

Eventos en JAVA - SWING

Todos los sistemas operativos están constantemente atendiendo a los eventos generados por los usuarios. Estos eventos pueden ser **pulsar una tecla del teclado, mover el ratón, hacer clic izquierdo, clic derecho, mover la rueda del ratón, etc.** (Java distingue entre simplemente pulsar el ratón en un sitio cualquiera o hacerlo, por ejemplo, en un botón).

Cuando se produce un evento el sistema operativo notifica a la aplicación involucrada y ellas deciden qué hacer con dicho evento (se ejecuta un método asociado al evento).

El modelo de Java se basa en la **delegación de eventos**: el evento se produce en un determinado componente, por ejemplo un **Jbutton**, que será la “fuente del evento” (**ActionListener**).

A continuación el evento se transmite a un “manejador de eventos” (EventListener) que este asignado al componente en el que se produjo el evento. El objeto que escucha los eventos es el que se encargará de responder a ellos adecuadamente. Esta separación de código entre generación del evento y actuación respecto a él simplifica el código y facilita la labor del programador.

LISTENERS MÁS COMUNES:

ActionListener

DESCRIPCIÓN:

Se produce al hacer click en un componente, también si se pulsa Enter teniendo el foco en el componente.

METODOS:

- public void **actionPerformed**(ActionEvent e)

EVENTS:

- **JButton**: click o pulsar Enter con el foco activado en él.
- **JList**: doble click en un elemento de la lista.
- **JMenuItem**: selecciona una opción del menú.
- **TextField**: al pulsar Enter con el foco activado.

KeyListener

DESCRIPCIÓN:

Se produce al pulsar una tecla, según el método cambiara la forma de pulsar la tecla.

METODOS:

- public void **keyTyped**(KeyEvent e)
- public void **keyPressed**(KeyEvent e)
- public void **keyReleased**(KeyEvent e)

EVENTS:

Cuando pulsamos una tecla, según el Listener:

- **keyTyped**: al pulsar y soltar la tecla.
- **keyPressed**: al pulsar la tecla.
- **keyReleased**: al soltar la tecla.

FocusListener

DESCRIPCIÓN:

Se produce cuando un componente gana o pierde el foco, es decir, que esta seleccionado.

METODOS:

- public void **focusGained**(FocusEvent e)
- public void **focusLost**(FocusEvent e)

EVENTS:

Recibir o perder el foco.

MouseListener

DESCRIPCIÓN:

Se produce cuando realizamos una acción con el ratón.

METODOS:

- public void **mouseClicked**(MouseEvent e)
- public void **mouseEntered**(MouseEvent e)
- public void **mouseExited**(MouseEvent e)
- public void **mousePressed**(MouseEvent e)
- public void **mouseReleased**(MouseEvent e)

EVENTS:

Según el Listener:

- **mouseClicked**: pinchar y soltar.
- **mouseEntered**: entrar en un componente con el puntero.
- **mouseExited**: salir de un componente con el puntero
- **mousePressed**: presionar el botón.
- **mouseReleased**: soltar el botón.

MouseMotionListener

DESCRIPCIÓN:

Se produce con el movimiento del mouse.

METODOS:

- public void **mouseDragged**(MouseEvent e)
- public void **mouseMoved**(MouseEvent e)

EVENTS:

Según el Listener:

- **mouseDragged**: clic y arrastrar un componente.
- **mouseMoved**: al mover el puntero sobre un elemento

AÑADIR EVENTOS:

Podemos añadir un evento de dos formas:

```
componente.add"tipo evento"(new "tipo evento"() {  
    // métodos del evento  
});
```

```
"Tipo evento" "nombre"= new "tipo evento"{  
    // métodos del evento  
};  
componente.add"Tipo evento"("nombre");
```

Por Ejemplo:

Forma 1:

```
boton1.addActionListener(new ActionListener(){  
    public void actionPerformed (){  
        //Acciones  
    }  
});
```

Forma 2:

```
ActionListener al=new ActionListener(){  
    public void actionPerformed (){  
        //Acciones  
    }  
};  
boton1.addActionListener(al);
```

```

package eventos;

import java.awt.EventQueue;
import javax.swing.*;
import java.awt.event.*;

public class EjemploEvento extends JFrame {

    private JPanel contentPane;
    private JTextField textField;
    private JTextField textField_1;
    private JTextField textField_2;
    private JComboBox<String> comboBox;
    private JRadioButton rdbtnOpcion;
    private JRadioButton rdbtnOpcion_1;
    private JRadioButton rdbtnOpcion_2;
    private JLabel lblEstasSobreLa;

    // Programa Principal

    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    EjemploEvento frame = new EjemploEvento();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    // CONSTRUCTOR del FRAME

    public EjemploEvento() {

        //Añade un titulo, no es estrictamente necesario
        setTitle("Titulo de la ventana");

        /* Coordenadas x y de la aplicación y su altura y longitud,
        * si no lo indicamos aparecerá una ventana muy pequeña */
        setBounds(400, 200, 655, 520);

        /*Indica que cuando se cierre la ventana se acaba la aplicación,
        * si no lo indicamos cuando cerremos la ventana la aplicación seguirá funcionando */

        setDefaultCloseOperation(EXIT_ON_CLOSE);

        //Hace visible la ventana, si no lo hacemos no veremos la aplicación
        setVisible(true);

        //Creamos el panel
        contentPane = new JPanel();

        //Indicamos su diseño
        contentPane.setLayout(null);

        //asigno el pannel a la ventana
        setContentPane(contentPane);

        //Componentes

        //Etiquetas

        JLabel lblEscribeElNombre = new JLabel("Escribe el nombre el nombre de una persona sin dígitos");
        lblEscribeElNombre.setBounds(369, 32, 229, 25);
        contentPane.add(lblEscribeElNombre);

        JLabel lblEligeUnaOpcin = new JLabel("Elige una opción:");
    }

```

```

lblEligeUnaOpcin.setBounds(176, 32, 109, 14);
contentPane.add(lblEligeUnaOpcin);

JLabel lblPulsaElBoton = new JLabel("Pulsa el botón");
lblPulsaElBoton.setBounds(47, 32, 85, 14);
contentPane.add(lblPulsaElBoton);

lblEstasSobreLa = new JLabel("Estas sobre la ");
lblEstasSobreLa.setBounds(158, 224, 192, 14);
contentPane.add(lblEstasSobreLa);

JLabel lblSoloSePuede = new JLabel("Solo se puede escribir dígitos");
lblSoloSePuede.setBounds(371, 208, 193, 14);
contentPane.add(lblSoloSePuede);

JLabel lblNombreElegido = new JLabel("Nombre Elegido");
lblNombreElegido.setBounds(175, 124, 110, 14);
contentPane.add(lblNombreElegido);

//Campo de texto

textField = new JTextField();
textField.setBounds(371, 68, 193, 26);
contentPane.add(textField);

textField_1 = new JTextField();
textField_1.setEditable(false);
textField_1.setBounds(175, 150, 141, 20);
contentPane.add(textField_1);

textField_2 = new JTextField();
textField_2.setBounds(371, 247, 126, 20);
contentPane.add(textField_2);

//Botones

JButton btnPulsame = new JButton("Púlsame");
btnPulsame.setBounds(43, 70, 89, 23);
contentPane.add(btnPulsame);

JButton btnAnadir= new JButton("A\u00F1adir");
btnAnadir.setBounds(371, 124, 89, 23);
contentPane.add(btnAnadir);

//Botones de radio

rdbtnOpcion= new JRadioButton("Opción 1");
rdbtnOpcion.setBounds(43, 194, 109, 23);
contentPane.add(rdbtnOpcion);

rdbtnOpcion_1 = new JRadioButton("Opción 2");
rdbtnOpcion_1.setBounds(43, 220, 109, 23);
contentPane.add(rdbtnOpcion_1);

rdbtnOpcion_2 = new JRadioButton("Opción 3");
rdbtnOpcion_2.setBounds(43, 246, 109, 23);
contentPane.add(rdbtnOpcion_2);

//Agrupamos los botones de radio
ButtonGroup bgroup = new ButtonGroup();
bgroup.add(rdbtnOpcion);
bgroup.add(rdbtnOpcion_1);
bgroup.add(rdbtnOpcion_2);

//Menu de opciones
comboBox = new JComboBox<>();
comboBox.setBounds(175, 70, 141, 22);
contentPane.add(comboBox);

//Añadimos opciones
comboBox.addItem("Fernando");
comboBox.addItem("Alberto");
comboBox.addItem("Arturo");

```

```

//Eventos

btnPulsame.addActionListener(new ActionListener(){
    public void actionPerformed (ActionEvent e){
        JOptionPane.showMessageDialog(contentPane, "¡Me has pulsado!");
    }
});

//En este caso hemos debido de convertir el componente en un atributo
comboBox.addActionListener(new ActionListener(){
    public void actionPerformed (ActionEvent e){
        textField_1.setText(comboBox.getItemAt(comboBox.getSelectedIndex()));
    }
});

textField.addKeyListener(new KeyListener(){
    public void keyTyped (KeyEvent e){
        //Si el caracter introducido es un dígito...
        if (Character.isDigit(e.getKeyChar())){
            //... no lo escribe
            e.consume();
        }
    }
    public void keyReleased(KeyEvent e){
    }
    public void keyPressed(KeyEvent e){
    }
});

btnAnadir.addActionListener(new ActionListener(){
    public void actionPerformed (ActionEvent e){
        comboBox.addItem(textField.getText());
        textField.setText("");
        JOptionPane.showMessageDialog(contentPane, "Nombre Añadido");
    }
});

rdbtnOpcion.addMouseListener(new MouseListener(){
    public void mouseClicked(MouseEvent e){
    }
    public void mouseEntered(MouseEvent e){
        lblEstasSobreLa.setText(lblEstasSobreLa.getText()+rdbtnOpcion.getText());
    }
    public void mouseExited(MouseEvent e){
        lblEstasSobreLa.setText("Estas sobre la ");
    }
    public void mousePressed(MouseEvent e){
    }
    public void mouseReleased(MouseEvent e){
    }
});

rdbtnOpcion_1.addMouseListener(new MouseListener(){
    public void mouseClicked(MouseEvent e){
    }
    public void mouseEntered(MouseEvent e){
        lblEstasSobreLa.setText(lblEstasSobreLa.getText()+rdbtnOpcion_1.getText());
    }
    public void mouseExited(MouseEvent e){
        lblEstasSobreLa.setText("Estas sobre la ");
    }
    public void mousePressed(MouseEvent e){
    }
    public void mouseReleased(MouseEvent e){
    }
});

```

```

    }
});

rdbtnOpcion_2.addMouseListener(new MouseListener(){
    public void mouseClicked(MouseEvent e){

    }
    public void mouseEntered(MouseEvent e){
        lblEstasSobreLa.setText(lblEstasSobreLa.getText()+rdbtnOpcion_2.getText());
    }
    public void mouseExited(MouseEvent e){
        lblEstasSobreLa.setText("Estas sobre la ");
    }
    public void mousePressed(MouseEvent e){

    }
    public void mouseReleased(MouseEvent e){

    }
});

textField_2.addKeyListener(new KeyListener(){
    public void keyTyped (KeyEvent e){
        //Si el caracter introducido no es un digito...
        if (!Character.isDigit(e.getKeyChar())){
            //... no lo escribe
            e.consume();
        }
    }
    public void keyReleased(KeyEvent e){
    }
    public void keyPressed(KeyEvent e){
    }
});
}
}

```

Titulo de la ventana

Pulsa el boton

Elige una opción:

Escribe el nombre el nombre de una p...

Pulsame

Fernando

Nombre Elegido

Añadir

Opcion 1

Opcion 2

Opcion 3

Estas sobre la

Solo se puede escribir digitos