUE
|       |
|       |             THISFORM.listascalicaptura
|       |
|       |             pcodcargo = 'T'
```

```
|       |           REQUERY('v_ipcrfunc1')
|       |
|       |           SELE v_ipcrfunc1
|       |           LOCATE FOR v_ipcrfunc1.zonafunci =
PADL(ALLTRIM(THISFORM.Pageframe1.page7.Pageframe1.Page4.Text3.VALUE),2,'0')
|       |       +-------IF FOUND()
|       |       |           THISFORM.Pageframe1.page7.Pageframe1.Page4.TEXT1.VALUE = v_ipcrfunc1.codfunci
|       |       |
|       |       |           REQUERY('v_ipcfunci')
|       |       |           SELE v_ipcfunci
|       |       |
|       |       |           LOCATE FOR ALLTRIM(v_ipcfunci.codfunci) = ALLTRIM(v_ipcrfunc1.codfunci)
|       |       |       +-------IF FOUND()
|       |       |       |           THISFORM.Pageframe1.page7.Pageframe1.Page4.text9.VALUE =
v_ipcfunci.nomfunci
|       |       |       +-------ENDIF
|       |       +-------ELSE
|       |       |           = MESSAGEBOX ('Zona NO Encontrada para este tipo de Funcionarios',0+64,'Informacion')
|       |       +-------ENDIF
|       |
|       +-------ENDIF
|
|           SELECT(calias)
|
+-------ENDPROC
```

5.21.5.23 PAGEFRAME1.PAGE7.pageframe1.page3.Navigator2

```
    skiptable = v_ftesplacap2
    NAME = "Navigator2"
    cmdTop.NAME = "cmdTop"
    cmdPrior.NAME = "cmdPrior"
    cmdNext.NAME = "cmdNext"
    cmdBottom.NAME = "cmdBottom"
    Datachecker1.NAME = "Datachecker1"
+-------PROCEDURE recordpointermoved
&& Definimos y cargamos variable con el nombre del archivo
|           LOCAL calias
|           calias=ALIAS()
|           THISFORM.Pageframe1.page7.Pageframe1.Page4.ECodFte.VALUE = v_ftesplacap2.codfte
|           THISFORM.Pageframe1.page7.Pageframe1.Page4.cnombrefuente.VALUE = v_ftesplacap2.NombFte
|           pcalictvo = v_ftesplacap2.IMPctv
|           vcalifte2 = THISFORM.Pageframe1.page7.Pageframe1.Page4.ECodFte.VALUE
&& Ejecutamos el evento para que actualice la informacion de las listas en pantalla
|           THISFORM.listascalicaptura
|
|           SELECT(calias)
+-------ENDPROC
```

5.21.5.24 PAGEFRAME1.PAGE7.pageframe1.page3.Filtrar

```
    NAME = "filtrar"
+-------PROCEDURE INTERACTIVECHANGE
|       +-------IF THIS.VALUE = .T.
|       |           THIS.CAPTION = 'Fuentes Grandes'
|       +-------ELSE
```

```
|       |             THIS.CAPTION = 'Todas'
|       +-------ENDIF
+-------ENDPROC
```

5.21.5.25 PAGEFRAME1.PAGE7.pageframe1.page4.grid1

```
RECORDSOURCE = "v_artcalicap"
NAME = "Grid1"

Column1.CONTROLSOURCE = "v_artcalicap.codart"
Column1.NAME = "Column1"
Column2.CONTROLSOURCE = "v_artcalicap.nomart"
Column2.NAME = "Column2"
Column3.CONTROLSOURCE = "v_artcalicap.pesart"
Column3.NAME = "Column3"
column4.CONTROLSOURCE = "v_artcalicap.cantant"
column4.NAME = "Column4"
column5.CONTROLSOURCE = "v_artcalicap.panart"
column5.NAME = "Column5"
column6.CONTROLSOURCE = "v_artcalicap.pesmvt"
column6.NAME = "Column6"
column7.CONTROLSOURCE = "v_artcalicap.cmvmvt"
column7.NAME = "Column7"
column8.CONTROLSOURCE = "v_artcalicap.crfmvt"
column8.NAME = "Column8"
column9.CONTROLSOURCE = "v_artcalicap.sinmmvt"
column9.NAME = "Column9"
column10.CONTROLSOURCE = "v_artcalicap.fcommvt"
column10.NAME = "Column10"
column11.FONTSIZE = 8
column11.CONTROLSOURCE = "v_artcalicap.canrmvt"
column11.ENABLED = .F.
column11.WIDTH = 40
column11.NAME = "Column11"
column12.FONTSIZE = 8
column12.CONTROLSOURCE = "v_artcalicap.pactmvt"
column12.NAME = "Column12"
column13.CONTROLSOURCE = "v_artcalicap.variacion"
column13.NAME = "Column13"
column14.CONTROLSOURCE = "v_artcalicap.obsmvt"
column14.NAME = "Column14"
Column15.CONTROLSOURCE = "v_artcalicap.calicap"
Column15.NAME = "Column15"
```

Column1

```
CAPTION = "Código"
CONTROLSOURCE = "v_artcalicap.codart"
NAME = "Text1"
```

Column2

```
CAPTION = "Nombre"
```

CONTROLSOURCE = "v_artcalicap.nomart"
NAME = "Text1"

Column3
CAPTION = "P.E."
check1.CONTROLSOURCE = "v_artcalicap.pesart"
check1.NAME = "Check1"

Column4
CAPTION = "Canti."
CONTROLSOURCE = "v_artcalicap.cantant"
NAME = "Text1"

Column5
CAPTION = "Precio"
NAME = "Text1"

Column6
CAPTION = "P.E."
check1.CONTROLSOURCE = "v_artcalicap.pesmvt"
check1.NAME = "Check1"

Column7
CAPTION = "C.M.V"
check1.CONTROLSOURCE = "v_artcalicap.cmvmvt"
check1.NAME = "Check1"

Column8
CAPTION = "C.R."
check1.CONTROLSOURCE = "v_artcalicap.crfmvt"
check1.NAME = "Check1"

Column9
CAPTION = "S.I."
check1.CONTROLSOURCE = "v_artcalicap.sinmmvt"
check1.NAME = "Check1"

Column10
CAPTION = "F.C."
CONTROLSOURCE = "v_artcalicap.fcommvt"
NAME = "Text1"

Column11
CAPTION = "Canti."
CONTROLSOURCE = "v_artcalicap.canrmvt"
NAME = "Text1"

Column12
CAPTION = "Precio"
CONTROLSOURCE = "v_artcalicap.pactmvt"
NAME = "Text1"

Column13

```
        CAPTION = "Varia."
        CONTROLSOURCE = "v_artcalicap.variacion"
        NAME = "Text1"
```

Column14

```
        CAPTION = "Obs."
        CONTROLSOURCE = "v_artcalicap.obsmvt"
        NAME = "Text1"

+-------PROCEDURE GOTFOCUS
|       +-------IF !EMPTY(THIS.VALUE)
|       |               LOCAL calias
|       |               calias = ALIAS()
|       |               THISFORM.LOCKSCREEN = .T.
|       |
|       |               pcalicodobs = PADL(ALLTRIM(THIS.VALUE),2,'0')
|       |               vcalifte5 = THISFORM.Pageframe1.page7.Pageframe1.Page4.ECodFte.VALUE
|       |
|       |               REQUERY('v_caliobs')
|       |               SELE v_caliobs
|       |
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page4.ecaliobs.VALUE = v_caliobs.observa
|       |
|       |               THISFORM.LOCKSCREEN = .F.
|       |               SELE(calias)
|       +-------ELSE
|       |               THISFORM.Pageframe1.page7.Pageframe1.Page4.ecaliobs.VALUE = ''
|       +-------ENDIF
+-------ENDPROC
```

Column15

```
        CAPTION = "©"
        check1.CONTROLSOURCE = "v_artcalicap.calicap"
        check1.NAME = "Check1"
+-------PROCEDURE check1.VALID
|               REPLACE v_artcalicap.diacalicap WITH SUBSTR(DTOC(DATE()),1,2)
+-------ENDPROC
```

5.21.5.26 PAGEFRAME1.PAGE7.pageframe1.page4.Ecodfte

```
        NAME = "ECodFte"
+-------PROCEDURE LOSTFOCUS
|       +-------IF !EMPTY(THIS.VALUE)
|       |               LOCAL calias
|       |               calias = ALIAS()
|       |
|       |               THISFORM.LOCKSCREEN = .T.
|       |               SELECT v_ftesplacap2
&& Almacenamos el numero del registro
&& actual en una variable para podernos devolver a el en caso de que la busqueda
&& no obtenga resultados
|       |               MarcaRegistro = RECNO()
```

```
|       |               SELECT v_ftesplacap2
|       |               LOCATE FOR v_ftesplacap2.codfte = THIS.VALUE
|       |       +-------IF !FOUND()
|       |       |               GO MarcaRegistro
|       |       |               THIS.VALUE = v_ftesplacap2.codfte
|       |       |               pcalictvo = v_ftesplacap2.IMPctv
|       |       +-------ELSE
&& nos devolvemos al registro que estabamaos inicialmente
|       |       |               pcalictvo = v_ftesplacap2.IMPctv
|       |       |               vcalifte2 = THIS.VALUE
&& cargamos la variable con el codigo de fuente y ejecutamos el metodo que actualiza
&& las listas en pantalla segun los articulos que tiene asignados la fuente y
&& los que no
&& Ejecutamos el evento para que actualice la informacion de las listas en pantalla
|       |       |               THISFORM.listascalicaptura
|       |       |
|       |       +-------ENDIF
|       |               THISFORM.LOCKSCREEN = .F.
|       |               SELE(calias)
|       +-------ENDIF
+-------ENDPROC
```

5.21.6 PAGEFRAME1.PAGE8

5.21.6.1 PAGEFRAME1.PAGE8.command3

```
        CAPTION = "Supervisión"
        TOOLTIPTEXT = "Calcular Indicadores Artículos Supervisados"
        FORECOLOR = 0,0,128
        NAME = "Command3"
+-------PROCEDURE CLICK
|               LOCAL calias
|               calias = ALIAS()
|               DO FORM indicado WITH '1'
|               SELECT (calias)
+-------ENDPROC
```

5.21.6.2 PAGEFRAME1.PAGE8.command1

```
        CAPTION = "Recolección"
        TOOLTIPTEXT = "Calcular Indicadores de Recolección (Artículos Supervisados y No Supervisados)"
        NAME = "Command1"
+-------PROCEDURE CLICK
|               LOCAL calias
|               calias = ALIAS()
|               DO FORM indicado WITH '5'
|               SELECT (calias)
+-------ENDPROC
```

5.21.6.3 PAGEFRAME1.PAGE8.command2

```
        CAPTION = "Crítica"
        TOOLTIPTEXT = "Calcular Indicadores para Planilla de Crítica"
        NAME = "Command2"
```

```
+-------PROCEDURE CLICK
|           LOCAL calias
|           calias = ALIAS()
|           DO FORM indicado WITH '2'
|           SELECT (calias)
+-------ENDPROC
```

5.21.6.4 PAGEFRAME1.PAGE8.command5

```
      CAPTION = "Indicadores Calidad Crítica"
      TOOLTIPTEXT = "Calcular Indicadores para Control de Calidad de Crítica"
      NAME = "Command5"
+-------PROCEDURE CLICK
|           LOCAL calias
|           calias = ALIAS()
|           DO FORM indicado WITH '3'
|           SELECT (calias)
+-------ENDPROC
```

5.21.6.5 PAGEFRAME1.PAGE8.command6

```
      CAPTION = "Indicadores Calidad Captura"
      TOOLTIPTEXT = "Calcular Indicadores para Control de Calidad de Captura"
      NAME = "Command6"
+-------PROCEDURE CLICK
|           LOCAL calias
|           calias = ALIAS()
|           DO FORM indicado WITH '4'
|           SELECT (calias)
+-------ENDPROC
```

5.21.6.6 PAGEFRAME1.PAGE8.command4

```
      CAPTION = "Generar Reporte de Inconsistencias"
      TOOLTIPTEXT = "Calcular Indicadores para Control de Calidad de Captura"
      NAME = "Command4"
+-------PROCEDURE CLICK
|           LOCAL calias
|           calias = ALIAS()
|
|           PUBLIC vtcantidad1, vtcantidad2, vtcantidad
|           STORE 0 TO vtcantidad1, vtcantidad2, vtcantidad
|
|           DO FORM FProgreso NAME progreso LINKED
|           progreso.LProgreso.CAPTION = " Procesando, un momento por favor...."
|           progreso.barraprogreso1.Maxvalor = 100
|           progreso.barraprogreso1.Minvalor = 10
|
&& Avanzamos la barra de progreso
|           progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 5
|           progreso.REFRESH
|           progress()
|           CREATE CURSOR cincon(ano C(4), mes C(2), decada C(1), cargo C(20), codfunci C(10), ;
|                 nomfunci C(30), umr C(3), pmr C(3), iape C(3), iasi C(3), iacr C(3), ;
|                 iafn C(3), adi C(3), onas C(3), nae C(3), crea1 C(3), crur C(3), ;
|                 crpr C(3), crape C(3), crasi C(3), ;
|                 cracr C(3), crafc C(3), crob C(3), caea C(3), caur C(3), capr C(3), ;
```

```
|           caape C(3), caasi C(3), ;
|           caacr C(3), caafc C(3), caob C(3), cantidad C(4), indicador N(8,4) DEFAULT 100.0000 )
|
|       CREATE CURSOR recolector(ano C(4), mes C(2), decada C(1), cargo C(20), codfunci C(10), ;
|           nomfunci C(30), umr C(3), pmr C(3), iape C(3), iasi C(3), iacr C(3), ;
|           iafn C(3), adi C(3), onas C(3), nae C(3), cantidad C(4), indicador N(8,4) DEFAULT 100.0000 )
|
|       CREATE CURSOR critico(ano C(4), mes C(2), decada C(1), cargo C(20), codfunci C(10), ;
|           nomfunci C(30), ea C(3), ur C(3), pr C(3), ape C(3), asi C(3), ;
|           acr C(3), afc C(3), ob C(3), cantidad C(4), indicador N(8,4) DEFAULT 100.0000 )
|
|       CREATE CURSOR captura(ano C(4), mes C(2), decada C(1), cargo C(20), codfunci C(10), ;
|           nomfunci C(30), ea C(3), ur C(3), pr C(3), ape C(3), asi C(3), ;
|           acr C(3), afc C(3), ob C(3), cantidad C(4), indicador N(8,4) DEFAULT 100.0000 )
|       **********
|       progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 5
|       progreso.REFRESH
|       progress()
|
|       REQUERY('v_indplacrixzona')
|       SELE v_indplacrixzona
|
|       progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 5
|       progreso.REFRESH
|       progress()
|
|       REQUERY('v_ctaplacrixzona')
|       SELE v_ctaplacrixzona
|
|       REQUERY('v_indplasupxzona')
|       SELE v_indplasupxzona
|
|       REQUERY('v_ctaplasupxzona')
|       SELE v_ctaplasupxzona
|
|       SELECT (((v_ctaplasupxzona.ctaftes * VAL(v_indplasupxzona.indicadeca)) + ;
|           (v_ctaplacrixzona.ctaftes * VAL(v_indplacrixzona.indicadeca))) / ;
|           (v_ctaplasupxzona.ctaftes + v_ctaplacrixzona.ctaftes)) AS indicazona,;
|           v_indplasupxzona.zona, PADL(ALLTRIM(STR(v_indplasupxzona.zona,4,0)), 2, '0') AS ;
|           zonafunciz ;
|           FROM  ipcred!v_indplasupxzona inner JOIN ipcred!v_ctaplasupxzona;
|           inner JOIN ipcred!v_ctaplacrixzona;
|           FULL JOIN ipcred!v_indplasupxzona ;
|           ON  v_indplasupxzona.zona = v_indplacrixzona.zona ;
|           ON  v_indplacrixzona.zona = v_ctaplacrixzona.zona ;
|           ON  v_indplasupxzona.zona = v_ctaplasupxzona.zona;
|           GROUP BY v_indplasupxzona.zona;
|           ORDER BY v_indplasupxzona.zona       ;
|           INTO CURSOR totalreco
|
|       SELE totalreco
|       INDEX ON zonafunciz TAG zonaz
|
```

```
|           *************
|           progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 5
|           progreso.REFRESH
|           progress()
|
|           REQUERY('v_funcirec')
|           SELE v_funcirec
|     +-------IF !EOF()
|     |           SELE recolector
|     |           APPEND BLANK
|     |           REPLACE recolector.cargo WITH 'RECOLECTOR', ;
|     |                 recolector.indicador WITH 0
|     |           SELE v_funcirec
|     |     +-------SCAN
|     |     |           SELE recolector
|     |     |           APPEND BLANK
|     |     |           REPLACE recolector.ano WITH STR(v_funcirec.ano,4,0) ;
|     |     |                 recolector.mes WITH PADL(ALLTRIM(STR(v_funcirec.mes, 2, 0)), 2, '0') ;
|     |     |                 recolector.codfunci WITH v_funcirec.codfunci ;
|     |     |                 recolector.nomfunci WITH v_funcirec.nomfunci
|     |     |           SELE totalreco
|     |     |           LOCATE FOR totalreco.zonafunciz = v_funcirec.zonafunci
|     |     |     +-------IF FOUND()
|     |     |     |           REPLACE recolector.indicador WITH totalreco.indicazona
|     |     |     +-------ENDIF
|     |     |           SELE v_funcirec
|     |     +-------ENDSCAN
|     +-------ENDIF
|
|           progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 10
|           progreso.REFRESH
|           progress()
|
|           REQUERY('v_funcimessupcri')
|           SELE v_funcimessupcri
|           SELE recolector
|           GO TOP
|     +-------SCAN FOR !EMPTY(codfunci)
|     |           SELECT DISTINCT (COUNT(sumr) + COUNT(cumr)) AS vumr ;
|     |                 FROM v_funcimessupcri ;
|     |                 WHERE (v_funcimessupcri.codfunci = recolector.codfunci) AND ;
|     |                 (v_funcimessupcri.sumr = .T. OR v_funcimessupcri.cumr = .T.) ;
|     |                 INTO CURSOR crecolectores
|     |
|     |           SELE crecolectores
|     |           REPLACE recolector.umr WITH STR(crecolectores.vumr,3,0) ;
|     |                 recolector.cantidad WITH STR(VAL(recolector.cantidad) + crecolectores.vumr,4,0)
|     |
|     |           SELECT (COUNT(spmr) + COUNT(cpmr)) AS vpmr ;
|     |                 FROM v_funcimessupcri ;
|     |                 WHERE (v_funcimessupcri.codfunci = recolector.codfunci) AND ;
|     |                 (v_funcimessupcri.spmr = .T. OR v_funcimessupcri.cpmr = .T.) ;
|     |                 INTO CURSOR crecolectores
```

```
|     |
|     |        SELE crecolectores
|     |        REPLACE recolector.pmr WITH STR(crecolectores.vpmr,3,0) ;
|     |            recolector.cantidad WITH STR(VAL(recolector.cantidad) + crecolectores.vpmr,4,0)
|     |
|     |        SELECT (COUNT(siape) + COUNT(ciape)) AS viape ;
|     |            FROM v_funcimessupcri ;
|     |            WHERE (v_funcimessupcri.codfunci = recolector.codfunci) AND ;
|     |            (v_funcimessupcri.siape = .T. OR v_funcimessupcri.ciape = .T.) ;
|     |            INTO CURSOR crecolectores
|     |
|     |        SELE crecolectores
|     |        REPLACE recolector.iape WITH STR(crecolectores.viape,3,0) ;
|     |            recolector.cantidad WITH STR(VAL(recolector.cantidad) + crecolectores.viape,4,0)
|     |
|     |        SELECT (COUNT(siasi) + COUNT(ciasi)) AS viasi ;
|     |            FROM v_funcimessupcri ;
|     |            WHERE (v_funcimessupcri.codfunci = recolector.codfunci) AND ;
|     |            (v_funcimessupcri.siasi = .T. OR v_funcimessupcri.ciasi = .T.) ;
|     |            INTO CURSOR crecolectores
|     |
|     |        SELE crecolectores
|     |        REPLACE recolector.iasi WITH STR(crecolectores.viasi,3,0) ;
|     |            recolector.cantidad WITH STR(VAL(recolector.cantidad) + crecolectores.viasi,4,0)
|     |
|     |        SELECT (COUNT(siacr) + COUNT(ciacr)) AS viacr ;
|     |            FROM v_funcimessupcri ;
|     |            WHERE (v_funcimessupcri.codfunci = recolector.codfunci) AND ;
|     |            (v_funcimessupcri.siacr = .T. OR v_funcimessupcri.ciacr = .T.) ;
|     |            INTO CURSOR crecolectores
|     |
|     |        SELE crecolectores
|     |        REPLACE recolector.iacr WITH STR(crecolectores.viacr,3,0) ;
|     |            recolector.cantidad WITH STR(VAL(recolector.cantidad) + crecolectores.viacr,4,0)
|     |
|     |        SELECT (COUNT(siafn) + COUNT(ciafn)) AS viafn ;
|     |            FROM v_funcimessupcri ;
|     |            WHERE (v_funcimessupcri.codfunci = recolector.codfunci) AND ;
|     |            (v_funcimessupcri.siafn = .T. OR v_funcimessupcri.ciafn = .T.) ;
|     |            INTO CURSOR crecolectores
|     |
|     |        SELE crecolectores
|     |        REPLACE recolector.iafn WITH STR(crecolectores.viafn,3,0) ;
|     |            recolector.cantidad WITH STR(VAL(recolector.cantidad) + crecolectores.viafn,4,0)
|     |
|     |        SELECT (COUNT(sadi) + COUNT(cadi)) AS vadi ;
|     |            FROM v_funcimessupcri ;
|     |            WHERE (v_funcimessupcri.codfunci = recolector.codfunci) AND ;
|     |            (v_funcimessupcri.sadi = .T. OR v_funcimessupcri.cadi = .T.) ;
|     |            INTO CURSOR crecolectores
|     |
|     |        SELE crecolectores
|     |        REPLACE recolector.adi WITH STR(crecolectores.vadi,3,0) ;
```

```
|     |                 recolector.cantidad WITH STR(VAL(recolector.cantidad) + crecolectores.vadi,4,0)
|     |
|     |           SELECT (COUNT(sonas) + COUNT(conas)) AS vonas ;
|     |                 FROM v_funcimessupcri ;
|     |                 WHERE (v_funcimessupcri.codfunci = recolector.codfunci) AND ;
|     |                 (v_funcimessupcri.sonas = .T. OR v_funcimessupcri.conas = .T.) ;
|     |                 INTO CURSOR crecolectores
|     |
|     |           SELE crecolectores
|     |           REPLACE recolector.onas WITH STR(crecolectores.vonas,3,0) ;
|     |                 recolector.cantidad WITH STR(VAL(recolector.cantidad) + crecolectores.vonas,4,0)
|     |
|     |           SELECT (COUNT(snae) + COUNT(cnae)) AS vnae ;
|     |                 FROM v_funcimessupcri ;
|     |                 WHERE (v_funcimessupcri.codfunci = recolector.codfunci) AND ;
|     |                 (v_funcimessupcri.snae = .T. OR v_funcimessupcri.cnae = .T.) ;
|     |                 INTO CURSOR crecolectores
|     |
|     |           SELE crecolectores
|     |           REPLACE recolector.nae WITH STR(crecolectores.vnae,3,0) ;
|     |                 recolector.cantidad WITH STR(VAL(recolector.cantidad) + crecolectores.vnae,4,0)
|     |
|     |           SELE recolector
|     +-------ENDSCAN
|
|           progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 10
|           progreso.REFRESH
|           progress()
|
```

&& Procedimiento para Critica

```
|           REQUERY('v_indcalicriz')
|           SELE v_indcalicriz
|
|           REQUERY('v_funcicri')
|           SELE v_funcicri
|     +-------IF !EOF()
|     |           SELE critico
|     |           APPEND BLANK
|     |           REPLACE critico.cargo WITH 'CRITICO', ;
|     |                 critico.indicador WITH 0
|     |           SELE v_funcicri
|     |     +-------SCAN
|     |     |           SELE critico
|     |     |           APPEND BLANK
|     |     |           REPLACE critico.ano WITH STR(v_funcicri.ano,4,0) ;
|     |     |                 critico.mes WITH PADL(ALLTRIM(STR(v_funcicri.mes, 2, 0)), 2, '0') ;
|     |     |                 critico.codfunci WITH v_funcicri.codfunci ;
|     |     |                 critico.nomfunci WITH v_funcicri.nomfunci
|     |     |
|     |     |           SELE v_indcalicriz
|     |     |           LOCATE FOR v_indcalicriz.codfunci = v_funcicri.codfunci
```

```
|     |     |     +-------IF FOUND()
|     |     |     |           REPLACE critico.indicador WITH v_indcalicriz.indicadeca
|     |     |     +-------ENDIF
|     |     |           SELE v_funcicri
|     |     +-------ENDSCAN
|     +-------ENDIF
|
|           progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 10
|           progreso.REFRESH
|           progress()
|
|           REQUERY('v_funcimescri')
|           SELE v_funcimescri
|           SELE critico
|           GO TOP
|     +-------SCAN FOR !EMPTY(codfunci)
|     |
|     |           SELECT DISTINCT COUNT(CREA) AS vcrea ;
|     |                 FROM v_funcimescri ;
|     |                 WHERE (v_funcimescri.codfunci = critico.codfunci) AND ;
|     |                 v_funcimescri.CREA = .T. ;
|     |                 INTO CURSOR ccriticos
|     |
|     |           SELE ccriticos
|     |           REPLACE critico.ea WITH STR(ccriticos.vcrea,3,0) ;
|     |                 critico.cantidad WITH STR(VAL(critico.cantidad) + ccriticos.vcrea,4,0)
|     |
|     |           SELECT DISTINCT COUNT(crur) AS vcrur ;
|     |                 FROM v_funcimescri ;
|     |                 WHERE (v_funcimescri.codfunci = critico.codfunci) AND ;
|     |                 v_funcimescri.crur = .T. ;
|     |                 INTO CURSOR ccriticos
|     |
|     |           SELE ccriticos
|     |           REPLACE critico.ur WITH STR(ccriticos.vcrur,3,0) ;
|     |                 critico.cantidad WITH STR(VAL(critico.cantidad) + ccriticos.vcrur,4,0)
|     |
|     |           SELECT DISTINCT COUNT(crpr) AS vcrpr ;
|     |                 FROM v_funcimescri ;
|     |                 WHERE (v_funcimescri.codfunci = critico.codfunci) AND ;
|     |                 v_funcimescri.crpr = .T. ;
|     |                 INTO CURSOR ccriticos
|     |
|     |           SELE ccriticos
|     |           REPLACE critico.pr WITH STR(ccriticos.vcrpr,3,0) ;
|     |                 critico.cantidad WITH STR(VAL(critico.cantidad) + ccriticos.vcrpr,4,0)
|     |
|     |           SELECT DISTINCT COUNT(crape) AS vcrape ;
|     |                 FROM v_funcimescri ;
|     |                 WHERE (v_funcimescri.codfunci = critico.codfunci) AND ;
|     |                 v_funcimescri.crape = .T. ;
|     |                 INTO CURSOR ccriticos
|     |
```

```
|     |          SELE ccriticos
|     |          REPLACE critico.ape WITH STR(ccriticos.vcrape,3,0) ;
|     |               critico.cantidad WITH STR(VAL(critico.cantidad) + ccriticos.vcrape,4,0)
|     |
|     |          SELECT DISTINCT COUNT(crasi) AS vcrasi ;
|     |               FROM v_funcimescri ;
|     |               WHERE (v_funcimescri.codfunci = critico.codfunci) AND ;
|     |               v_funcimescri.crasi = .T. ;
|     |               INTO CURSOR ccriticos
|     |
|     |          SELE ccriticos
|     |          REPLACE critico.asi WITH STR(ccriticos.vcrasi,3,0) ;
|     |               critico.cantidad WITH STR(VAL(critico.cantidad) + ccriticos.vcrasi,4,0)
|     |
|     |          SELECT DISTINCT COUNT(cracr) AS vcracr ;
|     |               FROM v_funcimescri ;
|     |               WHERE (v_funcimescri.codfunci = critico.codfunci) AND ;
|     |               v_funcimescri.cracr = .T. ;
|     |               INTO CURSOR ccriticos
|     |
|     |          SELE ccriticos
|     |          REPLACE critico.acr WITH STR(ccriticos.vcracr,3,0) ;
|     |               critico.cantidad WITH STR(VAL(critico.cantidad) + ccriticos.vcracr,4,0)
|     |
|     |          SELECT DISTINCT COUNT(crafc) AS vcrafc ;
|     |               FROM v_funcimescri ;
|     |               WHERE (v_funcimescri.codfunci = critico.codfunci) AND ;
|     |               v_funcimescri.crafc = .T. ;
|     |               INTO CURSOR ccriticos
|     |
|     |          SELE ccriticos
|     |          REPLACE critico.afc WITH STR(ccriticos.vcrafc,3,0) ;
|     |               critico.cantidad WITH STR(VAL(critico.cantidad) + ccriticos.vcrafc,4,0)
|     |
|     |          SELECT DISTINCT COUNT(crob) AS vcrob ;
|     |               FROM v_funcimescri ;
|     |               WHERE (v_funcimescri.codfunci = critico.codfunci) AND ;
|     |               v_funcimescri.crob = .T. ;
|     |               INTO CURSOR ccriticos
|     |
|     |          SELE ccriticos
|     |          REPLACE critico.ob WITH STR(ccriticos.vcrob,3,0) ;
|     |               critico.cantidad WITH STR(VAL(critico.cantidad) + ccriticos.vcrob,4,0)
|     |
|     |          SELE critico
|     +-------ENDSCAN
|
|          progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 10
|          progreso.REFRESH
|          progress()
|
```

&& Procedimientopara Captura

```
|           REQUERY('v_indcalicapz')
|           SELE v_indcalicapz
|
|           REQUERY('v_funcicap')
|           SELE v_funcicap
|     +-------IF !EOF()
|     |           SELE captura
|     |           APPEND BLANK
|     |           REPLACE captura.cargo WITH 'CAPTURA', ;
|     |                 captura.indicador WITH 0
|     |           SELE v_funcicap
|     |     +-------SCAN
|     |     |           SELE captura
|     |     |           APPEND BLANK
|     |     |           REPLACE captura.ano WITH STR(v_funcicap.ano,4,0) ;
|     |     |                 captura.mes WITH PADL(ALLTRIM(STR(v_funcicap.mes, 2, 0)), 2, '0') ;
|     |     |                 captura.codfunci WITH v_funcicap.codfunci ;
|     |     |                 captura.nomfunci WITH v_funcicap.nomfunci
|     |     |
|     |     |           SELE v_indcalicapz
|     |     |           LOCATE FOR v_indcalicapz.codfunci = v_funcicap.codfunci
|     |     |     +-------IF FOUND()
|     |     |     |           REPLACE captura.indicador WITH v_indcalicapz.indicadeca
|     |     |     +-------ENDIF
|     |     |
|     |     |           SELE v_funcicap
|     |     +-------ENDSCAN
|     +-------ENDIF
|
|           progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 10
|           progreso.REFRESH
|           progress()
|
|           REQUERY('v_funcimescap')
|           SELE v_funcimescap
|           SELE captura
|           GO TOP
|     +-------SCAN FOR !EMPTY(codfunci)
|     |
|     |           SELECT DISTINCT COUNT(caea) AS vcaea ;
|     |                 FROM v_funcimescap ;
|     |                 WHERE (v_funcimescap.codfunci = captura.codfunci) AND ;
|     |                 v_funcimescap.caea= .T. ;
|     |                 INTO CURSOR ccapturas
|     |
|     |           SELE ccapturas
|     |           REPLACE captura.ea WITH STR(ccapturas.vcaea,3,0) ;
|     |                 captura.cantidad WITH STR(VAL(captura.cantidad) + ccapturas.vcaea,4,0)
|     |
|     |           SELECT DISTINCT COUNT(caur) AS vcaur ;
|     |                 FROM v_funcimescap ;
|     |                 WHERE (v_funcimescap.codfunci = captura.codfunci) AND ;
|     |                 v_funcimescap.caur = .T. ;
```

```
|     |          INTO CURSOR ccapturas
|     |
|     |     SELE ccapturas
|     |     REPLACE captura.ur WITH STR(ccapturas.vcaur,3,0) ;
|     |          captura.cantidad WITH STR(VAL(captura.cantidad) + ccapturas.vcaur,4,0)
|     |
|     |     SELECT DISTINCT COUNT(capr) AS vcapr ;
|     |          FROM v_funcimescap ;
|     |          WHERE (v_funcimescap.codfunci = captura.codfunci) AND ;
|     |          v_funcimescap.capr = .T. ;
|     |          INTO CURSOR ccapturas
|     |
|     |     SELE ccapturas
|     |     REPLACE captura.pr WITH STR(ccapturas.vcapr,3,0) ;
|     |          captura.cantidad WITH STR(VAL(captura.cantidad) + ccapturas.vcapr,4,0)
|     |
|     |     SELECT DISTINCT COUNT(caape) AS vcaape ;
|     |          FROM v_funcimescap ;
|     |          WHERE (v_funcimescap.codfunci = captura.codfunci) AND ;
|     |          v_funcimescap.caape = .T. ;
|     |          INTO CURSOR ccapturas
|     |
|     |     SELE ccapturas
|     |     REPLACE captura.ape WITH STR(ccapturas.vcaape,3,0) ;
|     |          captura.cantidad WITH STR(VAL(captura.cantidad) + ccapturas.vcaape,4,0)
|     |
|     |     SELECT DISTINCT COUNT(caasi) AS vcaasi ;
|     |          FROM v_funcimescap ;
|     |          WHERE (v_funcimescap.codfunci = captura.codfunci) AND ;
|     |          v_funcimescap.caasi = .T. ;
|     |          INTO CURSOR ccapturas
|     |
|     |     SELE ccapturas
|     |     REPLACE captura.asi WITH STR(ccapturas.vcaasi,3,0) ;
|     |          captura.cantidad WITH STR(VAL(captura.cantidad) + ccapturas.vcaasi,4,0)
|     |
|     |     SELECT DISTINCT COUNT(caacr) AS vcaacr ;
|     |          FROM v_funcimescap ;
|     |          WHERE (v_funcimescap.codfunci = captura.codfunci) AND ;
|     |          v_funcimescap.caacr = .T. ;
|     |          INTO CURSOR ccapturas
|     |
|     |     SELE ccapturas
|     |     REPLACE captura.acr WITH STR(ccapturas.vcaacr,3,0) ;
|     |          captura.cantidad WITH STR(VAL(captura.cantidad) + ccapturas.vcaacr,4,0)
|     |
|     |     SELECT DISTINCT COUNT(caafc) AS vcaafc ;
|     |          FROM v_funcimescap ;
|     |          WHERE (v_funcimescap.codfunci = captura.codfunci) AND ;
|     |          v_funcimescap.caafc = .T. ;
|     |          INTO CURSOR ccapturas
|     |
|     |     SELE ccapturas
```

```
|     |           REPLACE captura.afc WITH STR(ccapturas.vcaafc,3,0) ;
|     |                 captura.cantidad WITH STR(VAL(captura.cantidad) + ccapturas.vcaafc,4,0)
|     |
|     |           SELECT DISTINCT COUNT(caob) AS vcaob ;
|     |                 FROM v_funcimescap ;
|     |                 WHERE (v_funcimescap.codfunci = captura.codfunci) AND ;
|     |                 v_funcimescap.caob = .T. ;
|     |                 INTO CURSOR ccapturas
|     |
|     |           SELE ccapturas
|     |           REPLACE captura.ob WITH STR(ccapturas.vcaob,3,0) ;
|     |                 captura.cantidad WITH STR(VAL(captura.cantidad) + ccapturas.vcaob,4,0)
|     |
|     |           SELE captura
|     +-------ENDSCAN
|
|           progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 10
|           progreso.REFRESH
|           progress()
&& Termina Procedimiento para cargar cursores con la informacion del formato de Inconsistencias Presentadas
&& aun estan cada uno por aparte, ahora los dejamos en uno solo
|           SELE recolector
|     +-------SCAN
|     |           SCATTER MEMVAR
|     |           SELE cincon
|     |           APPEND BLANK
|     |           GATHER MEMVAR
|     |           SELE recolector
|     +-------ENDSCAN
|
|           CALCULATE SUM(VAL(recolector.cantidad)) TO vtcantidad
|
|           progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 10
|           progreso.REFRESH

|           progress()
|
|           SELE critico
|     +-------SCAN
|     |           INSERT INTO cincon(ano, mes, decada, cargo, codfunci, nomfunci, crea1, crur, crpr, crape,
crasi, cracr, crafc, crob, cantidad, indicador);
|     |                 VALUE(critico.ano, critico.mes, critico.decada, critico.cargo, critico.codfunci,
critico.nomfunci, critico.ea, critico.ur, critico.pr, critico.ape, critico.asi, critico.acr, critico.afc, critico.ob,
critico.cantidad, critico.indicador)
|     |           SELE critico
|     +-------ENDSCAN
|           CALCULATE SUM(VAL(critico.cantidad)) TO vtcantidad1
|
|
|           SELE captura
|     +-------SCAN
|     |           INSERT INTO cincon(ano, mes, decada, cargo, codfunci, nomfunci, caea, caur, capr, caape,
caasi, caacr, caafc, caob, cantidad, indicador);
```

```
|       |                 VALUE(captura.ano, captura.mes, captura.decada, captura.cargo, captura.codfunci,
captura.nomfunci, captura.ea, captura.ur, captura.pr, captura.ape, captura.asi, captura.acr, captura.afc,
captura.ob, captura.cantidad, captura.indicador)
|       |         SELE captura
|       +-------ENDSCAN
|           CALCULATE SUM(VAL(captura.cantidad)) TO vtcantidad2
|           progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 10
|           progreso.REFRESH
|           progress()
|
|           PUBLIC vtumr, vtpmr, vtiape, vtiasi, vtiacr, vtiafn, vtadi, vtonas, vtnae
|           PUBLIC vtcrea1, vtcrur, vtcrpr, vtcrape, vtcrasi, vtcracr, vtcrafc, vtcrob
|           PUBLIC vtcaea, vtcaur, vtcapr, vtcaape, vtcaasi, vtcaacr, vtcaafc, vtcaob
|
|           SELE cincon
|           SUM  VAL(cincon.umr), VAL(cincon.pmr), VAL(cincon.iape), VAL(cincon.iasi), VAL(cincon.iacr), ;
|                VAL(cincon.iafn), VAL(cincon.adi), VAL(cincon.onas), VAL(cincon.nae), ;
|                VAL(cincon.crea1), VAL(cincon.crur), VAL(cincon.crpr), VAL(cincon.crape), VAL(cincon.crasi), ;
|                VAL(cincon.cracr), VAL(cincon.crafc), VAL(cincon.crob), ;
|                VAL(cincon.caea), VAL(cincon.caur), VAL(cincon.capr), VAL(cincon.caape), VAL(cincon.caasi), ;
|                VAL(cincon.caacr), VAL(cincon.caafc), VAL(cincon.caob) ;
|                TO vtumr, vtpmr, vtiape, vtiasi, vtiacr, vtiafn, vtadi, vtonas, vtnae, ;
|                vtcrea1, vtcrur, vtcrpr, vtcrape, vtcrasi, vtcracr, vtcrafc, vtcrob, ;
|                vtcaea, vtcaur, vtcapr, vtcaape, vtcaasi, vtcaacr, vtcaafc, vtcaob
|
|           progreso.barraprogreso1.Llene
|           progreso.RELEASE
|
|           SELE cincon
|           DO FORM ('impresion1.scx') WITH REPinconsis
|
+-------ENDPROC
```

5.21.6.7 PAGEFRAME1.PAGE8.List1

```
      NAME = "List1"
+-------PROCEDURE INIT
|           SELE v_zonas
|     +-------SCAN
|     |         THIS.ADDLISTITEM(ALLTRIM(STR(v_zonas.zona)))
|     +-------ENDSCAN
|           THIS.ADDLISTITEM('Todas')
+-------ENDPROC
```

5.21.6.8 PAGEFRAME1.PAGE8.List2

```
      NAME = "List2"
+-------PROCEDURE INIT
|           THIS.ADDLISTITEM('1')
|           THIS.ADDLISTITEM('2')
|           THIS.ADDLISTITEM('3')
|           THIS.ADDLISTITEM('Todas')
+-------ENDPROC
```

5.21.6.9 PAGEFRAME1.PAGE8.command7

```
      CAPTION = "Generar Reporte"
```

```
        TOOLTIPTEXT = "Procesar y Generar Reporte"
        FORECOLOR = 0,0,128
        NAME = "Command7"
+-------PROCEDURE CLICK
|               LOCAL calias
|               calias = ALIAS()
|       +-------WITH THISFORM.Pageframe1.Page8
|       |               DO FORM FProgreso NAME progreso LINKED
|       |               progreso.LProgreso.CAPTION = " Procesando, un momento por favor...."
|       |               progreso.barraprogreso1.Maxvalor = 100
|       |               progreso.barraprogreso1.Minvalor = 10
|       |
&& Avanzamos la barra de progreso
|       |               progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 10
|       |               progreso.REFRESH
|       |               progress()
&& Cargamos variables con el contenido de las listas
|       |               listazona = .list1
|       |               listadecada = .List2
|       |
|       |               LOCAL pzonat, pdecadat
|       |               STORE .F. TO pzonat, pdecadat
|       |
|       |               CREATE CURSOR czonarep(zona N(2))
|       |               CREATE CURSOR czonarept(zona N(2))
|       |
|       |               CREATE CURSOR cdecarep(decada N(1))
|       |               CREATE CURSOR cdecarept(decada N(1))
|       |
|       |       +-------IF listazona.LISTCOUNT > 0 AND listadecada.LISTCOUNT > 0
|       |       |               Cuenta = listazona.LISTCOUNT
|       |       |       +-------FOR i = 1 TO Cuenta
|       |       |       |               INSERT INTO czonarept(zona);
|       |       |       |                       VALUES(VAL(listazona.LIST(i)))
|       |       |       |
|       |       |       |       +-------IF listazona.SELECTED(i)
|       |       |       |       |       +-------IF listazona.LIST(i) = 'Todas'
|       |       |       |       |       |               pzonat = .T.
|       |       |       |       |       +-------ENDIF
|       |       |       |       |               INSERT INTO czonarep(zona);
|       |       |       |       |                       VALUES(VAL(listazona.LIST(i)))
|       |       |       |       +-------ENDIF
|       |       |       +-------ENDFOR
&& Avanzamos la barra de progreso
|       |       |               progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 10
|       |       |               progreso.REFRESH
|       |       |               progress()
|       |       |
|       |       |               Cuenta1 = listadecada.LISTCOUNT
|       |       |       +-------FOR i = 1 TO Cuenta1
|       |       |       |               INSERT INTO cdecarept(decada);
|       |       |       |                       VALUES(VAL(listadecada.LIST(i)))
|       |       |       |       +-------IF listadecada.SELECTED(i)
```

```
|     |     |     |     |     +-------IF listadecada.LIST(i) = 'Todas'
|     |     |     |     |     |           pdecadat = .T.
|     |     |     |     |     +-------ENDIF
|     |     |     |     |
|     |     |     |     |           INSERT INTO cdecarep(decada);
|     |     |     |     |                 VALUES(VAL(listadecada.LIST(i)))
|     |     |     |     +-------ENDIF

|     |     |     +-------ENDFOR
|     |     +-------ELSE
&& Validamos que hayan seleccionado parametros para procesar (Decada(s) y Zona(s)
|     |     |           = MESSAGEBOX('Debe seleccionar Zona(s) y Decada(s)...',0+64,'Informacion')
|     |     +-------ENDIF
&& Avanzamos la barra de progreso
|     |           progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 10
|     |           progreso.REFRESH
|     |           progress()
|     +-------ENDWITH
|           REQUERY('v_zonasuprep')
|           SELE v_zonasuprep
&& Avanzamos la barra de progreso
|           progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 10
|           progreso.REFRESH
|           progress()
|
|     +-------DO CASE
|     +-------CASE pzonat = .T.
|     |           SELECT * ;
|     |                 FROM v_zonasuprep ;
|     |                 WHERE v_zonasuprep.zona IN (SELECT zona FROM czonarept) ;
|     |                 INTO CURSOR czonas
|     |
|     +-------CASE pzonat = .F.
|     |           SELECT * ;
|     |                 FROM v_zonasuprep ;
|     |                 WHERE v_zonasuprep.zona IN (SELECT zona FROM czonarep) ;
|     |                 INTO CURSOR czonas
|     +-------ENDCASE
&& Avanzamos la barra de progreso
|           progreso.barraprogreso1.progreso=progreso.barraprogreso1.progreso + 10
|           progreso.REFRESH
|           progress()
|
|     +-------DO CASE
|     +-------CASE pdecadat = .T.
|     |           SELECT * ;
|     |                 FROM czonas ;
|     |                 WHERE czonas.decmvt IN (SELECT decada FROM cdecarept) ;
|     |                 INTO CURSOR che6
|     |
|     +-------CASE pdecadat = .F.
|     |           SELECT * ;
|     |                 FROM czonas ;
```

```
|       |                 WHERE czonas.decmvt IN (SELECT decada FROM cdecarep) ;
|       |                 INTO CURSOR che6
|       +-------ENDCASE
|
|               progreso.barraprogreso1.Llene
|               progreso.RELEASE
|
|               SELE che6
|               *casupervi
|       +-------IF !EOF()
|       |               DO FORM ('impresion1.scx') WITH repsupervi
|       +-------ELSE
|       |               = MESSAGEBOX('No existe informacion que cumpla los parametros de consulta, intente de
nuevo por favor...',0+64,'Informacion')
|       +-------ENDIF
|
|               SELECT(calias)
|
+-------ENDPROC
```

5.21.7 Command2

```
        TOOLTIPTEXT = "Descripcion de Abreviaturas de Inconsistencias"
        NAME = "Command2"
+-------PROCEDURE CLICK
&& Mostramos la descripcion de las abreviaturas para los inconsistencias de losformatos de Calidad
|               = MESSAGEBOX('UMR: Unidad Mal Recolectada - 16' + ;
|                       CHR(13) + 'PMR: Precio Mal Recolectado - 16' + ;
|                       CHR(13) + 'IAPE: Incorrecta Aplicación Periodo de Espera - 16' + ;
|                       CHR(13) + 'IASI: Incorrecta Aplicación Sustitución Inmediata - 16' + ;
|                       CHR(13) + 'IACR: Incorrecta Aplicación Cambio de Referencia - 16' + ;
|                       CHR(13) + 'IAFN: Incorrecta Aplicación Fuente Nueva - 5' + ;
|                       CHR(13) + 'ADI: Ausencia De Información - 5' + ;
|                       CHR(13) + 'ONAS: Observación No Acorde a Situación - 5' + ;
|                       CHR(13) + 'NAE: No Actualizó Especificaciones - 5' + ;
|                       CHR(13) + CHR(13) + ;
|                       'IMSXF: Información Mal Suministrada Por La Fuente' + ;
|                       CHR(13) + 'DGF: Datos Generales De La Fuente' + ;
|                       CHR(13) + CHR(13) + 'Procedimiento Adoptado:' + ;
|                       CHR(13) + 'PE: Periodo de Espera' + ;
|                       CHR(13) + 'SI: Sustitución Inmediata' + ;
|                       CHR(13) + 'CR: Cambio de Referencia' + ;
|                       CHR(13) + CHR(13) + 'Control Calidad Crítica:' + ;
|                       CHR(13) + 'EA: Especificaciones Del Articulo - 5' + ;
|                       CHR(13) + 'UR: Unidad Recolectada - 15' + ;
|                       CHR(13) + 'PR: Precio Recolectado - 15' + ;
|                       CHR(13) + 'PE: Aplicación Periodo de Espera - 15' + ;
|                       CHR(13) + 'SI: Aplicación Sustitución Inmediata - 15' + ;
|                       CHR(13) + 'CR: Aplicación Cambio de Referencia - 15' + ;
|                       CHR(13) + 'FC: Aplicación Fuente Complementaria - 10' + ;
|                       CHR(13) + 'OB: Observaciones - 10' + ;
|                       CHR(13) + 'DGF: Datos Generales de la Fuente',0+64,'Descripcion Abreviaturas de
Incosistencias')
+-------ENDPROC
```

5.21.8 Command1

TOOLTIPTEXT = "Descripcion de Formatos"
NAME = "Command1"
+-------PROCEDURE CLICK
&& Mostramos la descripcion de las abreviaturas para los formatos de Calidad
| = MESSAGEBOX('MI-IPC-FOR-01: Planilla de Recolección' + CHR(13) + 'MI-IPC-FOR-02: Planilla de Supervisión' + CHR(13) + 'MI-IPC-FOR-03: Planilla de Crítica' + CHR(13) + 'MI-IPC-FOR-04: Cargas de Trabajo' + CHR(13) + 'MI-IPC-FOR-05: Control de Calidad Procedimiento de Crítica' + CHR(13) + 'MI-IPC-FOR-06: Control de Calidad procedimiento de Captura',0+64,'Descripcion Formatos Indicadores de calidad')
+-------ENDPROC

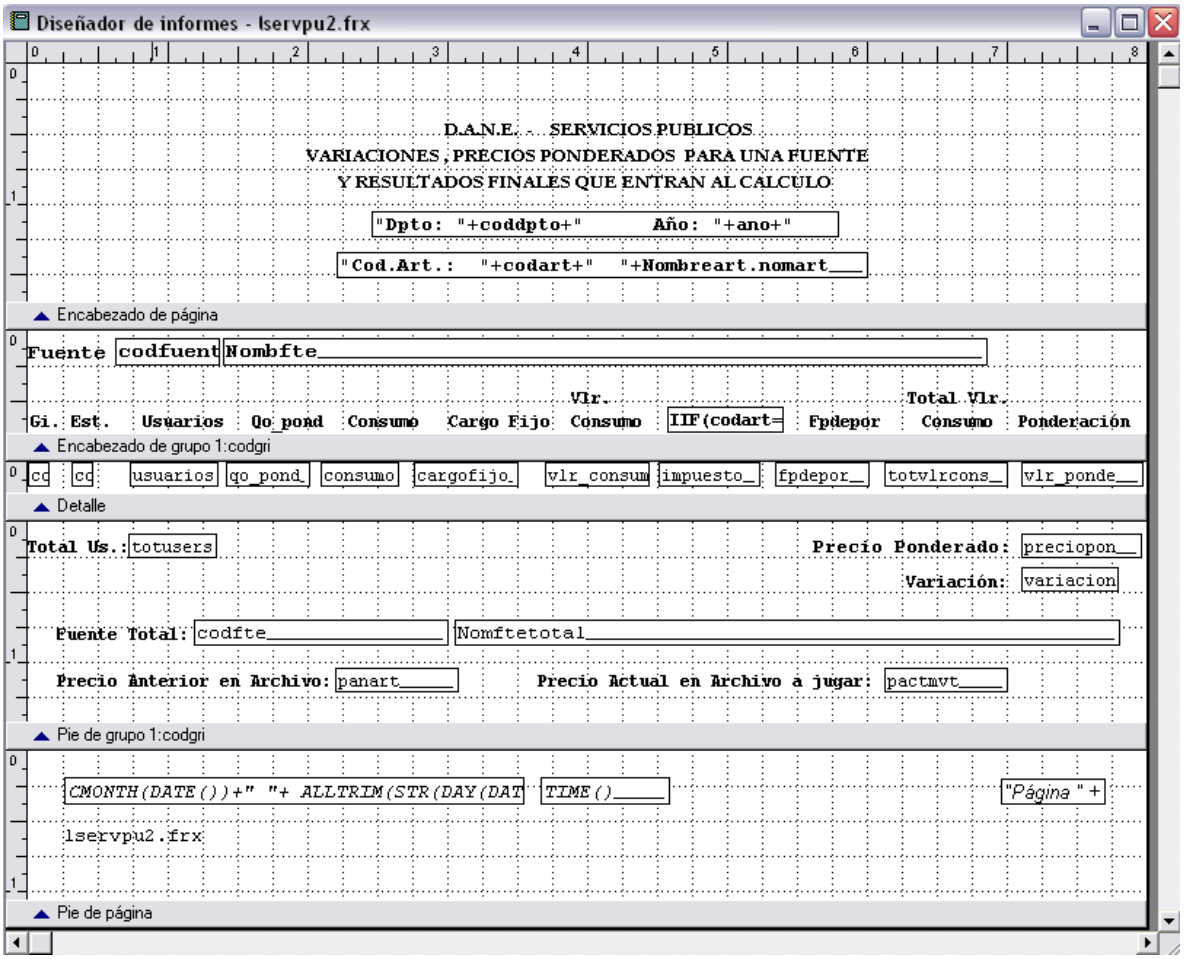
5.21.9 WzBoton4

TOOLTIPTEXT = "Salir de la pantalla"
NAME = "Wzboton4"
+-------PROCEDURE CLICK
| THISFORM.RELEASE
+-------ENDPROC

6 INFORMES

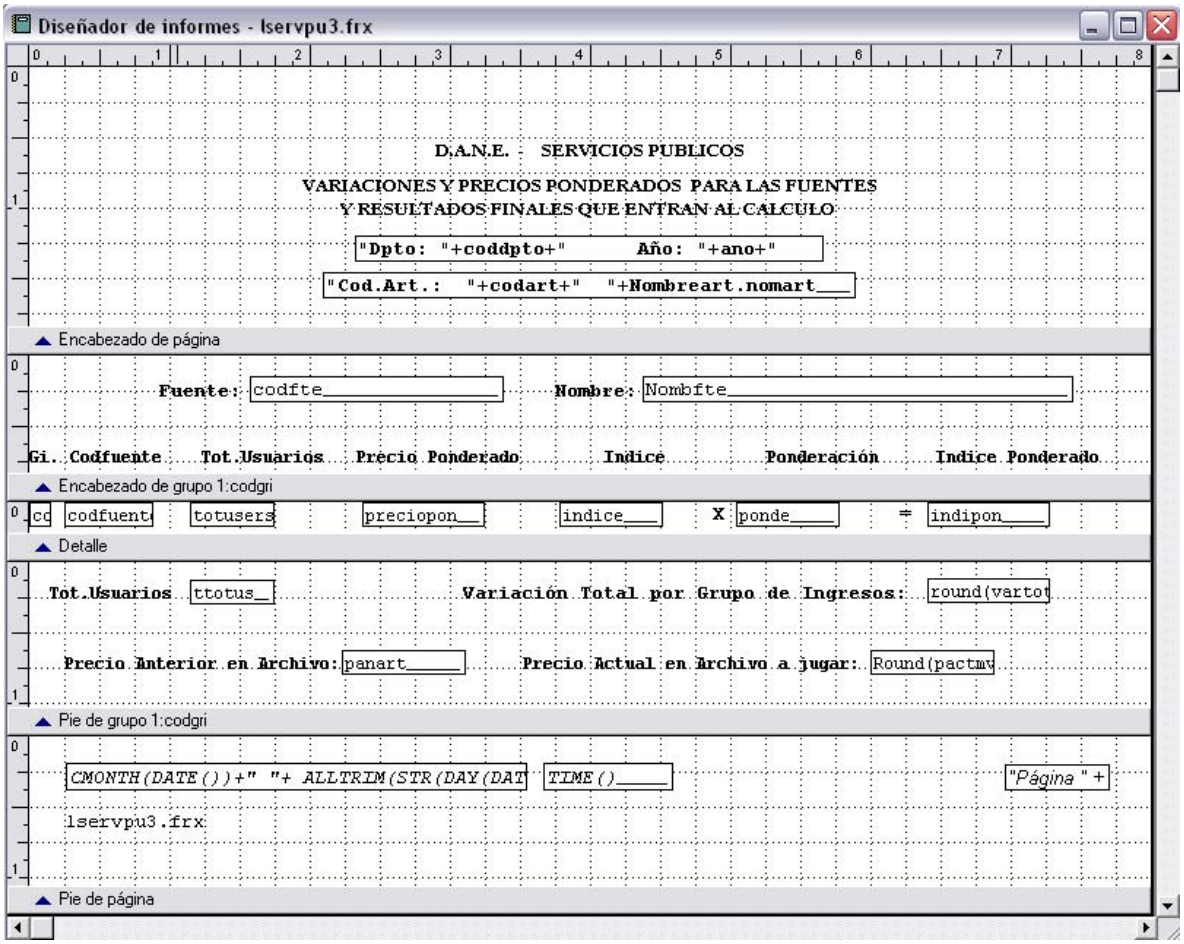
6.1 LSERVPUB2.FRX - VARIACIONES , PRECIOS PONDERADOS PARA UNA FUENTE Y RESULTADOS FINALES QUE ENTRAN AL CALCULO.

Permite la impresión de los precios ponderados y la variación por estratos y total; clase de fuente, para una fuente y resultados finales que entran al calculo. disponible en el formulario de Servicios públicos.


Diseñador de informes - lservpu2.frx
D.A.N.E. - SERVICIOS PUBLICOS
VARIACIONES , PRECIOS PONDERADOS PARA UNA FUENTE
Y RESULTADOS FINALES QUE ENTRAN AL CALCULO
"Dpto: "+coddpto+" Año: "+ano+"
"Cod.Art.: "+codart+" "+Nombreart.nomart
Encabezado de página
Fuente codfuent Nombfte
Gi. Est. Usuarios Qo_pond Consumo Cargo Fijo Vlr. Consumo IIF(codart= Fpdepor Total Vlr. Consumo Ponderación
Encabezado de grupo 1:codgri
cd cd usuarios qo_pond consumo cargofijo vlr_consum impuesto fpdepor totvlrcons vlr_ponde
Detalle
Total Us.: totusers
Precio Ponderado: preciopon
Variación: variacion
Fuente Total: codfte Nomftetotal
Precio Anterior en Archivo: panart
Precio Actual en Archivo a jugar: pactmvt
Pie de grupo 1:codgri
CMONTH(DATE())+" "+ ALLTRIM(STR(DAY(DAT
TIME()
"Página " +
lservpu2.frx
Pie de página

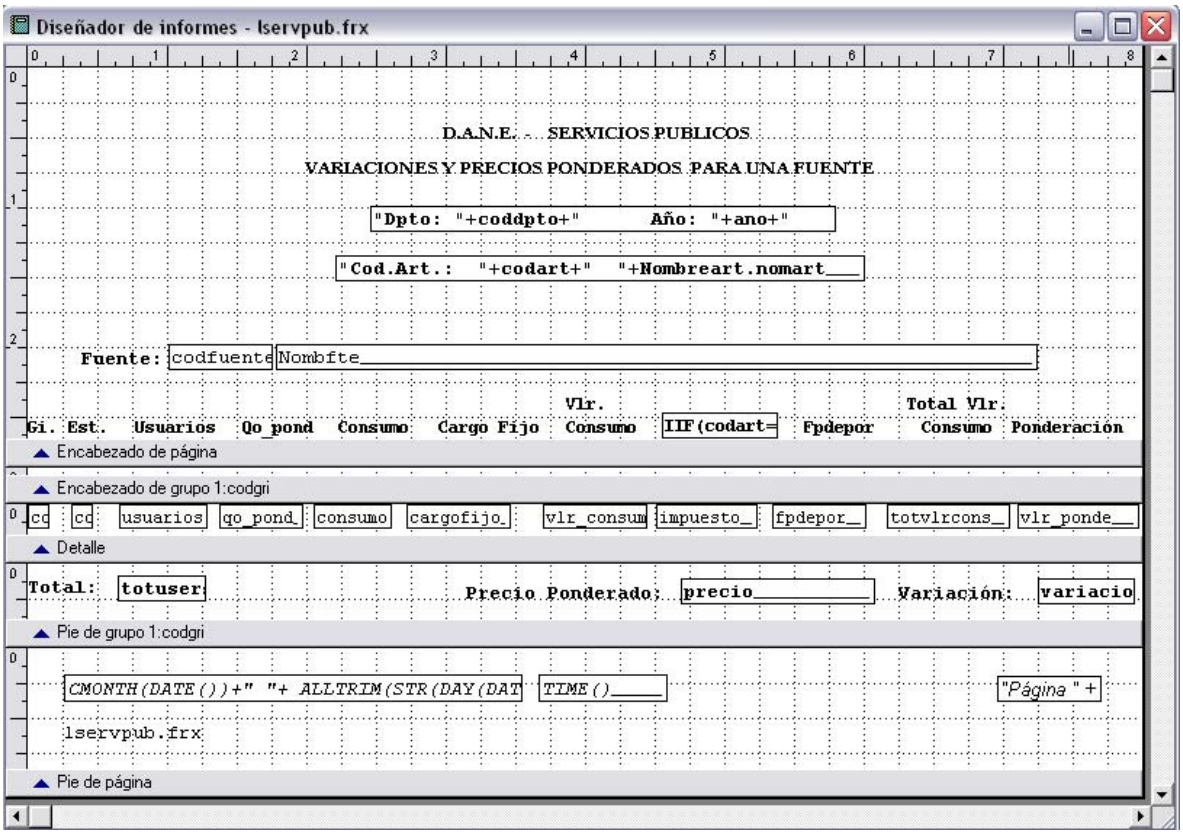
6.2 LSERVPUB3.FRX - VARIACIONES Y PRECIOS PONDERADOS PARA LAS FUENTES FINALES QUE ENTRAN AL CALCULO

Permite la impresión de los precios ponderados y la variación por estratos y total; clase de fuente, para una fuente. Disponible en el formulario de Servicios públicos.



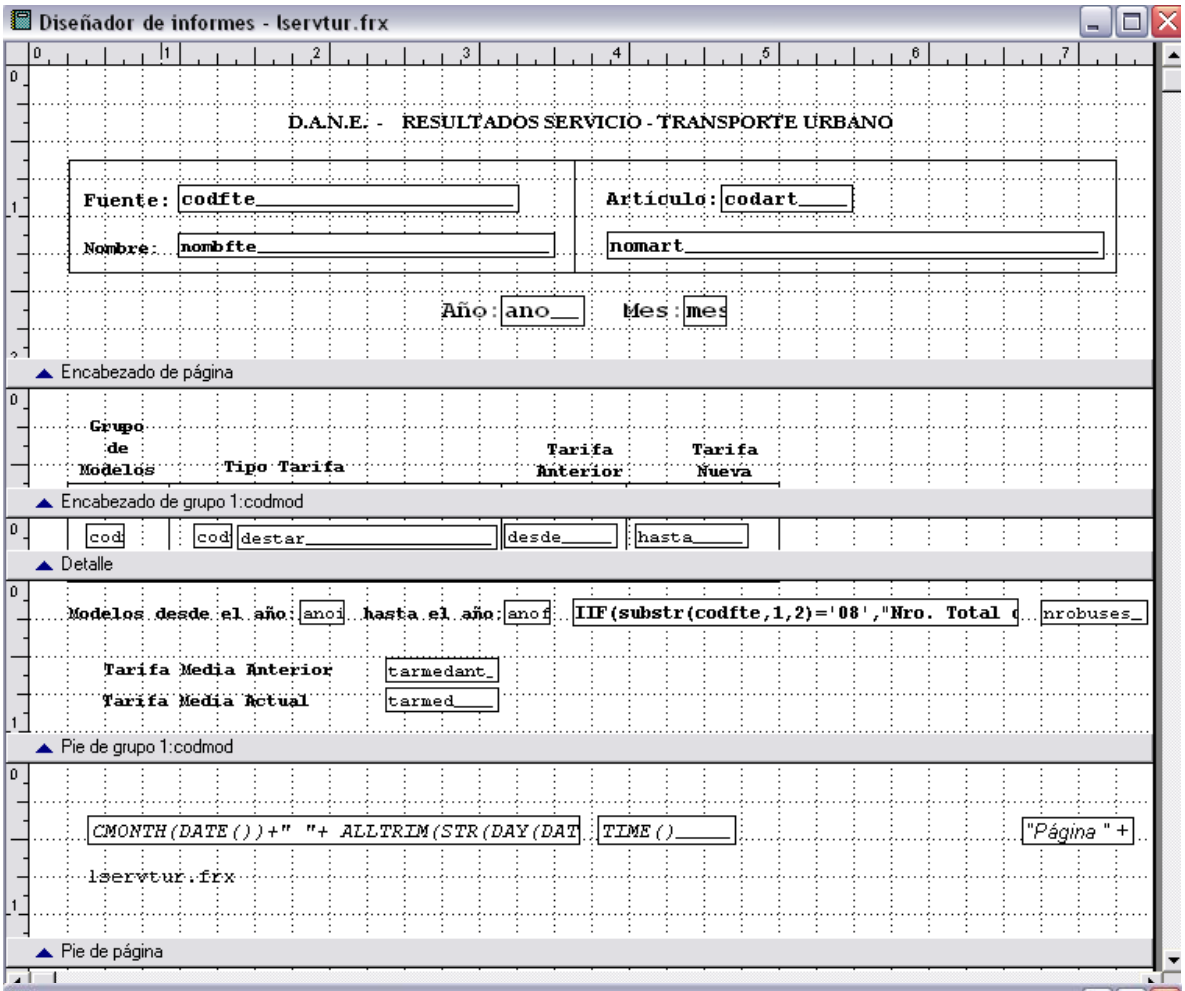
6.3 LSERVPUB.FRX - VARIACIONES Y PRECIOS PONDERADOS PARA UNA FUENTE

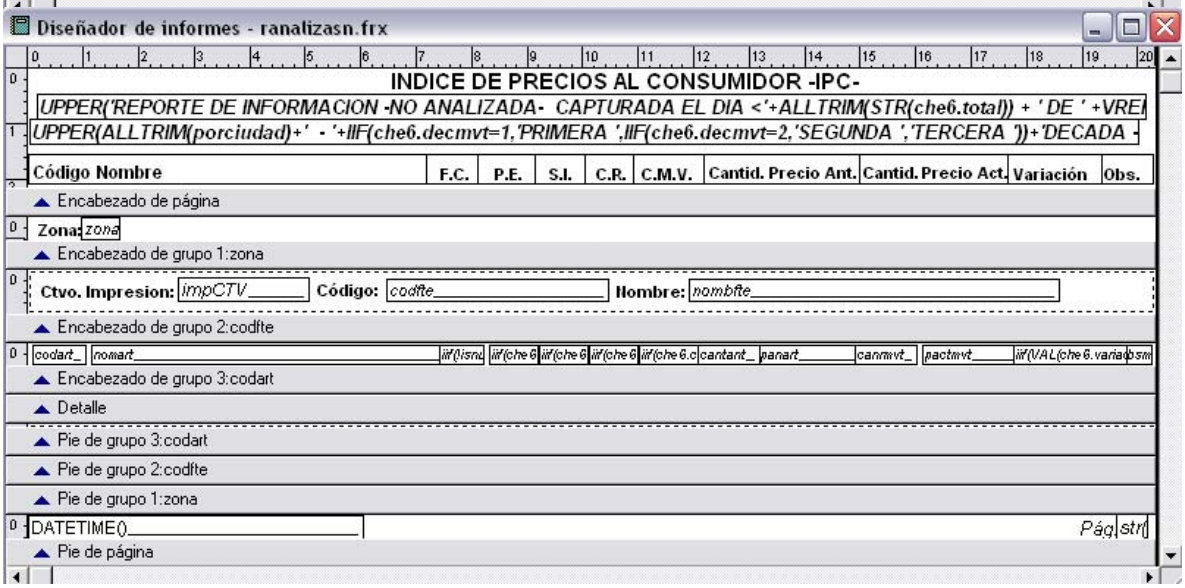
Disponible en el formulario de Servicios públicos.



6.4 LSERVTUR.FRX - RESULTADOS SERVICIO - TRANSPORTE URBANO

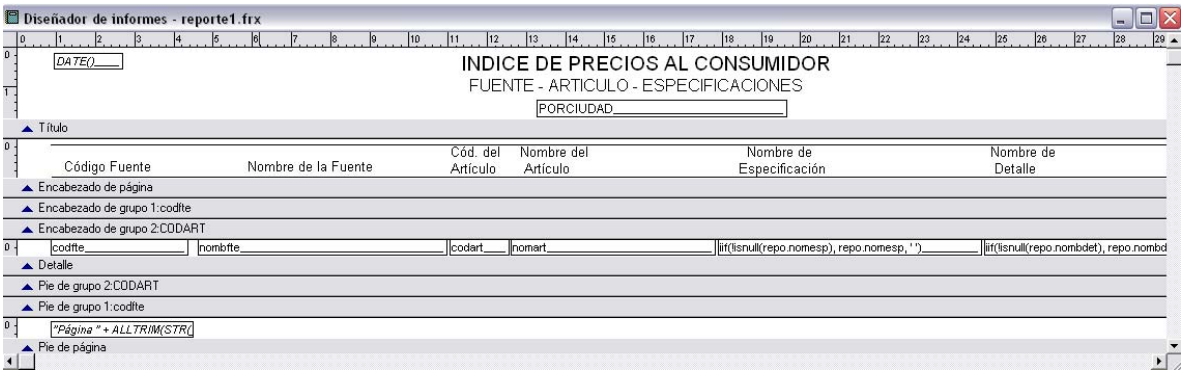
Imprime la tarifa media actual y anterior a ser incluida en transporte urbano. Disponible en el formulario de transporte urbano


Diseñador de informes - lservtur.frx
D.A.N.E. - RESULTADOS SERVICIO - TRANSPORTE URBANO
Fuente: codfte
Nombre: nombfte
Artículo: codart
nomart
Año: ano
Mes: mes
Encabezado de página
Grupo de Modelos
Tipo Tarifa
Tarifa Anterior
Tarifa Nueva
Encabezado de grupo 1:codmod
cod
cod
destar
desde
hasta
Detalle
Modelos desde el año: anoi hasta el año: anof
IIF(substr(codfte,1,2)='08',"Nro. Total
nrobuses
Tarifa Media Anterior
tarmedant
Tarifa Media Actual
tarmed
Pie de grupo 1:codmod
CMONTH(DATE())+" "+ ALLTRIM(STR(DAY(DAT
TIME()
"Página " +
lservtur.frx
Pie de página


Diseñador de informes - ranalizasn.frx
INDICE DE PRECIOS AL CONSUMIDOR -IPC-
UPPER('REPORTE DE INFORMACION -NO ANALIZADA- CAPTURADA EL DIA <'+ALLTRIM(STR(che6.total)) + ' DE ' +VREI
UPPER(ALLTRIM(porciudad)+' - '+IIF(che6.decmvt=1,'PRIMERA ',IIF(che6.decmvt=2,'SEGUNDA ','TERCERA '))+'DECADA -
Código Nombre
F.C.
P.E.
S.I.
C.R.
C.M.V.
Cantid. Precio Ant.
Cantid. Precio Act.
Variación
Obs.
Encabezado de página
Zona: zona
Encabezado de grupo 1:zona
Ctvo. Impresion: impCTV
Código: codfte
Nombre: nombfte
Encabezado de grupo 2:codfte
codart
nomart
cantant
panart
canmvt
pactmvt
Encabezado de grupo 3:codart
Detalle
Pie de grupo 3:codart
Pie de grupo 2:codfte
Pie de grupo 1:zona
DATETIME()
Pág
Pie de página

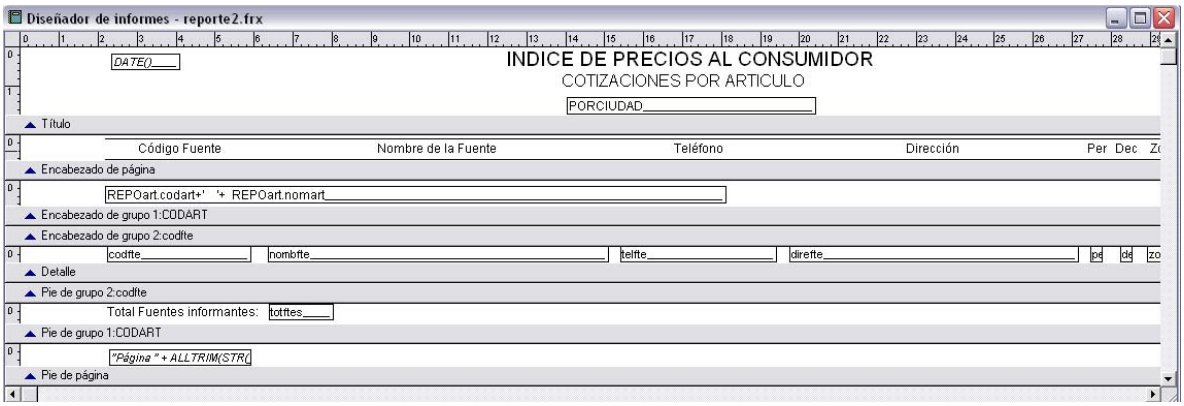
6.5 REPORTE1.FRX - FUENTE - ARTICULO – ESPECIFICACIONES

Imprime los datos de la fuente con sus articulo y especificaciones; para una fuente seleccionada; y un mes, década y zona seleccionado. Esta disponible en el formulario de reportes de canasta local.



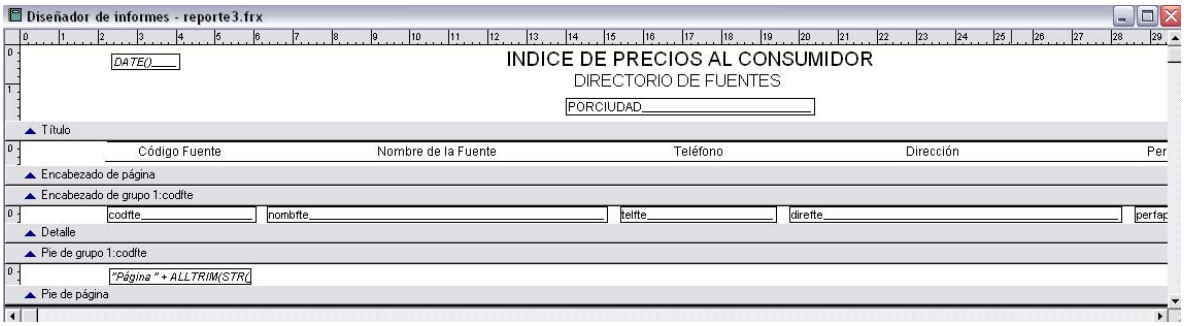
6.6 REPORTE2.FRX - COTIZACIONES POR ARTICULO

Imprime los datos de la fuentes que cotizan un articulo especifico para un mes, década y zona seleccionado por periodicidad, década y por zona. Esta disponible en el formulario de reportes de canasta local.



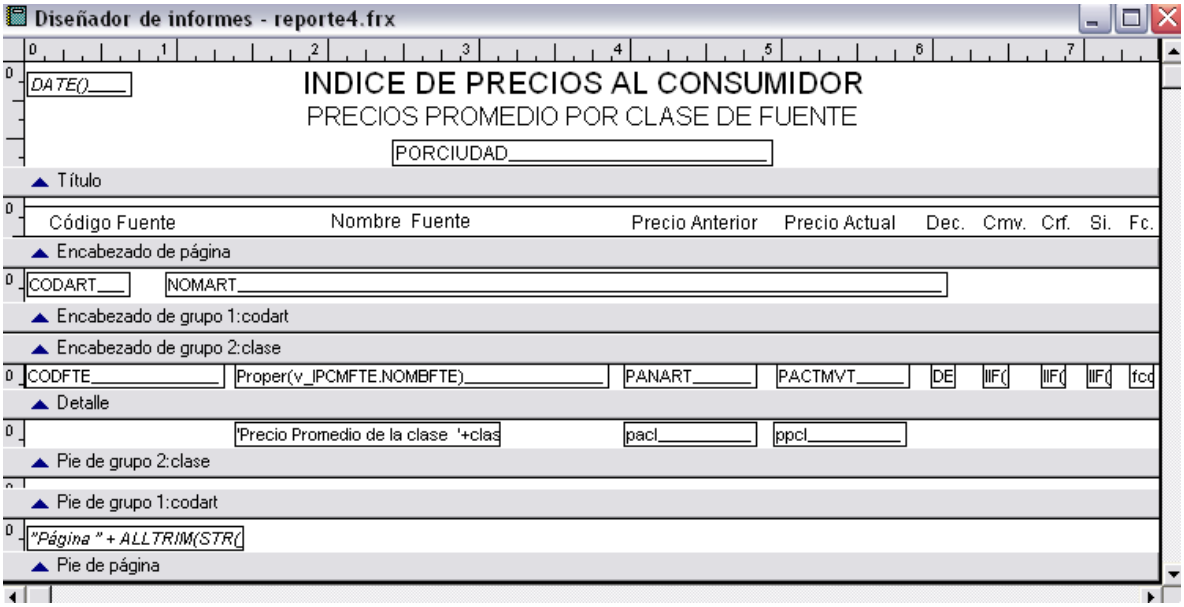
6.7 REPORTE3.FRX - DIRECTORIO DE FUENTES

Imprime los datos de la fuente, código, dirección, teléfono, periodicidad, zona y década a al que pertenece. Esta disponible en el formulario de reportes de canasta local.



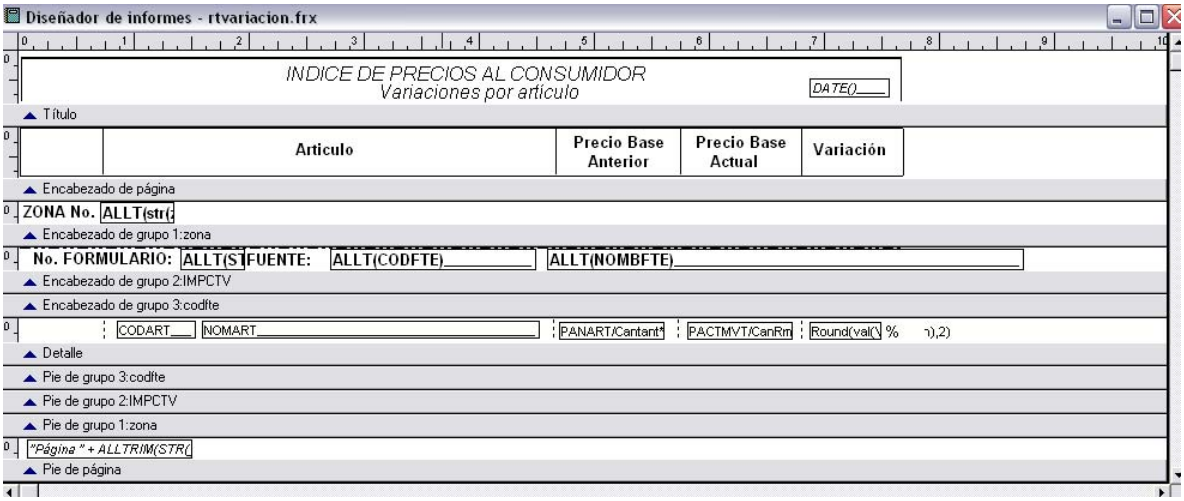
6.8 REPORTE4.FRX - PRECIOS PROMEDIO POR CLASE DE FUENTE

Permite imprimir los precios anteriores y actuales de todas las fuentes que cotizan un articulo seleccionado por década y el precio promedio por clase de fuente. Esta disponible en el formulario de reportes de canasta local.



6.9 RTVARIACION.FRX - VARIACIONES POR ARTÍCULO

Imprime la variación del precio base anterior y actual y la variación de los artículos capturados en una fuente. Disponible por década y zona en el formulario de captura mediante el botón de reportes.


Diseñador de informes - rtvariacion.frx
INDICE DE PRECIOS AL CONSUMIDOR
Variaciones por artículo
DATE()
Título
Articulo
Precio Base Anterior
Precio Base Actual
Variación
Encabezado de página
ZONA No.
Encabezado de grupo 1:zona
No. FORMULARIO:
FUENTE:
ALLT(CODFTE)
ALLT(NOMBFTE)
Encabezado de grupo 2:IMPCTV
Encabezado de grupo 3:codfte
CODART
NOMART
PANART/Cantant
PACTMVT/CanRm
Round(val(
%
),2)
Detalle
Pie de grupo 3:codfte
Pie de grupo 2:IMPCTV
Pie de grupo 1:zona
"Página " + ALLTRIM(STR(
Pie de página