



PUNE INSTITUTE OF COMPUTER TECHNOLOGY, PUNE -43

Department of Electronics and Telecommunication Engineering

ASSESSMENT YEAR: - 2021-22

CLASS: - TE-V

Subject: - Advanced Java Programming

Expt. No: 7

LAB Ref: ETC/2021-22/

ROLL NO:32147

SUBMISSION DATE:

Title: - Demonstrate status of mouse on GUI

Problem Statement: -

Write a program in Java to create a frame using AWT. Implement mouseClicked, mouseEntered() and mouseExited() events. Frame should invisible when the mouse left it.

Objectives: -

To learn the concepts of Mouse Event Class

To learn the concepts of Mouse Listener Interface

Theory (Write Theory of the new concept demonstrated in this Assignment)

Java AWT

Java AWT (Abstract Window Toolkit) is an API to develop Graphical User Interface (GUI) or windows-based applications in Java. Java AWT components are platform-dependent i.e. components are displayed according to the view of operating system. AWT is heavy weight i.e. its components are using the resources of underlying operating system (OS).

Mouse Event Class:

This event indicates a mouse action occurred in a component. This low-level event is generated by a component object for Mouse Events and Mouse motion events.

- a mouse button is pressed
- a mouse button is released
- a mouse button is clicked (pressed and released)
- a mouse cursor enters the unobscured part of component's geometry
- a mouse cursor exits the unobscured part of component's geometry
- a mouse is moved
- the mouse is dragged

Java MouseListener Interface

The Java MouseListener is notified whenever you change the state of mouse. It is notified against MouseEvent. The MouseListener interface is found in java.awt.event package. It has five methods

Methods of MouseListener interface

The signature of 5 methods found in MouseListener interface are given below:

1. public abstract void mouseClicked(MouseEvent e);
2. public abstract void mouseEntered(MouseEvent e);
3. public abstract void mouseExited(MouseEvent e);
4. public abstract void mousePressed(MouseEvent e);
5. public abstract void mouseReleased(MouseEvent e);

There are two types of events that MouseMotionListener can generate.

There are two abstract functions that represent these five events. The abstract functions are :

`void mouseDragged(MouseEvent e)` : Invoked when a mouse button is pressed in the component and dragged.

Events are passed until the user releases the mouse button.

`void mouseMoved(MouseEvent e)` : invoked when the mouse cursor is moved from one point to another within the component, without pressing any mouse buttons.

MouseListener vs MouseMotionListener:

MouseListener: MouseListener events are invoked when the mouse is not in motion and is stable . It generates events such as `mousePressed`, `mouseReleased`, `mouseClicked`, `mouseExited` and `mouseEntered` (i.e when the mouse buttons are pressed or the mouse enters or exits the component). The object of this class must be registered with the component and they are registered using `addMouseListener()` method.

MouseMotionListener: MouseMotionListener events are invoked when the mouse is in motion . It generates events such as `mouseMoved` and `mouseDragged` (i.e when the mouse is moved from one point to another within the component or the mouse button is pressed and dragged from one point to another). The object of this class must be registered with the component and they are registered using `addMouseMotionListener()` method.

Diagram: -

Learning Outcomes: -		
	1	I have learnt the concepts of mouse events class
	2	I have learnt the concepts of mouse listener interface
	3	I have implemented the methods from MouseListener event

Continuous Assessment			
RPP (out of 5)	SPO (out of 5)	Total (Out of 10)	Sign
			Date: -

#(RPP – Regularity, Punctuality, Performance), (SPO – Submission, Presentation, Oral)

Important Questions: -
1. Explain the syntax of Mouse Interface.
2. What are the different methods of mouse interface?
3. Explain real life application of Mouse Event.
4. What is MouseAdapter?