



**UNIVERSIDAD TÉCNICA DE COTOPAXI**  
**FACULTAD DE**  
**CIENCIAS DE LA INGENIERÍA Y APLICADAS**  
**CARRERA**  
**INGENIERÍA EN INFORMÁTICA Y SISTEMAS COMPUTACIONALES**

**ASIGNATURA:**  
**PROGRAMACION DE BASE DE DATOS**

**CURSO:**  
**SEXTO SISTEMAS**

**DOCENTE:**  
**ING. SILVIA BRAVO**

**NOMBRE:**  
**DANILO MOYA**  
**LILIANA TAPIA**

**FECHA:**  
**16-11-2017**  
**LATACUNGA-ECUADOR**

## 1. CONCEPTO DE JAVA

Java es un lenguaje de programación con el que podemos realizar cualquier tipo de programa [1], cabe recalcar que java es un lenguaje netamente orientado a objetos, trayéndonos muchos beneficios como la herencia y el polimorfismo, ayudando nos así a optimizar el código fuente.

## 2. CONCEPTO DE NETBEANS

NetBeans IDE es un entorno de desarrollo - una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java - pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el NetBeans IDE. NetBeans IDE es un producto libre y gratuito sin restricciones de uso [2], Cabe recalcar que actualmente el dueño de NetBeans es propietario de Oracle.

## 3. POR QUE USAR NETBEANS CON JAVA

NetBeans es una IDE de código abierto y que tiene una comunidad grande alrededor de todo el mundo, y una de las principales ventajas de usar java que de igual manera es un software de código abierto y un lenguaje de programación, por ende, según [3] NetBeans suele dar soporte a casi todas las novedades en el lenguaje Java. Cualquier preview del lenguaje es rápidamente soportada por NetBeans.

## 4. HERRAMIENTAS EMPLEADAS POR EL MODO GRÁFICO DE JAVA

### 4.1. Toggle Buttons

Un botón es un componente muy útil y uno de los mas usados ya que nos permite manejar datos en tiempo de ejecución, un botón puede tener solo texto, una imagen o texto con imagen.

Ejemplo de cómo crear un botón en java con NetBeans.

```
import java.awt.FlowLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.*;

public class AppletJButtonPrincipal extends JApplet {

    public void init(){
        setLayout(new FlowLayout());
```



```
JButton b=new JButton("JAVA ZONE");  
b.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent arg0) {  
        JOptionPane.showMessageDialog(null, "JavaZone hace parte de la comunidad El  
Rincon de tu Casa");  
    }  
});  
add(b);  
}  
}
```

## **4.2. JLabel**

Un Label o etiqueta puede mostrar texto plano, una imagen o una imagen con un texto.

Ejemplo de creación de un jlabel en java NetBeans.

```
import javax.swing.*;  
public class Principal extends JApplet {  
    public void init(){  
        JLabel l=new JLabel("JAVA ZONE");  
        l.setHorizontalAlignment(JLabel.CENTER);  
        add(l);    } }
```

## 4.3. JRadioButton

Los JRadioButtons nos ayudan a elegir una o varias opciones en un programa. Los JRadioButton se pueden crear con una imagen, con un texto o con una imagen y texto.

Ejemplo para crear un jradiobutton en java netbeans.

```
import java.awt.FlowLayout;
import javax.swing.ButtonGroup;
import javax.swing.JApplet;
import javax.swing.JRadioButton;

public class Principal extends JApplet {
    public void init(){
        setLayout(new FlowLayout());

        JRadioButton b1=new JRadioButton();
        JRadioButton b2=new JRadioButton("Java Zone");
        JRadioButton b3=new JRadioButton("El Rincon de tu Casa",true);
        ButtonGroup grupoBotones=new ButtonGroup();
        grupoBotones.add(b1);
        grupoBotones.add(b2);
        grupoBotones.add(b3);
        add(b1);
        add(b2);
        add(b3);
    }
}
```

## 4.4. JTable

Este es uno de los componentes mas complejos que tiene Java, y se trata de una tabla similares a las de Microsoft Excel y sirve entre muchas cosas para traer datos de una base de datos.

El código para crear una Jtable en java y NetBeans es:



```
package Clases;
import java.awt.BorderLayout;
import java.awt.GridLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import javax.swing.table.TableColumn;

public class Principal extends JApplet {

    JTable tabla;
    DefaultTableModel modelo;
    JTextField t;

    public void init(){
        modelo=new DefaultTableModel(15,6);
        tabla=new JTable(modelo);
        JScrollPane scroll=new JScrollPane(tabla);
        add(scroll);
        JPanel p=new JPanel();
        JButton agregarfila=new JButton("Agregar Fila");
        agregarfila.addActionListener(new ActionListener(){

            @Override
            public void actionPerformed(ActionEvent e) {
                String d[]=new String[modelo.getColumnCount()];
                modelo.addRow(d);
                tabla.updateUI();
                repaint();
            }
        });
        t=new JTextField(15);
        JButton agregarcol=new JButton("Agregar Columna");
        agregarcol.addActionListener(new ActionListener(){

            @Override
            public void actionPerformed(ActionEvent e) {
                modelo.addColumn(t.getText());
                tabla.updateUI();
                repaint();
            }
        })
    }
}
```

```
});
```

```
p.add(agregarfila);  
p.add(new JLabel("Nombre Columna: "));  
p.add(t);  
p.add(agregarcol);  
add(p, BorderLayout.SOUTH);  
}  
}
```

## 4.5. JTabbedPane

Este componente es un contenedor el cual contiene unas pestañas, esto es muy útil ya que solo esta visible lo que esta en la pestaña actual y las otras quedan ocultas.

Para crear un JTabbedPane en java netbean es el siguiente:

```
import javax.swing.*;

public class Principal extends JApplet {

    public void init(){
        JTabbedPane pestaña=new JTabbedPane();

        JPanel p1=new JPanel();
        JLabel l1=new JLabel("Este es el panel uno");
        p1.add(l1);

        JPanel p2=new JPanel();
        JLabel l2=new JLabel("Este es el panel dos");
        p2.add(l2);

        JPanel p3=new JPanel();
        JLabel l3=new JLabel("Este es el panel tres");
        p3.add(l3);

        pestaña.addTab("Pestaña Uno", p1);
        pestaña.addTab("Pestaña Dos", p2);
```

```
pestaña.addTab("Pestaña Tres", p3);  
add(pestaña);  
}  
}
```

## 4.6. JComboBox

Un JComboBox o un cuadro combinado es un componente que nos permite seleccionar un dato.

Para crear un Jcombobox en java NetBeans utilizamos el siguiente código.

```
import javax.swing.*;  
  
public class Principal extends JApplet {  
    public void init(){  
        setLayout(new FlowLayout());  
  
        JComboBox combo1=new JComboBox();  
  
        String datos[]=new String[3];  
  
        datos[0]="Java Zone";  
  
        datos[1]="El Rincon de tu Casa";  
  
        datos[2]="Blog Java";  
  
        JComboBox combo2=new JComboBox(datos);  
  
        add(combo1);  
  
        add(combo2);  
    }  
}
```

## 4.7. JCheckBox

Un CheckBox o también conocida como una casilla de verificación, es muy usada ya que nos permite seleccionar algunas opciones, este componente puede mostrar un texto, una imagen, una imagen con un texto.

El código para crear un jcheckbox en java NetBeans es:



```
import java.awt.FlowLayout;

import javax.swing.*.*;

public class Principal extends JApplet {

    public void init(){
        setLayout(new FlowLayout());

        JCheckBox casilla1=new JCheckBox();
        JCheckBox casilla2=new JCheckBox("Java Zone");
        JCheckBox casilla3=new JCheckBox("El rincon de tu casa",true);
        add(casilla1);
        add(casilla2);
        add(casilla3);
    }
}
```

## Bibliografía

- [1] M. A. Alvarez, ¿Qué es java?, España: Contans, 2010.
- [2] Oracle, «Netbeans proyect,» Oracle, 2010. [En línea]. Available: [https://netbeans.org/index\\_es.html](https://netbeans.org/index_es.html). [Último acceso: 15 11 2017].
- [3] J. Calendamaia, «Genbetadev,» 9 01 2014. [En línea]. Available: <https://www.genbetadev.com/herramientas/netbeans-1>. [Último acceso: 15 11 2017].