



Universidad Distrital Francisco José de caldas

Tecnología en Sistematización de Datos



Interfaz Gráfica de Usuario GUI ComboBox JRadioButton

Programación Multinivel
Sonia Alexandra Pinzón Nuñez



Elementos Gráficos Swing

JComboBox



El **JComboBox**, agrega las opciones dentro de una lista, donde cada elemento se registra en una posición.

Métodos:

.getSelectedIndex(): retorna el índice del elemento seleccionado
.getSelectedItem(): retorna el objeto seleccionado.

JRadioButton



El **JRadioButton**, agrega las opciones como elementos separados, para permitir la selección de una única opción se debe agregar el control **JButtonGroup**.

Métodos:

.isSelected(): retorna un valor boolean que indica si el control esta seleccionado o no.



JComboBox



Medida

Convertir a:

```
public void actionPerformed(ActionEvent e) {  
    if(objV.getCmbMedidas() == e.getSource()){  
        if(!objV.getTxtMedida().getText().equals(" ")){  
            objM.setValor(Double.parseDouble(objV.getTxtMedida().getText()));  
            switch(objV.getCmbMedidas().getSelectedIndex()){  
                case 0:{ objV.getTxtaResp().append("El valor de " + objM.getValor()+  
                    " en cms es " + objM.convertirCM());  
                    break; }  
                case 1:{ objV.getTxtaResp().append("El valor de " + objM.getValor()+  
                    " en dms es " + objM.convertirDM());  
                    break;}  
                case 2:{ objV.getTxtaResp().append("El valor de " + objM.getValor()+  
                    " en Kms es " + objM.convertirKM());  
                    break;}  
            }  
        }  
    }  
}
```



JRadioButton



Medida

Convertir a:

☐ Centímetros

☐ Decímetros

```
public void actionPerformed(ActionEvent ae) {  
    if(objV.getBtnEjecutar() == ae.getSource()) {  
        objM.setValor(Double.parseDouble(objV.getTxtMedida().getText()));  
        if(objV.getRbtnCms().isSelected()) {  
            objV.getTxtaResp().append("\n La medida "+objM.getValor()+  
                "\n en Centimetros es : "+objM.convertirEnCM());  
        } else if(objV.getRbtnDms().isSelected()) {  
            objV.getTxtaResp().append("\n La medida "+objM.getValor()+  
                "\n en Centimetros es : "+objM.convertirEnCM());  
        }  
    }  
}
```



Ejemplo:

Convertir una medida dada en metros a centímetros, decímetros o kilómetros

Casos de Uso

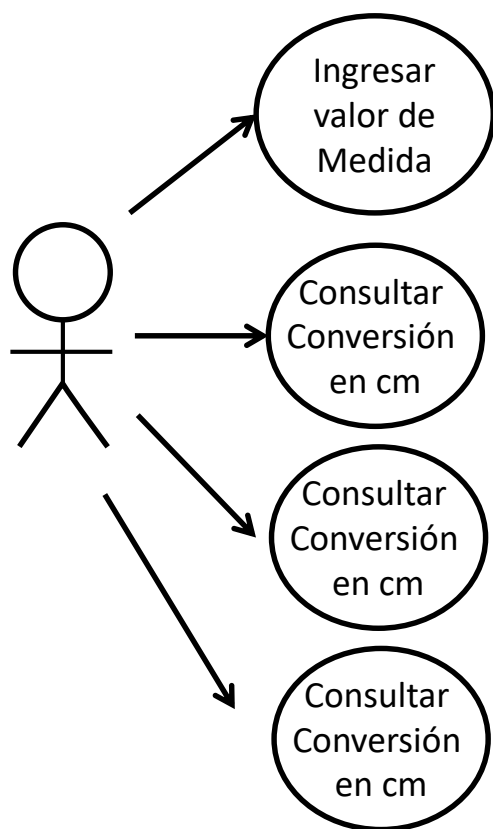
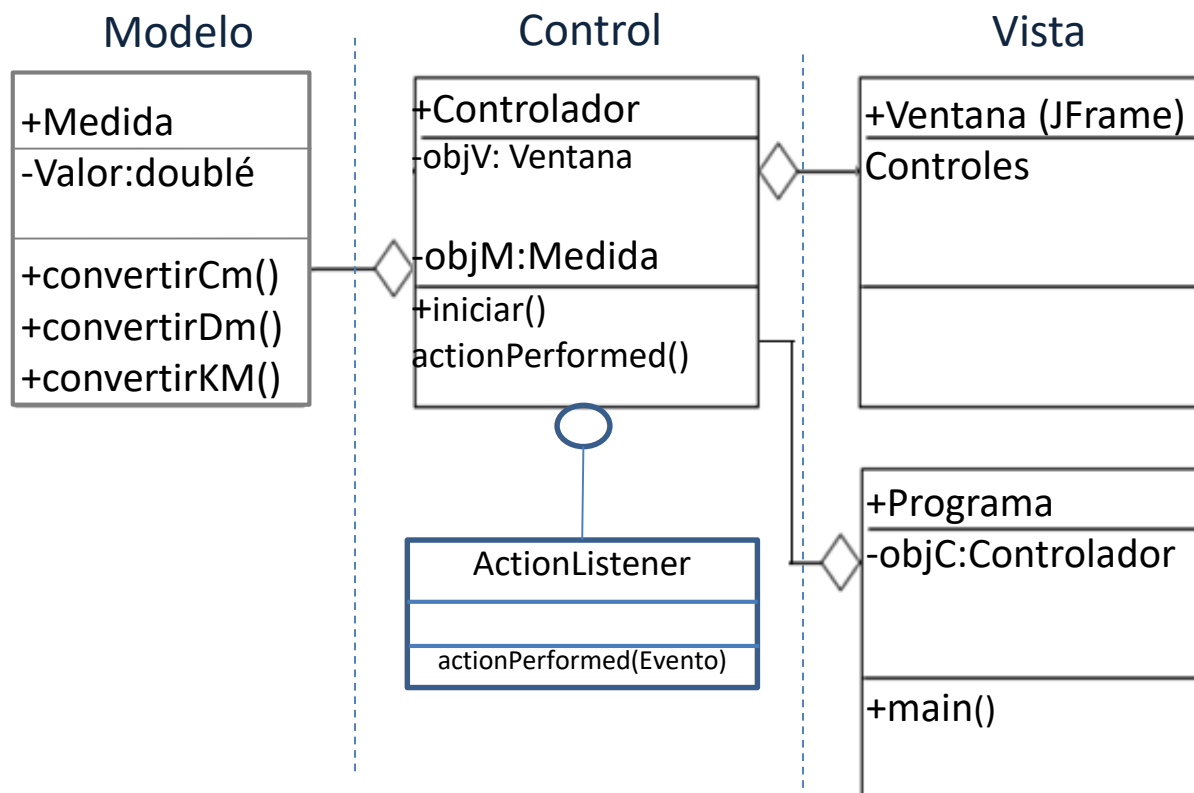


Diagrama de Clases

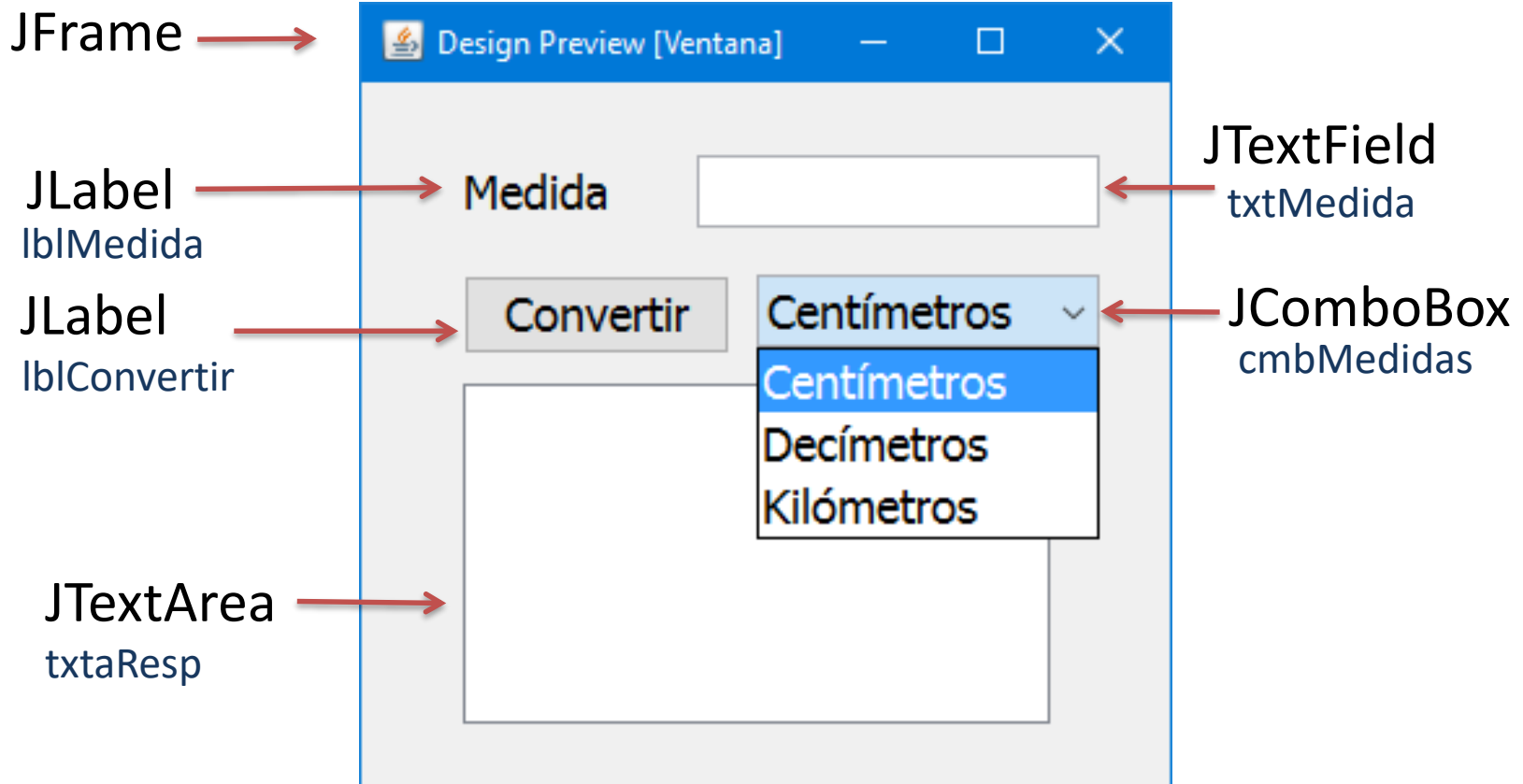




Diseño de interfaz – Formulario JFrame

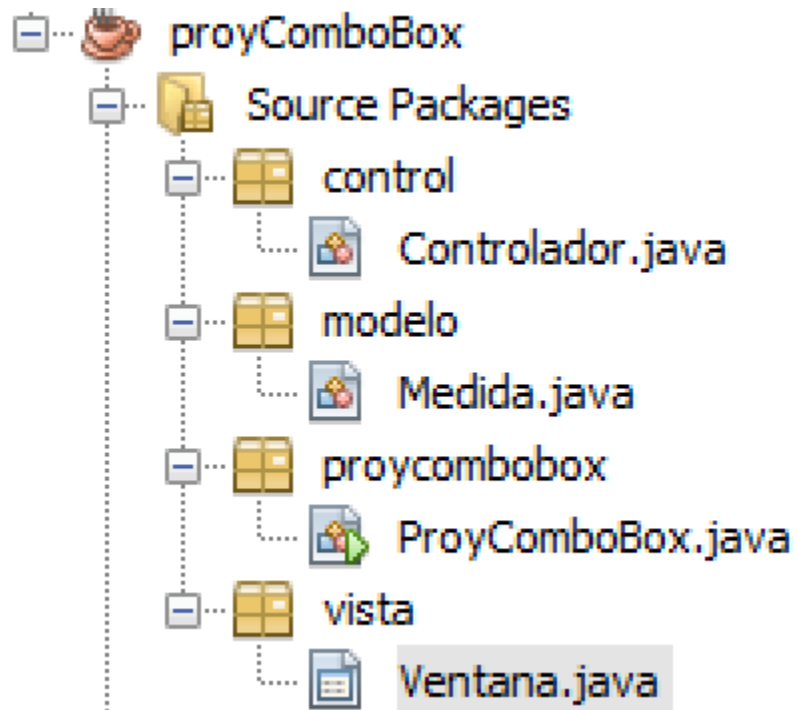
Representación gráfica de la vista de la aplicación.
(Grafico de ventanas)

Ejemplo Vista Ventana





Estructura del proyecto

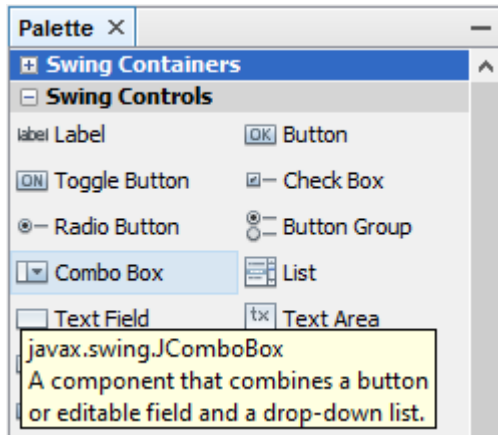




Configuración del ComboBox

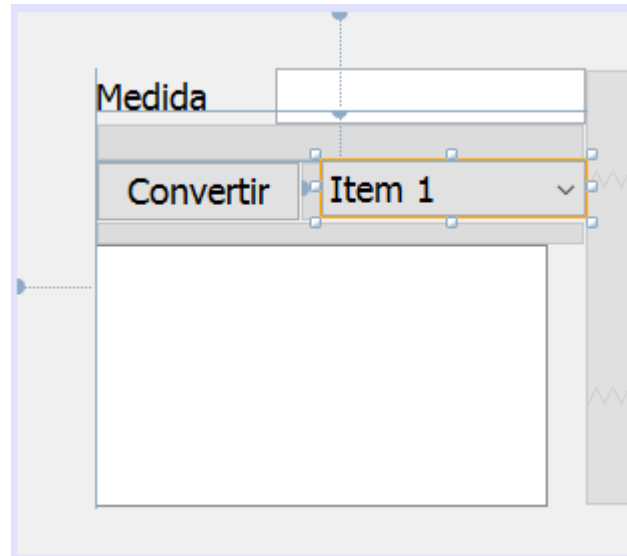
1

Seleccione el control ComboBox en la paleta de controles



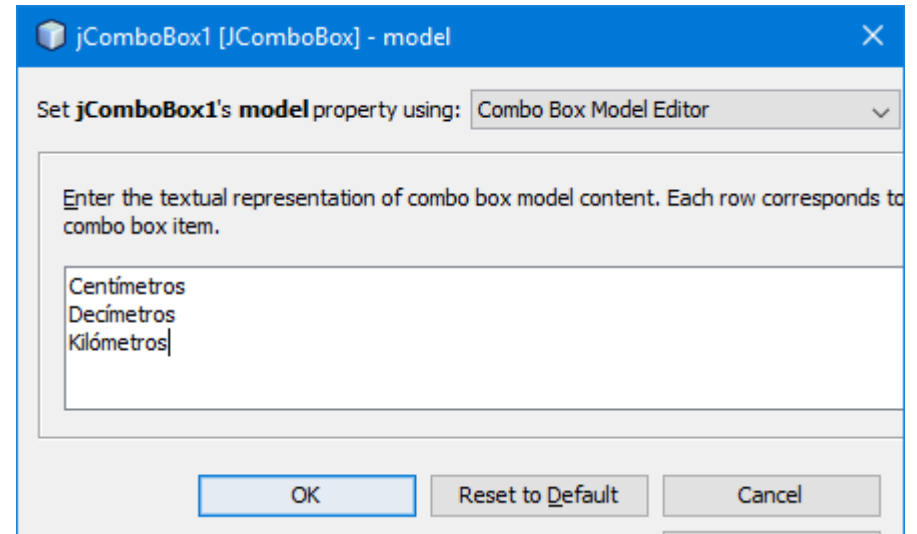
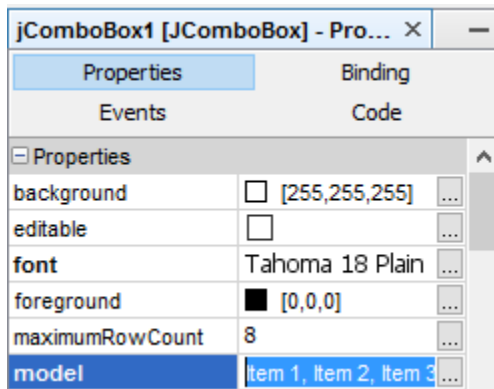
2

Ubique el control ComboBox dentro del Formulario



3

Seleccione la propiedad model del control en la caja de propiedades y agregue las opciones que debe contener.





Modelo: Clase Medida

+Medida

-Valor:double

+convertirCm()

+convertirDm()

+convertirKM()

```
package modelo;

public class Medida {
    private double valor;

    public Medida(double valor) {
        this.valor = valor;
    }

    public Medida() {
        this.valor = 0;
    }

    public double getValor() { ...3 lines }
    public void setValor(double valor) { ...3 lines }
    @Override
    public String toString() {
        return "Medida{" + "valor=" + valor + '}';
    }

    public double convertirCM(){
        return this.valor *100;
    }

    public double convertirDM(){
        return this.valor *10;
    }

    public double convertirKM(){
        return this.valor /1000;
    }
}
```



Control: Controlador

+Controlador

-objV: Ventana

-objM: Medida

+iniciar()

actionPerformed()

ChangeListener

actionPerformed(Evento)

```
package control;
import ...5 lines
public class Controlador implements ActionListener{
    Medida objM;
    Ventana objV;
    public Controlador(Medida objM, Ventana objV) {...5 lines }
    public Controlador() {
        this.objM = new Medida();
        this.objV = new Ventana();
        objV.getCmbMedidas().addActionListener(this);
    }
    public void iniciar(){
        objV.setTitle("Conversión de Medidas");
        objV.setVisible(true);
    }
    @Override
    public void actionPerformed(ActionEvent e) {
        if(objV.getCmbMedidas() == e.getSource()){
            if(!objV.getTxtMedida().getText().equals(" ")){
                objM.setValor(Double.parseDouble(objV.getTxtMedida().getText()));
                switch(objV.getCmbMedidas().getSelectedIndex()){
                    case 0:{ objV.getTxtaResp().append("El valor de " + objM.getValor() +
                        " en cms es " + objM.convertirCM());
                        break; }
                    case 1:{ objV.getTxtaResp().append("El valor de " + objM.getValor() +
                        " en dms es " + objM.convertirDM());
                        break; }
                    case 2:{ objV.getTxtaResp().append("El valor de " + objM.getValor() +
                        " en kms es " + objM.convertirKM());
                        break; }
                }
            }
        }
    }
}
```

Para obtener el contenido del Item seleccionado en el comboBox, se utilizan dos métodos:

getSelectedIndex():
obtiene el índice del ítem en la lista.

getSelectedItem():
obtiene el objeto seleccionado.



+Controlador

-objV: Ventana

-objM: Medida

+iniciar()

actionPerformed()

ChangeListener

actionPerformed(Evento)

```
public class Controlador implements ActionListener{
    Medida objM;
    frmVentana objV;

    public Controlador() {
        this.objM = new Medida();
        this.objV = new frmVentana();
        this.objV.getBtnGrupo().add(objV.getRbtnCms());
        this.objV.getBtnGrupo().add(objV.getRbtnDms());
        this.objV.getBtnEjecutar().addActionListener(this);
    }

    public void iniciar(){
        objV.setTitle("Conversor");
        objV.setLocationRelativeTo(null);
        objV.setVisible(true);
    }

    @Override
    public void actionPerformed(ActionEvent ae) {
        if(objV.getBtnEjecutar()== ae.getSource()){
            objM.setValor(Double.parseDouble(objV.getTxtMedida().getText()));
            if(objV.getRbtnCms().isSelected()){
                objV.getTxtaResp().append("\n La medida "+objM.getValor()+
                    "\n en Centimetros es : "+objM.convertirEnCM());
            } else if(objV.getRbtnDms().isSelected()){
                objV.getTxtaResp().append("\n La medida "+objM.getValor()+
                    "\n en Centimetros es : "+objM.convertirEnCM());
            }
        }
    }
}
```



Implementación y Ejecución

Vista
Programa

```
/**
 * @author Sonia Pinzón
 */
public class progConvertir {
    public static void main(String[] args) {
        Controlador objC= new Controlador();
        objC.iniciar();
    }
}
```

Vista
ejecución

Conversión de Medidas

Medida

Convertir a:

El valor de 10.0 en cms es 1000.0
El valor de 10.0 en dms es 100.0
El valor de 10.0 en Kms es 0.01



Bibliografía

- Documentación Java Oracle paquete Swing.
Disponible en:
<https://docs.oracle.com/javase/7/docs/api/javax/swing/package-summary.html>.
- Deitel y Deitel. Programación Java. Editorial McGraw Hill.
- Sonia Pinzón. Curso Programación Multinivel Moodle.