



INSTITUTO POLITÉCNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA DE
INGENIERÍA Y CIENCIAS SOCIALES ADMINISTRATIVAS
(UPIICSA)



TAREA

Saludador2 Applet

Applet Saludador2

2NV40

Fundamentos de Programación Orientada a Objetos.

VÁZQUEZ JUÁREZ BRANDON.

Prof. LUQUE MARQUEZ FERNANDO.

CODIGO

Encontré 2 posibles soluciones al punto número 6 del ejercicio el cual consiste en que el TextField del mensaje debe ser de solo lectura.

La primera solución fue utilizar el metodo `setEditable(false)` el cual impide que el usuario pueda modificar el texto y elimina todos los eventos del mouse. Sin embargo este metodo establece un color de fondo un poco más oscuro, lo que no me gusto, trate cambiar el color del fondo con `setBackground(Color.WHITE)` pero no funciono así que decidí usar dos métodos, `setCursor(null)` al que le pase un parámetro null para que no cambiara el mouse al pasar sobre el TextField y el metodo `setFocusable(false)`, para evitar que el usuario pudiera acceder al control del componente, con lo que obtuve el mismo efecto que con el metodo `setEditable(false)` pero, sin el color que establece el metodo automaticamente.

// Diferente saludo según la hora e implementación de eventos.

```
import java.applet.Applet;
```

```
import java.awt.*;
```

```
import java.awt.event.*;          // el import no es recursivo
```

```
import java.util.*;
```

```
public class Saludador2 extends Applet implements ActionListener,MouseListener{
```

```
    Button btnOK,btnOK1;
```

```
    TextField tfmensaje,tfnombre;
```

```
    Label lblnom;
```

```
@Override
```

```
    public void init(){
```

```
        System.out.println("entro al metodo init");
```

```
        showStatus("Bienvenido");
```

```
        // instanciar componentes
```

```
        btnOK = new Button("Saludar");
```

```
        btnOK1 = new Button("limpiar");
```

```
        tfmensaje = new TextField(30);
```

```
tfnombre = new TextField(15);
lblnom = new Label("Introduce tu nombre");
//Llamo métodos de los componentes
//tfmensaje.setEditable(true);
tfmensaje.setCursor(null);
tfmensaje.setFocusable(false);
//agrego los eventos a los botones
btnOK.addActionListener(this);
btnOK.addMouseListener(this);
btnOK1.addActionListener(this);
btnOK1.addMouseListener(this);
//agrego los componentes
add(lblnom);
add(tfnombre);
add(btnOK);
add(tfmensaje);
add(btnOK1);
}
@Override
// necesario para ser ActionListener
public void actionPerformed(ActionEvent e) {
    if (e.getSource()==btnOK){
        String saludo,nombre;
        nombre=tfnombre.getText();
        Date d= new Date();
        if(d.getHours() < 12)
            saludo = "!Buenos dias!";
        else
```

```
        if(d.getHours() < 19)
            saludo = "!Buenas tardes!";
        else
            saludo = "!Buenas noches!";
        tfmensaje.setText(saludo+" "+nombre);
    }else if(e.getSource()== btnOK1){
        tfmensaje.setText("");
        tfnombre.setText("");
    }
}
```

//metodos abstractos de la interface MouseListener

```
@Override
public void mouseClicked(MouseEvent e) {
}

@Override
public void mousePressed(MouseEvent e) {
}

@Override
public void mouseReleased(MouseEvent e) {
}

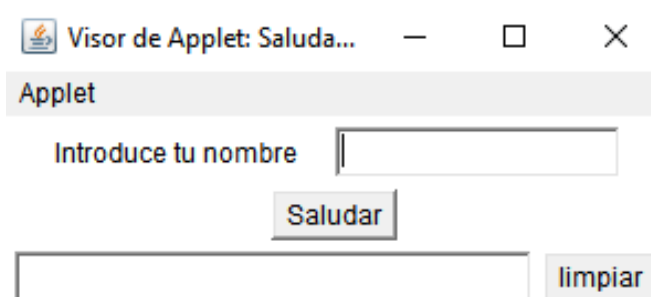
@Override
public void mouseEntered(MouseEvent e) {
    if (e.getSource()==btnOK) {
        showStatus("Este boton saluda");
    }else if (e.getSource()==btnOK1){
        showStatus("Este boton limpia");
    }
}
```

@Override

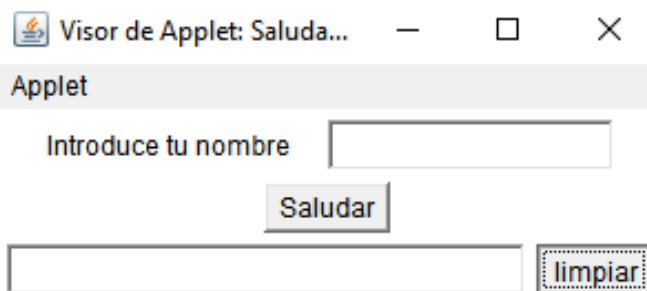
```
public void mouseExited(MouseEvent e) {  
    showStatus("Bienvenido.");  
}
```

```
} // clase
```

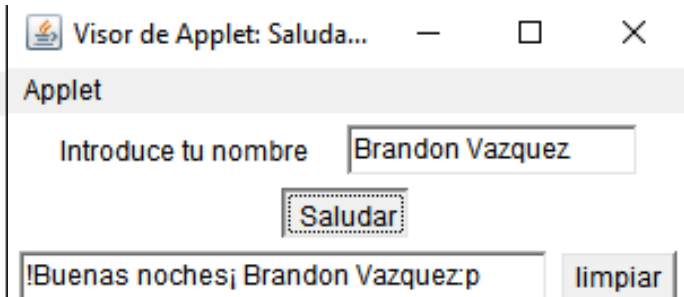
CAPTURA DE EJECUCION



Applet iniciado.



Este boton limpia



Este boton saluda

