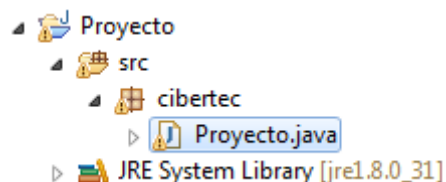


## ¿Cómo crear un menú en Java con Eclipse?

Año 2019

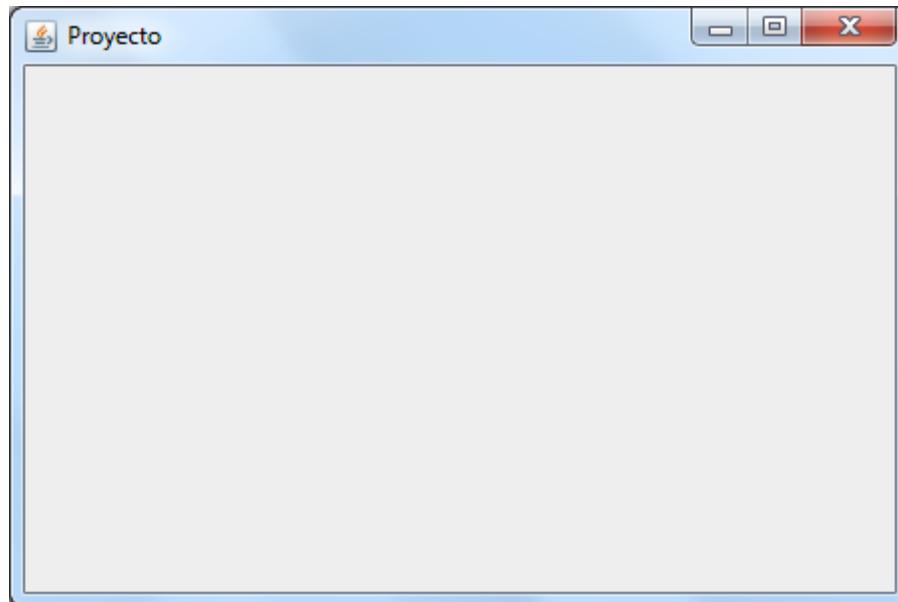
### Cree un Proyecto y añada un JFrame

- Seleccione **File / New / Java Project**
- Project name: **Proyecto** *(debe comenzar siempre con letra mayúscula)*
- Ilumine **src**
- Clic derecho y Seleccione **New / Other**
- En **WindowBuilder / SwingDesigner** seleccione **JFrame**
- Package: **cibertec** *(debe comenzar siempre con letra minúscula)*
- Name: **Proyecto** *(debe comenzar siempre con letra mayúscula)*

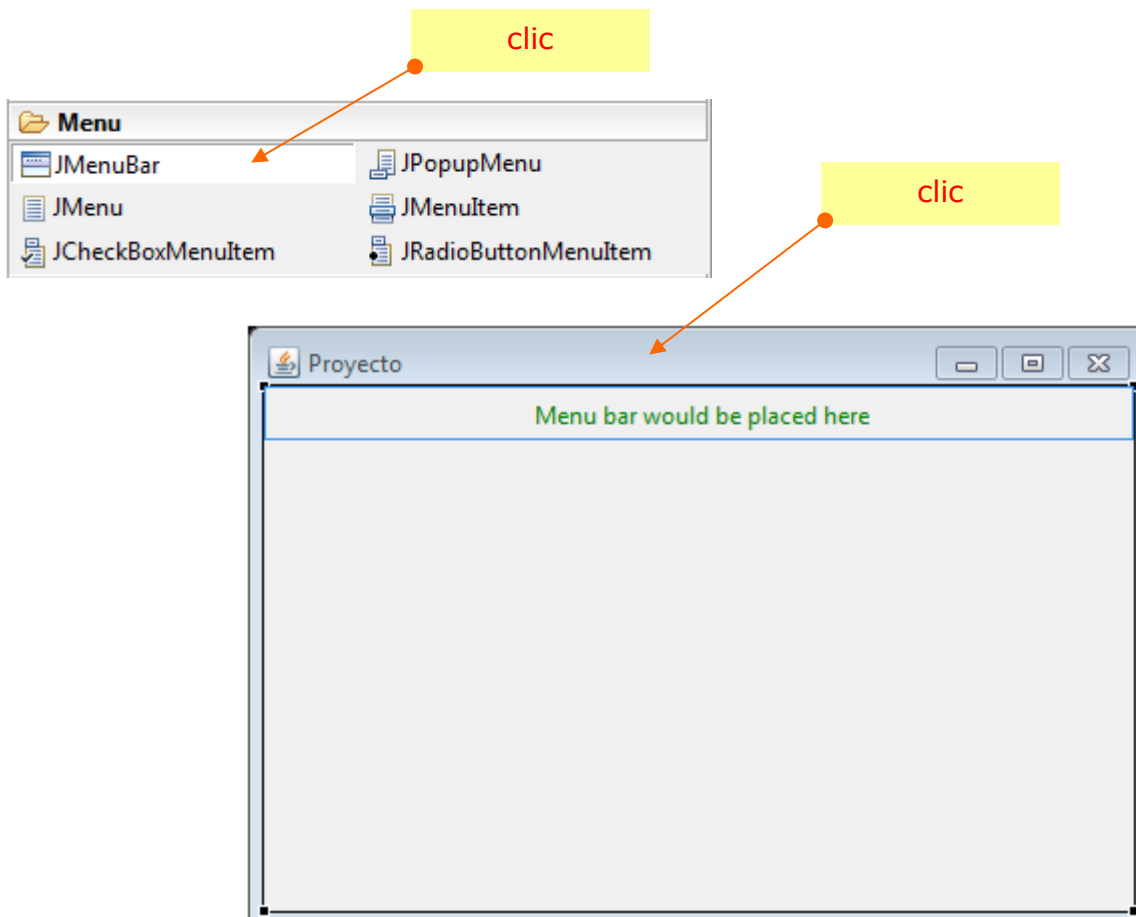


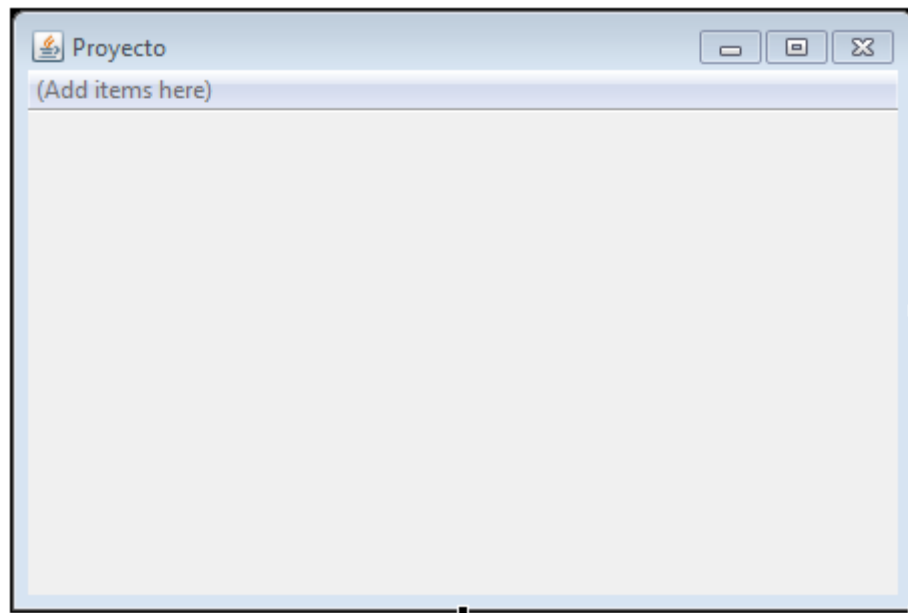
- Cambie el tamaño del Proyecto a 800 x 600
- ```
public Proyecto() {  
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
    setBounds(100, 100, 800, 600);  
    ...  
}
```

- Clic izquierdo en la pestaña Design
- Ubíquese en el interior del Componente Visual
- Haga clic derecho y seleccione **Set layout** / **Absolute layout**
- Ilumine el marco del Componente Visual
- title: **Proyecto**
- Grabe todo y ejecute el Proyecto

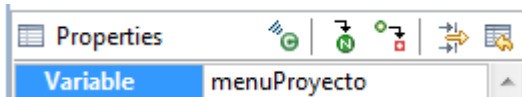


## Cree una barra de menú (JMenuBar) en el Proyecto

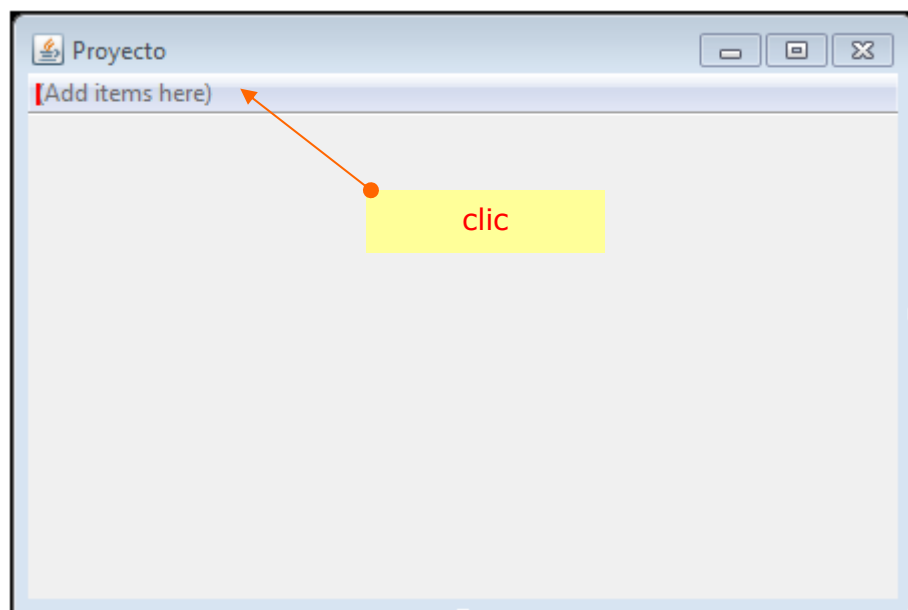
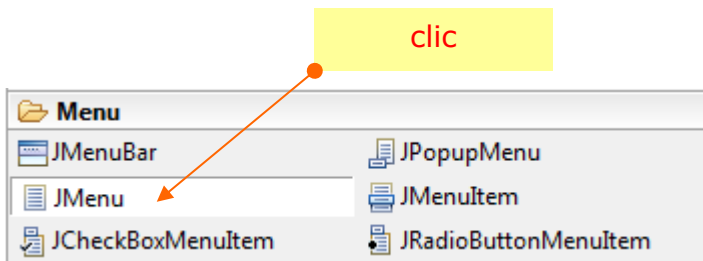


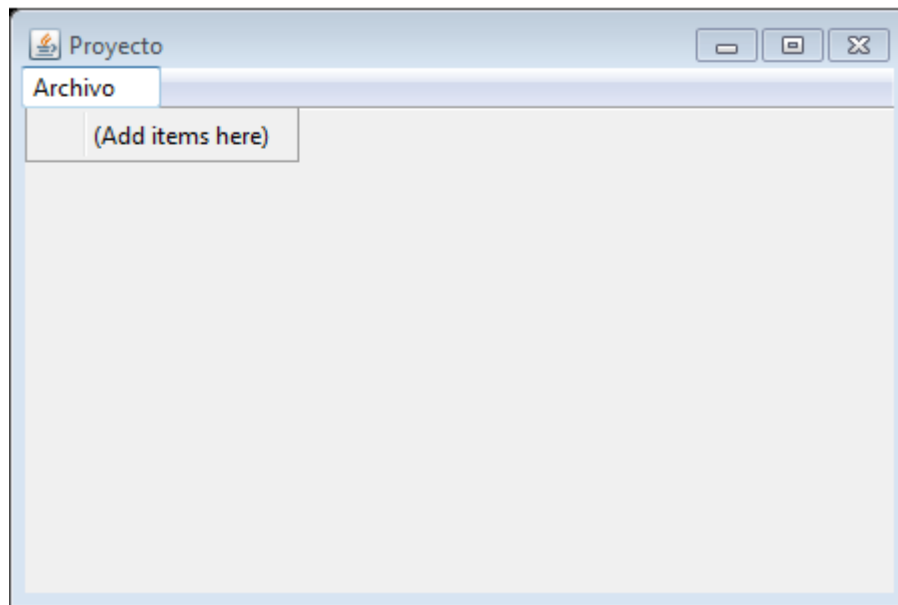


Usaremos el prefijo **menu**

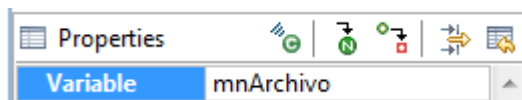


## Cree el menú “Archivo” (JMenu) en la barra de menú

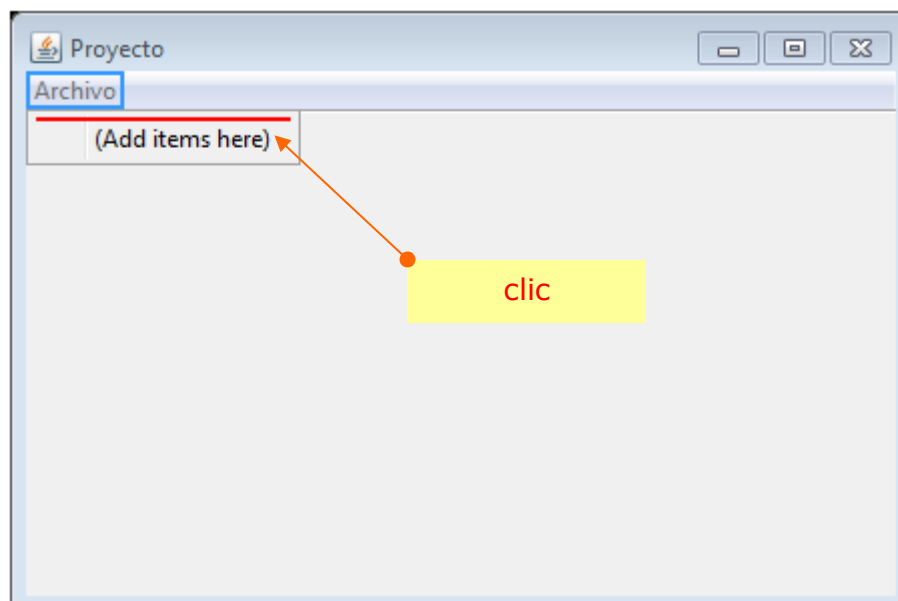
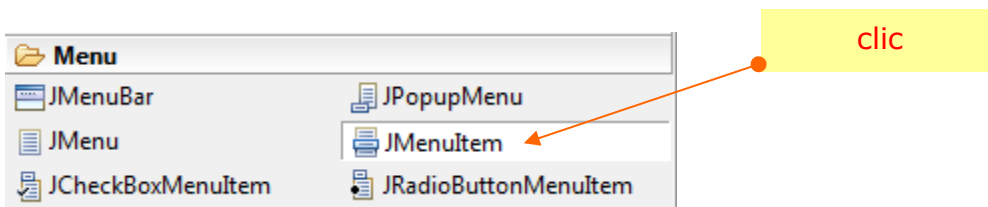


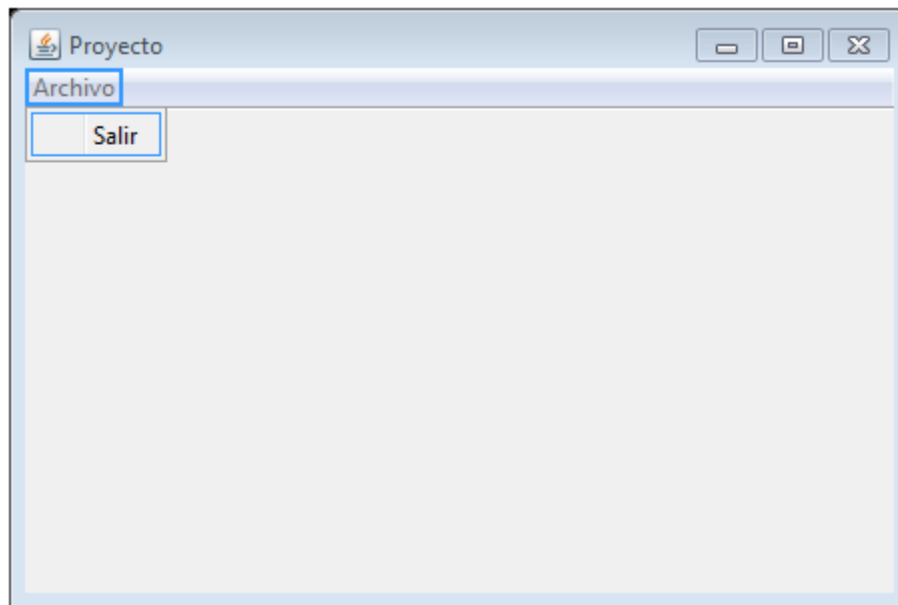


Usaremos el prefijo **mn**

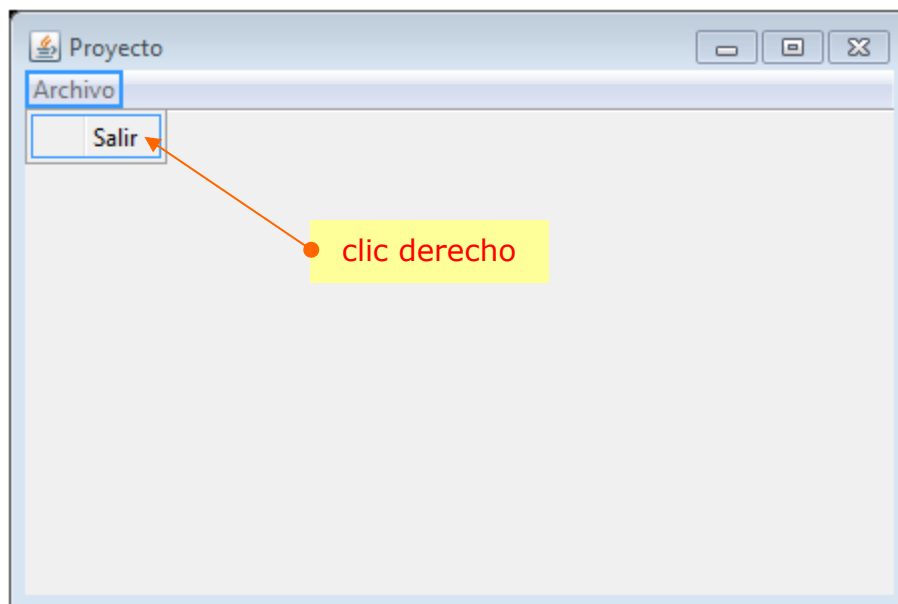
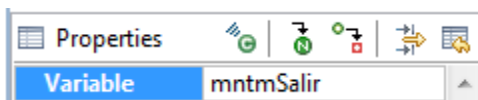


**Cree el ítem “Salir” (JMenuItem) en el menú “Archivo”**





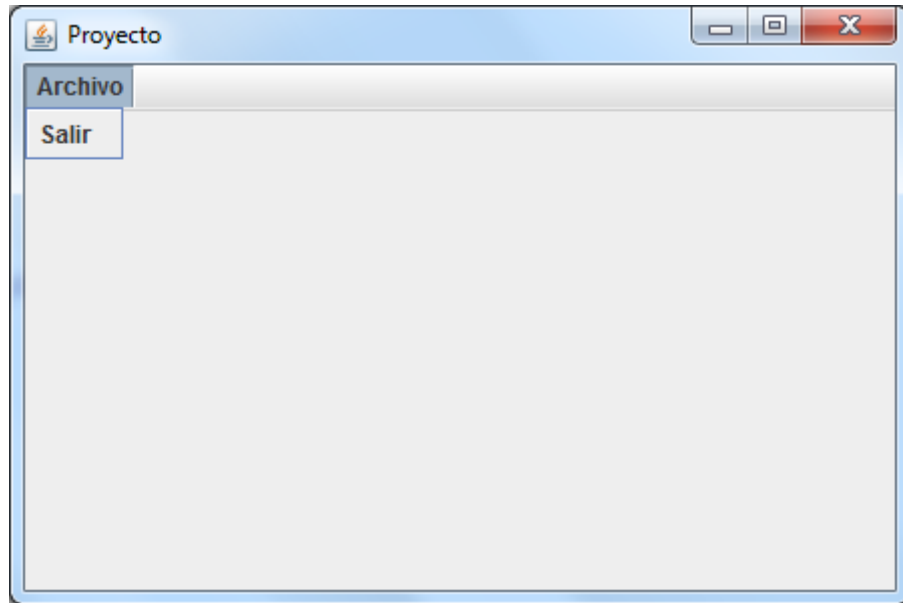
Usaremos el prefijo **mntm**



- Seleccione **Add event handler / action / actionPerformed**
- Eclipse autogenera el siguiente código:  

```
protected void actionPerformedMntmSalir(ActionEvent arg0) {
```
- Digite en el interior la sintaxis que cierra el Proyecto  

```
protected void actionPerformedMntmSalir(ActionEvent arg0) {  
    System.exit(0);  
}
```
- Grabe todo y ejecute el Proyecto



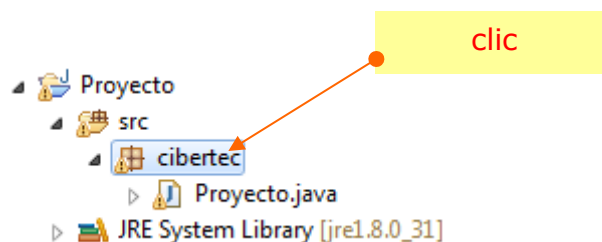
- Agregue en el Proyecto la variable precio y descuento, precedidas por el prefijo **public static**. El prefijo **public** permite el acceso a la variable desde cualquier parte y **static** para que dicha variable sea única en todo el Proyecto.

```
public class Proyecto extends JFrame implements ActionListener {
    // Declaración de variables globales
    public static double precio = 25.7, descuento = 15;

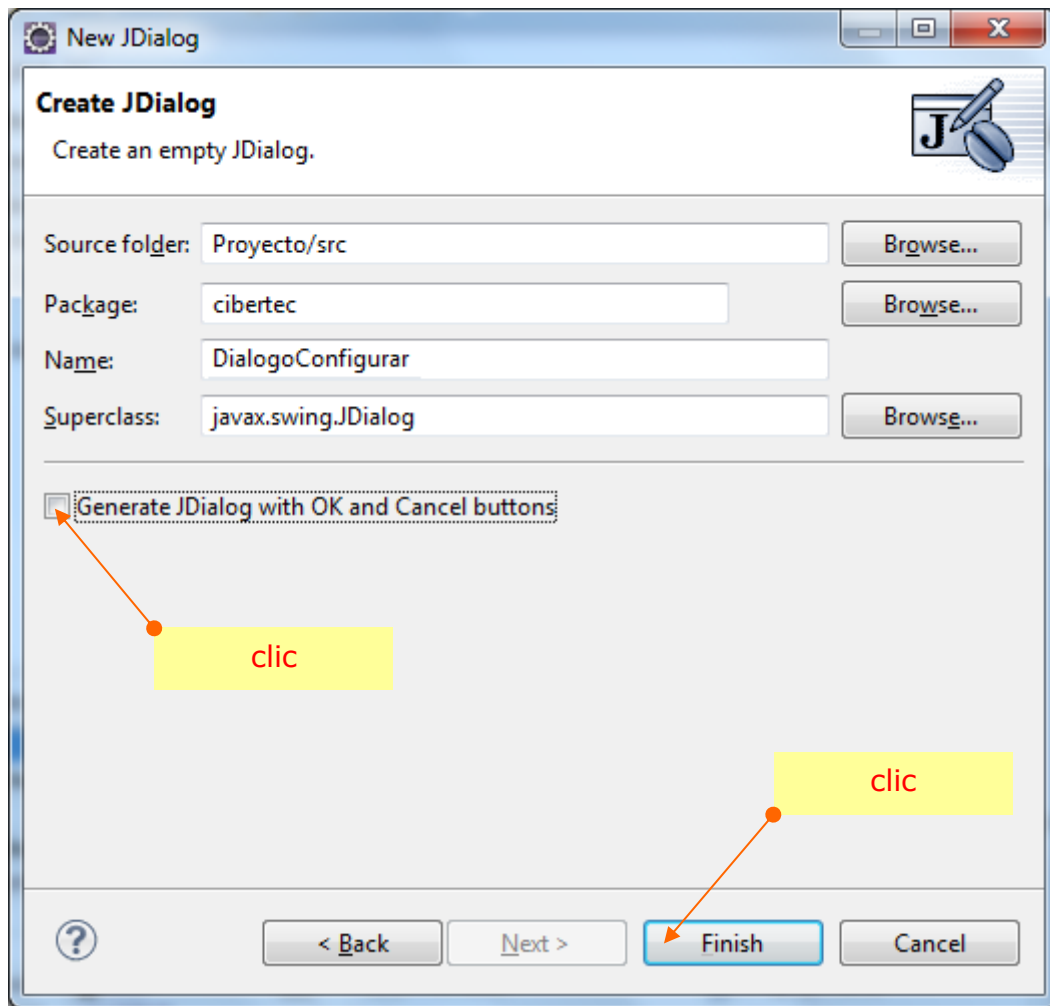
    private JPanel contentPane;
    ...
}
```

## Cree un JDialog (GUI) de nombre "DialogoConfigurar"

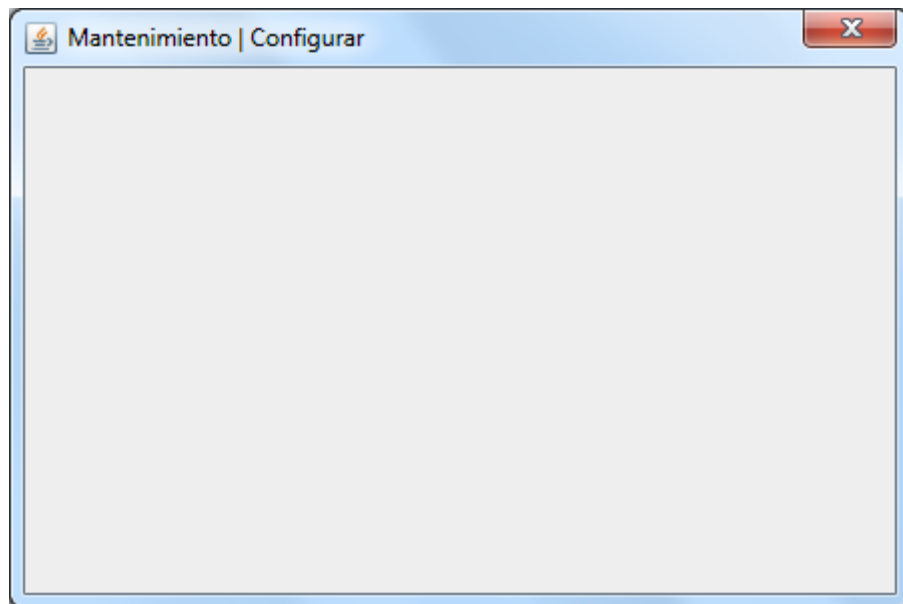
- Ilumine **cibertec**



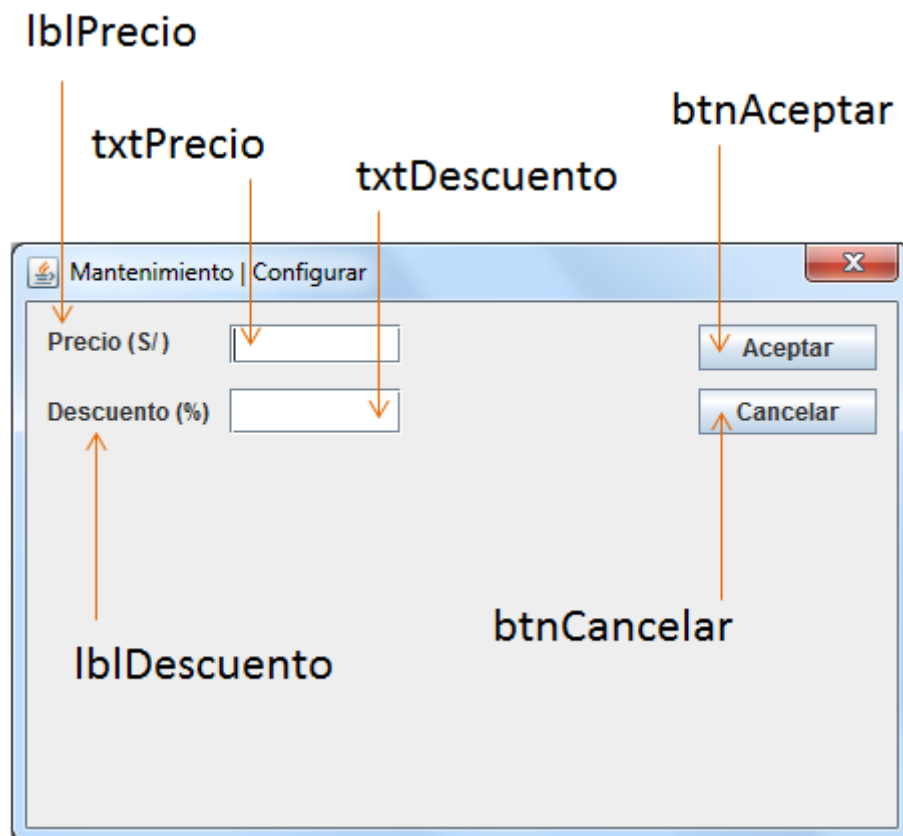
- Clic derecho y Seleccione **New / Other**
- En **WindowBuilder / SwingDesigner** seleccione **JDialog**
- Name: **DialogoConfigurar** *(debe comenzar siempre con letra mayúscula)*
- Desactive 'Generate JDialog with OK and Cancel buttons'



- Clic izquierdo en la pestaña Design
- Ubíquese en el interior del Componente Visual
- Haga clic derecho y seleccione **Set layout / Absolute layout**
- Ilumine el marco del Componente Visual
- title: **Mantenimiento | Configurar**
- Grabe y ejecute el JDialog



- Diseñe la siguiente GUI



- Grabe y ejecute el JDialog
- Doble clic izquierdo a los botones **Aceptar** y **Cancelar** para obtener:

```
protected void actionPerformedBtnAceptar(ActionEvent e) {
}
protected void actionPerformedBtnCancelar(ActionEvent e) {
}
```



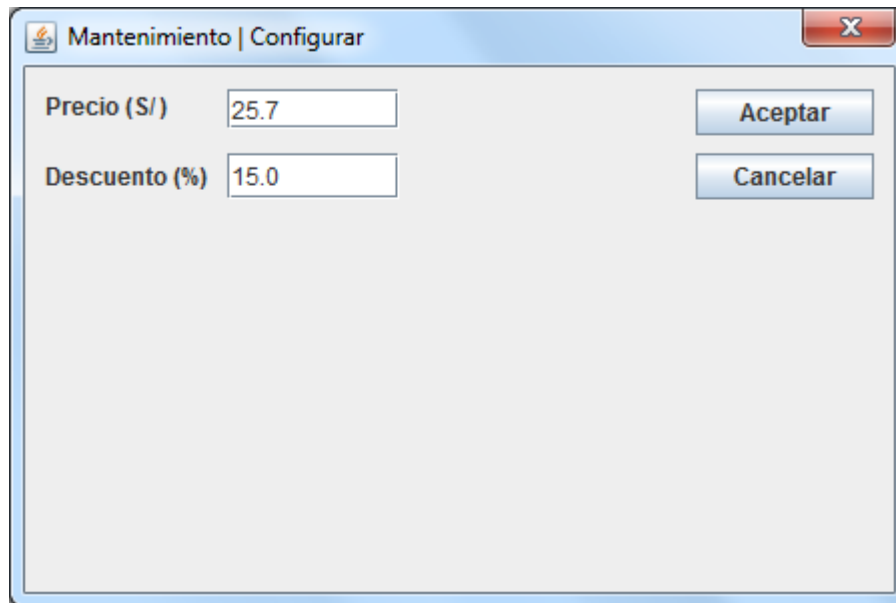
- Añada la sintaxis siguiente en `txtPrecio` y `txtDescuento`

```
txtPrecio = new JTextField("" + Proyecto.precio);
```

```
txtDescuento = new JTextField("" + Proyecto.descuento);
```

Se antepone **Proyecto** porque allí han sido declaradas las variables **precio** y **descuento**. El símbolo punto actúa como puerta de ingreso.

- Grabe y ejecute el JDialog



- A la pulsación del botón **Aceptar** actualice el contenido de las variables precio y descuento, luego cierre el JDialog (dispose cierra el subprograma)

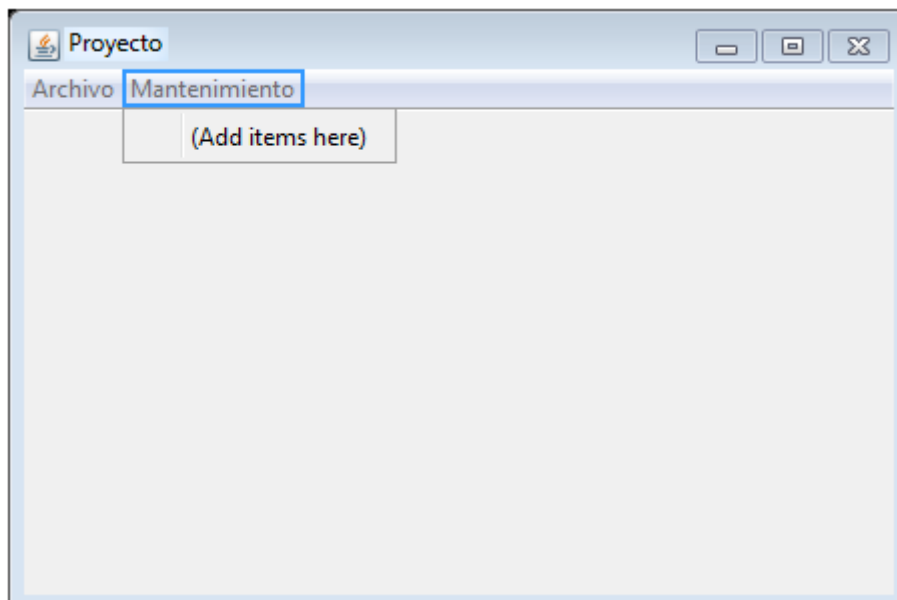
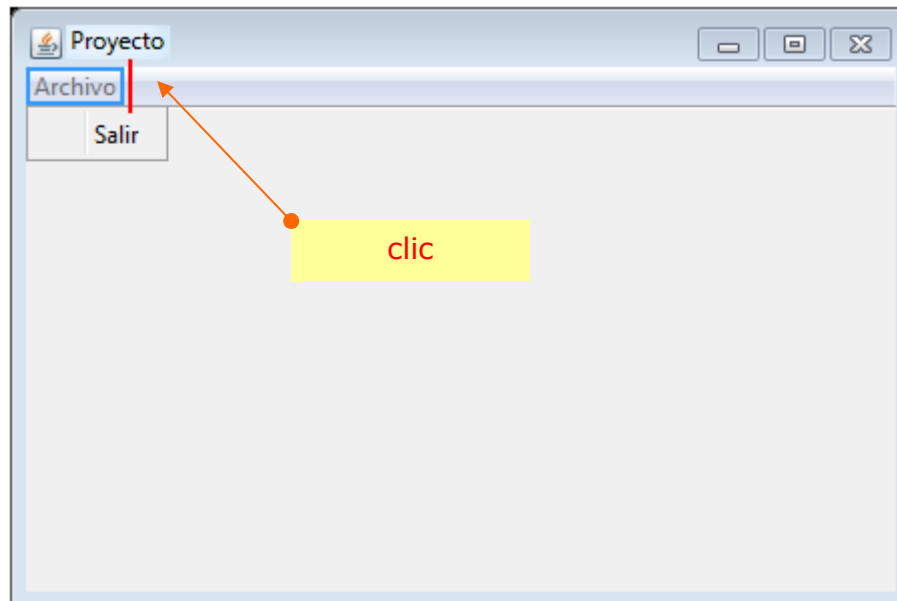
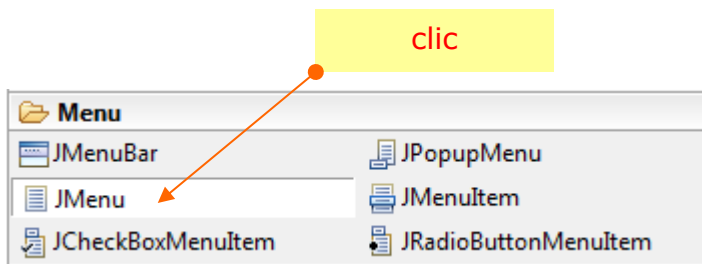
```
protected void actionPerformedBtnAceptar(ActionEvent e) {  
    Proyecto.precio = Double.parseDouble(txtPrecio.getText());  
    Proyecto.descuento = Double.parseDouble(txtDescuento.getText());  
    dispose();  
}
```

- A la pulsación del botón **Cancelar** cierre el JDialog sin hacer cambios

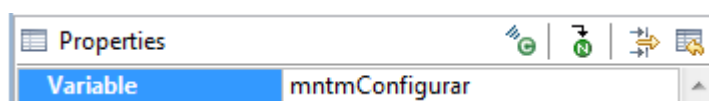
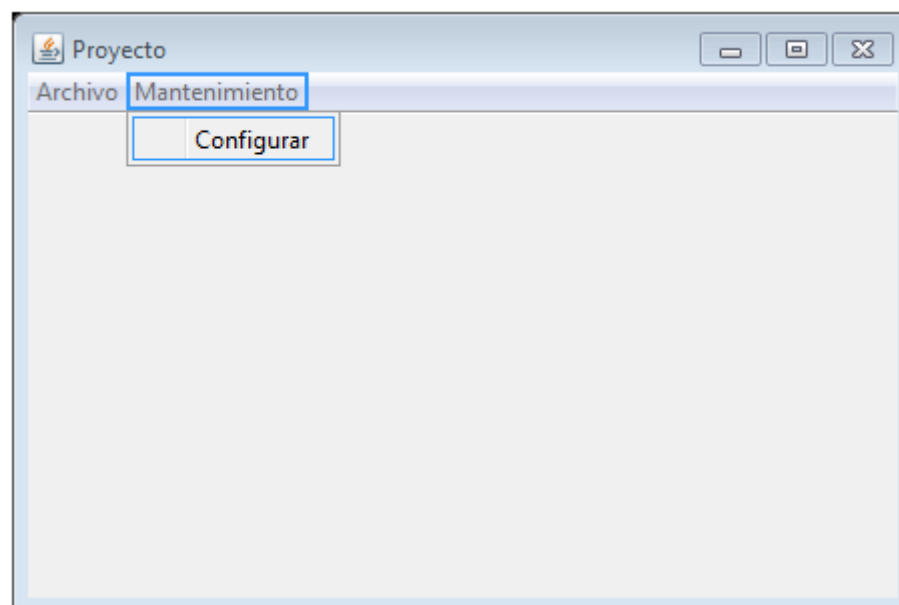
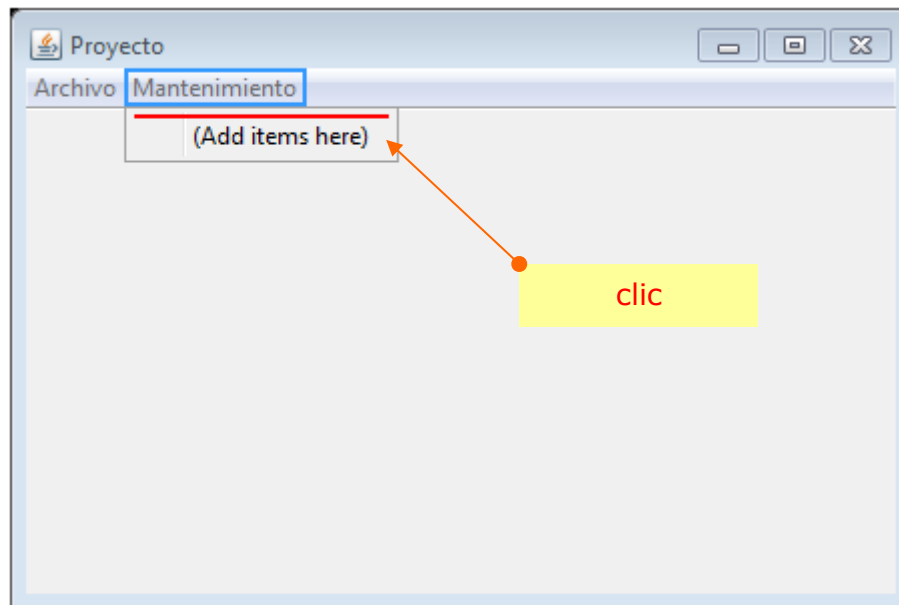
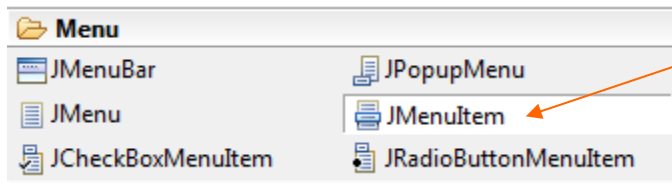
```
protected void actionPerformedBtnCancelar(ActionEvent e) {  
    dispose();  
}
```

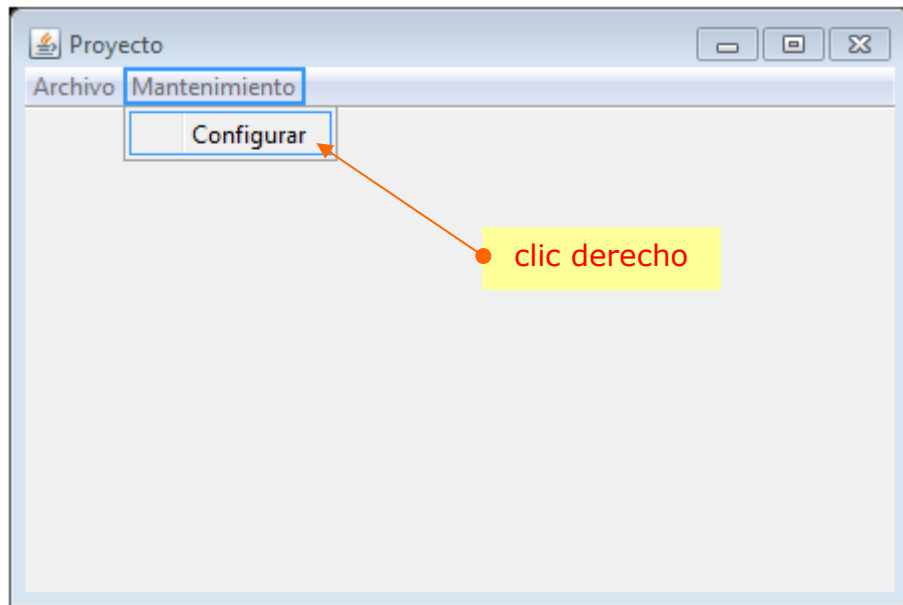
- Grabe el JDialog

## Agregue el menú "Mantenimiento" a la barra de menú



## Cree el ítem “Configurar” en el menú “Mantenimiento”





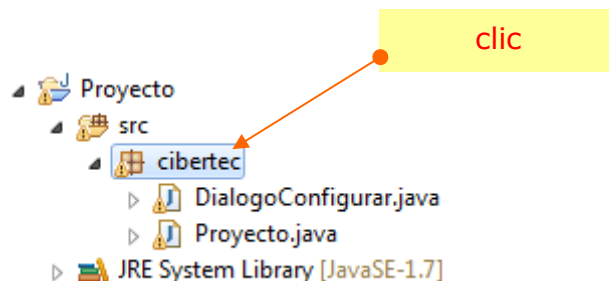
- Seleccione **Add event handler / action / actionPerformed**
- Eclipse autogenera el siguiente código:  

```
protected void actionPerformedMntmConfigurar(ActionEvent arg0) {
```
- Digite la sintaxis para vincular el JDialog **DialogoConfigurar** con el **Proyecto**  

```
protected void actionPerformedMntmConfigurar(ActionEvent arg0) {  
    DialogoConfigurar dc = new DialogoConfigurar();  
    dc.setLocationRelativeTo(this);  
    dc.setVisible(true);  
}
```
- Grabe todo y ejecute el Proyecto

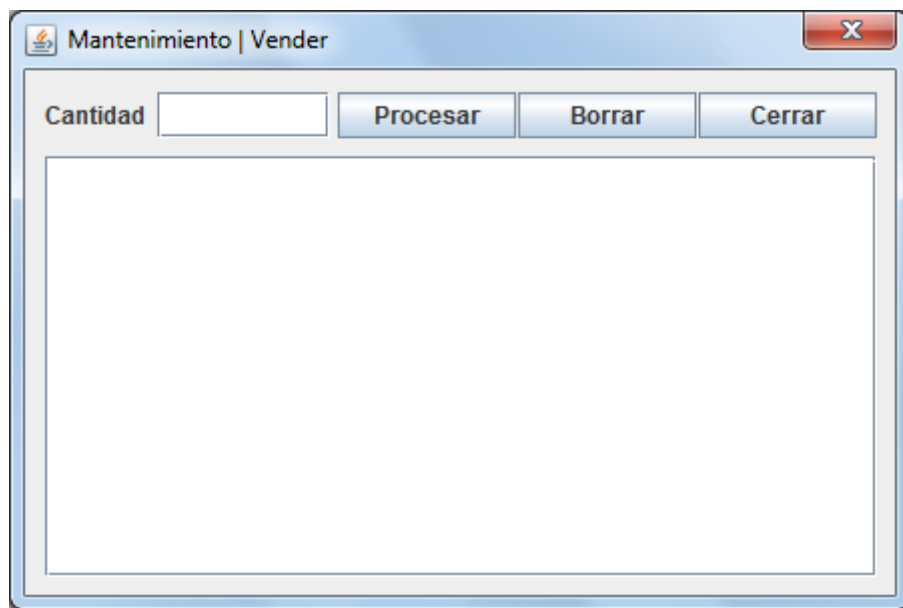
## Cree un JDialog (GUI) de nombre "DialogoVender"

- Ilumine **cibertec**



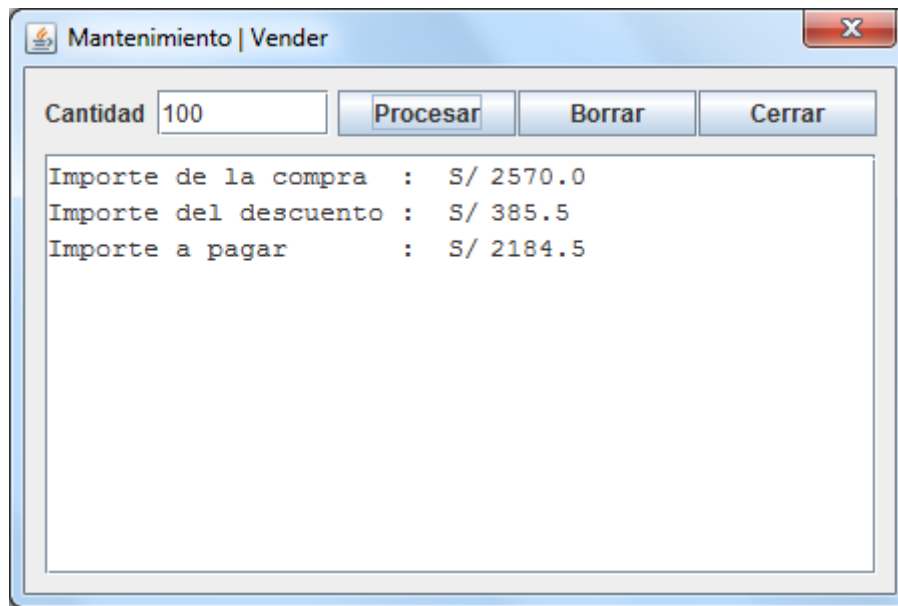
- Clic derecho y Seleccione **New / Other**
- En **WindowBuilder / SwingDesigner** seleccione **JDialog**
- Name: **DialogoVender** *(debe comenzar siempre con letra mayúscula)*
- Desactive 'Generate JDialog with OK and Cancel buttons'

- Pulse el botón Finish
- Clic izquierdo en la pestaña Design
- Ubíquese en el interior del Componente Visual
- Haga clic derecho y seleccione **Set layout / Absolute layout**
- Ilumine el marco del Componente Visual
- title: **Mantenimiento | Vender**
- Diseñe la siguiente GUI



- Grabe y ejecute el JDialog
- A la pulsación del botón **Procesar** genere una venta con los valores actuales de las variables precio y descuento.

```
protected void actionPerformedBtnProcesar(ActionEvent e) {
    // Declaración de variables
    int cantidad;
    double importeCompra, importeDescuento, importePagar;
    // Entrada de datos
    cantidad = Integer.parseInt(txtCantidad.getText());
    // Proceso de cálculo
    importeCompra = Proyecto.precio * cantidad;
    importeDescuento = Proyecto.descuento * importeCompra / 100;
    importePagar = importeCompra - importeDescuento;
    // Salida de resultados
    txtS.setText("Importe de la compra : S/ " + importeCompra + "\n");
    txtS.append("Importe del descuento : S/ " + importeDescuento + "\n");
    txtS.append("Importe a pagar : S/ " + importePagar);
}
```

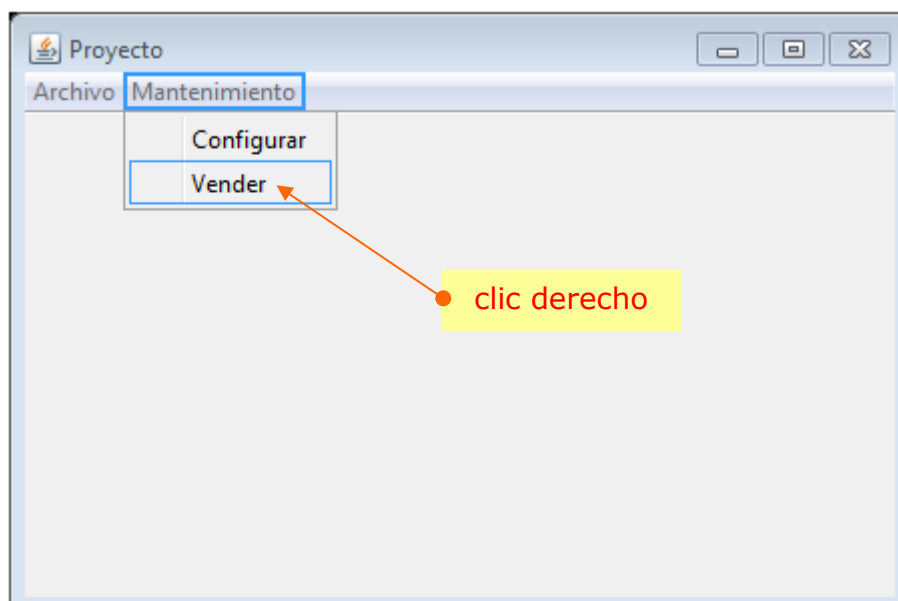
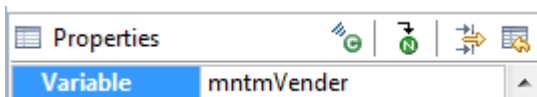


- A la pulsación del botón **Borrar** limpie las regiones y reponga el cursor  

```
protected void actionPerformedBtnBorrar(ActionEvent e) {
    txtCantidad.setText("");
    txtS.setText("");
    txtCantidad.requestFocus();
}
```
- A la pulsación del botón **Cerrar** cierre el JDialog  

```
protected void actionPerformedBtnCerrar(ActionEvent e) {
    dispose();
}
```

## Agregue el ítem "Vender" en el menú "Mantenimiento"



- Seleccione **Add event handler / action / actionPerformed**
- Eclipse autogenera el siguiente código:

```
protected void actionPerformedMntmVender(ActionEvent arg0) {  
}
```

- Digite la sintaxis para vincular el JDialog **DialogoVender** con el **Proyecto**

```
protected void actionPerformedMntmVender(ActionEvent arg0) {  
    DialogoVender dv = new DialogoVender();  
    dv.setLocationRelativeTo(this);  
    dv.setVisible(true);  
}
```

- Grabe todo y ejecute el Proyecto

*"Una buena enseñanza es aquella que nos conduce hacia un gran cambio."*

**MP**