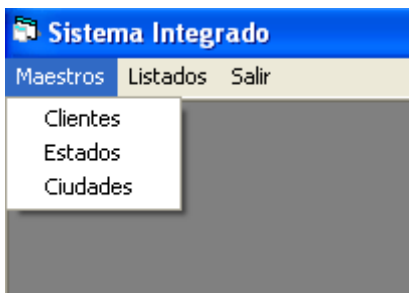


UTILIZANDO LISTA DE VALORES PARA CIUDADES.

Ya se ha revisado, como utilizar una Lista de Valores Sencilla en la Tabla de Estados, sin embargo, es de uso común que utilizar varias Listas de Valores en un mismo Formulario. Tal es el caso, de la actualización de Ciudades, ya que allí es necesario utilizar una Lista para los Estados y otra para las Ciudades.

Obedeciendo a este planteamiento, acto seguido, se encuentran los pasos necesarios para implementar varias Listas de Valores y lograr una presentación mas dinámica, lo cual es permitido por cualquier herramienta Visual.

Es importante recordar, que se ha estado trabajando con un programa pequeño de actualización de Archivos Maestros, cuyo Menú principal en el Formulario MDI es el siguiente:



Aquí en la opción de Maestros, se pueden actualizar: Clientes, Estados y Ciudades.

Ahora bien, si se selecciona la Opción de Ciudades, se irá al Formulario de actualización de Ciudades.

The screenshot shows a software window titled "Actualizando Maestro Ciudades." with a light beige background. Inside the window, there is a form with three input fields. The first field is labeled "Código Estado:" and consists of a small text box followed by a dropdown menu. The second field is labeled "Código Ciudad:" and also consists of a small text box followed by a dropdown menu. The third field is labeled "Nombre Ciudad:" and is a single-line text box. Below the form, there is a toolbar with five icons: a blue floppy disk (save), a document with a pencil (print), a red 'X' over a document (delete), a pair of binoculars (search), and a person walking (back).

En este Formulario se actualiza el Código del Estado (Cod_Estado), el Código de la Ciudad (Cod_Ciudad) y el Nombre de cada Ciudad (Nombre_Ciudad). La mencionada actualización, se realiza sobre una Tabla llamada Ciudades con la siguiente Estructura y con Cod_Estado+Cod_Ciudad como Clave Compuesta:

Nombre de Columna	Tipo de Columna	Longitud Columna.
*Cod_Estado	Texto	2
*Cod_Ciudad	Texto	2
Nombre_Ciudad	Texto	20
Estatus	Texto	1

La idea de la Actualización consiste en que cuando el usuario seleccione la Opción de Ciudades en el Menú Principal, aparezca el Formulario de Ciudades con la Opción de LISTAR y SALIR encendidas y las de INCLUIR, MODIFICAR y ELIMINAR apagadas. En este caso, el TEXTBOX donde se coloca el Código del Estado se encuentra apagado y el usuario debe seleccionar obligatoriamente un Estado de la Lista de Valores.


Actualizando Maestro Ciudades.

Código Estado:

Código Ciudad:

Nombre Ciudad:

Distrito Capital
Zulia
Lara
Portuguesa
Barinas
Miranda
Amazonas



Seguidamente, el Programa activará el TEXTBOX y el DATACOMBO que existe para las Ciudades (Estarán inicialmente apagados).


Actualizando Maestro Ciudades.

Código Estado: 03 Lara

Código Ciudad:

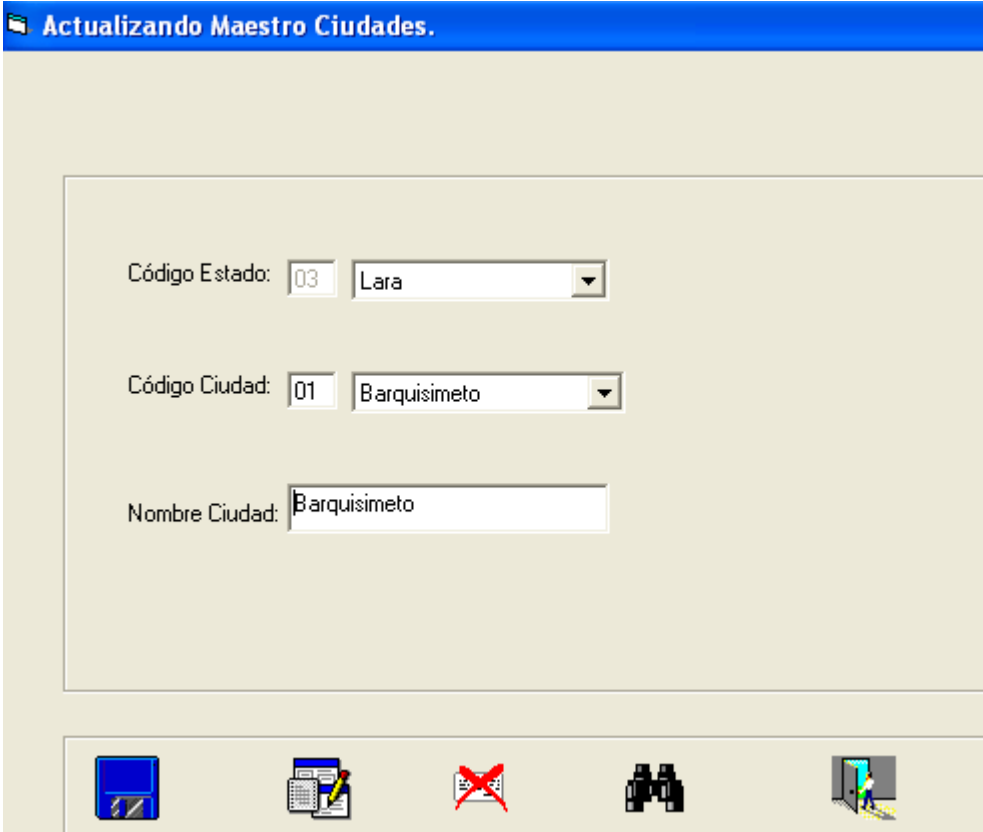
Nombre Ciudad:

Barquisimeto
Quibor
Cabudare



En ese DATACOMBO solo aparecerán las Ciudades pertenecientes al Estado Seleccionado. Si la Ciudad es nueva, el usuario escribirá un Código y ENTER, para que se encienda el Botón de Incluir.

Una Vez seleccionada la Ciudad (Si ya existe), se encenderán los Botones de Modificar y Eliminar. También se encenderá el TEXTBOX del Nombre de la Ciudad y se mostrará dicho Nombre para que el usuario pueda Modificarlo o Eliminarlo:

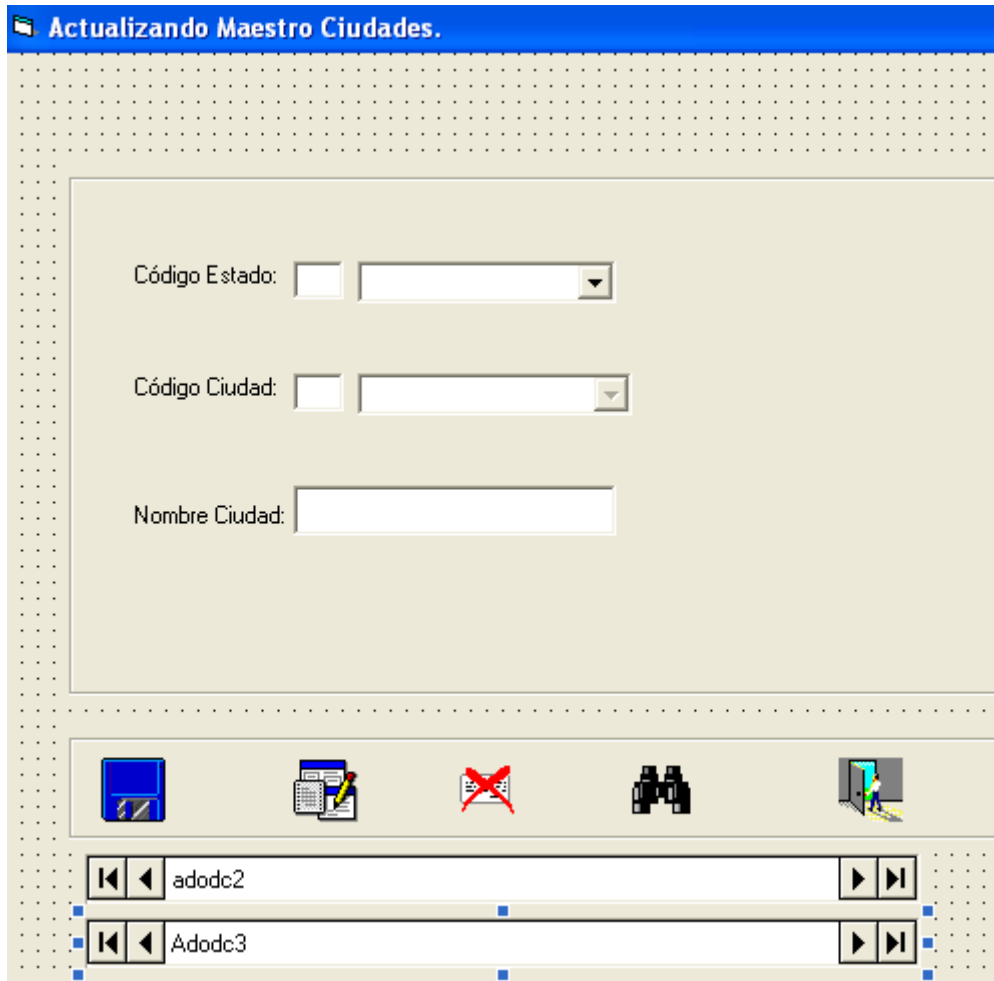


The screenshot shows a software window titled "Actualizando Maestro Ciudades." with a light beige background. Inside the window, there is a form with three rows of input fields. The first row is labeled "Código Estado:" and contains a text box with "03" and a dropdown menu showing "Lara". The second row is labeled "Código Ciudad:" and contains a text box with "01" and a dropdown menu showing "Barquisimeto". The third row is labeled "Nombre Ciudad:" and contains a text box with "Barquisimeto". Below the form, there is a horizontal bar containing five icons: a blue folder icon, a calendar icon with a pencil, a document icon with a red X, a binoculars icon, and a person walking icon.

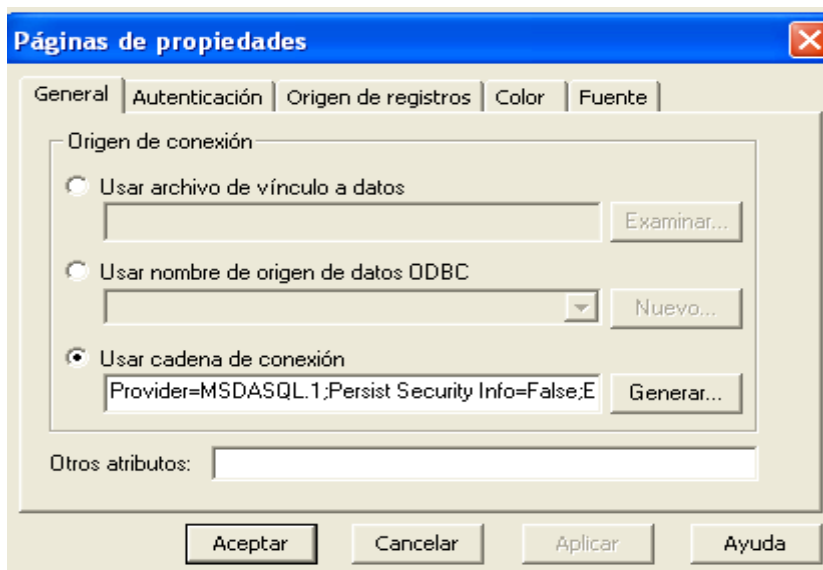
Para lograr esta actualización, se deben cumplir los siguientes pasos:

1. Las rutinas KEYPRESS se mantienen similares a las presentadas en la Sección No. 8. Solo hay que cambiar los COMMAND por IMAGE.
2. En el Formulario de Actualización de Ciudades FORMCIUDADES, colocar dos DATACOMBO. El Primero para los Estados y el Segundo para las Ciudades. El primero para los Estados, es similar al de la Sección 9.

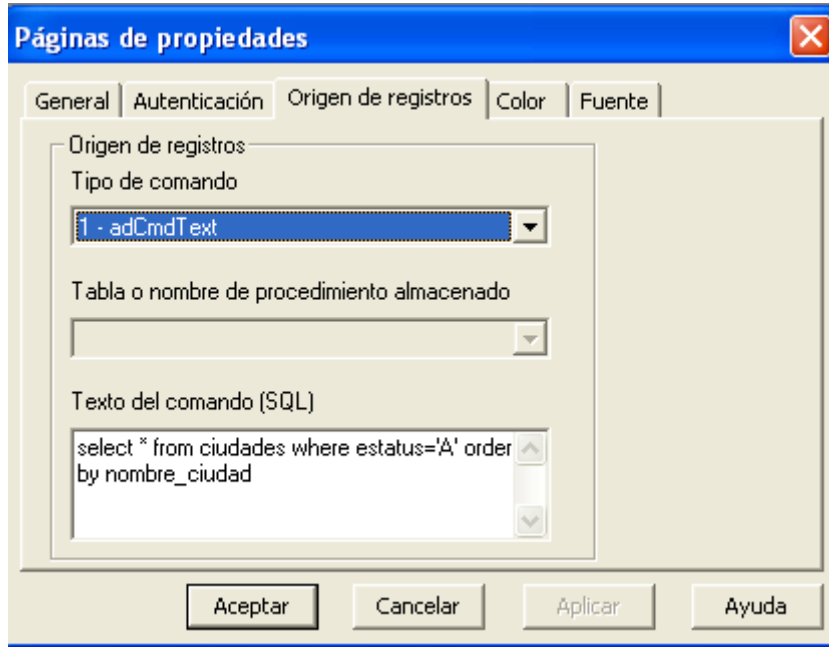
3. Colocar un nuevo ADODC3 para las Ciudades en el Formulario y crearle su cadena de conexión a la Base de Datos. El ADODC2 es para el COMBO de Estados, tal como se hizo en la sección en la sección 9.



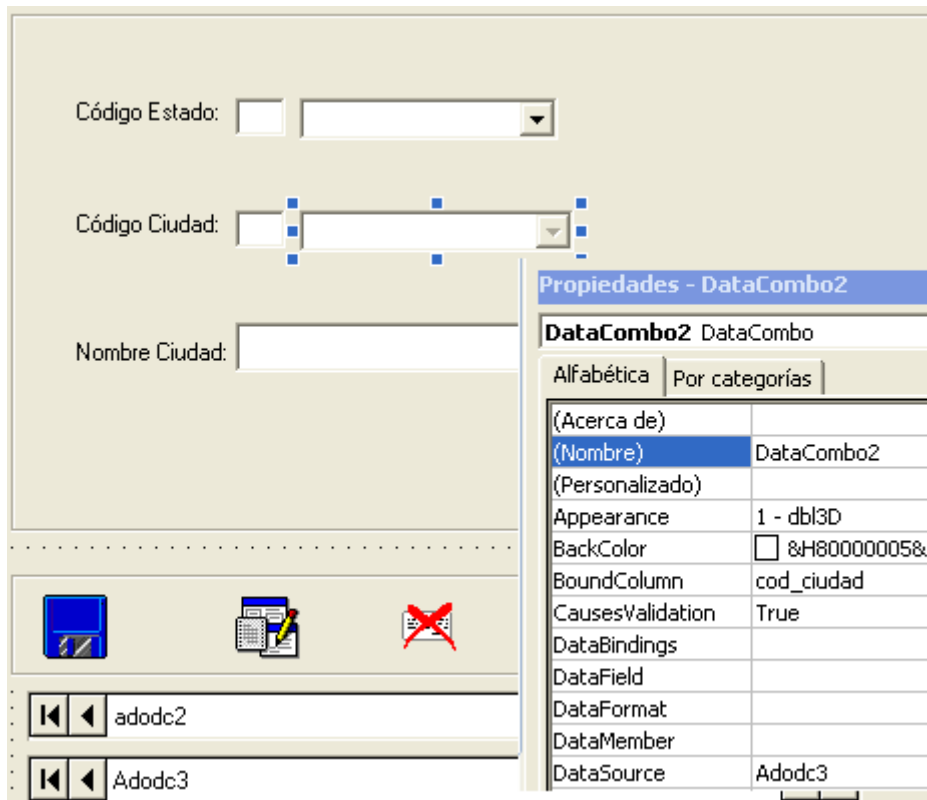
4. Una vez que se ha creado la conexión a la Base de Datos en el ADODC3, es necesario seleccionar la pestaña Origen de Registros.



- 5. Dentro de la Pestaña Origen de Registros, colocar
 - 5.1 Tipo de Comando: AdCmdText
 - 5.2 Texto de Comando SQL: `Select * from ciudades where estatus='A'`
Order By nombre_Ciudad



- 6. Ahora se debe seleccionar el DATACOMBO2 y cambiar las siguientes propiedades:



La propiedad BoundColumn = Cod_Ciudad

La propiedad ListField = Nombre_Ciudad

La propiedad RowSource = Adodc3.

7. Crear las siguientes rutinas en el Formulario:

```
Private Sub DataCombo1_Change()  
  
If Trim(DataCombo1.Text) <> "" Then  
txtcodestado.Text = DataCombo1.BoundText  
  
'Aquí se filtra que en el combo 2 solo aparezcan las ciudades asociadas al estado seleccionado  
sql = "Select cod_ciudad,nombre_ciudad" _  
    & " from ciudades where" _  
    & " cod_estado = '" + (txtcodestado.Text) + "'" _  
    & " and Estatus = 'A' "  
  
Adodc3.RecordSource = sql  
Adodc3.Refresh  
DataCombo2.Enabled = True  
txtcodciudad.Enabled = True  
txtcodciudad.SetFocus  
End If  
  
End Sub
```

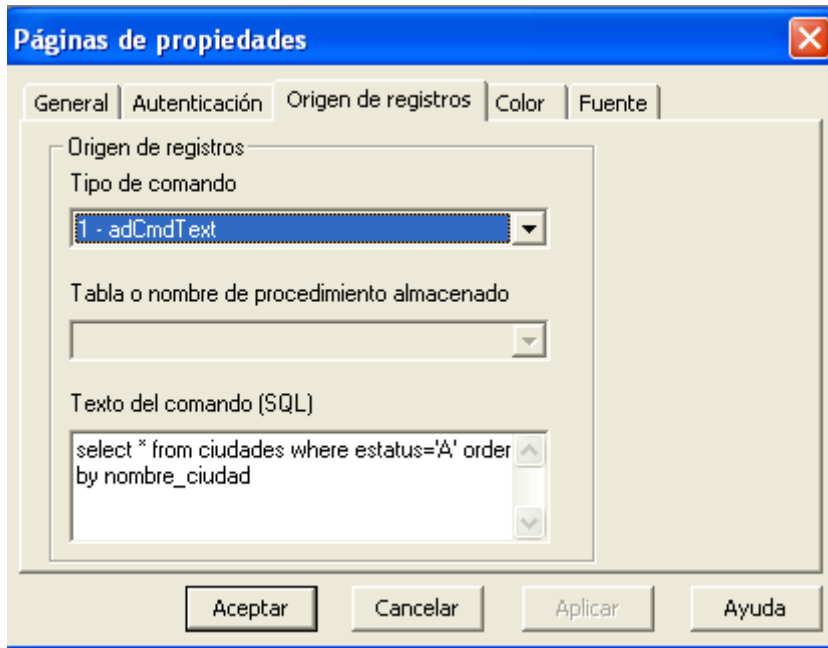
```
Private Sub DataCombo1_KeyPress(KeyAscii As Integer)  
KeyAscii = 0 ' No permite que escriban sobre el COMBO.  
End Sub
```

OJO PELAO: Se hizo necesario realizar algunos cambios en la rutina DATACOMBO1_CHANGE, obedeciendo a la siguiente situación:

Cuando el usuario seleccione un Estado, se coloca el Código de ese Estado en TXTCODESTADO.TEXT, para ello se usa la instrucción:

txtcodestado.Text = DataCombo1.BoundText. 'Ya se sabe el Código del Estado.

Ahora bien, cuando el usuario vaya a seleccionar del DATACOMBO2 (Ciudades) solo le deben salir las ciudades que pertenecen al Estado seleccionado, pero la instrucción SQL que se encuentra en el ADODC3 permite mostrar los nombres de todas las ciudades sin importar el Estado seleccionado:



Puede notarse, que el SELECT va a traer todas las ciudades sin importar el Estado a la cual pertenezcan.. Este SELECT se encuentra en la propiedad RECORDSOURCE del ADODC3:

Propiedades - Adodc3	
Adodc3	Adodc
MaxRecords	0
Mode	0 - adModeUnknown
Negotiate	False
Orientation	0 - adHorizontal
Password	
RecordSource	select * from ciudades where estatus='A' order by nombre_ciudad

En consecuencia, hay que agregar un SELECT al DATACOMBO1_CHANGE:

```
'Aqui se filtra que en el combo 2 solo aparezcan las ciudades asociadas al estado seleccionad  
sql = "Select cod_ciudad,nombre_ciudad"  
    & " from ciudades where"  
    & " cod_estado = '" + (txtcodestado.Text) + "'" _  
    & " and Estatus = 'A' "
```

Luego se le dice al ADODC3 que hay un cambio en su propiedad RECORDSOURCE:

```
Adodc3.RecordSource = sql
```


Luego, para que se actualice la Propiedad y se haga el cambio del RECORDSOURCE se coloca la instrucción:

```
Adodc3.Refresh
```

8. También hay que agregar un DATACOMBO2_CHANGE:

```
Private Sub DataCombo2_Change()  
  
If Trim(DataCombo2.Text) <> "" Then txtcodciudad.Text = DataCombo2.BoundText  
  
Imgincluir.Enabled = False  
imgmodificar.Enabled = True  
imgeliminar.Enabled = True  
txtnomciudad.Text = DataCombo2.Text  
txtnomciudad.Enabled = True  
txtnomciudad.SetFocus  
  
End Sub
```

```
Private Sub DataCombo2_KeyPress(KeyAscii As Integer)  
KeyAscii = 0  
End Sub
```

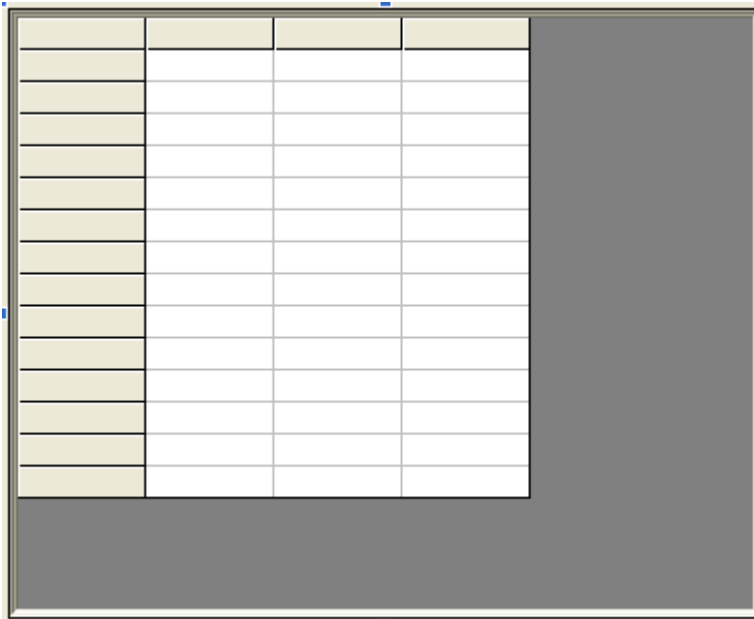
9. Colocar las rutinas de actualización:

```
Private Sub imgincluir_Click()  
sql = "Insert Into ciudades values"  
& " ('" & Trim(txtcodestado.Text) & "', '" & Trim(txtcodciudad.Text) & "', '" & Trim(txtnomciudad.Text) & "', 'A')"  
db.Execute sql, SOpt  
MsgBox "Registro Incluido ", vbExclamation, "Ciudad"  
txtnomciudad.Text = ""  
txtcodciudad.Text = ""  
txtcodestado.Text = ""  
DataCombo1.Text = ""  
DataCombo2.Text = ""  
Adodc2.Refresh  
DataCombo1.Refresh  
End Sub
```

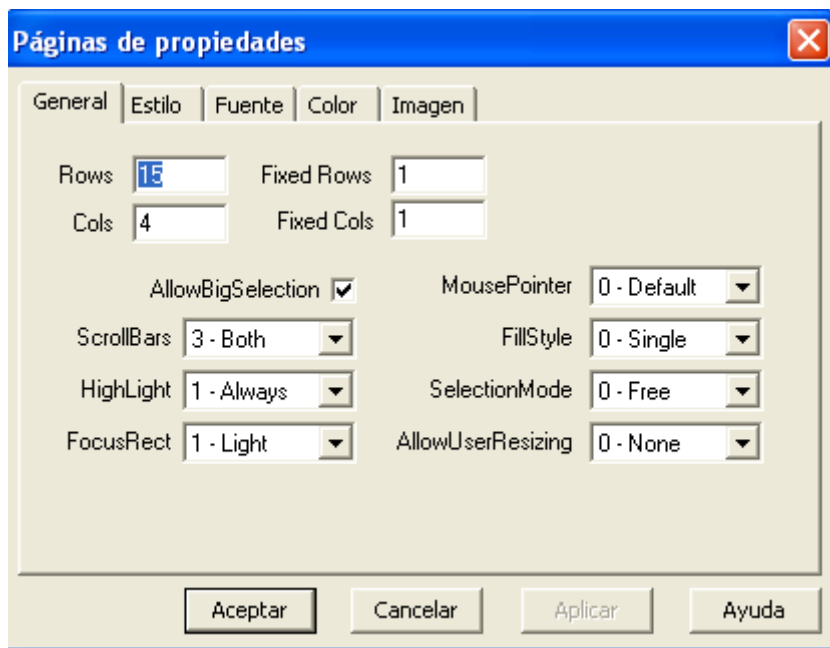
También realizar las de Modificar, Eliminar.

10. Con respecto al **Listado de Ciudades**, se hace necesario realizar un ajuste para que la salida sea mas amigable y de mas información al usuario. En este orden de ideas, se procede de la forma que sigue:

10.1 Colocar 4 Columnas al MSFlexGrid para que se vea como sigue:



Para ello, debe ir por las propiedades del MSFlexGrid e indicar que ahora hay 4 columnas.



10.2 Escribir la rutina para llenar el GRID utilizando un JOIN de las Tablas Estados y Ciudades.

```
Private Sub form_load()  
' La variable Titulo$ establece los encabezados de columna.  
  
Titulo$ = "^Código Estado|^Nombre Estado|^Código Ciudad|^Nombre de la Ciudad  "  
MSFlexGrid1.FormatString = Titulo$  
  
' Se coloca el carácter ^ en vez de < para Centrar el Título  
  
Set rs = New ADODB.Recordset  
sql = "Select e.cod_estado,nombre_estado,c.cod_ciudad,nombre_ciudad" _  
& " from estados as e,Ciudades as c" _  
& " where e.cod_estado=c.cod_estado" _  
& " and c.estatus='A'" _  
& " and e.estatus='A'" _  
& " order by e.cod_estado,cod_ciudad"  
rs.Open sql, db, , , adCmdText  
  
    If Not rs.EOF Then  
        I = 1  
        While Not rs.EOF  
            MSFlexGrid1.TextMatrix(I, 0) = rs!cod_estado  
            MSFlexGrid1.TextMatrix(I, 1) = rs!nombre_estado  
            MSFlexGrid1.TextMatrix(I, 2) = rs!cod_ciudad  
            MSFlexGrid1.TextMatrix(I, 3) = rs!nombre_ciudad  
            MSFlexGrid1.Rows = MSFlexGrid1.Rows + 1  
            I = I + 1  
            rs.MoveNext  
        Wend  
    End If  
End Sub
```

Es necesario resaltar, que en la instrucción SQL, se está utilizando un JOIN de las mencionadas Tablas. Si el estudiante tiene dudas al respecto, debe revisar el “Tema 1 Sección No. 3 Join” en donde se explica ampliamente este punto.

10.3 Ejecutar la salida:

Listado de Ciudades

Código Estado	Nombre Estado	Código Ciudad	Nombre de la Ciudad
01	Distrito Capital	01	Caracas
01	Distrito Capital	02	Baruta
01	Distrito Capital	97	Charallave
02	Zulia	01	Maracaibo
02	Zulia	03	Ciudad Ojeda
03	Lara	01	Barquisimeto
03	Lara	02	Quibor
03	Lara	05	Cabudare

