

Swing 08 – Introducción a las Tablas

En esta práctica se revisarán los conceptos básicos sobre el manejo de Tablas. Se comenzará creando un nuevo proyecto llamado **SW08** sin crear una clase Principal. Crear también un paquete llamado **uam.pvoe.sw.tablas.formas** y otro llamado **uam.pvoe.sw.tablas.operaciones**. En el paquete para las formas, se creará un nuevo **Frame** llamado **FrmTablas**.

Configurarla con las siguientes propiedades:

- Ancho Máximo: **800 x 600**
- Ancho Mínimo: **800 x 600**
- **No** permitir redimensionar
- Título: **Introducción a Tablas**

Se comenzará colocando una etiqueta (**lblTitulo**) con el texto “Introducción al Manejo de Tablas”. Se colocará una etiqueta (**lblTablaSencilla**) con el texto “Tabla Sencilla”. Debajo se colocará la primera tabla, un componente **JTable** al que se le llamará **tblTablaSencilla** y a su contenedor, **scrlTablaSencilla**.

Las tablas se construyen a partir de modelos, se invocará un método para configurar ésta tabla:

FrmTablas.java

```
public FrmTablas() {  
    initComponents();  
    configurarTablaBasica();  
}
```

FrmTablas.java

```
private void configurarTablaBasica(){  
    DefaultTableModel modelo = new DefaultTableModel(0,0);  
    tblTablaBasica.setModel(modelo);  
}
```

También es posible llenar una tabla a partir de arreglos (*arrays*) o elementos del tipo **Vector**.

FrmTablas.java

```
private void configurarTablaBasica(){  
  
    Vector columnas = new Vector();  
    columnas.add("Columna A");  
    columnas.add("Columna B");  
    columnas.add("Columna C");  
    columnas.add("Columna D");  
    columnas.add("Columna E");  
}
```

```

Vector filas = new Vector();
Vector fila = new Vector();

fila.add("1_1");
fila.add("1_2");
fila.add("1_3");
fila.add("1_4");
fila.add("1_5");
filas.add(fila);

fila = new Vector();
fila.add("2_1");
fila.add("2_2");
fila.add("2_3");
fila.add("2_4");
fila.add("2_5");
filas.add(fila);

DefaultTableModel modelo = new DefaultTableModel(0,0);
modelo.setDataVector(filas, columnas);

tblTablaBasica.setModel(modelo);
}

```

Se crean dos vectores, el primero representa los nombres de las columnas (encabezados) y el segundo representa el contenido (un **Vector** de **Vectores**), se crea un vector (la fila) y en él se van agregando otros vectores (las celdas). Finalmente se agregan tanto las filas como las columnas al modelo y se le asigna a la tabla.

Construcción Dinámica de Tablas

Se llenará una tabla de manera dinámica indicando cuántos renglones y cuantas columnas se desean. Primero se colocará una etiqueta (**lblTablaDinamica**) con el texto “Tabla Dinámica”, se tendrá una tabla (**tblDinamica**) y su scroll (**scrlTablaDinamica**). Se colocarán dos cajas de texto (**txtnFilas** y **txtnColumnas**) acompañadas de sus etiquetas (**lblFilas** y **lblColumnas**) además de un botón para generar la tabla (**btnGenerar**) con la leyenda Generar.

FrmTablas.java

```

public FrmTablas() {
    initComponents();
    configurarTablaBasica();
    iniciarTablaDinamica();
}

```

FrmTablas.java

```
private void iniciarTablaDinamica(){
    DefaultTableModel modelo = new DefaultTableModel(0,0);
    tblDinamica.setModel(modelo);
}
```

Posteriormente, al leer el número de filas y columnas (por el momento sin validación) se generará la tabla correspondiente

FrmTablas.java

```
private void btnGenerarActionPerformed(java.awt.event.ActionEvent evt) {
    int nCols = Integer.parseInt(txtnCols.getText());
    int nFilas = Integer.parseInt(txtnFilas.getText());

    generaTablaDinamica(nFilas,nCols);
}
```

FrmTablas.java

```
private void generaTablaDinamica(int nFilas, int nColumnas){
    DefaultTableModel modelo = new DefaultTableModel();
    OperacionesTabla operaciones = new OperacionesTabla();
    Vector filas = operaciones.generaFilas(nFilas, nColumnas);
    Vector columnas = operaciones.generaColumnas(nColumnas);
    modelo.setDataVector(filas, columnas);
    tblDinamica.setModel(modelo);
}
```

Será necesario crear la clase *OperacionesTabla* en el paquete de operaciones e implementar los métodos necesarios.

OperacionesTabla.java

```
public class OperacionesTabla {

    public Vector generaColumnas(int n){
        Vector columnas = new Vector();
        for(int i=0;i<n;i++){
            columnas.add("Columna " + (i+1));
        }

        return columnas;
    }

    public Vector generaFilas(int nFilas, int nColumnas){
        Vector filas = new Vector();
    }
```

```

    Vector fila = new Vector();

    for(int i=0;i<nFilas;i++){
        for(int j=0;j<nColumnas;j++){
            fila.add((i+1) + "_" + (j+1));
        }
        filas.add(fila);
        fila = new Vector();
    }
    return filas;
}
}

```

Agregando Filas y Columnas

Es posible agregar filas y columnas a una tabla ya existente. Se comenzará agregando una etiqueta (*lblFilasColumnas*) con la leyenda “**Agregando Filas y Columnas**”. Se colocará una tabla (*tblFilasColumnas*) junto con su scroll (*scrlFilasColumnas*), un botón (*btnNuevaFila*), un botón (*btnNuevaColumna*) junto con un campo de texto (*txtNombreColumna*) y una etiqueta (*lblNombreColumna*) con la leyenda “**Nombre de la Columna**”.

Lo primero será “limpiar” el modelo de la tabla

FrmTablas.java

```

public FrmTablas() {
    initComponents();
    configurarTablaBasica();
    iniciarTablaDinamica();
    iniciarTablaFilasColumnas();
}

```

FrmTablas.java

```

private void iniciarTablaFilasColumnas(){
    DefaultTableModel modelo = new DefaultTableModel(0,0);
    tblFilasColumnas.setModel(modelo);
}

```

Después se agregarán las operaciones para poder insertar una nueva fila al presionar el botón de nueva fila. Esto se hará solamente si hay columnas ya insertadas.

FrmTablas.java

```
private void btnNuevaFilaActionPerformed(java.awt.event.ActionEvent evt) {  
    DefaultTableModel modelo = (DefaultTableModel) tblFilasColumnas.getModel();  
    int nColumnas = tblFilasColumnas.getColumnCount();  
    if(nColumnas > 0){  
        Vector nuevaFila = new Vector();  
        modelo.addRow(nuevaFila);  
        tblFilasColumnas.setModel(modelo);  
    }  
}
```

Lo primero es recuperar el modelo de la tabla y agregarle una nueva fila a través de un objeto de tipo **Vector**, una vez agregado al modelo, se asigna éste nuevamente a la tabla.

El siguiente bloque de código muestra la funcionalidad para agregar columnas.

FrmTablas.java

```
private void btnNuevaColumnaActionPerformed(java.awt.event.ActionEvent evt) {  
    DefaultTableModel modelo = (DefaultTableModel) tblFilasColumnas.getModel();  
  
    String nombreColumnas = txtNombreColumna.getText();  
  
    if(nombreColumnas.length()==0){  
        nombreColumnas = "COLUMNA";  
    }  
  
    Vector nuevaColumna = new Vector();  
    modelo.addColumn(nombreColumnas, nuevaColumna);  
  
    tblFilasColumnas.setModel(modelo);  
    txtNombreColumna.setText("");  
}
```

Nuevamente se recupera el modelo, se crea un vector que es la columna y se le asigna el nombre que se leyó en la caja de texto o se le da uno por defecto. Una vez asignada la nueva columna al modelo, se vuelve a asignar éste a la tabla.

Con ésto finaliza la práctica de introducción a las tablas, específicamente a su generación y estructura.