

Graphical User Interface II
(Action Events)

Maharakham University

อาจารย์พระ พฤชชะศรี
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาการสารสนเทศ
Email: potchara.p@msu.ac.th

1

Swing

- Swing เป็นกลุ่มของ GUI Component ที่มีการพัฒนาต่อยอดจาก AWT
- Component ของ Swing ส่วนใหญ่สืบทอดมาจาก AWT แต่มีการเพิ่มเติมคุณสมบัติให้มีความสามารถมากขึ้น
- การเลือกใช้งาน ขึ้นอยู่กับความต้องการของผู้ใช้
- Swing และ AWT สามารถทำงานร่วมกันได้

Maharakham University

2

Swing

- การใช้งาน Swing ต้อง import `javax.swing.*`;
- Component ที่สำคัญของ Swing จะคล้ายกับ AWT แต่ในชื่อ Component ของ Swing จะขึ้นต้นด้วย J เช่น
 - JFrame
 - JButton
 - JTextArea
 - อื่นๆ

Maharakham University

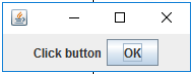
3

ตัวอย่าง Swing

```

1 import java.awt.FlowLayout;
2 import javax.swing.JButton;
3 import javax.swing.JFrame;
4 import javax.swing.JLabel;
5
6 public class DemoSwing {
7     public static void main(String[] args) {
8         MyFrame frame = new MyFrame();
9         frame.setVisible(true);
10    }
11 }
12
13 class MyFrame extends JFrame
14 {
15     JLabel jLabel = new JLabel("Click button");
16     JButton jButton = new JButton("OK");
17     public MyFrame() {
18         setSize(200, 75);
19         setLayout(new FlowLayout());
20         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
21         add(jLabel);
22         add(jButton);
23     }
24 }

```



Heart of the Northeast

Maharakrakham University

4

Windows (GUI) Programming

- รูปแบบของโปรแกรม
 - Procedural program** เป็นการทำงานแบบเป็นขั้นตอน มีลำดับการทำงานก่อนหลัง ทำงานตั้งแต่ต้นจนจบ
 - Windows (GUI) program** เป็นการทำงานที่มีการทำงานโดยอาศัยเหตุการณ์เป็นตัวกำหนดการเริ่มทำงาน เรียกอีกอย่างว่า **Event-Driven**

Heart of the Northeast

Maharakrakham University

5

Action Event และ Listener

- ในการเขียนแบบ Event-Driven ประกอบด้วยส่วนสำคัญ 3 ส่วน
 - Component ส่วนของ GUI เช่น AWT, Swing เป็นต้น
 - Action Event ส่วนของเหตุการณ์ต่างๆ
 - Listener ส่วนของตัวตรวจจับเหตุการณ์ที่เกิดขึ้นกับ Component

Heart of the Northeast

Maharakrakham University


6

Heart of the Northeast

Action Event และ Listener

- Action Event หรือ เรียกสั้นๆว่า Event
- คือ เหตุการณ์ใดๆก็ตามที่เกิดขึ้น ในขณะที่โปรแกรมกำลังทำงานอยู่ ตัวอย่างเช่น
 - การกดปุ่มบนเมาส์
 - การเลื่อนเมาส์
 - การย่อ ขยาย Frame
 - การพิมพ์ข้อความใน TextField
 - อื่นๆ อีกมากมาย

Maharakham University




7

Heart of the Northeast

Action Event และ Listener

- การเขียนโปรแกรมเพื่อทำงานเมื่อเกิดเหตุการณ์ต่างๆ จะมี ความแตกต่างกันไปตามแต่ละ Component
- ดังนั้น ในการเขียนโปรแกรมเพื่อจัดการกับเหตุการณ์ต่างๆ ที่เกิดขึ้น จึงต้องจัดการในทุกๆ Component ที่ต้องการ
- การตรวจจับ ว่ามีเหตุการณ์ใดๆ เกิดขึ้นกับ Component บ้าง จะใช้ตัวตรวจจับที่เรียกว่า **Listener**

Maharakham University




8

Heart of the Northeast

Action Event และ Listener

- Listener คือกลุ่มของ interface ที่ทำงานในการตรวจจับ เหตุการณ์ต่างๆ ที่เกิดขึ้นกับ Component
- การใช้งาน Listener ต้องมีการ implements Listener interface หรือต้องการ เรียกใช้ event adapter class เข้า กับ Component ที่ต้องการ หรืออาจเรียกว่า ต้องมีการผูกตัว ตรวจจับ (Listener) เข้ากับ Component ที่ต้องการ
- การ implements interface นั้น จะต้อง override methods ทุกตัวที่ interface นั้นกำหนด เพื่อเป็นการบอกว่า แต่ละเหตุการณ์ให้โปรแกรมทำอะไรบ้าง

Maharakham University



9

Heart of the Northeast

Action Event และ Listener

- ตัวอย่าง Action Event และ Listener ที่สำคัญ

Event Type	Listener Interface	Methods (เหตุการณ์)
ActionEvent	ActionListener	actionPerformed(...)
WindowEvent	WindowListener	windowActivated(...) windowClosed(...) windowClosing (...) windowDeactivated (...) windowDeiconified (...) windowIconified (...) windowOpened (...)

Mahasarakham University

www.msu.ac.th

10

Heart of the Northeast

Action Event และ Listener

- Event และ Listener อื่นๆ มีมากมาย สามารถศึกษาได้จาก Java Docs ในกลุ่มของ java.awt.event.*;
- ความรู้ OOP ก่อนการเขียนโปรแกรมแบบ Event-Driven
 - การสืบทอด (Inheritance) – extends, implements
 - ความรู้เรื่องของ Accessibility – private, public, ...
 - Static attributes - ค่าคงที่ต่างๆของคลาส

Mahasarakham University

www.msu.ac.th

11

Heart of the Northeast

WindowListener

- WindowListener ใช้ตรวจจับเหตุการณ์ที่เกิดขึ้นกับ window หรือ frame
- การใช้งาน WindowEvent ร่วมกับ Frame
 - สร้าง class ที่ implements listener ชื่อ WindowListener
 - Override interface methods ทั้งหมดและเขียนคำสั่งในการทำงานแต่ละเหตุการณ์
 - add listener ให้กับ Component (Frame) ที่ต้องการ

Mahasarakham University

www.msu.ac.th

12

Heart of the Northeast

WindowListener

1. Class ที่มีการ implements Listener interface

```

1 class MyListener implements WindowListener{
2
3     @Override
4     public void windowActivated(WindowEvent e) {}
5
6     @Override
7     public void windowClosed(WindowEvent e) {}
8
9     @Override
10    public void windowClosing(WindowEvent e) {
11        System.exit(0);
12    }
13
14    @Override
15    public void windowDeactivated(WindowEvent e) {}
16
17    @Override
18    public void windowDeiconified(WindowEvent e) {}
19
20    @Override
21    public void windowIconified(WindowEvent e) {}
22
23    @Override
24    public void windowOpened(WindowEvent e) {}
25
26 }
  
```

2. Override เมธอดของเหตุการณ์ที่ทุกตัว และเขียนคำสั่งในเหตุการณ์ที่ต้องการ


3. Add Listener (ลูก) เข้ากับ Component

Mahasarakham University

13

Heart of the Northeast

WindowListener



- หลังจากรันโปรแกรม frame ที่ได้ผู้ควบคุมตรวจสอบเหตุการณ์แล้วจะเริ่มตรวจสอบเหตุการณ์ต่าง ๆ ดังนี้
- ในตัวอย่างที่แล้ว โปรแกรมจะตรวจสอบเพียงเหตุการณ์เดียวคือ **windowClosing** ซึ่งจะเกิดเมื่อ frame กำลังจะถูกปิด เช่น การคลิกที่ปุ่ม x ของ frame เป็นต้น

Mahasarakham University

14

Heart of the Northeast

WindowListener

- Event อื่นๆ ที่เกิดขึ้นกับ Frame หรือ Window ต้องไปอ่านจาก Java Docs เพิ่มเติม เช่น WindowStateListener หรือ WindowFocusListener เป็นต้น
- การ implements listener ให้กับ Component นั้น สามารถ implements ได้มากกว่า 1 listener ซึ่งทำได้โดยการค้นด้วย , และจะต้อง override methods ให้ครบทุกตัว

Mahasarakham University

15

Heart of the Northeast

WindowListener

```

33 class MyListener implements WindowListener, MouseMotionListener{
34     @Override
35     public void windowActivated(WindowEvent e) {}
36     @Override
37     public void windowClosed(WindowEvent e) {}
38     @Override
39     public void windowClosing(WindowEvent e) {
40         System.exit(0);
41     }
42     @Override
43     public void windowDeactivated(WindowEvent e) {}
44     @Override
45     public void windowDeiconified(WindowEvent e) {}
46     @Override
47     public void windowIconified(WindowEvent e) {}
48     @Override
49     public void windowOpened(WindowEvent e) {}
50     @Override
51     public void mouseDragged(MouseEvent e) {}
52     @Override
53     public void mouseMoved(MouseEvent e) {}
54 }

```

Maharakrakham University

16

Heart of the Northeast

ActionListener

- ActionListener เป็น Listener อีกตัวที่ใช้อย่างง่าย เนื่องจากความง่ายต่อการใช้งาน
- เป็น listener ที่ตรวจจับการทำงานทั้งหมด ที่เกิดขึ้นบน window หรือ Frame มักนำมาประยุกต์ใช้กับ การกดปุ่ม เพื่อความสะดวกรวดเร็ว
- ActionListener มี interface method อยู่ 1 ตัวคือ
 - actionPerformed(ActionEvent e)

Maharakrakham University

17

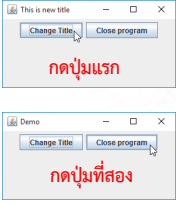
Heart of the Northeast

ActionListener

```

1 import java.awt.FlowLayout;
2 import java.awt.event.ActionEvent;
3 import java.awt.event.ActionListener;
4
5 import javax.swing.JButton;
6 import javax.swing.JFrame;
7
8 public class MyActionFrame {
9     public static void main(String[] args) {
10         MyWindow myWindow = new MyWindow();
11         myWindow.setVisible(true);
12     }
13 }

```



- เรียกใช้เมธอด getSource() ของ Object ActionEvent (e) เพื่อใช้ในการตรวจสอบว่า Component ใดเป็น Component ที่เกิดเหตุการณ์นั้นขึ้น
- Method อื่นๆที่เกี่ยวข้องสามารถเปิดได้จาก Java Docs

Maharakrakham University

18

Heart of the Northeast

ActionListener

1. Implements ActionListener

```

15 class MyWindow extends JFrame implements ActionListener{
16     JButton btnChange = new JButton("Change Title");
17     JButton btnClose = new JButton("Close Program");
18     public MyWindow() {
19         setSize(300, 100);
20         setLocationRelativeTo(null);
21         setLayout(new FlowLayout());
22         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
23         add(btnChange);
24         add(btnClose);
25
26         btnChange.addActionListener(this);
27         btnClose.addActionListener(this);
28     }
29
30     @Override
31     public void actionPerformed(ActionEvent e) {
32         if(e.getSource() == btnChange) {
33             setTitle("This is new title");
34         } else if(e.getSource() == btnClose) {
35             System.exit(0);
36         }
37     }
38 }

```

2. Override method actionPerformed

** ในตัวอย่าง เป็นการตรวจจับเหตุการณ์ และเช็ค ว่าเหตุการณ์เกิดที่ component ไດ เพื่อสั่งให้ โปรแกรมทำงานในงานที่แตกต่างกัน

19

Heart of the Northeast

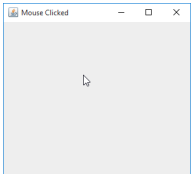
MouseListener

- MouseListener เป็น listener ที่ใช้ในการตรวจจับการทำงานของปุ่มบนเมาส์ที่มีต่อ Component ต่างๆ มี interface methods ดังนี้
 - mouseClicked(MouseEvent e)
 - mouseEntered(MouseEvent e)
 - mouseExited(MouseEvent e)
 - mousePressed(MouseEvent e)
 - mouseReleased(MouseEvent e)

20

Heart of the Northeast

MouseListener



```

1 import java.awt.event.MouseEvent;
2 import java.awt.event.MouseListener;
3 import javax.swing.JFrame;
4
5 public class DemoMouseListener {
6     public static void main(String[] args) {
7         MyDemoFrame frame = new MyDemoFrame();
8         frame.setVisible(true);
9     }
10 }
11
12 class MyDemoFrame extends JFrame implements MouseListener{
13     public MyDemoFrame() {
14         setSize(400, 200);
15         setLocationRelativeTo(null);
16         addMouseListener(this);
17     }
18
19     @Override
20     public void mouseClicked(MouseEvent e) {
21         setTitle("Mouse Click");
22     }
23
24     @Override
25     public void mouseEntered(MouseEvent e) {}
26
27     @Override
28     public void mouseExited(MouseEvent e) {}
29
30     @Override
31     public void mousePressed(MouseEvent e) {}
32
33     @Override
34     public void mouseReleased(MouseEvent e) {}
35 }

```

1. Implements MouseListener

2. Override methods

3. Add listener

21

MouseMotionListener

- MouseMotionListener เป็น listener ที่ใช้ในการตรวจจับการเคลื่อนที่ของเมาส์ที่มีต่อ Component ต่างๆ มี interface method ดังนี้
 - mouseDragged(MouseEvent e)
 - mouseMoved(MouseEvent e)

22

MouseMotionListener

```

1 import java.awt.event.MouseEvent;
2 import java.awt.event.MouseMotionListener;
3 import javax.swing.JFrame;
4
5 public class DemoMouseMotionListener {
6     public static void main(String[] args) {
7         MyMouseMotionFrame frame = new MyMouseMotionFrame();
8         frame.setVisible(true);
9     }
10 }
11 class MyMouseMotionFrame extends JFrame {
12     public MyMouseMotionFrame() {
13         setSize(400, 200);
14         setLocationRelativeTo(null);
15         addMouseMotionListener(new MouseMotionListener() {
16
17             @Override
18             public void mouseMoved(MouseEvent e) {
19                 setTitle("x=" + e.getX() + " y=" + e.getY());
20             }
21
22             @Override
23             public void mouseDragged(MouseEvent e) {}
24         });
25     }
26 }

```

* เขียนแบบ Inner class ก็ได้เหมือนกัน

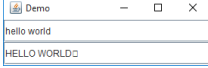
23

KeyListener

- KeyListener เป็น listener ที่ใช้ในการตรวจจับการกดปุ่มคีย์บอร์ดบน Component ต่างๆ มี interface method ดังนี้
 - keyPressed(KeyEvent e)
 - keyReleased(KeyEvent e)
 - keyTyped(KeyEvent e)

24

KeyListenerer



```

8 public class DemoKeyListener implements KeyListener {
9
10     JFrame frame = new JFrame("Demo");
11     JTextField input = new JTextField();
12     JTextField output = new JTextField();
13
14     public static void main(String[] args) {
15         DemoKeyListener demo = new DemoKeyListener();
16         demo.frame.setSize(200, 100);
17         demo.frame.setLayout(new GridLayout(2,1));
18         demo.frame.add(demo.input);
19         demo.frame.add(demo.output);
20
21         demo.input.addKeyListener(demo);
22
23         demo.frame.setVisible(true);
24     }
25
26     @Override
27     public void keyTyped(KeyEvent e) {}
28
29     @Override
30     public void keyPressed(KeyEvent e) {
31         char ch = e.getKeyChar();
32         String currentText = output.getText();
33         currentText += ch;
34         output.setText(currentText.toUpperCase());
35     }
36
37     @Override
38     public void keyReleased(KeyEvent e) {}
39 }

```

25

สรุปเกี่ยวกับ Action-event

- การเชื่อมโยงกันระหว่าง Component, Action และ Listener
- **ควรจะต้องให้ขึ้นใจ!!!**
 - Component เป็นหน้าต่างของโปรแกรมทำอะไรไม่ได้
 - Action Event เป็นเหตุการณ์ต่างๆ ที่เกิดขึ้น มันจะถูกตรวจจับด้วย Listener
 - Listener เหมือนยาม คอยตรวจดูว่าเกิดเหตุการณ์หรือ Event ไตขึ้นกับ Component บ้าง
 - เมื่อตรวจพบเหตุการณ์ จะต้องทำงานอะไร อย่างไร

Maharakham University

26

สรุปเกี่ยวกับ Action-event

- ตัวอย่าง
- ถ้าหากต้องการเขียนโปรแกรมเพื่อทำอะไรบางอย่างเมื่อมีการกดปุ่ม
 - Component คือ Button หรือ JButton
 - Action Event คือ actionPerformed หรือ mouseClicked
 - Listener คือ ActionListener หรือ MouseListener

Maharakham University

27

สรุปเกี่ยวกับ Action-event

- ตัวอย่าง
- ถ้าหากต้องการเขียนโปรแกรมเพื่อทำอะไรบางอย่างเมื่อมีการกด keyboard บน TextField
 - Component คือ TextField หรือ JTextField
 - Action Event คือ keyPressed
 - Listener คือ KeyListener



28

สรุปเกี่ยวกับ Action-event

- ตัวอย่าง
- ถ้าหากต้องการเขียนโปรแกรมเพื่อทำอะไรบางอย่างเมื่อมีการเลื่อนเมาส์ผ่าน TextArea
 - Component คือ TextArea หรือ JTextArea
 - Action Event คือ mouseMoved
 - Listener คือ MouseMotionListener



29

สรุปเกี่ยวกับ Action-event

- เมื่อรู้แล้วว่าถ้าต้องการให้โปรแกรมทำงานตามเหตุการณ์ต่างๆ มีสิ่งใดเกี่ยวข้องบ้าง แต่**ถ้าทั้ง 3 ส่วนไม่ได้นำมาผูกเข้าด้วยกัน มันจะไม่สามารถทำงานได้**
- การเชื่อมโยงกันเพื่อให้โปรแกรมทำงานได้ คือ การนำเอา component มาผูกติดกับ Listener
- ต้องการจะผูกกับ Listener ตัวใดต้อง add Listener ตัวนั้นเข้ากับ component



30

สรุปเกี่ยวกับ Action-event

- ตัวอย่างที่ 1
 - ชื่อButton.addActionListener(คลาสที่ implements ActionListener)
- ตัวอย่างที่ 2
 - ชื่อTextField.addKeyListener(คลาสที่ implements KeyListener)
- ตัวอย่างที่ 3
 - ชื่อTextArea.addMouseMotionListener(คลาสที่ implements MouseMotionListener)



31

สรุปเกี่ยวกับ Action-event

- หากต้องการเขียนโปรแกรมเพื่อทำงานอย่างใด 1 เมื่อมีการเลื่อนเมาส์ผ่านปุ่ม หรือทำงานอย่างใด 2 เมื่อกดปุ่ม
- สิ่งที่เกี่ยวข้องมีดังนี้
 - Component คือ Button หรือ JButton
 - Action Event คือ mouseMove และ mouseClicked
 - Listener คือ MouseMotionListener และ MouseListener



32

สรุปเกี่ยวกับ Action-event

- ชื่อButton.addMouseMotionListener(คลาสที่ implements MouseMotionListener)
- ชื่อButton.addMouseListener(คลาสที่ implements MouseListener)
- ต้อง addListener ทั้ง 2 ตัว เพราะมันต้องจับเหตุการณ์ที่เกิดขึ้นต่างกัน 2 เหตุการณ์



33

Heart of the Northeast

```

8 public class DemoTwoMouse implements MouseMotionListener, MouseListener {
9     JFrame frame = new JFrame("Demo");
10    public static void main(String[] args) {
11        DemoTwoMouse demo = new DemoTwoMouse();
12        demo.frame.setSize(200, 200);
13        demo.frame.addMouseListener(demo);
14        demo.frame.addMouseMotionListener(demo);
15        demo.frame.setVisible(true);
16    }
17
18    @Override
19    public void mouseClicked(MouseEvent e) {
20        frame.setTitle("Mouse clicked");
21    }
22
23    @Override
24    public void mouseMoved(MouseEvent e) {
25        frame.setTitle("Mouse moved");
26    }
27
28    @Override
29    public void mouseReleased(MouseEvent e) {}
30
31    @Override
32    public void mouseEntered(MouseEvent e) {
33        frame.setTitle("Mouse entered");
34    }
35
36    @Override
37    public void mouseExited(MouseEvent e) {
38        frame.setTitle("Mouse exited");
39    }
40 }

```

implements ได้มากกว่า 1 Listener
ถ้ามี Listener ที่ต้องการหลายตัว ต้อง add ทุกตัว

เขียนโปรแกรมตรวจสอบเมื่อเกิด Event ขึ้น

Maharakrakham University

34

Heart of the Northeast

เทคนิคและรูปแบบการเขียนโปรแกรม

- การเขียนโปรแกรมคอมพิวเตอร์ไม่มีรูปแบบตายตัว สามารถเขียนได้หลายรูปแบบ
- วิธีที่เขียนในตัวอย่างก็เป็นวิธีหนึ่งเท่านั้น สามารถเขียนวิธีอื่นให้มีผลลัพธ์เหมือนกันได้อีกหลายวิธี
- ดังนั้น ในการสอบ โอกาสที่จะเขียนคำสั่งที่มีลำดับ หรือรูปแบบเหมือนกัน เป็นไปได้้น้อยมาก
- อย่าลืม อ่านใน Java Docs และให้ใช้ Feature ของ eclipse ให้คล่องๆ จะช่วยได้มาก

Maharakrakham University

35

Heart of the Northeast

เทคนิคและรูปแบบการเขียนโปรแกรม

- นอกจากการใช้ Listener แล้ว การจัดการเหตุการณ์ของ GUI ยังสามารถใช้คลาสในกลุ่มของ Adapter ได้อีกด้วย เช่น
 - KeyListener/KeyAdapter
 - MouseListener/MouseAdapter
 - MouseMotionListener/MouseMotionAdapter
 - WindowListener/WindowAdapter

ให้ทดลองศึกษาด้วยตนเอง

Maharakrakham University

36
