



แบบฝึกปฏิบัติ ครั้งที่ 8

เรื่อง การจัดการเหตุการณ์

วัตถุประสงค์ 1. เพื่อฝึกฝนการจัดการเหตุการณ์ร่วมกับส่วนต่อประสานกราฟิกกับผู้ใช้

1. ให้นักศึกษาตอบคำถามจากโปรแกรมต่อไปนี้

```
1 import java.awt.*;  
2 import java.awt.event.*;  
3 import javax.swing.*;  
4  
5 public class App01 implements ActionListener {  
6  
7     private JFrame fr;  
8     private JPanel p1,p2;  
9     private JTextField txt1, txt2, txt3;  
10    private JButton btn1, btn2, btn3, btn4;  
11  
12    public App01() {  
13  
14        fr = new JFrame("เครื่องคิดเลข");  
15        p1 = new JPanel();  
16        p2 = new JPanel();  
17        txt1 = new JTextField();  
18        txt2 = new JTextField();  
19        txt3 = new JTextField();  
20        btn1 = new JButton("บวก");  
21        btn2 = new JButton("ลบ");  
22        btn3 = new JButton("คูณ");  
23        btn4 = new JButton("หาร");  
24  
25        // Add Listener  
26        btn1.addActionListener(this);  
27        btn2.addActionListener(this);  
28        btn3.addActionListener(this);  
29        btn4.addActionListener(this);  
30  
31        // Set Layout  
32        p1.setLayout(new GridLayout(4,1));  
33        p1.add(txt1);  
34        p1.add(txt2);  
35        p1.add(p2);  
36        p1.add(txt3);  
37  
38        p2.setLayout(new FlowLayout());  
39        p2.add(btn1);  
40        p2.add(btn2);  
41        p2.add(btn3);  
42        p2.add(btn4);  
43        fr.getContentPane().add(p1);  
44  
45    }
```

```

46      // Set JFrame Property
47      fr.setFont(new Font("TimesRoman",Font.BOLD,60));
48      fr.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
49      fr.pack();
50      fr.setVisible(true);
51
52  }
53
54  public static void main(String[] args) {
55      new App01();
56  }
57
58  @Override
59  public void actionPerformed(ActionEvent ae) {
60      if(ae.getSource().equals(btn1)){
61          System.out.println("btn1");
62      }else if(ae.getSource().equals(btn2)){
63          System.out.println("btn2");
64      }else if(ae.getSource().equals(btn3)){
65          System.out.println("btn3");
66      }else if(ae.getSource().equals(btn4)){
67          System.out.println("btn4");
68      }
69  }
70 }
```

- 1.1. จากโปรแกรมบรรทัดที่ 1 – 3 ให้นักศึกษาอธิบายว่า **Package** ต่อไปนี้จะถูกเรียกใช้เมื่อใด

Package	อธิบาย
<code>java.awt.*;</code>	เมื่อทำการ setLayout ในบรรทัดที่ 32 และ 38
<code>java.awt.event.*;</code>	เมื่อทำการ implements ActionListener และเมื่อเรียกใช้งาน method ใน ActionListener
<code>javax.swing.*;</code>	เมื่อประกาศ Attribute ในบรรทัด 5-8

- 1.2. จากโปรแกรมบรรทัดที่ 5 ให้นักศึกษาอธิบายหน้าที่ของคำสั่ง `implements ActionListener`

เพื่อที่จะสามารถเรียกใช้ **method actionPerformed**
เมื่อ **method addActionListener** ถูกเรียกใช้งาน

- 1.3. จากโปรแกรมบรรทัดที่ 7 – 10 นักศึกษาคิดว่ามีจำนวน **component** และ **container** กี่อัน

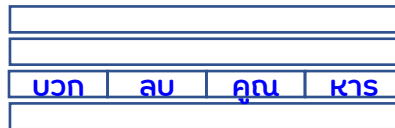
Component : 10 อัน
Container : 3 อัน

- 1.4. จากโปรแกรมบรรทัดที่ 26 – 29 ให้นักศึกษาอธิบายว่าโปรแกรมดังกล่าวมีหน้าที่อะไร

เรียกใช้ method actionPerformed โดยมี parameter เป็น object ที่เรียกใช้ addActionListener เมื่อทำการกดที่ object ตัวนั้น

- 1.5. จากโปรแกรมบรรทัดที่ 32 ให้นักศึกษาอธิบายหน้าที่ของคำสั่ง `p1.setLayout (new GridLayout (4, 1))` และ Panel ของ p1 จะถูกแบ่งออกเป็นรูปแบบใดพร้อมวาดรูป

ทำให้ object p1 แบ่ง Layout อยู่ในรูปแบบของ Grid ซึ่งจะเป็น 4 แถว 1 คอลัมน์



- 1.6. จากโปรแกรมบรรทัดที่ 38 ให้นักศึกษาอธิบายหน้าที่ของคำสั่ง `p2.setLayout (new FlowLayout ()) ;`

ทำให้ object p2 แบ่ง Layout อยู่ในรูปแบบของ Flow ก็คือจะทำการเรียง component หรือ container ที่เพิ่มเข้ามาตามลำดับ คือ จากซ้ายไปขวาและบนลงล่างเมื่อแถวเต็ม

- 1.7. จากโปรแกรมบรรทัดที่ 48 – 50 ให้นักศึกษาอธิบายหน้าที่ของคำสั่งต่อไปนี้

คำสั่ง	อธิบาย
<code>fr.setDefaultCloseOperation (JFrame.EXIT_ON_CLOSE)</code>	เมื่อทำการกดปิดโปรแกรมของ JFrame(fr) โปรแกรมจะถูกปิดลงทั้งทางกายภาพและเบื้องหลัง
<code>fr.pack ()</code>	ทำการ set หน้าจอของ JFrame(fr) ให้มีขนาดพอดีกับ component และ container
<code>fr.setVisible (true)</code>	ทำให้ JFrame(fr) เปลี่ยนการมองเห็นจาก false เป็น true ทำให้สามารถมองเห็น JFrame ได้

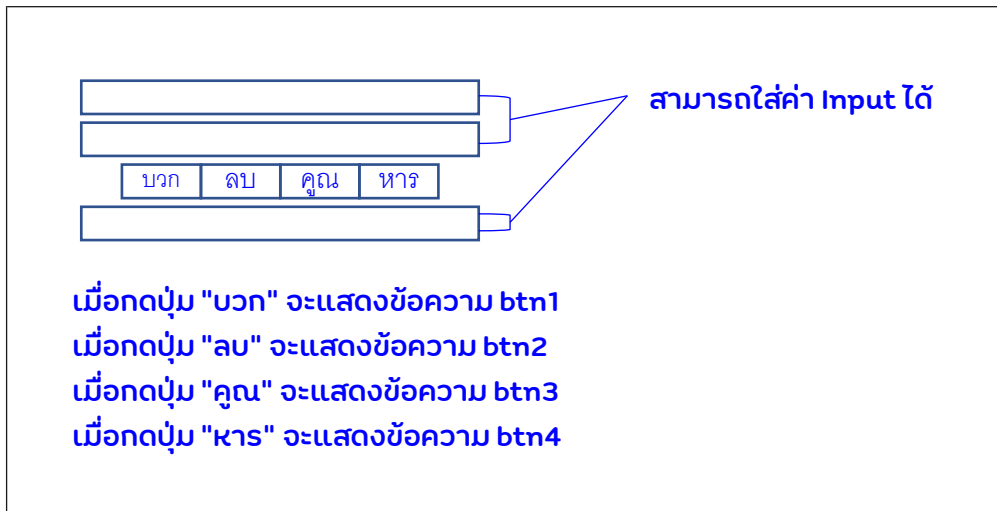
- 1.8. จากโปรแกรมบรรทัดที่ 59 ให้นักศึกษาอธิบายหน้าที่ของเมธอด `public void actionPerformed (ActionEvent ae)`

ทำงานเมื่อ method addActionListener ถูกเรียกใช้งาน หรือก็คือเมื่อปุ่มถูกกดและส่งค่าเข้าไปเป็น parameter(object ที่เรียกใช้งาน) แล้วทำงานตาม body ภายใน

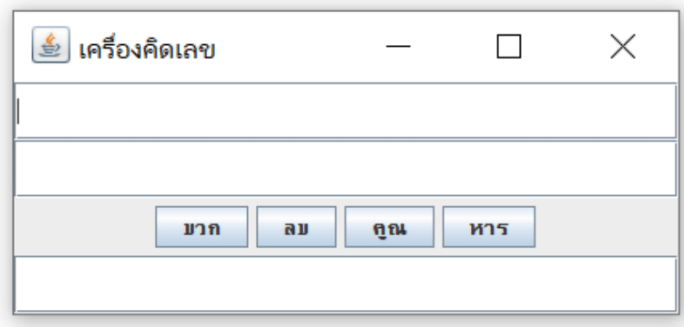
นอกจากนี้ จากโปรแกรมบรรทัดที่ 60 ให้นักศึกษาอธิบายหน้าที่ของเมธอด `ae.getSource ()`

เป็นการดึงค่าของ(ActionEvent) โดย method getSource() จะ return ค่าออกมาเป็นชื่อของ object

- 1.9. ให้นักศึกษาวาดรูปหน้าตาส่วนติดต่อผู้ใช้ที่เป็นผลลัพธ์ของโปรแกรมจากโปรแกรมข้างต้น พร้อมอธิบายหลักการ
ทำงาน



2. ให้นักศึกษาแก้ไขโปรแกรมจากข้อที่ 1 ให้กลายเป็นโปรแกรมเครื่องคิดเลขอย่างง่าย



```
@Override
public void actionPerformed(ActionEvent ae) {
    if(ae.getSource().equals(btn1)) {

    }else if(ae.getSource().equals(btn2)) {

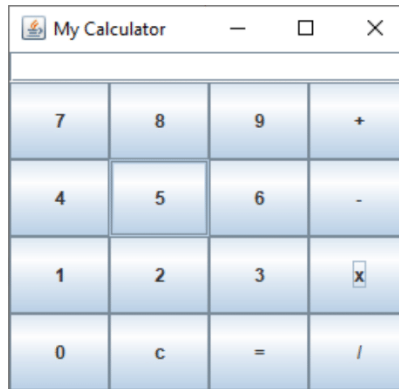
    }else if(ae.getSource().equals(btn3)) {

    }else if(ae.getSource().equals(btn4)) {

    }
}
```

3.ให้นักศึกษาร่างโปรแกรมเครื่องคิดเลข โดยอาศัยส่วนต่อประสานกราฟิกกับผู้ใช้งานของคลาส

“Calculator Sample” จาก Lab Sheet 07 มาใช้งาน



โดยมีรายละเอียดดังต่อไปนี้

- 1) ถ้าผู้ใช้กดปุ่ม “+” หรือ “-” หรือ “x” หรือ “/” ค่าที่ปรากฏใน **JTextField** จะถูกกำหนดให้เป็น ช่องว่าง
- 2) ถ้าผู้ใช้กดปุ่ม “0” ถึง “9” ค่าที่ปรากฏใน **JTextField** จะถูกเพิ่ม ต่อจากของเดิมทางด้านขวาไป เรื่อย ๆ
- 3) ถ้าผู้ใช้กดปุ่ม “c” ค่าที่ปรากฏใน **JTextField** จะถูกกำหนดให้เป็น ช่องว่าง
- 4) ถ้าผู้ใช้กดปุ่ม “=” ค่าที่ปรากฏใน **JTextField** จะเป็น ค่าผลลัพธ์

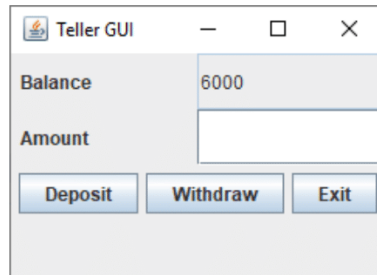
กำหนดโค้ดสำหรับทดสอบความถูกต้องของอินเทอร์เฟซที่นักศึกษาได้พัฒนาขึ้นดังนี้

โค้ด

```

public class Main {
    public static void main(String[] args) {
        new CalculatorSample();
    }
}
  
```

4. ให้นักศึกษาร่างโปรแกรมต่อไปนี้ โดยกำหนดให้นักศึกษานำส่วนต่อประสานกราฟิกกับผู้ใช้งานของคลาส ชื่อ **"TellerGUI"** จาก Lab Sheet 07 มาใช้งาน



นอกจากนี้ ให้นักศึกษานำคลาส **Account** จาก Lab Sheet 05 มาใช้งาน โดยมีรายละเอียดดังนี้

- ค่าที่แสดงในช่อง **JLabel** ของ **Balance** จะนำค่ามาจากแอตทริบิวต์ **Balance** ของอ็อบเจกต์จากคลาส **Account**
- ถ้าผู้ใช้กดปุ่ม **"Withdraw"** โปรแกรมจะหักยอดเงินตามค่าที่ปรากฏใน **JTextField** ของช่อง **Amount** ออกจากแอตทริบิวต์ **Balance** ของอ็อบเจกต์จากคลาส **Account** จากนั้น จึงอัปเดตค่าที่แสดงใน **JTextField** ของช่อง **Balance** ให้สอดคล้องกับค่าในแอตทริบิวต์ **Balance** สำหรับกรณีที่ยอดเงินในแอตทริบิวต์ **Balance** ของอ็อบเจกต์จากคลาส **Account** มีเพียงพอ ถ้าไม่ใช่โปรแกรมจะไม่ดำเนินการใด ๆ
- ถ้าผู้ใช้กดปุ่ม **"Deposit"** โปรแกรมจะเพิ่มยอดเงินตามค่าที่ปรากฏใน **JTextField** ของช่อง **Amount** ลงในแอตทริบิวต์ **Balance** ของอ็อบเจกต์จากคลาส **Account** จากนั้น จึงอัปเดตค่าที่แสดงใน **JTextField** ของช่อง **Balance** ให้สอดคล้องกับค่าในแอตทริบิวต์ **Balance**

กำหนดโค้ดสำหรับทดสอบความถูกต้องของอินเตอร์เฟซที่นักศึกษาได้พัฒนาขึ้นดังนี้

โค้ด

```

public class Main {
    public static void main(String[] args) {
        new TellerGUI();
    }
}
  
```