

บทที่ 7 : การจัดการกับเหตุการณ์

วัตถุประสงค์

- เพื่อให้เข้าใจคลาสประเภท Event ที่เกี่ยวข้องกับเหตุการณ์กราฟฟิกต่างๆ
- เพื่อให้เข้าใจอินเทอร์เฟซประเภท Listener ที่ใช้ในการรับฟังเหตุการณ์
- เพื่อให้เข้าใจวิธีการสร้างออบเจกต์ของคลาสประเภท Event Handler

แบบฝึกหัดเชิงปฏิบัติการ

แบบฝึกหัดที่ 1 การเขียนโปรแกรมจัดการกับเหตุการณ์

วัตถุประสงค์

- 1) เพื่อให้เข้าใจหลักการการเขียนโปรแกรมเพื่อจัดการกับเหตุการณ์ต่างๆที่เกิดขึ้นในโปรแกรม GUI
- 2) เพื่อให้เข้าใจคลาสประเภท Event และอินเทอร์เฟซประเภท Listener ที่กำหนดในแพ็คเกจ java.awt

คำสั่ง

ทดลองเขียนโปรแกรมจัดการเหตุการณ์เพื่อทำให้โปรแกรม GUI ในแบบฝึกหัดบทที่ 5 และ 6 สามารถทำงานได้อย่างถูกต้อง

ขั้นตอนการปฏิบัติการ

- 1) จงเขียนคลาส TellerGUI เพื่อให้สามารถจัดการกับเหตุการณ์ประเภทต่าง ๆ ได้ โดย implements อินเทอร์เฟซดังนี้

```
public class TellerGUI implements ActionListener {  
    }
```

และมีขั้นตอนในการเขียนคำสั่งต่าง ๆ ดังนี้

- (a) กำหนดให้ออบเจกต์ bn1, bn2 และ bn3 รับฟังเหตุการณ์ประเภท ActionEvent โดยให้ออบเจกต์ของคลาส TellerGUI เป็นตัวจัดการโดยใช้คำสั่ง

```
bn1.addActionListener(this);  
bn2.addActionListener(this);  
bn3.addActionListener(this);
```

- (b) เขียนเมธอดเพื่อ implements อินเทอร์เฟซ ActionListener โดยมีรูปแบบของการประกาศเมธอดดังนี้

```
public void actionPerformed(ActionEvent ev) {  
    ...  
}
```

- (c) ภายในเมธอด actionPerformed() ให้เขียนคำสั่งเพื่อเรียกค่า actionCommand มา โดยใช้คำสั่ง

```
String cmd = ev.getActionCommand();
```

- (d) เขียนคำสั่ง if..else เพื่อจัดการกับเหตุการณ์ต่างๆ ดังนี้

- กรณีที่ cmd มีค่าเป็น Exit ให้เขียนคำสั่ง

```
System.exit(0);
```

- กรณีที่ cmd มีค่าเป็น Withdraw ให้เรียกเมธอด withdraw() โดยมีค่า argument เป็นค่าที่ระบุในออบเจกต์ชื่อ tf2 โดยใช้คำสั่ง

```
double amt = Double.parseDouble(tf2.getText());  
acct.withdraw(amt);
```

และให้เปลี่ยนแปลงค่ายอดเงินในออปเจ็ค tf1 โดยใช้คำสั่ง

```
tf1.setText(acct.getBalance()+"");
```

และเขียนข้อความในออปเจ็ค err ถ้าไม่สามารถถอนเงินได้โดยใช้คำสั่ง

```
err.setText("Not Enough Money");
```

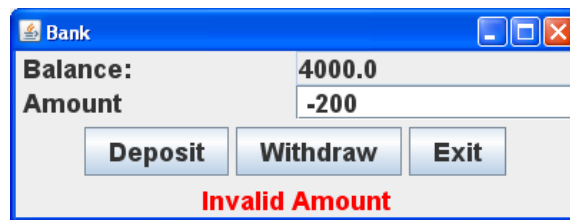
- กรณีที่ cmd มีค่าเป็น deposit ให้เรียกใช้เมธอด deposit() โดยมีค่า argument เป็นค่าที่ระบุในออปเจ็ค tf2 โดยใช้คำสั่ง

```
acct.deposit(amt);
```

และให้เปลี่ยนแปลงค่ายอดเงินในออปเจ็ค tf1 โดยใช้คำสั่ง

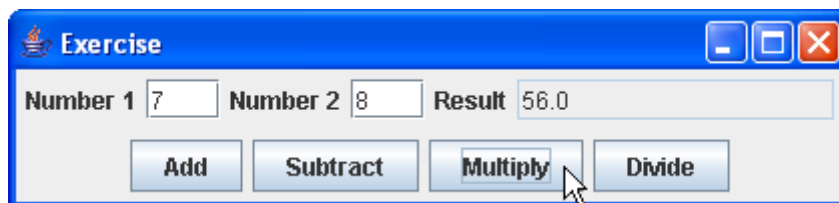
```
tf1.setText(acct.getBalance()+"");
```

2) คอมไพล์และรันโปรแกรม TellerGUI



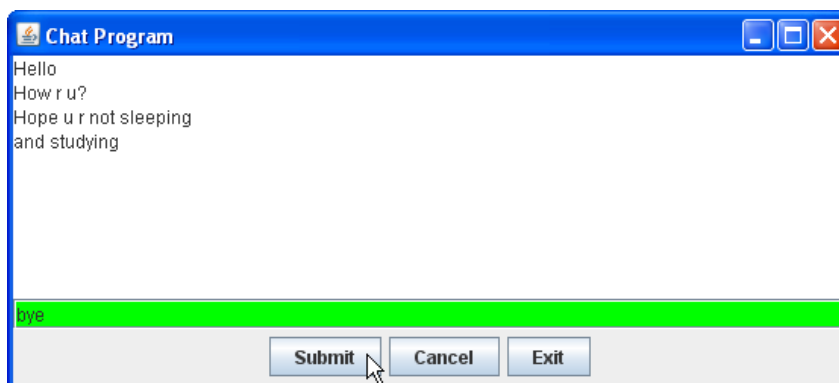
แบบฝึกหัดทบทวน

1. จงปรับปรุงโปรแกรม GUI ในข้อที่ 1 ของแบบฝึกหัดทบทวนในบทที่ 5 เพื่อให้คำนวณการบวก ลบ คูณ และหาร ตามปุ่มที่เลือกได้ ดังแสดงตัวอย่างของผลลัพธ์ในรูปที่ 7.2



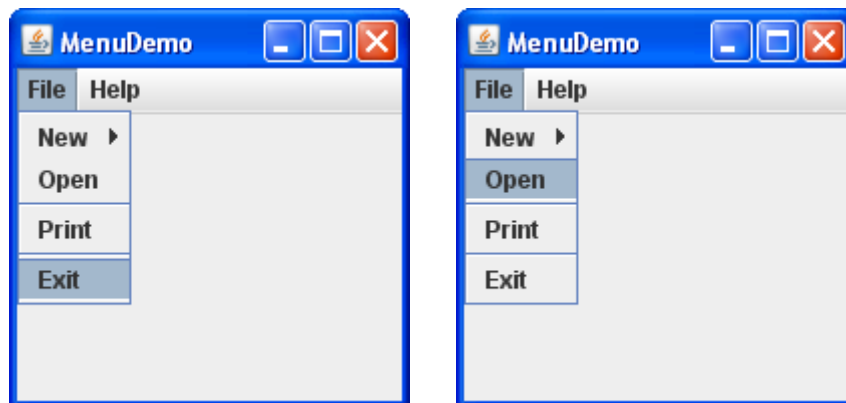
รูปที่ 7.1 โปรแกรม GUI จำลองเครื่องคิดเลขที่สามารถคิดเลขอย่างง่ายได้

2. จงปรับปรุงโปรแกรม GUI ในข้อที่ 1 ของแบบฝึกหัดปฏิบัติการในบทที่ 5 เพื่อให้สามารถสนทนากลุ่มได้ดังแสดงตัวอย่างของผลลัพธ์ในรูปที่ 7.2



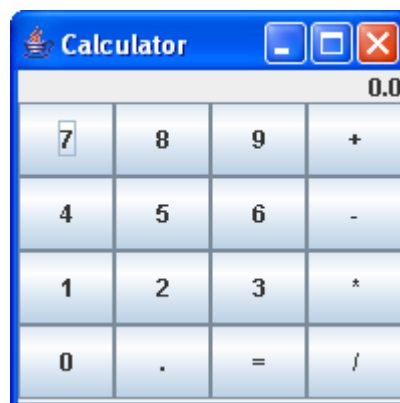
รูปที่ 7.2 โปรแกรม GUI จำลองการสนทนากลุ่ม

3. จงปรับปรุงโปรแกรม GUI ในข้อที่ 2 ของแบบฝึกหัดทบทวนในบทที่ 5 เพื่อให้สามารถเลือกรายการในเมนูได้ดังแสดงตัวอย่างของผลลัพธ์ในรูปที่ 7.3



รูปที่ 7.3 โปรแกรม GUI จำลองการเลือกเมนู

4. จงปรับปรุงโปรแกรม GUI ในข้อที่ 2 ของแบบฝึกหัดทบทวนในบทที่ 5 เพื่อให้สามารถเลือกรายการในเมนูได้ดังแสดงตัวอย่างของผลลัพธ์ในรูปที่ 7.4



รูปที่ 7.4 โปรแกรม GUI จำลองเครื่องคิดเลข