

Comando EDIT EXTENDED para la librería HMG.

(c) Cristóbal Mollá [cemese@terra.es]

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

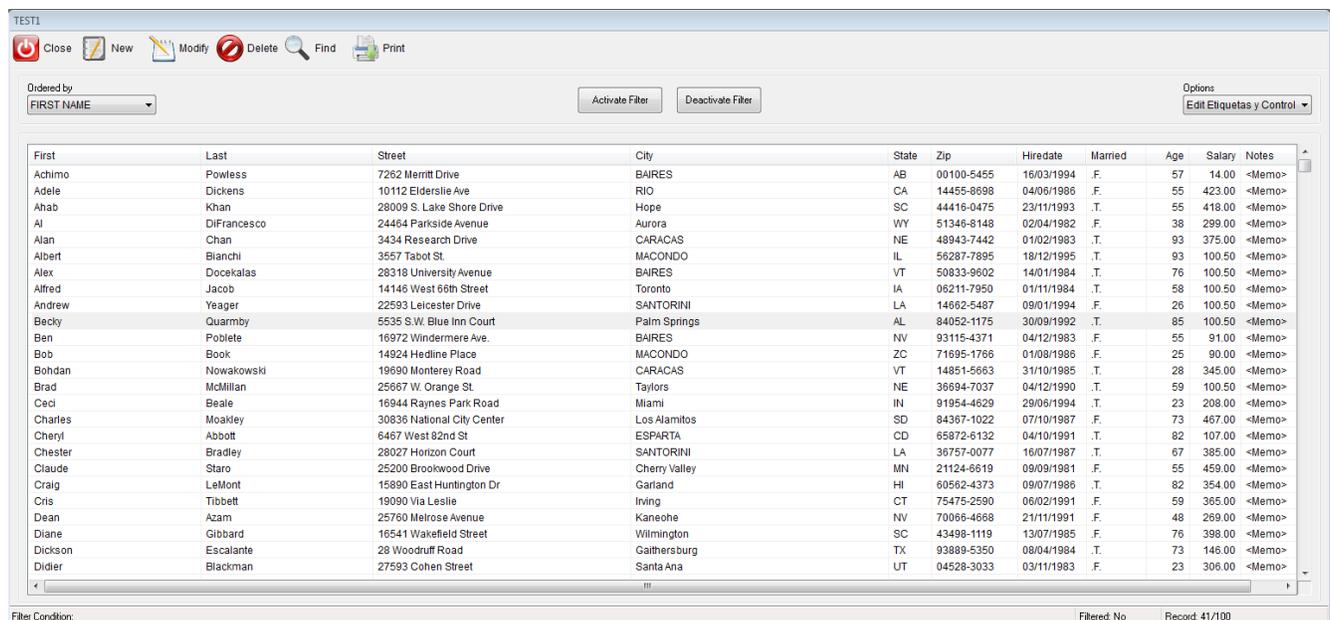
Descripción

EDIT EXTENDED, es un comando que permite realizar el mantenimiento de una bdd. En principio está diseñado para administrar bases de datos que usen los drivers DBFNTX y DBFCDX, presentando otras bases de datos con diferentes drivers resultados inesperados.

Limitaciones

- No implementa ListBox ni ComboBox
- No se pueden realizar búsquedas por campos lógico o memo.
- No se pueden realizar búsquedas en índices con claves compuestas, la búsqueda se realiza por el primer campo de la clave compuesta.

Listado (List Window)



The screenshot shows a window titled 'TEST1' with a menu bar (Close, New, Modify, Delete, Find, Print) and a toolbar. Below the toolbar, there is a dropdown menu for 'Ordered by' set to 'FIRST NAME', and buttons for 'Activate Filter' and 'Deactivate Filter'. On the right, there is an 'Options' dropdown menu set to 'Edit Etiquetas y Control'. The main area contains a table with the following data:

First	Last	Street	City	State	Zip	Hiredate	Married	Age	Salary	Notes
Achimo	Powless	7262 Merritt Drive	BAIRES	AB	00100-5455	16/03/1994	F.	57	14.00	<Memo>
Adele	Dickens	10112 Elderslie Ave	RIO	CA	14455-8698	04/06/1986	F.	55	423.00	<Memo>
Ahab	Khan	28009 S. Lake Shore Drive	Hope	SC	44416-0475	23/11/1993	.T.	55	418.00	<Memo>
Al	DiFrancesco	24464 Parkside Avenue	Aurora	WY	51346-8148	02/04/1982	F.	38	299.00	<Memo>
Alan	Chan	3434 Research Drive	CARACAS	NE	48943-7442	01/02/1983	.T.	93	375.00	<Memo>
Albert	Bianchi	3557 Tabot St.	MACONDO	IL	56287-7895	18/12/1995	.T.	93	100.50	<Memo>
Alex	Docekalas	28318 University Avenue	BAIRES	VT	50833-9602	14/01/1984	.T.	76	100.50	<Memo>
Alfred	Jacob	14146 West 66th Street	Toronto	IA	06211-7950	01/11/1984	.T.	58	100.50	<Memo>
Andrew	Yeager	22593 Leicester Drive	SANTORINI	LA	14662-5487	09/01/1994	F.	26	100.50	<Memo>
Becky	Quarmby	5535 S.W. Blue Inn Court	Palm Springs	AL	84052-1175	30/09/1992	.T.	85	100.50	<Memo>
Ben	Poblete	16972 Windermere Ave.	BAIRES	NV	93115-4371	04/12/1983	F.	55	91.00	<Memo>
Bob	Book	14924 Hedline Place	MACONDO	ZC	71695-1766	01/08/1986	F.	25	90.00	<Memo>
Bohdan	Nowakowski	19690 Monterey Road	CARACAS	VT	14851-5663	31/10/1985	.T.	28	345.00	<Memo>
Brad	McMillan	25667 W. Orange St.	Taylors	NE	36694-7037	04/12/1990	.T.	59	100.50	<Memo>
Ceci	Beale	16944 Raynes Park Road	Miami	IN	91954-4629	29/06/1994	.T.	23	208.00	<Memo>
Charles	Moakley	30836 National City Center	Los Alamitos	SD	84367-1022	07/10/1987	F.	73	467.00	<Memo>
Cheryl	Abbott	6467 West 82nd St	ESPARTA	CD	65872-6132	04/10/1991	.T.	82	107.00	<Memo>
Chester	Bradley	28027 Horizon Court	SANTORINI	LA	36757-0077	16/07/1987	.T.	67	385.00	<Memo>
Claude	Staro	25200 Brookwood Drive	Cherry Valley	MN	21124-6619	09/09/1981	F.	55	459.00	<Memo>
Craig	LeMont	15890 East Huntington Dr	Garland	HI	60562-4373	09/07/1986	.T.	82	354.00	<Memo>
Oris	Tibbett	19090 Via Leslie	Ivring	CT	75475-2590	06/02/1991	F.	59	365.00	<Memo>
Dean	Azam	25760 Melrose Avenue	Kaneohe	NV	70066-4668	21/11/1991	F.	48	269.00	<Memo>
Diane	Gibbard	16541 Wakefield Street	Wilmington	SC	43498-1119	13/07/1985	F.	76	398.00	<Memo>
Dickson	Escalante	28 Woodruff Road	Gaithersburg	TX	93889-6350	08/04/1984	.T.	73	146.00	<Memo>
Didier	Blackman	27593 Cohen Street	Santa Ana	UT	04528-3033	03/11/1993	F.	23	306.00	<Memo>

Filter Condition: Filtered: No Record: 41/100

Sintaxis

Todos los parámetros del comando EDIT son opcionales.

```
EDIT EXTENDED ;
[ WORKAREA cWorkArea ] ;
[ TITLE cTitle ] ;
[ FIELDNAMES acFieldNames ] ;
[ FIELDMESSAGES acFieldMessages ] ;
[ FIELDENABLED alFieldEnabled ] ;
[ TABLEVIEW alTableView ] ;
[ OPTIONS aOptions ] ;
[ ON SAVE bSave ] ;
[ ON FIND bFind ] ;
[ ON PRINT bPrint ]
```

Uso básico

```
EDIT EXTENDED WORKAREA cWorkArea
```

Ventana de Edición (Edit Window)

The image shows a 'Modify record' dialog box with a title bar containing 'Cancel', 'Ok', and 'Copy' buttons. The dialog contains a form with the following fields and values:

First	Andrew
Last	Yeager
Street	22593 Leicester Drive
City	Kempton Park
State	LA
Zip	14662-5487
Hiredate	<input checked="" type="checkbox"/> 09/01/1994
Married	<input type="checkbox"/>
Age	26
Salary	100.50
Notes	This is a test for record 52

Two red arrows point to the 'State' field and the text 'No ListBox'.

Si no se pasa ningún parámetro, el comando EDIT toma como bdd de trabajo el area de trabajo actual.

- [cWorkArea] Cadena de texto con el nombre del alias de la base de datos a editar. Por defecto el alias de la base de datos activa.
- [cTitle] Cadena de texto con el título de la ventana de visualización de registros.
Por defecto se toma el alias de la base de datos activa.
- [acFieldNames] Matriz de cadenas de texto con el nombre descriptivo de los campos de la base de datos. Tiene que tener el mismo número de elementos que campos la bdd.
Por defecto se toma el nombre de los campos de la estructura de la bdd.
- [acFieldMessages] Matriz de cadenas de texto con el texto que aparecerá en la barra de estado cuando se este añadiendo o editando un registro. Tiene que tener el mismo numero de elementos que campos la bdd.
Por defecto se rellena con valores internos.
- [alFieldEnabled] Matriz de valores lógicos que indica si el campo referenciado por la matriz esta activo durante la edición de registro. Tiene que tener el mismo numero de elementos que campos la bdd.
Por defecto toma todos los valores como verdaderos (.t.)
- [aTableView] Matriz de valores lógicos que indica si el campo referenciado por la matriz es visible en la tabla. Tiene que tener el mismo numero de elementos que campos la bdd.
Por defecto toma todos los valores como verdaderos (.t.).
- [aOptions] Matriz de 2 niveles. el primero de tipo texto es la descripción de la opción. El segundo de tipo bloque de código es la opción a ejecutar cuando se selecciona.
Si no se pasa esta variable, se desactiva la lista desplegable de opciones.
- [bSave] Bloque de codigo con el formato { |aValores, lNuevo| Accion } que se ejecutará al pulsar la tecla de guardar registro. Se pasan las variables aValores con el contenido del registro a guardar y lNuevo que indica si se está añadiendo (.t.) o editando (.f.). Esta variable ha de devolver .t. para salir del modo de edición.
Por defecto se graba con el código de la función.
- [bFind] Bloque de codigo a ejecutar cuando se pulsa la tecla de busqueda.
Por defecto se usa el código de la función.
- [bPrint] Bloque de código a ejecutar cuando se pulsa la tecla de listado.
Por defecto se usa el codigo de la función.

Función ABM3() <Editexplus> para la librería HMG.

Consiste en una modificación a la función ABM2() (comando EDIT EXTENDED)

Los parámetros son los mismos del comando EDIT EXTENDED más dos adicionales, el primero (parámetro N° 11) es el nombre de un archivo con la descripción del formulario. (xxxx.arr)

El otro (parámetro N° 12) Es una matriz que permite tener ComboBox o ListBox

Ejemplo con 2 columnas distribuidas horizontalmente.

The screenshot shows a 'Modify record' dialog box with a light blue title bar and standard Windows-style buttons (Cancel, Ok, Copy). The form is organized into two columns. The left column contains: First (Andrew), Street (22593 Leicester Drive), State (LA), Hiredate (09/01/199), Age (26), and Notes (This is a test for record 52). The right column contains: Last (Yeager), City (SANTORINI), Zip (14662-5487), Married (checkbox), and Salary (100.50). Red arrows point to the State and City dropdown menus, labeled 'ListBox', and to the Age field, labeled 'Size changed'.

El archivo con la descripción del formulario. (xxxx.arr) se genera con un solo clic, según se explica más abajo.

Ejemplo con 3 columnas distribuidas verticalmente.

The screenshot shows a 'Modify record' dialog box with a light blue title bar and standard Windows-style buttons (Cancel, Ok, Copy). The form is organized into three columns. The first column contains: First (Alan), Last (Chan), Street (3434 Research Drive), and City (CARACAS). The second column contains: State (NE), Zip (48943-7442), Hiredate (01/02/198), and Married (checkbox). The third column contains: Age (93), Salary (375.00), and Notes (This is a test for record 35).

Para los Combo / ListBox se puede: (ver página 6)

- **Pasar las opciones como parámetros.**
- **Indicar una tabla relacionada para obtenerlos.**
- **Crear una función que devuelva la lista de opciones.**

Creación de los archivos de definición de la matriz

11º parámetro → xxxx.arr

Simplemente ejecutando “ABM3generar()” en “h_edit_makeArray.prg”
(Esta función no debería estar disponible en el proyecto final)

4 archivos son creados para que elija el que más se adapte a su necesidad.

Organizados en 2 columnas o en 3 columnas.

Debe elegir uno y llevarlo donde se vaya a utilizar. Puede cambiarle el nombre si lo desea.

“ABM3generar()” Abre una caja de diálogo que permite elegir cualquiera de las tablas que estén abiertas. Una vez elegida la tabla, se generan los mencionados archivos “xxxx.arr”



En el proyecto final junto con el archivo *.arr recientemente creado, debe estar
“h_edit_exPlus.prg” donde se incluye la función **ABM3()**

En el DEMO en el menú / **Make arrays / make all 4 arrays.**
Se generan los 4 archivos mencionados.

En los otros ítems del menú, se muestra el resultado de todos ellos.

Menú de compatibilidad. Muestra la forma tradicional sin pasar ningún parámetro.

Menú 2 columnas: Organizadas de arriba hacia abajo o de izquierda a derecha.

Menú 3 columnas: Organizadas de arriba hacia abajo o de izquierda a derecha.

Como hacer los ComboBox

El parámetro 12° → ComboBox

Es una matriz que contiene un elemento por cada control (campo)

Este elemento puede ser:

- **Un campo de texto con la referencia a la tabla y campo de origen del ComboBox.**
- **Una matriz con la lista de opciones.**

Solo importan los que coinciden con un campo de texto, los demás serán ignorados.

En el primer caso debe contener la referencia a la tabla y campo de origen del ComboBox.

De la siguiente forma: **“Tabla -> Campo”**

```
aCombo := {,,,"Ciudades->City" ,,,,,,}  
ABM3 (,,,,,,,"TEST1.arr",aCombo)
```

En el segundo caso, las matrices contenidas para cada control de texto en que se desee tener un ComboBox, contendrán la lista de opciones. (Esto sería un ListBox)

```
aList := {,,,{ "AB", "CD", "AR", "CA", "AO", "ZC", "SC", "WY", "NE", "IL", "VT", "IA", "LA" },,,,,,}  
ABM3 (,,,,,,,"TEST1.arr",aList)
```

Pueden combinarse en una sola matriz ambos estilos. (ver Demo)

Se pueden tener ComboBox y ListBox en el mismo formulario.

Debe ser una sola matriz (array) que los contenga.

- **Los campos de texto correspondientes a posiciones de la matriz que estén vacíos, se comportarán como un TextBox normal.**
- **Si la correspondiente posición es un campo de texto, se utilizará la tabla referenciada (si existe y está abierta) para construir el ComboBox.**
- **Si la correspondiente posición es una matriz, se utilizarán los componentes (texto entrecomillado) para construir un ListBox.**
- **Cualquiera de estos métodos toma preferencia sobre una posible función definida según se explica en la página siguiente.**

Todos los métodos se muestran en el Demo

- La última forma (candidata a ser eliminada por ser ahora innecesaria):
No se pasa ningún parámetro, pero se crea una función según se indica:

Como hacer los ListBox

Para ésto debe crearse una simple función para cada control de texto.
El nombre de esta función debe tener la forma 'LB_dbName_FieldName' y debe retornar una matriz de texto con la lista de opciones.

Ejemplo en 'demo.prg'

```
FUNCTION LB_Test1_STATE()  
RETURN {"AB","CD","AR","CA","AO","ZC","SC","WY","NE","IL","VT","IA","LA"}
```

En este otro ejemplo en el campo 'nomina' de la tabla 'personal' se introduce un dato proveniente de la tabla 'NomDefi' (campo 'nomina')

```
FUNCTION LB_personal_NOMINA()  
  local nRegistro := 0  
  local aOpciones := {}  
  nRegistro := NomDefi->( RecNo() )  
  NomDefi->( dbGoTop() )  
  Do While NomDefi->( !Eof() )  
    Aadd (aOpciones, NomDefi->nomina)  
    NomDefi->( DbSkip() )  
  End  
  NomDefi->( dbGoTo( nRegistro ) )  
RETURN aOpciones
```

Dichas funciones deben estar visibles cuando se llama a **ABM3()**

Como hacer pequeñas modificaciones

Se provee una función relativamente simple para tal efecto.

Lee la matriz con la definición de los controles y etiquetas “xxxx.arr” y permite modificar sus posiciones relativas y su tamaño, así como el texto de los mensajes.

Atención: Utilizar con mucho cuidado.

Es una función muy elemental. No realiza ninguna comprobación. No tiene deshacer.

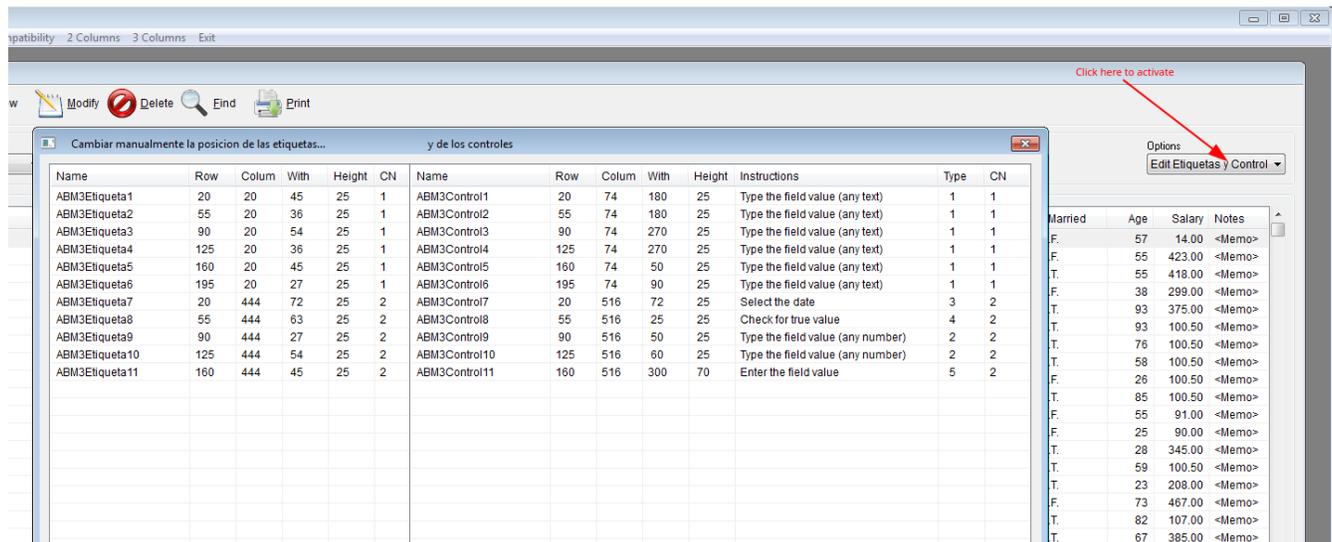
Si todo se estropea la única solución es volver a crear el archivo.

No vuelve a calcular el tamaño de la ventana así que un control puede quedar fuera del espacio visible.

Se recomienda solamente para pequeños ajustes aunque en la práctica es muy poderosa.

Se podrían hacer cambios radicales de posición, incluso agregar nuevas etiquetas de texto.

Su uso queda a criterio (y riesgo) del diseñador. Se provee solamente como una opción adicional.



El archivo “h_edit_ModifArray.prg” contiene la función “ModificarArray()”

En el DEMO se utiliza la ‘opción de usuario’ disponible en Edit Extended (ver demo.prg)

Esta se encuentra disponible en la ventana de lista (List Window) arriba a la derecha.

Se abre un cuadro con los parámetros de posición de las etiquetas y controles definidos.

Trabaja sobre el archivo de definición (xxxx.arr) que esté en uso en ese momento.

Con Click o Enter se puede modificar el valor de la casilla. Como se mencionó no hace ningún control.

Vuelve a grabar la matriz al cerrar esta ventana (no importa si se cambió algo o no)

Para hacer efectivos los cambios se debe cerrar la ventana de lista y volver a abrirla para que lea el archivo recientemente modificado.

Modo de compatibilidad:

Sin incluir el archivo "xxxx .arr" (El 11º parámetro) se comporta de forma tradicional.

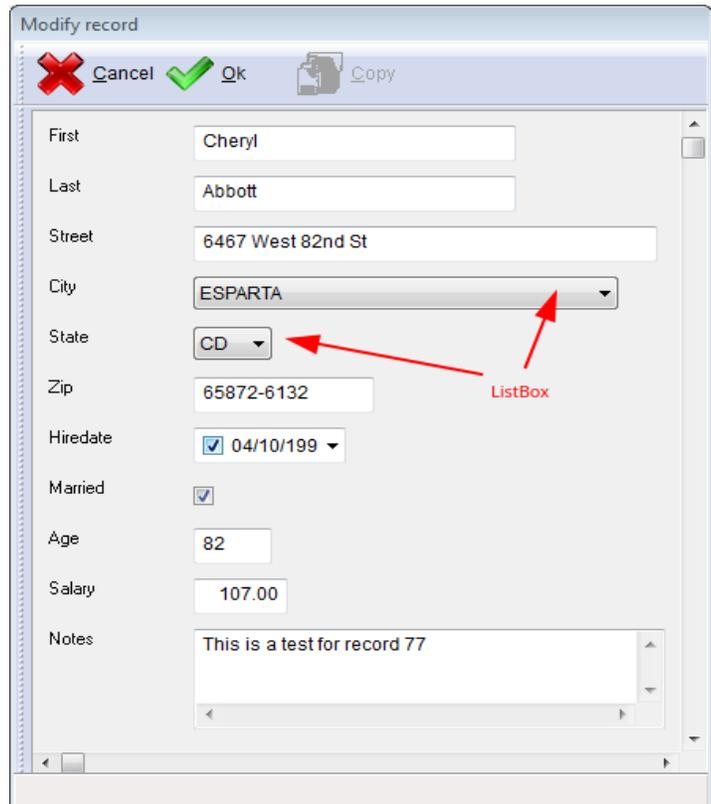
Los ListBox se implementan de forma independiente del parámetro 11, y pueden funcionar conjuntamente o sin él.

Formulario generado tradicionalmente pero que incluye ListBox → → → → →

El ListBox aparece solamente si se incluye el parámetro 12º o la función correspondiente, según se explicó en la página 7.

Al no incluir el 11º parámetro (xxxx.arr) se utiliza la disposición tradicional pero se reconocen las funciones que determinan el uso de los ListBox.

Si se quiere compatibilidad total simplemente no incluya este parámetro ni función.



Resumiendo

- El archivo 'h_edit_exPlus.prg' es el único que debe incluirse en el proyecto final.
-
- El archivo 'h_edit_makeArray.prg' es necesario solamente donde se definen los controles.
- Lo debe acompañar MsgExtended.prg.
-
- Si se utiliza 'h_edit_ModifArray.prg' debe estar solamente donde se definen los controles.
- Jamás debe estar disponible al usuario porque destruirá el archivo de definición (xxxx.arr)
-
- Los archivos 'xxxx.arr' generados deben copiarse al proyecto final, junto a las tablas.
-
- Las funciones 'LB_dbName_FieldName' -si se utilizan- deben existir (ser visibles) en el proyecto final, cuando se llama a ABM3()

Consideraciones Técnicas

.....A completar.....