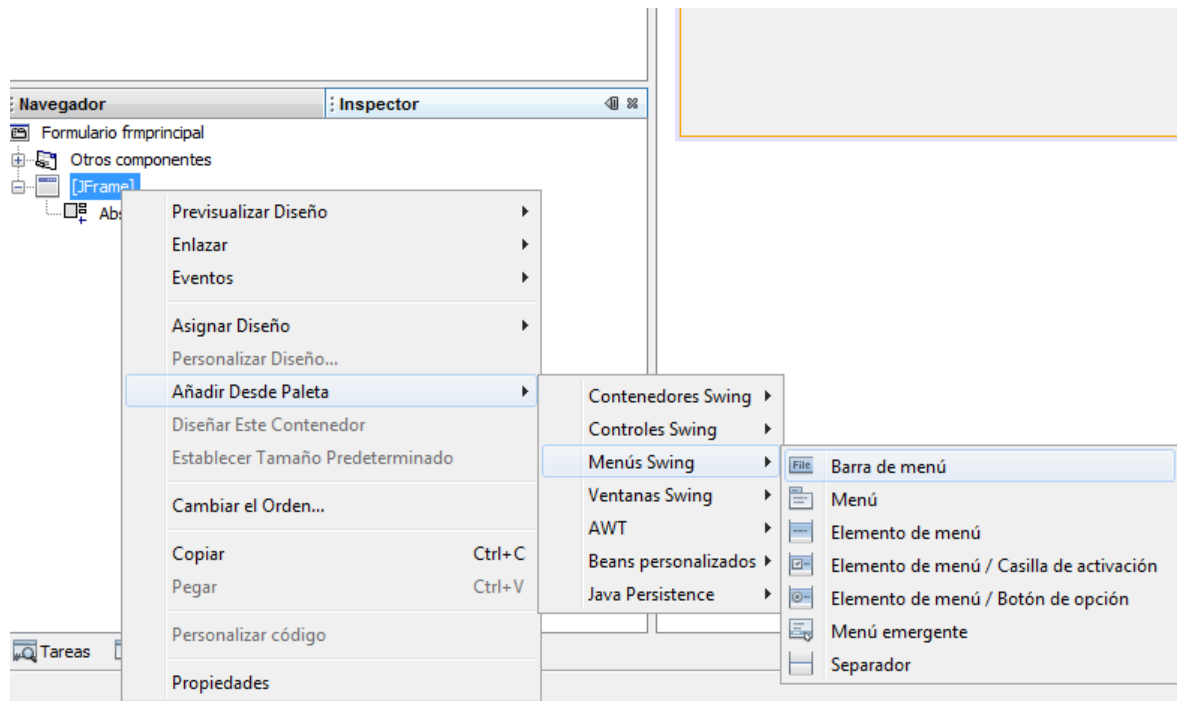


## Swing Menús y Mantenimiento de Datos Usando varias tablas.



Ahora vamos adicionar el menú principal.

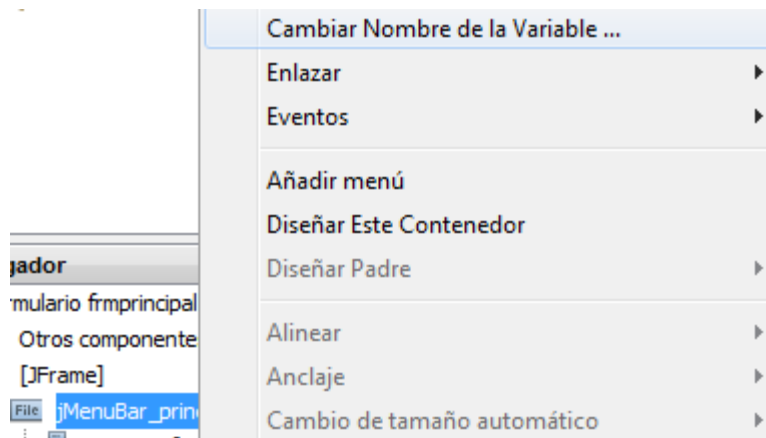
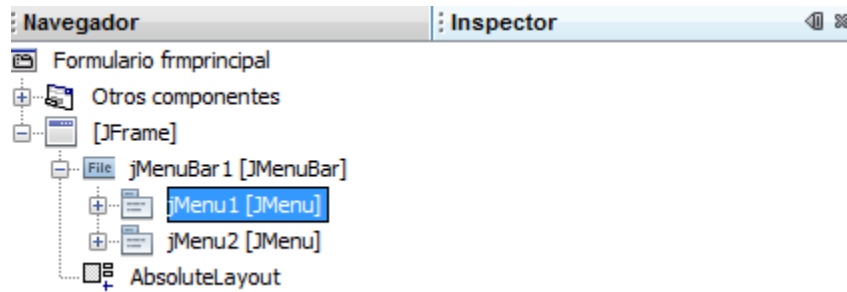
Ubicar el cursor en la etiqueta **[JFrame]**, anti clic. Se desplegará el siguiente menú flotante: en donde seleccionaremos Añadir Dese Paleta/Menús Swing/Barra de menú.



Es así como quedara el formulario.

Y en la pestaña inspector tiene que dar con la siguiente estructura:

Swing Menús y Mantenimiento de Datos Usando varias tablas.

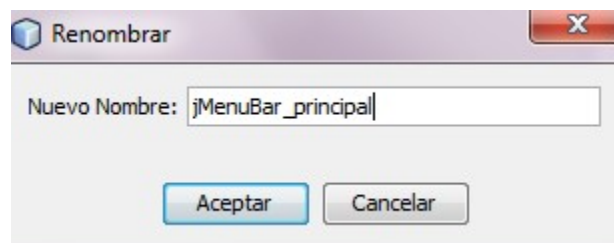


Ahora procedemos a cambiar el nombre de las variables. jMenuBar1 por jMenuBar\_principal

(anti clic en

JMenuBar1/CambiarNombre de la variable.

Se visualizará la ventana siguiente

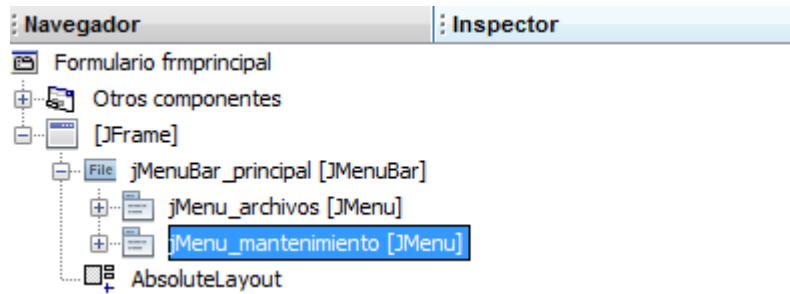


Igualmente con jMenu1 cambiamos el nombre de variable por jMenu\_archivos.

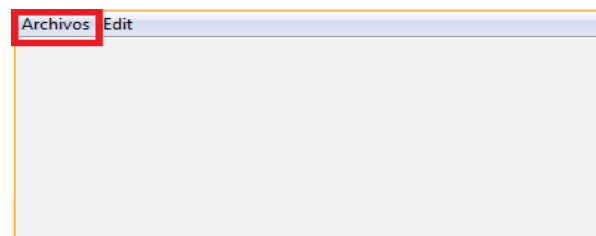
Y jMenu2 cambiamos por la variable jMenu\_mantenimiento.

Ahora se visualizará así:

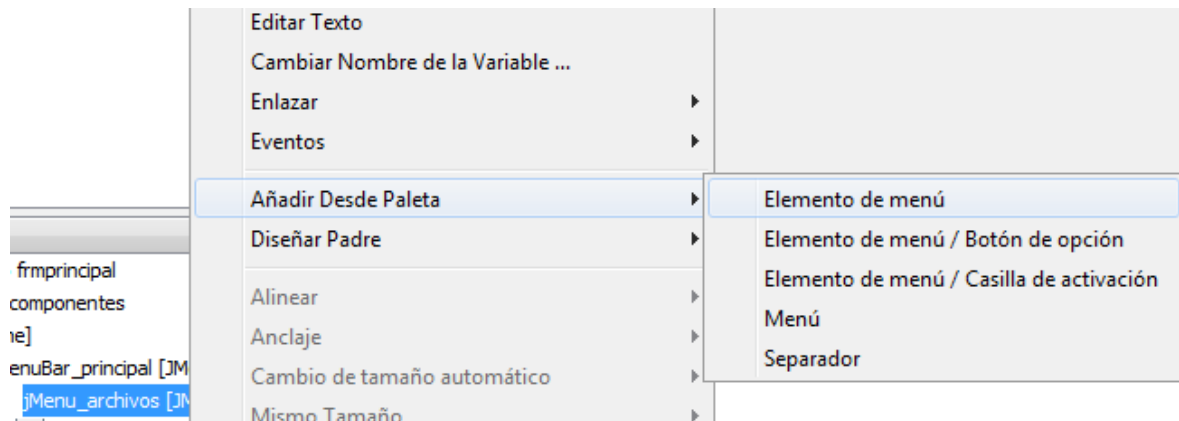
Swing Menús y Mantenimiento de Datos Usando varias tablas.



Ahora cambiaremos los nombres de la barra de menú. Nos dirigimos a propiedades. Y cambiamos el nombre en text: File por Archivos. La interfaz gráfica adopta la siguiente forma:

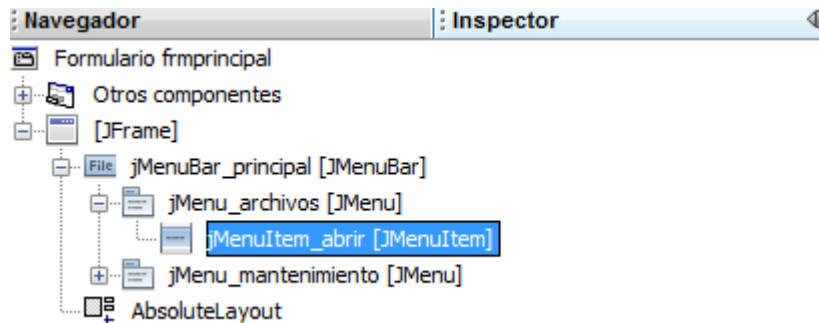


Ahora agregaremos Items a los menús. ( anti clic en jMenu\_archivos, en este caso del NetBeans 6.5 seleccionamos Añadir Desde Paleta/Elemento de menú para que aparezca el ítem.)

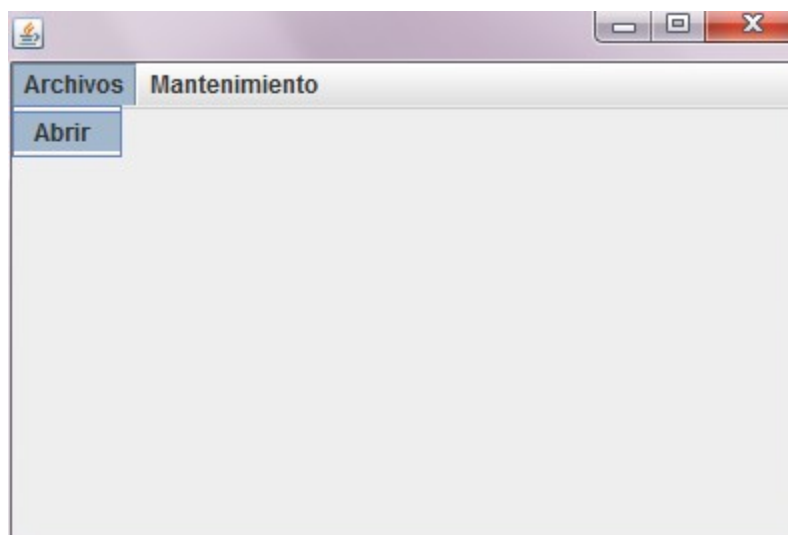


Ahora se visualizara de la siguiente forma: jMenuItem1 cambiamos el nombre de la variable por jMenuItem\_abrir.

Swing Menús y Mantenimiento de Datos Usando varias tablas.



Si procedemos a compilar y ejecutar la aplicación, vemos que al dar clic sobre el **Archivos**, se observará lo siguiente:

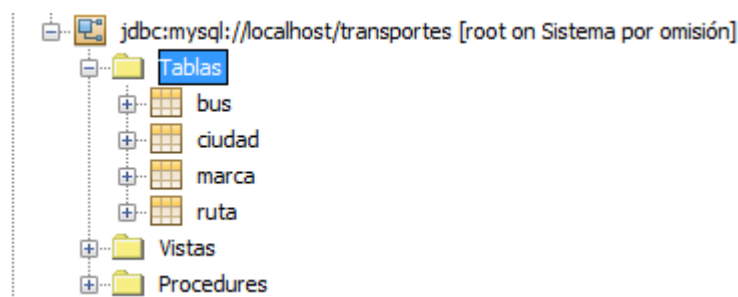


Ahora  
procedemos  
las tablas

a crear  
que

vamos a usar, que son: bus, ciudad, marca( que ya estaba creada, porque la usamos en la sesión anterior) y ruta.

La creación de tablas lo realice con el mysql, es opcional también lo pueden crear directo desde el NetBeans.



Ahora procederemos a agregar los métodos en la clase Main que utilizaremos para el mantenimiento de datos de las 4 tablas. Es el mismo que usamos en el Main

para la tabla marca en la sesión anterior, pero esta vez se agrega unos cuantos métodos más, para las nuevas tablas.

```
package transportes;
import java.io.*;
import java.sql.*;
import javax.swing.*;

public class Main {

    static Connection conn=null;
    static Statement st=null;
    static ResultSet rs = null;

    static String bd="transportes";
    static String login="root";
    static String password="123456";
    static String url="jdbc:mysql://localhost/"+bd;

    public static Connection Enlace(Connection conn) throws SQLException
    {
        try
        {
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(url,login,password);
        }
        catch(ClassNotFoundException e)
        {
            System.out.print("Clase no encontrada");
        }
        return conn;
    }
}
```

## Swing Menús y Mantenimiento de Datos Usando varias tablas.

```
}

public static Statement sta(Statement st) throws SQLException
{
    conn=Enlace(conn);
    st=conn.createStatement();
    return st;
}

public static ResultSet EnlEst(ResultSet rs) throws SQLException
{
    st=sta(st);
    rs=st.executeQuery("Select * from marca where Estado=1");
    return rs;
}
public static ResultSet EnlBus(ResultSet rs) throws SQLException
{
    st=sta(st);
    rs=st.executeQuery("Select * from bus where Estado=1");
    return rs;
}
public static ResultSet ObtenerCodMar(ResultSet rs, String nommarc) throws SQLException
{
    st=sta(st);
    String cad;
    cad="Select id_marca from marca where descripción='"+nommarc+"'";
    rs=st.executeQuery(cad);
    return rs;
}
public static ResultSet EnlCiud(ResultSet rs) throws SQLException
{
    st=sta(st);
    rs=st.executeQuery("Select * from ciudad where Estado=1");
    return rs;
}
public static ResultSet EnlRuta(ResultSet rs) throws SQLException
{
    st=sta(st);
    rs=st.executeQuery("Select * from ruta where Estado=1");
    return rs;
}
public static ResultSet ObtenerCodCiud(ResultSet rs, String nomciud) throws SQLException
{
    st=sta(st);
    String cad;
    cad="Select idciudad from ciudad where nombre='"+nomciud+"'";
    rs=st.executeQuery(cad);
    return rs;
}
```

Para obtener  
los registros  
de datos de

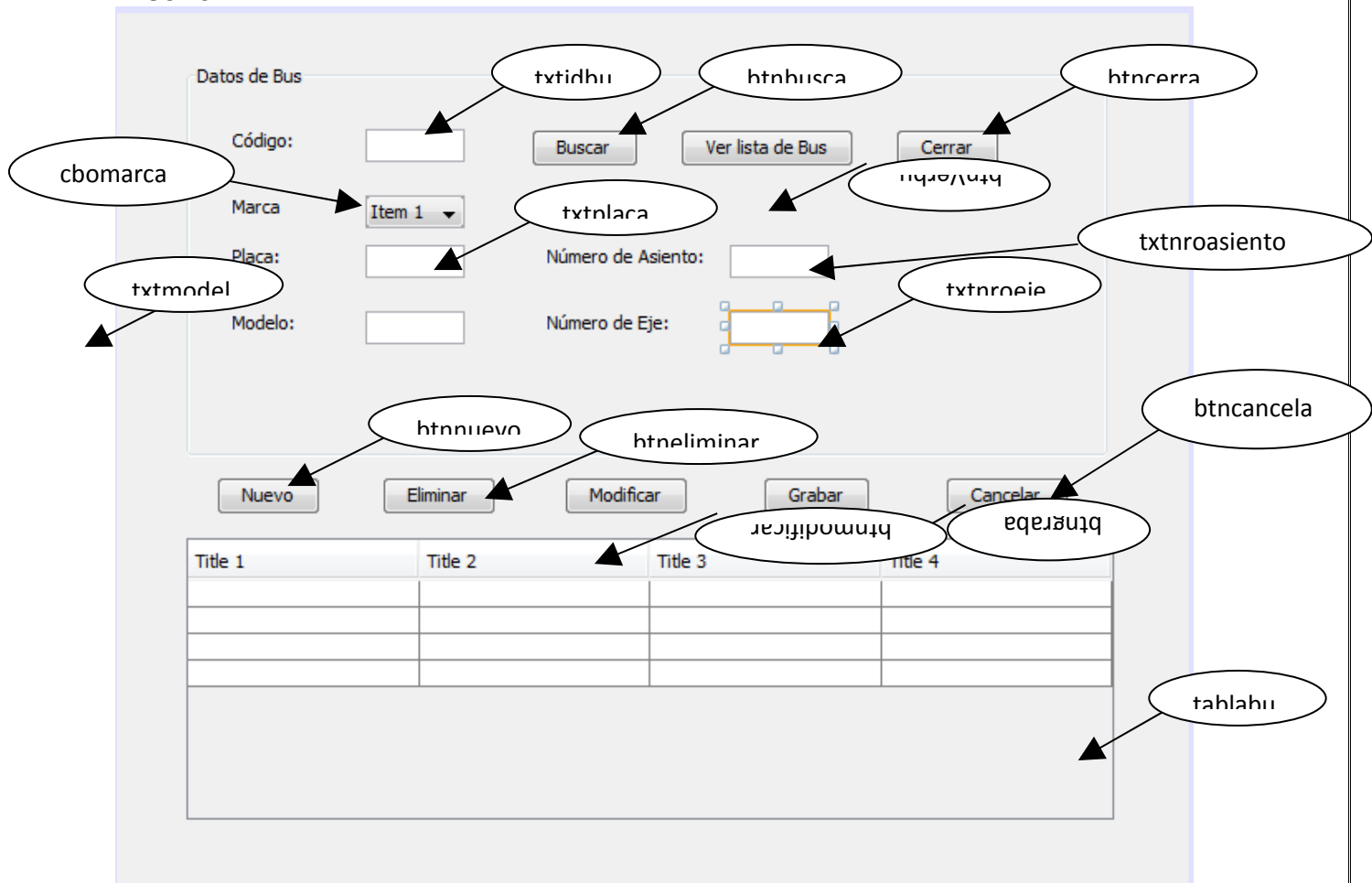
Para obtener  
los códigos  
de cada tabla

Swing Menús y Mantenimiento de Datos Usando varias tablas.

Ahora vamos a la parte de la codificación del formulario frmMantBus. ( el formulario marca: frmMantMarca ya fue realizado en la sesión anterior.

Como ya habíamos indicado en la sesión anterior los nombres de variables.

### Diseño



### Fuente

Agregar los paquetes antes de la definición de la clase frmMantBus

```
package transportes;  
  
import java.io.*;  
import java.sql.*;  
import javax.swing.*;  
import javax.swing.table.*;  
import transportes.Main.*;
```

**Luego:** Definir los atributos o variables a usar en los distintos métodos que forma la clase frmMantBus.

## Swing Menús y Mantenimiento de Datos Usando varias tablas.

```
* @author Randy
*/
public class frmMantBus extends javax.swing.JFrame {

    static Connection conn = null;
    static Statement st = null;
    static ResultSet rs = null;
    static ResultSet rs1 = null;
    DefaultTableModel dtm = new DefaultTableModel();

    public frmMantBus() {
        initComponents();
        activaBotones(true,false,false,false);
        String titulos[]={"idbus","id_marca","placa","modelo","nroasientos","nroejes","Estado"};
        dtm.setColumnIdentifiers(titulos);
        tablabus.setModel(dtm);
        setSize(700,600);
        setLocation(200,200);

        try
        {
            conn= Main.Enlace(conn);
            rs=Main.EnlEst(rs);
            while(rs.next());
            {
                cbomarca.addItem((String) rs.getString(2));
            }
        }
        catch(SQLException e)

        {
            JOptionPane.showMessageDialog(null,"Error BD"+e.getMessage());
        }
        activaBotones(true,false,false,false);
    }

    public void activaBotones(boolean n, boolean e, boolean m, boolean g)
    {
        btnnuevo.setEnabled(n);
        btnEliminar.setEnabled(e);
        btnmodificar.setEnabled(m);
        btngrabar.setEnabled(g);
    }

    public void limpiarObjetos()
    {
        txtidbus.setText("");
        cbomarca.setEnabled(true);
        cbomarca.setSelectedIndex(0);
        txtmodelo.setText("");
        txtplaca.setText("");
        txtnroasiento.setText("");
        txtnroeje.setText("");
    }

    public void llenartabla()
    {
        try
        {
```

Método constructo

Construimos los métodos activaBotones y limpiarObjetos()



Se tienen el método `llenarTabla()` para llenar de datos en la filas del objeto `JTable` denominado `tablabus`.

```
int f,i;
conn=Main.Enlace(conn);
rs=Main.EnlBus(rs);
String datos[]= new String[6];
f=dtm.getRowCount();
if (f>0)
    for(i=0;i<f;i++)
        dtm.removeRow(0);
    while(rs.next())
    {
        datos[0]=(String) rs.getString(1);
        datos[1]=(String) rs.getString(2);
        datos[2]=(String) rs.getString(3);
        datos[3]=(String) rs.getString(4);
        datos[4]=(String) rs.getString(5);
        datos[5]=(String) rs.getString(6);
    }
}
catch(SQLException e)
{
    JOptionPane.showMessageDialog(null,"Error BD"+e.getMessage());
}
- }
```

A continuación insertaremos el siguiente código en el botón Buscar del formulario Bus.

```
private void btnbuscarActionPerformed(java.awt.event.ActionEvent evt) {
String b;
if(btnnuevo.isEnabled())
{
    try
    {
        conn=Main.Enlace(conn);
        rs=Main.EnlBus(rs);
        b=txtidbus.getText();
        String pos;
        boolean encuentra=false;
        while(rs.next())
        {
            if(b.equals(rs.getString(1)))
            {
                rs1=Main.ObtenerCodMar(rs1, (rs.getString(2).toString()));
                rs1.next();
                pos=(String) rs1.getString(1);
                cbomarca.setSelectedItem(pos);
                txtplaca.setText((String) rs.getString(3));
                txtmodelo.setText((String) rs.getString(4));
                txtnroasiento.setText((String) rs.getString(5));
                txtnroejec.setText((String) rs.getString(6));
                this.activaBotones(true, true, true, false);
                encuentra=true;
                break;
            }
        }
    }
}
```

Swing Menús y Mantenimiento de Datos Usando varias tablas.

```
        if(encuentra==false)
        {
            txtidbus.setText("No existe");
            txtidbus.requestFocus();
        }
    }
    catch(SQLException e)
    {
        JOptionPane.showMessageDialog(null,"Error BD"+e.getMessage());
    }
}
}
```

Ahora seguimos con el botón Ver lista de Bus:

```
try
{
    this.setSize(700,600);
    int f, i;
    conn=Main.Enlace(conn);
    rs=Main.EnlEst(rs);
    String datos[]=new String[6];
    f=dtm.getRowCount();
    if(f>0)
        for(i=0; i<f; i++)
            dtm.removeRow(0);
    while(rs.next())
    {
        datos[0]=(String) rs.getString(1);
        datos[1]=(String) rs.getString(2);
        datos[2]=(String) rs.getString(3);
        datos[3]=(String) rs.getString(4);
        datos[4]=(String) rs.getString(5);
        datos[5]=(String) rs.getString(6);
        dtm.addRow(datos);
    }
}
catch(SQLException e)
{
    JOptionPane.showMessageDialog(null,"Error BD"+e.toString());
}
}
```

Swing Menús y Mantenimiento de Datos Usando varias tablas.

Botón cerrar:

```
private void btncerrarActionPerformed(java.awt.event.ActionEvent evt) {  
    dispose();  
}
```

Botón Nuevo:

```
private void btnnuevoActionPerformed(java.awt.event.ActionEvent evt) {  
    limpiarObjetos();  
    txtidbus.setEnabled(false);  
    txtidbus.requestFocus();  
    activaBotones(false, false, false, true);  
}
```

Botón Eliminar:

```
private void btnEliminarActionPerformed(java.awt.event.ActionEvent evt) {  
    int resp;  
    resp=JOptionPane.showConfirmDialog(null,"¿Desea Eliminar el registro?", "Pregunta",0);  
    if(resp==0)  
    {  
        try  
        {  
            conn=Main.Enlace(conn);  
            st=Main.sta(st);  
            rs=Main.EnlBus(rs);  
            String cod, comando;  
            cod=txtidbus.getText();  
            comando="DELETE FROM bus where idbus="+String.valueOf(cod);  
            st.execute(comando);  
            this.llenartabla();  
            conn.close();  
            activaBotones(true, false, false, false);  
            limpiarObjetos();  
            txtidbus.setEnabled(true);  
        }  
        catch (SQLException e)  
        {  
            JOptionPane.showMessageDialog(null,"Error "+e.toString());  
        }  
    }  
}
```

### Botón Modificar:

```
private void btnmodificarActionPerformed(java.awt.event.ActionEvent evt) {  
    int resp;  
    resp=JOptionPane.showConfirmDialog(null,"¿Desea Modificar los datos?", "Pregunta",0);  
    if(resp==0)  
    {  
        try  
        {  
            conn=Main.Enlace(conn);  
            st=Main.sta(st);  
            rs=Main.EnlBus(rs);  
  
            String cod, nommar=null, codmarca=null, placa, modelo, nroasien, nroeje, comando=null;  
            cod=txtidbus.getText();  
            nommar=(String)cbomarca.getSelectedItem();  
            rs1=Main.ObtenerCodMar(rs1, nommar);  
            if(rs1.next())  
            {  
                codmarca=rs1.getString(1);  
            }  
            else  
            {  
                JOptionPane.showMessageDialog(null,"Seleccione una Marca");  
            }  
            placa=txtplaca.getText();  
            modelo=txtmodelo.getText();  
            nroasien=txtnroasiento.getText();  
            nroeje=txtnroeje.getText();  
  
            comando="UPDATE bus SET id_marca='"+codmarca+"', placa='"+placa+"', modelo='"+modelo+"',";  
            st.executeUpdate(comando);  
            this.llenartabla();  
            conn.close();  
            activaBotones(true,false,false,false);  
            limpiarObjetos();  
        }  
        catch(SQLException e)  
        {  
            JOptionPane.showMessageDialog(null,"Error "+e.toString());  
        }  
    }  
}
```

### Botón Grabar:

## Swing Menús y Mantenimiento de Datos Usando varias tablas.

```
private void btngrabarActionPerformed(java.awt.event.ActionEvent evt) {  
    int resp;  
    resp=JOptionPane.showConfirmDialog(null,"¿Desea Grabar el registro?", "Pregunta",0);  
    if(resp==0)  
    {  
        try  
        {  
            conn=Main.Enlace(conn);  
            st=Main.sta(st);  
            rs=Main.EnlBus(rs);  
            String cod, nommar=null, codmarca=null, placa, modelo, nroasien, nroeje, comando;  
            cod=txtidbus.getText();  
            nommar=(String)cbomarca.getSelectedItem();  
            rs1=Main.ObtenerCodMar(rs1, nommar);  
            if(rs1.next())  
            {  
                codmarca=rs1.getString(1);  
            }  
            else  
            {  
                JOptionPane.showMessageDialog(null,"Seleccione una Marca");  
            }  
            placa=txtplaca.getText();  
            modelo=txtmodelo.getText();  
            nroasien=txtnroasiento.getText();  
            nroeje=txtnroeje.getText();  
  
            comando="INSERT INTO bus (idbus, id_marca, placa, modelo, nroasientos, nroejes, Estado) VALUES";  
  
            comando=comando+"('"+cod+"','"+codmarca+"','"+placa+"','"+modelo+"','"+nroasien+"','"+nroeje+"',1)";  
            st.executeUpdate(comando);  
            this.llenartabla();  
            conn.close();  
            activaBotones(true,false,false,false);  
            limpiarObjetos();  
            txtidbus.setEnabled(true);  
        }  
        catch(SQLException e)  
        {  
            JOptionPane.showMessageDialog(null,"Error "+e.toString());  
        }  
    }  
}
```

Botón Cancelar:

Swing Menús y Mantenimiento de Datos Usando varias tablas.

```
private void btncancelarActionPerformed(java.awt.event.ActionEvent evt) {  
    int resp;  
    resp=JOptionPane.showConfirmDialog(null,"¿Desea Cancelar el proceso?", "Pregunta",0);  
    if (resp==0)  
    {  
        limpiarObjetos();  
        txtidbus.setEnabled(true);  
        activaBotones(true,false,false,false);  
    }  
}
```

A continuación visualizamos el formulario de la tabla ciudad: frmMantCiudad

**Diseño:**

Title 1	Title 2	Title 3	Title 4

**Fuente:**

```
package transportes;  
import java.io.*;  
import java.sql.*;  
import javax.swing.*;  
import javax.swing.table.*;  
import transportes.Main.*;
```

**Seguimos el mismo proceso que usamos en el formulario bus.**

Swing Menús y Mantenimiento de Datos Usando varias tablas.

```
public class frmMantCiudad extends javax.swing.JFrame {
    static Connection conn=null;
    static Statement st=null;
    static ResultSet rs=null;
    DefaultTableModel dtm=new DefaultTableModel();

    /** Creates new form frmMantCiudad */
    public frmMantCiudad() {
        initComponents();
        setSize(530,290);
        setLocation(200,200);
        activaBotones(true,false,false,false);
        String titulos[]={"Codigo","Nombre"};
        dtm.setColumnIdentifiers(titulos);
        tablaciudad.setModel(dtm);
    }
    public void activaBotones(boolean n, boolean e, boolean m, boolean g)
    {
        btnnuevo.setText("");
        btneliminar.setText("");
        btnmodificar.setText("");
        btngrabar.setText("");
    }
    public void limpiarObjetos()
    {
        txtidciudad.setText("");
        txtnombre.setText("");
    }
}
```

Ahora vamos al botón buscar y realizamos el siguiente código



```
private void btnbuscarActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    String b;  
    if (btnbuscar.isEnabled())  
    {  
        try  
        {  
            conn=Main.Enlace(conn);  
            rs=Main.EnlCiud(rs);  
            b=txtidciudad.getText();  
            boolean encuentra=false;  
            while (rs.next())  
            {  
                if (b.equals(rs.getString(1)))  
                {  
                    txtnombre.setText((String) rs.getString(2));  
                    this.activaBotones(true, true, true, false);  
                    encuentra=true;  
                    break;  
                }  
            }  
            if (encuentra==false)  
            {  
                txtidciudad.setText("No existe");  
                txtidciudad.requestFocus();  
            }  
        }  
        catch(SQLException e)  
        {  
            {  
                JOptionPane.showMessageDialog(null, "Error BD"+e.getMessage());  
            }  
        }  
    }  
}
```



Swing Menús y Mantenimiento de Datos Usando varias tablas.

Ver lista Ciudad:

```
private void btnVerCiudadActionPerformed(java.awt.event.ActionEvent evt)
// TODO add your handling code here:
try
{
    this.setSize(700,600);
    int f, i;
    conn=Main.Enlace(conn);
    rs=Main.EnlEst(rs);
    String datos[]=new String[2];
    f=dtm.getRowCount();
    if(f>0)
        for(i=0; i<f; i++)
            dtm.removeRow(0);
    while(rs.next())
    {
        datos[0]=(String) rs.getString(1);
        datos[1]=(String) rs.getString(2);
        dtm.addRow(datos);
    }
}
catch(SQLException e)
{
    JOptionPane.showMessageDialog(null,"Error BD"+e.toString());
}
}
```

Botón nuevo:

```
private void btnnuevoActionPerformed(java.awt.event.ActionEvent evt) {
    limpiarObjetos();
    txtidciudad.requestFocus();
    activaBotones(false, false, false, true);
}
```

Eliminar: tabla ciudad

Swing Menús y Mantenimiento de Datos Usando varias tablas.

```
private void btneliminarActionPerformed(java.awt.event.ActionEvent evt) {  
    int resp;  
    resp=JOptionPane.showConfirmDialog(null,"¿Desea Eliminar el registro?", "Pregunta",0);  
    if(resp==0)  
    {  
        try  
        {  
            conn=Main.Enlace(conn);  
            st=Main.sta(st);  
            rs=Main.EnlCiud(rs);  
            String cod, comando;  
            cod=txtidciudad.getText();  
            comando="DELETE FROM ciudad where idciudad="+String.valueOf(cod);  
            st.execute(comando);  
            conn.close();  
            activaBotones(true,false,false,false);  
            limpiarObjetos();  
            txtidciudad.setEnabled(true);  
        }  
        catch(SQLException e)  
        {  
            JOptionPane.showMessageDialog(null,"Error "+e.toString());  
        }  
    }  
}
```

Modificar:

```
private void btnmodificarActionPerformed(java.awt.event.ActionEvent evt) {  
    int resp;  
    resp=JOptionPane.showConfirmDialog(null,"¿Desea Modificar los datos?", "Pregunta",0);  
    if(resp==0)  
    {  
        try  
        {  
            conn=Main.Enlace(conn);  
            st=Main.sta(st);  
            rs=Main.EnlBus(rs);  
  
            String cod, nom, comando;  
            cod=txtidciudad.getText();  
            nom=txtnombre.getText();  
            comando="UPDATE ciudad SET nombreciud='"+nom+"'";  
            comando=comando+"where idciudad="+String.valueOf(cod);  
            st.executeUpdate(comando);  
            conn.close();  
            activaBotones(true,false,false,false);  
            limpiarObjetos();  
        }  
        catch(SQLException e)  
        {  
            JOptionPane.showMessageDialog(null,"Error "+e.toString());  
        }  
    }  
}
```

Grabar: tabla ciudad

Swing Menús y Mantenimiento de Datos Usando varias tablas.

```
private void btngrabarActionPerformed(java.awt.event.ActionEvent evt) {  
    int resp;  
    resp=JOptionPane.showConfirmDialog(null,"¿Desea Grabar el registro?", "Pregunta",0);  
    if(resp==0)  
    {  
        try  
        {  
            conn=Main.Enlace(conn);  
            st=Main.sta(st);  
            rs=Main.EnlCiud(rs);  
            String cod, nom, comando;  
            cod=txtidciudad.getText();  
            nom=txtnombre.getText();  
            comando="INSERT INTO ciudad (idciudad, nombreciudad, Estado) VALUES";  
            comando=comando+"('"+cod+"', '"+nom+"',1)";  
            conn.close();  
            activaBotones(true,false,false,false);  
            limpiarObjetos();  
            txtidciudad.setEnabled(true);  
        }  
        catch(SQLException e)  
        {  
            JOptionPane.showMessageDialog(null,"Error "+e.toString());  
        }  
    }  
}
```

Y por último cancelar:

```
private void btncancelarActionPerformed(java.awt.event.ActionEvent evt) {  
    int resp;  
    resp=JOptionPane.showConfirmDialog(null,"¿Desea Cancelar el proceso?", "Pregunta",0);  
    if(resp==0)  
    {  
        limpiarObjetos();  
        txtidciudad.setEnabled(true);  
        activaBotones(true,false,false,false);  
    }  
}
```

A continuación el formulario de la tabla ruta: frmMantRuta

**Diseño:**

## Swing Menús y Mantenimiento de Datos Usando varias tablas.

Datos de Ruta

Código:

Buscar

Ver lista de Ruta

Cerrar

Ciudad de Origen: 

Item 1 ▾

Distancia:

Ciudad Destino: 

Item 1 ▾

Horas de Viaje:

Nuevo

Eliminar

Modificar

Grabar

Cancelar

Title 1	Title 2	Title 3	Title 4

### Fuente:

**Algunos botones que no pondré, son iguales a los anteriores solo cambiar las variables de cada tabla a la que pertenecen. Pondré los más tediosos y que a veces puede haber fallas si no están bien codificados.**



### Eliminar:

## Swing Menús y Mantenimiento de Datos Usando varias tablas.

```
private void btneliminarActionPerformed(java.awt.event.ActionEvent evt) {  
    int resp;  
    resp=JOptionPane.showConfirmDialog(null,"¿Desea Eliminar el registro?", "Pregunta",0);  
    if(resp==0)  
    {  
        try  
        {  
            conn=Main.Enlace(conn);  
            st=Main.sta(st);  
            rs=Main.EnlRuta(rs);  
            String cod, comando;  
            cod=txtidruta.getText();  
            comando="DELETE FROM ruta where idruta="+String.valueOf(cod);  
            st.execute(comando);  
            this.llenartabla();  
            conn.close();  
            activaBotones(true,false,false,false);  
            limpiarObjetos();  
            txtidruta.setEnabled(true);  
        }  
        catch(SQLException e)  
        {  
            JOptionPane.showMessageDialog(null,"Error "+e.toString());  
        }  
    }  
}
```

## Modificar:

```
private void btnmodificarActionPerformed(java.awt.event.ActionEvent evt) {  
    int resp;  
    resp=JOptionPane.showConfirmDialog(null,"¿Desea Modificar los datos?", "Pregunta",0);  
    if(resp==0)  
    {  
        try  
        {  
            conn=Main.Enlace(conn);  
            st=Main.sta(st);  
            rs=Main.EnlRuta(rs);  
  
            String cod, nomciuori=null, codciuori=null, nomciudes=null, codciudes=null, distan, horav, comando;  
            cod=txtidruta.getText();  
            nomciuori=(String)cbociudorig.getSelectedItem();  
            rs1=Main.ObtenerCodMar(rs1, nomciuori);  
            if(rs1.next())  
            {  
                codciuori=rs1.getString(1);  
            }  
            else  
            {  
                JOptionPane.showMessageDialog(null,"Seleccione una Ciudad");  
            }  
  
            nomciudes=(String)cbociudesti.getSelectedItem();  
            rs2=Main.ObtenerCodMar(rs2, nomciuori);  
            if(rs2.next())
```

## Swing Menús y Mantenimiento de Datos Usando varias tablas.

```
--(-----),,
{
    codciuori=rs2.getString(1);
}
else
{
    JOptionPane.showMessageDialog(null,"Seleccione una Ciudad");
}
distan=txtdistancia.getText();
horav=txthoras.getText();

comando="UPDATE ruta SET idciuori='"+codciuori+"', idciudes='"+codciudes+"', distancia='"+distan+"','";
comando=comando+"where idruta="+String.valueOf(cod);
st.executeUpdate(comando);
this.llenartabla();
conn.close();
activaBotones(true,false,false,false);
limpiarObjetos();
}
catch(SQLException e)
{
    JOptionPane.showMessageDialog(null,"Error "+e.toString());
}
}
}
```

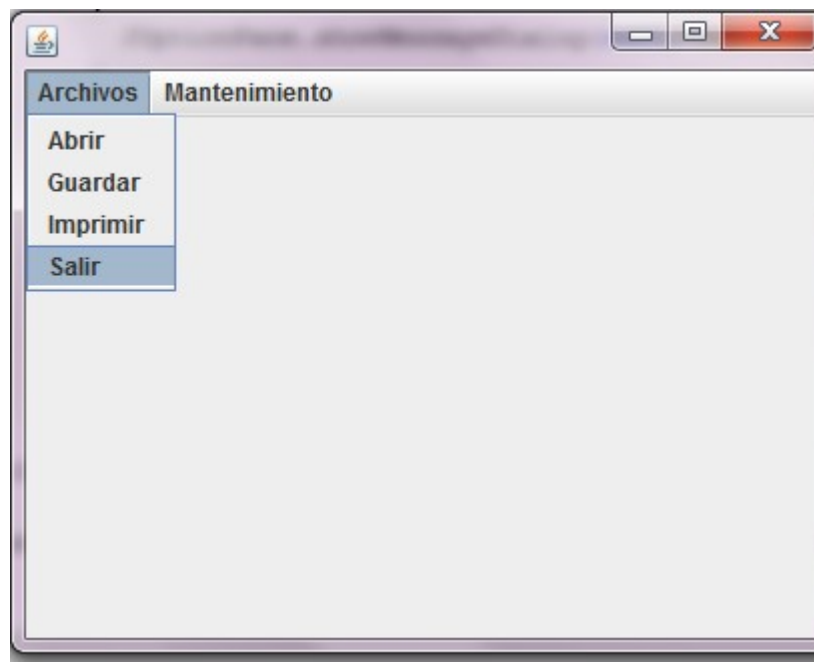
## Grabar:

```
private void btngrabarActionPerformed(java.awt.event.ActionEvent evt) {
    int resp;
    resp=JOptionPane.showConfirmDialog(null,"¿Deseas grabar el requisito?", "Pregunta",0);
    if (resp==0)
    {
        try
        {
            conn=Main.Enlace(conn);
            st=Main.sta(st);
            rs=Main.EnlRuta(rs);
            String cod, nomciuori=null, codciuori=null, nomciudes=null, codciudes=null, distan, horav, comando;
            cod=txtidruta.getText();
            nomciuori=(String)cbociudorig.getSelectedItem();
            rs1=Main.ObtenerCodMar(rs1, nomciuori);
            if(rs1.next())
            {
                codciuori=rs1.getString(1);
            }
            else
            {
                JOptionPane.showMessageDialog(null,"Seleccione una Ciudad");
            }
            nomciudes=(String)cbociudesti.getSelectedItem();
            rs1=Main.ObtenerCodMar(rs2, nomciuori);
            if(rs2.next())
            {
                codciuori=rs2.getString(1);
            }
        }
    }
}
```

Swing Menús y Mantenimiento de Datos Usando varias tablas.

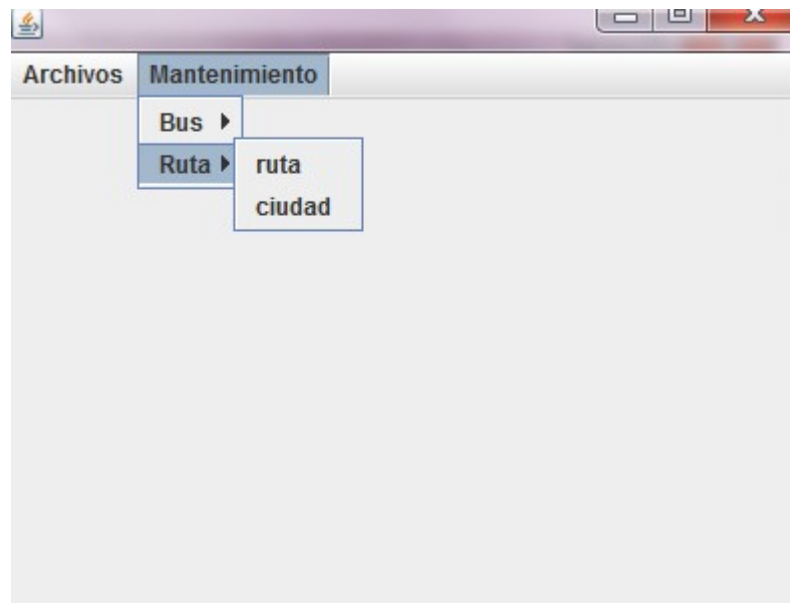
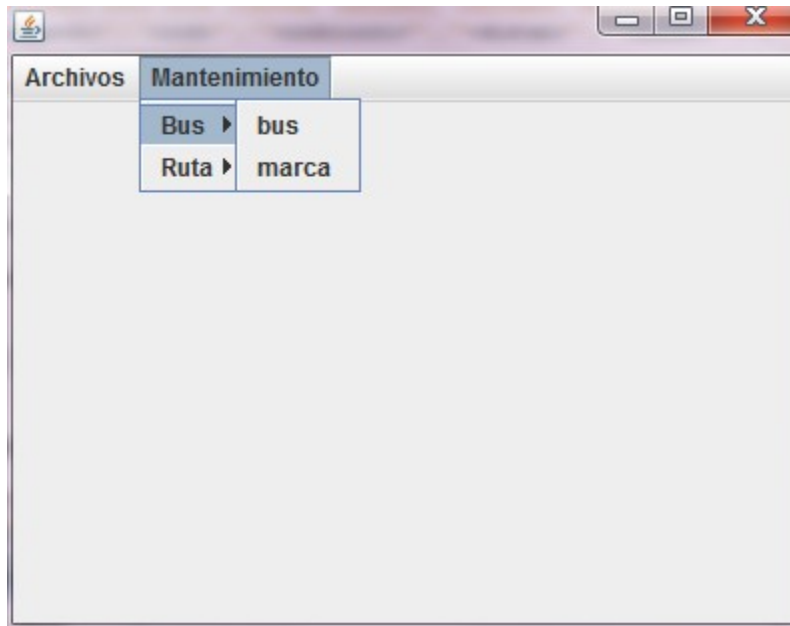
```
    }
    else
    {
        JOptionPane.showMessageDialog(null, "Seleccione una Ciudad");
    }
    distan=txtdistancia.getText();
    horav=txthoras.getText();
    comando="Insert into ruta (idruta, idciuori, idciudes, distancia, horas_viaje) Values";
    comando=comando+"('"+cod+"', '"+codciuori+"', '"+distan+"', '"+horav+"')";
    st.executeUpdate(comando);
    conn.close();
    activaBotones(true, false, false, false);
    limpiarObjetos();
    txtidruta.setEnabled(true);
}
catch(SQLException e)
{
    JOptionPane.showMessageDialog(null, "Error"+e.toString());
}
}
```

Entonces procederemos a ejecutar el formulario principal: frmprincipal.



Podemos apreciar los menús que agregamos al comienzo.

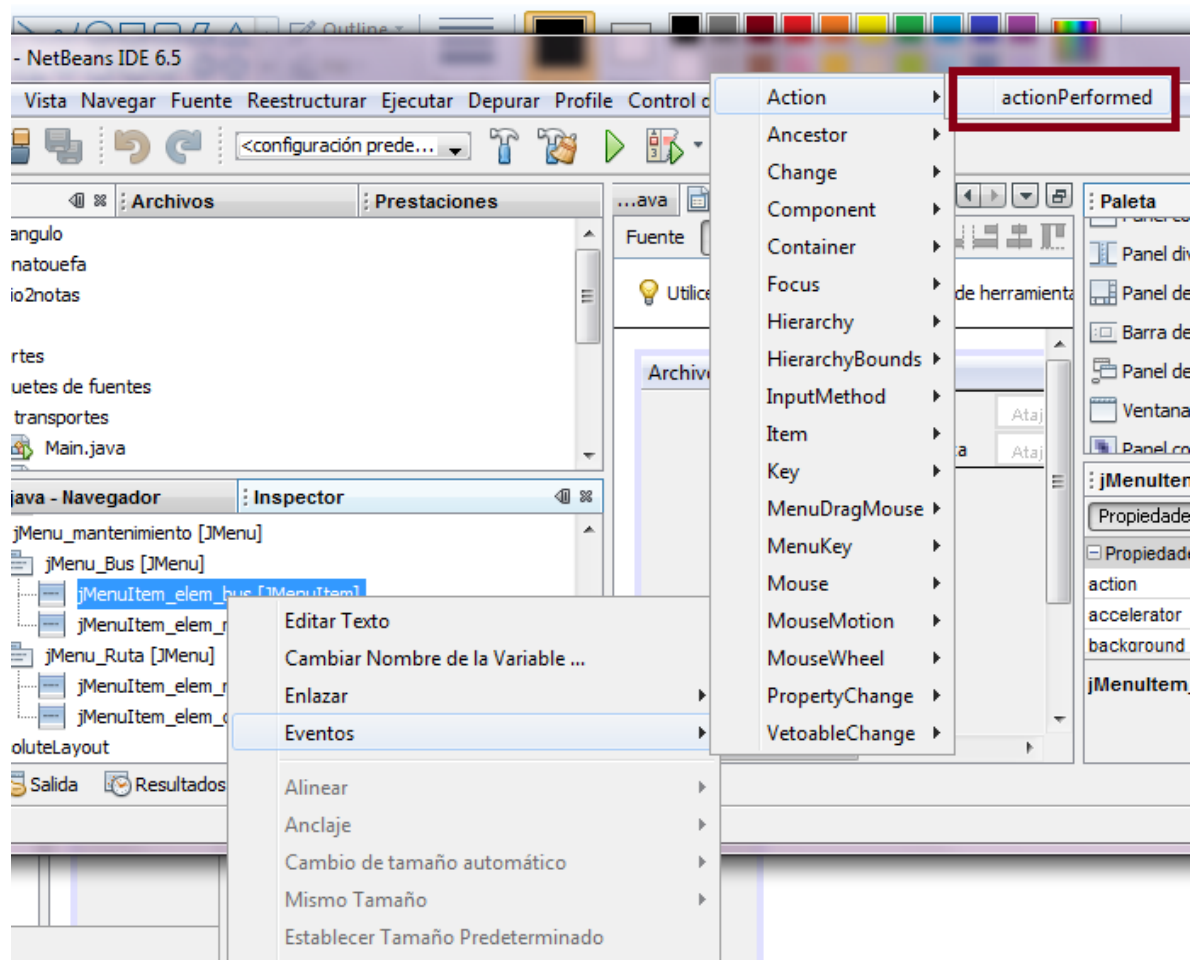
Swing Menús y Mantenimiento de Datos Usando varias tablas.



**Nota:** Para ello es necesario agregar programación en *frmprincipal*. Por ejemplo si queremos ejecutar el formulario referido a Mantenimiento de bus cuyo formulario se denomina *frmMantBus*, tendríamos que seleccionar el objeto **JMenuItem** correspondiente para indicar el evento a programar.



Swing Menús y Mantenimiento de Datos Usando varias tablas.



A continuación, debemos colocar la siguiente línea de programación:

```
private void jMenuItem_elem_busActionPerformed(java.awt.event.ActionEvent evt) {  
    new frmMantBus().setVisible(true);  
}
```

Instanciamos con **new** un objeto del tipo `frmMantBus()` y procedemos a mostrarlo con el método `setVisible(true)`.

Debemos tener claro que un formulario cuando es invocado desde un menú al momento de ser cerrado a través de un botón de comando que puede llamarse Cerrar o Salir ya no deberíamos usar el método `dispose()`, sino más bien `hide()`. Este método `hide()` solo oculta el formulario actual y no cierra al formulario que contiene el menú. Además es necesario establecer los valores de los métodos **`setSize()`** y **`setLocation()`** del `JFrame` para hacer que la ejecución de los formularios al momento de ser visualizados sea la más adecuada.