

Swing 03 – Introducción al Uso de Botones

En esta práctica se revisarán los conceptos básicos sobre el manejo de botones y lectura de datos. Se comenzará creando un nuevo proyecto llamado **SW03** sin crear una clase Principal. Crear también un paquete llamado **uam.pvoe.sw.botones.formas** y otro llamado **uam.pvoe.sw.botones.modelo**. En el paquete para las formas, se creará un nuevo **Frame** llamado **FrmBotonesTexto**.

Configurarla con las siguientes propiedades:

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

Agregar elementos para que tenga el siguiente aspecto:



Bienvenido al Sistema de Registro

Introduce los datos que se te solicitan

Nombre

Primer Apellido

Segundo Apellido

RFC *

Dirección

Limpiar Campos Registrar

Los nombres de las variables serán:

```
private javax.swing.JButton btnRegistrar;  
private javax.swing.JScrollPane jScrollPane1;  
private javax.swing.JSeparator jSeparator1;  
private javax.swing.JLabel lblBienvenida;  
private javax.swing.JLabel lblDireccion;  
private javax.swing.JLabel lblInstrucciones;  
private javax.swing.JLabel lblLimpiar;  
private javax.swing.JLabel lblNombre;  
private javax.swing.JLabel lblPrimerApellido;  
private javax.swing.JLabel lblRFC;  
private javax.swing.JLabel lblSegundoApellido;  
private javax.swing.JTextArea txtDireccion;  
private javax.swing.JTextField txtNombre;  
private javax.swing.JTextField txtPrimerApellido;  
private javax.swing.JTextField txtRFC;  
private javax.swing.JTextField txtSegundoApellido;
```

Agregando Eventos

Se agregará un evento a la etiqueta para limpiar datos para que al dar clic sobre ella, los campos se limpien.

```
private void lblLimpiarMouseClicked(java.awt.event.MouseEvent evt) {  
    txtNombre.setText("");  
    txtPrimerApellido.setText("");  
    txtSegundoApellido.setText("");  
    txtRFC.setText("");  
    txtDireccion.setText("");  
  
}
```

El objetivo de la práctica será que al presionar el botón de registrar se recuperen los datos de los campos para desplegarlos en consola. También se validará que el campo de RFC tenga un valor.

Para esto, en el paquete *uam.pvoe.sw.botones.modelo*, se creará una clase llamada *DatosPersona* y otra llamada *AdministrarPersona*

DatosPersona.java

```
package uam.pvoe.sw.btn.modelo;  
  
public class DatosPersona {  
  
    private String nombre;  
    private String primerApellido;  
    private String segundoApellido;  
    private String RFC;  
    private String direccion;  
  
    public DatosPersona (String n, String pA, String sA, String rfc, String dir){  
  
        nombre = n;  
        primerApellido = pA;  
        segundoApellido = sA;  
        RFC = rfc.toUpperCase();  
        direccion = dir;  
  
    }  
  
}
```

```
public String getNombre() {
    return nombre;
}

public void setNombre(String nombre) {
    this.nombre = nombre;
}

public String getPrimerApellido() {
    return primerApellido;
}

public void setPrimerApellido(String primerApellido) {
    this.primerApellido = primerApellido;
}

public String getSegundoApellido() {
    return segundoApellido;
}

public void setSegundoApellido(String segundoApellido) {
    this.segundoApellido = segundoApellido;
}

public String getRFC() {
    return RFC;
}

public void setRFC(String RFC) {
    this.RFC = RFC;
}

public String getDireccion() {
    return direccion;
}

public void setDireccion(String direccion) {
    this.direccion = direccion;
}
}
```

```
public String toString(){
    String cadena = "";
    cadena = nombre + " " + primerApellido + " " + segundoApellido + " " + RFC + " " + direccion;
    return cadena;
}
```

AdministrarPersona.java

```
package uam.pvoe.sw.btn.modelo;

public class AdministrarPersona {

    public boolean validarRFC(String rfc){
        if(rfc.length() == 10)
            return true;
        else
            return false;
    }

    public void registrarPersona(DatosPersona persona){
        System.out.println("Los Datos de la Persona registrada son: ");
        System.out.println(persona.toString());
    }
}
```

Validando el RFC

Una validación básica consiste en verificar que se tengan solamente 10 dígitos, esto se puede hacer en la misma clase que es la interfaz o en una clase separada. Lo correcto es que se realice en una clase por separado.

Primero hay que crear el evento que registre una acción en el botón, como es el evento por defecto de éste tipo de elementos, basta con dar doble clic sobre él en el diseño de la pantalla para que se creé el método correspondiente.

BotonesTexto.java

```
private void btnRegistrarActionPerformed(java.awt.event.ActionEvent evt) {

    AdministrarPersona administrar = new AdministrarPersona();
    String nombre = txtNombre.getText();
    String primerApellido = txtPrimerApellido.getText();
    String segundoApellido = txtSegundoApellido.getText();
```

```

String RFC = txtRFC.getText();
String direccion = txtDireccion.getText();

if(RFC.length() != 10){
    System.out.println("RFC NO Válido en la Forma");
}else{
    System.out.println("RFC Válido en la Forma");
}

if(administrar.validarRFC(RFC)){
    System.out.println("RFC Válido en la Clase AdministrarPersona");
}else{
    System.out.println("RFC NO Válido en la Clase AdministrarPersona");
}
}

```

Registrando la Persona

Para “registrar” a la persona, solo se imprimirán en consola los datos de la misma, una vez que se haya validado que el RFC es correcto. La validación que se usará es la realizada en una clase externa.

BotonesTexto.java

```

private void btnRegistrarActionPerformed(java.awt.event.ActionEvent evt) {

    AdministrarPersona administrar = new AdministrarPersona();
    String nombre = txtNombre.getText();
    String primerApellido = txtPrimerApellido.getText();
    String segundoApellido = txtSegundoApellido.getText();
    String RFC = txtRFC.getText();
    String direccion = txtDireccion.getText();

    if(administrar.validarRFC(RFC)){
        System.out.println("RFC Válido en la Clase AdministrarPersona");
        DatosPersona persona = new DatosPersona(nombre, primerApellido, segundoApellido, RFC,
direccion);
        administrar.registrarPersona(persona);
    }else{
        System.out.println("RFC NO Válido en la Clase AdministrarPersona");
    }
}
}

```