**Investigación de event listeners**

Facultad de Habitad, Infraestructura y Creatividad, Pontificia Universidad Católica del Ecuador

Programación 3

Ing. Édison Vicente Mora Londoño

Integrantes:

Israel Hernández

5 de mayo del 2025

**Interface MouseListener**

El listener es usado para recibir eventos “interesantes” del mous (presionar, soltar, click, enter y click) en un componente. Una clase que esta interesada en procesar un evento de un mouse puede, implementar esta interfaz o usar un MouseAdapter class (sobrecargando solo los métodos de interés).

Un evento que involucre el mouse es generado cuando el mouse es presionado, soltado (presionado y soltado). Cuando un evento de mouse ocurre, el método relevante para el listener es invocado y el MouseEvent es pasado a este.

**Metodos**

**void mouseClicked(MouseEvent e)**

Invocado cuando el botón del mouse ha sido presionado y liberado sobre un componente.

**void mousePressed(MouseEvent e)**

Invocado cuando el botón del mouse ha sido presionado en un componente

**void mouseReleased(MouseEvent e)**

Invocado cuando el moton del mouse ha sido liberado en un componente

**void mouseEntered(MouseEvent e)**

Invocado cuando el mouse entra sobre un componente

**Void mouseExited(MouseEvent e)**

Invocado cuando el mouse sale de un componente

**Interface WindowListener**

Es usado para recibir eventos de ventana, es usado cuando el estatus de una ventana cambia, estar abierta, cerrada, activada, desactivada, minimizada o no.

**Métodos**

**void windowOpened(WindowEvent e)**

Invocado la primera vez que una ventana es hecha visible

**void windowClosing(WindowEvent e)**

Invocado cuando el usuario intenta cerrar la ventana por el propio menu de la ventana

**void windowClosed(WindowEvent e)**

Invocado cuando una ventana ha sido cerrada como resultado de haber sido llamada a cerrar la ventana

**void windowIconified(WindowEvent e)**

Invocado cuando una ventana cambio de una forma “normal” a un estado minimizado. Para muchas plataformas, una ventana es minimizada usando el incono especifico situado en la ventana.

**void windowDeiconified(WindowEvent e)**

Invocado cuando una ventana cambia su estado de minimizado a normal.

**void windowActivated(WindowEvent e)**

Invocado cuando una ventana esta activa, en los sistemas de ventanas solo 1 puede estar activo, el sistema de ventana puede aplicar una decoración como una barra resaltada para indicar su estatus de activo. La ventana activa es aquella que esta en el “foco” de atención del usuarios.

**void windowDeactivated(WindowEvent e)**

Invocado cuando una ventana no esta activada, la ventana desactivada es aquella que no esta en el foco de atención del usuario, no tiene ninguna decoración que la resalte.

**bibliografía**

[**https://docs.oracle.com/javase/8/docs/api/java/awt/event/MouseListener.html**](https://docs.oracle.com/javase/8/docs/api/java/awt/event/MouseListener.html)

[**https://oregoom.com/java/mouselistener/#Dibujar\_con\_el\_mouse**](https://oregoom.com/java/mouselistener/#Dibujar_con_el_mouse)

[**https://docs.oracle.com/javase/8/docs/api/java/awt/event/WindowListener.html**](https://docs.oracle.com/javase/8/docs/api/java/awt/event/WindowListener.html)