

Asignatura

Programación



UNIVERSAE
Instituto Superior de FP

Asignatura

Programación

UNIDAD 15

Creación de interfaces gráficas



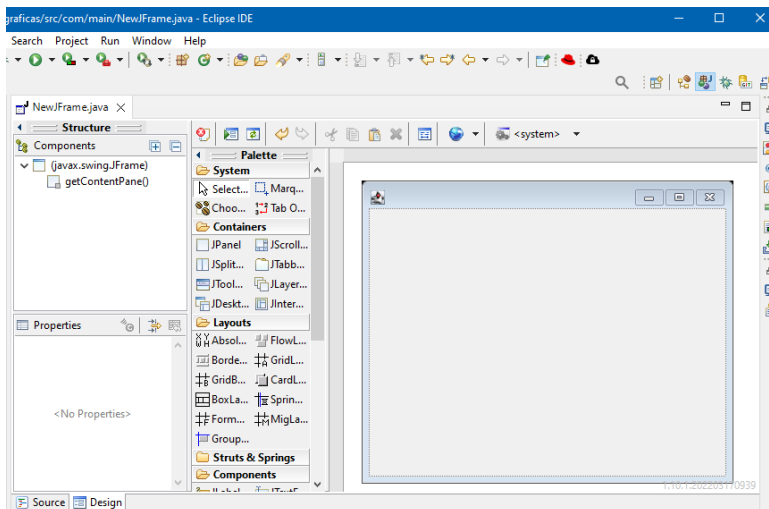
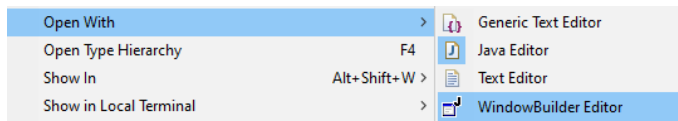
UNIVERSAE
Instituto Superior de FP

Herramientas gráficas

- Los IDEs incorporan herramientas para desarrollo de interfaces
- Ocultan el código y se muestra de forma visual
- Permiten el *drag and drop*
- Generan código automáticamente.
- Disponen de varias pestañas Source y Design

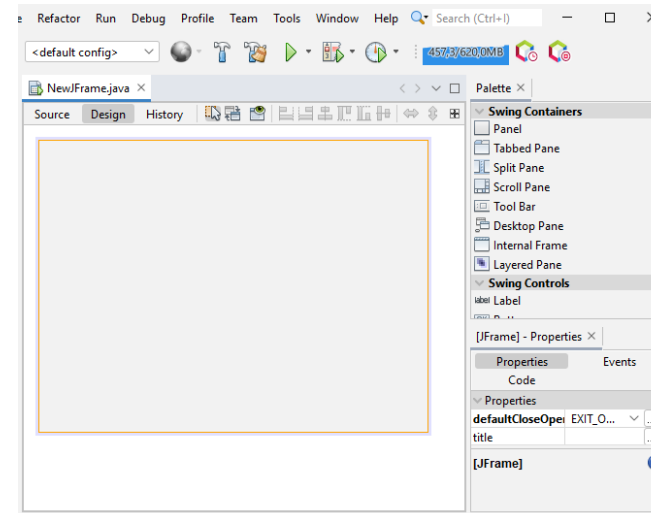
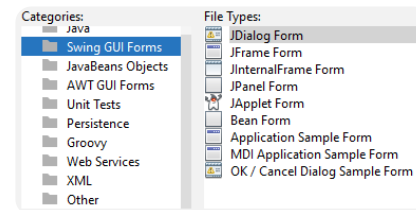
Eclipse

- WindowBuilder
- **Instalación.** *Help/Eclipse Marketplace*
- Se habilita al pulsar botón secundario sobre el fichero fuente y elegir Open With WindowBuilder Editor



Netbeans

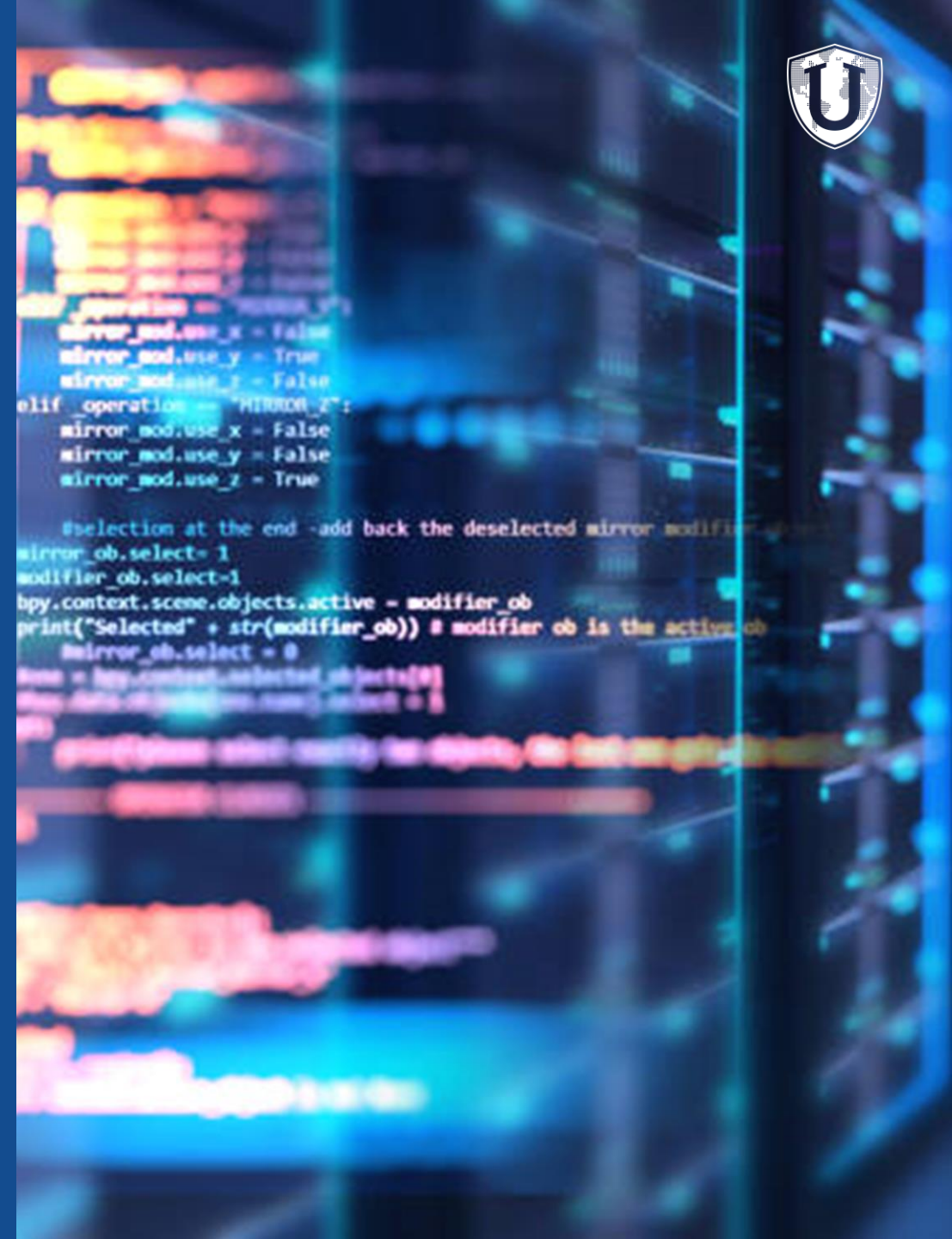
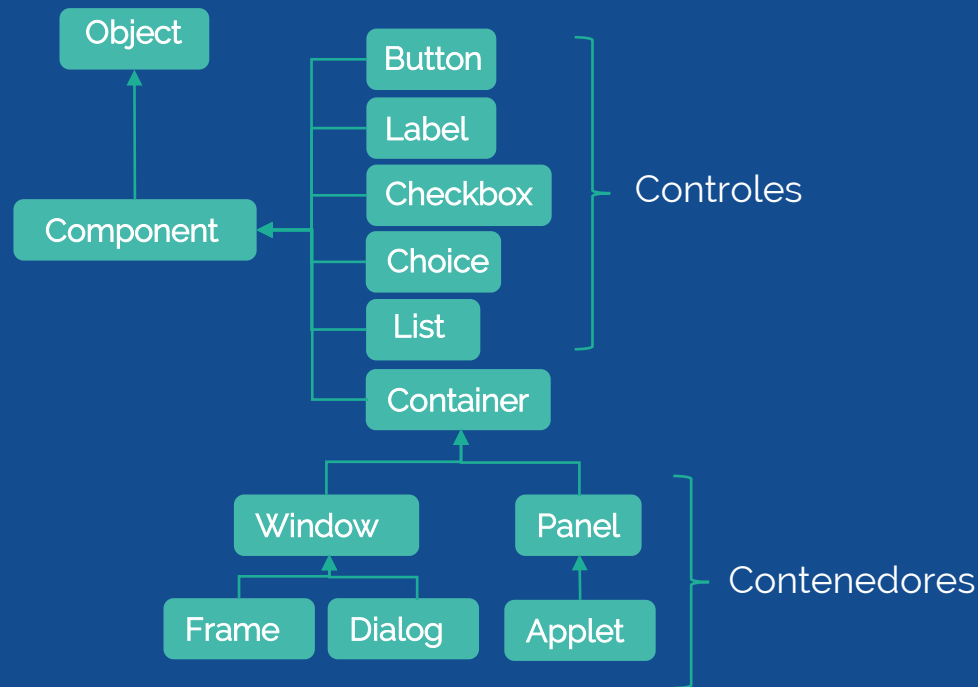
- **No necesita instalación.** Esta incorporado en el IDE
- Se habilita al generar un fichero fuente del tipo Swing o AWT





Librería AWT

- Abstract Window Toolkit, permite la creación de interfaces gráficas y nos da acceso a las clases Component y Container, las cuales definen los controles y la pantalla de la aplicación en desarrollo respectivamente.
- Esta librería utiliza el paquete *javax.awt.**





AWT. Contenedores

¿Qué són?

- Definen la estructura de ventanas de la aplicación
- Permiten almacenar otros componentes.

Tipos

- Window
- Frame
- Dialog
- FileDialog
- Panel



```
Frame frame = new Frame();
frame.setBounds(100, 100, 450, 300);

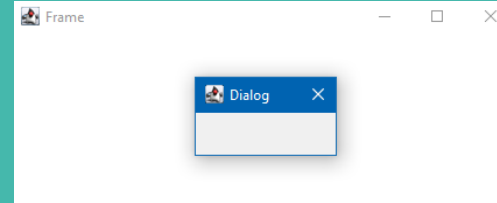
BorderLayout borderLayout = new BorderLayout();
frame.setLayout(borderLayout);

Label label = new Label("New label");
frame.add(label, BorderLayout.WEST);

TextField textField = new TextField();
frame.add(textField, BorderLayout.CENTER);

frame.addWindowListener(new WindowAdapter() {
    public void windowClosing(WindowEvent evt) {
        System.exit(0);
    }
});

frame.setVisible(true);
```



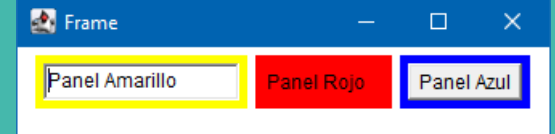
```
Frame frame = new Frame("Frame");
frame.setBounds(100, 100, 450, 300);

BorderLayout borderLayout = new BorderLayout();
frame.setLayout(borderLayout);

Dialog dialog = new Dialog(frame, "Dialog");
dialog.setSize(125, 75);
dialog.setVisible(true);

frame.addWindowListener(new WindowAdapter() {
    public void windowClosing(WindowEvent evt) {
        System.exit(0);
    }
});

frame.setVisible(true);
```



```
Frame frame = new Frame("Frame");
frame.setSize(500, 200);
frame.setLayout(new FlowLayout());

Panel pnlAmarillo = new Panel();
pnlAmarillo.setBackground(Color.yellow);
pnlAmarillo.add(new TextField("Panel Amarillo"));

Panel pnlRojo = new Panel();
pnlRojo.setBackground(Color.red);
pnlRojo.add(new Label("Panel Rojo"));

Panel pnlBlue = new Panel();
pnlBlue.setBackground(Color.blue);
pnlBlue.add(new Button("Panel Azul"));

frame.add(pnlAmarillo);
frame.add(pnlRojo);
frame.add(pnlBlue);

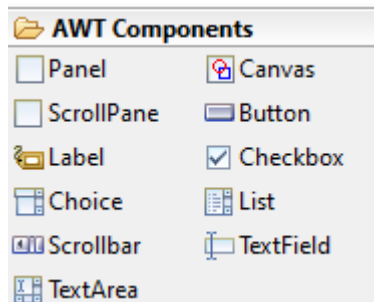
frame.addWindowListener(new WindowAdapter() {
    public void windowClosing(WindowEvent evt) {
        System.exit(0);
    }
});

frame.setVisible(true);
```

AWT. Controles

¿Qué són?

- Componentes visuales contenidos en un contenedor
- Permiten definir acciones.
- Se puede modificar aspectos visuales, dimensiones, posición, visibilidad, etc.



Tipos

- Button
- Checkbox
- Choice
- Label
- TextField
- List
- TextArea

```
Frame marco = new Frame();
marco.setSize(400,400);
marco.setLayout(null);
marco.setVisible(true);

marco.addWindowListener(new WindowAdapter() {
    @Override
    public void windowClosing(WindowEvent e) {
        marco.dispose();
    }
});

Label label = new Label("Etiqueta");
label.setBounds(100, 100, 100, 20);

TextField campo = new TextField();
campo.setBounds(200, 100, 100, 20);

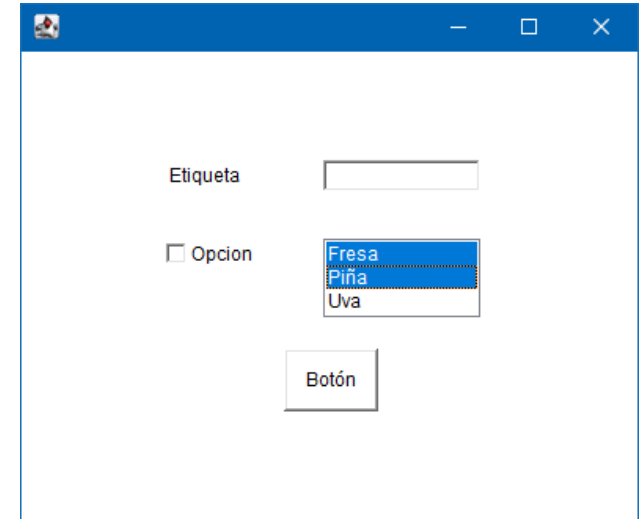
Checkbox checkbox = new Checkbox("Opcion");
checkbox.setBounds(100, 150, 100, 20);

List list = new List();
list.setBounds(200, 150, 100, 50);
list.setMultipleMode(true);
list.add("Fresa");
list.add("Piña");
list.add("Uva");

Button boton = new Button("Botón");

boton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        campo.setText("Acción del botón");
    }
});
boton.setBounds(175, 220, 60, 40);

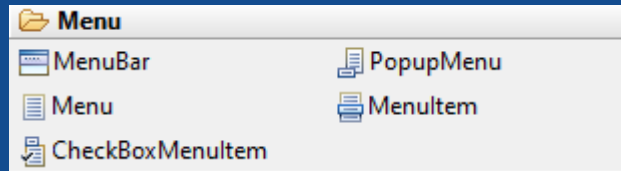
marco.add(label);
marco.add(campo);
marco.add(checkbox);
marco.add(list);
marco.add(boton);
```



AWT. Menu

¿Qué són?

- Es el componente que se sitúa en la parte de arriba de la ventana
- Permite organizar acciones de la aplicación



Tipos

- MenuBar
- Menu
- MenuItem
- MenuShortcut
- CheckboxMenuItem
- PopMenu

```
Frame frame = new Frame();
frame.setSize(400,348);
frame.setVisible(true);

MenuBar menuBar = new MenuBar();

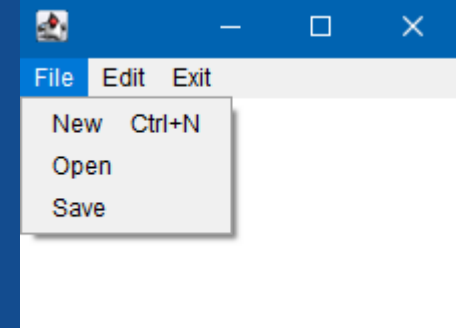
Menu menuFile = new Menu("File");
MenuItem itemNew = new MenuItem("New");
itemNew.setShortcut(new MenuShortcut(KeyEvent.VK_N, false));
MenuItem itemOpen = new MenuItem("Open");
MenuItem itemSave = new MenuItem("Save");
menuFile.add(itemNew);
menuFile.add(itemOpen);
menuFile.add(itemSave);

Menu menuEdit = new Menu("Edit");
MenuItem itemSelectA = new MenuItem("Select All");
menuEdit.add(itemSelectA);

Menu menuExit = new Menu("Exit");

menuBar.add(menuFile);
menuBar.add(menuEdit);
menuBar.add(menuExit);

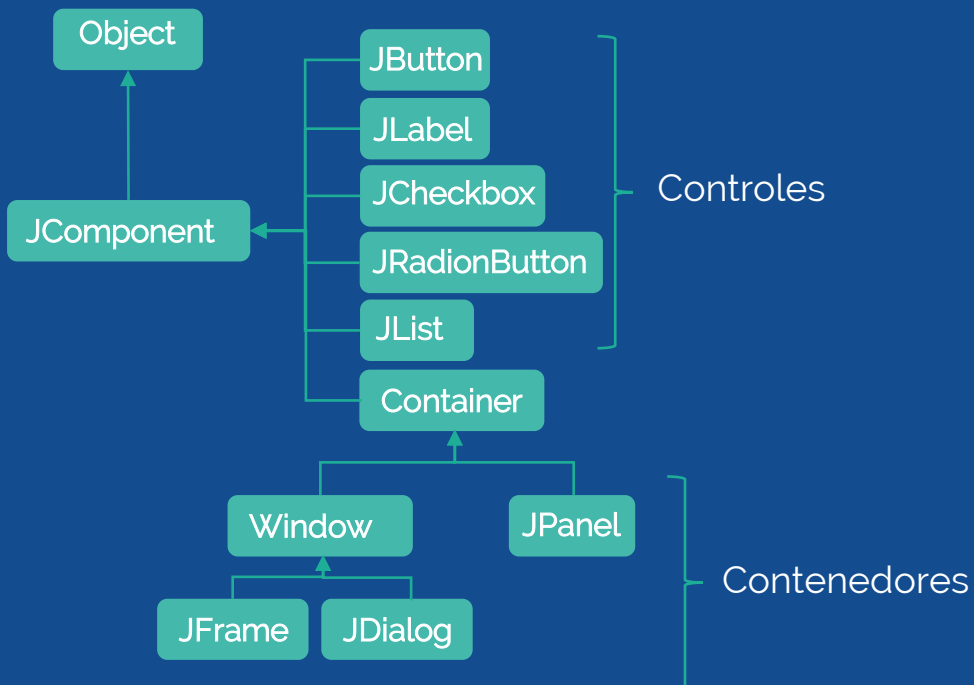
frame.setMenuBar(menuBar);
```





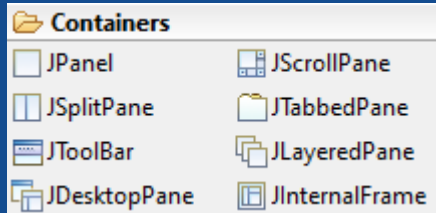
Librería Swing

- Evolución de AWT que incorpora nuevas herramientas al tiempo que elimina algunas limitaciones, dándonos una mayor libertad de trabajo.
- Con Swing se puede utilizar los componentes de AWT y los propios.
- Los componentes de Swing empiezan por J
- Esta librería utiliza el paquete para su uso es *javax.swing.**



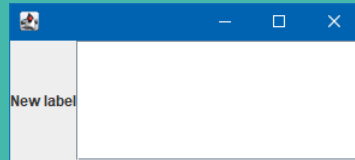
Swing. Contenedores

- Swing dispone de contenedores adicionales.



Tipos

- Window
- JFrame
- JDialog
- Jpanel
- JTabbedPane



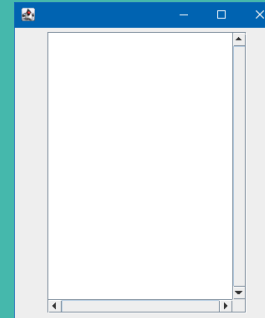
```
JFrame frame = new JFrame();
frame.setBounds(100, 100, 450, 300);

BorderLayout borderLayout = new BorderLayout();
frame.setLayout(borderLayout);

JLabel label = new JLabel("New label");
frame.add(label, BorderLayout.WEST);

JTextField textField = new JTextField();
frame.add(textField, BorderLayout.CENTER);

frame.setVisible(true);
```



```
JFrame frame = new JFrame();

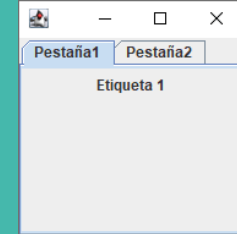
frame.setSize(500, 500);
frame.setVisible(true);
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

frame.getContentPane().setLayout(new FlowLayout());

JTextArea textArea = new JTextArea(20, 20);
JScrollPane scrollableTextArea = new JScrollPane(textArea);

scrollableTextArea.setHorizontalScrollBarPolicy(
    JScrollPane.HORIZONTAL_SCROLLBAR_ALWAYS);
scrollableTextArea.setVerticalScrollBarPolicy(
    JScrollPane.VERTICAL_SCROLLBAR_ALWAYS);

frame.getContentPane().add(scrollableTextArea);
```



```
JFrame jFrame = new JFrame();
jFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
jFrame.setSize(200, 200);
jFrame.setVisible(true);

JTabbedPane panelSolapas = new JTabbedPane();
JPanel jPanel1 = new JPanel();

JLabel jLabel = new JLabel("Etiqueta 1");
jPanel1.add(jLabel);

panelSolapas.add("Pestaña1", jPanel1);

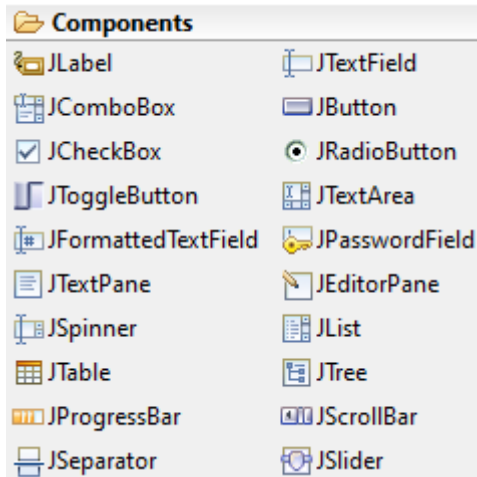
JPanel jPanel2 = new JPanel();

JLabel jLabel2 = new JLabel("Etiqueta 2");
jPanel2.add(jLabel2);

panelSolapas.add("Pestaña2", jPanel2);

jFrame.getContentPane().add(panelSolapas);
```

Swing. Controles



```
JFrame frame = new JFrame();
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
frame.setSize(194, 200);
frame.setVisible(true);
frame.getContentPane().setLayout(new FlowLayout());

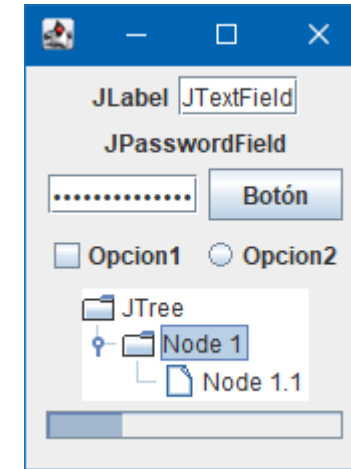
JLabel jLabel = new JLabel("JLabel");
JTextField campo = new JTextField("JTextField");
JLabel jLabel1 = new JLabel("JPasswordField");
JPasswordField passwordField = new JPasswordField("JPasswordField");
JButton boton = new JButton("Botón");

JCheckBox chkcbxNewCheckBox = new JCheckBox("Opcion1");
JRadioButton rdbtnNewRadioButton = new JRadioButton("Opcion2");

// Arbol
DefaultMutableTreeNode raiz = new DefaultMutableTreeNode("JTree");
DefaultMutableTreeNode n1 = new DefaultMutableTreeNode("Node 1");
DefaultMutableTreeNode ns1 = new DefaultMutableTreeNode("Node 1.1");
raiz.add(n1);
n1.add(ns1);
JTree arbol = new JTree(raiz);

// Barra de progreso
JProgressBar progressBar = new JProgressBar(0, 100);
progressBar.setValue(25);

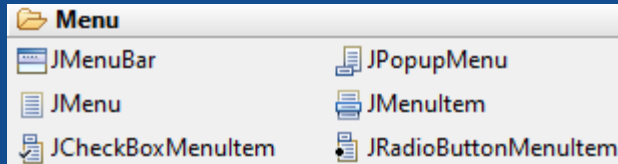
frame.getContentPane().add(jLabel);
frame.getContentPane().add(campo);
frame.getContentPane().add(jLabel1);
frame.getContentPane().add(passwordField);
frame.getContentPane().add(boton);
frame.getContentPane().add(chkcbxNewCheckBox);
frame.getContentPane().add(rdbtnNewRadioButton);
frame.getContentPane().add(arbol);
frame.getContentPane().add(progressBar);
```



Swing. Menu

El componente Menu

- Es el componente que se sitúa en la parte de arriba de la ventana
- Permite organizar acciones de la aplicación
- Tiene la misma aplicación que los menus de AWT



Tipos

- JMenuBar
- JMenu
- JMenuItem
- JCheckBoxMenuItem
- JRadioButtonMenuItem
- KeyStroke

```
JFrame frame = new JFrame();
frame.setSize(408,348);

JMenuBar menuBar = new JMenuBar();

JMenu menuFile = new JMenu("File");
JMenuItem itemNew = new JMenuItem("New");
itemNew.setAccelerator(KeyStroke.getKeyStroke("control N"));
JMenuItem itemOpen = new JMenuItem("Open");
JMenuItem itemSave = new JMenuItem("Save");
menuFile.add(itemNew);
menuFile.add(itemOpen);
menuFile.add(itemSave);

JMenu menuEdit = new JMenu("Edit");

JMenu menuSelect = new JMenu("Select");
JCheckBoxMenuItem chckbxmntmOp1 = new JCheckBoxMenuItem("Opción 1");
JRadioButtonMenuItem rdbtnmntmOp2 = new JRadioButtonMenuItem("Opción 2");
menuSelect.add(chckbxmntmOp1);
menuSelect.add(rdbtnmntmOp2);

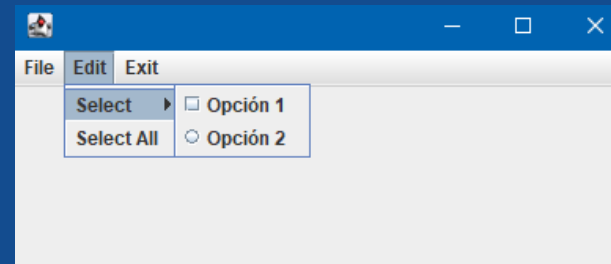
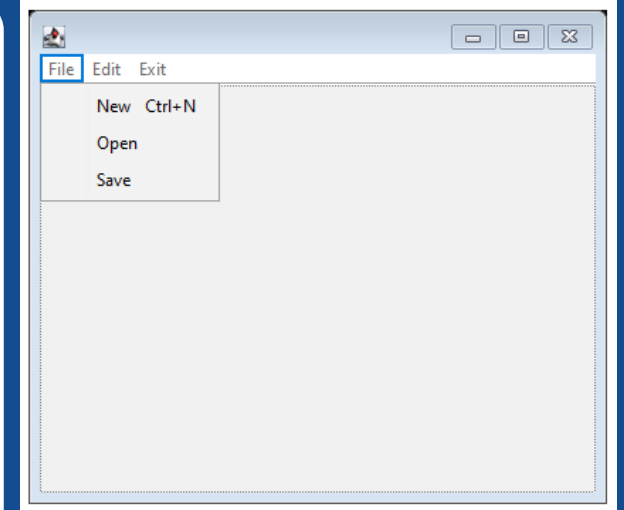
JMenuItem itemSelectA = new JMenuItem("Select All");

menuEdit.add(menuSelect);
menuEdit.add(itemSelectA);

JMenu menuExit = new JMenu("Exit");

menuBar.add(menuFile);
menuBar.add(menuEdit);
menuBar.add(menuExit);

frame.setJMenuBar(menuBar);
frame.setVisible(true);
```



Layouts

La forma de organizar los componentes dentro del contenedor es mediante los **Layouts**, en caso de su ausencia, será mediante posiciones absolutas.

Con `.setLayout()` podemos añadir un layout al contenedor.

Tipos

➤ BorderLayout

Permite colocar elementos en el centro de la ventana, así como en los cuatro bordes, North, South, East, West y Center

➤ CardLayout

Organiza los componentes sobre el mismo espacio. Permite mostrar uno de tras de otro como una lista haciendo uso del método `.next()`

➤ FlowLayout

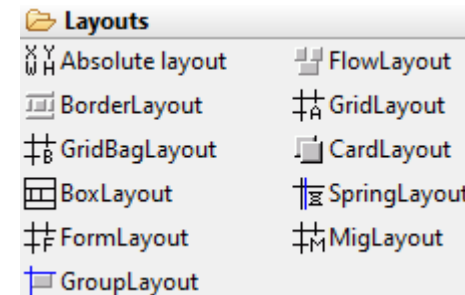
Nos permite colocar componentes en una misma fila de izquierda a derecha. Si se sobrepasa comienza en la siguiente fila.

➤ GridLayout

Siguiendo el patrón de una tabla. Distribuye cada elemento en filas y columnas

➤ GridBagLayout

Más flexible que GridLayout, permitiendo colocar cada componente en la celda deseada de una cuadrícula ya hecha





Layouts. Ejemplos

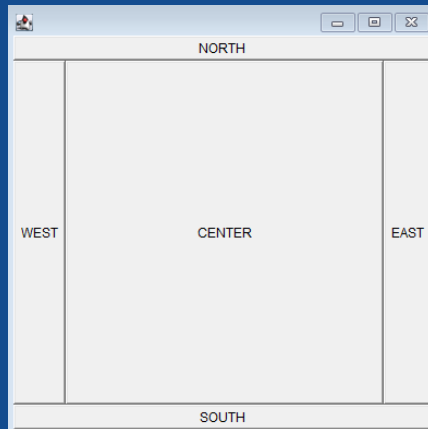
BorderLayout

```
Frame frame = new Frame();
frame.setSize(400,400);

BorderLayout borderLayout = new BorderLayout();
frame.setLayout(borderLayout);

frame.add(new Button("NORTH"),BorderLayout.NORTH);
frame.add(new Button("SOUTH"),BorderLayout.SOUTH);
frame.add(new Button("EAST"),BorderLayout.EAST);
frame.add(new Button("WEST"),BorderLayout.WEST);
frame.add(new Button("CENTER"),BorderLayout.CENTER);

frame.setVisible(true);
```



CardLayout

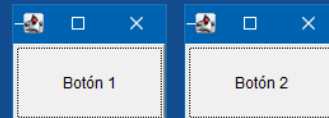
```
Frame frame = new Frame();
frame.setSize(100,100);

CardLayout cardLayout = new CardLayout();
frame.setLayout(cardLayout);

Button button = new Button("Botón 1");
button.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        cardLayout.next(frame);
    }
});
frame.add(button);
Button button2 = new Button("Botón 2");

button2.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        cardLayout.next(frame);
    }
});
frame.add(button2);

frame.setVisible(true);
```



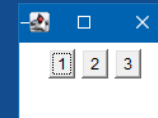
FlowLayout

```
Frame frame = new Frame();
frame.setSize(100,100);

FlowLayout flowLayout = new FlowLayout();
frame.setLayout(flowLayout);

frame.add(new Button("1"));
frame.add(new Button("2"));
frame.add(new Button("3"));

frame.setVisible(true);
```



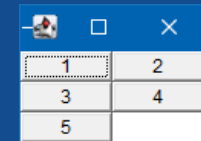
GridLayout

```
Frame frame = new Frame();
frame.setSize(100,100);

GridLayout GridLayout = new GridLayout();
GridLayout.setColumns(2);
GridLayout.setRows(3);
frame.setLayout(GridLayout);

frame.add(new Button("1"));
frame.add(new Button("2"));
frame.add(new Button("3"));
frame.add(new Button("4"));
frame.add(new Button("5"));

frame.setVisible(true);
```

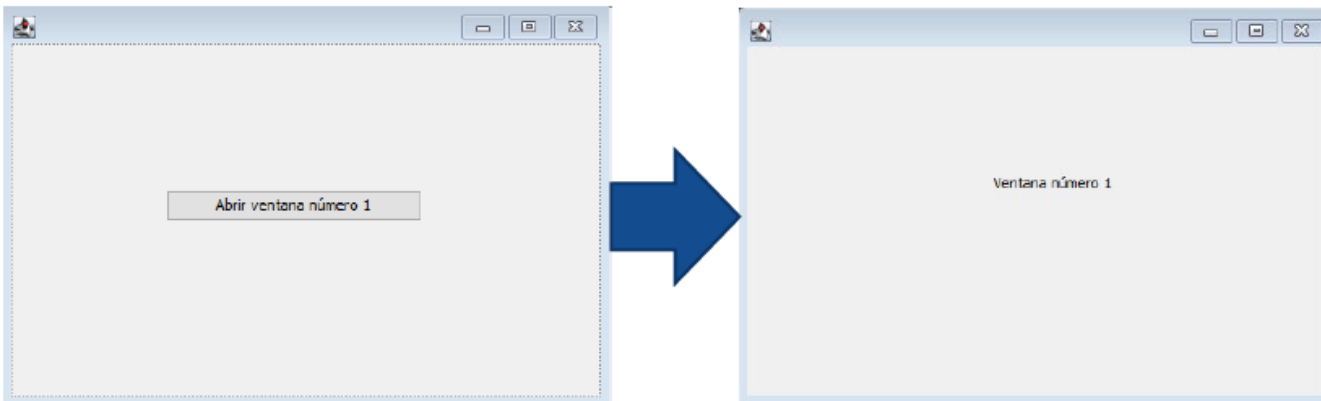
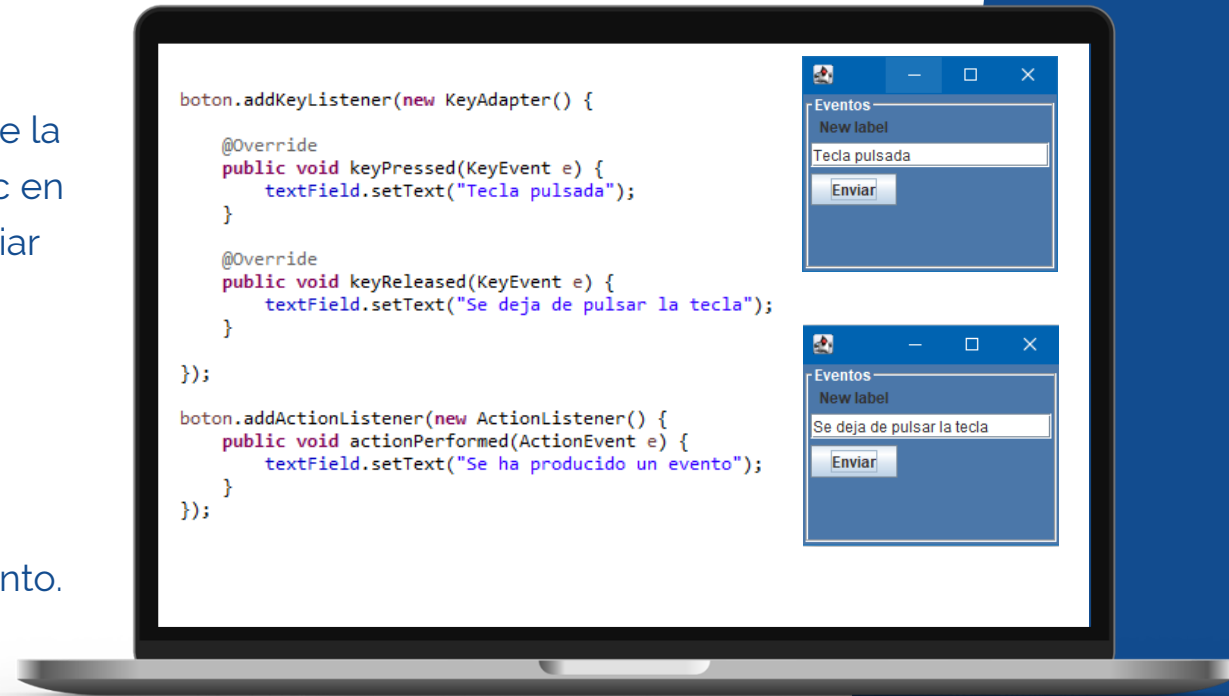


Eventos

Un evento es la recepción de un estímulo por parte de la aplicación, como puede ser el reconocimiento del clic en un botón, estos eventos permitirán a la aplicación iniciar una reacción que se asocie a ellos.

Podemos encontrar dos tipos de eventos:
externos e internos

Los componentes lleven asociados algún tipo de evento.
Este evento será observado por el escuchador





Resumen

1. Herramientas gráficas
2. Librería AWT
3. AWT. Contenedores
4. AWT. Controles
5. AWT. Menu
6. Librería Swing
7. Swing. Contenedores
8. Swing .Controles
9. Swing. Menu
10. Layouts
11. Layouts. Ejemplos
12. Eventos



UNIVERSAE

— CHANGE YOUR WAY —