

Juniorprogrammierer.de

Java 16: AWT II

2024/25 – Sascha Stojanovic

Agenda

- MouseListener
- MouseMotionListener
- ItemListener
- KeyListener
- WindowListener
- Exercise

MouseListener

- MouseListener is a interface that responds to the actions performed by mouse events generated by the user. Example: mouse clicks , mouse movements, etc.
- You need to implement it => implements MouseListener

```
//Creating a Button
Button button = new Button(label:"Click Me");
button.addMouseListener(this); // Add mouse listeners to button
add(button); // Add button to Frame

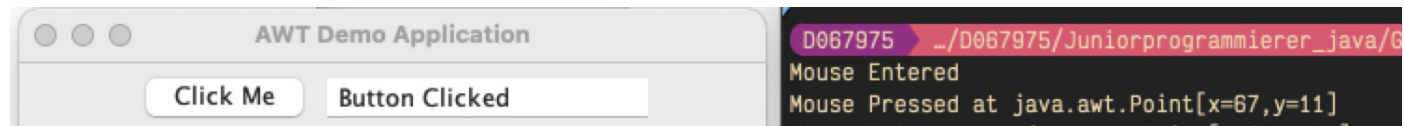
// Implement methods from MouseListener interface
public void mouseClicked(MouseEvent e) {
    System.out.println("Mouse Clicked at " + e.getPoint());
}

public void mousePressed(MouseEvent e) {
    System.out.println("Mouse Pressed at " + e.getPoint());
}

public void mouseReleased(MouseEvent e) {
    System.out.println("Mouse Released at " + e.getPoint());
}

public void mouseEntered(MouseEvent e) {
    System.out.println(x:"Mouse Entered");
}

public void mouseExited(MouseEvent e) {
    System.out.println(x:"Mouse Exited");
}
```



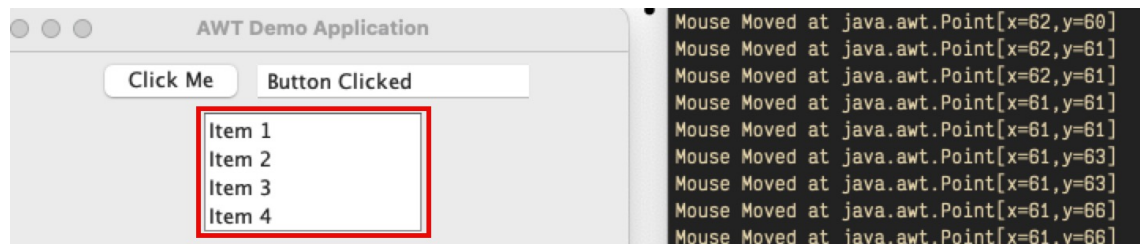
MouseMotionListener

- MouseMotionListener which is notified when mouse is moved or dragged.
- You need to implement it => implements MouseMotionListener

```
// Creating an AWT List
List list = new List(rows:4, multipleMode:false);
list.add(item:"Item 1");
list.add(item:"Item 2");
list.add(item:"Item 3");
list.add(item:"Item 4");
list.addMouseMotionListener(this); // Add mouse motion listener to list
add(list); // Add list to Frame
```

```
// Implement methods from MouseMotionListener interface
public void mouseDragged(MouseEvent e) {
    System.out.println("Mouse Dragged at " + e.getPoint());
}

public void mouseMoved(MouseEvent e) {
    System.out.println("Mouse Moved at " + e.getPoint());
}
```



KeyListener

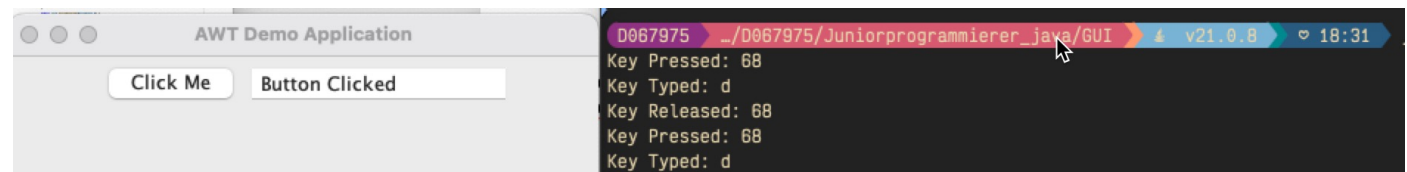
- ItemListener handles events whenever you change the state of key
- You need to implement it => implements KeyListener

```
//Creating a Button
Button button = new Button(label:"Click Me");
//button.addMouseListener(this); // Add mouse listeners to button
button.addKeyListener(this); // Add key listeners to button
add(button); // Add button to Frame
```

```
// Implement methods from KeyListener interface
public void keyTyped(KeyEvent e) {
    System.out.println("Key Typed: " + e.getKeyChar());
}

public void keyPressed(KeyEvent e) {
    System.out.println("Key Pressed: " + e.getKeyCode());
}

public void keyReleased(KeyEvent e) {
    System.out.println("Key Released: " + e.getKeyCode());
}
```



WindowListener

- As the ActionListener for Buttons it is not an Interface
- Closes and ends the corresponding application

```
// Window Event Handling for Close Operation
addWindowListener(new WindowAdapter() {
    public void windowClosing(WindowEvent we) {
        dispose(); // Close the window
        System.exit(status:0); // Terminate the application
    }
});
```

Exercise (see next slide for helping code)

- Create a class GreetingApp
 - Extend Frame
 - Implement ActionListener and KeyListener
- When you press "Greet"-Button whatever is written in the Textfield will be greeted in the label
- For the KeyListener when you type the letter "a" the label should notify you

