



El futuro digital
es de todos

MinTIC

Ciclo: Programación Básica

Interfaz Gráfica

»
Misión TIC 2022

xxx



**Misión
TIC 2022**



El futuro digital
es de todos

MinTIC

» Ciclo: Programación Básica

Interfaz Gráfica



- Programación Visual – Dirigida por Eventos
- Swing – JFrame en Java
- Swing -JSpinner
- Swing - JTable

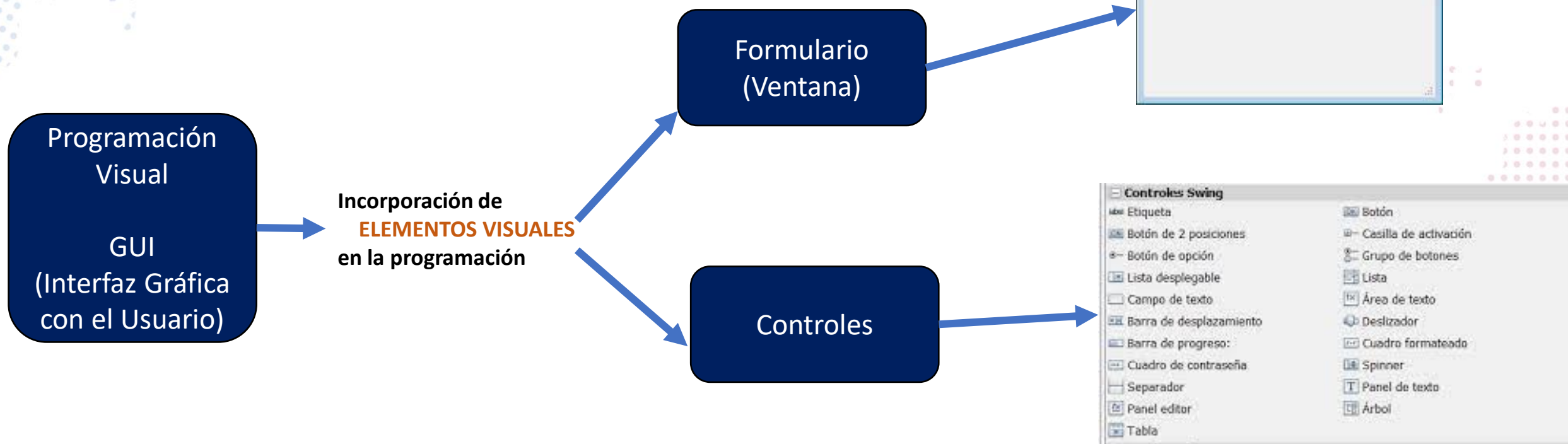


Tema: Interfaz Gráfica



Programación Visual y Dirigida por Eventos

x
x
x





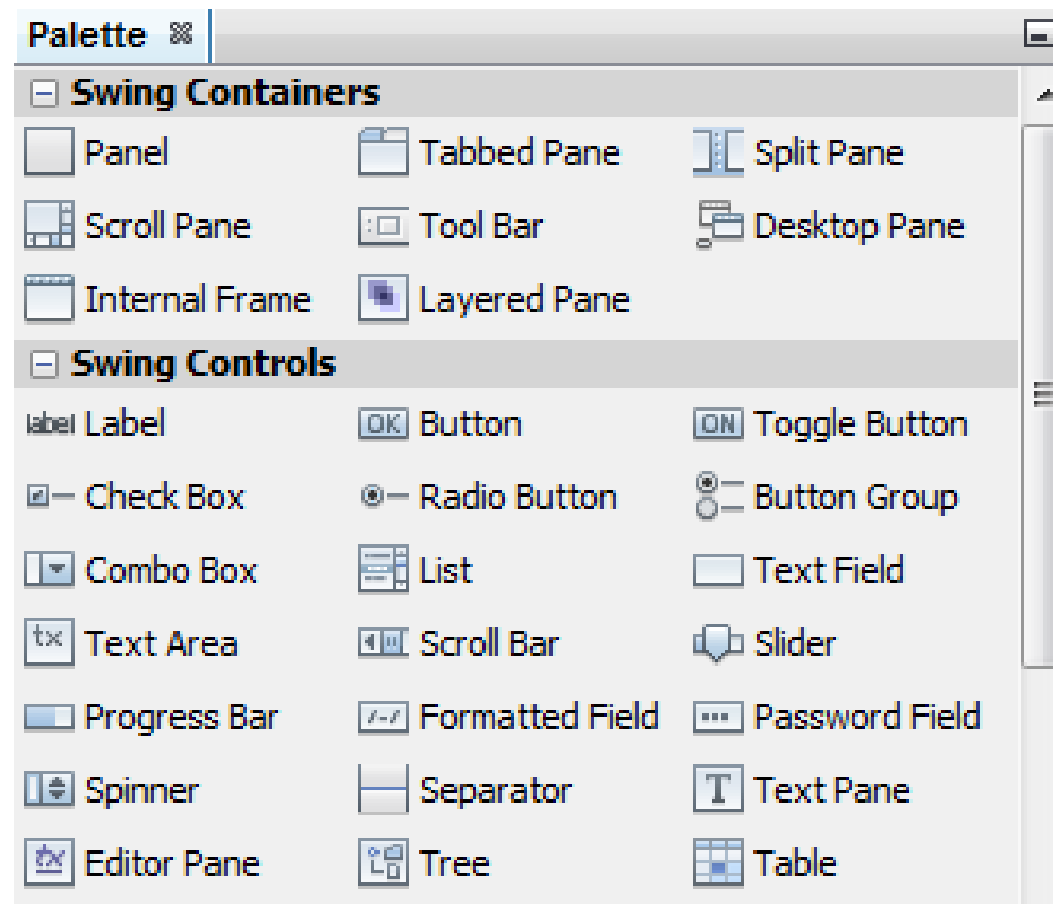
Tema: Interfaz Gráfica



Programación Visual y Dirigida por Eventos

x
x
x

Controles
visuales





Tema: Interfaz Gráfica



Programación Visual y Dirigida por Eventos

x
x
x

Programación
Dirigida por
Eventos

Paradigma de
Programación

La **estructura** como la **ejecución** de los programas van determinados por los **sucesos que ocurran en el sistema**, definidos por el usuario o que ellos mismos provoquen

Ejemplo: Hacer Clic sobre un botón



Tema: Interfaz Gráfica



Programación Visual y Dirigida por Eventos

x
x
x

Elementos
Visuales

Son Objetos
y poseen

Propiedades: Características (título, nombre, tamaño, color)

Eventos: Acciones que se realizan sobre el objeto y generan la ejecución de un proceso

Métodos: Operaciones propias del objeto



El futuro digital
es de todos

MinTIC

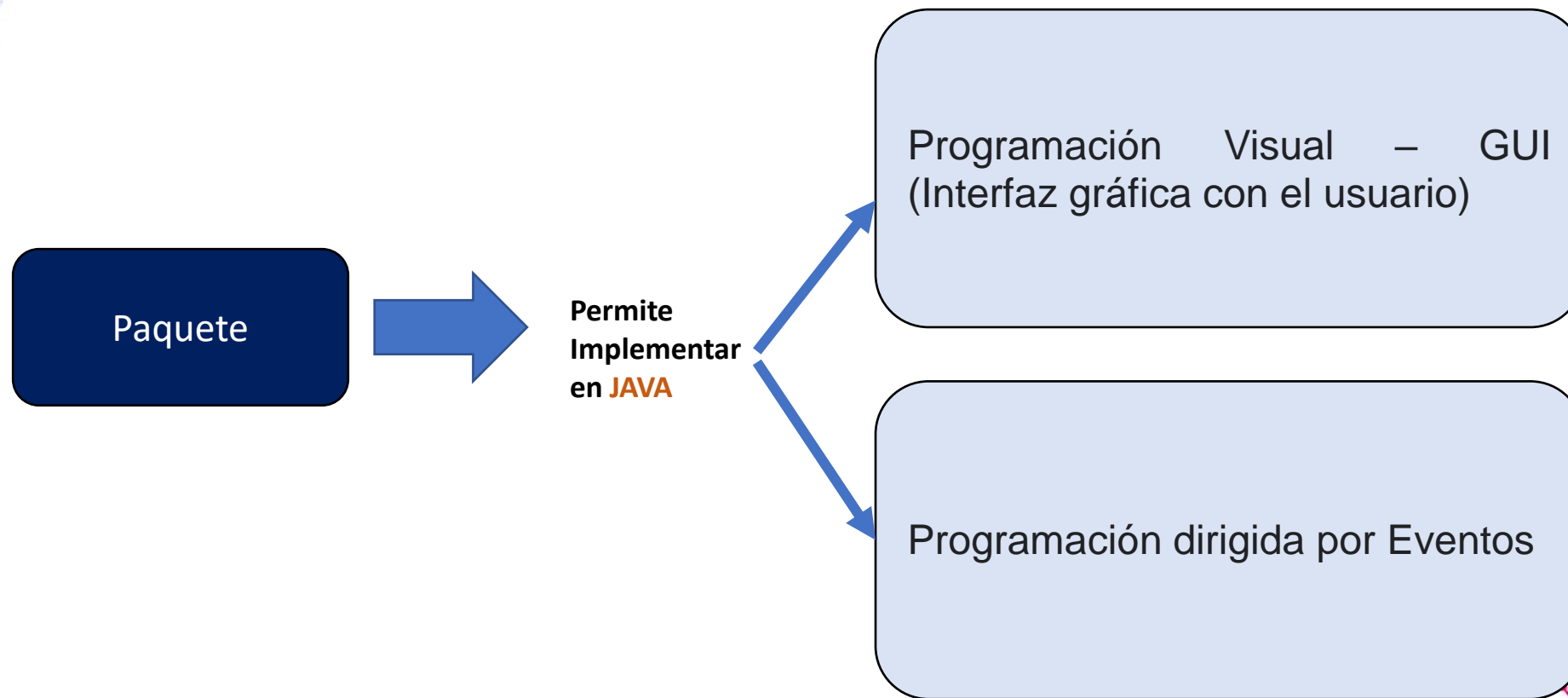


Tema: Modelamiento orientado a objetos



Swing - JFrame

x
x
x

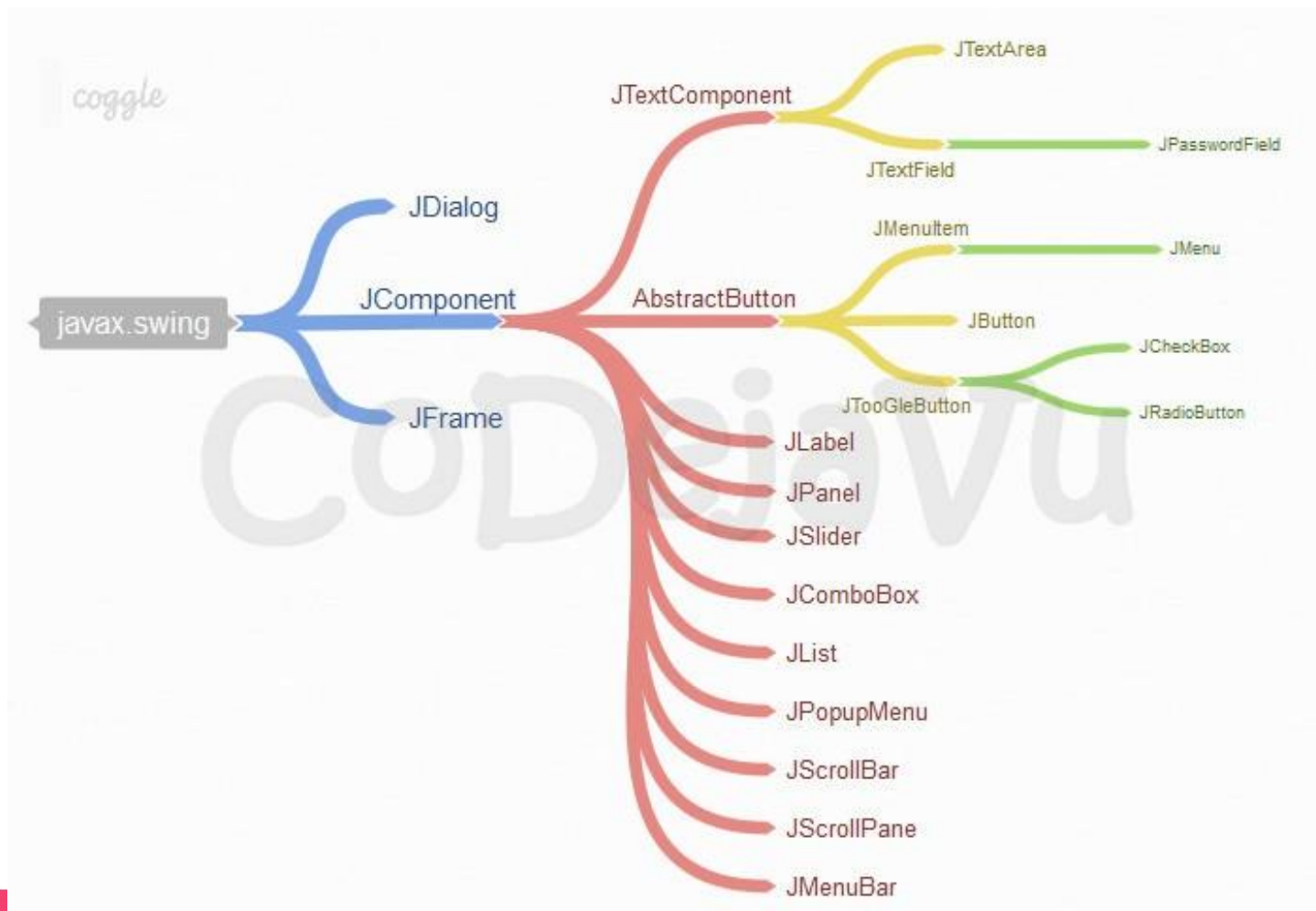




Tema: Modelamiento orientado a objetos



Swing - JFrame





El futuro digital
es de todos

MinTIC

Programación Orientada a Objetos

Ejercicio



Ejemplo: Swing -JFrame

Dada la siguiente información sobre un **docente** de una institución de educación para el trabajo:

- Documento de identidad
- Nombre
- Categoría (1= Auxiliar , 2=Asistente o 3= Especial)
- Cantidad de horas laboradas en el mes

Se pide **calcular el valor a pagar por concepto de honorarios**, de acuerdo con la siguiente indicación sobre le valor a pagar por hora de acuerdo con la categoría:

CATEGORIA	VALOR HORA
1	\$ 25.000
2	\$ 35.000
3	\$ 45.000

Realizar el programa en Java, utilizando Programación Visual dirigida por eventos, que resuelva la situación problema presentada.





El futuro digital
es de todos

MinTIC

Programación Orientada a Objetos

Ejercicios



Método

Parámetros de entrada

Categoría, horas



Calcular_honorarios()

Switch (Categoría)=> multiplicar el
valor de hora con horas laboradas



Parámetros de salida

honorarios

FUNCIÓN retorna o regresa un solo valor



Programación de la Clase

```
public class Docente {  
    private long documento;  
    private String nombre;  
    private int categoria;  
    private int horas;  
  
    public Docente() {  
    }  
    public Docente(long documento, String nombre, int categoria, int horas) {  
        this.documento = documento;  
        this.nombre = nombre;  
        this.categoria = categoria;  
        this.horas = horas;  
    }  
    public void setDocumento(long documento) {  
        this.documento = documento;  
    }  
    public void setNombre(String nombre) {  
        this.nombre = nombre;  
    }  
    public void setCategoria(int categoria) {  
        this.categoria = categoria;  
    }  
    public void setHoras(int horas) {  
        this.horas = horas;  
    }  
    public long getDocumento() {  
        return documento;  
    }  
}
```

```
    public String getNombre() {  
        return nombre;  
    }  
    public int getCategoria() {  
        return categoria;  
    }  
    public int getHoras() {  
        return horas;  
    }  
    public double calcular_honorarios() {  
        double honorarios=0;  
        switch(this.categoria){  
            case 1:honorarios=this.horas*25000;break;  
            case 2:honorarios=this.horas*35000;break;  
            case 3:honorarios=this.horas*45000;break;  
        }  
        return honorarios;  
    }  
}
```



El futuro digital
es de todos


MinTIC

Programación Orientada a Objetos

GUI – Interfaz Grafica con el Usuario Swing-Jframe

Ejercicios



 **Honorario Docente** — □ ×

Documento:

Nombre:

Categoría:

Horas laboradas:



```
private void btnhonorariosActionPerformed(java.awt.event.ActionEvent evt) {  
    // El código correspondiente al evento clic del botón  
    //Definición de variables  
    long documento;  
    int categoria, horas;  
    String nombre;  
    double honorarios;  
    String mensaje;  
    //Definición de la variable objeto  
    Docente obj_docente;  
    documento=Long.parseLong(txtdocumento.getText());  
    nombre=txtnombre.getText();  
    horas=Integer.parseInt(txthoras.getText());  
    categoria=cbocategoria.getSelectedIndex();  
    //Creación del Objeto  
    obj_docente=new Docente(documento,nombre,categoria, horas);  
    honorarios=obj_docente.calcular_honorarios();  
    mensaje="Honorarios: "+honorarios;  
    JOptionPane.showMessageDialog(null, mensaje);  
}
```



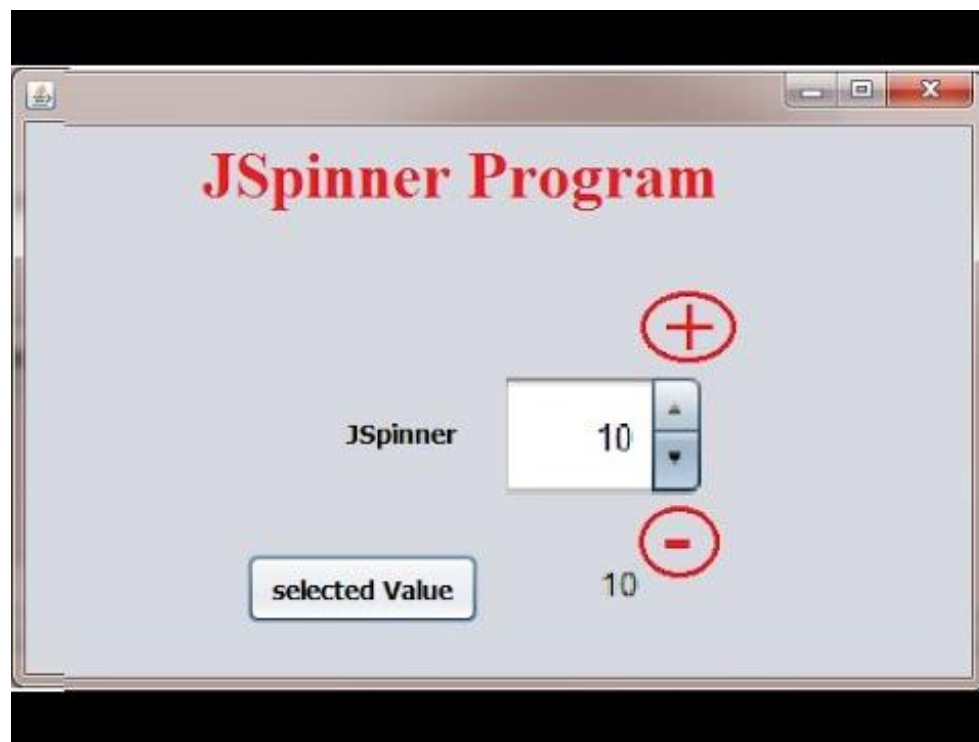
Tema: Modelamiento orientado a objetos



Swing - JSpinner

x
x
x

El **JSpinner** es un componente **java** en el que se presenta una caja de texto con dos flechitas en el lado derecho, una hacia arriba y la otra hacia abajo. En el campo de texto se muestra un valor. Con las flechitas arriba y abajo podemos incrementar o decrementar ese valor.





Programación Orientada a Objetos

Ejercicio

Ejemplo: Swing -JSpinner



Una institución Universitaria, desea realizar la proyección de ingresos y costos para grupos (de 8 estudiantes por grupo) para un **diplomado** virtual sobre TIC en educación, para lo cual suministra la siguiente información sobre ingresos y gastos:

Con respecto a los ingresos, se debe capturar el **valor del diplomado** por estudiante.

Con respecto a los costos y gastos, debe ingresar:

- **Costo por concepto de docencia** de un grupo
- **Costo por el uso de servicios de plataforma** de un grupo.
- **Costo de administración** para cada grupo

También, en la proyección se debe tener el **número de grupos**.

Realizar el programa en Java, utilizando Programación Visual dirigida por eventos, que permita proyectar los ingresos y costos del diplomado, donde se pueda manejar con una JSpinner la cantidad de grupos. El programa debe mostrar los **ingresos y costos** del diplomado.



Proyeccion_Diplomado

-grupos
-valor_diplomado
-docencia
-plataforma
-admon

+setGrupos(grupos)
+getGrupos()
+setValor_diplomado(valor)
+getValor_diplomado()
+setDocencia(docencia)
+getDocencia()
+setPlataforma(plataforma)
+getPlataforma()
+setAdmon(admon)
+getAdmon()
+calcular_ingresos()
+calcular_costos()



Programación Orientada a Objetos

Ejercicios



Método

Parámetros de entrada

Valor_diplomado,
grupos



Calcular_ingresos()

$\text{Ingresos} = \text{valor_diplomado} * \text{grupos} * 8$

Parámetros de salida

ingresos



FUNCIÓN retorna o regresa un solo valor



El futuro digital
es de todos

MinTIC

Programación Orientada a Objetos

Ejercicios



Método

Parámetros de entrada

Docencia,plataforma,admon



Calcular_costos()

$\text{Costos} = (\text{docencia} + \text{plataforma} + \text{admon}) * \text{grupos}$



Parámetros de salida

costos

FUNCIÓN retorna o regresa un solo valor



```
public class Diplomado {  
    private double matricula;  
    private double docencia;  
    private double plataforma;  
    private double admon;  
    private int grupos;  
  
    public Diplomado() {  
    }  
    public Diplomado(double matricula, double docencia, double plataforma, double admon, int grupos) {  
        this.matricula = matricula;  
        this.docencia = docencia;  
        this.plataforma = plataforma;  
        this.admon = admon;  
        this.grupos = grupos;  
    }  
    public void setMatricula(double matricula) {  
        this.matricula = matricula;  
    }  
    public void setDocencia(double docencia) {  
        this.docencia = docencia;  
    }  
    public void setPlataforma(double plataforma) {  
        this.plataforma = plataforma;  
    }  
}
```



```
public void setAdmon(double admon) {  
    this.admon = admon;  
}  
public void setGrupos(int grupos) {  
    this.grupos = grupos;  
}  
public double getMatricula() {  
    return matricula;  
}  
public double getDocencia() {  
    return docencia;  
}  
public double getPlataforma() {  
    return plataforma;  
}  
public double getAdmon() {  
    return admon;  
}  
public int getGrupos() {  
    return grupos;  
}
```



```
public double calcular_ingresos() {  
    double ingresos;  
    ingresos=this.matricula*this.grupos*8;  
    return ingresos;  
}  
public double calcular_gastos() {  
    double gastos;  
    gastos=(this.docencia+this.plataforma+this.admon)*this.grupos;  
    return gastos;  
}  
}
```



El futuro digital
es de todos

MinTIC

Programación Orientada a Objetos

GUI – Interfaz Grafica con el Usuario

Ejercicios



Proyección Ingresos-Costos Diplomado

Ingresos

Valor Diplomado:

Costos

Docencia:

Plataforma:

Administración:

Cantidad de grupos:



```
private void btnProyeccionActionPerformed(java.awt.event.ActionEvent evt) {  
    // Definición de variables  
    double matricula, docencia, plataforma, admon, ingresos, costos;  
    int grupos;  
    String mensaje;  
    //Definición variable objeto  
    Diplomado obj_diplomado;  
    matricula=Double.parseDouble(txtValor_matricula.getText());  
    docencia=Double.parseDouble(txtDocencia.getText());  
    plataforma=Double.parseDouble(txtPlataforma.getText());  
    admon=Double.parseDouble(txtAdmon.getText());  
    grupos=(int) jspGrupos.getValue();  
    //Creación Objeto Diplomado  
    obj_diplomado=new Diplomado(matricula, docencia, plataforma, admon, grupos);  
    ingresos=obj_diplomado.calcular_ingresos();  
    costos=obj_diplomado.calcular_gatos();  
    lblIngresos.setText(String.valueOf(ingresos));  
    lblCostos.setText(String.valueOf(costos));  
    mensaje="Ingresos: "+ingresos+" Costos: "+costos;  
    JOptionPane.showMessageDialog(null, mensaje);  
}
```



Tema: Modelamiento orientado a objetos



Swing - JTable



Como programadores, sabemos muy bien que la presentación de **datos tabulados** es una de las **tareas más comunes que se presentan al momento de crear interfaces gráficas**; desde la simple tabla que permite únicamente **mostrar el resultado** de una consulta, hasta las que permiten **editar** directamente el contenido de cada celda, ordenar las columnas, personalizar su apariencia, etc. Todas las tareas antes descritas, y muchas otras, son posibles de realizar utilizando la clase JTable; por supuesto, mientras más complejo sea el requerimiento a cubrir, se requerirá en igual medida utilizar más métodos o recursos de la clase.



El futuro digital
es de todos

MinTIC



Tema: Modelamiento orientado a objetos



Swing - JTable



Id:

Nombres:

Teléfono:

ID	Nombres	Teléfono
1	José	0899999
2	María	0877333
3	Luis	0980034
4	Luna	0723341



El futuro digital
es de todos

MinTIC

Programación Orientada a Objetos

Ejemplo: Swing - JTable

Ejercicio



Dada la siguiente información sobre los suscriptores de una revista:

- Nombre
- Genero (Masculino, Femenino)

Realizar el programa en Java, utilizando Swing - JTable, que permita el ingreso de información a la tabla presentada sobre los suscriptores.



El futuro digital
es de todos

MinTIC

Programación Orientada a Objetos

GUI – Interfaz Grafica con el Usuario

Ejercicios



Ingreso Datos

Nombre Completo:

Género:

Nombre completo	Género
-----------------	--------



```
package jtable_ejemplo;
```

```
import javax.swing.JOptionPane;  
import javax.swing.table.DefaultTableModel;
```

```
public class UsoTable extends javax.swing.JFrame {  
    DefaultTableModel tabla;
```

```
    public UsoTable() {  
        initComponents();  
        this.setLocationRelativeTo(this);
```

```
        tabla= new DefaultTableModel();  
        //Enlazar Tabla al JTable  
        tabla=(DefaultTableModel) tbDatos.getModel();  
    }
```



```
private void btnAgregarActionPerformed(java.awt.event.ActionEvent evt) {  
    // Definiición de variables  
    String datos[]=new String[2];  
  
    //Pasamos los datos de la interfaz a un vector  
    datos[0]=txtNombre.getText();  
    datos[1]=dbGenero.getSelectedItem().toString();  
    //Agregamos a la Tabla  
    tabla.addRow(datos);  
    txtNombre.setText("");  
    dbGenero.setSelectedIndex(0);  
}
```



Programación Eventos – Botón Eliminar

```
private void btnLimpiarActionPerformed(java.awt.event.ActionEvent evt) {  
    int fila;  
    fila=tbDatos.getSelectedRow();  
    if(fila>=0){  
        tabla.removeRow(fila);  
    }  
    else{  
        JOptionPane.showMessageDialog(null, "Seleccione Registro en la Tabla");  
    }  
}
```



Programación Eventos – Botón Modificar

```
private void btnmodificarActionPerformed(java.awt.event.ActionEvent evt) {  
    int fila;  
    String nombre;  
    fila=tbDatos.getSelectedRow();  
    if(fila>=0){  
        if (txtNombre.getText().isEmpty()){  
            JOptionPane.showMessageDialog(null, "Ingrese nombre a modificar");  
        }  
        else{  
            nombre=txtNombre.getText();  
            tabla.setValueAt(nombre, fila, 0);  
        }  
    }  
    else{  
        JOptionPane.showMessageDialog(null, "Seleccione Registro en la Tabla");  
    }  
}
```




El futuro digital
es de todos

MinTIC

Ciclo: Programación Básica

Interfaz Gráfica

»

Misión TIC 2022

xxx



**Misión
TIC 2022**