Edmundo A. Cáceres

Objetos y propiedades de Visual FoxPro

2011

Índice

Introducción	1
Primera parte. Formularios	2
Capítulo 1. Objetos de formulario	3
Label	4
TextBox	4
EditBox	5
CommandButton	6
CommandGroup	6
OptionGroup y OptionButton	7
CheckBox	8
ListBox	9
ComboBox	. 12
Grid, Column y Header	. 13
Spinner	. 16
PageFrame y Page	. 16
Line	. 17
Shape	. 17
Image	. 18
Capítulo 2. Propiedades de objetos de formulario	. 19
Alignment	. 19
AllowAddNew	. 19
AllowHeaderSizing	. 19
AllowRowSizing	. 20
AutoSize	. 20
BackColor	. 20
BackStyle	. 20
BoundColumn	. 20
BoundTo	. 20
ButtonCount	. 21
	.21
ChildOrder	. 21
ColumnCount	. 22
ColumnLines	. 22
ColumnOrder	. 22
ColumnWidths	.23
ControlSource	. 23
Curvature	. 27
DeleteMark	. 27
Enabled	.27
Font	. 27
ForeColor	. 27
Format	. 27
GridLines	. 28
HeaderHeight	. 28
Height	.28
Increment	. 28
IncrementalSearch	.28
InputMask	. 29

KeyboardHighValue	
KeyboardLowValue	
Left	
LineSlant	
LinkMaster	
Movable	
Name	
PageCount	
PageOrder	
Propiedades especiales de un Grid subordinado	
Propiedades de fuente	
Propiedades de color	34
Propiedades de tamaño y ubicación	35
ReadOnly	35
RecordMark	35
RecordSource	35
RelationalEvor	
Recizable	36
PowHoight	36
Now Teight	
CorollPare	
SciontOnEntry	
Selectonentian value	
SpinnerFlighvalue	
SpinnerLowvalue	
l adindex	
1 OP	
Wiath	
Capitulo 3. Generadores de objetos de formulario	
Generador de TextBox	
Generador de EditBox	
Generador de CommandGroup	
Generador de OptionGroup	
Generador de ComboBox y ListBox	
Generador de Grid	
Segunda parte. Informes	
Capítulo 4. Bandas de informe	
Encabezado de página, Detalle, Pie de página	
Encabezado de columna, Pie de columna	51
Encabezado de grupo, Pie de grupo	52
Título, Resumen	53
Capítulo 5. Objetos de informe	54
Etiqueta	54
Сатро	54
Línea	55
Rectángulo	55
Rectángulo redondeado	
Imagen	55

Capítulo 6. Propiedades de objetos de informe	57
Expresión de informe	57
Calcular campo	58
Formato (campo)	59
Imprimir condiciones	60
Rectángulo / Línea	63
Rectángulo redondeado	63
Texto	63
Variables de informe	64
Tercera parte. Entorno de datos	66
Capítulo 7. Objetos de entorno de datos	67
DataEnvironment	67
Cursor	67
Relation	67
Capítulo 8. Propiedades de entorno de datos	68
AutoCloseTables	68
AutoOpenTables	68
InitialSelectedAlias	68
Alias	70
CursorSource	71
Filter	71
Order	71
ReadOnly	71
ChildAlias	71
ChildOrder	71
OneToMany	72
ParentAlias	72
RelationalExpr	72
Cuarta parte. Generador de expresiones. Funciones. Distribución de objetos	74
Capítulo 9. Generador de expresiones	75
Generador de expresiones	75
Opciones del Generador de expresiones	75
Capítulo 10. Algunas funciones útiles	77
Cómo probar funciones	77
Abs(expresiónN)	78
Alltrim(expresiónC)	78
Avg(expresiónN)	79
Between(expresión, límite inferior, límite superior)	79
Cdow(expresiónF expresiónH)	79
Count(expresión)	79
Cmonth(expresiónF expresiónH)	79
Date()	80
Datetime()	80
Day(expresiónF expresiónH)	80
Deleted([alias])	80
Dmy(expresiónF expresiónH)	81
Dtoc(expresiónF expresiónH)	81
Dtos(expresiónF expresiónH)	81
Empty(expresión)	82
Hour(expresiónH)	82

lif(condición, expresión por verdadero, expresión por falso)	82
Inlist(expresión, valor1, valor2, valor3, valor4)	83
Left(expresiónC, tamaño)	83
Lower(expresionC)	83
Ltrim(expresiónC)	83
Max(expresión)	83
Max(expresión1, expresión2)	84
Min(expresión)	84
Min(expresión1, expresión2)	84
Minute (expresiónH)	84
Month(expresiónF / expresiónH)	84
Padc(expresión, tamaño [, carácter de relleno])	84
Padl(expresión, tamaño [, carácter de relleno])	85
Padr(expresión, tamaño [, carácter de relleno])	85
Proper(expresiónC)	85
Recno([alias])	85
Replicate(expresiónC, veces)	86
Right(expresiónC, tamaño)	86
Rtrim(expresiónC)	86
Seek(expresión [, alias [, índice]])	86
Sec(expresiónH)	87
Space(cantidad)	88
Sqrt(expresiónN)	88
Str(expresiónN [, tamaño [, decimales]])	88
Substr(expresiónC, inicio [, tamaño])	88
Sum(expresiónN)	89
Time()	89
Transform(expresión, formato)	89
Upper(expresiónC)	89
Year(expresiónF expresiónH)	89
Capítulo 11. Distribución de objetos	91
Barra de herramientas Distribución	91
Cuadrícula	97
Menú Formato	98
Cuarta parte. Ejercitación1	02
Capítulo 12. Ejercicios con formularios 1	04
Capítulo 13. Ejercicios con informes1	34

Introducción

Visual FoxPro, en adelante VFP, posee dos poderosos recursos para mantener las tablas actualizadas y producir listados: formularios e informes. Los formularios son herramientas de entrada y salida: de entrada porque permiten modificar el contenido de los campos, eliminar registros existentes y crear registros nuevos; de salida porque muestran el contenido de los campos en pantalla. Los informes son exclusivamente de salida, porque muestran el contenido de los campos en pantapapel impreso o en pantalla.

Al crear un formulario o un informe, inicialmente ambos aparecen vacíos. Para introducirles campos de datos, textos, recuadros, etc., se usan *objetos*. Hay varios tipos de objetos, cada uno de los cuales tiene una apariencia y una funcionalidad diferentes. Por ejemplo, en los formularios es posible crear etiquetas, cuadros de texto, cuadrículas o botones de comando; en los informes es posible crear etiquetas, campos o líneas.

Entre los objetos de formulario, varios muestran el contenido de uno o varios campos y permiten modificarlo. Así, si un formulario tiene un cuadro de texto que muestra el nombre de proveedor contenido en un campo del registro actual, si escribimos otro nombre en el cuadro de texto, lo que hacemos realmente es cambiarlo en el campo. Otros objetos no muestran contenido de campos, sino que son elementos auxiliares, como textos, líneas o rectángulos.

Los objetos tienen *propiedades*. Por ejemplo, el alto y ancho de un objeto, su posición, el tipo de borde, el tamaño de la letra que emplea, el campo que representa, etc., son propiedades. La mayoría de las propiedades puede ser modificadas.

Los formularios e informes generalmente toman los datos que necesitan de tablas o vistas. El grupo de tablas, vistas y relaciones entre ellas constituyen su entorno de datos. Este entorno puede ser preparado como un contenedor asociado al informe o formulario, de modo que, al ejecutar éstos, automáticamente se abren las tablas y vistas y se establecen las relaciones, sin necesidad de comandos de programación.

Las tablas y vistas no son la única fuente de datos de formularios e informes. También puede ser variables de memoria, funciones como *date(*), etc.

Esta obra abarca cinco Partes. La Primera Parte trata de formularios. La Segunda se ocupa de informes. La Tercera versa sobre entornos de datos. La Cuarta expone el generador de expresiones y algunas funciones útiles. La Quinta ejercita al lector con formularios e informes.

Primera parte. Formularios

Un formulario sirve para ver y modificar datos de tablas, vistas y variables. Para ello usa *objetos*. Al crear un objeto, VFP recurre a una definición, llamada *clase*. Cada tipo de objeto tiene su clase propia. Entonces, clase es la definición general de un tipo de objeto, mientras que objeto es lo concretamente generado en un formulario a partir de ella. VFP trae un conjunto de clases, llamadas *clases básicas*. A partir de ellas, el usuario puede generar no sólo objetos, sino también otras clases. Las *clases derivadas* de las básicas se guardan en *bibliotecas de clases*, que son archivos con extensión VCX.

Cada clase tiene *propiedades*, que trasmite a los objetos que genera. Esto se conoce como *herencia*. Si el usuario cambia posteriormente las propiedades que quiera en un objeto, éste mantiene heredadas todas las propiedades de su clase, excepto las modificadas específicamente por él. El cambio se puede realizar en *tiempo de diseño* o en *tiempo de ejecución*.

Las propiedades no pueden ser modificadas en las clases básicas; pero sí en las clases derivadas. De estas clases pueden derivarse otras, y así. Si se crea un objeto a partir de una clase derivada, el objeto hereda las propiedades modificadas en su clase generadora y las no modificadas de la clase generadora de ella. No hereda las propiedades modificadas en el objeto mismo.

La herencia es dinámica. Supongamos crear un objeto a partir de una clase derivada y modificar en él la propiedad B. Posteriormente modifiquemos las propiedades A, B y C en tal clase. Aunque es preexistente a la modificación, el objeto hereda las nuevas propiedades A y C. No hereda la propiedad B, porque ya tenía un valor propio en ella.

Además de propiedades, las clases y objetos tienen *eventos* y *métodos*, que también se heredan. Ejemplos de eventos son dar clic o doble clic en un objeto. Ejemplos de métodos son refrescar un objeto o hacer que adquiera enfoque. En esta obra no trataremos clases, eventos ni métodos, por lo que hemos dado una muy breve idea.

Esta Parte está compuesta por los Capítulos 1 a 3. El Capítulo 1 se dedica al estudio de objetos de formularios y el Capítulo 2 a propiedades de tales objetos. El Capítulo 3 se incorpora para describir en detalle el uso de generadores de objetos de formularios, mediante el cual se establecen varias propiedades, facilitando la tarea.

Capítulo 1. Objetos de formulario

Los formularios son una herramienta muy útil de VFP, porque facilitan ventajosamente la tarea de programar la entrada y la salida de datos de un sistema. Ya no es necesario escribir *comandos* y *funciones*, como se hacía antes, para lograr esos propósitos. Aunque en muchos casos hace falta que un botón de comando, por ejemplo, invoque la ejecución de programación oculta, el monto total de comandos y funciones necesitados se reduce drásticamente. La ventaja que representan los formularios, empero, supone conocer bien los objetos, propiedades, métodos y eventos. Los objetos y propiedades, categorías referentes al diseño de formularios, son relativamente fáciles para un novato. Los métodos y eventos, nuevas categorías de programación que se agregan a los comandos y funciones tradicionales, requieren un estudio más extenso y escapan al alcance de este trabajo.

Para crear un nuevo formulario se usa ARCHIVO – NUEVO – FORMULARIO; o se pulsa el botón NUEVO, estando en la categoría FORMULARIOS del ADMINISTRADOR DE PROYECTOS. También se puede usar el comando *modify form* en la ventana COMANDOS.

Para crear un objeto dentro del formulario hay varias posibilidades:

- Seleccionar el objeto deseado en la barra de herramientas CONTROLES DE FORMULARIOS y luego dar clic en el formulario o hacer un arrastre por él. En el primer caso, el tamaño del objeto está predeterminado; en el segundo, el tamaño queda establecido por el arrastre.
- Arrastrar al formulario el campo deseado desde un cursor del entorno de datos. El tipo de objeto creado depende del tipo de datos del campo:
 - Un campo numérico, de fecha o de caracteres genera un TEXTBOX;
 - Un campo lógico genera un CHECKBOX;
 - Un campo memo genera un EDITBOX.
 - Si se selecciona más de un campo y se los arrastra al formulario, se genera un GRID con tantas columnas como campos seleccionados, cuyos tipos dependen del tipo de datos de cada uno.
 - Si se arrastra el subtítulo *Campos* que aparece al comienzo del cursor, se supone selección de todos los campos, creándose un objeto para cada uno, cuyos tipos dependen de sus tipos de datos.
 - Si se arrastra el título de la ventana del cursor, se crea un GRID con todos los campos.
- Emplear FORMULARIO FORMULARIO RÁPIDO. Esto presenta el cuadro de diálogo GENERADOR DE FORMULARIOS, con dos fichas. La primera sirve para elegir la tabla y los campos necesitados. La segunda se usa para elegir un estilo.
- Emplear la herramienta GENERADOR DE FORMULARIOS de la barra de herramientas DISEÑADOR DE FORMULARIOS. Esto presenta el cuadro de diálogo GENERADOR DE FORMULARIOS.

Hay dos tipos de objetos: *contenedores* y *controles*. Contenedor es un objeto que incluye otros objetos en su interior. Son contenedores los objetos FORM, PAGE, PAGEFRAME, GRID, COLUMN, OPTIONGROUP y COMMANDGROUP. Control es un objeto que no incluye objetos. Son controles los objetos TEXTBOX, EDITBOX, LABEL, COMBOBOX, LISTBOX, SPINNER, OPTIONBUTTON, COMMANDBUTTON, LINE y SHAPE.

En este capítulo trataremos los objetos de uso más frecuente.

Label

A Label (etiqueta, rótulo) es un objeto que permite incorporar texto aclaratorio a un formulario, similar a los textos preimpresos de un formulario de papel. Así, por ejemplo, en el formulario de la Figura 1.1, son etiquetas los textos *Fecha*, *Desde* y *Hasta*. *Listado de ventas* es un texto componente del objeto FORM; *Imprimir* y *Salir* son textos componentes de botones de comando, y *Resumen* es un texto componente de una casilla de verificación.

Listado de ventas	
Fecha	
Desde	Imprimir
Hasta	Salir
Resumen	

Figura 1.1. Etiquetas y otros objetos con texto

La propiedad más importante de una etiqueta es CAPTION, que es donde se define el texto.

Todo objeto debe tener un nombre que lo identifique, es decir, debe ser único, no repetido. Este nombre sirve para hacer referencia a un objeto determinado en la programación de eventos y métodos. VFP inventa inicialmente un nombre a todo objeto que se crea, siguiendo una convención, que escribe en la propiedad NAME de cada uno. Como las etiquetas son objetos, sufren este proceso.

Si las etiquetas son creadas desde la barra de herramientas CONTROLES DE FORMULARIOS, la propiedad NAME de la primera toma el valor *Label1*; la segunda el valor *Label2*; la tercera el valor *Label3*, etc. Inicialmente, VFP también usa como valor de la propiedad CAPTION el nombre que usa para NAME. Obviamente, como estos textos no aclaran nada, deben ser cambiados por otros más adecuados. Por ejemplo, las etiquetas del formulario anterior, al ser creadas, fueron *Label1*, *Label2* y *Label3*. Esto serviría de muy poco para el usuario: por eso fueron cambiadas por *Fecha*, *Desde* y *Hasta*.

Si se arrastra un campo desde el entorno de datos, se coloca junto a él una etiqueta cuyo NAME es el prefijo *lbl* más el nombre del campo. Para la propiedad CAPTION se sigue una lógica mejor: si el campo en la tabla no tiene definido un título, se toma como valor de CAPTION el nombre del campo; pero si tiene definido un título, normalmente más explicativo, se toma a éste como el valor de tal propiedad. Lo anterior se sigue parcialmente con los campos lógicos, que generan un CHECKBOX: no se genera una etiqueta, sino un texto dentro del objeto; este texto es el título, si lo hay, o el nombre del campo, si no lo hay.

Los nombres creados automáticamente por VFP pueden ser cambiados por otros que sean más significativos para el programador.

TextBox

TEXTBOX (cuadro de texto) es un objeto que permite modificar el contenido de un campo o de una variable. Su aspecto es un área rectangular, normalmente con borde, como los rectángulos en blanco que aparecen a la derecha de las etiquetas *Desde* y *Hasta* en la Figura 1.1. El tamaño del rectángulo debería ser diseñado de acuerdo al tamaño del campo o variable que va a mostrar en tiempo de ejecución. Si es más corto, se producirá un desplazamiento horizontal a medida que se avanza por el contenido del campo. Si es más largo, la porción sobrante no se podrá editar.

En tiempo de ejecución, el cuadro de texto mostrará el valor que almacena el campo del registro actual o de la variable. Si tal contenido se modifica mediante el teclado, el nuevo valor se almacenará en ese campo o variable, reemplazando el contenido anterior. El almacenamiento se producirá cuando el cuadro de texto pierda el enfoque, es decir, cuando se produzca el evento LOSTFOCUS. Éste evento ocurre cuando se hace clic en otro objeto o se va al objeto siguiente o anterior con TAB, ENTER, SHIFT + TAB, etc.

La propiedad más importante de un TEXTBOX es CONTROLSOURCE, que sirve para definir cuál es el campo o la variable que se verá a través de él. Otras propiedades útiles son FORMAT e INPUTMASK, porque permiten "maquillar" el valor del campo o variable, e incluso controlar el nuevo valor que se pueda introducir por teclado.

Si los cuadros de texto son creados desde la barra de herramientas CONTROLES DE FORMULARIOS, el valor inicial de la propiedad NAME es *Text1*, *Text2*, *Text3*, etc., donde el número indica el orden de creación de estos objetos.

Si se arrastra un campo numérico, de caracteres, de fecha o de fecha-hora desde el entorno de datos, la propiedad NAME es el prefijo *txt* más el nombre del campo. Como se dijo antes, también se coloca junto a él una etiqueta cuyo NAME es el prefijo LBL más el nombre del campo y cuyo CAPTION es el nombre del campo si no hay título de campo, o el título de campo, si existe.

Un cuadro de texto sirve para cualquier tipo de campo o variable. Para campos lógicos, no obstante, resulta más apropiado el empleo de casillas de verificación. Para campos memo, es mejor usar cuadros de edición.

EditBox

El objeto EDITBOX (cuadro de edición) permite modificar el contenido de un campo de tipo memo con más facilidad que un TEXTBOX. Tiene el aspecto de un área rectangular, normalmente con un borde. Como un campo de este tipo puede ser tan largo como se desee, ya que no se almacena en la tabla sino en un archivo de texto asociado a ella, el tamaño con que se trace el cuadro de edición puede resultar insuficiente para el contenido. Por ello, este objeto puede mostrar una barra de desplazamiento vertical.

Un cuadro de edición puede servir también para campos o variables de caracteres. Si tuviéramos un campo de caracteres de 160 posiciones, por ejemplo, y el espacio disponible en el formulario fuera poco, podríamos usar un cuadro de edición que diera cabida a tres líneas de unos 20 caracteres cada una, con la posibilidad de desplazamiento vertical. Lo mismo sucede para variables de caracteres. Las variables de caracteres, a diferencia de los campos, no tienen longitud definida, sino que se hacen tan grandes como lo que van a contener. Por ello, si vamos a usar una variable de este tipo que pueda alcanzar un contenido grande, podríamos recurrir a este tipo de objeto, en lugar de usar un cuadro de texto.

Tal como dijimos para cuadros de texto, la propiedad más importante es CONTROLSOURCE, porque define el campo o la variable de los cuales se tomará el contenido inicial y donde se almacenarán los cambios que se realicen.

Si los cuadros de edición se crean desde la barra de herramientas CONTROLES DE FORMULARIOS, el valor inicial de la propiedad NAME es *Edit1*, *Edit2*, *Edit3*, etc., donde el número indica el orden en que se van creando estos objetos.

Si se arrastra un campo memo desde el entorno de datos, la propiedad NAME será el prefijo *edt* más el nombre del campo. Como se dijo, también se coloca junto a él una etiqueta cuyo NAME es el prefijo *IbI* más el nombre del campo y cuyo CAPTION es el nombre del campo, si no hay título de campo, o el título de campo, si existe.

CommandButton

El objeto COMMANDBUTTON (botón de comando) hace que se ejecute un procedimiento. Tiene el aspecto de un rectángulo sólido con un texto interior. La Figura 1.2 da ejemplos de estos botones.



Figura 1.2. Botones de comando

Un procedimiento es un conjunto de métodos, comandos o funciones para lograr un propósito, es decir, es un subprograma contenido en el formulario, que es también un programa. El estudio de estos temas lleva bastante tiempo, porque es muy amplio. Brevemente, consisten en lo siguiente:

- Los métodos son órdenes propias de la programación visual de VFP. Por ejemplo, *Refresh, Release* o *SetFocus*.
- Los comandos son órdenes propias de la programación tradicional de FoxPro. Por ejemplo, use, display memory, go o delete.
- Las funciones se usan dentro de los comandos. Devuelven resultados de acuerdo a los parámetros que se incluyen entre los paréntesis. Por ejemplo, *between()*, *dtoc()* o *str()*.

A menudo, un formulario por sí solo es insuficiente para producir lo que se desea, por lo que se requiere incluir más programación dentro de él. Esta programación se escribe formando procedimientos. Cada procedimiento se asocia a un evento dado de un objeto determinado, de modo que un formulario puede tener varios procedimientos, si se necesitan. Una forma común de procedimiento es el que se asocia al evento *click* de un botón de comando.

La propiedad más usada de un botón de comando es CAPTION, que indica qué va a producir cuando se le dé clic.

El valor inicial de la propiedad NAME de los botones de comando es *Command1*, *Command2*, *Command3*, etc., donde el número indica el orden de creación. Estos valores son también usados inicialmente para la propiedad CAPTION. Obviamente, por razones de claridad el valor de ambas propiedades se debe cambiar.

CommandGroup

COMMANDGROUP (grupo de botones de comando) es un objeto contenedor. El contenido son botones de comando. Su cometido es encerrar botones de comando que están relacionados en propósito. Por ejemplo, la Figura 1.3 muestra dos grupos de botones de comando, el primero con acciones referentes a la navegación por una tabla y el segundo a la impresión de un informe.

Primero	
Anterior	Imprimir Vista previa
Siguiente	
Último	

Figura 1.3. Grupos de botones de comando

El que los botones de comando vayan encerrados en el grupo no tiene otro fin que enfatizar el propósito similar o sencillamente porque puede resultar lindo. El grupo no produce otra cosa. Si se prefiere, se puede usar los botones sin agruparlos, pues los que invocan procedimientos asociados son éstos.

Las formas más comunes de disponer los botones de comando son uno bajo el otro o uno junto al otro, aunque la disposición puede ser cualquiera, como por ejemplo en diagonal. No existe propiedad relacionada con esta característica. Para mover los botones se debe: (1) seleccionar el grupo y darle valor *Falso* a su propiedad AUTOSIZE; (2) variar sus dimensiones para tener espacio suficiente; (3) seleccionar cada botón y variar su posición y dimensiones.

Al crear un grupo, la cantidad de botones de comando contenidos es de dos y la disposición es vertical. La cantidad puede variarse modificando la propiedad BUTTONCOUNT.

El generador de este objeto facilita establecer cuántos botones se desean, cuál es el espacio entre ellos, cuál CAPTION debe llevar cada uno y cómo se los va a disponer. El generador calcula dos disposiciones: un botón de comando bajo el otro o uno junto al otro.

El valor inicial de la propiedad NAME de los grupos de botones de comando es *CommandGroup1*, *CommandGroup2*, *CommandGroup3*, etc., donde el número indica el orden de creación. Los botones contenidos tienen los nombres *Command1*, *Command2*, etc., numeración que empieza por 1 en cada grupo.

OptionGroup y OptionButton

OPTIONGROUP (grupo de opciones) es un objeto contenedor. Tiene el aspecto de un rectángulo que encierra una lista de opciones. Cada opción, objeto OPTIONBUTTON, lleva un botón que puede estar activo o inactivo y un texto aclaratorio. Las opciones son mutuamente excluyentes, es decir, solamente una puede estar activa en un momento dado. Si se activa otra, se desactiva la que estaba activa. Vea el ejemplo de la Figura 1.4.



Figura 1.4. Grupos de opciones

A diferencia de los botones de comando, que pueden existir libremente o agrupados, los objetos OPTIONBUTTON solamente pueden existir dentro de un OPTIONGROUP. Por ello los tratamos juntos.

Las formas más comunes de disponer las opciones es una bajo la otra o una junto a la otra, aunque la disposición puede ser cualquiera. No existe propiedad relacionada con esta característica. Para mover los botones se debe proceder como con el grupo de botones de comando: (1) seleccionar el grupo y hacer *Falso* su propiedad AUTOSIZE; (2) variar sus dimensiones para tener espacio suficiente; (3) seleccionar cada botón y variar su posición y dimensiones.

Al crear un grupo, la cantidad de opciones es de dos y la disposición es una bajo otra. La cantidad de opciones puede ser variada, modificando la propiedad BUTTONCOUNT. Cada botón tiene un número asociado, que le otorga internamente VFP.

El generador de este objeto facilita establecer cuántas opciones se desean, cuál es el espacio entre ellas, cuál CAPTION debe llevar cada una y cómo van a disponerse. El generador calcula dos disposiciones: una

opción bajo la otra o una junto a la otra.

Una propiedad decisiva de un grupo de opciones es CONTROLSOURCE. Establece de dónde se toma el valor que se traduce inicialmente como botón activo. El valor es el contenido de un campo o variable, que deben ser de tipo numérico o de caracteres.

En el ejemplo anterior, sea que CONTROLSOURCE es el campo ESTADO, de tipo numérico. Las opciones *Soltero*, *Casado* y *Viudo* tienen, respectivamente, los valores internos otorgados por VFP 1, 2 y 3. Si en el campo ESTADO hubiera un 2, la opción seleccionada inicialmente en el grupo sería la segunda, que corresponde a *Casado*. Si hubiera un 3, la opción seleccionada inicialmente sería la tercera, que corresponde a *Viudo*.

CONTROLSOURCE establece también que tal campo o variable almacenarán cualquier cambio realizado en el grupo de opciones. Por ejemplo, si se da clic en la primera opción, que corresponde a *Soltero*, se guardará un *1* en el campo ESTADO.

Supongamos ahora que el campo ESTADO fuera de caracteres y contuviera el valor *Casado*. En ese caso, inicialmente aparecería activada la segunda opción. Si activásemos la primera, en ESTADO se guardaría el valor *Soltero*.

Puede inferirse que los valores admisibles para ESTADO, si fuera numérico, serían 1, 2 y 3. Si fuera de caracteres, serían *Soltero*, *Casado* y *Viudo*. Esto explica por qué las opciones son mutuamente excluyentes, pues el valor del campo ESTADO debe ser uno de los tres, no los tres.

Si con anterioridad al formulario el campo ESTADO fuera numérico y con un valor 8, inicialmente no aparecería ninguna opción marcada. Lo mismo acontecería si fuera de caracteres y con el valor *Divorciado*. Ambas situaciones se darían porque los valores no corresponden a ninguna opción. Esto no impide que, si se activa una de las opciones del grupo, se almacene en el campo el correspondiente valor elegido.

El texto de cada botón, que se define en la propiedad CAPTION, puede ir a la izquierda o a la derecha de él. Esta colocación se maneja con la propiedad ALIGNMENT de cada OPTIONBUTTON.

El valor inicial de la propiedad NAME de los grupos de opciones es *OptionGroup1*, *OptionGroup2*, *Option-Group3*, etc., donde el número indica el orden de creación. Las opciones contenidas tienen los nombres *Option1*, *Option2*, etc., comenzando por 1 en cada grupo, que también se toman inicialmente como el valor CAPTION de cada opción.

CheckBox

CHECKBOX (cuadro o casilla de verificación) es un objeto que representa un campo o una variable de tipo lógico. Este tipo de datos admite los valores *Verdadero* o *Falso*. Por ello, el objeto es una casilla, acompañada por un texto explicativo. La casilla puede aparecer con o sin tilde. Si está tildada, representa *Verdadero*; si no lo está, representa *Falso*. La tilde se pone o quita dando clic en el objeto. Inicialmente, si el campo o variable vale *Verdadero*, aparece con tilde; si vale *Falso*, aparece sin ella. Si se cambia el valor en el objeto, se cambia en el campo o variable. La Figura 1.5 da ejemplos de cuadros de verificación.

Trabaja Vi	ve 🗌
------------	------

Figura 1.5. Cuadros de verificación

El campo o la variable de donde se toma el valor inicial y donde se almacena el cambio de valor se define en CONTROLSOURCE. El texto asociado a cada casilla se define en CAPTION. La ubicación del texto a la izquierda o derecha de la casilla se establece en ALIGNMENT.

El valor inicial de la propiedad NAME de las casillas de verificación es *Check1*, *Check2*, *Check3*, etc., donde el número indica el orden de creación de estos objetos. Estos nombres también son usados inicialmente como valor *Caption* de cada casilla.

Si se arrastra un campo lógico del entorno de datos al formulario, el objeto tendrá un nombre compuesto por el prefijo *chk* seguido por el nombre el campo. El valor de CAPTION será el título del campo, si existe, o el nombre del campo, si no existe.

ListBox

LISTBOX (cuadro de lista) es un objeto que presenta opciones, pocas, muchas o muchísimas, de las cuales se puede elegir una. Las opciones van una bajo la otra, como filas o líneas. Dado que el tamaño vertical del objeto puede ser insuficiente para mostrar todas las opciones, posee una barra de desplazamiento vertical. Es un objeto control, de funcionamiento complejo.

Las opciones pueden tener una o más columnas. Las Figuras 1.6 y 1.7 muestran ejemplos de cuadros de lista con una y tres columnas.



Figura 1.6. Cuadro de lista con una columna

Puerta de aluminio	210	D20	
Puerta de cedro	380	D75	
Ventana de aluminio	275	C19	
Ventana de chapa	190	C15	
Ventana de roble	230	C08	▼

Figura 1.7. Cuadro de lista con tres columnas

La cantidad de columnas se define en la propiedad COLUMNCOUNT. Así, COLUMNCOUNT para el cuadro de lista de la Figura 1.6 es *1* y para el de la Figura 1.7 es 3.

Las opciones pueden tener distintos orígenes, es decir, se pueden "fabricar" de distintas formas. Los orígenes más comunes son un listado que se introduce manualmente en el control o un listado que VFP prepara a partir de campos de una tabla. ¿Cuándo conviene elegir uno u otro?

Si queremos introducir como opciones los nombres de los meses, es fácil escribir una lista de doce nombres separados por comas. Pero si queremos introducir como opciones los artículos que vendemos, supongamos unos tres mil, escribir un listado con tantos nombres sería muy difícil, con gran posibilidad de errores y nos faltaría espacio. En este caso, como seguramente ya tenemos una tabla con los nombres, precios, etc., de nuestros artículos, podemos usarla como origen de la lista de opciones. Esta solución es fácil y evita errores.

La propiedad ROWSOURCETYPE establece cuál es el origen de las filas. Si éstas son construidas con valo-

res introducidos a mano en el cuadro de lista, ROWSOURCETYPE debe valer 1. Si son construidas con campos de una tabla, ROWSOURCETYPE debe valer 6.

Cuando ROWSOURCETYPE es 1, otra propiedad, ROWSOURCE, debe incluir textualmente el listado de valores. Cuando ROWSOURCETYPE es 6, ROWSOURCE debe incluir el nombre de la tabla y los nombres de los campos, siguiendo el formato **Tabla. Campo[,Campo]...** La tabla se separa del primer campo por un punto. Si hay varios campos, deben ir separados por comas.

Supongamos que el cuadro de lista de la Figura 1.6 tiene su origen en un listado textual. En este caso, COLUMNCOUNT debe valer 1, ROWSOURCETYPE debe valer 1 y ROWSOURCE debe ser

Buenos Aires, Catamarca, Corrientes, Córdoba, Mendoza, Salta

Si el mismo cuadro tiene su origen en el campo NOMBRE de la tabla PROVINCIAS, COLUMNCOUNT deberá ser 1, ROWSOURCETYPE 6 y ROWSOURCE

Provincias.Nombre

Supongamos que el cuadro de lista de la Figura 1.7 tiene su origen en un listado textual. En este caso, COLUMNCOUNT debe valer 3, ROWSOURCETYPE debe valer 1 y ROWSOURCE debe ser

Puerta de Aluminio, 210, D20, Puerta de cedro, 380, D75, Ventana de aluminio, 275, C19, etc.

Note que primero se completan los componentes de una opción, columna a columna, luego los de la siguiente y así.

Si el mismo cuadro se origina en los campos NOMBRE, PRECIO y CÓDIGO de la tabla ARTÍCULOS, COLUMNCOUNT debe ser 3, ROWSOURCETYPE debe ser 6 y ROWSOURCE debe ser

Artículos. Nombre, Precio, Código

Cuando hay varias columnas, es posible mostrar líneas entre ellas. Para esto se emplea la propiedad COLUMNLINES. Si COLUMNLINES es *Falso*, no se verán líneas entre las columnas; si es *Verdadero*, se verán.

A diferencia de GRID, que es un contenedor de objetos COLUMN, LISTBOX es un control, por lo que no tiene objetos contenidos. En GRID cada COLUMN define su propio ancho en la propiedad WIDTH. En LISTBOX se usa un listado de anchos en la propiedad COLUMNWIDTHS. Si hay varios anchos, se los debe separar por comas. Un caso especial es ancho 0: si una columna tiene ancho 0, se hace invisible. Aunque sea invisible, la columna sigue estando presente en el cuadro de lista.

LISTBOX tiene varias aplicaciones, siendo dos las más características:

- La primera aplicación es usar una tabla que origine las opciones, tengan éstas una o más columnas. Cuando se usa una tabla como origen de las opciones, al elegir una opción, VFP pone el puntero de registros de esa tabla en el registro que origina tal opción. Este cambio indirecto del registro actual a través del LISTBOX permite, por ejemplo, mostrar en un GRID subordinado los registros de otra tabla correspondientes al registro elegido indirectamente. Más adelante veremos qué es un GRID subordinado. Las Figuras 1.8 y 1.9 ejemplifican lo dicho.
- La segunda aplicación consiste en usar un LISTBOX, construido desde una tabla o con valores introducidos a mano, para que, cuando se elija una opción, VFP guarde un valor componente de esa opción (el de la primera, segunda, tercera, etc., columna) en un campo de otra tabla (en el registro actual) o en una variable. Por ejemplo, para llenar el campo CLI del registro que se está añadiendo a la tabla ENCFAC, se puede usar un LISTBOX construido con los campos NOM y CLI de la tabla CLI. Al elegir un artículo en el LISTBOX, se guarda el componente de la segunda columna, tomado del campo CLI de la tabla CLI, para llenar el campo CLI de la tabla ENCFAC. Esto se muestra en las Figuras 1.10.





Figura 1.8. ListBox sin cambio de opción



En el LISTBOX se elige la opción Escritorio.

El registro actual de la tabla origen es el de la tercera fila.

Este "salto" de registro es automático, sincronizado con lo que pasa en el ListBox.

Figura 1.9. Cambio de opción en el ListBox: cambio de registro actual de la tabla origen

	Tabla Encfac			
	Fac	Fec	Cli	Ven
	00000125	10-08-2007	17	4
	00000126	10-08-2007	25	7
•	00000127	10-08-2007		

Originado en la tabla CLI		
Furlani, Marisa	1200	
García, Luis	258	
Gutiérrez, César	422	
Guzmán, Abel	17	
Hernández, José	567	
Hierrezuelo. Ana	784	▼

Si se elige la opción *Hernández, José*, en el registro que estamos llenando de ENCFAC.CLI se guardará 567, valor de la segunda columna del LISTBOX. La tabla ENCFAC quedará:

Fac	Fec	Cli	Ven
00000125	10-08-2007	17	4
00000126	10-08-2007	25	7
00000127	10-08-2007	567	

Figura 1.10. ListBox con ControlSource definido como campo de una tabla

Para la segunda aplicación hay tres propiedades decisivas.

- 1. La primera propiedad decisiva es CONTROLSOURCE, que define:
 - El campo o la variable de dónde se tomará el valor que determina cuál opción del LISTBOX aparecerá seleccionada inicialmente.
 - Dónde se guardará el nuevo valor elegido en el LISTBOX.

En el ejemplo anterior, el valor de CONTROLSOURCE será *Encfac.Cli*. Si hiciéramos actual el registro de la tabla ENCFAC que se ve primero, cuyo campo CLI vale 17 y activáramos el LISTBOX veríamos que estaría seleccionada la opción de *Guzmán, Abel*. Esta selección inicial se debe a que VFP busca en la

segunda columna del LISTBOX la opción que corresponde al valor del campo CLI de la tabla ENCFAC, definido como CONTROLSOURCE del control. Si en esas circunstancias eligiéramos la segunda opción, correspondiente a *García, Luis*, en ese campo se guardaría el valor 258. Pero, ¿de qué depende que se guarde el código del cliente y no el nombre, ya que cada opción tiene dos columnas?

- 2. La segunda propiedad decisiva es BOUNDCOLUMN, que define de cuál columna hay que tomar el valor a guardar en el campo o variable establecido en CONTROLSOURCE. En el ejemplo, si esta propiedad es 2, el valor que se guardará en ENCFAC.CLI será 258, esté o no visible esa columna.
- 3. La tercera propiedad decisiva es BOUNDTO. Cuando CONTROLSOURCE es un campo o variable de tipo numérico, VFP puede guardar en él el valor real que se visualiza en la opción elegida o el número de orden de esa opción, contando desde 1 todas las opciones del LISTBOX. Esto lo gobierna BOUNDTO. Si su valor es *Falso*, VFP guarda el número de orden; si es *Verdadero*, guarda el valor real. En el ejemplo anterior, como CONTROLSOURCE es ENCRFAC.CLI, como este campo es numérico deberemos recurrir a esta propiedad. Si el valor de BOUNDTO fuera *Verdadero*, al elegir la opción de *García, Luis*, se guardaría el valor 258, como seguramente querremos. Pero si BOUNDTO fuera *Falso*, se podría guardar 25, ó 47, u otro número, no un número cualquiera, sino exactamente el que corresponda al número de orden de esa opción, contando desde el comienzo. El comportamiento de esta propiedad es molesto, porque lo que normalmente se quiere es lo que se obtiene con *Verdadero*. No obstante, el objeto tiene por defecto el valor *Falso* en esta propiedad, por lo que hay que corregirlo.

Usar varias columnas facilita ubicar los clientes, con datos adicionales, como nombre y código. Cuál campo poner en la primera columna depende del concepto por el que queramos buscar. Si queremos buscar los clientes por sus nombres, el campo NOM debe ir en la primera columna. Aunque leyendo detenidamente todas las columnas podremos encontrar lo que necesitamos, si la primera tiene los nombres ordenados alfabéticamente, encontraremos rápidamente la línea necesitada, más si nos valemos del teclado. Si quisiéramos buscar por código, el campo CLI debería ir en la primera columna.

La propiedad relacionada con la búsqueda mediante teclado es INCREMENTALSEARCH. Si esta propiedad es *Falso*, VFP busca la primer entrada que coincida con la letra pulsada. Por ejemplo, la secuencia de pulsaciones M, A, R, hará que VFP busque la primera entrada que empieza con M, luego la primera que empieza con A y luego la primera que empieza con R. Si la secuencia es M, M, VFP buscará la primera entrada que empieza con M y luego la segunda que empieza con M. Si INCREMENTALSEARCH es *Verdadero*, la secuencia M, A, R, si se escribe rápido, buscará la primera entrada que empiece con estas tres letras, como puede ser *Martínez*.

La búsqueda se facilita para VFP y para el usuario si los datos de la primera columna están ordenados. En este caso, estarán juntas y ordenadas todas las entradas que empiezan con A, luego todas las que empiezan con B, etc. En consecuencia, si el origen es un listado de valores, deberían escribirse en orden; y si el origen es una tabla, ésta debería tener activado un índice que la ordene por el campo que va en la primera columna, lo que se consigue mediante la propiedad ORDER del CURSOR de la tabla.

El valor inicial de la propiedad NAME de los cuadros de lista es *List1*, *List2*, *List3*, etc., donde el número indica el orden de creación.

ComboBox

COMBOBOX (cuadro combinado) es un objeto que combina un cuadro de texto y un cuadro de lista. El cuadro de texto va acompañado por una flecha hacia abajo, que sirve para abrir el cuadro de lista. En el cuadro de texto se visualiza la primera columna de la línea últimamente seleccionada. El cuadro de lista funciona tal como se explicó en LISTBOX.

Sea el ejemplo dado anteriormente de la tabla ENCFAC y supongamos estar en un registro cuyos campos FAC y *Cli* son respectivamente *00000125* y 17. Sea además un cuadro combinado con las siguientes ca-

racterísticas: ROWSOURCETYPE es 6; ROWSOURCE es *Cli.Nom,Cli*; CONTROLSOURCE es *Encfac.Cli*; BOUNDCOLUMN es 2. Inicialmente, el cuadro de texto tendrá el aspecto de la Figura 1.11.

Avellaneda, Mauricio



El valor visualizado se debe a que *Avellaneda, Mauricio* fue el último cliente elegido en el COMBOBOX antes de ahora. Si damos clic en la flecha descendente se abrirá el cuadro de lista, donde aparecerá seleccionada la tercera fila, que corresponde al código del registro actual, *17*, como muestra la Figura 1.12.

Avellaneda, Mauricio	▼		
Furlani, Marisa	12	200	
García, Luis	2	258	
Gutiérrez, César	4	22	
Guzmán, Abel		17	
Hernández, José	5	67	
Hierrezuelo, Ana	7	'84	▼

Figura 1.12. Cuadro combinado con la lista desplegada

Si elegimos la primera fila, el valor 1200 de la segunda columna se almacenará en el campo CLI del registro actual de ENCFAC. En el cuadro de texto aparecerá *Furlani, Marisa*.

Hay dos estilos de este objeto, establecidos en la propiedad STYLE: lista desplegable y cuadro desplegable. *Lista desplegable* no deja escribir en el cuadro de texto: si se da clic en él, se abre la lista, donde se puede elegir cualquier opción. *Cuadro desplegable* deja escribir en el cuadro de texto y lo escrito se puede incluir como otra opción de la lista mediante programación.

El valor inicial de la propiedad NAME de los cuadros combinados es *Combo1*, *Combo2*, *Combo3*, etc., donde el número indica el orden de creación de estos objetos.

Grid, Column y Header

GRID (cuadrícula, grilla) es un contenedor, similar en aspecto y funcionamiento a la ventana EXAMINAR. Está formado por objetos COLUMN. Cada COLUMN es un contenedor de un objeto HEADER (encabezado) y otro objeto, normalmente un TEXTBOX. El GRID muestra registros y campos de una tabla o vista, permitiendo desplazarse por ellos.

	Nom	Cli	
	Gutiérrez, Ana	120	
	Herrera, Javier	375	
	Juárez, Pedro	897	
	La Hoz, Alberto	493	
	López, Daniel	141	
	Martín, Susana	167	
	Moreno, Felipe	224	
	Nieto, Gustavo	201	▼

Figura 1.13. Grid de una sola tabla

En el ejemplo anterior, los registros están ordenados por nombre. El orden no permite buscar escribiendo en el teclado, como en LISTBOX o COMBOBOX. Por ello, los registros pueden aparecer ordenados por cualquier columna. El orden se establece en el entorno de datos, escribiendo el índice a usar en la propiedad ORDER del cursor de la tabla mostrada en el GRID.

Un GRID puede mostrar también campos de dos o más tablas o vistas, para lo cual es necesario relacionarlas. La Figura 1.14, por ejemplo, representa un GRID con los campos NOM y CLI de la tabla CLI y los campos FAC y FEC de la tabla ENCFAC.

Nom	Cli	Fac	Fec	
Gutiérrez, Ana	120	25489	12/05/99	
Herrera, Javier	375	17626	05/04/99	
Juárez, Pedro	897	18941	18/10/99	
La Hoz, Alberto	493	22397	30/11/99	
López, Daniel	141	19477	01/05/99	
Martín, Susana	167	19537	08/03/99	
Moreno, Felipe	224	21804	15/06/99	
Nieto, Gustavo	201	20045	14/08/99	▼

Figura 1.14. Grid de dos tablas relacionadas, con relación uno a uno

El GRID anterior supone que se han relacionado los cursores de las tablas CLI y ENCFAC por el campo común CLI. La relación debe ir de CLI a ENCFAC, tabla ésta que debe tener un índice por facturas.

En la Figura 1.14 parece que, para cada cliente, hay una sola factura. Lo común es que a cada cliente corresponda más de una factura. Incluso si cada cliente tiene más de una factura, el GRID seguiría presentando el mismo aspecto de la Figura 1.14, si la relación está definida como uno a uno. Para ver todas las facturas de cada cliente, hay que hacer verdadera la propiedad ONETOMANY de la relación. En tal caso, el GRID se vería ahora como indica la Figura 1.15.

El relleno que aparece en en las columnas *Nom* y *Cli* de algunas filas indica que se trata del mismo cliente que figura anteriormente con valores. Este efecto visual enfatiza la relación uno a varios.

Los objetos GRID, COLUMN, HEADER y el TEXTBOX u otro objeto de la columna tienen propiedades. Veamos las más importantes.

Nom	Cli	Fac	Fec
Gutiérrez, Ana	120	25489	12/05/99
		25740	17/06/99
		26247	05/10/99
Herrera, Javier	375	17626	05/04/99
		18888	01/05/99
		18970	02/05/99
		19004	13/06/99
		19017	14/07/99

Figura 1.15. Grid de dos tablas relacionadas, con relación uno a varios

Propiedades de GRID

COLUMNCOUNT establece la cantidad de columnas del GRID. Así, COLUMNCOUNT del GRID en la Figura 1.13

vale 2, mientras que en las Figuras 1.14 y 1.15 vale 4.

RECORDSOURCE establece la tabla en que se basa el GRID. Si un GRID presenta campos de dos tablas unidas por una relación, se debe establecer la tabla de donde parte la relación. En cada GRID de las Figuras 1.13, 1.14 y 1.15, tal tabla es CLI, por lo que RECORDSOURCE debe llevar ese valor. En los dos últimos hay campos de la tabla ENCFAC. Esto es posible porque los campos que van a llenar las columnas, con sus tablas respectivas, se declaran en la propiedad CONTROLSOURCE de cada una de ellas.

RECORDMARK establece si el GRID debe llevar o no marcas de registros, es decir, los recuadros grises de la primera columna de las Figuras 1.13 y 1.14. Estos recuadros sirven para mostrar cuál es el registro actual, que se identifica con el símbolo .

DELETEMARK muestra una columna de recuadros que sirven para marcar o desmarcar registros a eliminar. Estos recuadros se ven en la segunda columna de la Figura 1.13 y en la primera de la Figura 1.15. Cuando un registro está marcado, el recuadro se ve negro. Si no lo está, se ve blanco.

SCROLLBARS muestra o no barras de desplazamiento horizontal o vertical. Las Figuras 1.13 y 1.14 tienen barra de desplazamiento vertical. La Figura 1.15 no tiene barra.

ALLOWADDNEW permite que el GRID se comporte como hace el comando VER – MODO AÑADIR, cuando se está examinando una tabla. Si la propiedad vale *Verdadero*, es posible agregar nuevos registros. Para agregar un nuevo registro, se debe ir al último visible y bajar con \downarrow .

HEADERHEIGHT define la altura de la fila de encabezado, igual para todas las columnas. En tiempo de diseño, la altura puede modificarse arrastrando el extremo izquierdo inferior del encabezado. Lo mismo se puede hacer en tiempo de ejecución, si ALLOWHEADERSIZING es *Verdadero*. Sin embargo, este último cambio vale para la sesión actual: al volver a ejecutar el formulario, la altura será la establecida en el diseño.

ROWHEIGHT define la altura de las filas de datos. En tiempo de diseño, la altura puede modificarse arrastrando el extremo izquierdo de la primera línea. Lo mismo se puede hacer en tiempo de ejecución, si ALLOWROWSIZING es *Verdadero*, pero sólo para la sesión actual.

GRIDLINES permite ver o no líneas separadoras de filas y columnas dentro del GRID.

El nombre inicial de cada GRID es *Grid1*, *Grid2*, *Grid3*, etc., donde el número indica el orden de creación de estos objetos.

Propiedades de COLUMN

CONTROLSOURCE establece el campo que ocupará la columna. El campo se debe indicar por su nombre, precedido por el nombre de la tabla y un punto. Así, en la columna *Cli* de la Figura 1.15 debe ser *Cli.Cli* y en la columna *Fec* debe ser *Encfac.Fe*. CONTROLSOURCE puede ser también un campo calculado, como la suma de dos campos, el producto de un campo por una constante, una expresión compleja, etc.

COLUMNORDER es el número de orden de cada columna en el conjunto. Al establecer por primera vez COLUMNOCOUNT, se genera igual número de columnas: la primera tiene COLUMNORDER 1, la segunda COLUMNORDER 2, etc. Si se desea modificar el orden de las columnas sin rehacer todo el GRID, basta con alterar sus COLUMNORDER. Ello se puede hacer también en tiempo de diseño arrastrando las columnas a otra posición, cuando el cursor toma la forma I en el encabezado. Lo mismo vale en tiempo de ejecución, si la propiedad MOVABLE de la columna deseada es *Verdadero*, pero sólo para la sesión actual.

WIDTH establece el ancho de la columna. Éste se puede modificar también en tiempo de diseño, arrastrando la línea vertical derecha del encabezado. Lo mismo se puede hacer en tiempo de ejecución, si RESIZABLE de la columna es *Verdadero*, pero sólo para la sesión actual.

FORMAT E INPUTMASK dan formatos de presentación y edición al campo contenido en un COLUMN.

Las propiedades CONTROLSOURCE, FORMAT e INPUTMASK se trasmiten al objeto contenido dentro de la co-

lumna, por lo que no hace falta que sean vueltas a definir en él. Si se las define, no deben contradecir lo establecido en COLUMN.

El valor inicial de la propiedad NAME de cada COLUMN dentro de un GRID es *Column1*, *Column2*, *Column3*, etc., en correspondencia con el número de columnas indicado en COLUMNCOUNT.

Propiedades de HEADER

CAPTION establece el texto que aparece en el encabezado de la columna. Si el GRID se crea usando el generador o arrastrando el título de un cursor desde el entorno al formulario, CAPTION tomará como valor el nombre del campo, si no hay título, o el título de campo, si existe.

ALIGNMENT establece la ubicación del texto en el encabezado, habiendo tres posibilidades horizontales y tres verticales, más una automática.

El nombre inicial de HEADER, en cualquier columna, siempre es Header1.

Spinner

250 🗢

Figura 1.16. Modelo de control numérico

CONTROLSOURCE establece el campo numérico (o variable numérica) del cual se tomará el valor inicial del objeto y en el cual se almacenará cualquier cambio realizado en él. El funcionamiento, entonces, es similar al de un cuadro de texto.

KEYBOARDHIGHVALUE y KEYBOARDLOWVALUE establecen los valores máximo y mínimo permitidos por teclado.

SPINNERHIGHVALUE y SPINNERLOWVALUE establecen los valores máximo y mínimo que se puede introducir con el mouse, oprimiendo las flechas ascendente y descendente.

INCREMENT establece el incremento por el cual variará el número visualizado en el cuadro de texto cuando se usen las flechas, un incremento (o decremento) por cada clic.

FORMAT E INPUTMASK sirven para modificar el aspecto del número del cuadro de texto.

El valor inicial de la propiedad NAME de cada SPINNER es *Spinner1*, *Spinner2*, *Spinner3*, etc., donde el número indica el orden en que se van creando estos objetos.

PageFrame y Page

PAGEFRAME (marco de páginas) es un contenedor. Los objetos de primer nivel de inclusión son de tipo PAGE (página). PAGEFRAME se asemeja a un conjunto de fichas, cada una con una "oreja". Sólo puede haber una ficha activa por vez, que muestra todos los objetos que contiene, como TEXTBOX, GRID, COMBOBOX, SPINNER, etc. La Figura 1.17 muestra un PAGEFRAME de dos fichas.

PAGE (página o ficha) es un contenedor de objetos de cualquier tipo, excepto formularios, marcos de páginas o páginas. Cada PAGE tiene el aspecto de una ficha con una "oreja" donde hay un texto. En la Figura 1.17 hay dos fichas, cuyos textos son Proveedores y Clientes.

Propiedades de PAGEFRAME

La propiedad más importante de PAGEFRAME es PAGECOUNT, donde se establece la cantidad de objetos PAGE que contiene, es decir, la cantidad de fichas.

El valor inicial de la propiedad NAME de cada PAGEFRAME es *PageFrame1*, *PageFrame2*, *PageFrame3*, etc., donde el número indica el orden de creación.

Proveedores		Clientes	
Código			
Nombre			
Domicilio			
Teléfono			
Fax			



Propiedades de PAGE

La propiedad donde se define el texto de la "oreja" es CAPTION.

El orden relativo de cada PAGE en el conjunto se establece en la propiedad PAGEORDER. Para variar su posición en el conjunto, basta con modificar esta propiedad.

Al crear un PAGEFRAME de 4 páginas, por ejemplo, el valor inicial de la propiedad NAME de cada PAGE va de *Page1* a *Page4*. Estos valores son usados también para CAPTION.

Line

LINE (línea) es un objeto de dibujo para trazar una línea horizontal, vertical u oblicua en un formulario. La inclinación de la línea viene dada por el arrastre del mouse cuando se la crea. En el caso de describir un rectángulo, la línea será la diagonal.

Es posible variar el color, estilo, grosor, etc. de una línea, mediante propiedades que no es importante detallar.

La propiedad LINESLANT permite variar la pendiente de la diagonal dentro del rectángulo imaginario donde está inscripta. Los valores de esta propiedad, curiosamente, son las barras / y \.

El valor inicial de la propiedad NAME de cada LINE es *Line1*, *Line2*, *Line3*, etc., donde el número indica el orden en que se van creando estos objetos.

Shape

Shape (forma) es un objeto de dibujo que traza rectángulos, rectángulos con esquinas redondeadas,

elipses y círculos. Inicialmente se crea un rectángulo o cuadrado, cuyas dimensiones se fijan por el arrastre del mouse.

Este objeto, que no es un contenedor, se usa para enmarcar otros objetos. De aquí que, si se crea un SHAPE alrededor de un grupo de objetos y no es transparente, los tapará. La solución es hacerlo transparente o mandarlo al fondo.

La propiedad BACKSTYLE sirve para hacer transparente u opaco un SHAPE.

Es posible variar el color de relleno, el color, grosor y estilo del borde, etc., mediante propiedades que no es importante detallar.

La propiedad CURVATURE permite redondear las esquinas de un SHAPE. Varía entre 0 y 99. Si vale 0, no hay redondeo. A mayor valor, mayor redondeo. Cuando vale 99 se obtiene una elipse, si se ha dibujado un rectángulo, o un círculo, si se ha dibujado un cuadrado. El cuadrado se obtiene cuando los valores de HEIGHT (altura) y WIDTH (ancho) son iguales, con independencia de las deformaciones visuales que pueda producir el ajuste horizontal y vertical del monitor.

El valor inicial de la propiedad NAME de cada SHAPE es *Shape1*, *Shape2*, *Shape3*, etc., donde el número indica el orden de creación de estos objetos.

Image

El objeto IMAGE muestra una imagen contenida en un archivo mapa de bits (extensión BMP) o ícono (extensión ICO). En la propiedad PICTURE se define el camino y el archivo que contienen la imagen.

Capítulo 2. Propiedades de objetos de formulario

Una propiedad puede ser exclusiva de un objeto, pero también puede ser común a varios o a todos ellos. Debido a que hay gran número de propiedades, cuyo estudio superaría el alcance de un curso elemental, se exponen a continuación algunas propiedades muy usuales, ordenadas alfabéticamente.

Alignment

ALIGNMENT (alineación) permite alinear el texto dentro del objeto que lo lleve. Ejemplos son los textos definidos como CAPTION y el contenido de un campo o de una variable que se visualice en un TEXTBOX (lo definido como CONTROLSOURCE).

LABEL y EDITBOX admiten tres posiciones horizontales: *Izquierda*, *Derecha* y *Centro*.

TEXTBOX, además de las anteriores, admite alineación *Automática*, es decir, dejar que VFP la determine en base al tipo de datos de CONTROLSOURCE: si el tipo es numérico, la alineación es *Derecha*; para los demás tipos, es *Izquierda*. Estos valores son iguales para SPINNER, aunque *Automático* carece de sentido.

En COLUMN y HEADER de un GRID, para cada alineación horizontal se puede establecer alineación vertical (superior, inferior y media) o dejar que VFP la determine. Hay, entonces, 10 posibilidades:

Medio izquierda	Superior izquierda	Inferior izquierda	Automático
Medio derecha	Superior derecha	Inferior derecha	
Medio centro	Superior centro	Inferior centro	

Las diez posibilidades de COLUMN permiten dar más posibilidades al TEXTBOX que normalmente es el objeto que contiene. Esto se debe a que el ancho de la columna y el alto de las filas se puede variar a voluntad, logrando que el contenido del TEXTBOX aparezca adecuadamente ubicado. Recuerde que TEXBOX tiene solamente tres posibilidades, las cuales no conseguirían la adecuación.

En OPTIONGROUP, sus OPTIONBUTTON componentes están compuestos por un botón y un texto. El texto puede ir a la izquierda o derecha del botón. Esta posición se controla con la propiedad ALIGNMENT, que puede tomar los valores *Izquierda* y *Derecha*. Lo mismo acontece con CHECKBOX, compuesto por un pequeño cuadro y un texto: ALIGNMENT permite que el texto vaya a la izquierda o derecha del cuadro.

AllowAddNew

ALLOWADDNEW (permitir añadir nuevos registros) posibilita agregar nuevos registros a la tabla que se visualiza en un GRID. Para añadirlos, esta propiedad debe valer *Verdadero*. Para no añadirlos, debe valer *Falso*. Cuando es *Verdadero*, el GRID funciona como el VER – MODO AÑADIR, al examinar una tabla. Si el GRID contiene campos de dos tablas relacionadas, sólo deja añadir registros a la que está definida en RECORDSOURCE, es decir, la tabla parent de la relación. Los campos de la tabla child son de sólo lectura.

AllowHeaderSizing

ALLOWHEADERSIZING (permitir variar encabezado) deja modificar, mediante arrastre, la altura del encabezado de un GRID en tiempo de ejecución. Es una propiedad de GRID. *Verdadero* permite la variación; *Falso* impide hacerlo. Para variar la altura, se debe tomar el extremo izquierdo de la línea inferior del encabezado y arrastrar hacia arriba o abajo. La modificación de la altura vale solamente para la ejecución actual. En ejecuciones posteriores, la altura inicial del encabezado será la establecida en HEADERHEIGHT.

La variación del ancho del encabezado se logra variando el ancho de la columna.

AllowRowSizing

ALLOWROWSIZING (permitir variar filas) deja modificar, mediante arrastre, la altura de las filas de un GRID en tiempo de ejecución. Es una propiedad de GRID. *Verdadero* permite la variación; *Falso* impide hacerlo. Para variar la altura, se debe tomar el extremo izquierdo de la primera línea de división y arrastrar hacia arriba o abajo. Modificar la altura vale solamente para la ejecución actual. En ejecuciones posteriores, la altura ra inicial será la establecida en ROWHEIGHT.

AutoSize

A veces, el texto definido como CAPTION de un objeto aparece trunco dentro de él, o el objeto resulta demasiado ancho con respecto al texto. Esto se debe a que el ancho del objeto difiere del ancho del texto. Si AUTOSIZE (tamaño automático) es *Verdadero*, se resuelve este problema, haciendo que el ancho del objeto se adapte al ancho del texto. Tienen esta propiedad LABEL, COMMANDBUTTON, OPTIONBUTTON y CHECKBOX. OPTIONGROUP y COMMANDGROUP, cuyos componentes pueden variar su ancho en función del texto que los acompaña, también poseen esta propiedad, porque están indirectamente afectados por el ancho de los textos contenidos. TEXTBOX y COLUMN carecen de esta propiedad: el ancho se determina aproximadamente por el tamaño del campo definido como CONTROLSOURCE y por el tipo y tamaño de la fuente usada, información que tienen en cuenta los generadores de estos objetos.

BackColor

Vea Propiedades de color.

BackStyle

BACKSTYLE (estilo de fondo) maneja el relleno de un objeto, que puede ser *Opaco* o *Transparente*. Si es transparente, un objeto que esté por detrás puede verse, sin tener que llevarlo al frente. El color de fondo de un objeto, definido con BACKCOLOR, se ve únicamente si el objeto es opaco.

BoundColumn

En un COMBOBOX o LISTBOX de varias columnas, establece cuál de ellas se corresponde con el campo o la variable definidos en CONTROLSOURCE, si se usa esta propiedad. BOUNDCOLUMN establece cuál es tal columna, indicándola por un número. La primera columna tiene el número 1, la segunda el 2, etc.

Sea un COMBOBOX cuya lista se construye con los campos NOM y CLI de la tabla CLI. Al elegir un cliente, queremos guardar el valor *Cli* de la opción en el campo CLI de la tabla ENCFAC. Entonces, ROWSOURCETYPE del COMBOBOX será 6, ROWSOURCE será *Cli.Nom,Cli* y CONTROLSOURCE será *Encfac,Cli*. Como lo que queremos es guardar el valor de *Clie*, que ocupa la segunda columna, BOUNDCOLUMN será 2.

BoundTo

Sea un objeto COMBOBOX o LISTBOX, cuyo CONTROLSOURCE corresponde a un campo o variable de tipo numérico. Si se elige una opción cualquiera, el valor que se guardará en el campo o variable será el núme-

ro de orden de esa opción. Este número es *1* para la primera opción, *2* para la segunda, etc. Normalmente esto no será lo deseado, porque lo que suele necesitarse es guardar en el campo o en la variable el valor de la opción que aparece en la columna indicada por BOUNDCOLUMN. Este aparente mal funcionamiento se debe a que el valor por defecto de BOUNDTO es *Falso*. Para obtener el valor de la columna deseada, no su número de orden, BOUNDTO debe establecerse como *Verdadero*.

ButtonCount

BUTTONCOUNT (cantidad de botones) establece cuántos botones tiene un objeto OPTIONGROUP o COMMANDGROUP. Cada botón de OPTIONGROUP es un OPTIONBUTTON y cada botón de COMMANDGROUP es un COMMANDBUTTON.

Caption

CAPTION (leyenda) define el texto constante que aparece en algunos objetos. Los que llevan esta propiedad son: CHECKBOX, COMMANDBUTTON, FORM, HEADER, LABEL, OPTIONBUTTON, PAGE. La Figura 2.1 indica dónde aparece el texto de CAPTION.

<u>Objeto</u>	El texto aparece
СнескВох	Junto a la casilla, a izquierda o derecha
COMMANDBUTTON	Dentro del botón
Form	En la barra de título del formulario
Header	En el encabezado de columna de un GRID
LABEL	Como lo único visible
OptionButton	Junto al botón, a izquierda o derecha
Page	En la "oreja" de la página

Figura 2.1. Posición de CAPTION, según el objeto

Al crear un objeto que tenga CAPTION desde la barra de herramientas CONTROLES DE FORMULARIOS, inicialmente se da a esta propiedad una abreviatura que indica el tipo de objeto más un número. Este número va aumentando de uno en uno a medida que se crean otros objetos del mismo tipo. Por ejemplo, el primer COMBOBOX será *Combo1*, el segundo *Combo2*, etc.; el primer COMMANDBUTTON será *Command1*, el segundo *Command2*, etc. Este valor de CAPTION se usa también como valor inicial de la propiedad NAME. Al arrastrar un campo desde el entorno de datos al formulario, según el tipo de datos del campo se crea un TEXTBOX, EDITBOX o CHECKBOX. Los dos primeros objetos van acompañados por un LABEL, cuyo CAPTION será el nombre del campo (o su título, si lo hay). El tercer objeto tiene propiedad CAPTION, cuyo valor será análogo al de los dos anteriores.

Si el texto de CAPTION se ve trunco dentro del objeto, es porque el tamaño de éste es insuficiente para mostrarlo todo. Para solucionar este problema, haga *Verdadero* la propiedad AUTOSIZE. El objeto cambiará su tamaño para dar cabida a todo el texto. Si el objeto es un OPTIONGROUP o COMMANDGROUP, haga *Verdadero* las propiedades AUTOSIZE de los botones y la del grupo. De este modo, cada botón tomará el tamaño adecuado a su CAPTION y el grupo se adecuará al tamaño del botón con texto más grande. La misma solución vale para el caso contrario, cuando el tamaño del objeto es mayor que el texto de CAPTION.

ChildOrder

Consulte Propiedades especiales de un GRID subordinado o el objeto RELATION.

ColumnCount

COLUMNCOUNT (cantidad de columnas) establece cuántas columnas tiene un GRID, un COMBOBOX o un LISTBOX. En GRID el valor de esta propiedad genera igual cantidad de objetos COLUMN. En COMBOBOX y LISTBOX, por ser controles, no se generan objetos contenidos, sino que la información de esta propiedad sirve para dibujar adecuadamente las columnas y para identificar, mediante la propiedad BOUNDCOLUMN, el valor a devolver y depositar en el receptor definido en CONTROLSOURCE. Si un GRID tiene ya un valor definido en COLUMNCOUNT y se lo cambia por otro menor, se eliminan las columnas excedentes. Si el nuevo valor es mayor, se añaden nuevas columnas.

Un GRID trabaja con una sola o con varias tablas relacionadas. No es necesario que contenga todos los campos, sino los necesitados. Para ello hay que establecer cuántos son estos campos, lo cual se expresa como un número en COLUMNCOUNT. Esto produce un GRID con esa cantidad de columnas. Luego hay que definir la propiedad CONTROLSOURCE de cada columna, indicando cuál es el campo que se quiere en ella. El valor inicial de COLUMNCOUNT de un GRID es -1. Este valor manda mostrar todos los campos de la tabla definida en la propiedad RECORDSOURCE. Si desean algunos campos, se debe cambiar el valor -1 por un número positivo, igual al número de campos a mostrar en la correspondiente cantidad de columnas.

En COMBOBOX o LISTBOX, para construir las opciones, se puede usar una lista de valores escritos a mano o los campos de una tabla. Si se quiere que los valores escritos sean distribuidos en varias columnas, el número de éstas se define en COLUMNCOUNT. La distribución se hace llenando todos los componentes de la primera opción, luego los de la segunda, etc. Si las opciones se construyen con una lista de campos, debería haber tantas columnas como campos. Pero si se da a COLUMNCOUNT un número menor a la cantidad de campos, prevalece COLUMNCOUNT. Por defecto, el valor de COLUMNCOUNT para COMBOBOX y LISTBOX es 0, que indica formar una sola columna con el primer campo de la lista.

ColumnLines

COLUMNLINES (líneas entre columnas) permite poner o quitar líneas verticales entre las columnas de un COMBOBOX o LISTBOX. Si COLUMNLINES es *Verdadero*, se colocan. Si es *Falso*, se eliminan.

ColumnOrder

Al crear un GRID, se definen cuántas columnas debe tener en la propiedad COLUMNCOUNT. Por ejemplo, si COLUMNCOUNT es 4, se crean cuatro columnas, cuyos nombres iniciales son *Column1*, *Column2*, *Column3* y *Column4*. Cada columna lleva una propiedad COLUMNORDER (orden de columna), que indica su posición relativa en el conjunto.

Inicialmente las columnas y sus COLUMNORDER se corresponden, como indica la Figura 2.2. Si luego se quiere que la secuencia de columnas sea *Column2*, *Column1*, *Column4* y *Column3*, no hay que rehacer el GRID. Basta cambiar sus COLUMNORDER, como muestra la Figura 2.3.

Objeto	ColumnOrder
COLUMN1	1
COLUMN2	2
COLUMN3	3
COLUMN4	4

Figura 2.2. Valores iniciales de COLUMNORDER

Objeto	ColumnOrder
COLUMN1	2
COLUMN2	1
COLUMN3	4
COLUMN4	3

Figura 2.3. Valores modificados de COLUMNORDER

ColumnWidths

COMBOBOX y LISTBOX usan una lista de opciones de una o más columnas. Como estos objetos no son contenedores, sino controles, deben incluir en sí toda la información necesaria para ser producidos adecuadamente. Es el caso del ancho de cada columna. COLUMNWIDTHS (anchos de columna) fija el ancho de cada una, mediante números separados por comas. Si hay 4 columnas, COLUMNWIDTHS podría ser *30,45,200,150*. Si no se define COLUMNWIDTHS y hay varias columnas, éstas podrían verse mal alineadas. Si se quiere que una columna forme parte de la lista pero que no sea visible, debe valer 0. En el ejemplo anterior, si quisiéramos que no se vea la segunda columna, COLUMNWIDTHS sería *30,0,200,150*.

Las columnas de estos objetos no se definen con la propiedad WIDTH, como sucede en GRID. GRID es un contenedor de objetos COLUMN, cada uno de los cuales tiene un ancho, que se define en WIDHT. Sin embargo, COMBOBOX y LISTBOX también tienen propiedad WIDTH: esta propiedad define el ancho de COMBOBOX cuando está "plegado" y el ancho de LISTBOX, que siempre está desplegado.

En COMBOBOX, la suma de los anchos de las columnas puede diferir del valor de la propiedad WIDTH del cuadro de texto que despliega la lista. Al ser desplegada, la lista tendrá los anchos estipulados para cada columna, más el ancho de la barra de desplazamiento. En el caso de un LISTBOX, que es en sí la lista desplegada, su WIDTH no debería ser menor a la suma de valores de COLUMNWIDTHS, más el ancho de la barra de desplazamiento. Si WIDTH es menor a la suma, no se verá la totalidad de las columnas.

ControlSource

CONTROLSOURCE (origen del control) establece el nombre de un campo de datos o de una variable de memoria. Tales campo o variable sirven para un doble propósito:

- Proporcionar el contenido que aparece inicialmente en el objeto. Cuando CONTROLSOURCE hace referencia a un campo, cada vez que se cambie de registro cambiará el contenido del objeto. A veces, para ver el nuevo contenido, es necesario usar el método REFRESH.
- Guardar el nuevo contenido del objeto, si el usuario lo modifica. Si CONTROLSOURCE es un campo, el nuevo contenido se guardará en el campo del registro actual.

Si CONTROLSOURCE es una variable, se escribe su nombre. Si es un campo, se escribe su nombre prefijado con el alias de la tabla, en el formato **Tabla.Campo**.

Sea un cuadro de texto cuyo CONTROLSOURCE es *Art. Nom.* Si en el registro actual de la tabla ART el valor del campo NOM es *Armario 4 puertas*, veremos *Armario 4 puertas* en el cuadro de texto. Si en éste cambiamos ese dato por *Mesa desplegable*, lo que realmente hacemos es cambiar el valor del campo NOM por *Mesa desplegable*. Véase la Figura 2.4.

El cambio efectivo en el campo se produce cuando se selecciona otro objeto, es decir, al perderse el enfoque sobre el objeto en cuestión (evento LOSTFOCUS).

Propiedades de objetos de Formulario - Edmundo A. Cáceres - 2011



Figura 2.4. Funcionamiento de un cuadro de texto

La propiedad CONTROLSOURCE se aplica a varios objetos, como detallamos a continuación.

TextBox. Este objeto es adecuado para ver y modificar campos o variables de tipo carácter, fecha y numérico. Puede usarse para tipos lógico y memo, pero hay objetos específicos para ellos. El ejemplo anterior es suficiente para entender como funciona CONTROLSOURCE.

EditBox. Se usa para ver y modificar campos memo, que no tienen límite en tamaño. El objeto puede llevar una barra de desplazamiento vertical, pues el área con que se lo diseña puede no alcanzar para mostrar todo lo que contiene o puede contener. También puede servir para variables de caracteres de gran tamaño. La propiedad CONTROLSOURCE funciona como se ha expuesto en el ejemplo anterior.

CheckBox. Este objeto se usa para ver y modificar campos o variables de tipo lógico. Tiene el aspecto de un pequeño recuadro acompañado por un texto. El recuadro puede llevar o no una tilde, que se pone o saca dando clic en él. CONTROLSOURCE es el nombre de un campo o variable lógicos. Si inicialmente el campo vale *.T.*, el recuadro se ve tildado. Si inicialmente vale *.F.*, el recuadro no tiene tilde. Si se pone o quita la tilde, el campo cambia a *.T.* o *.F.*, respectivamente.

OptionGroup. Es un contenedor de opciones que sirve para ver y modificar campos o variables. Cada opción tiene un botón y un texto. Sólo una opción puede estar activa, lo que se ve como una marca en el botón respectivo. Cada opción se identifica por su número de botón que otorga VFP o por su CAPTION. CONTROLSOURCE es el nombre de un campo o de una variable de tipo numérico o de caracteres. Si el tipo es numérico, su valor activa inicialmente el botón de igual número. Así, si el campo definido en CONTROLSOURCE tiene el valor 1, inicialmente aparecerá activado el primer botón. Si luego se activa otro botón, digamos el segundo, se guardará 2 en el campo, tal como ilustra la Figura 2.5.



Figura 2.5. El campo ESTADO es de tipo numérico

Si el campo o la variable son de caracteres, su valor activa inicialmente el botón con CAPTION exactamente igual. Sea un OPTIONGROUP cuyas opciones son *Casado, Soltero, Viudo*. Si el campo definido en CONTROLSOURCE contiene *Casado*, el botón inicialmente activo es el que corresponde al texto *Casado*. Si se activa el botón *Soltero*, se guardará el texto *Soltero* en el campo, tal como muestra la Figura 2.6.

Propiedades de objetos de Formulario - Edmundo A. Cáceres - 2011



Figura 2.6. El campo ESTADO es de tipo carácter

Spinner. Se usa para ver y modificar campos o variables de tipo numérico. Tiene la forma de un cuadro de texto, asociado a un ◆. Inicialmente muestra el valor que tiene el campo definido en CONTROLSOURCE. Si se aumenta el valor con ◆, se lo disminuye con ▼, o se escribe otro número, este nuevo valor se almacena en el campo o variable.

ComboBox y ListBox. COMBOBOX es un TEXTBOX asociado a una lista desplegable, que se abre dando clic en el adjunto. LISTBOX es una lista desplegada. En ambos objetos, la lista está formada por varias líneas que representan opciones. Cada línea está compuesta por uno o más datos, que constituyen columnas. Si hay un dato, habrá una columna; si hay dos datos, dos columnas, etc. Los datos pueden tomarse de distinto origen, siendo los más característicos de una lista de datos introducidos manualmente o de los campos de una tabla. De dónde se toman los componentes y cuáles son ellos, se definen mediante las propiedades ROWSOURCETYPE y ROWSOURCE, respectivamente, como indica la Figura 2.7.

RowSourceType	RowSource
(de dónde tomar los componentes)	(cuáles son los componentes)
1 – Valor: De una lista manual	Los componentes se escriben, separados por comas:
	a) Una columna: Casado, Soltero, Viudo
	b) Dos columnas: Casado, 1, Soltero, 2, Viudo, 3
6 – Campos: De campos de una tabla.	Se indica la tabla, un punto y la lista de campos:
	c) Una columna: <i>Art.Nom</i>
	d) Tres columnas: Art.Nom,Cod,Pre

Figura 2.7. Relación entre las propiedades ROWSOURCETYPE y ROWSOURCE

Para el ejemplo a) de la figura anterior habrá una columna con tres líneas y para b) dos columnas con tres líneas. Eso supone que COLUMNCOUNT vale 1 y 2, respectivamente. La Figura 2.8 ilustra lo dicho mediante dos listas, sin importar sin son de un LISTBOX o un COMBOBOX.



Figura 2.8. Listas construidas con uno y dos valores

Para el ejemplo c) habrá una columna y para d) tres columnas. En ambos casos, habrá tantas opciones como registros tenga la tabla ART. COLUMNCOUNT debe valer 1 y 3, respectivamente. La Figura 2.9 ilustra lo dicho mediante dos listas.

c)	Cama 1 plaza		d)	Cama 1 plaza	A02	120	
	Mesa desplegable			Mesa desplegable	A04	375	
	Mesa de TV			Mesa de TV	A03	245	
	Ropero 3 cuerpos			Ropero 3 cuerpos	A01	897	
	Ropero 2 cuerpos			Ropero 2 cuerpos	A06	720	
	Silla escandinava	▼		Silla escandinava	A05	280	▼

Figura 2.9. Listas construidas con uno y tres campos

Para el ejemplo a), supongamos definir como CONTROLSOURCE el campo SOCIOS. ESTADO, de caracteres. Cuando estemos en el primer registro de la Figura 2.6, el de *Suárez, Esteban*, la opción inicialmente seleccionada en la lista será la que dice *Casado*. Estando en ese registro, si en la lista elegimos *Viudo*, se guardará *Viudo* en el campo ESTADO. Esto se debe a la propiedad BOUNDCOLUMN, que indica cuál columna está relacionada con el campo definido en CONTROLSOURCE. Como el valor por defecto es 1, cuando trabajamos con una columna no hay que cambiarlo.

Para el ejemplo b), sea también que CONTROLSOURCE es el campo SOCIOS.ESTADO. Pero en este caso, en lugar de guardar en él las palabras *Casado*, *Soltero* o *Viudo*, queremos guardar los códigos 1, 2 ó 3, que los representan. Cuando estemos en el primer registro de la Figura 2.5, la línea inicialmente seleccionada en la lista será la primera, donde aparecen los valores *Casado* y 1. Si elegimos la tercera línea, compuesta por los valores *Viudo* y 3, ¿cuál de estos valores se guardarán en ESTADO? Como queremos guardar el código, que está en la segunda columna, deberemos definir BOUNDCOLUMN con el valor 2. ¿Para qué hemos usado dos columnas? Pues porque es mucho más claro buscar un estado civil por su nombre, pero guardar en el campo ESTADO de la tabla SOCIOS el código correspondiente, que es más corto. El código debe estar en la lista, incluso si no es visible.

Para el ejemplo c), sea que CONTROLSOURCE es el campo FACT. NOM. Como se trata de una columna, BOUNDCOLUMN debe ser 1, valor por defecto. Al elegir una opción, se guardará el único valor de ella en FACT.NOM. Note que no ahorraríamos espacio en FACT, porque su campo NOM debería ser igual en tamaño al de ART.NOM, de donde tomaríamos el dato a través de la lista. Lo único que ahorraríamos sería no teclear los nombres de los artículos, evitando errores de escritura.

Para el ejemplo d), supongamos que CONTROLSOURCE sea el campo ART de la tabla FACT. Como el objeto tiene tres columnas para NOM, COD y PRE, tomados de la tabla ART, ¿cuál de ellas está relacionada con CONTROLSOURCE? Los nombres ART y COD no coinciden en la forma, pero sí conceptualmente, pues contienen códigos de artículos. En la lista, la columna que muestra el contenido de este campo es la segunda, por lo que BOUNDCOLUMN debe ser 2.

Observe que, cuando hay una sola columna, el comportamiento de LISTBOX y COMBOBOX es similar al de OPTIONGROUP. Pero en OPTIONGROUP la propiedad CONTROLSOURCE debe referirse a un campo o variable de tipo numérico o de caracteres, mientras que en LISTBOX y COMBOBOX puede ser también de otros tipos, como fecha o fecha – hora.

Grid. Este objeto produce en un formulario algo análogo a la ventana EXAMINAR. GRID muestra campos de una tabla o vista, incluso campos de tablas relacionadas. GRID es un contenedor, compuesto por columnas. Cada una de éstas es un contenedor, compuesto por un objeto HEADER (encabezado) y otro objeto, como TEXTBOX, EDITBOX, SPINNER, COMBOBOX o CHECKBOX. El que se genera por defecto es TEXTBOX.

Cada columna tiene la propiedad CONTROLSOURCE. Los objetos contenidos (excepto HEADER) también la tienen. Como CONTROLSOURCE es el nombre de un campo que muestra su valor y almacena los cambios introducidos, es la propiedad que debemos usar para definir qué campo va en cada columna.

Si cada COLUMN y su objeto contenido tienen CONTROLSOURCE, ¿cuáles de las dos propiedades hay que

definir? Por regla general, defina únicamente CONTROLSOURCE en COLUMN. Incluso si ninguna columna lo tiene definido, VFP toma los campos en el orden de la estructura de la tabla. La definición es necesaria cuando los campos no siguen el orden de la estructura o son algunos campos, no todos. Por ejemplo, si queremos un GRID con los campos NOM, ART y PRE de la tabla ART, usando un TEXTBOX para cada uno, debemos definir lo que indica la Figura 2.10.

Column1	Column2	Column3
Defina ControlSource:	Defina ControlSource:	Defina ControlSource:
Art.Nom	Art.Art	<i>Ar.Pre</i>
TEXTBOX	TEXTBOX	TEXTBOX
No defina ControlSource.	No defina ControlSource.	No defina ControlSource.

Figura 2.10. Cómo definir CONTROLSOURCE para cada columna de un GRID

Curvature

CURVATURE (curvatura) afecta las esquinas de un objeto SHAPE. Varía entre 0 y 99. Si vale 0, las esquinas son rectas. A medida que se incrementa el valor, aumenta la redondez de las esquinas. Cuando vale 99, el objeto toma la forma de una elipse, si es rectangular, o de un círculo, si es un cuadrado.

DeleteMark

DELETEMARK (marca de destrucción) hace que un GRID muestre o no una columna de cuadritos antes del campo que está más a la izquierda. Cada cuadrito indica si el registro está marcado para destruir. Si el registro no está marcado, el cuadrito está en blanco; si está marcado, el cuadrito está en negro. La columna también permite marcar o desmarcar registros, haciendo clic en el cuadrito del registro deseado. Si DELETEMARK es *Verdadero*, esta columna es visible; si es *Falso*, no se ve.

Enabled

ENABLED (habilitado) posibilita que un objeto se habilite o deshabilite, es decir, que se le pueda dar enfoque o no. Si ENABLED es *Falso*, el objeto se deshabilita; si es *Verdadero*, se habilita. Cuando un objeto está deshabilitado, toma otro color de fondo, por defecto gris.

Font...

Para todas las propiedades que comienzan con esta palabra, vea Propiedades de fuente.

ForeColor

Vea Propiedades de color.

Format

FORMAT (formato) establece características para todo el contenido del campo o variable que se muestra o modifica en un objeto. Para ello se vale de *códigos de formato*. Se diferencia de la propiedad INPUTMASK en que ésta define características para cada posición del contenido. La Figura 2.11 proporciona algunos códigos usuales. FORMAT puede llevar un solo código o varios, pero todos deben ser compatibles con el ti-

po de datos al que se aplican. Los códigos de esta propiedad, al igual que los códigos de INPUTMASK, no sólo se aplican en formularios, sino que tienen validez en otros ámbitos, como en los campos de una tabla de base de datos, en los objetos CAMPO de los informes y en la función *transform()*.

<u>Código</u>	Efecto
٨	Muestra datos numéricos en notación científica.
L	Por defecto, los ceros izquierdos de una expresión numérica no se ven.
	El código L modifica este comportamiento, mostrando tales ceros.
R	Permite que la propiedad INPUTMASK admita caracteres que no son códigos, que se
	visualizan pero no se almacenan. Se aplica a datos numéricos y de caracteres.
Z	El dato numérico 0 de un objeto se ve en blanco, salvo cuando tiene el enfoque.
!	Convierte todas las letras de un dato de caracteres a mayúscula.
Α	Sólo admite letras en un dato de caracteres.
\$	Muestra un dato numérico precedido del símbolo monetario.

Figura 2.11. Códigos de formato

GridLines

GRIDLINES (líneas del GRID) hace que un GRID muestre o no líneas horizontales y verticales para separar los registros y los campos. Sus valores son: *0 – Ninguna*, *1 – Horizontal*, *2 – Vertical*, *3 – Ambas*.

HeaderHeight

HEADERHEIGHT (altura del encabezado) es un valor numérico, que establece en píxeles la altura de la línea de encabezado de un GRID. Esta altura será la que tenga inicialmente el encabezado cada vez que se ejecute el formulario. La propiedad ALLOWHEADERSIZING permite variar esta altura en tiempo de ejecución, pero la variación vale solamente para la ejecución en que se hace el cambio. Advierta que es una propiedad de GRID, no de HEADER, porque no podría haber un HEADER con una altura y otro con otra distinta. Si fuera así, se quebraría la continuidad de los renglones visualizados en el GRID.

Height

Vea Propiedades de tamaño y ubicación.

Increment

IncrementalSearch

INCREMENTALSEARCH (búsqueda incremental) se aplica a COMBOBOX y LISTBOX para buscar en la primera columna de la lista de opciones mediante el teclado. Si vale *Falso*, cuando se pulsa una tecla, VFP busca la opción más próxima que comienza con esa tecla. Si se pulsan dos teclas, primero busca la opción más próxima que comienza con la primera tecla y luego la opción más próxima que comienza con la segunda tecla. Si vale *Verdadero*, si se pulsan velozmente varias teclas, busca la opción más próxima que comienza con esas teclas, VFP las interpreta como en el caso de *Falso*. Así, si una lista tiene su primera columna formada por apellidos y queremos buscar uno que empiece con *S*, basta oprimir

la tecla S: aparecerá seleccionado el más próximo que empiece con S. Esto se dará siempre, sin importar el valor de INCREMENTALSEARCH. Si queremos un apellido que empiece con *So*, INCREMENTALSEARCH deberá ser *Verdadero*: basta pulsar rápido las teclas S y O y aparecerá seleccionada la opción más próxima que empiece con So.

No es necesario que la primera columna esté ordenada, pero si lo está la búsqueda es más efectiva. En el ejemplo anterior, si los apellidos no están ordenados, los que empiecen con S pueden estar dispersos en opciones lejanas de la lista, lo cual no facilita buscar el deseado. Pero si están ordenados, todos los que empiecen con S estarán en opciones contiguas, lo cual facilitará hallar el que se quiere. Cuando la lista se construye con campos de una tabla, para ordenarlos por la primera columna se usa un índice cuya expresión sea el campo que aparece en esa columna. Para activar el índice, se define su nombre en la propiedad ORDER del cursor que contiene esa tabla.

InputMask

INPUTMASK (máscara de entrada) establece características que se desean para el contenido de un campo o variable mostrado o editado en un control, posición por posición. Para ello usa una serie de códigos, los más usuales de los cuales se muestran en la Figura 2.12. Como se explicó en la propiedad FORMAT, estos códigos se usan también en otros ámbitos.

Los códigos \$, \$\$ y , (coma) no son parte del dato, sino intercalaciones visuales para hacerlo más legible. No se almacenan con el dato. Si hace falta usar caracteres intercalados que no sean éstos, sino a gusto del usuario, hay que advertírselo a VFP. Para ello se usa el código R en la propiedad FORMAT.

Para esta propiedad, el punto representa la coma decimal y la coma el separador de miles. Que se visualicen de otra manera depende de cómo se configuren en HERRAMIENTAS – OPCIONES – REGIONAL.

Código Efecto

- **X** Posición que admite cualquier carácter.
- 9 Posición que sólo acepta números y los signos + y –.
- A Posición que sólo acepta letras.
- # Posición que sólo acepta un espacio, número o los signo + y -.
- \$ Muestra el símbolo monetario en la posición donde aparece.
- **\$\$** Muestra el símbolo monetario en forma flotante, es decir, precediendo inmediatamente a la primera cifra izquierda de un dato numérico.
- . Representa dónde va la coma decimal, para que coincida con la del dato.
- , Separador de miles que se intercala en la posición donde aparece, excepto que haya ceros no significativos. No es parte del dato.

Figura 2.12. Códigos de máscara de entrada

KeyboardHighValue

KEYBOARDHIGHVALUE (valor máximo por teclado) es una propiedad de SPINNER. Establece el mayor valor que se puede introducir al cuadro de texto con el teclado. Si se escribe un valor superior, se rechaza.

KeyboardLowValue

KEYBOARDLOWVALUE (valor mínimo por teclado) es una propiedad de SPINNER. Establece el menor valor que se puede introducir al cuadro de texto con el teclado. Si se escribe un valor inferior, se rechaza.

Left

Vea Propiedades de tamaño y ubicación.

LineSlant

LINESLANT (pendiente de la línea) es una propiedad de LINE. Afecta líneas oblicuas. Como la pendiente puede ser positiva o negativa, los valores admisibles son las barras \ y /, que las imitan gráficamente.

LinkMaster

Consulte Propiedades especiales de un GRID subordinado.

Movable

MOVABLE (movible) es una propiedad de COLUMN que permite cambiarla de lugar en el GRID, en tiempo de ejecución. Si es *Verdadero*, se puede mover; si es *Falso*, no. Si se mueve una columna, COLUMNORDER no varía. Al ejecutar nuevamente el formulario, la columna aparecerá en el lugar definido en la propiedad COLUMNORDER.

Name

NAME (nombre) da un nombre único a cualquier objeto, para que pueda ser mencionado sin ambigüedad en programación. Al crear un objeto con la barra de herramientas CONTROLES DE FORMULARIOS, toma un nombre inicial, formado por una palabra predefinida más un número que indica el orden de creación. Si el objeto se crea por arrastre desde el entorno, se usa un prefijo predefinido más el nombre del campo. Vea la Figura 2.13, donde se supone que el campo que se arrastra del entorno es *Precio*.

No confunda NAME con CAPTION. NAME es el nombre del objeto. Por ejemplo, NAME de un formulario podría ser *Empleados*. CAPTION es un texto que aclara para qué sirve el objeto y se visualiza en pantalla. Por ejemplo, CAPTION de ese formulario podría ser *Mantenimiento de empleados*.

PageCount

PAGECOUNT (cantidad de páginas o fichas) establece la cantidad de objetos PAGE de un PAGEFRAME. Si un PAGEFRAME ya tiene definidos PAGEs con objetos en su interior y se reduce el valor de PAGECOUNT, se pierden los PAGEs en exceso con todos sus objetos contenidos.

PageOrder

Cada PAGE tiene una propiedad PAGEORDER (orden de página o ficha), que da su posición relativa en el conjunto. Inicialmente los PAGEs y sus PAGEORDER se corresponden: el primer PAGE tendrá PAGEORDER 1, el segundo 2, etc. Pero el orden se puede variar. Si un PAGEFRAME tiene cinco fichas y se desea que la tercera sea la última, hay que hacer 5 su PAGEORDER. El de la cuarta ficha se hará 3 y el de la quinta 4.

El objeto se crea	Objeto	Nombre
Con la barra de herramientas	СнескВох	Check1
	СомвоВох	Combo1
	COMMANDBUTTON	Command1
	CommandGroup	CommandGroup1
	EDITBOX	Edit1
	LABEL	Label1
	LINE	Line1
	LISTBOX	List1
	Grid	Grid1
	OPTIONGROUP	OptionGroup1
	OptionButton	Option1
	Page	Page1
	PAGEFRAME	PageFrame1
	Shape	Shape1
	Spinner	Spinner1
	ТехтВох	Text1
Por arrastre	СнескВох	chkPrecio
	EDITBOX	edtPrecio
	LABEL	IblPrecio
	ТехтВох	txtPrecio

La numeración de los objetos de un contenedor comienza por 1.

Figura 2.13. Nombres de objetos por defecto

Propiedades especiales de un Grid subordinado

Es muy común que dos tablas se relacionen conceptualmente. Por ejemplo, la tabla con los datos de los clientes se relaciona con la tabla de los depósitos o extracciones que cada cliente realiza en su cuenta. Cada registro de CLIENTES puede tener varios registros en CUENTAS, porque cada cliente puede realizar varios movimientos. Recíprocamente, cada registro de CUENTAS tiene un registro relacionado en CLIENTES, pues un depósito o extracción no ha sido realizado sino por un único cliente. En ambos casos, hay algo en común en ambas tablas que permite la relación: en este caso, el número de cliente.

Cuando queramos hacer que VFP maneje esta relación, deberemos definirla. Una forma de hacerlo es en el entorno de datos, arrastrando un campo de una tabla a un índice de la otra; también se puede arrastrar un campo de una tabla a un campo de otra, siempre que ésta última tenga un índice por tal campo. En ambos casos, el campo de la primera tabla debe corresponder a la expresión por la que está construido el índice de la otra y el índice no debe ser cualquiera, sino el que contemple ese algo en común. Llamemos *parent* a la primera tabla y *child* a la segunda. Parent y child son conceptos relativos a la relación: parent es la tabla de donde tal relación parte y child es la tabla donde termina. Al trazar una relación en el entorno de datos se crea un objeto RELATION y se establecen automáticamente cuatro de sus propiedades: PARENTALIAS, CHILDALIAS, CHILDORDER y RELATIONALEXPR. Cuando la expresión del índice de la tabla child no es un campo, sino algo más complejo, como funciones sobre uno o más campos, la definición de RELATIONALEXPR es más laboriosa.

Una aplicación importante de las relaciones es la que sigue. Sea un formulario que muestre campos del
registro actual de una tabla parent en objetos apropiados y que permita cambiar de registro. En un GRID el cambio es fácil, porque permite ir a cualquier registro. En un LISTBOX o COMBOBOX, su lista debe estar construida con campos de una tabla: si bien lo que muestra la lista son opciones, no registros, al seleccionar una opción VFP salta automáticamente al registro que ha creado esa opción. Si el registro actual se muestra mediante TEXTBOX, EDITBOX o SPINNER, el formulario deberá tener botones que permitan ir al registro siguiente, al anterior, etc.

Si queremos que, cada vez que cambiemos de registro en la tabla parent, se muestren los registros relacionados de la tabla child, lo más efectivo es usar un GRID para ésta. Llamémoslo *GRID subordinado*. El GRID subordinado se logra de dos formas. Una forma es trazar una relación en el entorno, como se ha explicado. La otra forma es prescindir de la relación en el entorno pero definir tres propiedades del GRID, según resume la Figura 2.14.

Propiedad	Valor
CHILDORDER	Índice de la tabla child. Es <i>imprescindible</i> , pues permite encontrar al instante los registros child de cada registro parent
LINKMASTER	Nombre de la tabla parent.
RELATIONALEXPR	Expresión con campos de la tabla parent que "imita" la expresión del índice.

Figura 2.14. Propiedades especiales de un GRID subordinado

Repitámoslo porque es muy importante: el GRID subordinado se puede lograr relacionando las tablas en el entorno o en el mismo GRID, como muestran las Figuras 2.15 y 2.16.

Un GRID subordinado también puede mostrar campos de varias tablas. Para ello, las tablas que intervienen necesitan de una segunda relación: la tabla child de la primera relación será parent de esta segunda relación, cuyo child será una tercera tabla. En la propiedad RECORDSOURCE del GRID se debe definir la que es parent de la segunda relación. Esta relación no es para subordinar el GRID a otra tabla, sino para relacionar las tablas a mostrar en él. Por ello, debe ser definida exclusivamente en el entorno de datos. La Figura 2.17 ejemplifica lo que se dice en este párrafo.



Figura 2.15. GRID subordinado por una relación definida en el entorno de datos



Figura 2.16. GRID subordinado por propiedades de subordinación de él mismo



Figura 2.17. GRID subordinado que muestra campos de dos tablas

En la Figura 2.17, Tabla2 subordina a una sola tabla. También podría subordinar a varias, mostrando en el GRID subordinado campos de todas ellas, como muestra la Figura 2.18.



Figura 2.18. GRID subordinado que muestra campos de tres tablas

Hay otra posibilidad. Sea que Tabla1 subordina a Tabla2 y que ésta se muestre en GRID1 subordinado. Tabla2 está relacionada a Tabla3, que se muestra en GRID2 subordinado. Al cambiar de registro actual en Tabla1, cambian los registros mostrados por GRID1 y GRID2. Al cambiar de registro en Tabla2, desplazándose por GRID1, cambian los registros mostrados por GRID2. Lo dicho se esquematiza en la Figura 2.19.



Figura 2.19. GRID subordinado a otro GRID subordinado

Propiedades de fuente

Estas propiedades definen las variedades de fuente de todos los objetos que usan texto. Por ejemplo, en la propiedad CAPTION, en el contenido de un TEXTBOX, SPINNER, EDITBOX, etc., en un LISTBOX y COMBOBOX. La Figura 2.16 da una lista de estas propiedades y su uso.

Propiedad	<u>Uso</u>
FONTBOLD	Activa o desactiva negritas.
FONTITALIC	Activa o desactiva cursivas.
FONTNAME	Establece el tipo de fuente.
FONTSIZE	Establece el tamaño de fuente.
FONTUNDERLINE	Activa o desactiva subrayado.

Figura 2.16.	Propiedades de fuente
--------------	-----------------------

Propiedades de color

Estas propiedades definen el color de fondo ("papel") y de frente (fuente o borde, es decir "tinta") de varios objetos. En la Figura 2.17 se listan las propiedades y su uso.

Propiedad	<u>Uso</u>
BACKCOLOR	Establece el color de fondo.
FORECOLOR	Establece el color de frente.

Figura 2.17. Propiedades de color

Propiedades de tamaño y ubicación

Estas propiedades definen el tamaño de todo objeto gráfico y su ubicación dentro de su contenedor. Los contenedores que interesan a efectos de la ubicación son FORM y PAGE, porque permiten poner los objetos que contienen en el lugar que se desee. La Figura 2.18 lista las propiedades y lo que establecen. Note que un objeto queda determinado si se sabe dónde está y qué tamaño tiene. Como todo objeto ocupa un área rectangular (una línea inclinada es la diagonal de un rectángulo), dando las coordenadas de su vértice superior izquierdo, su ancho y su altura, queda determinado con precisión dentro del contenedor.

	Propiedad	<u>Establece</u>
Tamaño	HEIGHT	La altura del objeto
	WIDTH	El ancho del objeto
Ubicación	Left	La distancia entre ese vértice y el borde izquierdo de su contenedor
	Тор	La distancia entre ese vértice y el borde superior de su contenedor

Figura 2.18. Propiedades de tamaño y ubicación

ReadOnly

READONLY (sólo lectura) permite o impide que el contenido de un objeto, aunque visible y accesible, pueda ser modificado. Sus valores son *Verdadero* y *Falso*.

Esta propiedad es admitida también por el objeto CURSOR del entorno de datos de un formulario. Si un CURSOR se define con READONLY *Verdadero*, ninguno de los objetos que usen campos de la tabla contenida en tal cursor podrá ser modificado en el formulario, independientemente de cómo se hayan definido sus propios READONLY.

En el caso de GRID, tienen READONLY el GRID, sus COLUMNS y los objetos componentes de cada COLUMN (excepto HEADER, que carece de esta propiedad). Inicialmente, estos objetos tienen READONLY *Falso*. Si READONLY del GRID se hace *Verdadero*, este valor se trasmite a igual propiedad de todos los componentes. Esto no impide definir posteriormente uno en particular como *Falso*.

El valor READONLY de un COLUMN también se trasmite a igual propiedad del objeto contenido, la cual puede ser definida con el valor opuesto. En este caso, si el READONLY de COLUMN o del objeto contenido es *Verdadero*, el objeto será de sólo lectura.

RecordMark

RECORDMARK (marca de registro) hace que un GRID muestre o no la columna que indica cuál es el registro actual, mediante el símbolo ►. Los valores admisibles son *Verdadero* y *Falso*.

RecordSource

RECORDSOURCE (origen de registros) establece cuál es la tabla usada por el GRID. Si el GRID usa una sola tabla, será ella. Si usa varias tablas relacionadas, como las relaciones deben partir de una de ellas, debe establecerse esta tabla inicial.

RelationalExpr

Consulte Propiedades especiales de un GRID subordinado o el objeto RELATION.

Resizable

RESIZABLE (ancho modificable) pertenece a COLUMN. Si vale *Verdadero*, permite que el ancho de la columna pueda ser modificado en tiempo de ejecución. Si vale *Falso*, no. El cambio de ancho vale para la ejecución actual, pero no afecta la propiedad WIDTH.

RowHeight

ROWHEIGHT (altura de las filas) pertenece a GRID. Acepta un valor numérico, que establece la altura común de las filas que muestran los registros.

RowSource

ROWSOURCE (origen de las filas) define los contenidos de las opciones de una lista. Se aplica a COMBOBOX y LISTBOX. La forma de definir estos contenidos depende de lo que establezca la propiedad ROWSOURCETYPE. Veamos los casos más usuales.

RowSourceType se establece como 1 – Valor. ROwSOURCE debe llevar una lista de valores literales, separados por comas. Por ejemplo:

Enero, Febrero, Marzo, Abril, Mayo, Junio

Cómo se interprete este listado depende de la propiedad COLUMNCOUNT. Si COLUMNCOUNT es 1, el listado se interpretará como componentes de una sola columna:

Enero Febrero Marzo Abril Mayo Junio

Si COLUMNCOUNT es 2, el listado se interpretará como componentes de dos columnas:

Enero	Febrero
Marzo	Abril
Mayo	Junio

Si COLUMNCOUNT es 3, el listado se interpretará como componentes de tres columnas:

Enero	Febrero	Marzo
Abril	Mayo	Junio

RowSourceType se establece como 6 – Campos. ROWSOURCE debe incluir los nombres de tantos campos como columnas se deseen en la lista, separados por comas. El primer campo debe ser precedido por el nombre de la tabla y un punto. Así, si se quieren tres columnas, correspondientes a los campos ART, NOM y PRE de la tabla ART, ROWSOURCE debe ser

Art.Art,Nom,Precio

RowSourceType

ROWSOURCETYPE (tipo de origen de las filas) establece de dónde se tomarán las opciones (componentes para filas y columnas) de un COMBOBOX o LISTBOX. Los tipos más importantes son los que siguen.

1 - Valor. Los valores de las opciones, columna a columna, se escriben en la propiedad ROWSOURCE.

2 – Alias. Los valores de cada columna corresponden a los campos de una tabla o vista, en el orden estructural. Las opciones se forman con los registros de esa tabla o vista. La cantidad de columnas se define mediante COLUMNCOUNT: los campos se toman en el orden que tienen en la estructura, hasta agotar la cantidad definida de columnas. En ROWSOURCE sólo se define la tabla o vista, sin indicar campos.

4 – Consulta. Funciona como el caso 2 - Alias, pero ROWSOURCE establece el nombre de una consulta.

6 – Campos. Funciona como el caso 2 – *Alias*, pero permite que los campos aparezcan en el orden deseado (no en el orden estructural). Por esta razón, en ROWSOURCE hay que listar los campos queridos, en el orden deseado. Es preciso prefijar la lista de campos con el nombre de la tabla o vista, seguido por un punto. Por ejemplo:

Clientes.Nombre,Domicilio,Saldo

ScrollBars

SCROLLBARS (barras de desplazamiento) hace que un GRID muestre o no barras de desplazamiento horizontal y vertical. Puede tomar cuatro valores:

- 0 Ninguno
- 1 Horizontal
- 2 Vertical
- 3 Ambos

EDITBOX también admite esta propiedad, pero sólo con los valores 0 y 2.

SelectOnEntry

SELECTONENTRY (seleccionar al entrar) selecciona o no todo el contenido de un objeto cuando éste adquiere el enfoque mediante el teclado. Se aplica a TEXTBOX, EDITBOX y COLUMN.

Sea un TEXTBOX cuyo contenido sea Armario 4 estantes. Si SELECTONENTRY es Verdadero, cuando adquiera el enfoque mediante el teclado se verá escrito en letras blancas sobre fondo azul, similar a

Armario 4 estantes

¿Para qué sirve esto? Si quisiéramos cambiar el contenido, podremos comenzar a escribirlo sin tener que borrar el anterior: apenas pulsemos la primera tecla del nuevo contenido, el anterior se borrará automáticamente. Si SELECTONENTRY fuera *Falso*, no habría selección inicial de todo el contenido, por lo que, si guisiéramos reemplazarlo, deberíamos borrarlo manualmente.

SpinnerHighValue

SPINNERHIGHVALUE (valor máximo por mouse) es una propiedad de SPINNER. Sirve para establecer el valor mayor que se puede introducir dando clic en ^.

SpinnerLowValue

SPINNERLOWVALUE (valor mínimo por mouse) es una propiedad de SPINNER. Sirve para establecer el valor menor que se puede introducir dando clic en 🔽.

Style

STYLE (estilo) es la propiedad de COMBOBOX que establece si el objeto se comportará como lista desplegable o como cuadro desplegable. En el primer caso, no es posible escribir datos en el cuadro de texto componente. En el segundo caso, sí. Los nuevos valores escritos pueden agregarse a la lista de opciones mediante programación con métodos. Como este aspecto escapa a este curso, la diferencia establecida por esta propiedad no es significativa.

TabIndex

A medida que se crean objetos en un formulario, se les va dado un número creciente, comenzando en 1. Este valor se almacena en la propiedad TABINDEX (índice de tabulación). Al ejecutar el formulario, si vamos enfocando los objetos mediante las teclas TAB o SHIFT + TAB, el objeto al que se vaya depende de tal índice. Para terminar el ingreso de un dato a un objeto, suele darse ENTER. Esta tecla actúa como TAB, pasando el enfoque al objeto siguiente según el índice de tabulación.

Dado que los objetos de un formulario pueden haberse creado a medida que se nos ocurrían, sus índices de tabulación provocan que los saltos de un objeto a otro mediante teclas sean erráticos. Para arreglar este problema, hay que cambiar las propiedades TABINDEX de los objetos. Esto puede hacerse en cada uno, pero la forma fácil es mediante el botón ESTABLECER ORDEN DE TABULACIÓN de la barra de herramientas DISEÑADOR DE FORMULARIOS o el comando ORDEN DE TABULACIÓN del menú VER.

Si damos enfoque a los objetos haciendo clic, el orden de tabulación no es significativo. Pero en la vida real de las oficinas, para ingresar grandes volúmenes de datos, se prefiere manejar un formulario exclusivamente con el teclado. De este modo el operador no tiene que retirar las manos de las teclas para tomar el mouse y enfocar otro objeto. En estas situaciones, el orden de tabulación tiene gran importancia.

El índice de tabulación es relativo al objeto contenedor, sea éste FORM, PAGE, OPTIONGROUP u OPTIONBUTTON. Esto quiere decir que cuando se está en FORM, se pueden cambiar los valores TABINDEX de los objetos que contiene, sean contenedores o controles, en un primer plano de numeración. Si se entra al interior de un objeto contenedor del formulario, los objetos contenidos tienen un segundo plano de valores TABINDEX. Por ejemplo, si un FORM tiene un TEXTBOX, un GRID, un OPTIONGROUP con tres OPTION-BUTTONS y un COMMANDGROUP con cinco COMMANDBOUTTONS, en el primer plano habrá cuatro objetos con sus respectivos TABINDEXes, que variarán de 1 a 4. Si se entra al OPTIONGROUP, en este segundo plano hay tres objetos con sus respectivos TABINDEXes, que variarán de 1 a 3. Si se entra al COMMANDGROUP, en este otro segundo plano hay cinco objetos con sus respectivos TABINDEXes, que variarán de 1 a 5.

Los COLUMNS de GRID carecen de esta propiedad, pero un papel similar lo cumple la propiedad COLUMNORDER de cada una.

Тор

Vea Propiedades de tamaño y ubicación.

Width

Vea Propiedades de tamaño y ubicación.

WordWrap

Esta propiedad pertenece al objeto LABEL. Si su valor es Falso, el texto se extiende en una sola línea, ajus-

tándose o no adecuadamente al largo del texto de CAPTION según AUTOSIZE sea Verdadero o Falso. Si el valor de WORDWRAP es Verdadero, el texto se acomoda en varias líneas dentro del tamaño con que se diseña el objeto. Es una propiedad importante cuando el texto es largo. Sea el texto *El resultado de este proceso se colocará en GASTOS.DBF*. La Figura 2.19 muestra un ejemplo de cómo se acomoda el texto en distintas situaciones.

AUTOSIZE	WordWrap	El objeto se ve
Falso	Falso	El resultado de este
Verdadero	Falso	El resultado de este proceso se colocará en GASTOS.DBF.
_	Verdadero	El resultado de este proceso se colocará en GASTOS.DBF.

Figura 2.19. Distintas disposiciones del texto de Caption dentro de un Label

Capítulo 3. Generadores de objetos de formulario

Hemos visto que hay varios tipos de objeto. Cada objeto tiene propiedades, algunas comunes a varios tipos y otras específicas de uno solo. Para crear un objeto, entonces, se requiere definir varias propiedades, sin olvidar las decisivas. Esto lleva bastante trabajo. Para facilitar la tarea, hay generadores para varios tipos de objeto. Lamentablemente, hay objetos que deben ser creados "a mano", porque carecen de estas herramientas.

Un generador es un cuadro de diálogo con varias fichas, donde se definen varias características de un objeto. Al aceptar lo establecido, estas características se convierten en valores de propiedades. Si luego examinamos las propiedades del objeto generado, podremos apreciar que sus valores corresponden a lo que se decidió en el generador. Para retocar un objeto existente, también se puede usar un generador.

Para llamar al generador de un objeto particular, se puede seguir alguno de estos procedimientos:

- Crear el objeto a partir de la barra de herramientas CONTROLES DE FORMULARIOS y luego dar clic secundario sobre él y elegir GENERADOR.
- Activar el botón BLOQUEO DEL GENERADOR en la barra de herramientas CONTROLES DE FORMULARIOS y luego crear distintos objetos. Si cualquiera de ellos tiene generador, éste se activará automáticamente tras la creación. Este efecto estará en vigencia mientras el botón de bloqueo esté activo.
- Dar clic secundario sobre un objeto ya creado y elegir GENERADOR.

Un generador pregunta por *algunas* características, no por todas. Si necesitamos refinar más un objeto, no podremos hacerlo con el generador, sino que deberemos establecer las propiedades correspondientes en la ventana PROPIEDADES. De aquí que sea decisivo conocer las propiedades más importantes. Cuando un generador necesita una nueva tabla, la agrega automáticamente al entorno de datos.

Tienen generador TEXTBOX, EDITBOX, COMMANDGROUP, OPTIONGROUP, LISTBOX, COMBOBOX y GRID. Veamos las fichas y opciones de cada uno de ellos.

Generador de TextBox

Ficha Formato

Tipo de datos. No afecta a ninguna propiedad. Los tipos que ofrece son: carácter, fecha, lógico y numérico. El tipo elegido tiene cierta independencia del tipo real del campo o de la variable definidos como CONTROLSOURCE, permitiendo tratar un dato numérico, por ejemplo, con características de caracteres. Sin embargo, se recomienda elegir el tipo de acuerdo al tipo de datos de ControlSource. Según el tipo que se elija, serán las demás opciones de la ficha. Estas, en orden alfabético, son las siguientes.

Activar en tiempo de ejecución. Corresponde a la propiedad ENABLED. Cuando es *Verdadero*, el TEXTBOX estará habilitado. A veces conviene impedir el acceso a este objeto, haciendo *Falso* tal propiedad. Por ejemplo, en un formulario donde se modifican los datos laborales de los empleados, se podría mostrar el contenido de los campos del nombre y del legajo, pero haciéndolos inalcanzables.

Fecha europea. Genera el símbolo E en la propiedad FORMAT. Se aplica a datos de tipo fecha y fecha – hora. Los datos serán presentados en formato europeo, sin importar el formato de fecha en vigencia.

Hacer de sólo lectura. Corresponde a la propiedad READONLY. Activar esta opción hará que el objeto se pueda enfocar, pero su contenido no se podrá modificar.

Máscara de entrada. Corresponde a la propiedad INPUTMASK. Se aplica a datos numéricos y de caracteres. Se pueden escribir códigos INPUTMASK e incluso códigos FORMAT, como se hace en la función Transform(). Estos últimos, sin embargo, se pueden generar con otras opciones.

Mostrar ceros iniciales. Genera el símbolo L de la propiedad FORMAT. Sirve para ver o no los ceros iniciales que completan el tamaño de un dato numérico.

Ocultar selección. Corresponde a la propiedad HIDESELECTION. Cuando esta opción esté inactiva, cualquier porción del texto que se seleccione dentro del TEXTBOX seguirá visible cuando éste pierda el enfoque. Para que no se produzca tal efecto, hay que activar esta opción.

Seleccionar al entrar. Genera el símbolo K de la propiedad FORMAT. Activar esta opción hará que todo el contenido del TEXTBOX se seleccione cuando se le dé enfoque con el teclado. Si el texto está seleccionado, cualquier tecla que se pulse lo reemplazará en su totalidad. Para anular la selección se usa HOME o END o se da clic dentro del control. Si no hay selección inicial y se quiere reemplazar el texto, hay que borrarlo con DELETE o BACKSPACE (retroceso).

Sólo caracteres alfabéticos. Genera el símbolo A de la propiedad FORMAT. Tildar esta opción permitirá ingresar únicamente letras a un dato de caracteres.

Usar SET DATE actual. Genera el símbolo D de la propiedad FORMAT. Se aplica a datos de tipo fecha y fecha – hora. Cuando se activa esta opción, el dato será visualizado de acuerdo al formato de fecha en vigencia. Éste se establece en FORMATO – OPCIONES – REGIONAL o con el comando *set date* en la ventana COMANDOS.

	Elec	ción en	Tipo de	datos
Opciones	Carácter	Fecha	Lógico	Numérico
Activar en tiempo de ejecución	Х	Х	Х	Х
Ocultar selección	Х	Х	Х	Х
Seleccionar al entrar	Х	_	_	-
Hacer de sólo lectura	Х	Х	Х	Х
Máscara de entrada	Х	_	Х	Х
Mostrar ceros iniciales	_	_	-	Х
Sólo caracteres alfabéticos	Х	_	_	_
Fecha europea	-	Х	-	-
Usar SET DATE actual	_	Х	_	_

El cuadro siguiente resume la disponibilidad de las opciones, según lo elegido en TIPO DE DATOS.

Ficha Estilo

Efectos especiales. Corresponde a la propiedad SPECIALEFFECT. Ofrece los valores:

3D. El TEXTBOX se ve con profundidad.

Normal. El TEXTBOX se ve plano.

Borde. Corresponde a la propiedad BORDERSTYLE. Los valores ofrecidos son:

Simple. El TEXTBOX tiene borde.

Ninguno. El TEXTBOX no tiene borde.

Alineación de caracteres. Corresponde a la propiedad ALIGNMENT. Los valores son:

Izquierda Derecha

Centro

Automático. La alineación depende de lo elegido en TIPO DE DATOS.

Ajustar al contenido del texto. Activar esta opción produce que el ancho del TEXTBOX sea adecuado a su contenido. Afecta a la propiedad WIDTH, que tomará un valor aproximado para mostrar el contenido en su totalidad. No tiene relación con la propiedad AUTOSIZE, pues TEXTBOX no la posee. Cuando se define una máscara de entrada en la ficha FORMATO, VFP estima el tamaño basándose en ella. En caso contrario, usa la longitud de campo definido en la propiedad CONTROLSOURCE. La fuente, su tamaño y estilo afectan el cálculo del ancho; pero como el generador no maneja estas propiedades, usa una fuente predeterminada.

Ficha Valor

Nombre de campo. Corresponde a la propiedad CONTROLSOURCE. Sirve para definir el campo o variable que muestra el TEXTBOX.

Generador de EditBox

Ficha Formato

Permitir al usuario insertar tabulaciones. Corresponde a la propiedad ALLOWTABS (permitir tabulaciones). Por defecto, la tecla TAB hace que se pierda el enfoque sobre un objeto y se vaya al siguiente, según el índice de tabulación. Esto impediría insertar tabulaciones en el texto del EDITBOX, por ejemplo, para sangrarlo. Para poder insertarlas se debe activar esta opción. En tal caso, para abandonar el EDITBOX y enfocar el objeto siguiente mediante el teclado se usa CTRL + TAB.

Agregar barras de desplazamiento vertical. Corresponde a la propiedad SCROLLBARS.

Las demás opciones coinciden con las ofrecidas por el generador de TEXTBOX. Para mayores detalles, vea la ficha FORMATO de tal generador. Tales opciones son:

SELECCIONAR AL ENTRAR OCULTAR SELECCIÓN ACTIVAR EN TIEMPO DE EJECUCIÓN HACER DE SÓLO LECTURA

Fichas Estilo y Valor

Son como las fichas de igual nombre del generador de TEXTBOX.

Generador de CommandGroup

Ficha Botones

Número de botones. Corresponde a la propiedad BUTTONCOUNT. Esta opción permite establecer la cantidad de botones de comando, entre 0 y 100.

Título. Es una columna cuyas líneas corresponden a la propiedad CAPTION de cada COMMANDBUTTON. Sirve para escribir el texto de cada botón.

Gráfico. Es una columna cuyas líneas corresponden a la propiedad PICTURE de cada COMMANDBUTTON. Sirve para definir la imagen a mostrar dentro de cada botón, que debe estar contenida en un archivo de mapa de bits (extensiones BMP o ICO). Un botón puede tener título y gráfico a la vez.

Ficha Distribución

Distribución de los botones. Permite definir si los botones estarán dispuestos horizontal o verticalmente. Afecta a las propiedades de tamaño y ubicación.

Espacio entre botones. Permite definir un valor constante de separación entre los botones. Afecta a las propiedades de tamaño y ubicación.

Estilo de borde. Corresponde a la propiedad BORDERSTYLE. Sirve para establecer el tipo de borde, que puede ser *Simple* o *Ninguno*.

Generador de OptionGroup

Ficha Botones

Número de botones. Corresponde a la propiedad BUTTONCOUNT. Sirve para definir la cantidad de botones de opción que tendrá el OPTIONGROUP.

Estándar. No corresponde a ninguna propiedad. VFP entiende que se van a definir botones estándar, para lo cual activa la columna Título y desactiva la columna GRÁFICO.

Gráfico. No corresponde a ninguna propiedad. VFP entiende que se va a definir botones con texto e imágenes, para lo cual activa las columnas Título y Gráfico.

Título. Sirve para escribir el texto de la propiedad CAPTION de cada OPTIONBUTTON . El texto aparece a la derecha de cada botón. Si se lo quiere a la izquierda, luego de generado el objeto hay que modificar la propiedad ALIGNMENT de cada OPTIONBUTTON .

Gráfico. Corresponde a la propiedad PICTURE. Sirve para establecer la ruta y archivo del mapa de bits a mostrar en cada botón. Un mismo botón puede tener un texto y un gráfico a la vez.

Ficha Distribución

Distribución de los botones. Establece si los botones están colocados vertical u horizontalmente. Afecta a las propiedades de tamaño y ubicación.

Espacio entre botones. Establece el espacio constante entre los botones. Afecta a las propiedades de tamaño y ubicación.

Estilo del borde. Corresponde a la propiedad BORDERSTYLE. Define el borde, que puede ser *Simple* o *Ninguno*.

Ficha Valor

Nombre de campo. Corresponde a la propiedad CONTROLSOURCE. Sirve para definir el campo o variable origen del OPTIONGROUP, que debe ser de tipo numérico o de caracteres.

Generador de ComboBox y ListBox

Ficha Elementos de la lista

Rellenar la lista con. Determina qué se usará para construir las opciones (es decir, las filas o líneas) de la lista. Nos interesa *Campos de una tabla o vista* y *Datos introducidos manualmente*. Según lo seleccionado, varían las demás opciones de esta ficha. Los tipos corresponde a los valores 1 y 6, respectivamente, de la propiedad ROWSOURCETYPE. No veremos la tercera opción, *Valores de una matriz*.

Opciones de Campos de una tabla o una vista

Bases de datos y tablas. Permite indicar si la tabla necesaria para construir el objeto forma parte de una base de datos o es una tabla libre. Para buscar entre las bases de datos o las tablas libre tiene un botón asociado, que activa el cuadro de diálogo ABRIR.

Lista de tablas. Muestra las tablas y vistas de la base de datos seleccionada o las tablas libres.

Campos disponibles. Presenta todos los campos de la tabla o vista seleccionada en la LISTA DE TABLAS.

Botones de dirección. Permiten mover campos entre los cuadros CAMPOS DISPONIBLES y CAMPOS SELECCIONADOS, en uno y otro sentido.

Campos seleccionados. Muestra los campos elegidos para construir las columnas de la lista. El orden en que se seleccionaron determina el orden de las columnas; pero éste puede alterarse usando los botones que están a la izquierda de cada campo seleccionado.

Opciones de Datos introducidos manualmente

Columnas. Sirve para establecer la cantidad de columnas que tendrá la lista.

Cuadrícula de columnas. Permite introducir los datos en filas y columnas. El ancho de las columnas puede ser modificado, arrastrando las líneas separadoras.

Ficha Estilo

Tridimensional. Corresponde al valor *3D* de la propiedad SPECIALEFFECT. El objeto parece tener profundidad.

Normal. Corresponde al valor Normal de la propiedad SPECIALEFFECT. El objeto se ve plano.

Combinado desplegable. Corresponde al valor *0* de la propiedad STYLE. Además de desplegar la lista, el objeto permitirá introducir una nueva opción en el cuadro de texto. Hace falta algo de programación.

Lista desplegable. Corresponde al valor 2 de la propiedad STYLE. Solamente se podrá desplegar la lista, no siendo posible introducir opciones nuevas en el cuadro de texto.

Si, activar la búsqueda incremental. Corresponde al valor *Verdadero* de INCREMENTALSEARCH. La lista seleccionará la opción de la primera columna que coincida con las letras introducidas por teclado.

No, desactivar la búsqueda incremental. Corresponde al valor *Falso* de INCREMENTALSEARCH. El objeto no buscará una opción al escribir varias letras en el teclado, sino que buscará una opción por letra.

Mostrar filas (sólo para LISTBOX). Permite establecer cuantas filas se quieren ver a la vez. El número introducido sirve para calcular el valor de la propiedad HEIGHT, basándose en la fuente predeterminada.

Ficha Distribución

Ajustar la anchura del cuadro combinado / cuadro de lista para mostrar todas las columnas. Da a estos objetos un ancho adecuado para mostrar las columnas completas, incluyendo la barra de desplazamiento vertical. Corresponde a la propiedad WIDTH.

Cuadrícula de columnas. Para *Campos de una tabla o vista*, la cuadrícula mostrará datos tomados de los registros reales. Para *Datos introducidos manualmente*, la cuadrícula muestra tales datos. En ambos casos, se puede modificar el ancho de cada columna, arrastrando el borde derecho de su encabezado. Los encabezados sirven de referencia: para el primer caso, se verá el nombre de cada campo; para el segundo, se verá *Columna1, Columna2*, etc. Los encabezados no aparecerán en el objeto generado, pues COMBOBOX y LISTBOX carecen de ellos. También permite modificar el alto del encabezado y de las filas, pero sólo para ver mejor el contenido en el generador. Si se da doble clic en una columna, ésta se pone en gris: esto indica que será invisible, pues tendrá ancho *0*.

Ficha Valor

Devolver el valor de. Corresponde a la propiedad BOUNDCOLUMN. Sirve para definir cuál es la columna relacionada con el campo o variable definido en CONTROLSOURCE. Para campos de una tabla, la columna se elige por el nombre del campo que la llenará. Para datos introducidos manualmente, se elige *Columna1*, *Columna2*, etc. En ambos casos, el resultado es un número de columna.

Nombre de campo. Corresponde a la propiedad CONTROLSOURCE. Permite definir el campo o variable que sirve como origen del COMBOBOX. Para una variable, se escribe su nombre en el cuadro. Para un campo, se escribe el nombre de la tabla, un punto y el nombre del campo. Esto último también se logra con un botón de tres puntos, que permite elegir la tabla necesitada y luego uno de sus campos. En caso de construir la lista con campos de una tabla, el campo definido en CONTROLSOURCE debe pertenecer a otra tabla.

Generador de Grid

Ficha Elementos de la cuadrícula

Es igual a la ficha ELEMENTOS DE LA LISTA del generador de COMBOBOX, bajo la modalidad Opciones de Campos de una tabla o una vista.

Ficha Estilo

Estilo. Afecta a varias propiedades del GRID y sus componentes. Ofrece varios estilos de GRID: *Profesio-nal, Estándar, Elegante* o *Doble carta*. La opción *Mantener el estilo actual* anula cualquier otra selección. La ficha brinda una presentación preliminar de cada estilo.

Ficha Distribución

Título. Corresponde a la propiedad CAPTION de cada HEADER. Inicialmente este cuadro presenta el nombre del campo como el encabezado de la columna seleccionada, pero permite modificarlo.

Tipo de control. Permite elegir un objeto para el campo que ocupa la columna. Para los campos de tipo carácter ofrece TEXTBOX y EDITBOX. Para los de tipo lógico, ofrece TEXTBOX y CHECKBOX. Para los de tipo numérico, ofrece TEXTBOX y SPINNER. Otras variedades útiles, como introducir un COMBOBOX en una columna, no son manejados por el generador, por lo que hay que diseñarlo más artesanalmente. **Presentación de cuadrícula**. Brinda una presentación preliminar del GRID, con datos tomados de la tabla. La presentación se actualiza si se cambian los títulos o los tipos de control. Se puede cambiar:

- El ancho de cada columna, arrastrando horizontalmente el borde derecho del encabezado correspondiente. Afecta la propiedad WIDTH de la columna modificada.
- La altura del encabezado, arrastrando verticalmente el extremo izquierdo del borde inferior. Afecta la propiedad HEADERHEIGHT del GRID.
- La altura de las filas donde se presentan los registros, arrastrando verticalmente el extremo izquierdo de la primera línea. Afecta la propiedad ROWHEIGHT del GRID.

Ficha Relación

Esta ficha sirve exclusivamente para un GRID subordinado.

Campo clave de la tabla primaria. Corresponde a las propiedades LINKMASTER y RELATIONALEXPR. Para subordinar la tabla del GRID a otra tabla parent, es necesario que un campo de ésta última produzca la subordinación. La opción que nos ocupa permite estipular la tabla y el campo. El nombre de la tabla genera la propiedad LINKMASTER y el nombre del campo la propiedad RELATIONALEXPR. Si hace falta subordinar por expresiones más complejas que el simple nombre del campo, la opción no lo permite, por lo que habrá que introducir manualmente el valor de RELATIONALEXPR después de generar el GRID.

Índice relacionado de la tabla secundaria. Establece el índice de la tabla definida como RECORDSOURCE del GRID, necesario para la subordinación. Esta opción corresponde a la propiedad CHILDORDER.

Segunda parte. Informes

Los informes requeridos por las organizaciones toman formas muy variadas. VFP facilita diseñar muchas de estas variedades, mediante un DISEÑADOR DE INFORMES. Esta herramienta sirve para crear un programa que, al ser ejecutado, producirá un informe impreso o en pantalla. La palabra *informe* es ambigua: se usa tanto para el programa como para lo que produce. El contexto donde se la emplea indicará a cuál de estos dos significados se refiere. Cosa similar ocurre con la palabra *formulario*, como seguramente el lector ya habrá advertido.

Note que los informes siempre son salidas. No es posible modificar los datos que muestran, los cuales se toman de archivos. Se podrá tachar y escribir en el papel, pero ello no modifica los datos residentes en los archivos. Para hacer esto, se usan otros recursos, como formularios, vistas actualizables, la ventana EXAMINAR, etc.

Los informes que maneja VFP se diseñan utilizando *bandas*. Hay distintos tipos de éstas, que se imprimen en diferentes circunstancias de la impresión: al comienzo y final del informe, al comienzo y final de cada página, por cada registro, etc. Dentro de las bandas se colocan *objetos de informe*, para imprimir textos fijos, el contenido de campos o variables, líneas, cuadros o imágenes. Al igual que los objetos de formulario, los objetos de informe tienen propiedades.

Esta Parte abarca los Capítulos 4 a 6. El Capítulo 4 trata de bandas, el Capítulo 5 de objetos y el Capítulo 6 de propiedades.

Capítulo 4. Bandas de informe

Los datos contenidos en un informe presentan ciertas regularidades. Por ejemplo, es normal que tengan textos al comienzo de cada página, como un encabezado, la fecha y el número de página; textos que describan los distintos datos listados, ya sea como cabeceras de columnas (similar a VER – MODO EXAMINAR) o junto a ellos (como en VER – MODO EDITAR); los datos en sí; totales de los datos numéricos al final de cada página o al final del informe; etc. Estas regularidades han permitido determinar distintas partes de los informes, a las que se denomina *bandas*. Las bandas son franjas horizontales que atraviesan la página, cuya altura puede ser tanta como se necesite. En el diseñador, cada banda va acompañada por una barra horizontal, por debajo y en color gris, que permite variar su altura.

Las bandas pueden necesitarse en pareja: si hay una banda para el comienzo de la página, puede haber otra para el final de ella; si hay una para el comienzo del informe, puede haber otra para el final. De hecho, muchas bandas aparecen en parejas, siendo las necesidades específicas de un problema las que determinan si se usarán ambas o una sola. La excepción son los datos que corresponden a cada registro a imprimir, que requieren una sola banda.

Encabezado de página, Detalle, Pie de página

VFP ofrece distintos tipos de bandas. Al diseñar un informe, el contenido de cada tipo de banda se define una sola vez. Al ejecutar el informe, cada tipo de banda produce impresiones las veces que haga falta a través del listado. Cuando se crea un nuevo informe, aparecen tres bandas, llamadas ENCABEZADO DE PÁGINA, DETALLE y PIE DE PÁGINA. La primera y la última forman pareja. Las bandas ocuparán espacio en la página si se les da una altura mayor a 0. Para que tengan contenido, se colocan objetos dentro de ellas. Con las tres bandas iniciales se puede imprimir informes como los ejemplificados a continuación.

El informe de la Figura 4.1 es muy pobre. Como no hay objetos en las bandas ENCABEZADO DE PÁGINA ni PIE DE PÁGINA, no se sabe a qué se refiere el informe. La única banda usada ha sido DETALLE, que lista, supongamos, los campos NOMBRE y DEUDA de los registros de una tabla llamada DEUDORES.

Informe resultante		Producido por la banda
		Encabezado de página
Castro, José Oscar	500	Detalle
Díaz, Eduardo	780	Detalle
González, Ana Rosa	1.072	Detalle
Rodríguez, Patricia	515	Detalle
		Pie de página

Figura 4.1. Informe que usa objetos solamente en la banda DETALLE

La banda DETALLE define cómo presentar los campos de cada registro. Dado que todos los registros tienen la misma estructura, esta banda se define una sola vez, aplicándose del mismo modo a todos ellos, cualquiera sea su cantidad. En el ejemplo, hay dos CAMPOS de informe en la banda DETALLE. Estos objetos definen, mediante expresiones, de dónde tomar los datos a listar. En el ejemplo, una expresión será *Deudores.Nombre* y la otra *Deudores.Deuda*. Ambos CAMPOS están a una misma altura horizontal; pero podrían ubicarse a distintas alturas, para lo cual la banda debería ser más alta. En la Figura 4.2 se muestra el contenido de las bandas para producir el informe anterior, tal como se verían en el diseñador.

▲ End	cabezado de página	
	Nombre	Deuda
▲ Det	talle	
▲ Pie	de página	

Figura 4.2. Bandas para producir el informe de la Figura 4.1.

El informe de la Figura 4.3 es mejor. Se ha agregado la banda ENCABEZADO DE PÁGINA, donde se indica de qué trata el informe, a qué fecha está realizado y cuáles son los conceptos listados.

Informe resultante			Producido por la banda	
	Informe de Morosos, al 10-09-199)	Encabezado de página	
	Cliente	Deuda		
	Castro, José Oscar	500	Detalle	
	Díaz, Eduardo	780	Detalle	
	González, Ana Rosa	1.072	Detalle	
	Rodríguez, Patricia	515	Detalle	
			Pie de página	

Figura 4.3. Informe con objetos en las bandas ENCABEZADO DE PÁGINA Y DETALLE

La banda ENCABEZADO DE PÁGINA ahora tiene contenido. Hay dos ETIQUETAS, que se ven como *Cliente* y *Deuda*. Las ETIQUETAS son otro tipo de objeto, consistentes en texto, similares al objeto LABEL de los formularios. Hay un objeto LÍNEA, que subraya las ETIQUETAS. Finalmente está el título superior, compuesto por la parte fija *Informe de Morosos al* y la parte variable *10-09-1999*. Se puede lograr de dos maneras:

- 1. Con una ETIQUETA cuyo texto sea *Informe de Morosos al* y un CAMPO cuyo contenido sea la función *date()*. Esta solución es clara, pero necesita que ambos objetos se coloquen a una distancia tal que parezcan formar un solo texto.
- Con un solo CAMPO, cuyo contenido sea "Informe de Morosos, al " + dtoc(date()). Esta expresión es de caracteres, que une el texto entre comillas a la fecha convertida a caracteres mediante la función dtoc(). Da mejores resultados que la solución anterior, porque logra un acabado uniforme.

En la Figura 4.4 se muestra el contenido de las bandas para producir el informe anterior, tal como se verían en el DISEÑADOR DE INFORMES.

	"Informe de Morosos, al "	+ dtoc(date())	
	Cliente	Deuda	
▲ Encabez	zado de página		
	Nombre	Deuda	
▲ Detalle			
A Pie de p	ágina		

Figura 4.4. Bandas para producir el informe de la Figura 4.3.

Producido por la banda

El informe de la Figura 4.5 usa la banda PIE DE PÁGINA para el total de deuda por página.

Informe resultante

Informe de Morosos, al 10-09-1999		Encabezado de página
Cliente	Deuda	
Castro, José Oscar	500	Detalle
Díaz, Eduardo	780	Detalle
González, Ana Rosa	1.072	Detalle
Rodríguez, Patricia	515	Detalle
Total de página	2.867	Pie de página

Figura 4.5. Informe usando las bandas ENCABEZADO DE PÁGINA, DETALLE Y PIE DE PAGINA

En la banda PIE DE PÁGINA hay una ETIQUETA y un CAMPO. La primera muestra el texto *Total de página*. El segundo muestra la suma de las deudas listadas en la página. Este dato no está en la tabla DEUDORES, pero el informe lo puede calcular a partir del campo DEUDA. Para ello, la expresión del CAMPO será *Deudores.Deuda*, indicando que debe sumarse para cada página. La Figura 4.6 muestra el contenido de las bandas para producir el informe anterior.

"Informe de Morosos,	al " + dtoc(date())
Cliente	Deuda
▲ Encabezado de página	
Nombre	Deuda
▲ Detalle	
Total de página	Deuda
▲ Pie de página	

Figura 4.6. Bandas para producir el informe de la Figura 4.5.

El informe podría todavía mejorarse más. Por ejemplo, el encabezado de página podría ir dentro de un rectángulo, tener una fuente mayor, usar distintos colores, etc. Estos son aspectos de estética que hacen más agradables los informes y contribuyen a su funcionalidad.

Encabezado de columna, Pie de columna

El usuario puede agregar otros tipos de banda, según las necesidades. Supongamos un informe como el de la Figura 4.7. Podemos apreciar que hay un desperdicio de la hoja de papel, ya que una gran porción de la derecha queda desaprovechada.

	Códigos de Sucursales		
Sucursal	Código		
Ascochinga	345		
Azul	901		
Berisso	648		
Carlos Paz	165		
Chilecito	200		
Concordia	629		
Coronel Díaz	277		
Diamante	068		

Figura 4.7. Informe a una "columna"

Para evitar este desperdicio, se podría usar la mitad derecha de la hoja para información similar. Sería como poner dos páginas en una hoja, tal como muestra la Figura 4.8. Cuando se da esta situación, se dice que la página se ha dividido en dos "columnas". Cuidado: por columna, aquí, no debemos entender cada CAMPO con su correspondiente ETIQUETA de encabezado, sino la totalidad de objetos que conforman una banda DETALLE. Aunque la terminología sea confusa, la idea debe quedar clara. Para lograr el informe de la Figura 4.8 se hace necesario establecer un par de bandas adicionales: ENCABEZADO DE COLUMNA y PIE DE COLUMNA. Estas bandas se identifican por sus respectivas barras horizontales, que inicialmente aparecen con altura 0. Si se quiere usar cualquiera de ellas, es menester abrirlas y colocar los objetos necesitados. En la Figura 4.8, por ejemplo, solamente se ha usado la banda ENCABEZADO DE COLUMNA.

	Códigos de Sucursales				
Sucursal	Código	Sucursal	Código		
Ascochinga	345	Jáchal	715		
Azul	901	La Plata	112		
Berisso	648	Luján	629		
Carlos Paz	613	Olavarría	528		
Chilecito	135	Rafaela	379		
Concordia	200	Rosario	433		
Coronel Díaz	277	Salta	966		
Diamante	068	Tandil	257		

Figura 4.8. Informe a dos "columnas"

Nótese que, si se usan varias columnas, el tamaño horizontal de las bandas ENCABEZADO DE COLUMNA, DETALLE *y* PIE DE COLUMNA se reduce. En el ejemplo anterior, si vamos a tener dos columnas por hoja (dos páginas "virtuales" en la misma hoja física), estas bandas van a tener que entrar dos veces en el ancho de la hoja física. Pero como se definen una sola vez en el informe, su ancho va a reducirse a la mitad. La Figura 4.9 ejemplifica lo dicho.

Códigos de Sucursales		
▲ Encabezado de página		
Nombre	Código	
Encabezado de columna	1	
Nombre	Código	
▲ Detalle		
Pie de columna		
Pie de página		

Figura 4.9. Bandas del DISEÑADOR DE INFORMES con dos columnas

Los ejemplos de la Figuras 4.8 y 4.9 muestran que hay dos "columnas". Teniendo espacio, podría haber tres o más. La cantidad de columnas se establece con ARCHIVO – CONFIGURAR PÁGINA, mientras se diseña el informe.

Encabezado de grupo, Pie de grupo

Sea un informe que liste las ventas de un mes, ordenadas por vendedor, con un subtotal por cada vendedor. Para ello, la tabla VENTAS debe estar comandada por el índice VENDEDOR, que activaremos en el cursor de esta tabla en el entorno de datos, donde definiremos un filtro que muestre los registros de marzo 1999. También tendremos que usar INFORME – AGRUPAR POR. Este comando trae un cuadro de diálogo en el cual indicaremos agrupar los registros por la expresión *Ventas. Vendedor*. Esto generará otra pareja de bandas: ENCABEZADO DE GRUPO 1 y PIE DE GRUPO 1. La Figura 4.10 da un ejemplo de tal tipo de informe.

Informe resultante			Producido por la banda
Ventas por vendedor, corre	espondiente a marzo de 1999		Encabezado de página
Vendedor	Factura	Importe	
Álvarez, Pedro			Encabezado de grupo 1
	A000100034428	715	Detalle
	A000100034470	210	Detalle
	A000100034500	58	Detalle
	Subtotal	983	Pie de grupo 1
Becerra, Juan Florentino			Encabezado de grupo 1
	A000100034431	150	Detalle
	A000100034454	442	Detalle
	Subtotal	592	Pie de grupo 1

Figura 4.10. Informe de un grupo

Un grupo puede incluir un subgrupo; éste otro; y así sucesivamente. Por ejemplo, en el grupo de cada vendedor podemos hacer grupos por artículo; en cada uno de éstos podemos hacer grupos por fecha; etc. El agrupamiento, entonces, es jerárquico. VFP da los nombres *grupo 1*, *grupo 2*, *grupo 3*, etc., al grupo mayor, al subgrupo, al sub subgrupo, etc.

Si se trabaja con una sola tabla, un informe con varios niveles de agrupamiento necesita que esté indexada por múltiples campos. Si se trabaja con tablas relacionadas, algunos grupos pueden estar en la tabla parent y los restantes en la tabla child.

Los grupos se definen con INFORME – AGRUPAR POR o con el botón AGRUPAR DATOS de la barra de herramientas DISEÑADOR DE INFORMES. En el cuadro de diálogo, la jerarquía se representa verticalmente: el grupo de arriba es el de mayor amplitud; el que le sigue representa su subgrupo inmediato, etc. Cada jerarquía genera sus correspondientes parejas de bandas ENCABEZADO DE GRUPO y PIE DE GRUPO, las de mayor jerarquía incluyendo a las subordinadas, éstas a sus subordinadas, etc.

Título, Resumen

Las bandas que resta ver son TÍTULO y RESUMEN. Aunque conceptualmente formen pareja, no se generan simultáneamente, sino que, en el cuadro de diálogo traído por INFORME – TÍTULO / RESUMEN, se elige si se desea una, la otra o ambas. Estas bandas se imprimen una sola vez en el informe y es posible hacer que cada una ocupe una página entera.

La Figura 4.11 resume las distintas bandas, por jerarquía. Observe que la inclusión sucesiva de grupo 1, grupo 2, etc., puede llegar hasta nueve niveles de subordinación. También observe la inclusión total de un grupo menor en uno mayor: por ejemplo, el grupo 2 comienza luego del grupo 1 pero termina antes que él; el grupo 3 comienza luego de los grupos 1 y 2, pero termina antes que ellos.

Banda	<u>Una por cada</u>
Título	Informe
Encabezado de página	Página
Encabezado de columna	Columna
Encabezado de grupo 1	Grupo 1
Encabezado de grupo 2	Grupo 2
Encabezado de grupo 3	Grupo 3
Detalle	Registro
Pie de grupo 3	Grupo 3
Pie de grupo 2	Grupo 2
Pie de grupo 1	Grupo 1
Pie de columna	Columna
Pie de página	Página
Resumen	Informe

Figura 4.11. Tipos de bandas

Capítulo 5. Objetos de informe

Los informes están compuestos por bandas, dentro de las cuales se colocan objetos adecuados a lo que se necesite producir. Los objetos de informes son ETIQUETA, CAMPO, LÍNEA, RECTÁNGULO, RECTÁNGULO REDONDEADO e IMAGEN. Al igual que los objetos de formularios, tienen propiedades, que se establecen mediante cuadros de diálogo, uno para cada tipo de objeto. En este Capítulo veremos una descripción del aspecto y propósito de estos objetos.

Etiqueta

A Una ETIQUETA es similar en propósito al objeto LABEL de los formularios. No representa un contenido variable, sino un texto fijo. Las ETIQUETAS se usan para títulos del informe, títulos de página, títulos de columna, encabezados de CAMPO, etc. El texto de una etiqueta acepta que se pulse la tecla ENTER, que produce una nueva línea de texto dentro de la misma etiqueta.

Campo

Un CAMPO es un objeto que muestra el valor de una expresión. Como es un objeto de informes, que son salidas, no permite modificar tal valor. Por ello se puede asimilar a un TEXTBOX de sólo lectura. La expresión puede ser:

- Una expresión simple, como el nombre de un campo de una tabla, el nombre de una variable o una función. Por ejemplo, si la expresión es *Artículo*. *Nombre*, el campo del informe presentará el valor del campo NOMBRE del registro actual de la tabla ARTÍCULO. Si la expresión es la variable *_pageno*, el campo del informe presentará el número de la página actual del informe. Si la expresión es la función *date()*, el campo del informe presentará la fecha actual del sistema.
- Una expresión compleja, como funciones sobre campos de tablas o variables unidas mediante operadores. Por ejemplo, la expresión *Facturas.Cantidad * Facturas.Precio*, presentará el producto entre los campos CANTIDAD y PRECIO del registro actual de la tabla FACTURAS.
- Un cálculo sobre expresiones simples o complejas, realizado a medida que se procesa un conjunto de registros. En tal caso, el CAMPO se dice calculado. Por ejemplo, la expresión *Facturas. Cantidad * Facturas. Precio*, definida como una suma para un conjunto de registros, presentará la suma de los productos entre los campos CANTIDAD y PRECIO de los registros de la tabla FACTURAS que forman ese conjunto. Por conjunto puede entenderse la totalidad de registros de la tabla, los registros que entran en una página o columna, o los registros que forman un grupo. Para agrupar registros es necesario: (1) que los registros de la tabla FACTURA estén ordenados, por ejemplo, por FECHA; (2) que se defina la expresión de agrupamiento, para el ejemplo *Facturas.Fecha.* El valor final del CAMPO calculado se obtiene cuando termina el conjunto. Por ejemplo, para la expresión dada como ejemplo, la suma de CANTIDAD por PRECIO para todos los registros se obtiene cuando termina el proceso de la tabla. Para los registros que entran en una página o columna. Para el grupo correspondiente a la fecha *10-11-2003*, la suma final de CANTIDAD por PRECIO se obtiene al terminar ese día, es decir, cuando siga un registro con otra fecha.

Línea

Una LÍNEA es similar al objeto LINE de los formularios. A diferencia de éste, solamente puede ser horizontal o vertical, no admitiendo otras pendientes. Las LÍNEAS se usan para separar las distintas partes de un informe, dándoles claridad y estética.

Rectángulo

Un RECTÁNGULO es similar al objeto SHAPE de los formularios con CURVATURE 0. Como caso especial, puede adquirir la forma de un cuadrado. Los rectángulos se usan para enmarcar CAMPOS y ETIQUETAS, dándoles realce.

Rectángulo redondeado

Un RECTÁNGULO REDONDEADO es similar al objeto SHAPE de los formularios con CURVATURE mayor que 0. A diferencia de SHAPE, admite pocos casos de redondez.

Imagen

Una IMAGEN es similar al objeto IMAGE de los formularios. La imagen se puede tomar de un archivo BMP o PIC o de un campo de tipo general. No nos interesa profundizar este objeto.

La Figura 5.1 da un ejemplo de un informe con distintos objetos. Hay una LÍNEA, un RECTÁNGULO, un RECTÁNGULO REDONDEADO y una IMAGEN fácilmente identificables. Los textos *Cliente, Deuda* y *Total* son ETIQUETAS. Los nombres de los clientes y sus deudas han sido producidos por CAMPOS. El importe total ha sido producido por un CAMPO calculado. El título, que aparece dentro del RECTÁNGULO REDONDEADO, ha sido producido por un CAMPO, cuya expresión es *'Informe de Morosos, al '+ dtoc(date())*.

Cliente	Deud
Castro, José Oscar	50
Díaz, Eduardo	78
González, Ana Rosa	1.07
Rodríguez, Patricia	51

Figura 5.1. Informe con distintos tipos de objetos

Para crear un objeto en una banda, hay varios procedimientos:

• En la barra de herramientas CONTROLES DE INFORMES, primero dar clic en el objeto deseado y luego dar clic o hacer un arrastre en la banda donde se lo quiere colocar. Si es una ETIQUETA, hay que definir su texto. Si es un CAMPO, hay que definir su expresión de informe. Las demás propiedades son optativas. En el caso de CAMPO, si no se hace un arrastre en la banda, que toma un tamaño por defecto.

- En el entorno de datos, seleccionar un campo y arrastrarlo a la banda donde se lo quiere ubicar. Se genera un CAMPO cuya expresión de informe es el nombre del campo arrastrado. El tamaño del objeto depende del ancho del campo y de la fuente usada por defecto para el informe.
- Usar INFORME RÁPIDO del menú INFORME. Esto permite elegir una tabla o vista y los campos que se desean. También permite elegir si se quiere un CAMPO al lado del otro o uno bajo el otro. En el primer caso, los nombres de los campos se usan como ETIQUETAS en la banda ENCABEZADO DE PÁGINA y los campos como CAMPOS en la banda DETALLE, debajo de las ETIQUETAS respectivas. En el segundo caso, los nombres de los campos y los campos se transforman en ETIQUETAS y CAMPOS, respectivamente, dentro de la banda DETALLE. Cada ETIQUETA va junto al CAMPO correspondiente, formado pareja. Una pareja va debajo de la otra. Como INFORME RÁPIDO permite elegir campos de una sola tabla por vez, su utilidad es poca.

El menú FORMATO permite cambiar el aspecto de los objetos seleccionados. Así, para ETIQUETAS y CAMPOS se puede modificar fuente, estilo, tamaño, color de fuente, alineación del texto, color de fondo, transparencia u opacidad del fondo. Para LÍNEAS se puede alterar grosor, color y estilo. Para RECTÁNGULOS y RECTÁNGULOS REDONDEADOS se puede modificar borde y color de relleno.

Dos posibilidades importantes del menú FORMATO cuando se han seleccionado varios objetos son poder agruparlos o desagruparlos. No confunda esta agrupación con AGRUPAR DATOS del menú INFORME. La primera es una agrupación gráfica, de modo que los objetos se tratan como un todo cuando se los mueve y cuando varían dinámicamente de tamaño en tiempo de ejecución. La agrupación del menú INFORME se refiere a conjuntos de registros que tienen igual valor en uno o más campos, para lo que se necesita que estén ordenados por ellos, lo cual permite encabezar cada grupo y dar información resumen para cada uno.

Con respecto a la fuente que se va a emplear en los objetos que llevan fuente, se puede predeterminar con FUENTE PREDETERMINADA del menú INFORME. Esto produce que, en adelante, la fuente de las ETIQUETAS y CAMPOS sea la de la última predeterminación.

Capítulo 6. Propiedades de objetos de informe

A diferencia de los objetos de formulario, los objetos de informe no presentan una única ventana PROPIEDADES, sino un cuadro de diálogo para cada tipo. Para acceder al diálogo correspondiente a un objeto dado, se le da doble clic, o clic secundario y se elige PROPIEDADES. Si bien muchas propiedades se manejan de la forma expuesta, algunas se establecen mediante menús y barras de herramientas, como fuente, color, alineación, grosor, etc. Veamos los diferentes cuadros de diálogo para establecer propiedades. Siempre que sea posible, para facilitar su entendimiento, asociaremos estas propiedades a las de funcionamiento equivalente en los objetos de formularios.

Expresión de informe

Es el cuadro de diálogo principal del objeto CAMPO. Sirve para definir la expresión de un CAMPO, es decir, lo que se debe imprimir. La expresión puede ser sencilla, como el nombre de un campo o de una variable, o compuesta por funciones y operadores que se aplican a campos y variables. El cuadro de diálogo ofrece las propiedades que siguen.

Expresión. Equivale a la propiedad CONTROLSOURCE en formularios. Posee un cuadro de texto para escribir la expresión. Si el CAMPO ha sido creado por arrastre desde el entorno de datos, la expresión será el nombre del campo. Si ha sido creado usando la barra de herramientas CONTROLES DE INFORME, inicialmente carece de expresión. Junto al cuadro de texto hay un botón que activa el GENERADOR DE EXPRESIONES, que facilita escribir la expresión, porque brinda más espacio, permite elegir la tabla, vista, campos, funciones, operadores y variables necesitados, y verifica la corrección sintáctica de la expresión escrita.

Formato. Equivale a las propiedades FORMAT e INPUTMASK en formularios y usa los mismos códigos. Posee un cuadro de texto para escribir un formato para la expresión. Junto al cuadro de texto hay un botón que activa el cuadro de diálogo FORMATO (CAMPO), que facilita establecer los códigos deseados.

Cálculos. Activa el cuadro de diálogo CALCULAR CAMPO, mediante el cual se puede convertir el CAMPO en calculado.

Imprimir Condiciones. Activa el cuadro de diálogo IMPRIMIR CONDICIONES, donde es posible escribir condiciones que deben cumplirse para que se produzca la impresión del CAMPO.

Posición del campo. Es un grupo de tres opciones.

Flotante. Si sobre el CAMPO que se está configurando hay otro CAMPO con la propiedad AJUSTAR AL CONTENIDO DEL TEXTO activada, parte de él podría superponerse al que se configura. Para evitar este mal funcionamiento, el que se configura debería descender a medida que el otro crece hacia abajo. La propiedad FLOTANTE logra esto.

Borde superior de la banda. El CAMPO tiene una posición fija con respecto a ese borde.

Borde inferior de la banda. El CAMPO tiene una posición fija con respecto a ese borde. Si la banda aumenta o disminuye su altura en tiempo de ejecución, el CAMPO conservará la misma distancia con respecto a este borde. Es otra forma de hacerlo flotar.

Ajustar al contenido del texto. El ancho de un CAMPO es invariable en tiempo de ejecución.¿Cómo ver todo su contenido si no cabe en ese ancho? La solución es que el CAMPO crezca hacia abajo, usando tantas líneas de impresión como hagan falta hasta mostrar todo el contenido. Esta opción logra exactamente

ese comportamiento. Si hay objetos debajo del CAMPO que se configura, para evitar que éste se les superponga, deben ser establecidos con posición FLOTANTE o BORDE INFERIOR DE LA BANDA.

Para entender cómo funcionan AJUSTAR AL CONTENIDO DEL TEXTO, FLOTANTE y BORDE INFERIOR DE LA BANDA, consideremos la Figura 6.1.

[Campo 1]	
[Campo 2	Campo	53
De	etalle		

Figura 6.1. Informe cuyo CAMPO1 puede crecer hacia abajo

Sea que Campo 1 tiene activadas las propiedades BORDE SUPERIOR DE LA BANDA y AJUSTAR LA CONTENIDO DEL TEXTO. La primera propiedad hace que Campo 1 siempre comience en la misma posición con respecto al comienzo de la banda. Si su contenido no cabe en el ancho de este objeto, la segunda propiedad hace que se lo escriba en varias líneas, con lo que Campo 1 se acercará a Campo 2. Si campo 2 también fuera BORDE SUPERIOR DE LA BANDA, se podría provocar para algunos registros que las escrituras de Campo 1 y Campo 2 se superpusieran, produciendo un efecto sucio e ilegible. Para evitar este efecto, conviene que Campo 2 tenga propiedad FLOTANTE o BORDE INFERIOR DE LA BANDA. Si fuera flotante y las líneas de Campo 1 se le acercaran, Campo 2 se desplazaría hacia abajo, manteniendo la distancia separadora con la que han sido diseñados ambos objetos. Como consecuencia, el borde inferior de la banda estará también más abajo. Si la posición de Campo 2 fuera relativa al borde inferior de la banda, siempre se conservaría la distancia que hay entre él y ese borde. Al aumentar Campo 1 en virtud de que se escribe en varias líneas, el borde inferior de la banda descendería, arrastrando a Campo 2 hacia abajo.

Ahora consideremos Campo 3. Si queremos que siempre esté a la misma altura que Campo 2, no podríamos usar FLOTANTE, porque Campo 1 no lo amenaza con superponérsele. La única posibilidad, en este caso, es BORDE INFERIOR DE LA BANDA.

Comentario. Es un cuadro de edición para escribir texto que documente el objeto. El texto no aparece en la impresión del informe.

Calcular campo

Este cuadro de diálogo es exclusivo para el objeto CAMPO. Permite transformarlo en calculado. Un CAMPO calculado es aquél que incorpora una operación matemática o lógica sobre la EXPRESIÓN, como suma, promedio, mínimo, etc. Tiene sentido para un conjunto de registros, como por ejemplo todos los registros listados, los registros que forman un grupo, los que entran en una página, los que entran en una columna. Por esa razón, son muy usados en las bandas de pie y en la banda RESUMEN. En la banda DETALLE no tienen mucha aplicación, salvo para mostrar la acumulación que van sufriendo dentro de un conjunto, a medida que se imprimen los registros. En las bandas de encabezado de página y columna, sirven para repetir los valores alcanzados en los pies de página y columna anteriores, como se hace con el transporte en los libros diario y mayor. El cuadro de diálogo ofrece las propiedades que siguen.

Restablecer. Si queremos que un CAMPO sume su expresión dentro de cada página, sin transportar lo acumulado a la siguiente, su valor inicial al comenzar cada página deberá ser 0. Luego irá sumando el valor de la expresión de cada registro impreso en la página. Cuando ésta termine, en la banda PIE DE PÁGINA imprimiremos el CAMPO calculado. Al empezar a imprimir registros en la página siguiente, el CAMPO calculado volverá a 0 y se repetirá el proceso.

La opción RESTABLECER permite indicar cuándo poner a 0 el CAMPO calculado. Si no se usa agrupamiento, las posibilidades son *Fin del informe* y *Fin de página*. Si el informe tiene varias columnas, se agrega *Fin de columna*. Si se usa agrupamiento, se agregan tantas posibilidades como grupos haya.

Calcular. Abarca varias opciones que definen los cálculos a realizar sobre la expresión de un CAMPO. Las opciones son equivalentes a las *funciones de agrupamiento* que se usan en vistas y consultas. El cálculo se realiza para la totalidad de registros de un grupo, página, columna o informe, según lo indicado en RESTABLECER.

Nada. No se hace ningún cálculo en el CAMPO. Este no es, entonces, un CAMPO calculado.

Recuento. Cuenta el número de veces que se repite la expresión del CAMPO.

Suma. Calcula la suma de la expresión del CAMPO.

Promedio. Calcula la media aritmética de la expresión del CAMPO.

Mínimo. Encuentra el valor mínimo de la expresión del CAMPO.

Máximo. Encuentra el valor máximo de la expresión del CAMPO.

Desviación estándar. Calcula la desviación estándar de la expresión del CAMPO.

Varianza. Calcula la varianza de la expresión del CAMPO.

Formato (campo)

Este cuadro de diálogo es exclusivo para objetos CAMPO. Sirve para definir un formato, que afectará la visualización del contenido del CAMPO. El formato se expresa mediante códigos, iguales a los usados en FORMAT e INPUTMASK. El cuadro de diálogo ofrece las opciones que siguen.

Formato. Es un cuadro de texto para escribir códigos FORMAT e INPUTMASK. Como los códigos FORMAT pueden elegirse con las casillas de verificación que aparecen más abajo, el cuadro de texto suele dejarse para códigos INPUTMASK.

Tipo de dato. Es un grupo de tres opciones, que permite establecer cómo se va a tratar el contenido del CAMPO a los efectos del formateo. Esto es independiente de cuál es el tipo del contenido en sí. Según el tipo elegido, varían las opciones de edición ofrecidas.

Carácter. Indica que el CAMPO va a ser tratado como si fuera de caracteres.

Numérico. Indica que el CAMPO va a ser tratado como si fuera numérico.

Fecha. Indica que el CAMPO va a ser tratado como si fuera de fecha.

Opciones de edición. Es un grupo de casillas de verificación que actúan sobre el contenido del CAMPO. Varían de acuerdo a TIPO DE DATO. Cada casilla genera un código FORMAT. Los códigos producidos de esta manera se añadirán al comienzo del cuadro FORMATO, precedidos de @. Las casillas son las siguientes.

Mayúsculas. Presenta las letras del contenido en mayúsculas.

Ignorar máscara de entrada. Permite intercalar caracteres que no sean códigos, a gusto del usuario, entre los códigos INPUTMASK. El nombre de la opción es poco feliz, porque no indica lo que hace.

Alinear a la izquierda. Alinea el contenido a la izquierda.

Alinear a la derecha. Alinea el contenido a la derecha.

Alinear al centro. Centra el contenido.

En blanco si es cero. Si el contenido es 0, lo muestra en blanco.

(Negativo). Si el contenido es negativo, lo pone entre paréntesis.

Formato SET DATE. Muestra el contenido usando el formato de fecha en vigencia.

Formato de fecha europea. Muestra el contenido con ese formato, ignorando el vigente.

CR si es positivo. Si el contenido es positivo, muestra la sigla CR tras él.

DB si es negativo. Si el contenido es negativo, muestra la sigla DB tras él.

Ceros iniciales. Rellena con ceros iniciales un contenido numérico.

Moneda. Usa el formato de moneda vigente, establecido en HERRAMIENTAS – OPCIONES - REGIONAL.

Notación científica. Usa notación científica para mostrar un valor numérico.

Imprimir condiciones

Este cuadro de diálogo es común a todos los objetos, con variaciones según el tipo. Sirve para condicionar la impresión de un objeto. Ofrece las opciones que siguen.

Una vez por banda. Es un grupo de dos opciones, para objetos que no son CAMPOS. En la versión 5.0 aparece como *Imprimir valores repetidos*, lo cual es un error, pues estos objetos no contienen valores.

Si. El objeto se imprime siempre.

No. El objeto no se imprime, salvo que esté activa alguna opción de IMPRIMIR TAMBIÉN.

Imprimir valores repetidos. Es un grupo de dos opciones, para objetos CAMPO.

Si. Cuando el CAMPO contiene un valor que se repite en registros sucesivos, imprime tal valor para todos ellos. En el informe de la Figura 6.2, el CAMPO del cliente posee esta propiedad.

Ventas por cliente, enero de 1999			
Cliente	Factura	Importe	
Guardia, Pedro Daniel	B000100044117	1.345	
Guardia, Pedro Daniel	B000100045209	632	
Herrera, Alberto	A000100017749	2.807	
Herrera, Alberto	A000100017805	583	

Figura 6.2. Informe cuyo CAMPO izquierdo admite repetir valores

No. Cuando hay varios registros sucesivos para los cuales el valor del CAMPO se repite, sólo imprime el valor para el primero de tales registros. Vea el informe de la Figura 6.3.

Ventas por cliente, enero de 1999			
Cliente	Factura	Importe	
Guardia, Pedro Daniel	B000100044117	1.345	
	B000100045209	632	
Herrera, Alberto	A000100017749	2.807	
	A000100017805	583	

Figura 6.3. Informe cuyo CAMPO izquierdo no admite repetir valores

Imprimir también. Si bien el informe de la Figura 6.3 es más claro, es incómodo cuando los registros de un mismo cliente ocupan varias páginas. Salvo aquélla donde aparece el nombre por única vez, las demás páginas no lo llevarán. Si estamos consultando una de éstas, para saber de quién son las facturas debe-

remos hojear hacia atrás, hasta encontrar el nombre. La Figura 6.4 muestra el final de una página y el comienzo de otra. En la primera aparece *Herrera, Alberto*, que no se repite en esa página ni en la siguiente.

Pági	na n		Ventas	por cliente)
Guardia, Pedro	44117	1.345	Cliente	Factura	Importe
	45209	632		19721	168
Herrera, Alberto	17749	2.807		19806	232
	17805	583	Hinojosa, César	20545	2.105
	18472	2.870		20617	782
	19628	7.236	Pági	na n + 1	



IMPRIMIR TAMBIÉN soluciona la dificultad anterior. Tiene tres opciones que son excepciones a cuando se elige No en IMPRIMIR VALORES REPETIDOS o en UNA VEZ POR BANDA. Se aplica a todos los objetos.

En primera banda completa de nueva página / columna. Imprime el objeto cuando la banda que lo contiene cabe completa al principio de una nueva página (o columna). La Figura 6.5 queda, si se activa esta propiedad, como indica la Figura 6.6, que repite *Herrera, Alberto* en la página siguiente.

Pági	na n		Ventas	por cliente	<u></u>
Guardia, Pedro	44117	1.345	Cliente	Factura	Importe
	45209	632	Herrera, Alberto	19721	168
Herrera, Alberto	17749	2.807		19806	232
	17805	583	Hinojosa, César	20545	2.105
	18472	2.870	-	20617	782
	19628	7.236	Pági	na n + 1	

Figura 6.5. Modificación para imprimir el nombre al comienzo de una nueva página

Cuando cambie este grupo. Se activa cuando hay agrupamiento. El grupo que va a producir la impresión del valor repetido se elige en el cuadro de lista adjunto. Es importante entender bien cómo funciona este caso. Sea el informe de la Figura 6.6.

Ventas por cliente				
Cliente	Fecha	Importe		
Guardia. Pedro Daniel				
	10-05-1999	1.345		
		632		
	25-06-1999	196		
		2.173		
Herrera, Alberto				
	25-06-1999	2.807		
	17-07-1999	583		
		3.390		

Figura 6.6. Informe cuyo CAMPO izquierdo no admite repetir valores

El informe anterior está agrupado por cliente. En la banda ENCABEZADO DE GRUPO va el nombre de cliente y en la banda PIE DE GRUPO el total comprado por cada uno. Los CAMPOS que listan la fecha y el

importe van en la banda DETALLE. El CAMPO de fecha no imprime valores repetidos, lo cual se observa en la segunda compra de *Guardia, Pedro Daniel*, por 632 pesos. El mismo efecto podría suceder entre la última compra de este cliente y la primera de *Herrera, Alberto*, efectuadas el mismo día. Sin embargo, se ha considerado importante repetir la fecha, porque pertenece a clientes distintos. La regla general de no repetir valores iguales, entonces, ha sufrido una excepción. Esta se establece con IMPRIMIR TAMBIÉN - CUANDO CAMBIE ESTE GRUPO, indicando que tal grupo es el que corresponde al campo cliente.

Cuando detalle pase a nueva página / columna. Si la banda DETALLE tiene objetos a distinta altura, puede suceder que, al llegar al final de una página, se imprima en dos partes: una parte al final de esa página (o columna) y la otra al comienzo de la página (o columna) siguiente. Si en tales situaciones se desea que un objeto de la banda vuelva a ser impreso en la nueva página o columna, hay que activar esta opción. Veamos cómo se comporta el CAMPO del nombre de la empresa al final de una página y al comienzo de la opción inactiva (Figura 6.7) y activa (Figura 6.8).

Comercial del Sur	Empresa	
Tucumán 527	Domicilio	Banda completa
Córdoba	Provincia	
	(Empresa)	
Echeverría 107	Domicilio	
		Banda fraccionada en dos páginas
Jujuy	Provincia	
Buen Gusto	Empresa	
San Martín 145	Domicilio	Banda completa
Catamarca	Provincia	

Figura 6.7. Cambio de página con CUANDO DETALLE PASE A NUEVA PÁGINA / COLUMNA inactiva

Comercial del Sur	Empresa	••••••••••••••••••••••••••••••••••••••
Tucumán 527	Domicilio	Banda completa
Córdoba	Provincia	
	(Empresa)	
Echeverría 107	Domicilio	
		Banda fraccionada en dos páginas
Comercial del Sur	Empresa	
Jujuy	Provincia	
Buen Gusto	Empresa	
San Martín 145	Domicilio	Banda completa
Catamarca	Provincia	

Figura 6.8. Cambio de página con Cuando detalle pase a nueva página / columna activa

Quitar líneas en blanco. Cuando los objetos de una banda se disponen en varios niveles, puede que, para un registro, todos los que están en un mismo nivel resulten en blanco porque no tienen contenido. En tal caso, la impresión produciría "huecos" poco agradables. Para impedir ese efecto, se activa esta opción en todos los objetos de ese nivel. La banda, al eliminar la impresión de esos huecos, resultará menos alta.

Imprimir sólo cuando la expresión sea verdadera. Es un cuadro de texto donde se puede escribir una expresión lógica que condicione la impresión del objeto. En tiempo de ejecución, si la expresión es verdadera, el objeto se imprimirá; pero si resulta falsa, no. Cuando se usa esta opción, todas las anteriores quedan inhabilitadas, excepto QUITAR LÍNEAS EN BLANCO. Junto al cuadro hay un botón que invoca al GENERADOR DE EXPRESIONES, el cual facilita construir la expresión.

Rectángulo / Línea

Es el cuadro de diálogo principal de los objetos RECTÁNGULO y LÍNEA. Tiene algunas opciones similares a las del cuadro de diálogo EXPRESIÓN DE INFORME y otras específicas. Por ello, trataremos en detalle solamente éstas últimas.

Imprimir Condiciones. Activa el cuadro de diálogo de igual nombre, mediante el cual se controla cuándo hay que imprimir el RECTÁNGULO o la LÍNEA.

Posición del objeto. Vea el cuadro de diálogo Expresión de INFORME.

Alargar hacia abajo. Es un grupo de tres opciones que controlan si la altura del objeto será fija o variable.

No alargar. Hace que la altura del objeto sea fija.

Alargar con relación al objeto más alto del grupo. Cuando un RECTÁNGULO encierra un CAMPO cuya cantidad de líneas puede variar (AJUSTAR AL CONTENIDO DEL TEXTO activa), lo deseable es que el rectángulo se estire verticalmente con él. Esta opción logra ese efecto. Lo mismo puede lograrse con una LíNEA vertical que resalta un CAMPO de altura variable. El procedimiento a seguir es: (1) Activar esta propiedad en el RECTÁNGULO o la LÍNEA; (2) Seleccionar todos los objetos a encerrar o resaltar, incluyendo el RECTÁNGULO o la LÍNEA, y formar un grupo gráfico con FORMATO – AGRUPAR.

Alargar con relación al alto de la banda. Hace que la altura del RECTÁNGULO o la LÍNEA sea variable, pero no con relación a los objetos que contiene o resalta, sino con relación a la altura de la banda. Si la banda crece porque crecen los objetos contenidos, estos objetos se hacen más altos. Si la banda decrece porque se eliminan líneas en blanco, estos objetos se hacen menos altos.

Comentarios. Vea el cuadro de diálogo Expresión de INFORME.

Rectángulo redondeado

Es el cuadro de diálogo principal del objeto RECTÁNGULO REDONDEADO. Coincide en sus opciones con el cuadro de diálogo RECTÁNGULO / LÍNEA. Como propia, tiene la opción ESTILO, grupo de opciones gráficas que permite elegir diferentes tipos de esquinas redondeadas.

Texto

Es el cuadro de diálogo principal del objeto ETIQUETA. Tiene las opciones que siguen.

Posición del objeto. Vea el cuadro de diálogo Expresión de INFORME.

Imprimir Condiciones. Activa el cuadro de diálogo de igual nombre, mediante el cual se controla cuándo hay que imprimir la ETIQUETA.

Comentarios. Vea el cuadro de diálogo Expresión de INFORME.

Variables de informe

Hasta ahora hemos visto objetos CAMPO, calculados o no, cuyas expresiones estaban construidas con campos de tabla. Los campos calculados también se pueden construir con variables. Estas variables pueden haberse creado antes de ejecutar un informe, pero también se pueden crear en el informe, para almacenar valores y resultados de cálculos que se realizan durante la impresión. Las variables se pueden imprimir o no. Para imprimirlas, debe hacérselo a través de CAMPOS. En ocasiones, el uso de variables es necesario o conveniente, como se ilustra a continuación.

- Las variables permiten cálculos imposibles de lograr con CAMPOS. Por ejemplo, sea resolver la fórmula ∑(Ingreso) + 15000. Si usamos un CAMPO calculado cuya expresión sea el campo Ingreso y el cálculo Suma, obtendremos el resultado parcial ∑(Ingreso), sin posibilidad de sumarle 15000 al escribirlo. En cambio, si usamos la variable Resultado, podemos darle el valor inicial 15000 y ordenarle que sume el campo Ingreso de cada registro. Otra forma sería definir la variable Parcial que haga el cálculo Suma de Ingreso, como haríamos con un campo calculado, pero en el CAMPO que muestre el resultado final, escribir la expresión Parcial + 15000.
- Las variables simplifican la escritura de las expresiones a calcular. Así, para las fórmulas

 \sum (Ingreso1 + Ingreso2 + Ingreso3 + Ingreso4 + Ingreso5 + Ingreso6)

 \sum (Egreso1 + Egreso2 + Egreso3 + Egreso4 + Egreso5 + Ingreso6)

 \sum (Ingreso1 + Ingreso2 + Ingreso3 + Ingreso4 + Ingreso5 + Ingreso6)

 Σ ((Egreso1 + Egreso2 + Egreso3 + Egreso4 + Egreso5 + Ingreso6)

podríamos usar un CAMPO calculado para cada expresión. Pero la tercera sería difícil de escribir. Más fácil sería usar tres variables, digamos A, B y C, con las definiciones del cuadro siguiente:

Variable	Valor inicial	Expresión a almacenar	Cálculo
А	0	Ingreso1 + Ingreso2 + Ingreso3 + Ingreso4 + Ingreso5 + Ingreso6	Suma
В	0	Egreso1 + Egreso2 + Egreso3 + Egreso4 + Egreso5 + Ingreso6	Suma
С	0	A/B	Nada

El cálculo Nada de la última expresión hace que el cociente se realice cuando se la va a imprimir, con lo acumulado hasta ese momento en las otras dos expresiones.

Para crear, modificar y eliminar variables se usa el cuadro de diálogo VARIABLES DEL INFORME, que se invoca con INFORME – VARIABLES. Sus opciones son las siguientes.

Variables. Es una lista para crear variables, una por línea. Muestra las variables definidas y permite seleccionar cualquiera de ellas para editarla o eliminarla.

Almacenar valor. Es un cuadro de texto para escribir la expresión a almacenar en la variable seleccionada. Tiene un botón para llamar al GENERADOR DE EXPRESIONES.

Valor Inicial. Es un cuadro de texto para escribir el valor inicial de la variable seleccionada. Tiene un botón para llamar al GENERADOR DE EXPRESIONES.

Liberar tras informe. Es una casilla de verificación. Si se activa, la variable seleccionada va a eliminarse de memoria al terminar de imprimir el informe. Si no se activa, la variable permanecerá en memoria.

Reiniciar en. Es un cuadro combinado que permite establecer cuándo eliminar el valor alcanzado por la variable y reponer el valor inicial. Las opciones son FIN DEL INFORME, FIN DE PÁGINA Y FIN DE COLUMNA. Si el informe usa agrupamiento de registros, se agrega una opción por cada grupo.

Insertar. Es un botón de comando que inserta una línea en blanco en la lista VARIABLES, para intercalar la definición de una variable nueva.

Eliminar. Es un botón de comando que elimina la variable seleccionada de la lista VARIABLES.

Calcular. Es un grupo de opciones para establecer el cálculo con la expresión de la variable. Las opciones, ya conocidas, son: NADA, RECUENTO, SUMA, PROMEDIO, MÍNIMO, MÁXIMO, DESVIACIÓN TÍPICA, VARIANZA.

Tercera parte. Entorno de datos

Por lo general, los formularios e informes trabajan con tablas o vistas. Para ello, necesitan que las tablas a usar estén abiertas y que las vistas se ejecuten, que los índices que comandarán las tablas estén activos, que las relaciones estén establecidas, etc. Esta preparación de los archivos necesitados se puede lograr mediante programación, escribiendo comandos en los eventos apropiados del formulario o informe, lo cual requiere conocer comandos y eventos. Mucho más sencillo, y menos complicado para el principiante, es definir un *entorno de datos*. La confección de este entorno puede ser manual o automática.

En la confección manual se abre la ventana ENTORNO DE DATOS y se le agregan tablas o vistas, que generan cursores. Habiendo más de un cursor, se pueden trazar relaciones entre ellos. Hay, entonces, tres tipos de objeto: DATAENVIRONMENT (entorno de datos), CURSOR y RELATION. Cada uno posee propiedades, como por ejemplo el índice que comandará la tabla contenida en un CURSOR, el que un CURSOR sea de sólo lectura, la ampliación de una relación al tipo uno a varios, etc.

La confección automática se produce cuando: (1) se usa un generador de un objeto de formulario que requiera tablas o vistas; (2) se usa FORMULARIO – FORMULARIO RÁPIDO O INFORME – INFORME RÁPIDO. Ambos procedimientos agregan uno o dos cursores al entorno de datos, si todavía no están en él:

- Para TEXTBOX, EDITBOX y SPINNER, agregan un CURSOR.
- Para GRID subordinado, agregan dos CURSORES: uno el de la tabla subordinada que se verá en el GRID y otro el de la tabla que lo subordinará.
- Para LISTBOX o COMBOBOX, si la lista está construida con campos de una tabla, agregan un CURSOR. Si CONTROLSOURCE es el campo de otra tabla, agregan otro CURSOR.

Ambos procedimientos para confeccionar el entorno de datos no son mutuamente excluyentes. Debe advertirse que la confección automática requiere a veces el ajuste manual de ciertas propiedades.

Esta Parte tiene dos Capítulos. El Capítulo 7 trata sobre objetos de entornos de datos. El Capítulo 8 trata de propiedades de tales objetos.

Capítulo 7. Objetos de entorno de datos

Los entornos de datos tienen tres tipos de objetos: DATAENVIRONMENT, CURSOR y RELATION.

DataEnvironment

DATAENVIRONMENT es un objeto contenedor, que puede incluir objetos CURSOR y RELATION. Su aspecto es el de una ventana de color gris, que se puede mover y ajustar en tamaño. Para abrirlo se usa la opción ENTORNO DE DATOS del menú VER o del menú contextual (en el formulario o informe, donde no haya objetos), o se pulsa el botón de igual nombre de la barras de herramientas DISEÑADOR DE INFORMES y DISEÑADOR DE FORMULARIOS. DATAENVIRONMENT no se puede eliminar, sino solamente cerrar. Sí se puede eliminar cualquier objeto contenido, CURSOR o RELATION.

Cursor

CURSOR es un objeto que representa una tabla o vista. Su aspecto es el de un rectángulo pequeño, cuyo título es el alias de la tabla o vista, con un renglón para cada campo y para cada índice. Tiene una barra de desplazamiento vertical para llegar a las líneas no visibles. Se puede mover y cambiar de tamaño. Un CURSOR se crea cuando se agrega una tabla o vista mediante la opción AGREGAR del menú ENTORNO DE DATOS o del menú contextual (en la ventana de entorno, fuera de los objetos) o pulsando el botón AGREGAR del cuadro de diálogo AGREGAR TABLA O VISTA. También se puede crear por algunos generadores de objetos de formulario o por los comandos FORMULARIO RÁPIDO e INFORME RÁPIDO. Para eliminar un CURSOR, se lo debe seleccionar y pulsar DELETE, o elegir la opción QUITAR en el menú ENTORNO DE DATOS o en el menú contextual.

Relation

Este objeto establece una relación entre dos cursores por algo en común. RELATION nace en un campo de un CURSOR y termina en un índice adecuado de otro. Por índice adecuado queremos decir que su expresión es un campo del segundo CURSOR que corresponde al campo del primero. Llamemos *parent* (padre, madre) al primer CURSOR y *child* (hijo, hija) al segundo, palabras que aplicamos por extensión a sus respectivas tablas, vistas, campos e índices. El aspecto del objeto RELATION es el de una línea que une el campo parent con el índice child. RELATION se crea arrastrando el campo parent al índice child. También se puede crear arrastrando el campo parent a un campo child, siempre y cuando éste sea la expresión del índice adecuado. Para eliminar un objeto RELATION, se lo debe seleccionar y pulsar DELETE. Si la expresión del índice child es más compleja que el simple nombre de un campo, será necesario corregir, luego de trazada la relación, su propiedad RELATIONALEXPR.

El generador de un GRID subordinado también crea una relación entre la tabla que lo subordina y la tabla que se ve en él. Esta relación, sin embargo, no crea un objeto RELATION, porque existe implícitamente en las propiedades de subordinación del GRID.
Capítulo 8. Propiedades de entorno de datos

Para formularios e informes los objetos de entorno de datos son los mismos. Cada objeto tiene propiedades, métodos y eventos específicos. Solamente veremos algunas propiedades importantes de cada tipo de objeto. Las propiedades se consultan y modifican en la ventana PROPIEDADES, la misma que se usa con los objetos de formulario.

Propiedades de DataEnvironment

AutoCloseTables

AUTOCLOSETABLES (cierre automático de tablas) puede ser *Verdadero* o *Falso*. Si es *Verdadero*, al terminar la ejecución del formulario o del informe, VFP cierra automáticamente todas las tablas y vistas que fueron usadas por ellos. Si es *Falso*, las tablas y vistas permanecen abiertas.

AutoOpenTables

AUTOOPENTABLES (apertura automática de tablas) puede ser *Verdadero* o *Falso*. Si es *Verdadero*, al iniciar la ejecución del formulario o del informe, VFP abre automáticamente las tablas y vistas con las cuales van a trabajar. Si es *Falso*, las tablas y vistas se deben abrir previamente, usando otros procedimientos.

InitialSelectedAlias

INITIALSELECTEDALIAS (alias seleccionado inicialmente) establece cuál CURSOR se seleccionará al comenzar la ejecución de un formulario o informe. Más adelante veremos el concepto de alias, pero podemos pensarlo por ahora como CURSOR, o, más específicamente, como la tabla o vista de un CURSOR. Cuando se trabaja con varias tablas o vistas, todas deben estar abiertas, pero sólo una puede estar seleccionada en un momento dado. En un formulario, el alias seleccionado cambia al enfocar objetos cuyo CONTROLSOURCE sean campos de distintas tablas o vistas. En un informe esto no es posible: la selección de un único alias se realiza automáticamente al comienzo de la ejecución, y no cambia nunca. Lógicamente, si hay relaciones, debe ser aquel cursor que las comanda.

Si en el entorno de un informe hay tres tablas, A, B y C, y dos relaciones, pueden tener la forma



o la forma

En ambas situaciones, el alias que debe estar seleccionado al iniciar la ejecución del informe debe ser A, pues es la única que puede mover correctamente el mecanismo de las relaciones.

De modo predeterminado, el primer CURSOR que se agrega a un entorno hace que su alias sea el inicialmente seleccionado. Esto no siempre resulta, porque puede producir mal funcionamiento del mecanismo de relaciones. Veamos el ejemplo que sigue.

Sea que se quiere producir un informe como el de la Figura 8.1. El nombre de alumno está en la tabla ALUMNOS; la fecha, condición y nota en la tabla EXÁMENES y el nombre de materia en la tabla MATERIAS. Habrá que agregar estas tres tablas al entorno. También habrá que establecer un objeto RELATION1, por matrícula, de ALUMNOS a EXÁMENES, y otro RELATION2, por código de materia, de EXÁMENES a MATERIAS. De este modo, por cada alumno, RELATION1 buscará todos los exámenes rendidos por él; y por cada examen, RELATION2 buscará el nombre de la materia rendida. Es obvio que RELATION1 deberá ser de uno a varios, porque un alumno puede haber rendido varias materias.

Asignaturas rendidas en 1999, por nombre de alumno				
Alumno	Materia	Fecha	Cond.	Nota
Aguirre, Alberto	Administración I Costos	17-02-99 25-07-99	R R	8 7
	Derecho Civil	06-12-99	R	6
Arce, María	Estadísticas	10-03-99	R	10
	Contabilidad	25-11-99	L	5
Arturo, Julio	Sistemas de Información I	19-02-99	R	7
	Matemáticas II	03-07-99	R	8

Figura 8.1. Informe que se quiere obtener

Al diseñar el entorno, las tablas se agregaron en el orden ExÁMENES, ALUMNOS y MATERIAS. El alias inicial, entonces, es *Exámenes*, por ser el primer CURSOR agregado. Luego se completó el entorno, trazando correctamente RELATION1 y RELATION2. Esquemáticamente, el entorno es el que muestra la Figura 8.2.



Figura 8.2. Entorno incorrecto para el informe de la Figura 6.7.

Al ejecutar el informe, se obtiene el impreso de la Figura 8.3. Todos los exámenes se atribuyeron a un solo alumno. Esto es porque el alias inicialmente seleccionado es *Exámenes*. VFP avanzó por sus registros, sin que RELATION1 tuviera efecto, porque EXÁMENES es la tabla child en tal relación. ALUMNOS quedó en el primer registro, pues el desacuerdo de RELATION1 y alias inicialmente seleccionado le impidió avanzar.

Alumno	Materia	Fecha	Cond.	Nota
Aguirre, Alberto	Administración I	17-02-99	R	8
	Costos	25-07-99	R	7
	Derecho Civil	06-12-99	R	6
	Estadísticas	10-03-99	R	1(
	Contabilidad	25-11-99	L	Ę
	Sistemas de Información I	19-02-99	R	7
	Matemáticas II	03-07-99	R	8

Figura 8.3. Informe incorrecto producido por el entorno de la Figura 6.8.

Para corregir el error, ¿habrá que borrar el entorno y construirlo de nuevo? Felizmente no: basta con dar a la propiedad INITIALSELECTEDALIAS el valor *Alumnos*. El entorno ahora tomará la forma esquemática de la Figura 8.4, que es lo correcto.



Figura 8.4. Entorno correcto para el informe de la Figura 8.1.

Propiedades de Cursor

Alias

Al cargar por primera vez una tabla o vista en un entorno de datos, ALIAS toma su nombre. Por ejemplo, si se carga la tabla ARTÍCULOS, el alias del cursor respectivo toma el valor *Artículos*. Si se vuelve a cargar esa misma tabla, la propiedad ALIAS de este nuevo cursor será *Artículos*1. Si se vuelve a repetir el proceso, ALIAS del nuevo cursor será *Artículos*2. Si examinamos ahora la propiedad CURSORSOURCE de cada CURSOR, todas tendrán el valor *Artículos*.

Lo anterior parece desconcertante; pero no es así. CURSORSOURCE establece el nombre real de la tabla que contiene cada CURSOR: como la tabla real de los tres cursores es la misma, sus respectivos CURSORSOURCE es *Artículos*. Supongamos que en el primer cursor se activa el índice CóDIGO, en el segundo el índice NOMBRE y en el tercero se pone un filtro para los registros cuyo campo PRECIO es mayor a *250*. La tabla es siempre la misma, pero la forma de usarla es diferente. Para distinguir estas formas de uso, a cada CURSOR se le da un nombre distinto en su respectiva propiedad ALIAS.

El valor de ALIAS no puede repetirse en un mismo entorno. No debe ser necesariamente el que propone

VFP, sino que el usuario lo puede cambiar por otro. Por ejemplo, los alias de los cursores mencionados podrían cambiarse por *Código*, *Nombre* y *Filtrado*, si resultasen más significativos. En programación, para referir la tabla, no se podrá usar su nombre real, *Artículos*, pues es igual en las tres formas de usarla. Habrá que referir una de las tres formas de uso, lo que se logra empleando el alias. Este es, precisamente, el propósito de esta propiedad.

CursorSource

CURSORSOURCE (origen del cursor) establece el nombre real que tiene la tabla contenida en un cursor. Por ello, esta propiedad es inmodificable. Si una misma tabla es usada simultáneamente por varios cursores, para distinguir las distintas "versiones" o "formas de uso", se recurre la propiedad ALIAS.

Filter

FILTER (filtro) es una expresión lógica que selecciona algunos registros de una tabla, ignorando los demás. Por ejemplo, si el CURSOR de la tabla ARTÍCULOS tiene en FILTER la expresión *between(precio, 200, 250)*, sólo se tendrá acceso a aquellos registros cuyos campos PRECIO tengan valores entre 200 y 250. Los demás parecerán no existir.

Order

ORDER (orden) establece el nombre del índice que comandará la tabla representada por el CURSOR en tiempo de ejecución. Por ejemplo, si un CURSOR usa la tabla ALUMNOS y se quiere activar uno de sus índices, llamado MATRÍCULA, habrá que establecer en ORDER el valor *Matrícula*. El nombre del índice no se prefija con el nombre de la tabla, como el principiante tiende a hacer: dado que en un CURSOR la propiedad CURSORSOURCE tiene el nombre de la tabla, es erróneo en la propiedad ORDER.

ReadOnly

READONLY (sólo lectura) acepta valores *Verdadero* y *Falso*. Cuando es *Verdadero*, la tabla asociada no permite modificaciones. En tal caso, los objetos de formulario cuyos CONTROLSOURCE sean campos de esa tabla no se pueden modificar en tiempo de ejecución, no importa cómo se definan sus propios READONLY. Los objetos de informe nunca pueden ser modificados en tiempo de ejecución, pues producen salida impresa o en pantalla. Si usted cambia un dato en el papel impreso, sólo afecta el papel, no el archivo de donde proviene ese dato.

Propiedades de Relation

ChildAlias

CHILDALIAS (alias de la hija) establece cuál es la tabla child en una relación.

ChildOrder

CHILDORDER (orden de la hija) establece el índice de la tabla child. En toda relación, para que un registro de la tabla parent encuentre los registros correspondientes en la tabla child, es imprescindible que esta última tenga activo un índice que permita la búsqueda. Este tema es tratado varias veces a lo largo de esta obra, de modo que es innecesario insistir aquí.

OneToMany

ONETOMANY (uno a varios) amplía una relación al tipo uno a varios, cuando es *Verdadero*, o la reduce al tipo uno a uno, cuando es *Falso*. Al crear una relación, el tipo por defecto es uno a uno.

En la Figura 8.4, RELATION1 debe ser de uno a varios: para un registro la tabla ALUMNOS puede haber varios registros en la tabla EXÁMENES. Si RELATION1 no se amplía, parecerá que cada alumno ha rendido un solo examen, lo cual es incorrecto. RELATION2 es de uno a uno: para un registro de la tabla EXÁMENES hay solamente uno en la tabla MATERIAS, porque cada examen se refiere a una única materia. Si RELATION2 se amplía, no tiene efecto.

Esta propiedad tiene mucha incidencia en entornos de informes para lograr resultados correctos. En formularios se usa menos, porque el GRID subordinado, donde se ve el efecto uno a varios, no la tiene en cuenta para lograr su cometido. Sí tiene incidencia cuando en un único GRID se quieren mostrar campos de varias tablas, como el nombre del alumno y los exámenes rendidos. En el entorno hay que relacionar las tablas ALUMNOS y EXÁMENES: si la relación es de uno a uno, por cada alumno en el GRID se verá un solo examen; si es de uno a varios, por cada alumno aparecerán todos los exámenes rendidos por él.

En informes hay que tener en cuenta que ONETOMANY con valor Verdadero sólo tiene efectos para la primera relación del esquema



La tabla que mueve el esquema es A, y siempre está seleccionada. Si la relación que la une con B es uno a varios, A puede alcanzar todos los registros correspondientes de B. Pero si la relación que une a B con C es uno a varios, como B nunca se puede seleccionar, no puede alcanzar todos los registros correspondientes de C, sino sólo el primero, por lo que es lo mismo que esta relación fuera uno a uno.

ParentAlias

PARENTALIAS (alias de la madre) establece cuál es la tabla parent de una relación.

RelationalExpr

RELATIONALEXPR (expresión relacional) define qué expresión debe usar la tabla parent para encontrar los registros correspondientes de la tabla child. La forma de hacerlo es usar un índice de la tabla child. Como éste se ha construido según una expresión en el DISEÑADOR DE TABLAS, la tabla parent debe seguir (o "imitar") esa expresión, pero usando sus propios campos.

Para entender bien lo que queremos decir, supongamos que queremos relacionar la tabla ALUMNOS con la tabla EXÁMENES. El campo que los relaciona, lógicamente, es el que contiene la matrícula. El índice de la tabla hija necesario para la relación, entonces, es el que ordene los exámenes por matrícula. Veamos los siguientes casos.

- Si el campo de la matrícula se llama MATRÍCULA en ambas tablas y la expresión del índice es *Matrícula*, el valor de RELATIONALEXPR será *Matrícula*.
- Si el campo de la matrícula se llama MU en la tabla ALUMNOS y MATRÍCULA en la tabla EXÁMENES, RELATIONALEXPR deberá ser *MUr*, porque esta propiedad define qué usar de la tabla madre para buscar en el índice.
- Sea que el campo de la matrícula se llame MATRÍCULA en ambas tablas, pero la expresión del índice de la hija es *Rojo(Matrícula)*. La función *Rojo()* no existe, pero supongamos que existe y que devuelve el argumento en color rojo. El valor de RELATIONALEXPR no podrá ser *Matrícula*, porque en la tabla madre la matrícula está en negro y nunca encontrará una matrícula en rojo. El valor correcto para que

la búsqueda no fracase, entonces, es Rojo(Matrícula).

• Si el campo de la matrícula se llama MU en la tabla ALUMNOS y MATRÍCULA en la tabla EXÁMENES y la expresión del índice de la hija es *Rojo(Matrícula)*, el valor de RELATIONALEXPR deberá ser *Rojo(MU)*.

Ahora, con los ejemplos anteriores, podrá entender los siguientes ejemplos.

Expresión del índice de la tabla child	Los campos correspondientes en la tabla parent se llaman	RELATIONALEXPR debe ser
Matrícula	Matrícula	Matrícula
Str(Matrícula, 6)	Matrícula	Str(Matrícula, 6)
Matrícula	Matric	Matric
Str(Matrícula, 6)	Matric	Str(Matric, 6)
Dtos(Fecha)	Fecha	Dtos(Fecha)
Dtos(Fecha)	Fec	Dtos(Fec)
Dtos(Fecha) + Código	Fecha y Código	Dtos(Fecha) + Código
Dtos(Fecha) + Código	Fec y Artic	Dtos(Fec) + Artic
Código + Str(Factura, 8)	Código solamente	Código
Código + Str(Factura, 8)	Artic solamente	Artic

En los dos últimos ejemplos se usa coincidencia parcial de claves. Esto es posible cuando se comparan dos cadenas de caracteres de distinto tamaño. Si la opción SET EXACT ACTIVADO de HERRAMIENTAS – OPCIONES – DATOS no tiene tilde, ambas cadenas se consideran iguales si coinciden inicialmente carácter a carácter hasta que se agota la menor. Si SET EXACT ACTIVADO tiene tilde, ambas cadenas se consideran iguales únicamente si tienen igual tamaño y coinciden carácter a carácter. La configuración de los dos estados de la opción que comentamos se maneja también con los comandos *set exact off* y *set exact on*.

Cuarta parte. Generador de expresiones. Funciones. Distribución de objetos

Cuando se diseña un formulario, un informe o un entorno de datos, es necesario definir valores para las propiedades de sus diversos componentes. Tales valores no son más que expresiones, cuyo tipo de datos depende de lo que se está definiendo. Para un programador experimentado, escribir expresiones de cualquier complejidad es una tarea rutinaria. Para un novato, es algo complicado y sujeto a errores.

VFP ofrece un GENERADOR DE EXPRESIONES, cuadro de diálogo que facilita escribir expresiones y comprobar su corrección sintáctica. Para obtener un propósito, las expresiones a menudo deben usar funciones. El generador puede construirlas; pero saber cuál función usar en un contexto dado depende de la capacitación del usuario. El conocimiento de por lo menos las funciones más frecuentes, entonces, se convierte en algo necesario.

Un aspecto complementario, pero útil para el diseño, es el de facilidades que logran con poco esfuerzo una distribución funcional y agradable de los objetos de formularios e informes.

Esta Parte trata estos tres aspectos. El Capítulo 9 se ocupa del GENERADOR DE EXPRESIONES. El Capítulo 10 explica y ejemplifica funciones que aparecen en ese generador, muchas de las cuales se usan con frecuencia. El capítulo 11 trata de la barra de herramientas DISTRIBUCIÓN, la CUADRÍCULA y el menú FORMATO.

Capítulo 9. Generador de expresiones

El cuadro de diálogo GENERADOR DE EXPRESIONES es de uso muy general. Como hay diversos contextos donde se requiere una expresión, en todos ellos se lo puede invocar. Por ejemplo, para definir el contenido de un CAMPO de informe, para crear reglas de validación en el DISEÑADOR DE TABLAS, para hacer pruebas en la ventana COMANDOS, etc.

Las expresiones pueden ser tan simples como el nombre de un campo o tan complejas como una mezcla de campos, variables, funciones, operadores y paréntesis. El GENERADOR DE EXPRESIONES facilita crearlas, mediante listas de componentes adecuados. También permite verificar la corrección sintáctica de las expresiones construidas. Un experto puede escribir directamente cualquier expresión donde sea necesaria. Un novato puede valerse de este generador para ir más seguro.

Generador de expresiones

El GENERADOR DE EXPRESIONES ofrece las opciones que siguen.

Expresión. Es un cuadro de edición donde se puede escribir directamente la expresión o pegar sus componentes.

Funciones. Son cuatro cuadros combinados, cuyas opciones son funciones, operadores y delimitadores. La lista de cada cuadro combinado permite búsqueda incremental. Seleccionada la función u operador deseado, VFP la pega en el cuadro EXPRESIÓN. Los cuadros combinados son los que siguen. Dentro de cada uno, se reconocen las funciones porque comienzan con un nombre seguido por un juego de paréntesis.

Cadena. Lista de operadores y funciones de caracteres, más el delimitador " ".

Lógicas. Lista de operadores y funciones lógicas.

Matemáticas. Lista de operadores y funciones matemáticas.

Fecha. Lista de operadores y funciones de fecha y fecha - hora, más los delimitadores { }.

De tabla. Lista de las tablas y vistas abiertas. Permite seleccionar aquélla de donde se van a tomar los campos necesitados en la expresión.

Campos. Lista de los campos de la tabla o vista activa. Para pegar un campo en el cuadro EXPRESIÓN, se puede dar doble clic sobre él, o seleccionarlo y dar ENTER.

Variables. Lista de las variables creadas por el usuario y las variables del sistema. Las del usuario, si existen al momento de usar el generador, encabezan la lista. Las del sistema se reconocen porque su nombre comienza con un _ (subrayado o guión bajo). Para pegar cualquier variable en el cuadro EXPRESIÓN, se da doble clic en la deseada o se la selecciona y se da ENTER.

Verificar. Botón para validar la sintaxis de la expresión contenida en el cuadro EXPRESIÓN. Si es válida, aparece el mensaje *La expresión es válida*. Si no lo es, aparece un mensaje de error.

Opciones. Botón que activa el cuadro de diálogo OPCIONES DEL GENERADOR DE EXPRESIONES, mediante el cual se puede modificar lo presentado en el GENERADOR DE EXPRESIONES.

Opciones del Generador de expresiones

Este cuadro de diálogo sirve para modificar los elementos que muestra el GENERADOR DE EXPRESIONES. Ofrece las opciones que siguen.

Funciones. Es un grupo de cuatro opciones y un cuadro de lista. Las opciones son CADENA, MATEMÁTICAS, LÓGICAS y FECHA. Según la que se active, el cuadro de lista presenta los delimitadores, operadores y funciones correspondientes. La lista sirve para seleccionar lo que se quiere ver en los cuadros combinados del GENERADOR DE EXPRESIONES. Para seleccionar elementos no consecutivos se mantiene presionada CTRL y se da clic en ellos. Para seleccionar elementos consecutivos, se da clic en el primero, se presiona SHIFT y se da clic en el último. Para cancelar una selección, se oprime CTRL y se da clic.

Todas. Selecciona todos los delimitadores, operadores y funciones del cuadro de lista.

Borrar. Cancela todas las selecciones realizadas en el cuadro de lista.

Alias de campo. Es un grupo de tres opciones que permiten establecer si los campos tendrán como prefijo el alias de la tabla a la que pertenecen. En un entorno de muchas tablas abiertas, una expresión debe referir los campos sin ambigüedades, pues de lo contrario VFP no sabrá a qué tablas pertenecen. Esto se logra escribiendo delante del campo el nombre de la tabla más un punto. Las opciones son las que siguen.

Añadir alias siempre. VFP prefijará siempre cada campo con el nombre de la tabla de donde se toma.

Añadir sólo alias no seleccionados. Aunque en un momento haya varias tablas abiertas, solamente una puede estar seleccionada. Esta opción hace que VFP no prefije los campos de la tabla seleccionada, pero sí los de las otras. Como en tiempo de ejecución la tabla seleccionada en un momento puede ser distinta a la seleccionada cuando se construyó la expresión, puede producirse ambigüedad, por lo que ésta es una opción poco recomendable.

Nunca agregar alias. Los campos no serán prefijados con el alias. Este caso es válido solamente cuando se trabaja con una sola tabla o vista. Esto puede ser en el entorno de datos o en la ventana COMANDOS, cuando se diseña una tabla, etc.

Mostrar variables de memoria del sistema. Es una casilla de verificación que sirve para mostrar o no las variables del sistema en la lista VARIABLES del GENERADOR DE EXPRESIONES. No afecta a las variables definidas por el usuario, que siempre son visibles, si las hay.

Capítulo 10. Algunas funciones útiles

VFP tiene centenares de funciones. Su estudio completo supera ampliamente el propósito de este trabajo. Sin embargo, es útil conocer varias de las que aparecen en el cuadro de diálogo GENERADOR DE EXPRESIONES.

Toda función tiene un nombre seguido por un juego de paréntesis. Según lo que realice, la función puede requerir datos complementarios, llamados *argumentos*. Por ejemplo, la función *date()*, que devuelve la fecha del sistema operativo, no necesita argumentos. La función *sqrt()*, que devuelve la raíz cuadrada de un valor, requiere como argumento el radicando. La función *mod()*, que devuelve el resto de una división, necesita como argumentos el dividendo y el divisor. Si la función requiere uno o varios argumentos, éstos se escriben entre los paréntesis, separados por comas si son dos o más.

A los efectos de la exposición que sigue, en que se define de modo general el formato de las funciones, usaremos la notación que pasamos a exponer.

<u>La notación</u>	Representa
expresión	Una expresión de cualquier tipo.
expresiónC	Una expresión de caracteres, como un campo o variable de tipo carácter, o un valor li- teral encerrado entre apóstrofes o comillas, como 'Jujuy 145' o "Manuel Belgrano".
expresiónN	Una expresión numérica, como un campo o variable de tipo numérico; un valor literal numérico, como 156, -1598 ó 27 * 416.
expresiónF	Una expresión de fecha, como un campo o variable de tipo fecha, o una fecha escrita literalmente, entre llaves, como {25-12-98}.
expresiónH	Una expresión de fecha – hora, que se usa para fecha y hora o solamente para hora. Puede ser un campo o variable de tipo fecha – hora, o una fecha – hora escrita literal- mente, entre llaves, como {11:44:39AM} o {25-12-98 11:44:39AM}. Si contiene fecha y hora, ambas se separan por un espacio.
	Un espacio en blanco que queremos resaltar. Los espacios en blanco no se ven, por- que son espacios. Como en algunos ejemplos es necesario saber dónde están, hemos usado este "punto alto" para representarlos.
[]	Que lo encerrado en los corchetes es de uso opcional.
I	Dos o más partes mutuamente excluyentes: sólo se puede usar una de ellas.
	Que lo que les antecede puede repetirse las veces que se desee.

Cómo probar funciones

Para probar lo que realizan las funciones, lo más efectivo es usar la ventana COMANDOS y el escritorio de VFP. En la ventana COMANDOS usaremos los comandos ?, =, *Display Memory* y *Clear Memory*.

El comando ? evalúa una expresión y muestra el resultado en el escritorio. Dado que las funciones no trabajan por sí solas, sino que necesitan estar en algún comando, podemos usar como expresión del comando ? la función a probar. Por ejemplo, para probar la función *date()*, en la ventana COMANDOS podemos escribir **? date()** y dar ENTER. Esto provocará que VFP evalúe esta función y escriba el valor que ella le devuelve en el escritorio, es decir, la fecha actual del sistema operativo.

El comando = manda crear una variable con el contenido de una expresión. Así, el comando **hoy = date()** crea la variable *hoy* con la fecha actual del sistema operativo. El tipo de la expresión determina el tipo de la variable. En el ejemplo, como la función *date()* devuelve un valor de tipo fecha, que se almacena en *hoy*, la variable *hoy* será de tipo fecha. A veces, probar algunas funciones mediante el resultado en el escritorio es impreciso. En lugar de ello, como haremos, conviene crear una variable que almacene tal resultado y después examinar el contenido de tal variable.

El comando *display memory* muestra una, algunas o todas las variables de memoria en el escritorio. Si en la ventana COMANDOS se da el comando **display memory**, el escritorio mostrará todas las variables del usuario existentes en ese momento más todas las variables del sistema. Si se da el comando **display memory like** *, el escritorio sólo mostrará todas las variables del usuario. El comando display **memory like** ??m?a* mostrará todas las variables del usuario que tengan una *m* en la tercera posición y una *a* en la quinta. El comando **display memory like importe** mostrará únicamente la variable *importe*.

El comando *clear memory* borra todas las variables del usuario. Si en las pruebas la cantidad de variables listadas por *display memory* es grande, de modo que encontrar la necesitada sea incómodo, se las puede eliminar con este comando.

En la exposición que sigue, a modo de ejemplo se prueban algunas funciones en la ventana COMANDOS. Como guía visual, estos ejemplos se reconocen por un borde en el margen izquierdo.

Abs(expresiónN)

Devuelve el valor absoluto de expresiónN. Por ejemplo

```
abs( 7 * -4 ) devuelve 28.
abs( saldo ), si saldo es 150 ó -150, devuelve 150.
```

Pruebe en la ventana COMANDOS lo siguiente:

```
? abs( 7 * -4 )
? abs( saldo )
a = abs( 7 * -4 )
display memory like a
```

Alltrim(expresiónC)

Devuelve el valor de *expresiónC* sin los espacios iniciales y finales que pueda tener. Por ejemplo, si *domicilio* es un campo cuyo contenido es '····San Martín y Mendoza····', seis espacios a la izquierda y seis a la derecha,

alltrim(domicilio) devuelve 'San Martín y Mendoza'.

Como los espacios no se pueden ver, use la ventana COMANDOS para crear dos variables y examinarlas:

```
nombre1 = '···· San Martín y Mendoza ···· '
nombre2 = alltrim( nombre1 )
```

display memory like nombre*

Avg(expresiónN)

Devuelve la media aritmética de los distintos valores de *expresiónN* que hay en un conjunto de registros. Se usa en vistas y consultas que emplean agrupamiento. Por ejemplo, para obtener el promedio de compras de cada cliente, se agrupa por cliente y se usa la función

```
avg(linfac.can * linfac.pre)
```

Between(expresión, límite inferior, límite superior)

Si *expresión* está contenida entre *límite inferior* y *límite superior*, devuelve .T. En caso contrario, devuelve .F. *Expresión* puede ser numérica, de caracteres, de fecha o de fecha – hora. Los límites deben ser de igual tipo a *expresión*. Por ejemplo

```
between( 15, 8, 45 ) devuelve .T.
between( { 10-10-1997 }, { 01-01-1999 }, { 31-12-1999 } ) devuelve .F.
```

```
? between( 15, 8, 45 )
? between( { 10-10-1997 }, { 01-01-1999 }, { 31-12-1999 } )
a1 = between( 15, 8, 45 )
a2 = between( { 10-10-1997 }, { 01-01-1999 }, { 31-12-1999 } )
display memory like a*
```

Cdow(expresiónF | expresiónH)

Devuelve el nombre del día de semana correspondiente a expresiónF o expresiónH. Por ejemplo

cdow({ 04-08-1999 }) devuelve 'Miércoles'.

? cdow({ 04-08-1999 }) dia = cdow({ 10-10-2000 }) disp memo like dia

En el último comando anterior, observe que se ha escrito *disp* en lugar de *display* y *memo* en lugar de *memory*. VFP reconoce abreviaturas, si tienen por lo menos cuatro letras iniciales.

Count(expresión)

Devuelve la cantidad de registros que hay en un grupo. Se usa en vistas y consultas que emplean agrupamiento. *Expresión* puede ser de cualquier tipo.

Cmonth(expresiónF | expresiónH)

Devuelve el nombre del mes correspondiente a expresiónF o expresiónH. Por ejemplo

```
cmonth({04-08-1999}) devuelve 'Agosto'
```

```
? cmonth( { 04-08-1999 } )
mes = cmonth( { 04-08-1999 } )
disp memo like mes
```

Date()

Devuelve la fecha del sistema operativo.

```
? date()
a = date()
disp memo like a
b = date() + 1
disp memo like b
? cmonth( date())
? cmonth( date() - 2)
```

Si se suma una cantidad a una expresión de fecha, se la interpreta como días.

Datetime()

Devuelve la fecha y hora del sistema operativo.

```
? datetime()
a = datetime()
? a
? a +60
? a + 3600
```

Si se suna una cantidad a una expresión de fecha – hora, se la interpreta como segundos. En los dos últimos comandos, 60 representan un minuto y 3600 una hora. La variable a se ha usado para dejar fija la fecha y hora del sistema operativo al momento de consultarla en el tercer comando.

Day(expresiónF | expresiónH)

Devuelve el día de expresiónF o expresiónH. El día devuelto es una expresión numérica. Por ejemplo

```
day( { 15-11-1999 } ) devuelve 15.
```

? day({ 20-11-2004}) ? day(date()) + 4

Deleted([alias])

Devuelve .T. si el registro actual de la tabla indicada en *alias* está marcado para ser destruido. En caso contrario, devuelve .F. Si no se indica *alias*, supone la tabla actualmente seleccionada. Por ejemplo, si se usan las tablas CLI y ENCFAC, la seleccionada es ENCFAC y se está en el registro 8 de CLI, marcado, y en el registro 129 de ENCFAC, no marcado

```
deleted( 'cli' ) devuelve .T. deleted( ) devuelve .F.
```

close tables all	Cierra todas las tablas
use cli in 1	Abre la tabla C⊔ en el área 1
go 8	Hace actual el registro 8 de CLI
delete	Marca el registro actual de CLI, que es el 8
use encfac in 2	Abre la tabla ENCFAC en el área 2
select encfac	Selecciona la tabla ENCFAC
go 129	Hace actual el registro 129 de ENCFAC

?deleted('cli')	Pregunta si el registro actual de CLI, tabla no seleccionada, está marcado
? deleted()	Pregunta si el registro actual de ENCFAC, tabla seleccionada, está marcado
select cli	Selecciona CLI
recall all	Desmarca todos los registros marcados
close tables all	Cierra todas las tablas

Dmy(expresiónF | expresiónH)

Devuelve *expresiónF* o *expresiónH* transformada a expresión de caracteres. Por ejemplo, si la variable *Hoy* vale { 06-08-1999 }

dmy(hoy) devuelve '06 Agosto 1999'

```
hoy = { 06-08-1999 }
? dmy( hoy)
texto = dmy( hoy)
disp memo like texto
```

Dtoc(expresiónF | expresiónH)

Devuelve *expresiónF* o *expresiónH* convertida a expresión de caracteres, usando el formato vigente de fecha y de hora. Este formato no es conveniente para construir índices. Por ejemplo, si el campo *Fecha* tiene el valor {10-09-2005}

```
dtoc( fecha) devuelve '10-09-2005'
'Fecha de pago: ' + dtoc( fecha) devuelve 'Fecha de pago: 10-09-2005'
```

```
f = date()
set date italian
f1 = dtoc(f)
set date american
f2 = dtoc(f)
set date japan
f3 = dtoc(f)
set date dmy
f4 = dtoc(f)
disp memo like f*
```

Los comandos set date anteriores funcionan como los formatos de fecha de HERRAMIENTAS – OPCIONES – REGIONAL.

Dtos(expresiónF | expresiónH)

Devuelve *expresiónF* o *expresiónH* convertida a expresión de caracteres. El formato que usa para el resultado es AAAAMMDD, adecuado para construir índices por fechas.

```
f = date()
set date italian
f1 = dtos(f)
set date american
f2 = dtos(f)
set date japan
```

```
f3 = dtos( f )
set date dmy
f4 = dtos( f )
disp memo like f*
```

Empty(expresión)

Devuelve .T. si *expresión* está vacía. En caso contrario, devuelve .F. La interpretación de vacío depende del tipo de *expresión*, según sigue:

TipoSe interpreta vacío cuando contieneCarácterEspacios, tabulaciones, Enter o una combinación de ellos.Numérico0FechaCarencia de contenidoFecha horaCarencia de contenidoLógico.F.MemoCarencia de contenido

clear memory texto = " " número = 2 * 5 – 10 fecha = ctod(" ") hora = ctot (" ") falso = 3 > 8 disp memo like * clear memory

Hour(expresiónH)

Devuelve la porción de hora correspondiente a *expresiónH*. La hora devuelta es una expresión numérica. Por ejemplo,

```
hour( { 16/02/95 1:00pm } ) devuelve 13.
```

```
? hour( date( ) )
x = ( { 16/02/95 1:00pm } )
h = hour( x )
disp memo like *
```

lif(condición, expresión por verdadero, expresión por falso)

Esta función prueba una *condición*. Si la prueba resulta .T., devuelve *expresión por verdadero*. Si resulta .F., devuelve *expresión por falso*. Ejemplos

iif(ingreso > 1000, 'Insuficiente', 'Adecuado') si ingreso vale 870, devuelve 'Insuficiente'. iif(12 * 5 < 50, 20, 'Error') devuelve 'Error', porque 12 * 5 no es menor a 50. iif(fecha<> date(), 18, 25) si fecha es distinta a hoy, devuelve 18; si es igual, devuelve 25. ingreso = 870 ? iif(ingreso > 1000, 'Ingreso insuficiente', ' Ingreso adecuado') ingreso = ingreso + 1000 ? iif(ingreso > 1000, 'Ingreso insuficiente', ' Ingreso adecuado') ? iif(12 * 5 < 50, 20, ' Error')</pre> ? iif(12 * 5 > 50, 20, ' Error')

Inlist(expresión, valor1, valor2, valor3, valor4 ...)

Si *expresión* coincide con alguno de los valores listados, devuelve .T. En caso contrario, devuelve .F. Todos los argumentos deben ser del mismo tipo. El tipo puede ser cualquiera. La lista puede tener hasta 24 argumentos. Por ejemplo

inlist(9, 4 * 2, 5 + 4, 12, 17, 28) devuelve .T., porque 9 es igual a 5 + 4.

```
? inlist( 9, 4 * 2, 5 + 4, 12, 17, 28 )
clear memory
código = 'B' + '03'
a = 'A01'
b = 'A02'
c = 'B03'
d = 'F30'
resultado = inlist( código, a, b, c, d )
disp memo like *
```

Left(expresiónC, tamaño)

Devuelve una porción de *expresiónC*, comenzando por la izquierda, con el *tamaño* que se indica. *Tamaño* es una expresión numérica. Por ejemplo,

```
left( 'Cocodrilo', 4 ) devuelve 'Coco'.
```

? left(' Pantalla color', 5)

Lower(expresiónC)

Devuelve las letras contenidas en expresiónC convertidas a minúsculas. Por ejemplo

lower('Casa Grande') devuelve 'casa grande'.

```
? lower( ' CUADERNO' )
```

Ltrim(expresiónC)

Devuelve *expresiónC* sin los espacios iniciales. Por ejemplo, si *domicilio* es un campo cuyo contenido es '....San Martín y Mendoza.....

Itrim(domicilio) devuelve 'San Martín y Mendoza · · · · · '.

```
ma = '·····Sol radiante····'
me = Itrim( ma )
disp memo like m*
```

Max(expresión)

Devuelve el valor máximo que toma *expresión* en un grupo de registros. Se usa en vistas y consultas que emplean agrupamiento. *Expresión* puede ser de cualquier tipo. Por ejemplo, para calcular la compra máxima de un cliente, se agrupará por el campo *cliente* y se empleará la expresión

```
max( linfac.can * linfac.pre )
```

Max(expresión1, expresión2 ...)

No confunda esta forma de *Max* con la anterior. Devuelve la *expresión* de la lista que tiene el valor mayor. Todas las expresiones deben ser del mismo tipo. El tipo puede ser cualquiera. Por ejemplo

max(5, 3 * 7, 8 – 1, 18 + 1) devuelve 21, resultado de 3 * 7, el mayor valor de la lista.

? max(5, 3 * 7, 8 – 1, 18 + 1) ? max(date(), date() – 8, date(9) + 10) ? max('poco', 'mucho', 'algo', 'nada')

Min(expresión)

Devuelve el valor mínimo que toma *expresión* en un grupo de registros. Se usa en vistas y consultas que emplean agrupamiento. *Expresión* puede ser de cualquier tipo. Por ejemplo, para calcular la compra mínima de un cliente, se agrupará por el campo *cliente* y se empleará la expresión

```
min( linfac.can * linfac.pre )
```

Min(expresión1, expresión2 ...)

No confunda esta forma de *Min* con la anterior. Devuelve la *expresión* de la lista que tiene el valor menor. Todas las expresiones deben ser del mismo tipo. El tipo puede ser cualquiera. Por ejemplo

min(5, 3 * 7, 8 – 1, 18 + 1) devuelve 5, el menor valor de la lista.

? min(5, 3 * 7, 8 – 1, 18 + 1) ? min(date(), date() – 8, date(9) + 10) ? min('poco', 'mucho', 'algo', 'nada')

Minute(expresiónH)

Devuelve la porción de minutos correspondiente a *expresiónH*. Los minutos devueltos son una expresión numérica. Por ejemplo

minute({ 02/16/95 1:15pm }) devuelve 15.

```
? minute( date( ) )
? minute( date( ) ) + 8
```

Month(expresiónF | expresiónH)

Devuelve el mes de expresiónF o expresiónH. El mes devuelto es una expresión numérica. Por ejemplo

```
month( {15-11-1999 } ) devuelve 11.
```

```
? month( date( ) )
? month( date( ) ) * 5
```

Padc(expresión, tamaño [, carácter de relleno])

Transforma una expresión de cualquier tipo, excepto lógico o general, agregándole un carácter de relleno

a ambos lados, repitiéndolo si hace falta. Si no se indica *carácter de relleno*, se supone un espacio en blanco. El valor devuelto es una expresión de caracteres con el *tamaño* que se indica. *Tamaño* es una expresión numérica. Por ejemplo:

padc('Importante', 14, 'x') produce 'xxImportantexx'. padc(987, 7, ':') devuelve '::987::'.

? padc(123, 5, ':') ? padc(123, 6, ':') ? padc(123, 7, ':') ? padc(123, 8, ':') ? padc(123, 9, ':')

Padl(expresión, tamaño [, carácter de relleno])

Transforma una *expresión* de cualquier tipo, excepto lógico o general, agregándole un *carácter de relleno* a la izquierda, repitiéndolo si hace falta. El valor devuelto es una expresión de caracteres con el *tamaño* que se indica. *Tamaño* es una expresión numérica. Por ejemplo:

padl('Todo', 8, '!') devuelve '!!!!Todo'.

```
? padl( 123, 5, '-' )
? padl( 123, 6, '-' )
? padl( 123, 7, '-' )
? padl( 123, 8, '-' )
? padl( 123, 9, '-' )
```

Padr(expresión, tamaño [, carácter de relleno])

Transforma una *expresión* de cualquier tipo, excepto lógico o general, agregándole un *carácter de relleno* a la derecha, repitiéndolo si hace falta. El valor devuelto es una expresión de caracteres con el *tamaño* que se indica. *Tamaño* es una expresión numérica. Por ejemplo:

padr({ 10-05-96 }, 12, '=') devuelve '10-05-96===='.

? padr(123, 5, 'x') ? padr(123, 6, 'x') ? padr(123, 7, 'x') ? padr(123, 8, 'x') ? padr(123, 9, 'x')

Proper(expresiónC)

Convierte la primera letra de cada palabra de *expresiónC* a mayúscula y las restantes a minúsculas. Por ejemplo

proper('mar AZUL') devuelve 'Mar Azul'.

? proper('cristóbal colón') ?proper('CRISTÓBAL COLÓN')

Recno([alias])

Devuelve el número de registro actual de la tabla indicada por alias. Si no se indica alias, se supone la ta-

bla actualmente seleccionada. Por ejemplo, si se usan las tablas CLI y ENCFAC, la seleccionada es ENCFAC y se está en el registro 8 de CLI y en el registro 129 de ENCFAC

```
recno( 'cli' ) devuelve 28.
recno( ) devuelve 129.
```

close tables all	Cierra todas las tablas
use cli in 1	Abre la tabla C⊔ en el área 1
go 8	Hace actual el registro 8 de C⊔
use encfac in 2	Abre la tabla ENCFAC en el área 2
select encfac	Selecciona la tabla ENCFAC
go 129	Hace actual el registro 129 de ENCFAC
? recno('cli')	Da el número de registro actual de CLI, tabla no seleccionada
? recno ()	Da el número de registro actual de ENCFAC, tabla seleccionada
close tables all	Cierra todas las tablas

Replicate(expresiónC, veces)

Repite expresiónC las veces indicadas. Por ejemplo

```
replicate( 'feo ', 3 ) produce 'feo feo feo '
```

```
?replicate( '¡Bonitos zapatos! ', 100 )
```

Right(expresiónC, tamaño)

Devuelve una porción de *expresiónC*, comenzando por la derecha, con el *tamaño* que se indica. *Tamaño* es una expresión numérica. Por ejemplo,

right('ZAPATO', 4) devuelve 'PATO'.

?right('metodológico', 6)

Rtrim(expresiónC)

Quita los espacios finales de *expresiónC*. Por ejemplo, si *domicilio* es un campo cuyo contenido es '....San Martín y Mendoza....'

rtrim(domicilio) produce '···· San Martín y Mendoza'.

```
ja = '·····
je = rtrim( ma )
disp memo like j*
```

Seek(expresión [, alias [, índice]])

Busca el primer registro cuya clave coincida total o parcialmente con *expresión*. Busca en la tabla indicada en *alias*, usando el *índice* indicado, que no necesita estar activo. Si encuentra un registro con tal clave, devuelve .T. De lo contrario, devuelve .F. Si se omite *alias*, busca en la tabla actualmente seleccionada. Si se omite *índice*, usa el índice actualmente activo. La coincidencia entre la clave y *expresión* debe ser total si SET EXACT está activada; si no lo está, la coincidencia puede ser parcial, comenzando por la izquierda. La exactitud se establece con el comando *set exact* o en HERRAMIENTAS – OPCIONES – DATOS – COMPARACIÓN DE CADENAS.

set exact off close tables all use cli in 0 use encfac in 0 ? seek(7, 'cli', 'cli') ? seek('Castro', 'cli', 'nom') set order to cli	Deshabilita la exactitud de las comparaciones Cierra todas las tablas y selecciona el área 1 Abre la tabla CLI en la mayor área disponible no ocupada, la 1 Abre la tabla ENCFAC en la mayor área disponible no ocupada, la 2 Busca un cliente 7 en la tabla CLI, valiéndose del índice CLI Busca un cliente <i>Castro</i> en la tabla CLI, valiéndose del índice NOM Activa el índice CLI de la tabla CLI
? seek(23, 'cli', 'cli')	Busca un cliente 23 en la tabla CLI, valiéndose del índice CLI
? seek('Castro')	Busca un cliente <i>Castro</i> en la tabla seleccionada, <i>Cli</i> , valiéndose del índice NOM, que es el activo
? seek(18, 'encfac', 'cli')	Busca la primera factura con cliente 18 en la tabla ENCFAC, valiéndose del índice CLI
? seek(15, 'encfac', 'fac') close tables all	Busca la factura <i>15</i> en la tabla ENCFAC, valiéndose del índice FAC Cierra todas las tablas

Estas formas de buscar coincidencias exactas o parciales en un índice no sólo valen en Seek(), sino que tienen validez general. Así, en una relación, la tabla parent necesita un índice de la tabla child para poder hallar en ella los registros correspondientes a cada registro de la primera. La propiedad RELATIONALEXPR, conforme a lo que se está diciendo, puede coincidir total o parcialmente con la expresión usada para construir el índice declarado en la propiedad CHILDORDER. Sea querer relacionar las tablas CLI y ENCFAC. El cuadro siguiente da información necesaria para hacerlo, donde el código de cliente en la tabla CLI se llama CÓDIGO.

Tabla	Campo	Tipo y Ancho	Descripción
CLI	Código	Numérico, 5	Código de cliente
ENCFAC	Cli	Numérico, 5	Código de cliente
	FAC	Numérico, 8	Número de factura
	Indice necesario		
	CLIFAC	Expresión: str(Cli, 5) + str(Fac, 8)

Las propiedades de la relación tendrán los siguientes valores:

PARENTALIAS:	Cli
CHILDALIAS:	Encfac
CHILDORDER:	Clifac
RELATIONALEXPR:	str(Código, 5)

RELATIONALEXPR usa campos de la tabla parent, imitando la expresión del índice. Habrá que usar el campo CÓDIGO de la tabla CLI,que corresponde al campo CLI de la tabla ENCFAC. Pero como CLI no tiene un campo correspondiente al campo FAC de ENCFAC, el valor de RELATIONALEXPR debe limitarse a una coincidencia parcial. El resultado de esta relación será que, para un determinado registro de CLI, se obtendrán los registros relacionados en ENCFAC, con la ventaja adicional de estar ordenados por factura.

Sec(expresiónH)

Devuelve los segundos contenidos en *expresiónH*. El resultado devuelto es una expresión numérica. Por ejemplo, si *ahora* contiene el valor { 02/16/95 1:15:42pm },

sec(ahora) devuelve 42.

? sec(date())

Space(cantidad)

Devuelve tantos espacios como indica cantidad. Por ejemplo

space(10) devuelve '······.

```
a = 'Listado de precios' + space(35) + 'Ordenado por código de artículo' disp memo like a
```

Sqrt(expresiónN)

Devuelve la raíz cuadrada de expresiónN. Por ejemplo

sqrt(alfa), si alfa vale 81, devuelve 9.

```
? sqrt( 3472 )
```

Str(expresiónN [, tamaño [, decimales]])

Convierte *expresiónN* a expresión de caracteres. *Tamaño* es la cantidad de posiciones que tendrá la expresión de caracteres. Si no se escribe *tamaño*, VFP usa diez posiciones enteras, convirtiendo a notación científica el resultado si el número supera esa magnitud. Si *expresiónN* tiene decimales y se indica *tamaño* pero no *decimales*, la expresión de caracteres no los incluirá. Para que los incluya, hay que escribir el argumento *decimales*, teniendo en cuenta contar en *tamaño* los decimales y el punto decimal. Por ejemplo, si el campo *importe* tiene el valor 1500.25

```
str( importe, 8 ) devuelve '···· 1500'
str( importe, 7, 2) devuelve '1500.25'
str( importe, 4) devuelve '****', resultado que indica error por insuficiencia de Tamaño.
importe = 1500.25
imp1 = str( importe, 8 )
imp2 = str( importe, 7, 2 )
imp3 = str( importe, 4 )
disp memo like imp*
```

Substr(expresiónC, inicio [, tamaño])

Devuelve una porción de *expresiónC*. La porción comienza en *inicio* y su largo viene dado por *tamaño*. *Inicio* y *tamaño* son expresiones numéricas. Por ejemplo, si el campo *patente* tiene el valor 'ESA163'

```
substr( patente, 4, 2 ) devuelve '16'.
```

Si no se indica tamaño, la porción va desde inicio hasta el final de expresiónC. Por ejemplo

substr(patente, 4) devuelve '163'.

```
a = 'Avenida San Martín'
? substr( a, 8, 3 )
? substr( a, 5 )
```

Sum(expresiónN)

Devuelve la suma de los distintos valores de *expresiónN* que hay en un conjunto de registros. Se usa en vistas y consultas que emplean agrupamiento. Por ejemplo, para obtener la suma de compra de cada cliente, se agrupa por el campo *cliente* y se usa la función

```
sum( linfac.can * linfac.pre )
```

Time()

Devuelve la hora actual del sistema, en 24 horas, en una ExpresiónC de formato HH : MM : SS.

? time(date())

Transform(expresión, formato)

Devuelve *expresión* transformada según los códigos FORMAT e INPUTMASK usados en *formato*. El resultado es una expresión de caracteres. *Expresión* puede ser numérica, de caracteres, de fecha o de fecha – hora. *Formato* es una expresión de caracteres. Dentro de ella se pueden escribir códigos FORMAT, códigos INPUTMASK o ambos. Antes de los códigos FORMAT se usa @ (arroba). Si se usan códigos FORMAT y códigos INPUTMASK, se deja un espacio separador entre ellos.

transform(13257, '999,999,99 ') devuelve ' 13.257,00' transform(texto, '@!'), si *texto* contiene 'Resultado final', devuelve 'RESULTADO FINAL' transform(importe, '@\$ 999,999'), si *importe* contiene 4537, devuelve '\$4.537'.

- ? transform(78645.75, '999,999,999.99')
- ? transform(78645.75, '\$99,999,999.99')
- ? transform(78645.75, '\$\$9,999,999.99')
- ? transform(78645.75, '@c 999,999,999.99')
- ? transform(-78645.75, '@c 999,999,999.99')
- ? transform(78645.75, '@x 999,999,999.99')
- ? transform(-78645.75, '@c 999,999,999.99')
- ? transform('casa', '!aaa')
- ? transform('casa', !a!a')
- ? transform('casa', '@!')
- ? transform('3225467', '@r 99-999/9')

En el último ejemplo, la "r" en la porción FORMAT indica que en INPUTMASK se usarán intercalaciones que no son códigos, sino símbolos personalizados, como son "–" y "/".

Upper(expresiónC)

Convierte las letras contenidas en expresiónC a mayúsculas. Por ejemplo

```
upper( 'pantalla' ) devuelve 'PANTALLA'.
upper( 'Auto' ) devuelve 'AUTO'.
```

```
? upper( 'pantalla' )
? upper( 'Auto' )
```

```
? upper( '25 de mayo' )
```

Year(expresiónF | expresiónH)

Devuelve el año de *expresiónF* o *expresiónH*. El año es una expresión numérica. Por ejemplo year({ 15-11-1999 }) devuelve 1999. ? year(date())

– 90 –

Capítulo 11. Distribución de objetos

Al diseñar formularios o informes, ubicar los objetos con funcionalidad y gusto lleva buen tiempo. VFP posee utilidades que ayudan en la tarea. Son la barra de herramientas DISTRIBUCIÓN, la cuadrícula de diseño y algunas opciones del menú FORMATO.

Barra de herramientas Distribución

Esta barra de herramientas contiene los botones que exponemos a continuación. Algunos de ellos funcionan con un objeto seleccionado y otras con por lo menos dos. Los botones que se activan en esta barra indican las posibilidades de distribución a que pueden ser sometidos.

Alinear los bordes izquierdos E. Funciona con por lo menos dos objetos seleccionados. Tiene dos posibilidades, según se mantenga pulsada o no la tecla CTRL antes de dar clic en este botón.

Si no se usa CTRL, alinea los bordes izquierdos de los objetos con respecto al borde izquierdo del objeto que está más a la izquierda, como indica la Figura 11.1. Las líneas punteadas indican lo que se toma como referencia.



Figura 11.1. Antes y después de alinear los bordes izquierdos sin Ctrl

Si se usa CTRL, alinea los bordes izquierdos de los objetos seleccionados con respecto al borde izquierdo del objeto que está más a la derecha, como indica la Figura 11.2.

|--|

Figura 11.2. Antes y después de alinear los bordes izquierdos con Ctrl

Alinear los bordes derechos . Funciona con por lo menos dos objetos seleccionados. Tiene dos posibilidades, según se mantenga pulsada Ctrl antes de dar clic en este botón.

Si no se usa CTRL, alinea los bordes derechos de los objetos con respecto al borde derecho del objeto que está más a la derecha, como indica la Figura 11.3.



Figura 11.3. Antes y después de alinear los bordes derechos sin Ctrl

Si se usa CTRL, alinea los bordes derechos de los objetos seleccionados con respecto al borde derecho del objeto que está más a la izquierda, como indica la Figura 11.4.



Figura 11.4. Antes y después de alinear los bordes derechos con Ctrl

Alinear los bordes superiores . Funciona con por lo menos dos objetos seleccionados. Tiene dos posibilidades, según se mantenga pulsada Ctrl antes de dar clic en este botón.

Si no se usa CTRL, alinea los bordes superiores de los objetos con respecto al borde superior del objeto que está más arriba, como indica la Figura 11.5.



Figura 11.5. Antes y después de alinear los bordes superiores sin Ctrl

Si se usa CTRL, alinea los bordes superiores de los objetos seleccionados con respecto al borde superior del objeto que está más abajo, como indica la Figura 11.6.



Figura 11.6. Antes y después de alinear los bordes superiores con Ctrl

Alinear los bordes inferiores . Funciona con por lo menos dos objetos seleccionados. Tiene dos posibilidades, según se mantenga pulsada Ctrl antes de dar clic en este botón.

Si no se usa CTRL, alinea los bordes inferiores de los objetos con respecto al borde inferior del objeto que está más abajo, como indica la Figura 11.7.



Figura 11.7. Antes y después de alinear los bordes inferiores sin Ctrl

Si se usa CTRL, alinea los bordes inferiores de los objetos seleccionados con respecto al borde inferior del objeto que está más arriba, como indica la Figura 11.8.



Figura 11.8. Antes y después de alinear los bordes inferiores con Ctrl

Alinear centros verticalmente Funciona con por lo menos dos objetos seleccionados. Considera la "franja" vertical determinada por los bordes verticales extremos de la selección. Calcula el medio horizontal de esa franja, por donde pasa un eje vertical. Traslada los objetos seleccionados, de modo que el centro horizontal de cada uno de ellos coincida con ese eje. La Figura 11.9, donde las líneas de punto indican los extremos de la franja y el eje vertical, ilustra lo explicado.



Figura 11.9. Antes y después de alinear verticalmente los centros

Alinear centros horizontalmente Funciona con por lo menos dos objetos seleccionados. Considera la "franja" horizontal determinada por los bordes horizontales extremos de la selección. Calcula el medio vertical de esa franja, por donde pasa un eje horizontal. Traslada los objetos seleccionados, de modo que el centro vertical de cada uno de ellos coincida con ese eje. La Figura 11.10, donde las líneas de punto indican los extremos de la franja y el eje horizontal, ilustra lo explicado.



Figura 11.10. Antes y después de linear horizontalmente los centros

Centrar horizontalmente E. Funciona con uno o más objetos. La selección se considera un grupo. Se calcula el medio horizontal de ese grupo, por donde pasa un eje vertical. Se traslada el grupo, haciendo coincidir su eje vertical con el eje vertical del contenedor (formulario o página) o de la banda (informe o etiqueta). La Figura 11.11, donde las líneas de punto indican el grupo, el eje del grupo y el eje del contenedor, ilustra lo explicado.



Figura 11.11. Antes y después de centrar horizontalmente

Centrar verticalmente E. Funciona con uno o más objetos. La selección se considera un grupo. Se calcula el medio vertical de ese grupo, por donde pasa un eje horizontal. Se traslada el grupo, haciendo coincidir su eje horizontal con el eje horizontal del contenedor (formulario o página) o de la banda (informe

o etiqueta). La Figura 11.12, donde las líneas de punto indican el grupo, el eje del grupo y el eje del contenedor, ilustra lo explicado.



Figura 11.12. Antes y después de centrar verticalmente

En las bandas de informes este botón no siempre funciona bien para varios objetos. Para lograrlo, primero hay que alinear los centros horizontalmente o agrupar los objetos con FORMATO – AGRUPAR.

Mismo ancho E. Funciona con por lo menos dos objetos seleccionados. Tiene dos posibilidades, según se mantenga pulsada Ctrl antes de dar clic en este botón.

Si no se usa CTRL, todos los objetos toman el ancho del objeto más ancho, como ilustra la Figura 11.13.



Figura 11.13. Antes y después de igualar anchos, sin Ctrl

Si se usa Ctrl, todos los objetos toman el ancho del objeto menos ancho, como ilustra la Figura 11.14.

Figura 11.14. Antes y después de igualar anchos, con Ctrl

Mismo alto II. Funciona con por lo menos dos objetos seleccionados. Tiene dos posibilidades, según se mantenga pulsada Ctrl antes de dar clic en este botón.

Si no se usa CTRL, todos los objetos toman el alto del objeto más alto, como ilustra la Figura 11.15.



Figura 11.15. Antes y después de igualar altos, sin Ctrl

Si se usa Ctrl, todos los objetos toman el alto del objeto menos alto, como ilustra la Figura 11.16.



Figura 11.16. Antes y después de igualar altos, con Ctrl

Mismo tamaño D. Funciona con por lo menos dos objetos seleccionados. Tiene dos posibilidades, según se mantenga pulsada Ctrl antes de dar clic en este botón.

Si no se usa CTRL, todos los objetos toman el alto del objeto más alto y el ancho del objeto más ancho, como ilustra la Figura 11.17.



Figura 11.17. Antes y después de igualar tamaños, sin Ctrl

Si se usa Ctrl, todos los objetos toman el alto del objeto menos alto y el ancho del objeto menos ancho, como ilustra la Figura 11.18.



Figura 11.18. Antes y después de igualar tamaños, con Ctrl

Traer al primer plano L. Funciona con uno o más objetos. Si hay objetos encimados que tapen parcial o totalmente a otros, los tapados son traídos al frente, como muestra la Figura 11.19.



Figura 11.19. Antes y después de traer a primer plano dos objetos

Cuando se apilan objetos, se lleva un número del plano donde está cada uno. En la parte "antes" del ejemplo anterior, hay tres planos: uno al frente, ocupado por un rectángulo; uno intermedio, ocupado por dos rectángulos; y uno al fondo, ocupado por el rectángulo que recuadra a los otros.

El efecto de ver un objeto tapado por otro se produce si son opacos. Si son transparentes, no se produce. En el ejemplo, sin embargo, si el rectángulo que sirve de recuadro estuviera en primer plano y fuera trasparente, se verían los otros, pero no se los podría seleccionar. Para hacerlo, habría que enviar al fondo el recuadro.

Enviar al fondo E. Funciona con uno o más objetos. El efecto es el contrario a lo explicado en TRAER AL PRIMER PLANO.

Cuadrícula

Al diseñar formularios e informes, VFP puede mostrar un reticulado de líneas de puntos, que se hace visible o se oculta con VER – LíNEAS DE CUADRÍCULA. El líneas se pueden aproximar o alejar, modificando la cuadrícula, que es su componente básico. Una cuadrícula no es igual que una celda del reticulado, porque ésta normalmente abarca cuatro cuadrículas, dos horizontales y dos verticales. La cuadrícula, entonces, no es visible directamente. La cuadrícula se configura en el menú FORMATO – CONFIGURAR CUADRÍCULA, definiendo la cantidad de píxeles horizontales y verticales.

Una función del reticulado es servir como guía visual para ubicar los objetos con cierta precisión.

Otra función es cuando los objetos se arrastran con el mouse. El movimiento puede ir de cuadrícula en cuadrícula o de píxel en píxel. Este movimiento depende si está o no activa la opción FORMATO – FORZAR A

LA CUADRÍCULA.

Si está activo el forzado a la cuadrícula, el objeto arrastrado va saltando tantos píxeles como mida cada cuadrícula en alto o en ancho, según se lo desplace vertical u horizontalmente. Si no lo está, el objeto arrastrado avanza de píxel en píxel, con un movimiento más suave.

El forzado de la cuadrícula activo solamente afecta al arrastre con el mouse. Si el objeto seleccionado se mueve con las teclas de movimiento, siempre se va de píxel en píxel.

Menú Formato

Este menú varía cuando se está diseñando un formulario, un informe o una etiqueta.

Alinear. Tiene los siguientes comandos: ALINEAR LOS BORDES IZQUIERDOS, ALINEAR LOS BORDES DERECHOS, ALINEAR LOS BORDES SUPERIORES, ALINEAR LOS BORDES INFERIORES, ALINEAR CENTROS VERTICALMENTE, ALINEAR CENTROS HORIZONTALMENTE, CENTRAR VERTICALMENTE, CENTRAR HORIZONTALMENTE. El funcionamiento es igual a los correspondientes botones de la barra de herramientas DISTRIBUCIÓN.

Tamaño. En formularios, tiene los dos comandos siguientes:

AJUSTAR AL CONTENIDO funciona como la propiedad AUTOSIZE de los objetos de formulario, aunque no cambia el valor *Falso* de esta propiedad.

AJUSTAR A LA CUADRÍCULA cambia los bordes derecho e inferior del objeto de modo que coincidan con los bordes de las cuadrículas más próximas.

En informes, tiene los cinco comandos siguientes:

AJUSTAR A LA CUADRÍCULA funciona como en los formularios.

AJUSTAR AL MÁS ALTO necesita al menos dos objetos seleccionados y funciona como el botón MISMO ALTO de la barra de herramientas DISTRIBUCIÓN sin CTRL.

AJUSTAR AL MÁS BAJO necesita al menos dos objetos seleccionados y funciona como el botón MISMO ALTO de la barra de herramientas DISTRIBUCIÓN con CTRL.

AJUSTAR AL MÁS ANCHO necesita al menos dos objetos seleccionados y funciona como el botón MISMO ANCHO de la barra de herramientas DISTRIBUCIÓN sin CTRL

AJUSTAR AL MÁS ESTRECHO necesita al menos dos objetos seleccionados y funciona como el botón MISMO ANCHO de la barra de herramientas DISTRIBUCIÓN con CTRL.

Espacio horizontal. Tiene tres comandos.

HACER IGUAL funciona con al menos tres objetos seleccionados. Toma como referencia la separación entre el primer y segundo objetos de la izquierda y aplica esa separación entre el segundo y el tercero, entre el tercero y el cuarto, etc. La Figura 11.20.



Figura 11.20. Antes y después de espaciar horizontalmente

AUMENTAR funciona con al menos dos objetos seleccionados. Incrementa en un píxel la separación entre los objetos.

DISMINUIR funciona con al menos dos objetos seleccionados. Recude en un píxel la separación entre los objetos.

Espacio vertical. Tiene tres comandos, iguales en nombres a los de ESPACIADO HORIZONTAL, solamente que funcionan en sentido vertical. La figura 11.21 muestra un ejemplo de HACER IGUAL.



Figura 11.21. Antes y después de espaciar verticalmente

Traer al primer plano. Funciona como el correspondiente botón de la barra de herramientas DISTRIBUCIÓN.

Enviar al fondo. Funciona como el correspondiente botón de la barra de herramientas DISTRIBUCIÓN.

Agrupar. Este comando aparece al diseñar informes. Funciona con al menos dos objetos seleccionados. Forma con los objetos seleccionados un grupo gráfico, de modo que se los pueda seleccionar, mover y eliminar como un todo. Cuando se selecciona un grupo, no es posible acceder a las propiedades de los objetos componentes, por lo que éstos deben ser configurados individualmente antes de agruparlos. Si ya forman un grupo y se quiere alterar la propiedad de alguno de estos objetos, habrá que desagruparlos, modificarlos individualmente y luego volver a agruparlos, si hace falta.

El agrupamiento es jerárquico. Suponga que inicialmente se hacen tres grupos, A, B y C. Todos están en un primer nivel de agrupamiento. Luego se hace un grupo con dos de los grupos del primer nivel, A y B, con lo que habrá un segundo nivel de agrupamiento, (AB). Finalmente se agrupa el grupo del segundo nivel, (AB), con el del primer nivel, C, produciéndose un tercer nivel de agrupamiento, ((AB)C). Vea la Figura 11.22.



Figura 11.22. Tres niveles de agrupación gráfica

Desagrupar. Este comando aparece al diseñar informes. Funciona sobre uno o varios grupos, desagrupándolos. Si el agrupamiento es jerárquico, desagrupa los niveles correspondientes. Por ejemplo, en la Figura anterior, al desagrupar el grupo ((AB)C) quedan los grupos (AB) y C. Si se desagrupa C, se obtienen sus objetos libres. Si se desagrupa (AB), se obtienen los grupos A y B. Para obtener sus objetos libres, habrá que desagrupar A o B, según lo que se quiera.

Forzar a la cuadrícula. Cuando se activa, el movimiento de los objetos con el mouse avanza de cuadrícula en cuadrícula. Cuando se desactiva, el movimiento con el mouse es de píxel en píxel. El estado de la cuadrícula no afecta el movimiento de los objetos con las teclas, que siempre es de píxel en píxel.

Configurar cuadrícula. Muestra el cuadro de diálogo CONFIGURAR CUADRÍCULA, donde se pueden definir las dimensiones horizontal y vertical de la cuadrícula en píxeles.

En HERRAMIENTAS – OPCIONES – FORMULARIOS se puede modificar y predeterminar el valor de la cuadrícula para el DISEÑADOR DE FORMULARIOS.

En informes se puede especificar si la regla aparece en pulgadas o en píxeles.

Fuente. Este comando aparece al diseñar informes. Funciona en etiquetas y campos. Presenta el cuadro de diálogo FUENTE, donde se puede establecer el tipo de fuente, el estilo y el tamaño.

Alineación del texto. Este comando aparece al diseñar informes. Funciona en etiquetas y campos. Para etiquetas brinda opciones de alineación y de espaciado. Para campos sólo ofrece opciones de alineación.

Las opciones de alineación son: IZQUIERDA, CENTRAR, DERECHA.

Obsérvese que, al escribir el texto de una etiqueta, si se pulsa la tecla ENTER se abre una nueva línea de texto. Para terminar la creación de la etiqueta, se debe dar clic en un punto alejado del área de diseño o pulsar otro botón de la barra de herramientas CONTROLES DE INFORMES. Si la etiqueta tiene varias líneas de texto, posiblemente no todas tengan el mismo largo, por lo que son posibles las opciones de alineación.

Las opciones de espaciado son ESPACIO SENCILLO, ESPACIO1 1/2 y ESPACIO DOBLE.

Relleno. Este comando aparece al diseñar informes. Funciona en rectángulos y rectángulos redondeados. Posibilita rellenarlos con líneas horizontales, verticales y oblicuas, ofreciendo ocho modelos gráficos de

tramas.

Borde. Este comando aparece al diseñar informes. Funciona en líneas, rectángulos y rectángulos redondeados. Ofrece las opciones TRAZO FINO, 1 PUNTO, 2 PUNTOS, 4 PUNTOS, 6 PUNTOS, NADA, PUNTEADO, CON GUIÓN, GUIÓN – PUNTO y GUIÓN – PUNTO – PUNTO.

Modo. Este comando aparece al diseñar informes. Funciona en etiquetas, campos, rectángulos y rectángulos redondeados. Ofrece las opciones OPACO y TRANSPARENTE. Si un objeto es opaco, impide ver los controles que se encuentren por debajo. Si es transparente, los deja ver.

Cuarta parte. Ejercitación

Esta se ha agregado para ayudar a los lectores a practicar, a su propio ritmo, distintas formas de resolver problemas. La explicación de los pasos en cada ejercicio es acumulativa. Esto quiere decir que, al principio, se explican detalladamente, pero en ejercicios posteriores se van omitiendo.

Los problemas se plantean a partir de un conjunto de tablas, que se supone cargadas con una cantidad de registros importante. Las tablas a emplear tienen la estructura e índices que resume el siguiente cuadro:

Tabla	Campos				Índices		
	Nombre	Tipo	Ancho	Dec.	Nombre	Tipo	Expresión
art	art	С	3		art	principal	art
	nom	С	30		nom	normal	upper(nom)
	pre	n	8	2			
cli	cli	n	5		cli	principal	cli
	nom	С	30		nom	normal	upper(nom)
	dom	С	30		prvcli	normal	prv+str(cli,5)
	prv	С	2		prvnom	normal	prv+nom
	tel	С	10				
ven	ven	n	2		ven	principal	ven
	nom	С	30		nom	normal	upper(nom)
	com	n	5	2			
encfac	fac	n	8		fac	principal	fac
	fec	f	8		cli	normal	cli
	cli	n	5		ven	normal	ven
	ven	n	2		clifac	normal	str(cli,5)+str(fac,8)
linfac	fac	n	8		facart	principal	str(fac,8)+art
	art	С	3		art	normal	art
	can	n	4		fac	normal	fac
	pre	n	8	2			

En Tipo, c indica carácter; n, numérico; f, fecha.

Si no dispone de las tablas, puede construir su estructura e índices de acuerdo al cuadro. Pero también deberá cargar registros en cada una. Puede definir unos pocos artículos, clientes y vendedores en las respectivas tablas ART, CLI y VEN. Las facturas están distribuidas en dos tablas: ENCFAC contiene los datos de cabecera y LINFAC los datos de los distintos artículos vendidos en cada factura. Ambas tablas están vinculadas por el número de factura. Conviene que confeccione varios registros en ENCFAC, para distintos clientes y vendedores. Por cada registro de ENCFAC, debería haber varios registros en LINFAC.

Los nombres de campos, tablas e índices consisten en una abreviatura. Para entender su significado, se proporciona este otro cuadro:

Abreviatura art can cli clifac com dom encfac fac facart fec linfac nom pre prv prvcli prvnom tel ven	Significado Tabla de artículos; código de artículo Cantidad Tabla de clientes; código de cliente Índice por número de cliente + número de factura Porcentaje de comisión Domicilio Tabla con los encabezados de las facturas Número de factura Índice por número de factura + código de artículo Fecha Tabla con las líneas repetitivas de las facturas Nombre de artículo, cliente o vendedor Precio unitario Provincia Índice por provincia + código de cliente Índice por provincia + nombre de cliente Índice por provincia + nombre de cliente Índice por provincia + nombre de cliente
ven	rabia de vendedores; codigo de vendedor

Para no producir confusiones, seguiremos las siguientes convenciones y notación:

- Por brevedad, diremos diálogo por cuadro de diálogo; entorno por entorno de datos; barra por barra de herramientas.
- Dar clic secundario quiere decir pulsar el botón derecho del mouse.
- Los nombres de objetos, propiedades, eventos, tablas, campos e índices están en versalita. Los valores están en cursiva. Los procedimientos están en minúscula de igual ancho para cualquier carácter.
- Cuando digamos cosas como ARCHIVO NUEVO FORMULARIO NUEVO ARCHIVO, queremos decir que abra el menú ARCHIVO, elija la opción NUEVO, en el diálogo que resulta active la categoría ARCHIVO y pulse el botón NUEVO ARCHIVO.
- Para referir los valores lógicos que se pueden dar a una propiedad, los cuales aparecen en la lista de configuración de la ventana Propiedades, diremos *Verdadero* y *Falso*, aunque en la lista aparecen como *.T. Verdadero* y *.F. Falso*. Esto se aplica también a otros valores, como *0 Ninguna*, *1 Horizontal*, *2 Vertical*, *3 Ambas*, etc., que referiremos sin los números.
- A menudo haremos referencia a un campo de una tabla prefijando el campo con el nombre de la tabla. Así, LINFAC.ART quiere decir el campo Art de la tabla Linfac.

Esta parte tiene dos Capítulos. El Capítulo 12 contiene ejercitación sobre formularios y el Capitulo 13 sobre informes. En ambos se usan entornos de datos, cuando se trabaja con tablas.
Capítulo 12. Ejercicios con formularios

La mayoría de las propiedades de objetos de formulario son conocidas de los capítulos anteriores. Los comandos, eventos y métodos no lo son, pues no han sido tratados en esta obra. Sin embargo, para conseguir el cometido de varios de los formularios que se desarrollan a continuación, es necesario recurrir a ellos. El lector deberá aceptarlos, aunque no llegue a entenderlos bien. Si tiene interés en conocer estos temas, le dejamos la tarea de estudiarlos, para lo cual la Ayuda es suficiente. Habremos logrado un propósito no pretendido en esta obra, pero de indudable beneficio para él.

Ejercicio 1 a

Cree el formulario EJER1, que muestre todos los campos de la tabla CLI.

- 1. Cree un formulario nuevo. Una manera de hacerlo es con Archivo NUEVO FORMULARIO NUEVO ARCHIVO. Se abre el DISEÑADOR DE FORMULARIOS, en blanco.
- 2. Si no está visible la barra DISEÑADOR DE FORMULARIOS, muéstrela con VER BARRA DE HERRAMIENTAS DISEÑADOR DE FORMULARIOS ACEPTAR. Siempre es útil tenerla a la vista.
- 3. Pulse el botón ENTORNO DE DATOS de esa barra.
- 4. En la ventana del entorno, dé clic secundario y elija AGREGAR.
- 5. En el diálogo ABRIR, seleccione la tabla CLI y pulse ACEPTAR. Esto crea el cursor CLI en el entorno, que representa esa tabla.



- 6. Pulse CERRAR en el diálogo AGREGAR TABLA O VISTA. Este diálogo permaneció abierto para seguir agregando más tablas, si hacía falta.
- 7. Arrastre la línea que dice *Campos*, del cursor CLI al Formulario. Esto crea un LABEL y un TEXTBOX para cada campo. Las parejas están dispuestas verticalmente. La propiedad CAPTION de cada LABEL es el nombre del respectivo campo.
- 8. Salve los cambios con ARCHIVO GUARDAR. Dado que es la primera vez que hace esto, se abre el diálogo GUARDAR COMO. En GUARDAR FORMULARIO escriba EJER1 (sin espacio entre Ejer y 1) y pulse

GUARDAR. En adelante, como el formulario ya tiene nombre, ARCHIVO – GUARDAR salvará los cambios sin abrir ningún diálogo.

9. Pruebe lo hecho con FORMULARIO – EJECUTAR FORMULARIO. También puede dar clic secundario en el fondo del formulario y elegir EJECUTAR FORMULARIO.

Form1	
Cliente	txtCli
Nombr	e txtNom
Domici	lio txtDom
Provine	cia txtPrv
Teléfor	no txtTel

- 10. Estudie el formulario en ejecución. Luego cierre la ventana con el botón de cierre. Esto vuelve a tiempo de diseño.
- 11. Cierre el DISEÑADOR DE FORMULARIOS con ARCHIVO CERRAR o con ESCAPE.

El cliente mostrado en el formulario es el que aparece primero en la tabla. Como no hay un índice activado, el primero es el que tiene número de registro 1.

Ejercicio 1 b

Mejore EJER1 para que muestre distintos clientes, registro a registro.

- 1. Abra EJER1 con ARCHIVO ABRIR. En TIPO DE ARCHIVOS elija *Formulario*. Seleccione EJER1 en la lista de archivos y pulse ACEPTAR.
- 2. Active el botón CONTROLES DE FORMULARIOS en la barra DISEÑADOR DE FORMULARIOS. Esto abre la barra de igual nombre que el botón.
- 3. Dé clic en BOTÓN DE COMANDO y dé clic en el formulario, donde quiera ubicarlo.
- 4. Dé clic secundario en el COMMANDBUTTON y elija PROPIEDADES o use el botón PROPIEDADES de la barra DISEÑADOR DE FORMULARIOS. Ambas acciones abren el diálogo PROPIEDADES.
- 5. En la ficha TODAS o en la ficha DISTRIBUCIÓN, elija la propiedad CAPTION y escriba Anterior.
- 6. Cierre la ventana PROPIEDADES con el botón de cierre.
- 7. Cree otro botón de comando y en CAPTION escriba Siguiente.
- 8. Dé doble clic en el botón de comando Anterior. También puede seleccionarlo y pulsar el botón VENTANA CÓDIGO de la barra DISEÑADOR DE FORMULARIOS. Ambos procedimientos abren la ventana CÓDIGO. En la barra superior gris, junto a donde dice Objeto se ve Command1 y a donde dice Procedimiento se ve Click. Escriba las siguientes líneas de comandos:

```
skip -1
if bof()
go top
```

```
endif
thisform.refresh
```

9. En la lista Objeto, elija Command2. En la lista Procedimiento, elija Click. Escriba:

```
skip
if eof()
  go top
endif
thisform.refresh
```

- 10. Guarde los cambios y cierre la ventana CÓDIGO. Lo puede hacer con ARCHIVO GUARDAR y ARCHIVO CERRAR. Ambas acciones se consiguen de una vez con CTRL + W.
- 11. Ejecute el formulario. Pulse los botones que dicen *Anterior* y *Siguiente*. Cuando está al principio de la tabla, *Anterior* no tiene efecto; lo mismo sucede al final con el botón *Siguiente*. Es lo que se ha programado. Cierre el formulario en ejecución.

xtCli
xtNom
xtDom
xtPrv
xtTel
Anterior Siguiente

12. Cierre el formulario en diseño.

Ejercicio 1 c

Mejore EJER1 con un botón de comando para cerrarlo y con otros detalles.

- 1. Abra EJER1 con ARCHIVO ABRIR. En TIPO DE ARCHIVO elija FORMULARIOS, seleccione EJER1 y pulse ACEPTAR.
- Agregue un botón de comando cuyo CAPTION sea Salir. Como se llama COMMAND3, que no indica nada, escriba Salir en la propiedad NAME. Este nuevo nombre describe su propósito. De paso, corrija NAME de COMMAND1 y COMMAND2 con Anterior y Siguiente.
- 3. En la ventana Código, procedimiento CLICK del botón SALIR, escriba la línea

thisform.release

- 4. Corrija la propiedad CAPTION del formulario, que es el objeto FORM1, con el texto *Clientes de nuestra empresa*.
- 5. Guarde los cambios para ejecutar el formulario. Si quiere ejecutarlo y hay cambios no guardados, un mensaje le recordará hacerlo. Ejecute el formulario. Termine la ejecución pulsando el botón SALIR.

Clientes de nuestra e	empresa
Cliente	txtCli
Nombre	txtNom
Domicilio	txtDom
Provincia	txtPrv
Teléfono	txtTel
	Anterior Siguiente Salir

Ejercicio 1 d

Mejore EJER1 para avance o retroceda por los registros ordenados por nombre de cliente.

- 1. Abra el entorno de EJER1.
- 2. Seleccione el cursor CLI y abra la ventana PROPIEDADES. En la propiedad ORDER, elija en la lista que está bajo las orejas de la fichas el índice NOM.
- 3. Cierre el entorno, guarde los cambios y ejecute el formulario. Avance y retroceda por los registros. Observe que están en orden alfabético.

Ejercicio 1 e

Mejore EJER1 para cambiar el orden de los registros: por nombre o por número de cliente.

- 1. Agregue un LABEL. En CAPTION escriba Ordenar por.
- 2. Agregue un grupo de opciones. Por defecto son dos, precisamente las requeridas.
- 3. En CONTROLSOURCE del OPTIONGROUP escriba *Ordenar*, que será el nombre de una variable de memoria. Elija *Verdadero* en la lista de la propiedad AUTOSIZE. Cuando una propiedad tiene valores predefinidos, los puede cambiar uno a uno dando doble clic en la línea donde está la propiedad. Si los valores predefinidos son *Verdadero* y *Falso*, también puede pulsar la tecla del punto en esa línea.
- 4. En la ventana PROPIEDADES, elija el objeto OPTION1 en la lista de objetos, que está arriba. Escriba *Nombre* en CAPTION y en NAME. Elija *Verdadero* en AUTOSIZE.
- 5. Elija el objeto OPTION2. Escriba Cliente en CAPTION y en NAME. Elija Verdadero en AUTOSIZE.
- 6. Cierre la ventana PROPIEDADES. Si ve que está en el interior del OPTIONGROUP, dé clic fuera de él.
- Dé doble clic en el OPTIONGROUP. Se abre la ventana CÓDIGO para este objeto. Busque el procedimiento CLICK y escriba:

```
if ordenar = 1
   thisform.dataenvironment.cursorl.order = 'nom'
else
   thisform.dataenvironment.cursorl.order = 'cli'
endif
go top
```

thisform.refresh

Otra forma de cambiar el índice activo es:

```
if ordenar = 1
   set order to nom
else
   set order to cli
endif
go top
thisform.refresh
```

 Salga de la ventana CÓDIGO con CTRL + W. Dé doble clic en el fondo del formulario. Esto lo lleva nuevamente a la ventana CÓDIGO, para este objeto. Si prefiere, sin cerrar esta ventana, puede elegir cualquier objeto existente en la lista OBJETO y el evento necesitado en la lista PROCEDIMIENTO. Asegúrese de estar en el objeto FORM1, procedimiento LOAD. Escriba:

```
clear memory
public ordenar
ordenar = 1
```

El primer comando eliminará todas las variables existentes en memoria. El segundo creará la variable ORDENAR de tipo público, es decir, disponible en todos los procedimientos del formulario e incluso al terminar la ejecución de éste. El tercero dará a ORDENAR el valor 1, para que se vea activo inicialmente el botón de opción llamado NOMBRE, en correspondencia con el índice activado inicialmente en CURSOR1. El evento LOAD se produce antes de comenzar la ejecución del formulario, momento apropiado para los comandos anteriores.

9. Si quiere eliminar la variable ORDENAR al terminar la ejecución del formulario, en su evento UNLOAD escriba:

clear memory

10. Pruebe las mejoras.

Clientes de nuestra e	empresa	
Cliente	txtCli	Ordenar por
Nombre	txtNom	O Nombre O Cliente
Domicilio	txtDom	
Provincia	txtPrv	
Teléfono	txtTel	
	Anterior Siguiente	Salir

Importante. En tiempo de diseño, dé clic secundario en el OPTIONGROUP y seleccione *Editar* (o *Modificar*, depende de la versión). El objeto adquiere un borde de color. OPTIONGROUP es un contenedor de objetos OPTIONBUTTON. El borde indica que está en su interior y que puede seleccionar cualquiera de los objetos

contenidos. Si presta atención, cuando está en la ventana PROPIEDADES y penetra a los componentes del OPTIONGROUP, también verá el borde de color.

Ejercicio 2 a

Cree el formulario EJER2. Deberá tener un GRID que muestre la tabla CLI y un botón SALIR. Los registros deberán estar ordenados por el nombre de cliente y no podrán ser modificados.

1. Cree el formulario EJER2. Para hacerlo, en la ventana COMANDOS puede emitir:

```
modify command ejer2
```

- 2. En el entorno, agregue la tabla CLI. En el cursor CLI, haga *Verdadero* la propiedad READONLY y elija el índice NOM en la propiedad ORDER.
- 3. Arrastre el título del cursor CLI al formulario. Esto producirá un GRID. Cierre el entorno.
- 4. Añada un botón de comando al formulario, con NAME y CAPTION Salir. En el procedimiento CLICK del botón escriba:

thisform.release

5. Ejecute el formulario. No podrá modificar lo que ve, porque la tabla es de sólo lectura.

Form1							
	-						
		Cli	Nom	Dom	Prv		
							Salir
							Oalli
						$\mathbf{\nabla}$	

Ejercicio 2 b

Mejore EJER2. Como el ancho de las columnas y del GRID es insuficiente, corríjalos. Cuando vea correctamente todos los campos, la barra de desplazamiento horizontal no tendrá sentido. Como la tabla del GRID es de sólo lectura, tampoco tendrá sentido la columna para marcar registros a destruir.

- Dé clic secundario en el GRID y elija *Editar* (o *Modificar*). Active el botón PROPIEDADES. Cambie la propiedad WIDTH de cada columna, de COLUMN1 a COLUMN5, por los siguientes valores: 50, 200, 200, 60 y 80. Puede elegir los objetos en la misma ventana PROPIEDADES o alternar entre ésta y el GRID. En este caso, si está en la ventana, dé clic en el GRID para activarlo; luego, como está en su interior, dé clic en la columna deseada y vuelva a dar clic en la ventana, para activarla.
- 2. Modifique los valores CAPTION de cada HEADER, por *Cliente, Nombre, Domicilio, Provincia, Teléfono.* Centre estos encabezados valiéndose de la propiedad ALIGNMENT de cada HEADER.
- 3. Elija GRID1 en la lista de objetos de la ventana PROPIEDADES. Active *Falso* en DELETEMARK y VERTICAL en SCROLLBARS. Cierre la ventana PROPIEDADES.

- 4. Seleccione el GRID. Con SHIFT + → o SHIFT + ← varíe el ancho para ver bien las cuatro columnas. Para variar la ubicación del GRID en el formulario, use las teclas de flechas.
- 5. Pruebe el formulario.

Form1							
	Cliente	Nombre	Domicilio	Provincia	Teléfono		
							Salir
						▼	
			1		<u> </u>		

Ejercicio 3

Si tiene un proyecto que contenga las tablas, este ejercicio es innecesario. Cree el proyecto FAC para contener las tablas y formularios creados hasta ahora. Esto facilitará el manejo de esos y otros archivos.

- Con ARCHIVO NUEVO, en TIPO DE ARCHIVOS elija PROYECTO y pulse el botón NUEVO ARCHIVO. En el diálogo CREAR escriba el nombre *Fac* para el proyecto y pulse GUARDAR. Se abrirá el ADMINISTRADOR DE PROYECTOS, con varias fichas.
- 2. Elija la ficha DATOS. Seleccione BASES DE DATOS. Pulse el botón NUEVO. Escriba el nombre *Fac* en el cuadro BASE DE DATOS y pulse GUARDAR. Aparece la ventana DISEÑADOR DE BASES DE DATOS.
- Con clic secundario elija AGREGAR TABLA. En el diálogo ABRIR elija la tabla ART y pulse ACEPTAR. En el DISEÑADOR DE BASES de datos aparecerá un cursor con el nombre de la tabla, sus campos e índices. Cierre el diseñador con el botón X o con ESCAPE. Advierta que sólo puede agregar tablas libres.
- 4. En la ficha DATOS del administrador, el rubro BASES DE DATOS tiene un + a la izquierda. Púlselo. Verá que tiene una base de datos llamada FAC, con +. Púlselo. Verá el rubro TABLAS, con +. Púlselo. Verá que tiene una tabla llamada ART, con +. Púlselo. Verá los campos e índices, ordenados alfabéticamente. Cada vez que pulsó un +, se convirtió en -. Pulse el de ART. Dejan de verse los campos e índices y el se convierte en +.
- 5. Para agregar las demás tablas, use otro método. Asegúrese de estar en el ámbito del rubro TABLAS del proyecto. Pulse el botón AGREGAR. En el diálogo ABRIR, elija la tabla CLI y pulse ACEPTAR. Repita el proceso para las tablas VEN, ENCFAC y LINFAC.
- 6. Elija la ficha DOCUMENTOS, rubro FORMULARIOS. Siguiendo un procedimiento similar al explicado, agregue los FORMularios EJER1 y EJER2.
- 7. Cierre el administrador con el botón X o con ESCAPE.

Ejercicio 4 a

Cree el formulario EJER4 para que muestre en un GRID los registros de LINFAC, ordenados por código de artículo. Use el generador de GRID.

- Abra el proyecto FAC con ARCHIVO ABRIR. Es posible que el menú ARCHIVO tenga, entre sus opciones, un listado de los últimos proyectos abiertos. Si aparece el nombre del proyecto FAC, también puede abrirlo eligiéndolo.
- 2. Elija la ficha DOCUMENTOS, rubro FORMULARIOS. Pulse el botón NUEVO. Es otra forma de crear un formulario, esta vez desde un proyecto.
- 3. Introduzca un GRID al formulario.
- 4. Dé clic secundario sobre el GRID y elija GENERADOR. Aparece el diálogo GENERADOR DE CUADRÍCULAS.
- 6. Seleccione la tabla LINFAC. Aparecerán sus campos en CAMPOS DISPONIBLES. Pulse el botón ►► para transferir todos los campos a CAMPOS SELECCIONADOS.
- 7. En la ficha ESTILO, si desea, elija un formato para el GRID.
- 8. En la ficha DISTRIBUCIÓN verá los registros de la tabla LINFAC. Cada campo ocupa una columna, cuyo encabezado es el nombre del campo. Puede elegir cualquier columna dando clic en ella. En el cuadro Título puede escribir otro encabezado para la columna. Para aumentar o disminuir el ancho de una columna, tome con el mouse el borde derecho del encabezado y arrástrelo lateralmente. Para cambiar de lugar una columna, ponga el mouse en el encabezado hasta que tome la forma ♣ y arrástrela.
- 9. Acepte lo realizado en el generador, para cerrarlo.
- 10. Puede ahora retocar el GRID que está en el formulario, con la ventana PROPIEDADES. Por ejemplo, cambiando las propiedades DELETEMARK y SCROLLBARS.
- 11. Guarde el formulario con ARCHIVO GUARDAR o con CTRL + S, dándole el nombre *Ejer4*. Ejecútelo. Deberá terminar la ejecución con el botón X, porque no ha incluido un botón de comando para ello.

Form1					
	Factura	Artículo	Cantidad	Precio	
		1			
					▼

Ejercicio 4 b

Modifique EJER4 para que muestre el nombre correspondiente a cada código de artículo.

1. Abra el entorno y agregue la tabla ART. Arrastre el campo ART del cursor LINFAC al índice ART del cursor ART. Esto crea una relación, que se ve como una línea que conecta el campo al índice. En este caso particular, como el índice está construido por una expresión que es el campo ART, también puede trazar la relación del campo LINFAC.ART al campo ART.ART. Cierre el entorno.

- 2. Corrija la propiedad COLUMNCOUNT del GRID, con el valor 5. Esto agrega una nueva columna. Selecciónela. Dé a CONTROLSOURCE el valor *Art. Nom* y a WIDTH un ancho de 200. Elija el HEADER de esta columna y dé a CAPTION el valor *Descripción*.
- 3. Agregue un botón SALIR, como ya conoce.
- 4. Ejecute el formulario.

Form1							
							_
	Factura	Artículo	Cantidad	Precio	Descripción		
							Salir
						▼	
L		1			I.		I

Ejercicio 5 a

A partir de EJER4, cree EJER5, para que cada artículo muestre sus ventas.

- 1. Abra EJER4.
- 2. Emita ARCHIVO GUARDAR COMO. Escriba Ejer5 para el nuevo archivo y guárdelo. Cierre EJER4.
- 3. Agregue EJER5 al proyecto y modifíquelo.
- 4. En el entorno, seleccione la relación con un clic y pulse DELETE para destruirla.
- 5. Trace una nueva relación, de ART.ART al índice ART de LINFAC.
- 6. En el GRID, reemplace el valor *Linfac* de la propiedad RECORDSOURCE por *Art*. Debe hacerlo, porque ahora la tabla desde donde parte la relación es ART.
- 7. Escriba los valores CONTROLSOURCE de las columnas, pues al cambiar RECORDSOURCE se perdieron:
 - COLUMN1 Linfac.Fac COLUMN2 Art.Art COLUMN3 Linfac.Can COLUMN4 Linfac.Pre COLUMN5 Art.Nom
- 8. Ponga la columna del nombre de artículo en primer lugar y la del código de artículo en el segundo. Puede hacerlo modificando COLUMNORDER de cada columna:
 - COLUMN1 3 COLUMN2 2 COLUMN3 4 COLUMN4 5 COLUMN5 1

También puede hacerlo seleccionando cada columna dentro del encabezado correspondiente y arrastrándola al lugar deseado. 9. Pruebe el formulario. Observe que cada artículo parece haberse vendido una vez, lo cual no es cierto. Eso se debe que la relación es uno a uno.

Artículo Código Factura Cantidad Precio Salir Salir	Form1							
Artículo Código Factura Cantidad Precio ▲	_	- 1				n		
Salir		Artículo	Código	Factura	Cantidad	Precio		
Salir								
								Salir
							V	
							•	

Ejercicio 5 b

Modifique EJER5 para que muestre todas las ventas de un artículo.

- 1. En el entorno, seleccione la relación. En el diálogo PROPIEDADES, ficha DATOS, elija el valor Verdadero para la propiedad ONETOMANY.
- 2. Pruebe el formulario. Observe el aspecto del GRID en los datos provenientes de la tabla ART. Esto enfatiza la relación uno a varios. El GRID ejecutado se verá más o menos así:

Artículo	Código	Factura	Cantidad	Precio	
Mesa de madera	C01	527	10	250	
		649	8	250	
		715	15	260	
Mesa de metal	C02	247	6	180	
		1268	24	195	
		1360	8	195	
		1724	50	200	
Mesa de plástico	C03	75	25	85	
		128	30	85	▼

Ejercicio 5 c

¿Qué hace la una relación uno a muchos? A simple vista vemos que afecta los campos de la tabla ART, madre de la relación. Cuando la relación era 1 – 1, el sombreado no se veía. Ahora, que es 1 – N, aparecen varios registros, tantos como registros hay en LINFAC, hija de la relación. ¿Qué significa el sombreado? Si nos colocamos en la línea cuyo nombres es *Mesa de madera* y luego descendemos a la segunda y a la tercera, en la tabla ART siempre estaremos en el mismo registro, pero en la tabla LINFAC iremos cambiando de registro (los que tienen Factura 527, 649 y 715). Lo mismo pasará con *Mesa de metal* y *Mesa de plástico*. La conclusión es que la relación uno a varios permite seleccionar registros de la tabla hija indirectamente, desplazándonos por los registros reales y "virtuales" de la tabla madre. Comprobemos esto.

1. Agregue los siguientes objetos, como se indica:

Objeto	Propiedad	Valor
LABEL	CAPTION	Registro de Art
ТехтВох	CONTROLSOURCE	regart
	NAME	regart
LABEL	CAPTION	Registro de Linfac
ТехтВох	CONTROLSOURCE	reglinfac
	NAME	reglinfac

2. En el evento AFTERROWCOLCHANGE del GRID, escriba:

```
regart=recno('art')
reglinfac=recno('linfac')
thisform.refres
```

3. Guarde y ejecute el formulario. Compruebe en los TextBoxes los números de registro.

	Artículo	Código	Factura	Cantidad	Precio		
	Mesa de madera	C01	527	10	250		
			649	8	250		
			715	15	260		
	Mesa de metal	C02	247	6	180		
			1268	24	195		
			1360	8	195		
			1724	50	200		
	Mesa de plástico	C03	75	25	85		
			128	30	85	▼	
Re	egistro de Art	Regist	tro de Linfa	C			

El efecto del sombreado que produce la relación 1 - N en el GRID se observa en los campos de la tabla madre. Incluso si el GRID no mostrara campos de la tabla hija, el efecto también se vería.

Podría probar esto si elimina una por una las columnas que muestran campos de LINFAC. Para eliminar una columna se la debe seleccionar en la ventana PROPIEDADES, luego debe dar un solo clic en el Grid para activarlo y pulsar la tecla Delete. VFP preguntará si está seguro de eliminar la columna y todos sus objetos. Este procedimiento es para cuando ha alterado la propiedad COLUMNORDER de una o más columnas. Si no alteró COLUMNORDER y las columnas a eliminar fueran las últimas del GRID, basta con reducir el número de columnas indicado en COLUMNCOUNT.

Ejercicio 6

Cree el formulario EJER6, para que muestre el desarrollo completo de cada factura, que se encuentra desdoblada en las tablas ENCFAC y LINFAC. Muestre para cada artículo el producto de la cantidad por el precio.

- 1. En el ámbito FORMULARIOS del proyecto FACTURAS, pulse NUEVO.
- 2. Introduzca las tablas ENCFAC y LINFAC al entorno.
- 3. Cree una relación entre ENCFAC.FAC y LINFAC.FAC.
- 4. Cree GRID1, arrastrando el título del cursor ENCFAC del entorno al formulario. Siguiendo igual procedimiento, cree GRID2, con el cursor LINFAC.

- 5. Modifique la propiedad COLUMNCOUNT de GRID2, dándole el valor 5.
- 6. En la quinta columna de GRID2, en la propiedad CONTROLSOURCE escriba Linfac.Can * Linfac.Pre.
- 7. En el HEADER de esta columna, escriba Importe en CAPTION.
- 8. Ejecute el formulario. Observe que los registros de GRID2 tienen el mismo número de factura que el registro actual en GRID1. Cambie de registro en GRID1 y observe GRID2.

Form	า1											
_												
		Fac	Fec	Cli	Ven		Fac	Art	Can	Pre	Importe	
		25	10-05-03	14	8	▲	27	A02	1	100	100	
		26	10-05-03	22	1		27	A15	2	35	70	
		27	10-05-03	1	5		27	C47	10	237	2370	
		28	10-05-03	45	2		27	D98	5	64	320	
		29	11-05-03	6	2							
		30	11-05-03	36	7							

Sin importar el valor de ONETOMANY, por el hecho de ver cada tabla en un GRID distinto se da el efecto uno a varios automáticamente. Si ambas tablas se vieran en un único GRID, la propiedad tiene importancia, como se ha visto en los Ejercicios 5 a y 5 b.

Ejercicio 7

Cree EJER7 a partir de EJER6. Debe mostrar el nombre de los clientes y vendedores.

- 1. Agregue las tablas CLI y VEN al entorno.
- 2. Trace una relación de ENCFAC.CLI a CLI.CLI y otra de ENCFAC.VEN a VEN.VEN.
- 3. En GRID1, escriba *Cli. Nom* en CONTROLSOURCE de la columna que antes decía *Encfac. Cli.* Escriba *Ven.Nom* en CONTROLSOURCE de la columna que antes decía *Encfac.Ven*.
- 4. Pruebe el formulario. Tome nota de las columnas que resultan estrechas para el contenido.
- 5. En tiempo de diseño, corrija las propiedades WIDTH de las columnas estrechas. Puede escribir directamente el ancho o modificarlo indirectamente, arrastrando la línea vertical derecha del encabezado, cuando el puntero toma la forma de una doble flecha horizontal.
- 6. Vuelva a ejecutar el formulario. Repita los ajustes si hace falta.

Ejercicio 8 a

Cree EJER8 para que muestre un COMBOBOX con el nombre y código de cada vendedor.

- 1. Cree un nuevo formulario.
- 2. Agregue un cuadro combinado.
- 3. Dé clic secundario en el COMBOBOX y elija GENERADOR.
- 4. En la ficha ELEMENTOS DE LA LISTA, elija CAMPOS DE UNA TABLA O UNA VISTA en RELLENAR LOS ELEMENTOS DE LA LISTA CON.

- 5. En BASES DE DATOS Y TABLAS active la base de datos FAC.
- 6. Seleccione la tabla VEN en la lista inferior.
- 7. Mueva los campos Nom y Ven, de Campos DISPONIBLES a CAMPOS SELECCIONADOS.
- 8. En la ficha ESTILO, active la opción Sí, ACTIVAR LA BÚSQUEDA INCREMENTAL.
- 9. En la ficha Distribución, active Ajustar el ancho del cuadro combinado para mostrar todas las columnas.
- 10. En el modelo, con datos reales, ajuste si hace falta el ancho de las columnas.
- 11. Pulse ACEPTAR para que el generador transforme las opciones elegidas en propiedades y se cierre. Si abre el entorno, verá que el generador ha agregado un cursor para la tabla VEN.
- 12. Guarde y ejecute el formulario. Pruebe la búsqueda incremental.
- 13. Los nombres de la primera columna no están ordenados, porque la tabla VEN no lo está. Vuelva al tiempo de diseño y, en el entorno, elija NOM para la propiedad ORDER del cursor.
- 14. Guarde y ejecute el formulario.

Form1		
Rosas, Carlo	os	▼
Ante, César	28	
Aranda, Inés	s 5	
Cortés, Davi	id 21	
Díaz, Luis	16	
Figueroa,	3	▼
Cortes, Davi Díaz, Luis Figueroa,	a 21 16 3	▼

Ejercicio 8 b

En el EJERCICIO 8 a las opciones del COMBOBOX se construyeron con los campos NOM y VEN de la tabla VEN. Una característica importante cuando las opciones se construyen de esta forma es que, al elegir una opción, el puntero de la tabla se ubica en el registro de donde proviene tal opción. Esta sincronía es muy útil, como veremos en el EJERCICIO 8 c. Antes de ello, comprobemos que la sincronía se produce.

- 1. Abra el entorno de datos y arrastre al formulario el título del cursor VENpara generar un GRID.
- 2. En el evento CLICK del COMBOBOX escriba lo siguiente:

```
thisform.grid1.setfocus
```

- 3. Salve los cambios y ejecute el formulario. Verá que el puntero de registros se coloca en el registro correspondiente a la opción elegida. Esto comprueba la sincronía mencionada.
- Ahora que hemos comprobado la sincronía, el GRID construido para la misma tabla que se usa para fabricar las opciones del COMBOBOX carece de sentido. Elimine el GRID, el evento CLICK del COMBO-BOX y guarde los cambios.

Ejercicio 8 c

Modifique EJER8, para que muestre en un GRID las facturas del vendedor elegido en el COMBOBOX.

1. Agregue la tabla ENCFAC al entorno.

- 2. Trace una relación desde VEN.VEN a ENCFAC.VEN.
- 3. Arrastre el título del cursor ENCFAC del entorno al formulario, para generar un GRID.
- 4. Cierre el entorno, retoque el aspecto del GRID si desea, guarde el formulario y ejecútelo.

Al seleccionar un vendedor en la lista del COMBOBOX, el puntero de la tabla VEN se ubica en el registro correspondiente, como comprobamos en virtud de la sincronía. La relación del entorno hace que el GRID muestre los registros de ENCFAC que corresponden al vendedor seleccionado. Este Ejercicio muestra una aplicación típica de COMBOBOX, para lo cual no es necesario definir su propiedad CONTROLSOURCE.

Form	1					
	٦í					
		az, Luis Fac	Fec	Cli	Ven	
		25	10-05-03	129	16	
		32	12-05-03	34	16	
		796	11-11-03	27	16	
		1025	20-12-03	89	16	
		1238	29-12-03	256	16	▼
		1230	29-12-03	200	10	•

Ejercicio 9 a

Cree el formulario EJER9, con un COMBOBOX de una columna y opciones construidas con valores introducidos a mano. Al seleccionar una opción, el COMBOBOX debe guardar el valor visualizado en una variable.

- 1. Cree un nuevo formulario, agréguele un COMBOBOX y llame al generador de este objeto.
- 2. En la ficha Elementos de la lista, elija Datos introducidos manualmente en Rellenar los ELEMENTOS DE LA LISTA CON.
- 3. En COLUMNAS, establezca 1.
- 4. En COLUMNA1, escriba un valor por línea: Argentina, Bolivia, Brasil, Chile, Perú.
- 5. En la ficha ESTILO, active LISTA DESPLEGABLE.
- 6. En la ficha DISTRIBUCIÓN, active AJUSTAR EL ANCHO DEL CUADRO COMBINADO para ver bien la columna.
- 7. En la ficha VALOR, escriba País en NOMBRE DE CAMPO. País es una variable, no un campo.
- 8. Acepte lo definido en el generador.
- 9. Agregue un TEXTBOX, cuyo CONTROLSOURCE sea País.
- 10. En el procedimiento LOAD del formulario, escriba:

```
public país
país = ''
```

11. Guarde y ejecute el formulario. Elija una opción del COMBOBOX. Esto almacena en *País* el valor elegido. Parece no ser así, pues el contenido del TEXTBOX no se modifica, pese a que su CONTROLSOURCE también es *País*. Esto es aparente: si le da enfoque, con clic o TAB, verá el nuevo contenido.

Form1			
	Perú	▼	
	Argentina		
	Bolivia		
	Brasil		Perú
	Chile		
	Perú	▼	

Ejercicio 9 b

Corrija EJER9 para que refresque automáticamente lo visualizado en el TEXTBOX.

1. En el procedimiento CLICK del COMBOBOX, escriba:

```
país = this.value
thisform.text1.refresh
```

This indica el objeto para el que se escribe el procedimiento, en este caso el COMBOBOX. *Value* es una propiedad que guarda el valor seleccionado en el COMBOBOX. *Thisform* se refiere al formulario. *Text1* es el nombre convencional del TEXTBOX, ya que no lo hemos modificado en la propiedad NAME. *Refresh* es un método que "refresca" TEXT1, haciendo que muestre el valor actual de su CONTROLSOURCE.

2. Pruebe el formulario.

Ejercicio 10

Cree el formulario EJER10. Debe tener un COMBOBOX de dos columnas, construido a mano, sin usar el generador. Al elegir una opción, debe guardar el valor de la segunda columna de esa opción en una variable. Muestre la variable en un TEXTBOX.

- 1. Cree un nuevo formulario y agréguele un COMBOBOX.
- 2. Seleccione el COMBOBOX y abra la ventana PROPIEDADES.
- 3. En la ficha DATOS, modifique lo siguiente:

PropiedadValorRowSourceType1 – ValoresRowSourceArgentina,4,Bolivia,9,Brasil,7,Chile,14,Perú,1ControlSourceCódigoBoundColumn2

1 – Valores indica que las opciones van a ser escritas en el mismo COMBOBOX. Código es la variable para almacenar un valor de la opción elegida. El 2 de BOUNDCOLUMN indica que el valor a almacenar debe ser el de la segunda columna de la opción elegida.

4. En la ficha DISTRIBUCIÓN, modifique lo siguiente:

PropiedadValorCOLUMNCOUNT2COLUMNLINESVerdaderoCOLUMNWIDTHS50,20

COLUMNCOUNT establece dos columnas, por lo que los valores de ROWSOURCE se distribuirán en la lis-

ta de esta manera:

Argentina	4	Opción 1
Bolivia	9	Opción 2
Brasil	7	Opción 3
Chile	14	Opción 4
Perú	1	Opción 5

- 5. Agregue un TEXTBOX al formulario, cuyo CONTROLSOURCE sea la variable Código.
- 6. En el evento LOAD del formulario, escriba el siguiente procedimiento:

```
public código
código = 0
```

7. En el procedimiento CLICK del COMBOBOX, escriba:

```
código = this.value
thisform.text1.refresh
```

- Guarde y ejecute el formulario. Elija la opción *Brasil* en el COMBOBOX y dé clic en el TEXTBOX para refrescarlo. Verá que no tiene 7, sino 3. ¿Qué pasó? Elija la opción *Chile* y dé clic en el TEXTBOX. No tiene 14, sino 4. ¿Qué pasó de nuevo? No se ha guardado el valor de la segunda columna, sino el número de la opción.
- 9. El funcionamiento extraño del punto anterior se debe a que Código es una variable numérica. En este caso, cumple un papel fundamental la propiedad BOUNDTO. Si vale Falso, guarda el número de opción. Si vale Verdadero, guarda el valor real que está en la columna indicada. Corrija entonces la propiedad BOUNDTO, haciéndola Verdadero y vuelva a probar el formulario. ¡Ahora sí!

Ejercicio 11

Cree el formulario EJER11. Debe mostrar los registros de ENCFAC en un GRID. Debe tener un COMBOBOX con los campos NOM y CLI de la tabla CLI. El COMBOBOX servirá para cambiar el dato del campo CLI del registro actual de la tabla ENCFAC.

- 1. Cree un nuevo formulario.
- 2. Agregue un GRID con los campos de la tabla ENCFAC. Use el procedimiento que más le guste.
- 3. Agregue un COMBOBOX, usando el procedimiento que desee para establecer las propiedades. Asegúrese que tengan los valores que siguen:

Valor
Verdadero
2
Encfac.Cli
Cli.Nom,Cli
6 – Campos
2
Verdadero
150,30

6 - Campos indica que la lista de opciones se va a construir con campos de una tabla. *Encfac.Cli* indica que el valor de la segunda columna de la opción elegida se debe guardar en el campo CLI de la tabla ENCFAC. La tabla donde se guardará lo elegido (ENCFAC) no debe ser la tabla con la cual se construye la lista (CLI).

4. Ejecute el formulario. Estando en un registro cualquiera del GRID, pulse el COMBOBOX. Elija un cliente y recuerde el valor de la segunda columna. Dé clic en el GRID y verá que se refresca el campo CLI con el valor recordado.

En este Ejercicio aparece una doble necesidad de refrescado. Una, que cuando se cambia de registro en el GRID, debería cambiar automáticamente el valor visualizado en el COMBOBOX, en correspondencia con el valor del campo CLI de tal registro. Otra, que cuando se elige otro cliente en el COMBOBOX, debería verse inmediatamente el cambio en el campo CLI del registro actual del GRID.

5. Para resolver el primer refrescado, en el procedimiento AFTERROWCOLCHANGE del GRID, escriba (la primera línea ya viene escrita):

```
LPARAMETERS nColIndex thisform.combol.refresh
```

6. Para resolver el segundo refrescado, en el procedimiento CLICK del COMBOBOX, escriba:

```
=dodefault()
thisform.grid1.refresh
```

Ejercicio 12 a

Cree el formulario EJER12. Vaya guardándolo a medida que avanza en los pasos.

1. Cree un formulario nuevo. Agregue:

Un LABEL1 junto a un SPINNER1 Un LABEL2 junto a un OPTIONGROUP1 Un LABEL3 junto a un OPTIONGROUP2 Un LABEL4 junto a un SPINNER2 Un LINE1 oblicuo Un SHAPE1 grande que los cubra

- 2. En la barra DISEÑADOR DE FORMULARIOS, pulse el botón DISTRIBUCIÓN. Se abre la barra DISTRIBUCIÓN.
- 3. Seleccione SHAPE1 y pulse el botón ENVIAR AL FONDO, para que quede tras los otros objetos.
- 4. Mueva LABEL1, SPINNER1, LABEL2, OPTIONGROUP1, LABEL3, OPTIONGROUP2, LABEL4 y SPINNER2, uno por uno, de modo que queden más o menos a la misma altura y con igual separación horizontal.
- Seleccione esos ocho objetos manteniendo oprimida SHIFT y dando clic en cada uno. Si por error selecciona uno no deseado, vuelva a dar clic sobre él. Pulse el botón ALINEAR LOS BORDES SUPERIORES de la barra DISTRIBUCIÓN. La alineación toma como referencia el borde superior del objeto seleccionado que está más arriba.
- 6. Sin perder la selección, emita FORMATO ESPACIO HORIZONTAL HACER IGUAL.
- 7. Coloque LINE1 por debajo de los seis objetos anteriores.
- 8. Seleccione los ocho objetos superiores y pulse CENTRAR HORIZONTALMENTE.
- 9. Seleccione LINE1 y pulse CENTRAR HORIZONTALMENTE.
- 10. Seleccione SHAPE1 y pulse CENTRAR HORIZONTALMENTE.



Ejercicio 12 b

Mejore EJER12, para hacerlo operativo. Proceda de a poco.

1. Configure las siguientes propiedades:

<u>Objeto</u>	Propiedad	Valor	<u>Objeto</u>	Propiedad	<u>Valor</u>
LABEL1	AUTOSIZE	Verdadero	OPTIONGROUP3	AUTOSIZE	Verdadero
	CAPTION	Grosor		CONTROLSOURCE	Color
SPINNER1	CONTROLSOURCE	Grosor		BUTTONCOUNT	4
	INCREMENT	1		NAME	Color
	KeyboardHighValue	10	OPTION1	AUTOSIZE	Verdadero
	KeyboardLowValue	1		CAPTION	Azul
	SpinnerHighValue	10	OPTION2	AUTOSIZE	Verdadero
	SpinnerLowValue	1		CAPTION	Rojo
	NAME	Grosor	OPTION3	AUTOSIZE	Verdadero
LABEL2	AUTOSIZE	Verdadero		CAPTION	Verde
	CAPTION	Pendiente	OPTION4	AUTOSIZE	Verdadero
OPTIONGROUP1	AUTOSIZE	Verdadero		CAPTION	Amarillo
	CONTROLSOURCE	Pendiente	LABEL4	AUTOSIZE	Verdadero
	NAME	Pendiente		CAPTION	Curvatura
OPTION1	AUTOSIZE	Verdadero	SPINNER2	CONTROLSOURCE	Curvatura
	CAPTION	Positiva		INCREMENT	1
OPTION2	AUTOSIZE	Verdadero		KEYBOARDHIGHVALUE	99
	CAPTION	Negativa		KeyboardLowValue	0
LABEL3	AUTOSIZE	Verdadero		SPINNERHIGHVALUE	99
	CAPTION	Color		SPINNERLOWVALUE	0
LINE1	BORDERCOLOR	0,0,255		NAME	Curvatura
	BorderWidth	1			
	LINESLANT	/			

2. En el evento LOAD del formulario, escriba el siguiente procedimiento:

```
clear memory
public color, grosor, pendiente, curvatura
store 1 to color, grosor, pendiente
store 0 to curvatura
```

3. En el evento INTERACTIVECHANGE de GROSOR, escriba:

```
grosor = this.value
thisform.line1.borderwidth = grosor
```

- 4. Pruebe el formulario, variando el grosor de la línea.
- 5. En el evento CLICK de PENDIENTE, escriba:

```
if pendiente = 1
  thisform.line1.lineslant = '/'
else
  thisform.line1.lineslant = '\'
endif
```

- 6. Pruebe el formulario, variando la pendiente de la línea.
- 7. En el evento CLICK de COLOR, escriba:

```
do case
  case color = 1
    thisform.line1.bordercolor = rgb(0,0,255)
  case color = 2
    thisform.line1.bordercolor = rgb(255,0,0)
  case color = 3
    thisform.line1.bordercolor = rgb(255,255,0)
  case color = 4
    thisform.line1.bordercolor = rgb(0,255,0)
endcase
```

La función *rgb()*, abreviatura de "red, green, blue", devuelve un color resultante de mezclar los colores primarios en las cantidades indicadas como argumentos.

- 8. Pruebe el formulario, variando el color de la línea.
- 9. En el evento INTERACTIVECHANGE de CURVATURA, escriba:

```
curvatura = this.value
thisform.shapel.urvature = curvatura
```

- 10. Pruebe el formulario, variando la curvatura de SHAPE1.
- 11. Dé clic en el fondo del formulario y en la barra DISEÑADOR DE FORMULARIOS active el botón PALETA DE COLORES. Se abre una barra de igual nombre. Vea que esté activo solamente el botón COLOR DE FONDO. Elija el color que le guste para fondo del formulario.



Ejercicio 13

Cree el formulario EJER13. Debe mostrar la tabla ART en un GRID. La columna del campo PRE será la única que se pueda modificar y tendrá un SPINNER en lugar de un TEXTBOX.

- 1. Cree un nuevo formulario.
- 2. Agregue un GRID.
- 3. Llame al generador de GRID.
- 4. En la ficha ELEMENTOS DE LA CUADRÍCULA, elija la tabla ART y seleccione todos los campos.
- 5. En la ficha DISTRIBUCIÓN, seleccione la columna del precio y en TIPO DE CONTROL elija CONTROL NUMÉRICO. Salga del generador con ACEPTAR.
- 6. En la ventana Propiedades haga *Verdadero* el valor de READONLY. Esto se trasmite a todas las columnas, incluso a la del precio.
- 7. Elija la columna del precio y haga Falso el valor de READONLY.
- 8. Ejecute el formulario.

El objeto COLUMN es un contenedor de dos objetos: (1) Un HEADER, para el encabezado de la columna; (2) Otro objeto, por defecto un TEXTBOX, que contiene un campo. El TEXTBOX puede ser reemplazado por otro objeto, siempre que sea adecuado al tipo de datos del campo. En el ejemplo, como PRE es un campo numérico, se puede reemplazar por un SPINNER. Las propiedades de estos objetos se modifican como cuando están fuera del GRID. También puede haber varios tipos de estos objetos dentro de una columna, pero el que está activo en un momento dado se controla por programación.

Ejercicio 14

Cree el formulario EJER14 que use un COMBOBOX dentro de una columna.

- 1. Cree un formulario nuevo y agréguele un GRID con todos los campos de la tabla ENCFAC.
- 2. Active las barras Propiedades y Controles de Formularios.
- 3. En PROPIEDADES seleccione la columna donde está el campo CLI.
- 4. En CONTROLES DE FORMULARIOS pulse el botón COMBOBOX y dé clic en la columna del GRID.
- 5. En PROPIEDADES vea los objetos contenidos en la columna: HEADER1, TEXT1 y COMBO1. Seleccione

Text1.

- 6. Dé clic en el GRID y pulse DELETE. Eliminó TEXT1, como puede ver en PROPIEDADES.
- 7. Seleccione COMBO1 y configúrelo como sigue.

Propiedad	Valor
BOUNDCOLUMN	2
BoundTo	Verdadero
RowSource	Cli.Nom,Cli
RowSourceType	6 – Campos
BorderStyle	0
COLUMNCOUNT	2
COLUMNLINES	Verdadero
COLUMNWIDTHS	200,20
SPECIALEFFECT	1
Style	0

Observe que CONTROLSOURCE ya está definido con el valor Encfac. Cli.

- 8. Agregue la tabla Cli, comandada por el índice Nom.
- 9. Ejecute el formulario.

Ejercicio 15

En EJER14, la columna del cliente muestra un número, no un nombre. Éste sólo aparece para el registro actual al entrar a tal columna. Cree EJER15 para corregir este inconveniente.

- 1. Cree un formulario nuevo.
- 2. En el entorno, introduzca las tablas ENCFAC, CLI y nuevamente CLI.
- 3. Trace una relación de ENCFAC.CLI a CLI.CLI.
- 4. Elija el índice FAC en la propiedad ORDER del cursor ENCFAC, para ver los registros del GRID que creará luego, ordenados por número de factura.
- 5. Elija el índice NOM en la propiedad ORDER del cursor CLI1. Cambie la propiedad ALIAS por *Nombre*. Este cursor servirá para construir un COMBOBOX dentro del GRID.
- 6. Cree el GRID, arrastrando el título del cursor ENCFAC al formulario. Cierre el entorno.
- 7. Haga 5 la propiedad COLUMNCOUNT del GRID1.
- 8. Modifique las siguientes propiedades de COLUMN5: CONTROLSOURCE *Cli.Nom*; COLUMNORDER *4*; WIDTH 200. Esta columna mostrará el nombre de cada cliente, en virtud de la relación.
- 9. Seleccione COLUMN3, introdúzcale un COMBOBOX y elimine TEXT1.
- 10. Seleccione COMBO1 y modifique estas propiedades:

Propiedad	Valor
BoundColumn	2
BoundTo	Verdadero
RowSource	Nombre.Nom,Cli
RowSourceType	6 – Campos
BorderStyle	0
ColumnCount	2
ColumnLines	Falso

Propiedad	Valor
COLUMNWIDTHS	200,0
SPECIALEFFECT	1
Style	0

En COLUMNWIDTHS, el segundo ancho, 0, hace que la segunda columna, la del campo CLI, sea invisible. Quedando una sola columna visible, la del campo NOM, es más elegante que COLUMNLINES sea *Falso*.

11. Abra la ventana Código. En el procedimiento CLICK del objeto COMBO1 escriba lo siguiente:

```
replace encfac.cli with nombre.cli
go record recno() in 'encfac'
thisform.grid1.column4.refresh
```

11. Pruebe el formulario. Al elegir un cliente en el COMBOBOX, el procedimiento refrescará la columna de los nombres.

Ejercicio 16

Cree el formulario EJER16, para que muestre distintos objetos en un marco de páginas.

- 1. En un nuevo formulario, agregue un PAGEFRAME grande desde la barra CONTROLES DE INFORMES.
- PAGEFRAME es un contenedor. Por defecto, tiene dos páginas. Déle clic secundario y elija EDITAR. Aparece el borde de color alrededor de PAGEFRAME1 y se selecciona una página. Dé clic en la otra página, para seleccionarla. Repita esta prueba.
- 3. Abra la ventana PROPIEDADES. En PAGE1, defina CAPTION y NAME con el valor *Clientes*. Dé doble clic en FORECOLOR y elija un color para el texto. En PAGE2, defina CAPTION y NAME con el valor *Vendedores*. Haga lo mismo en FORECOLOR. Seleccione la página CLIENTES. Dé clic en el fondo del formulario.
- 4. En el entorno de datos, agregue las tablas CLI y VEN.
- 5. Arrastre el título del cursor CLI a la página CLIENTES.
- 6. Seleccione el marco de páginas y luego el GRID. Si el GRID estuviera dentro del marco, no lo podría seleccionar sin antes entrar al interior de éste. Quiere decir que no está adentro. Compruebe lo dicho en la ventana PROPIEDADES.
- 7. Para introducir el GRID a la página CLIENTES, selecciónelo y córtelo. Luego, eligiendo EDITAR en el menú contextual sobre el PAGEFRAME, seleccione la página y pegue el GRID. Compruebe la inclusión seleccionando alternativamente una y otra página. También puede ver esto en la ventana PROPIEDADES.
- 8. Teniendo en cuenta lo experimentado, introduzca a la página VENDEDORES otro GRID que muestre los campos de la tabla VEN. Si la página está activa, puede introducirlo directamente en ella.
- 9. Dé clic en el fondo del formulario. Agregue el consabido botón SALIR. Póngalo sobre el PAGEFRAME, no dentro de él. Si estuviera adentro, debería estar en una página o en la otra. Para terminar la ejecución, habría que activar la página donde estuviera. Una solución podría ser colocar dos botones SALIR, uno en cada página. Por economía de diseño, conviene que sea uno solo y esté afuera.

Forr	n1						
		Clientes		Vend	ledores		
	Cli	Nom	Dom		Tel		1
						_	
						▼	
					Salir		
Forr	n1						
		Clientes		Vend	ledores		
	Ven	Nom	Com				
				1			

Ejercicio 17 a

Cree el formulario EJER17 con los datos de los clientes en una página y sus compras en otra.

- 1. Cree un nuevo formulario.
- 2. Agréguele un PAGEFRAME. Modifique las propiedades CAPTION y NAME de las páginas, con los valores *Clientes* y *Compras*, respectivamente.

Salir

- 3. Genere un GRID en la página CLIENTES que muestre los campos de la tabla CLI.
- 4. Genere un GRID en la página COMPRAS con los campos de la tabla ENCFAC. En la ficha RELACIÓN, introduzca *Cli. Cli* en CAMPO CLAVE DE LA TABLA PRIMARIA: esto va a dar el nombre de la tabla a la propiedad LINKMASTER y el del campo a RELATIONALEXPR. Seleccione el índice CLI en ÍNDICE RELACIONADO DE LA TABLA SECUNDARIA, que será el valor de CHILDORDER. Las tres propiedades van a producir una relación implícita de CLI.CLI a ENCFAC.CLI, subordinando el GRID al registro actual de la tabla CLI.
- 5. Genere otro GRID en la página COMPRAS, a la derecha del anterior, con los campos de la tabla LINFAC. En la ficha RELACIÓN, introduzca *Encfac. Fac* en CAMPO CLAVE DE LA TABLA PRIMARIA y el índice FAC en ÍNDICE RELACIONADO DE LA TABLA SECUNDARIA. GRID2 estará subordinado al registro actual de GRID1.
- 6. Pruebe el formulario. Elija un cliente en la página CLIENTES y recuerde el número del campo CLI. Active la página COMPRAS y compruebe que el GRID izquierdo únicamente muestra facturas cuyo campo CLI tiene el mismo número recordado. Elija cualquier registro y vea que el GRID derecho únicamente muestra registros cuyo campo FAC es igual al del GRID izquierdo.

	Clientes		Compras
Cliente	Nombre	Domicilio	Teléfono

	Clientes					Cor	npras	
L	Fac	Fec	Cli	Ven	Fac	Art	Can	Pre

Ejercicio 17 b

Mejore EJER17. Como en el ejercicio anterior comprobó la correcta subordinación de cada GRID, es innecesario y poco elegante que el de la izquierda muestre el campo CLI y el de la derecha el campo FAC. Agregue el producto de la cantidad por el precio y el nombre de cada artículo en el GRID derecho.

- 1. Seleccione la tercera columna del GRID izquierdo de la página COMPRAS y pulse DELETE.
- 2. Seleccione el GRID derecho y cambie la propiedad COLUMNCOUNT por 5.
- 3. Seleccione la primera columna y arrástrela tras la segunda. Cambie CONTROLSOURCE por *Art. Nom* y WIDTH por 200.
- 4. Seleccione el HEADER de la segunda columna y cambie CAPTION por Descripción.
- 5. Seleccione la última columna y escriba en CONTROLSOURCE Linfac.Can * Linfac.Pre.
- 6. Seleccione el HEADER de esta columna y cambie CAPTION por Total.
- 7. Modifique los anchos de cada GRID para ver exactamente todas las columnas.
- 8. Abra el entorno y agregue la tabla ART. Trace una relación de LINFAC.ART a ART.ART.
- 9. Pruebe el formulario.

Fac Fec Ven Art Descripción Can Descripción Can	m1 Cliente	es		Compra	as	
	Fac Fec V	Ven Art	Descripción	Can	Pre	Total

Ejercicio 17 c

Mejore EJER17, incorporando al pie del GRID derecho de la página COMPRAS la suma de la columna cantidad por precio.

- 1. En la página COMPRAS, al pie del GRID derecho, introduzca un TEXTBOX, con CONTROLSOURCE y NAME *Monto*.
- 2. En el evento LOAD del formulario, escriba:

```
clear memory
public monto
monto = 0
```

3. En el evento AfterRowColChange del GRID izquierdo, escriba:

```
LPARAMETERS nColIndex
select linfac
set key to encfac.fac
calculate sum(can * pre) to monto
set key to
thisform.pageframe1.compras.monto.refresh
```

4. Pruebe el formulario.



Ejercicio 17 d

Mejore EJER17. El GRID derecho de la página COMPRAS está subordinado a la tabla ENCFAC por el índice FAC. La tabla LINFAC tiene un índice más apropiado, FACART, que hará ver mejor los registros, pues, para una factura, aparecerán sus registros en orden creciente de código de artículo. Para corregir la subordinación, hay que hacerlo en el entorno de datos, en la relación. Haga lo siguiente:

- 1. Escriba *Facart* en la propiedad CHILDORDER. Es el nombre de un índice construido por la expresión *str(Fac, 8)* + *Art*.
- 2. Escriba *str(Encfac. Fac, 8)* en la propiedad RELATIONALEXPR. Esto hará que la tabla ENCFAC use su campo FAC para filtrar, mediante el índice de LINFAC, los registros a mostrar en el GRID.

La función *str()* es imprescindible. Si RELATIONALEXPR fuera *Linfac. Fac*, de tipo numérico, el uso del índice no sería exitoso, porque está construido por una expresión de caracteres.

VFP sabe cuál es la tabla subordinante, porque está definida en la propiedad PARENTALIAS. Como RELATIONALEXPR es siempre un campo de la tabla subordinante, es innecesario usar a ésta como prefijo del campo. Hemos escrito *str*(*Encfac.Fac, 8*) para enfatizar que el campo FAC es de ENCFAC. Basta escribir *str*(*Fac, 8*).

La expresión de RELATIONALEXPR coincide parcialmente con la estructura de la expresión del índice FACART. No hay ningún inconveniente, siempre que la exactitud de las comparaciones no esté activa.

 Compruebe en HERRAMIENTAS – OPCIONES – DATOS que SET EXACT ACTIVADO no tenga una tilde. Si la tiene, sáquesela. Esto hará que las comparaciones den como iguales cosas que comiencen con el mismo valor. Por ejemplo, "Pesca" se considerará igual a "Pescado". En nuestro problema, serán iguales los siguientes valores:

	Encfac			Linfac
Fac	RELATIONALEXPR	Fac	Art	Expresión del índice
25	•••••25	25	A10	•••••25A10
			F24	•••••25F24
2673	····2673	2673	B14	••••2673B14

Los "puntos altos" simbolizan espacios en blanco. Hasta donde se agota RELATIONALEXPR, coincide con la expresión del índice.

4. Ejecute el formulario. Los artículos de cada factura estarán ordenados por código.

Ejercicio 18

Cree el formulario EJER18 con un cuadro de edición.

- 1. Cree un nuevo formulario.
- 2. Agréguele un cuadro de texto grande. Un EDITBOX es el objeto adecuado para campos memo; también lo es para variables de caracteres cuyo contenido sea grande.

Los campos memo se definen en la estructura de una tabla y ocupan 4 posiciones. Si en un registro cualquiera se quiere llenar un campo memo, en éste se guarda una referencia y se abre una ventana de edición para escribir el contenido, que puede tener cualquier tamaño. Este contenido se guarda con la misma referencia en un archivo asociado, cuyo nombre es igual al de la tabla, con extensión FPT. De este modo, si hay varios campos memo llenos, el contenido de cada uno se recupera del archivo asociado en base a la referencia correspondiente.

- Escriba Texto en la propiedad CONTROLSOURCE del EDITBOX. Es el nombre de una variable de caracteres, que usaremos de prueba.
- 4. En el evento LOAD del formulario, escriba:

```
clear memory
public texto
texto = replicate('Esto va a ser largo. ',100)
```

5. Pruebe el formulario.



Ejercicio 19 a

Cree el formulario EJER19 para usar tres casillas de verificación que controlen lo que será visualizado en un GRID.

- 1. Cree un nuevo formulario.
- 2. En el entorno, agregue la tabla CLI.
- 3. Arrastre el título del cursor para crear un GRID en el formulario y cierre el entorno.
- 4. Agregue tres casillas de verificación. El objeto CHECKBOX maneja un campo de datos o una variable de tipo lógico.
- 5. Configure las siguientes propiedades:

Propiedad	<u>Check1</u>	<u>Check2</u>	Check3
ALIGNMENT	Izquierda	Derecha	Izquierda
CAPTION	Ver Nombre	Ver Domicilio	Ver Teléfono
CONTROLSOURCE	Nombre	Domicilio	Teléfono
NAME	Nombre	Domicilio	Teléfono

6. En el evento LOAD del formulario, escriba:

```
clear memory
public nombre, domicilio, teléfono
store .t. to nombre, domicilio, teléfono
```

7. En el evento CLICK del objeto NOMBRE, escriba:

```
if nombre
   thisform.grid1.column2.visible = .t.
else
   thisform.grid1.column2.visible = .f.
endif
```

8. En el evento CLICK del objeto DOMICILIO, escriba:

```
if domicilio
   thisform.grid1.column3.visible = .t.
else
   thisform.grid1.column3.visible = .f.
endif
```

9. En el evento CLICK del objeto TELÉFONO, escriba:

```
if teléfono
   thisform.grid1.column4.visible = .t.
else
   thisform.grid1.column4.visible = .f.
endif
```

10. Pruebe el formulario. Como la alineación de los botones es igual en dos CHECKBOX y diferente en el restante, iguale la propiedad ALIGNMENT.

Ejercicio 19 b

El formulario EJER19, al mandar que se oculte una columna, la muestra en blanco; pero queda un hueco. Corríjalo para que este espacio desaparezca.

1. Corrija el evento LOAD del formulario:

```
clear memory
public nombre, domicilio, teléfono
public anombre, adomicilio, ateléfono
store .t. to nombre, domicilio, teléfono
```

Las variables anombre, adomicilio y ateléfono guardarán el ancho de las columnas 2, 3 y 4.

2. En el evento ACTIVATE del formulario, escriba:

```
anombre = thisform.grid1.column2.width
adomicilio = thisform.grid1.column3.width
ateléfono = thisform.grid1.column4.width
```

3. Corrija el evento CLICK del objeto NOMBRE:

```
if nombre
  thisform.grid1.column2.width = anombre
  thisform.grid1.column2.visible = .t.
else
  thisform.grid1.column2.width = 0
  thisform.grid1.column2.visible = .f.
endif
```

4. Corrija el evento CLICK del objeto DOMICILIO:

```
if domicilio
  thisform.grid1.column3.width = adomicilio
  thisform.grid1.column3.visible = .t.
else
  thisform.grid1.column3.width = 0
  thisform.grid1.column3.visible = .f.
endif
```

5. Corrija el evento CLICK del objeto TELÉFONO:

```
if teléfono
```

```
thisform.grid1.column4.width = ateléfono
thisform.grid1.column4.visible = .t.
else
thisform.grid1.column4.width = 0
thisform.grid1.column4.visible = .t.
endif
```

Ejercicio 19 c

El GRID de EJER19 tiene ancho fijo, aunque no se vean una o varias columnas. Para que el ancho varíe dinámicamente, hay que retocar los procedimientos.

1. Corrija el evento LOAD del Formulario:

```
clear memory
public nombre, domicilio, teléfono
public anombre, adomicilio, ateléfono, agrid
store .t. to nombre, domicilio, teléfono
```

2. En el evento ACTIVATE del formulario, escriba:

```
anombre = thisform.grid1.column2.width
adomicilio = thisform.grid1.column3.width
ateléfono = thisform.grid1.column4.width
agrid = thisform.grid1.width
```

3. Corrija el evento CLICK del objeto NOMBRE:

```
if nombre
  agrid = agrid + anombre
  thisform.grid1.width = agrid
  thisform.grid1.column2.width = anombre
  thisform.grid1.column2.visible = .t.
else
  agrid = agrid - anombre
  thisform.grid1.width = agrid
  thisform.grid1.column2.width = 0
  thisform.grid1.column2.visible = .f.
endif
```

4. Corrija el evento CLICK del objeto DOMICILIO:

```
if domicilio
agrid = agrid + adomicilio
thisform.grid1.width = agrid
thisform.grid1.column3.width = adomicilio
thisform.grid1.column3.visible = .t.
else
agrid = agrid - adomicilio
thisform.grid1.width = agrid
thisform.grid1.column3.width = 0
thisform.grid1.column3.visible = .f.
endif
```

5. Corrija el evento CLICK del objeto TELÉFONO:

```
if teléfono
  agrid = agrid + ateléfono
  thisform.grid1.width = agrid
```

```
thisform.grid1.column4.width = ateléfono
thisform.grid1.column4.visible = .t.
else
agrid = agrid - ateléfono
thisform.grid1.width = agrid
thisform.grid1.column4.width = 0
thisform.grid1.column4.visible = .f.
endif
```

6. Pruebe el formulario.

Ejercicio 20

Cree el formulario EJER20 para trabajar con un cuadro de lista. Al elegir una opción en este cuadro, se verá el efecto en un GRID subordinado. Un LISTBOX es igual en funcionamiento a un COMBOBOX. Se puede configurar ya sea con un generador, ya sea introduciendo a mano el valor de sus propiedades.

- 1. Cree un nuevo formulario.
- 2. En el entorno, agregue las tablas ART y LINFAC.
- 3. Arrastre el título del cursor LINFAC al formulario, para crear un GRID.
- 4. Agregue un cuadro de lista al formulario.
- 5. Configure las siguientes propiedades de LIST1:

Propiedad	Valor
BOUNDCOLUMN	2
RowSource	Art.Nom,Art
RowSourceType	6 – Campos
COLUMNCOUNT	2
COLUMNLINES	Verdadero
ColumnWidths	200,20

6. Configure las siguientes propiedades de GRID1:

Propiedad	Valor
CHILDORDER	Art
PARENTALIAS	Art
RELATIONALEXPR	Art
CHILDORDER PARENTALIAS RELATIONALEXPR	Art Art Art

Guarde el formulario y ejecútelo. Elija distintas opciones en el LISTBOX y observe el efecto en el GRID.

Capítulo 13. Ejercicios con informes

Cuando se diseña un informe, en cualquier momento se puede obtener una vista preliminar de cómo se vería impreso. Aunque esta modalidad difiere en algunos detalles de la impresión real, para no inutilizar papel la usaremos continuamente. Entonces, cuando le digamos imprimir, querremos decir que produzca una vista previa.

La vista preliminar es tan permisiva que ni siquiera necesita que el informe se haya guardado. Sin embargo, por precaución, conviene que dé un nombre al informe a poco andar y vaya guardando a medida que avanza en el diseño.

Ejercicio 1 a

Cree el informe EJER1, para que imprima los campos de la tabla ENCFAC.

1. Para crear un informe nuevo, hay varios métodos, algunos de los cuales son:

- Seleccionar la ficha DOCUMENTOS en el ADMINISTRADOR DE PROYECTOS, situarse en el ámbito de informes y pulsar NUEVO.
- Emitir Archivo NUEVO INFORME NUEVO ARCHIVO.
- Emitir modify report ejemplo1 en la ventana COMANDOS.

Cualquiera de estos métodos abre el DISEÑADOR DE INFORMES. En los dos primeros, el nombre del archivo se da recién al guardarlo por primera vez. En el último, el informe es creado con su nombre. La forma más conveniente es la primera, porque registra el nuevo informe en el proyecto, sin tener que agregarlo especialmente. El diseñador se abre con las bandas ENCABEZADO DE PÁGINA, DETALLE y PIE DE PÁGINA, con una regla horizontal y otra vertical. En estas bandas se pueden colocar o no objetos. Aún si están vacías, van a ocupar un espacio en la impresión igual a su altura. Por ello, esta altura se puede aumentar o reducir para aprovechar el papel, incluso anulándola.

Diseñador de Informes
▲ Encabezado de página
▲ Detalle
▲ Pie de página

- 2. Si no está presente la barra DISEÑADOR DE INFORMES, hágala visible con VER BARRA DE HERRAMIENTAS DISEÑADOR DE INFORMES ACEPTAR.
- 3. Pulse el botón ENTORNO DE DATOS.
- 4. Agregue la tabla ENCFAC.

5. Para arrastrar campos de la tabla ENCFAC al informe, hay dos procedimientos: (1) Arrastrar un campo a una banda: esto produce un objeto CAMPO de informe; (2) Arrastrar el título del cursor o la línea que dice *Campos* a una banda: esto produce un objeto CAMPO por cada campo de la tabla, uno bajo el otro, haciendo más alta la banda si hace falta. En general, para listar los registros de la tabla, se usa la banda DETALLE, porque en ella se define el formato de impresión común a aplicar a cada registro. Arrastre todos los campos de una vez, usando el segundo procedimiento.

Diseñador de Informes
▲ Encabezado de página
fac
fec
cli
ven
▲ Detalle
▲ Pie de página

- 6. Dé clic secundario en el informe, donde no hay CAMPOS, y elija VISTA PRELIMINAR. También puede hacerlo con VER – VISTA PREVIA. Examine el modelo de cómo se vería el informe si lo imprime. Si no está presente la barra VISTA PRELIMINAR, actívela con VER – BARRA DE HERRAMIENTAS. Estudie los botones, colocando un momento el puntero del mouse sobre cada uno. Pulse el botón CERRAR VISTA PREVIA o pulse ESCAPE.
- 7. El informe "impreso" tiene muchas páginas, en parte porque hay muchos registros, en parte porque cada registro muestra sus campos uno bajo el otro. Lo primero es inevitable si quiere ver el contenido total de la tabla. Lo segundo se puede mejorar, disponiendo los CAMPOS horizontalmente. Arrástrelos uno por uno para que estén más o menos al mismo nivel, o selecciónelos y use las teclas de flecha. Selecciónelos a todos, describiendo un rectángulo que los incluya o pulsando SHIFT y dando clic en cada uno. Pulse el botón DISTRIBUCIÓN de la barra DISEÑADOR DE INFORMES. Pulse el botón ALINEAR LOS BORDES SUPERIORES. Llévelos en conjunto cerca del borde superior de la banda. Arrastre la barra gris que dice *Detalle* hacia arriba, hasta que tope con los CAMPOS. La separación que habrá entre cada registro impreso será el espacio que ha quedado entre el borde superior de la banda y los CAMPOS.

Dise	eñado	r de Informes						
	Encab	ezado de págir	na					
		fac	1	fec	1	cli	ven	
		100		100			VOIT	
	Detall	e						
	Pie de	e página						

- 8. Haga una presentación preliminar del informe.
- 9. Guarde el informe, dándole nombre si hace falta.

Ejercicio 1 b

Mejore EJER1, para que las páginas tengan un encabezado y los datos listados lleven cabeceras.

- 1. Abra EJER1.
- 2. Arrastre la barra gris que dice Encabezado de página hacia abajo, para tener espacio donde trabajar.
- 3. Pulse el botón CONTROLES DE INFORMES, que abrirá la barra de igual nombre.
- 4. En la barra CONTROLES DE INFORMES, pulse el botón ETIQUETA y dé clic en la banda ENCABEZADO DE PÁGINA. Escriba *Listado de facturas, ordenado por número*. Luego dé clic fuera de este texto.
- 5. Debajo del texto anterior, sobre el CAMPO que dice *fac*, introduzca una ETIQUETA con el texto *Factura*.
- 6. Introduzca ETIQUETAS para los restantes CAMPOS, con los textos Fecha, Cliente y Vendedor.
- 7. Ponga en el mismo nivel las cuatro ETIQUETAS inferiores y alinee cada una con su respectivo CAMPO por los bordes izquierdos.
- 8. Arrastre la barra Encabezado de página hacia arriba, hasta que tope con los encabezados.

Diseñador	de Informes								
Listado de facturas, ordenado por número									
F	Factura Fecha Cliente Vendedor								
▲ Encabe	▲ Encabezado de página								
f	fac	fec	cli	ven					
▲ Detalle	▲ Detalle								
▲ Pie de p	▲ Pie de página								

- 9. Pruebe el informe.

El mismo problema se da con los CAMPOS de cliente y vendedor. Puede usar la misma solución anterior. También puede alinear los centros verticalmente de cada pareja, consistente en calcular el medio vertical entre el borde más a la izquierda y borde más a la derecha de los objetos y luego centrar horizontalmente cada objeto en ese medio. Esta nueva solución va a funcionar bien para el vendedor.

11. Pruebe el informe.

Diseñador de Informes									
	Listado de facturas, ordenado por número								
	Factura Fecha Cliente Vendedor								
▲ Encab	▲ Encabezado de página								
	fac	fec		cli			ven		
▲ Detall	▲ Detalle								
A Pie de	▲ Pie de página								

Ejercicio 1 c

Mejore EJER1, para que las facturas estén ordenadas y el encabezado se destaque.

- Como la primera línea del encabezado de página dice que el impreso está ordenado por número de factura, la tabla debe seguir ese orden. En el entorno de datos, configure la propiedad ORDER del cursor ENCFAC con el índice FAC.
- 2. Seleccione las ETIQUETAS del encabezado de página. Llame al diálogo FUENTE con FORMATO FUENTE. En ESTILO DE FUENTE elija *Negrita* y en TAMAÑO un número mayor al actual. Centre la Etiqueta superior horizontalmente.
- 3. Haga un poco más alta la banda ENCABEZADO DE PÁGINA. En la barra CONTROLES DE INFORMES pulse el botón LÍNEA y arrastre horizontalmente debajo de los encabezados de campo. Si hace falta cambiar el largo de este nuevo objeto, puede hacerlo arrastrando su extremo derecho o pulsando Shift y movien-do las flechas horizontales. Luego reduzca el alto de la banda, hasta que tope con la LÍNEA.
- 4. Pulse el botón CAMPO en la barra CONTROLES DE INFORMES y dé clic en la banda PIE DE PÁGINA. Se abre el diálogo EXPRESIÓN DE INFORME. En EXPRESIÓN escriba _pageno o llame al GENERADOR DE EXPRESIONES con el botón ... y elija la variable de igual nombre. Esta variable del sistema devuelve el número de página actual del informe. Como no es un campo de la tabla Encfac, no se puede obtener de ella por arrastre. No queda sino crear un CAMPO y definir su expresión. Centre el CAMPO horizon-talmente en la banda.

Diseñador de Informes								
Listado de facturas, ordenado por número								
	<u>Factura Fecha Cliente Vendedor</u>							
▲ Encabezad	▲ Encabezado de página							
fac		fec		cli			ven	
▲ Detalle								
_pageno								
A Pie de pág	ina							

5. Pruebe el informe, guárdelo y cierre el diseñador.

Ejercicio 2 a

Cree el informe EJER2. Debe imprimir los tres primeros campos de la tabla CLI.

1. Cree un nuevo informe.

- 2. En el encabezado de página coloque una ETIQUETA con el texto *Prueba de objetos que se estiran, flotan, etc.*
- 3. Agréguele un CAMPO desde la barra CONTROLES DE INFORMES. Defina su EXPRESIÓN como Cli. Cli.
- 4. Agregue otros dos CAMPOS, con EXPRESIÓN Cli.Nom y Cli.Dom.
- 5. En el entorno, agregue la tabla CLI.
- 6. Disponga los CAMPOS horizontalmente, a espacios regulares y a una misma altura.
- 7. Pruebe el informe. Puede ver que hay nombres truncados, porque el Campo es demasiado estrecho.

Diseñador de Informes								
Prueba de objetos que se estiran, flotan, etc.								
▲ Encabezado de página								
	cli		nom		dom			
▲ Detalle								
A Pie de pa	ágina							

- 8. Dé doble clic en el CAMPO Nom. Active la casilla AJUSTAR AL CONTENIDO DEL TEXTO.
- 9. Pruebe el informe. El nombre de los clientes, ya que el ancho del CAMPO que los muestra no se puede modificar en tiempo de ejecución, se escribe en más de una línea.
- 10. Aumente la altura de la banda DETALLE. Coloque debajo del CAMPO *Nom* el CAMPO *Dom*. Reduzca la altura de la banda al mínimo.
- 11. Pruebe el informe. El nombre del cliente, cuando necesita varias líneas, se superpone al domicilio.

Diseñador de Informes						
Prueba de objetos que se estiran, flotan, etc.						
▲ Encabezado de página						
cli	nom					
	dom					
▲ Detalle						
▲ Pie de página						

- 12. Dé doble clic en el CAMPO Dom. En Posición del campo, active el botón FLOTANTE.
- 13. Pruebe el informe. Ahora sí: Cuando el CAMPO *Nom* necesita más de una línea, el CAMPO *Dom* "flota", es decir, se desplaza hacia abajo, conservando la misma distancia que hay entre ambos objetos en el diseño del informe.

Ejercicio 2 b

Modifique EJER2 para que los Campos estén dispuestos verticalmente y tengan Etiquetas aclaratorias.

- 1. Introduzca un CAMPO cuya expresión sea Cli. Tel.
- 2. Disponga los Campos verticalmente, en el orden Cli, Nom, Dom, Tel.
- 3. Agregue a la izquierda una columna de ETIQUETAS, con los textos Cliente, Nombre, Domicilio, Teléfono.
- 4. Alinee las ETIQUETAS por los bordes izquierdos y con FORMATO ESPACIADO vertical elija HACER IGUAL.
- 5. Alinee por los bordes superiores cada ETIQUETA con su correspondiente CAMPO.
- 6. Seleccione los campos y alinéelos por los bordes izquierdos.
- 7. Pruebe el informe. El teléfono está invadido por el domicilio y los datos se han desfasado con respecto a las ETIQUETAS.

Diseñador de Informes							
Prueba de objetos que se estiran, flotan, etc.							
▲ Encabezado de página							
Cliente	cli						
Nombre	nom						
Domicilio	dom						
Teléfono	tel						
▲ Detalle							
▲ Pie de página							

- 8. Haga flotante el CAMPO *Tel*. Elija posición BORDE INFERIOR DE LA BANDA para las ETIQUETAS *Domicilio* y *Teléfono*.
- 9. Pruebe el informe. Trabaja bien. Si la banda se hace más alta porque el CAMPO Nom se escribe en varias líneas, las dos últimas ETIQUETAS se escriben correctamente, pues su posición con relación al borde inferior de la banda se mantiene fija.

Ejercicio 2 c

Modifique EJER2 para que el número de cliente impreso esté alineado por la izquierda y todos los objetos estén dentro de un RECTÁNGULO.

- Cambie la expresión del CAMPO Cli por la expresión alltrim(str(cli.cli)). Str() va a convertir el número de cliente a caracteres, con espacios iniciales. Alltrim() va a eliminar los espacios iniciales. Entre con el botón de tres puntos al diálogo FORMATO. Vea que está activado el botón NUMÉRICO. No es un sinsentido, como parece: un tipo de datos determinado puede tener un tratamiento de otro tipo en este diálogo, a fin de usar códigos de formato convenientes. En este caso, puede activar ALINEAR A LA IZQUIERDA. También puede activar CARÁCTER y activar ALINEAR A LA IZQUIERDA.
- 2. Trace un objeto RECTÁNGULO que encierre a todos los demás objetos.
- 3. Pruebe el informe. El tamaño programado del RECTÁNGULO queda chico, porque el CAMPO *Nom* ha ocupado más de una línea de texto.
- 4. Hay dos soluciones posibles para hacer que el RECTÁNGULO crezca en tiempo de ejecución. Veamos la
primera. Dé doble clic en el RECTÁNGULO o dé clic secundario sobre él y elija PROPIEDADES. Aparece el diálogo RECTÁNGULO / LÍNEA. En ALARGAR HACIA ABAJO active el botón ALARGAR CON RELACIÓN AL ALTO DE BANDA.

5. Pruebe el informe. Funciona bien.

Diseñador de Informes								
Prueba de objetos que se estiran, flotan, etc.								
Encabezado de pág	▲ Encabezado de página							
	Cliente	cli]					
	Nombre	nom						
	Domicilio	dom						
	Teléfono	tel						
▲ Detalle								
A Pie de página								

- 6. Veamos la segunda solución. En el diálogo RECTÁNGULO / LÍNEA, active el botón ALARGAR CON RELACIÓN AL OBJETO MÁS ALTO DEL GRUPO. Luego seleccione los objetos rodeados por el RECTÁNGULO, incluyendo a éste, y en el menú FORMATO elija AGRUPAR. Este agrupamiento de objetos hace que los objetos seleccionados sean tratados como una unidad al moverlos. No lo confunda con el agrupamiento de datos que aparece en el menú INFORME, consistente en agrupar registros según algún criterio.
- 7. Pruebe el informe. No funciona bien. Ello se debe a que entre los objetos que forman el grupo hay dos con posición relativa al borde inferior de la banda. Esto impide que funcionen bien dentro del grupo.
- 8. Desagrupe los objetos con FORMATO DESAGRUPAR. Seleccione el CAMPO *Domicilio* y déle posición flotante. Haga lo mismo con el CAMPO TELÉFONO. Vuelva a agrupar la totalidad de los objetos.
- 9. Pruebe el informe. Ahora funciona bien.

Ejercicio 2 d

Modifique EJER2 para que tenga dos columnas.

- 1. Seleccione todos los objetos y córralos bien a la izquierda.
- 2. En el menú ARCHIVO elija PREPARAR PÁGINA. Se abre el diálogo PREPARAR PÁGINA. En COLUMNAS, introduzca un 2 en NÚMERO. Pulse ACEPTAR. Aparecen dos nuevas bandas: ENCABEZADO DE COLUMNA y PIE DE COLUMNA. Por columna hay que entender el ancho ocupado por los objetos en la banda DETALLE. Si la página puede dar cabida dos o más veces a este ancho, se puede establecer que el informe tenga dos, tres, etc. columnas. Observe que las dos nuevas bandas y la banda DETALLE son la mitad de las demás, ya que van a entrar dos veces en la página.
- 3. Seleccione todos los objetos y céntrelos en la banda DETALLE.
- 4. Pruebe el informe.

Ejercicios con informes y entornos de datos - Edmundo A. Cáceres - 2011

Diseñador de Informes							
Prueba de objetos que se estiran, flotan, etc.							
▲ Encabezado de pág	ina						
▲ Encabezado de colu	ımna						
Cliente	cli						
Nombre	nom						
Domicilio	dom						
Teléfono	tel						
▲ Detalle							
▲ Pie de columna							
Pie de página	▲ Pie de página						

Ejercicio 3 a

Cree el informe EJER3. Debe listar los datos de la tabla LINFAC, agregando el producto de la cantidad por el precio.

- 1. Cree un nuevo informe.
- 2. En el entorno, agregue la tabla LINFAC, gobernada por el índice FACART.
- 3. Arrastre uno a uno los campos FAC, ART, CAN, PRE, colocándolos en la banda DETALLE más o menos a la misma altura.
- 4. Agregue, a la izquierda de los otros, un CAMPO cuya expresión sea *Linfac. Can* * *Linfac. Pre*. Este CAMPO no corresponde a ningún campo de la tabla, pero se puede obtener a partir de ellos.
- 5. En el encabezado de página, coloque una ETIQUETA sobre cada CAMPO, con los textos *Factura*, *Artículo*, *Cantidad*, *Precio*, *Importe*, en negritas.
- 6. Use la barra DISTRIBUCIÓN y el menú FORMATO para refinar la disposición de los objetos.
- 7. Pruebe el informe. El número de factura se repite. No queda bien.

Diseñador de Informes								
	Factura		Artículo		Cantidad		Precio	Importe
	Encabezad	o de	e página					
	fac		art		can		pre	linfac. can*
	Detalle							
	Pie de pági	na						

- 8. Llame al diálogo EXPRESIÓN DE INFORME del CAMPO *Fac.* Pulse el botón IMPRIMIR–CONDICIONES. En el diálogo de igual nombre, active el botón No en IMPRIMIR VALORES REPETIDOS.
- 9. Pruebe el informe. Ahora está mejor. Avance de página en página. Vea que al comienzo de algunas páginas, a veces el número de factura no aparece. Para saber de qué factura se trata, deberá volver a la página anterior.

- 10. Volver a la página anterior es incómodo. Para evitar esta molestia, podría hacer que no se impriman valores repetidos, salvo si se comienza una nueva página. En IMPRIMIR–CONDICIONES del campo FAC, en IMPRIMIR TAMBIÉN active el botón EN PRIMERA PÁGINA COMPLETA DE NUEVA PÁGINA / COLUMNA.
- 11. Pruebe el informe. Avance página a página. La primera línea siempre imprime el número de factura.

Ejercicio 3 b

Mejore EJER3, haciendo que imprima el nombre de artículo, en lugar del código.

- 1. En el entorno de datos, agregue la tabla ART.
- 2. Trace una relación desde LINFAC.ART a ART.ART.
- 3. En el informe, cambie la expresión del CAMPO Art por Art.Nom.
- 4. Seleccione los CAMPOS *Can*, *Pre* y *Linfac*. *Can* * *Linfac*. *Pre* y muévalos hacia la derecha, para dar cabida al nombre de artículo.
- 5. Seleccione el CAMPO Nom y agrande su ancho, para que quepa el nombre de artículo.
- 6. Retoque la disposición de los objetos.
- 7. Pruebe el informe.

Ejercicio 4 a

En EJER3 hemos obtenido el producto de la cantidad por el precio de cada artículo vendido en una factura, pero no la suma de esos productos para obtener el total de la factura. Para lograr esto, hay que agrupar la tabla LINFAC por el campo FAC. Abra EJER3 si hace falta y guárdelo como EJER4. Trabaje con EJER4.

- 1. En el menú INFORME elija la opción AGRUPAR DATOS. Aparece un diálogo de igual nombre.
- 2. En EXPRESIONES DE AGRUPACIÓN introduzca *Linfac.Fac*, a mano o con el generador de expresiones. Pulse ACEPTAR. En el diseñador aparecen dos nuevas bandas: ENCABEZADO DE GRUPO 1 y PIE DE GRUPO 1, con altura 0. Un grupo es un conjunto de registros con igual valor en el campo FAC. Así, hay un grupo de registros para la factura 1, otro grupo para la factura 2, otro para la factura 3, etc. Si hubiera 1000 números de facturas distintas, habría 1000 grupos; si hubiera 5000, habría 5000 grupos, etc. AGRUPAR POR hace que el informe esté atento a los cambios de valor dentro del campo FAC, para saber cuándo empieza y cuándo termina cada grupo. Esto exige, obviamente, que los registros estén ordenados de acuerdo al agrupamiento. Al empezar un grupo, se puede usar la banda ENCABEZADO DE GRUPO para datos que lo identifiquen. Al terminar un grupo, porque empieza otro, se puede usar la banda PIE DE GRUPO para datos resultantes del proceso del grupo, como sumas, promedios, etc.
- 3. Abra la banda PIE DE GRUPO1, arrastrando la barra gris hacia abajo.
- 4. Agréguele un CAMPO. En el diálogo EXPRESIÓN DE CAMPO escriba la expresión *Linfac. Can* * *Linfac. Pre*. Pulse el botón CÁLCULOS. En el diálogo CALCULAR CAMPOS active el botón SUMA.
- 5. Pruebe el informe. El producto entre cantidad y precio muestra cuatro decimales. Lo mismo ocurre con la suma de ese producto. En *Expresión de informe* del CAMPO donde está el producto, escriba 999,999.99 en FORMATO. Haga lo mismo para el CAMPO de la sumatoria.
- 6. Pruebe el informe. Bastante mejor. Pero el importe total de cada factura no se destaca. Seleccione el CAMPO correspondiente y póngalo en negritas.
- 7. Pruebe el informe.

Dis	eñador de li	nforr	nes				
	Factura	ļ	Artículo		Cantidad	Precio	Importe
	Encabezad	o de	página				
	Encabezad	o de	grupo 1:	fac			
	fac		art		can	pre	linfac.can*
	Detalle						
							linfac.can
	Pie de grup	o 1: '	fac				
	Pie de pági	na					

Ejercicio 4 b

Mejore EJER4 para que muestre la fecha y el nombre del cliente.

- 1. Agregue la tabla ENCFAC, que tiene la fecha y el código de cliente. Trace una relación de LINFAC.FAC a ENCFAC.FAC por el campo común FAC.
- 2. Agregue la tabla CLI, que tiene el código y el nombre de cliente. Trace una relación desde ENCFAC.CLI a CLI.CLI por el campo CLI.
- Abra la banda ENCABEZADO DE GRUPO. Desde el entorno, arrastre los campos ENCFAC. FEC y CLI. NOM. Ponga en negritas la fuente de estos nuevos CAMPOS, dispóngalos adecuadamente y reduzca el alto de la banda.
- 4. Pruebe el informe. El número de factura estaría mejor en el encabezado de grupo.
- 5. Mueva el CAMPO *Fac* a la banda ENCABEZADO DE GRUPO. Elimine la ETIQUETA *Factura*. Recomponga la disposición de los objetos.
- 6. Pruebe el informe. Hay todavía que arreglar detalles, pero los dejaremos pendientes.

Diseñador de Inf	formes			
	Artículo	Cantidad	Precio	Importe
▲ Encabezado	de página			
fac	fec	nom		
▲ Encabezado	de grupo 1: fac	•		
	art	can	pre	linfac.can*
▲ Detalle				·
				linfac.can
Pie de grupo	1: fac			
▲ Pie de página	а			

Ejercicio 4 c

Mejore EJER4, de modo que el listado semeje facturas reales.

- 1. Haga más alta la banda ENCABEZADO DE GRUPO. Coloque tres ETIQUETAS, con los textos Factura, Fecha y Cliente, en negritas.
- Ponga las ETIQUETAS una bajo otra, a la izquierda, y cada CAMPO a la derecha de la ETIQUETA correspondiente. Alinee por los bordes izquierdos el grupo de ETIQUETAS. Haga lo mismo con el grupo de CAMPOS. Haga igual el espacio vertical de los CAMPOS. Alinee al borde superior cada pareja de ETIQUETA y CAMPO.
- 3. Traslade las ETIQUETAS Artículo, Cantidad, Precio e Importe, del encabezado de página al encabezado de grupo, debajo de los objetos anteriores.
- 4. Adapte la altura de la banda.
- 5. Trace un RECTÁNGULO que comience incluyendo las cabeceras *Artículo*, *Cantidad*, *Precio*, *Importe* y se extienda hacia abajo, abarcando los objetos de las bandas DETALLE y PIE DE GRUPO.
- 6. Pruebe el informe. Sería mejor que cada factura ocupara una página.

Diseñador de Infor	mes			
▲ Encabezado de	e página			
Factura	fac			
Fecha	fec			
Cliente	nom			
Artículo		Cantidad	Precio	Importe
Encabezado de	e grupo 1: fac	;		· · · · · ·
nom		can	pre	linfac.can*
▲ Detalle				
				linfac.can
▲ Pie de grupo 1:	fac			
A Pie de página				

- 7. Emita INFORME AGRUPAR DATOS. Active la casilla COMENZAR CADA GRUPO EN UNA NUEVA PÁGINA.
- 8. Pruebe el informe. El número de factura está demasiado lejos del texto Factura.
- 9. Cambie la expresión del CAMPO Fac por alltrim(str(fac)). Déle formato de caracteres, con alineación izquierda.
- 10. Pruebe el informe. Sería mejor separar las cabeceras de los datos.
- 11. Emita VER MOSTRAR POSICIÓN. En la parte inferior del diseñador aparece la barra de estado. Arrastre el mouse y verá en ella que varían las coordenadas horizontal y vertical del puntero. Seleccione cualquier objeto y verá números que describen sus extremos superior, inferior, izquierdo y derecho, su alto y su ancho. Seleccione el RECTÁNGULO y tome nota de sus coordenadas izquierda y derecha.
- 12. Trace una LíNEA horizontal, que comience en la coordenada izquierda anotada y termine en la derecha. Colóquela por debajo de las etiquetas *Artículo*, *Cantidad*, *Precio*, *Importe*.
- 13. Pruebe el informe. Quizás debería haber líneas verticales entre las columnas de datos.
- 14. Seleccione el RECTÁNGULO y tome nota de las coordenadas superior e inferior.
- 15. Trace tres LÍNEAS verticales, que comiencen en la coordenada superior y terminen en la inferior, entre

las columnas de artículo y cantidad, cantidad y precio, precio e importe.

16. Pruebe el informe.

Ejercicio 5 a

En el Ejercicio 4, los registros del informe EJER4 estaban ordenados por número de factura. Si queremos que estén ordenados por número de cliente, no hay modo de corregir ese informe. Ello se debe a cómo están trazadas las relaciones:



LINFAC es el origen de la cadena de relaciones. En este esquema, ninguna relación necesita ser definida de uno a varios, pues para varios artículos vendidos con el mismo número de factura en LINFAC, hay una sola factura en ENCFAC y un solo nombre en ART, y para cada cliente que ha originado una factura en ENCFAC, hay un solo cliente en CLI. Es imposible ordenar el impreso por nombre del cliente, pues la tabla CLI es movida por ENCFAC, que es movida por LINFAC. Debemos plantear otro esquema, por ejemplo, donde las relaciones partan de CLI:



En este caso, las relaciones 1 y 2 deben ser uno a varios, pues cada cliente de CLI puede tener varias facturas en ENCFAC y cada factura de ENCFAC puede estar compuesta por varios artículos en LINFAC. La relación 3 sigue siendo uno a uno. Aunque este segundo esquema es lógico, no funciona, porque la tabla que siempre estará seleccionada será Cli, no pudiendo mover la relación 2, que actúa como uno a uno. Hay que plantear, si es posible, un tercer esquema, que sí funciona bien:



Ahora la única relación que necesita ser uno a varios es la 1.

Cree el informe EJER5.

- 1. Cree un nuevo informe y defina su entorno como el último esquema anterior.
- 2. Para que todos los objetos con texto tengan otra fuente predeterminada, emita INFORME FUENTE PREDETERMINADA y elija otra fuente y/o otro tamaño.
- 3. Agregue a la banda de detalle cuatro CAMPOS, con las expresiones *Art.Nom*, *Linfac.Can*, *Linfac.Pre* y *Linfac.Can* * *Linfac.Pre*.
- 4. Define dos agrupamientos de datos: el primero por CLI.CLI y el segundo por ENCFAC.FAC. Esto quiere decir que va a formar grupos de registros que tengan el mismo código de cliente, y dentro de cada uno de ellos grupos menores de registros que tengan el mismo número de factura.
- 5. Abra el encabezado de grupo 1 y coloque un CAMPO con expresión Cli.Nom, en negrita y cursiva.
- 6. Abra el encabezado de grupo 2 y coloque dos CAMPOS con expresiones *Encfac.Fec* y *Encfac.Fec*, en negrita.
- 7. Abra el pie de grupo 2 y coloque un CAMPO con expresión *Encfac.Can* * *Linfac.Pre*, con el cálculo *Suma* y en negrita. Fíjese que al definir el cálculo, para poner a 0 la suma le ofrece el grupo *Fac*. Podría elegir otro concepto, pero el ofrecido es correcto, porque hay que borrar el acumulador de la suma cada vez que se empiece con una nueva factura.
- 8. Abra el pie de grupo 1 y coloque un CAMPO con expresión *Encfac.Can* * *Linfac.Pre*, con el cálculo *Suma*, en negrita y cursiva.

Diseñador de informe	es					
Cliente	Factura	Fecha	Artículo	Cantidad	Precio	Importe
▲ Encabezado de P	Página					
nom						
▲ Encabezado d	e grupo 1:	cli				
	fac	fec				
▲ Encabezado de g	rupo 2: fac					
			nom	can	pre	linfac.can*
▲ Detalle				1 1	<u> </u>	1 1
						linfac.can*
▲ Pie de grupo 2: fa	IC					-
						linfac.can*
▲ Pie de grupo 1: cl	i					т – т
Pie de página						

9. Pruebe el informe.

Ejercicio 5 b

Modifique EJER5. ¿Para qué usar una línea entera para encabezar el grupo 1 y otra para encabezar el grupo 2?¿No podría ir en la banda DETALLE? En este ejemplo, tal como se ha diseñado, hay lugar suficien-

te para lo que se escribe en los encabezados de grupo.

- 1. Traslade los CAMPOS que están en los encabezados de grupo 1 y 2 a la banda DETALLE.
- 2. Cierre las bandas de encabezados de grupo.
- 3. Pruebe el informe. Como ve, otra vez aparecen datos repetidos.
- 4. Haga que esos tres CAMPOS no impriman valores repetidos, excepto cuando el detalle pasa a otra hoja.
- 5. Pruebe el informe.

Diseñador de informe	S					
Cliente	Factura	Fecha	Artículo	Cantidac	I Precio	Importe
▲ Encabezado de P	ágina					
▲ Encabezado de g	rupo 1: cli					
▲ Encabezado de g	rupo 2: fac					
nom	fac	fec	Nom	can	pre	linfac.can*
▲ Detalle						
						linfac.can*
▲ Pie de grupo 2: fa	C					· ·
						linfac.can*
▲ Pie de grupo 1: cli						
▲ Pie de página						

Ejercicio 5 c

¿Podría hacer lo recíproco del ejercicio anterior, colocando los CAMPOS CALCULADOS de los pies de grupo en la banda DETALLE? Depende de lo que quiera hacer. Si sólo le interesa mostrar el valor acumulado en los CAMPOS CALCULADOS cuando cambie el cliente o la factura, no es posible, porque carecería de lógica. Si le interesa mostrar cómo se van incrementando esos campos registro a registro dentro de los grupos, sí. Modifique EJER5.

- Como posiblemente haya que tener más espacio en la página, emita ARCHIVO PREPARAR PÁGINA. En el diálogo que aparece, pulse el botón ESPECIFICAR IMPRESORA. En el nuevo diálogo, active orientación VERTICAL. Salga de ambos diálogos con ACEPTAR. El diseñador se ve más ancho, porque ahora es "apaisado".
- 2. Arrastre los Campos de los pies de grupos a la derecha de la banda DETALLE.
- 3. Cierre las bandas de pie de grupo.
- 4. Pruebe el informe.

Diseñador o	le informes							
Cliente	Factura	Fecha	Artículo	Cant.	Precio	Importe	Ac.Fact.	Ac.Cte.
▲ Encabez	zado de Pági	na						
▲ Encabez	zado de grup	o 1: cli						
▲ Encabez	zado de grup	o 2: fac						
nom	fac	fec	nom	can	pre	linfac.can*	linfac.ca	linfac.ca
▲ Detalle								
A Pie de g	rupo 2: fac							
▲ Pie de g	rupo 1: cli							
A Pie de p	ágina							

Ejercicio 6

Cree el informe EJER6. Va a listar los campos de la tabla LINFAC y escribir un texto sólo si el producto de la cantidad por el precio es mayor a 1000.

- 1. Cree un nuevo informe y agregue la tabla LINFAC al entorno.
- 2. Arrastre uno a uno todos los campos, formando una fila.
- Agregue un CAMPO, a la derecha de los anteriores y a la misma altura, con la expresión iif(linfac.can * linfac.pre) > 1000, "Supera 1000", "").
- 4. Pruebe el informe.
- 5. Traslade el CAMPO de la derecha debajo de los demás.
- 6. Pruebe el informe. Se imprima o no el texto *Supera 1000*, se respeta el espacio físico que ocupa el CAMPO inferior. Sería mejor eliminarlo cuando esté en blanco.
- 7. En EXPRESIÓN DE INFORME del CAMPO inferior, entre a IMPRIMIR–CONDICIONES. Active QUITAR LÍNEAS EN BLANCO.
- 8. Pruebe el informe.

Ejercicio 7

Hay otra manera de condicionar el texto del CAMPO inferior del Ejercicio anterior.

- 1. A partir de EJER6, cree el Formulario EJER7.
- 2. Corrija la expresión del CAMPO inferior por "Supera 1000".
- 3. Entre a IMPRIMIR CONDICIONES. EN IMPRIMIR SÓLO CUANDO LA EXPRESIÓN SEA VERDADERA, escriba *Linfac*. *Can* * *Linfac*.*Pre* > 10000.
- 4. Pruebe el informe.

Ejercicio 8

Cree el informe EJER8, que calcule la fórmula Σ [(Linfac.Can * Linfac.Pre) + 1000)] / 7 por cada página.

La expresión (*Sum* (*Linfac.Can* * *Linfac.Pre*) + 1000) / 7, que traduce la fórmula, no es posible en los campos de informes. Estos no aceptan la función Sum(), porque disponen en su lugar del botón SUMA en el diálogo CALCULAR CAMPO. No podremos entonces usar un campo calculado. La solución es usar una variable de informe.

- 1. Cree un nuevo informe.
- 2. Agregue la tabla LINFAC.
- 3. Arrastre uno por uno los campos del cursor a la banda DETALLE.
- 4. Ponga los campos a la misma altura.
- 5. En el diálogo traído por INFORME VARIABLES, defina la variable Sumatoria. En VALOR INICIAL escriba 1000 y en ALMACENAR VALOR escriba Linfac. Can * Linfac. Pre. En CALCULAR active el botón SUMA. En REINICIAR EN elija Fin de página. El valor inicial interviene al crear o restaurar la variable. Este valor se va incrementando registro a registro, con el producto entre cantidad y precio. Cuando termine el proceso de los registros que entren en una página, se tendrá el numerador de la fórmula.
- 6. Coloque tres CAMPOS calculados en la banda PIE DE PÁGINA, de izquierda a derecha, como sigue: el primero con la expresión *Linfac.Can * Linfac.Pre* y el cálculo SUMA; el segundo con la expresión *Sumatoria*, numerador de la fórmula; el tercero con la expresión sea *Sumatoria / 7*, que completa la fórmula.
- 7. Pruebe el informe. En el pie de página, el dato de la izquierda difiere en 1000 del que está en el centro. El de la derecha es igual al que está en el centro dividido en 7.

Ejercicio 9

Cree el informe EJER9, que muestre en hojas separadas el importe de facturas de cada vendedor. Debe llevar un título inicial y un resumen de ventas.

- 1. Cree un nuevo informe.
- 2. Agregue las tablas ENCFAC, VEN y LINFAC. Active el índice VEN de la primera.
- 3. Trace dos relaciones: de ENCFAC.VEN a VEN.VEN y de ENCFAC.FAC a LINFAC.FAC, ésta de uno a varios.
- 4. Cree dos agrupamientos por las expresiones *Encfac. Ven* y *Encfac. Fac.* Indique que cada grupo de *Encfac. Ven* debe comenzar en una nueva página.
- 5. Cierre la banda DETALLE. Abra las bandas PIE DE GRUPO 1 y PIE DE GRUPO 2.
- 6. En PIE DE GRUPO 2, agregue tres CAMPOS, horizontalmente. El primero con expresión *Ven.Nom*, que no imprima valores repetidos, excepto en primera banda completa de nueva página. El segundo con expresión *Encfac.Fac*. El tercero con expresión *Linfac.Can* * *Linfac.Pre*, cálculo SUMA y restablecimiento (puesta a 0) cuando cambie *Encfac.Fac*.
- 7. En la banda ENCABEZADO DE PÁGINA, coloque ETIQUETAS adecuadas a cada Campo: Vendedor, Factura, Importe.
- 8. Copie el campo calculado y péguelo en PIE DE GRUPO 2. Haga que se restablezca cuando cambie *Ven. Ven*.
- 9. Pruebe el informe.

Ejercicios con informes y entornos de datos - Edmundo A. Cáceres - 2011

Diseñador de informes		
Vendedor	Factura	Importe
Encabezado de página		
▲ Encabezado de grupo 1: ven		
▲ Encabezado de grupo 2: fac		
▲ Detalle		
nom	fac	linfac.can * linfac.pre
▲ Pie de grupo 2: fac		
		linfac.can * linfac.pre
▲ Pie de grupo 1: ven		· · ·
▲ Pie de página		

- 10. Falta agregar un título y un resumen, como se ha pedido. En INFORME TÍTULO / RESUMEN, active los cuatro botones. Aparecen dos nuevas bandas, en los extremos del diseñador: Título y Resumen.
- Aumente el alto de la banda TÍTULO. Agregue dos ETIQUETAS con los textos Transportes del Centro Argentino S. A. e Informe de producción de vendedores. Agregue un CAMPO con la expresión "Fecha de emisión: " + date(). Coloque los objetos uno bajo el otro, déle efectos de fuente que los destaquen y céntrelos horizontalmente.
- 12. Aumente el alto de la banda *Resumen*. Agregue un CAMPO con la expresión *"Resumen de producción de vendedores, al día" + date()*. Agregue una ETIQUETA con el texto *Total*. Agregue una copia del CAMPO que tiene la sumatoria, controlando que se restaure al final del informe.
- 13. Pruebe el informe.

Diseñador de informes							
Transportes del Centro Argentino S. A. Informe de producción de vendedores "Fecha de emisión: " + date()							
▲ Título							
Vendedor ▲ Encabezado de página	Factura	Importe					
▲ Encabezado de grupo 1: ven							
▲ Encabezado de grupo 2: fac							
▲ Detalle							
nom	fac	linfac.can * linfac.pre					
 Pie de grupo 2: fac Pie de grupo 1: ven 		linfac.can * linfac.pre					
Pie de página							
"Resumen de producción de vendedores, al día" + date() Total Linfac.can *Linfac.Pre							