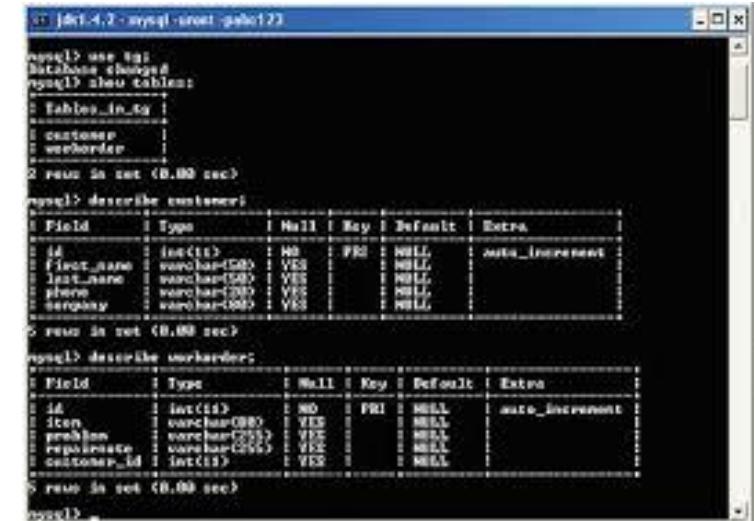
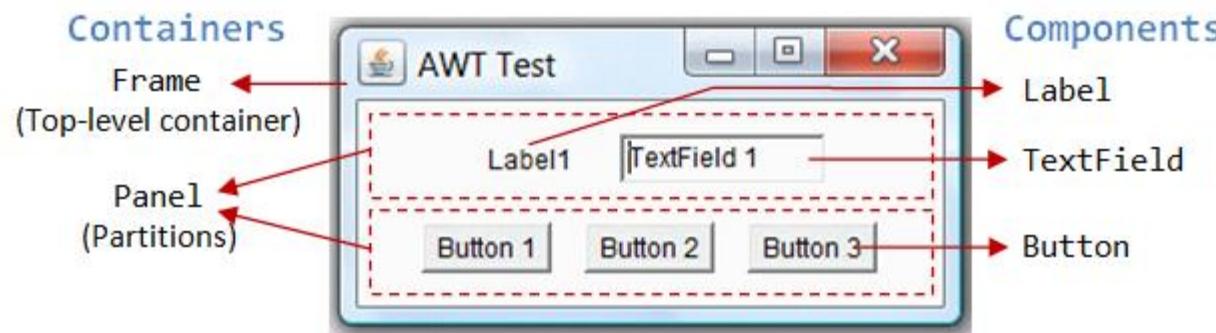


# บทที่ 10

## การเขียนโปรแกรม GUI ด้วยภาษา Java

# GUI (Graphic User Interface)

- GUI (Graphical User Interface) = ส่วนติดต่อผู้ใช้แบบกราฟิก
- ประกอบด้วย: หน้าต่าง, ปุ่ม, กล่องข้อความ, เมนูช่วยให้ผู้ใช้โต้ตอบกับโปรแกรมได้ง่าย



- ตรงข้ามกับ CLI (Command Line Interface)

# Java GUI Frameworks

Framework หลักๆ สำหรับ Java GUI

## 1. AWT (Abstract Window Toolkit)

- Framework แรกของ Java
- Components พื้นฐาน
- ใช้ Native Components ของระบบปฏิบัติการ

## 2. Swing

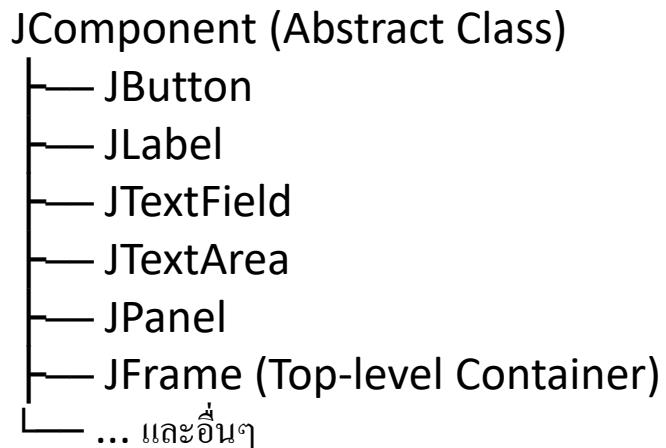
- พัฒนาต่อจาก AWT
- Pure Java Components
- รูปแบบสวยงามและยืดหยุ่นกว่า

## 3. JavaFX

- Framework ใหม่กว่า
- รองรับ Rich Media และ Animation

# Swing Architecture

- Swing Component Hierarchy



- ลักษณะสำคัญ:

- Lightweight Components - เขียนด้วย Java ทั้งหมด

- Pluggable Look and Feel - เปลี่ยนรีมได้

- Model-View-Controller Architecture

# การสร้าง JFrame พื้นฐาน

- JFrame คือหน้าต่างหลักของโปรแกรม

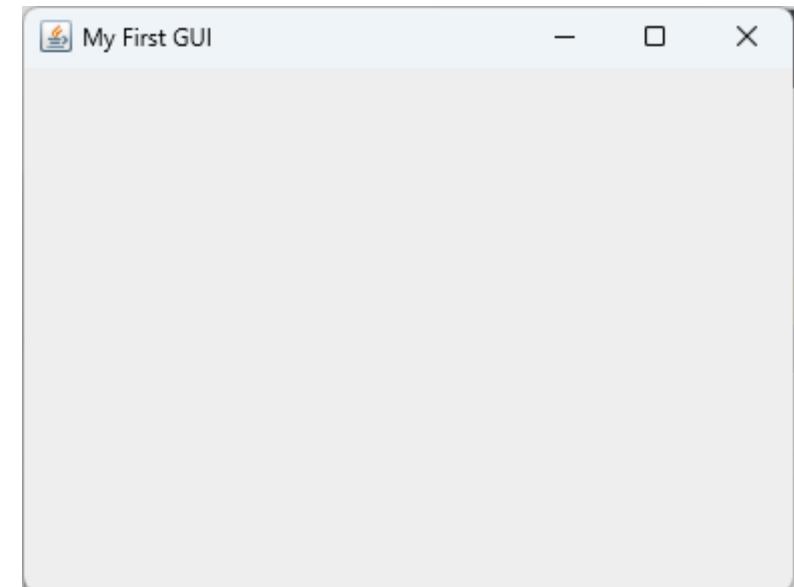
```
import javax.swing.JFrame;

public class BasicFrame {
    public static void main(String[] args) {
        // สร้าง JFrame
        JFrame frame = new JFrame("My First GUI");

        // กำหนดขนาด
        frame.setSize(400, 300);

        // กำหนดการปิดโปรแกรม
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        // แสดงหน้าต่าง
        frame.setVisible(true);
    }
}
```



# Methods พื้นฐานของ JFrame

Method	คำอธิบาย
setTitle(String)	กำหนดชื่อหน้าต่าง
setSize(int, int)	กำหนดขนาดหน้าต่าง
setLocation(int, int)	กำหนดตำแหน่งหน้าต่าง
setResizable(boolean)	กำหนดว่าปรับขนาดได้หรือไม่
setVisible(boolean)	แสดง/ซ่อนหน้าต่าง
setDefaultCloseOperation()	กำหนดพฤติกรรมเมื่อปิดหน้าต่าง

# Swing Components พื้นฐาน

- **JLabel** – แสดงข้อความหรือรูปภาพ
- **JButton** – ปุ่มสำหรับกดเพื่อทำงาน
- **JTextField** – กล่องข้อความบรรทัดเดียว
- **JTextArea** – กล่องข้อความหลายบรรทัด
- **JCheckBox** – กล่องติ๊กเลือก
- **JRadioButton** – ปุ่มเลือกแบบเลือกได้อันเดียว
- **JComboBox** – รายการแบบเลื่อนลง
- **JPanel** - พื้นที่สำหรับจัดกลุ่ม Components

# ตัวอย่าง JLabel และ JButton



```
import javax.swing.*;
import java.awt.*;

public class LabelButtonDemo extends JFrame {
    public LabelButtonDemo() {
        setTitle("Label and Button Demo");
        setSize(300, 150);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new FlowLayout());
        setLocationRelativeTo(null);

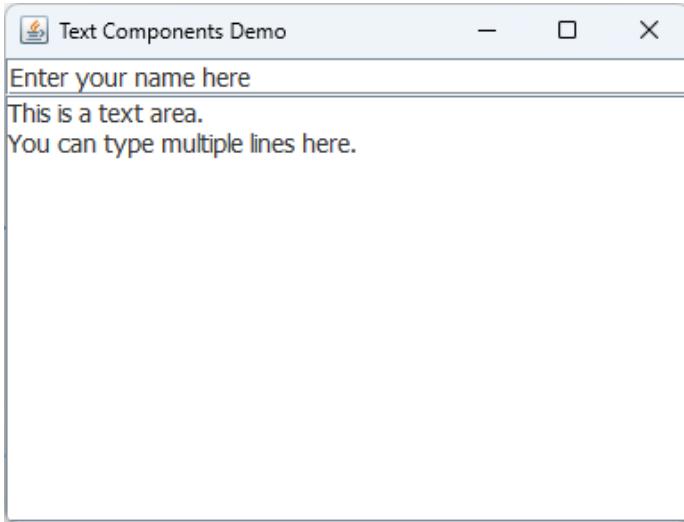
        // สร้าง JLabel
        JLabel label = new JLabel("Hello World!");
        label.setFont(new Font("Arial", Font.BOLD, 16));
        label.setForeground(Color.BLUE);

        // สร้าง JButton
        JButton button = new JButton("Click Me");
        //button.setBackground(Color.YELLOW);

        // เพิ่มลงใน Frame
        add(label);
        add(button);
    }

    public static void main(String[] args) {
        new LabelButtonDemo().setVisible(true);
    }
}
```

# ตัวอย่าง JTextField และ JTextArea



```
import javax.swing.*;
import java.awt.*;

public class TextDemo extends JFrame {
    public TextDemo() {
        setTitle("Text Components Demo");
        setSize(400, 300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new BorderLayout());
        setLocationRelativeTo(null);

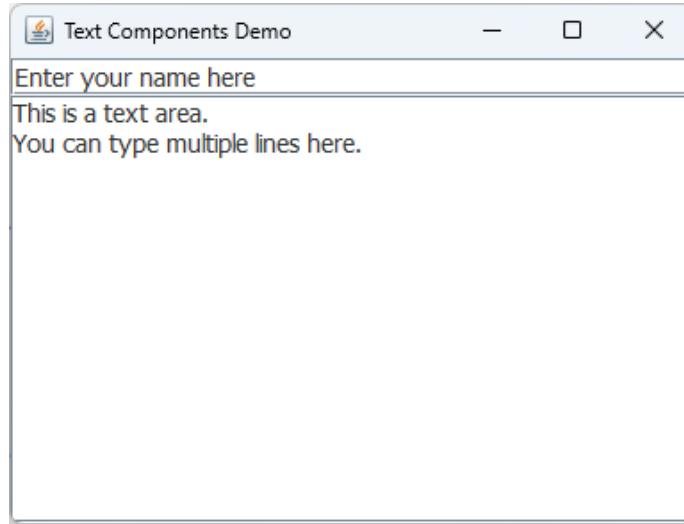
        // JTextField
        JTextField textField = new JTextField("Enter your name here");
        textField.setFont(new Font("Tahoma", Font.PLAIN, 14));

        // JTextArea with JScrollPane
        JTextArea textArea = new JTextArea(10, 30);
        textArea.setText("This is a text area.\nYou can type multiple lines here.");
        textArea.setFont(new Font("Tahoma", Font.PLAIN, 14));
        textArea.setLineWrap(true);
        textArea.setWrapStyleWord(true);

        JScrollPane scrollPane = new JScrollPane(textArea);

        add(textField, BorderLayout.NORTH);
        add(scrollPane, BorderLayout.CENTER);
    }
}
```

# ตัวอย่าง JTextField และ JTextArea (ต่อ)



```
public static void main(String[] args) {  
    new TextDemo().setVisible(true);  
}
```

# ตัวอย่าง JRadio และ JCheckBox



```
import javax.swing.*;
import java.awt.*;

public class SelectionDemo extends JFrame {
    public SelectionDemo() {
        setTitle("Selection Components");
        setSize(350, 200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new FlowLayout());
        setLocationRelativeTo(null);

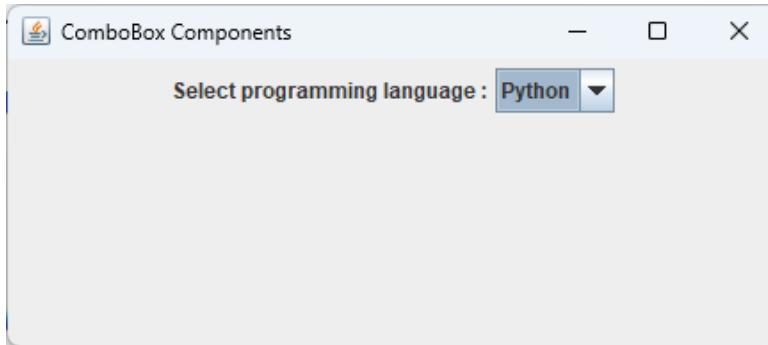
        // CheckBox
        JCheckBox check1 = new JCheckBox("Java", true);
        JCheckBox check2 = new JCheckBox("Python");
        JCheckBox check3 = new JCheckBox("C++");

        // RadioButton
        JRadioButton radio1 = new JRadioButton("Male", true);
        JRadioButton radio2 = new JRadioButton("Female");
```

## ตัวอย่าง JRadio และ JCheckBox (ต่อ)

```
// Group RadioButtons  
ButtonGroup group = new ButtonGroup();  
group.add(radio1);  
group.add(radio2);  
  
add(new JLabel("Select Languages:"));  
add(check1); add(check2); add(check3);  
add(new JLabel("Gender:"));  
add(radio1); add(radio2);  
}  
  
public static void main(String[] args) {  
    new SelectionDemo().setVisible(true);  
}  
}
```

# ตัวอย่าง JComboBox



```
import javax.swing.*;
import java.awt.*;

public class ComboBoxDemo extends JFrame {
    public ComboBoxDemo() {
        setTitle("ComboBox Components");
        setSize(450, 200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new FlowLayout());
        setLocationRelativeTo(null);

        // create component
        JLabel lblLanguage = new JLabel("Select programming language :");

        String[] languages = {"Python", "Java", "C#", "C", "PHP"};
        JComboBox<String> cboLanguage = new JComboBox<>(languages);

        add(lblLanguage);
        add(cboLanguage);
    }

    public static void main(String[] args) {
        new ComboBoxDemo().setVisible(true);
    }
}
```

# Layout Managers

Layout Manager คือระบบจัดวางตำแหน่งของ Components

ประเภทหลักๆ ที่ใช้งาน ได้แก่

- FlowLayout - เรียงแนวอน ซ้ายไปขวา
- BorderLayout - แบ่งเป็น 5 พื้นที่ (North, South, East, West, Center)
- GridLayout – เรียงแบบตาราง
- GridBagConstraints - ยึดหยุ่นที่สุด แต่ซับซ้อน
- CardLayout - แสดงทีละ Panel
- BoxLayout - เรียงแนวตั้งหรือนอน

# ตัวอย่าง FlowLayout



ผลลัพธ์: Components จะเรียงเป็นแนวจากซ้ายไปขวา และขึ้นลงใหม่เมื่อไม่พอที่

```
import javax.swing.*;
import java.awt.*;

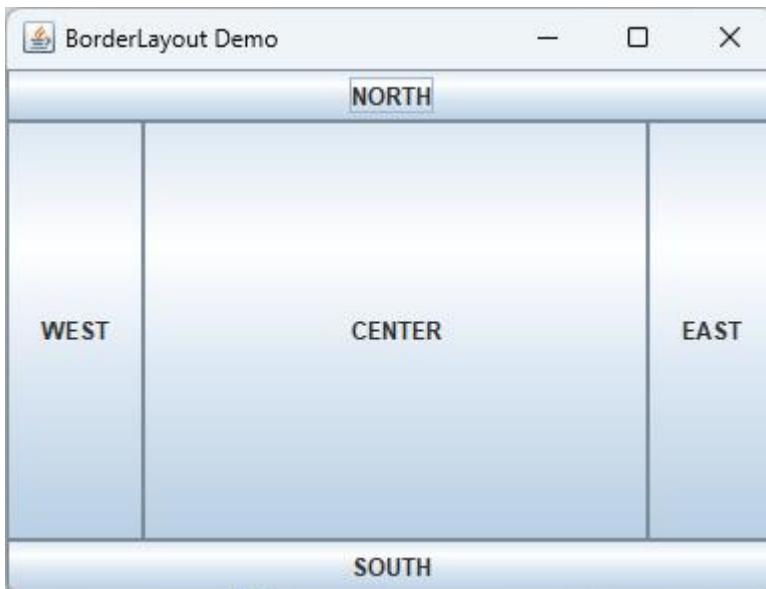
public class FlowLayoutDemo extends JFrame {
    public FlowLayoutDemo() {
        setTitle("FlowLayout Demo");
        setSize(400, 150);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        // กำหนด FlowLayout
        setLayout(new FlowLayout(FlowLayout.CENTER, 10, 10));

        // เพิ่ม Components
        add(new JButton("Button 1"));
        add(new JButton("Button 2"));
        add(new JButton("Button 3"));
        add(new JButton("Button 4"));
        add(new JButton("Button 5"));
        add(new JTextField("Text Field", 15));
    }

    public static void main(String[] args) {
        new FlowLayoutDemo().setVisible(true);
    }
}
```

# ตัวอย่าง BorderLayout



```
import javax.swing.*;
import java.awt.*;

public class BorderLayoutDemo extends JFrame {
    public BorderLayoutDemo() {
        setTitle("BorderLayout Demo");
        setSize(400, 300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        // BorderLayout เป็น default ของ JFrame
        setLayout(new BorderLayout());

        // เพิ่ม Components ໃນ 5 พื้นที่
        add(new JButton("NORTH"), BorderLayout.NORTH);
        add(new JButton("SOUTH"), BorderLayout.SOUTH);
        add(new JButton("EAST"), BorderLayout.EAST);
        add(new JButton("WEST"), BorderLayout.WEST);
        add(new JButton("CENTER"), BorderLayout.CENTER);
    }

    public static void main(String[] args) {
        new BorderLayoutDemo().setVisible(true);
    }
}
```

# ตัวอย่าง GridLayout



```
import javax.swing.*;
import java.awt.*;

public class GridLayoutDemo extends JFrame {
    public GridLayoutDemo() {
        setTitle("GridLayout Demo - Calculator");
        setSize(300, 400);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        // GridLayout 4 แถว 4  colummn
        setLayout(new GridLayout(4, 4, 5, 5));

        // สร้างบุ่มเครื่องคิดเลข
        String[] buttons = {
            "7", "8", "9", "/",
            "4", "5", "6", "*",
            "1", "2", "3", "-",
            "0", ".", "=", "+"
        };
    }
}
```

## ตัวอย่าง GridLayout (ต่อ)

```
        for (String text : buttons) {
            JButton button = new JButton(text);
            button.setFont(new Font("Arial", Font.BOLD, 20));
            add(button);
        }
    }

    public static void main(String[] args) {
        new GridLayoutDemo().setVisible(true);
    }
}
```

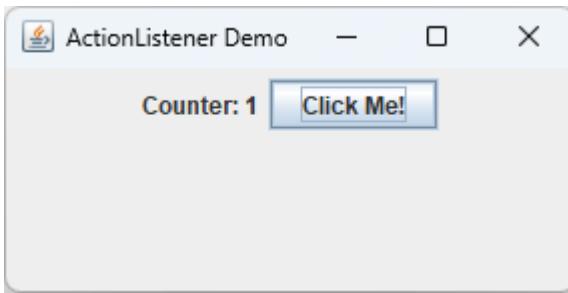
# Event Handling

- การจัดการเหตุการณ์ (Events)
  - เมื่อผู้ใช้โต้ตอบกับ GUI (คลิก, พิมพ์, เลื่อน)
  - โปรแกรมต้องตอบสนองต่อเหตุการณ์เหล่านี้
- Event Handling Model ของ Java
  1. Event Source - วัตถุที่เกิดเหตุการณ์ (เช่น JButton)
  2. Event Object - ข้อมูลเกี่ยวกับเหตุการณ์
  3. Event Listener - วัตถุที่รับฟังและจัดการเหตุการณ์

# Event Handling

- Event Listeners ที่สำคัญ
  - ActionListener - สำหรับปุ่ม, เมนู
  - MouseListener – สำหรับเหตุการณ์เมาส์
  - KeyListener – สำหรับเหตุการณ์คีย์บอร์ด
  - WindowListener - สำหรับเหตุการณ์หน้าต่าง

# ตัวอย่าง ActionListener



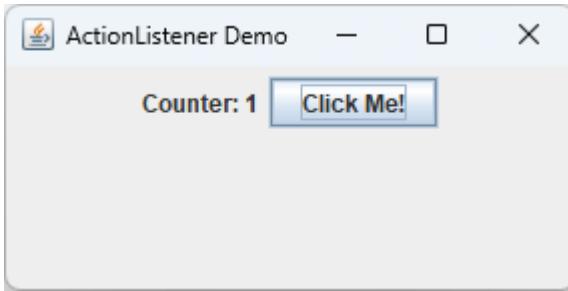
```
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class ActionListenerDemo extends JFrame {
    private JLabel label;
    private int counter = 0;

    public ActionListenerDemo() {
        setTitle("ActionListener Demo");
        setSize(300, 150);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new FlowLayout());

        label = new JLabel("Counter: 0");
        JButton button = new JButton("Click Me!");
    }
}
```

## ตัวอย่าง ActionListener (ต่อ)

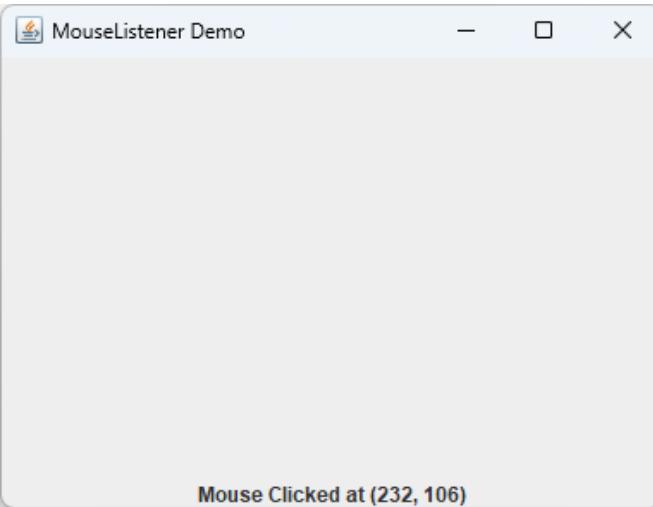


```
// เพิ่ม ActionListener
button.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        counter++;
        label.setText("Counter: " + counter);
    }
});

add(label);
add(button);
}

public static void main(String[] args) {
    new ActionListenerDemo().setVisible(true);
}
}
```

# ตัวอย่าง MouseListener



```
import javax.swing.*;
import java.awt.*;
import java.awt.event.MouseEvent;
import java.awt.event.MouseListener;

public class MouseListenerDemo extends JFrame implements MouseListener {
    private JLabel statusBar;

    public MouseListenerDemo() {
        setTitle("MouseListener Demo");
        setSize(400, 300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new BorderLayout());

        statusBar = new JLabel("Mouse Status: Ready", JLabel.CENTER);
        add(statusBar, BorderLayout.SOUTH);

        // เพิ่ม MouseListener ให้กับ Frame
        addMouseListener(this);
    }

}
```

## ตัวอย่าง MouseListener (ต่อ)

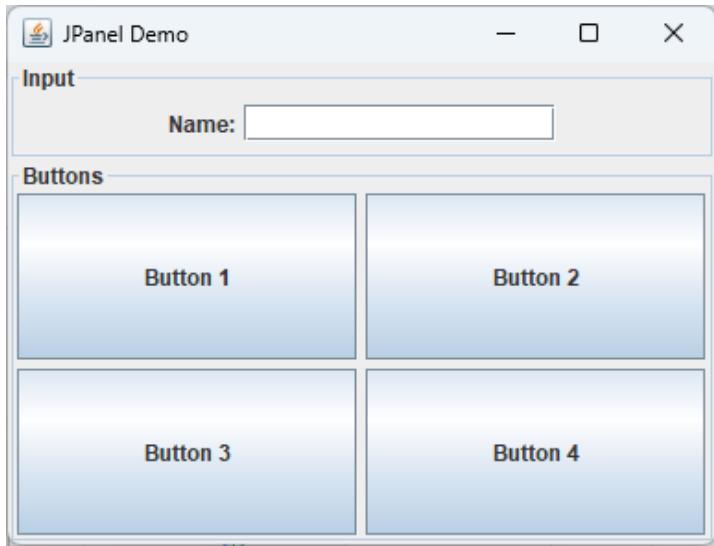
```
@Override  
public void mouseClicked(MouseEvent e) {  
    statusLabel.setText("Mouse Clicked at (" + e.getX() + ", " + e.getY() + ")");  
}  
  
@Override  
public void mousePressed(MouseEvent e) {  
    statusLabel.setText("Mouse Pressed");  
}  
  
@Override  
public void mouseReleased(MouseEvent e) {  
    statusLabel.setText("Mouse Released");  
}  
  
@Override  
public void mouseEntered(MouseEvent e) {  
    statusLabel.setText("Mouse Entered Window");  
}
```

## ตัวอย่าง MouseListener (ต่อ)

```
@Override
public void mouseExited(MouseEvent e) {
    statusLabel.setText("Mouse Exited Window");
}

public static void main(String[] args) {
    new MouseListenerDemo().setVisible(true);
}
}
```

# JPanel และการจัดกลุ่ม



```
import javax.swing.*;
import java.awt.*;

public class PanelDemo extends JFrame {
    public PanelDemo() {
        setTitle("JPanel Demo");
        setSize(400, 300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new BorderLayout());

        // สร้าง JPanel สำหรับปุ่ม
        JPanel buttonPanel = new JPanel();
        buttonPanel.setLayout(new GridLayout(2, 2, 5, 5));
        buttonPanel.setBorder(BorderFactory.createTitledBorder("Buttons"));

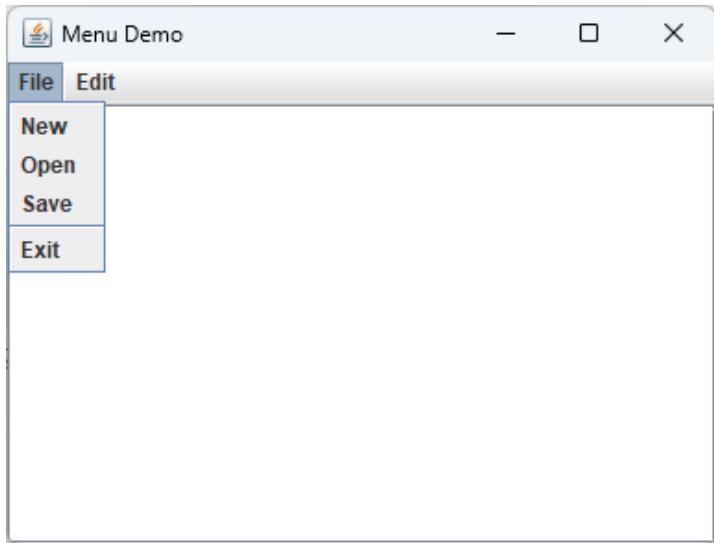
        buttonPanel.add(new JButton("Button 1"));
        buttonPanel.add(new JButton("Button 2"));
        buttonPanel.add(new JButton("Button 3"));
        buttonPanel.add(new JButton("Button 4"));
    }
}
```

```
// สร้าง Panel สำหรับ Input
JPanel inputPanel = new JPanel(new FlowLayout());
inputPanel.setBorder(BorderFactory.createTitledBorder("Input"));
inputPanel.add(new JLabel("Name:"));
inputPanel.add(new JTextField(15));

// จัด Layout
add(inputPanel, BorderLayout.NORTH);
add(buttonPanel, BorderLayout.CENTER);
}

public static void main(String[] args) {
    new PanelDemo().setVisible(true);
}
}
```

# JMenuBar และ JMenu



```
import javax.swing.*;
import java.awt.*;

public class MenuDemo extends JFrame {
    public MenuDemo() {
        setTitle("Menu Demo");
        setSize(400, 300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        createMenuBar();

        // เพิ่มนีอหางใน Frame
        JTextArea textArea = new JTextArea();
        add(new JScrollPane(textArea));
    }
}
```

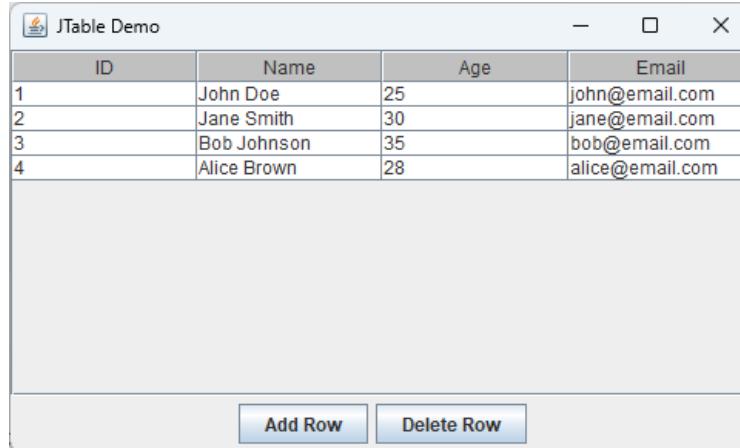
```
private void createMenuBar() {  
    JMenuBar menuBar = new JMenuBar();  
  
    // File Menu  
    JMenu fileMenu = new JMenu("File");  
    fileMenu.add(new JMenuItem("New")).addActionListener( ActionEvent e ->  
        JOptionPane.showMessageDialog(this, "New File"));  
    fileMenu.add(new JMenuItem("Open")).addActionListener( ActionEvent e ->  
        JOptionPane.showMessageDialog(this, "Open File"));  
    fileMenu.add(new JMenuItem("Save")).addActionListener( ActionEvent e ->  
        JOptionPane.showMessageDialog(this, "Save File"));  
    fileMenu.addSeparator();  
    fileMenu.add(new JMenuItem("Exit")).addActionListener( ActionEvent e ->  
        System.exit(0));  
  
    // Edit Menu  
    JMenu editMenu = new JMenu("Edit");  
    editMenu.add(new JMenuItem("Cut"));  
    editMenu.add(new JMenuItem("Copy"));  
    editMenu.add(new JMenuItem("Paste"));  
}
```

```
menuBar.add(fileMenu);
menuBar.add(editMenu);

setJMenuBar(menuBar);
}

public static void main(String[] args) {
    new MenuDemo().setVisible(true);
}
}
```

# JTable



```
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.awt.*;

public class TableDemo extends JFrame {
    public TableDemo() {
        setTitle("JTable Demo");
        setSize(500, 300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new BorderLayout());

        // สร้าง Table Model
        String[] columns = {"ID", "Name", "Age", "Email"};
        Object[][][] data = {
            {1, "John Doe", 25, "john@email.com"},
            {2, "Jane Smith", 30, "jane@email.com"},
            {3, "Bob Johnson", 35, "bob@email.com"},
            {4, "Alice Brown", 28, "alice@email.com"}
        };
        DefaultTableModel model = new DefaultTableModel(data, columns);
        JTable table = new JTable(model);
```

```
// จัดแต่งตาราง
table.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);
table.getTableHeader().setBackground(Color.LIGHT_GRAY);

// เพิ่มลงใน ScrollPane
JScrollPane scrollPane = new JScrollPane(table);
add(scrollPane, BorderLayout.CENTER);

// เพิ่มปุ่มสำหรับจัดการข้อมูล
JPanel buttonPanel = new JPanel();
JButton addButton = new JButton("Add Row");
JButton deleteButton = new JButton("Delete Row");

addButton.addActionListener( ActionEvent e -> {
    model.addRow(new Object[]{model.getRowCount() + 1, "New User", 0, ""});
});

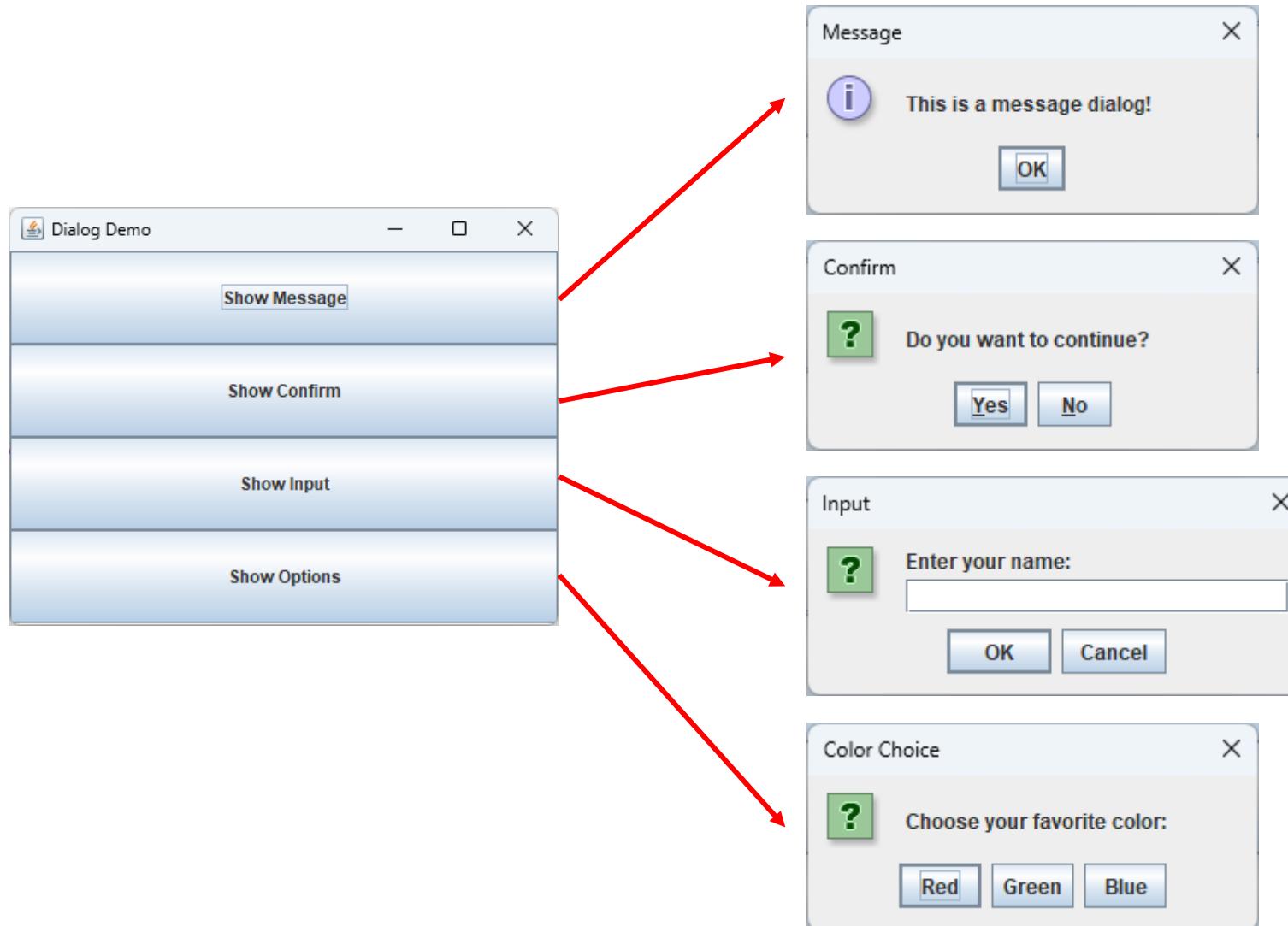

```

```
deleteButton.addActionListener( ActionEvent e -> {
    int selectedRow = table.getSelectedRow();
    if (selectedRow >= 0) {
        model.removeRow(selectedRow);
    }
});

buttonPanel.add(addButton);
buttonPanel.add(deleteButton);
add(buttonPanel, BorderLayout.SOUTH);
}

public static void main(String[] args) {
    new TableDemo().setVisible(true);
}
}
```

# Dialog Boxes



```
import javax.swing.*;
import java.awt.*;

public class DialogDemo extends JFrame {
    public DialogDemo() {
        setTitle("Dialog Demo");
        setSize(400, 300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new GridLayout(4, 1));

        // Message Dialog
        JButton msgButton = new JButton("Show Message");
        msgButton.addActionListener( ActionEvent e ->
            JOptionPane.showMessageDialog(this, "This is a message dialog!"));
    }
}
```

```
// Confirm Dialog
JButton confirmButton = new JButton("Show Confirm");
confirmButton.addActionListener( ActionEvent e -> {
    int result = JOptionPane.showConfirmDialog(this,
        "Do you want to continue?", "Confirm", JOptionPane.YES_NO_OPTION);
    if (result == JOptionPane.YES_OPTION) {
        JOptionPane.showMessageDialog(this, "You clicked Yes!");
    }
});

// Input Dialog
JButton inputButton = new JButton("Show Input");
inputButton.addActionListener( ActionEvent e -> {
    String name = JOptionPane.showInputDialog(this, "Enter your name:");
    if (name != null && !name.trim().isEmpty()) {
        JOptionPane.showMessageDialog(this, "Hello, " + name + "!");
    }
});
```

```
// Option Dialog
JButton optionButton = new JButton("Show Options");
optionButton.addActionListener( ActionEvent e -> {
    String[] options = {"Red", "Green", "Blue"};
    int choice = JOptionPane.showOptionDialog(this,
        "Choose your favorite color:", "Color Choice",
        JOptionPane.YES_NO_CANCEL_OPTION, JOptionPane.QUESTION_MESSAGE,
        null, options, options[0]);
    if (choice >= 0) {
        JOptionPane.showMessageDialog(this, "You chose: " + options[choice]);
    }
});
add(msgButton);
add(confirmButton);
add(inputButton);
add(optionButton);
}

public static void main(String[] args) {
    new DialogDemo().setVisible(true);
}
}
```

**Ex.** ให้ออกแบบฟอร์มเพื่อรับ/แสดงข้อมูลดังนี้

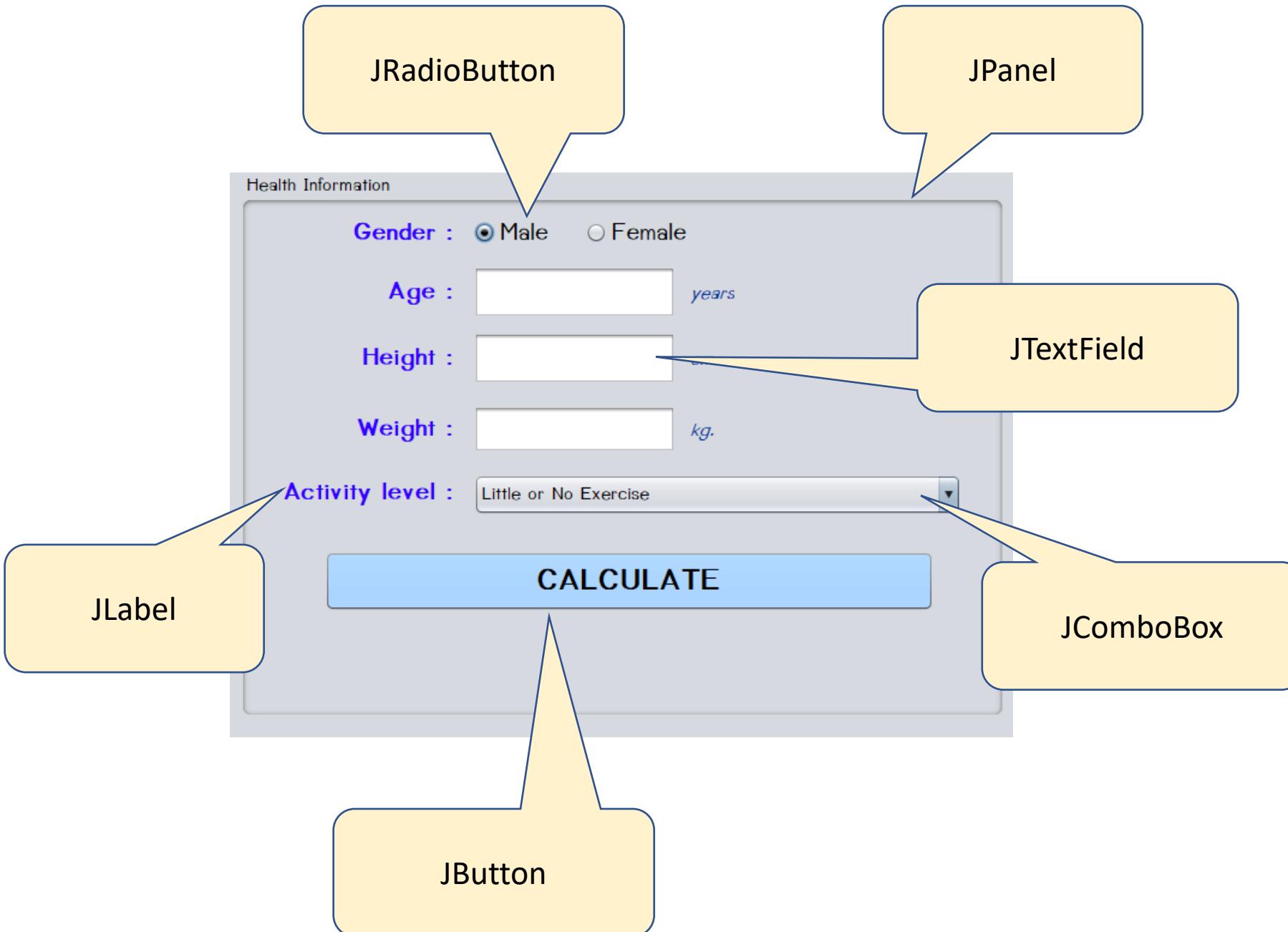
The screenshot shows a Windows application window titled "Health Tracker". The left side is labeled "Health Information" and contains the following fields:

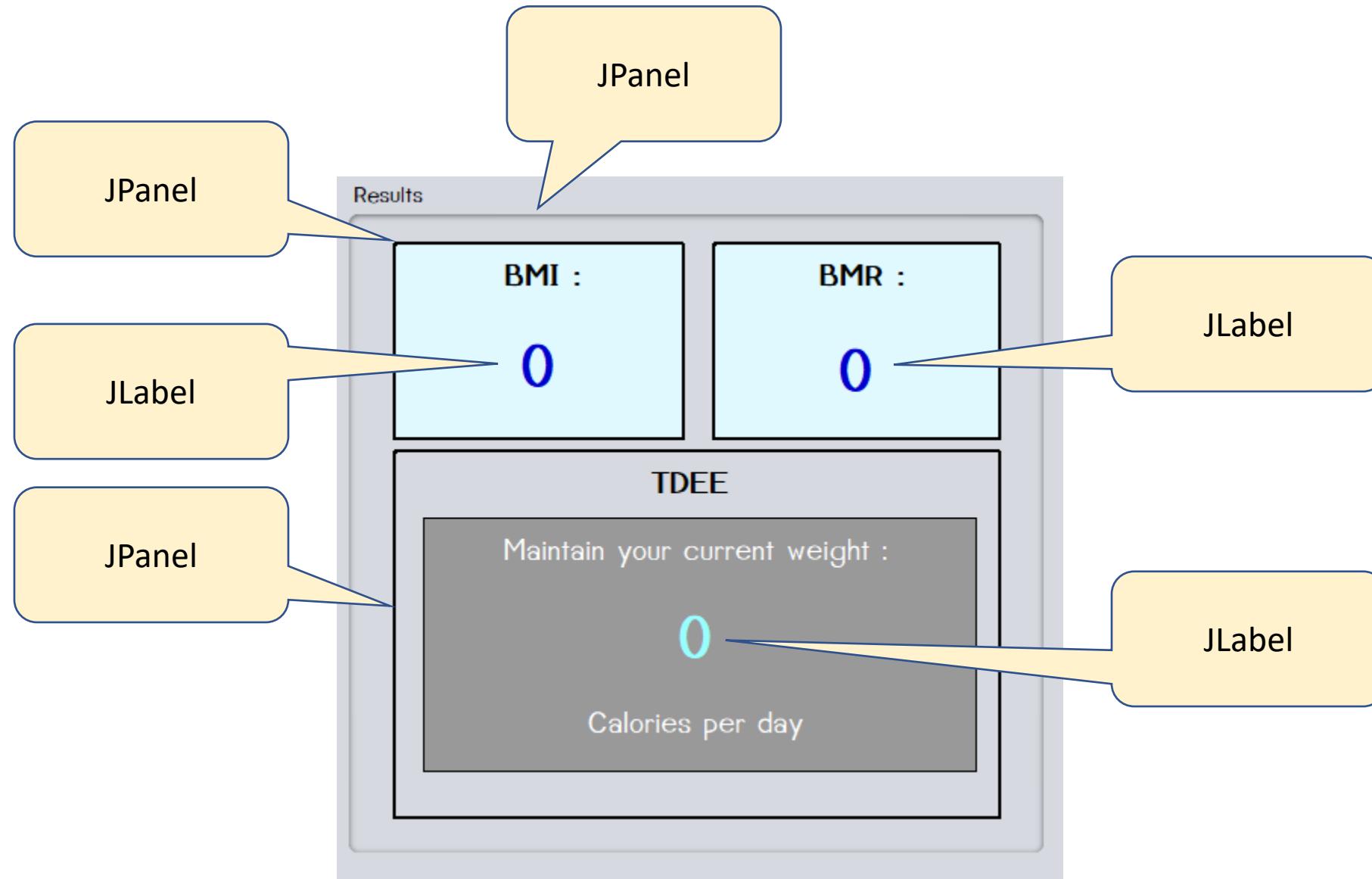
- Gender :**  Male  Female
- Age :** [Input field] *years*
- Height :** [Input field] *cm.*
- Weight :** [Input field] *kg.*
- Activity level :** Little or No Exercise

Below these fields is a large blue button labeled "CALCULATE".

The right side is labeled "Results" and displays the following information:

- BMI :** 0
- BMR :** 0
- TDEE**  
Maintain your current weight :  
0  
Calories per day





# คำนัดชื่อ

Male

rdbMale

Female

rdbFemale

Age :

txtAge

Height :

txtHeight

Weight :

txtWeight

Activity level : Little or No Exercise

cboActivity

**CALCULATE**

btnCalculate

# กำหนดชื่อ

BMI :  
0

IbBMI

BMR :  
0

IbBMR

IbTDEE

TDEE  
Maintain your current weight :  
0  
Calories per day

# กำหนดข้อมูลตัวเลือก



Model

Little or No Exercise

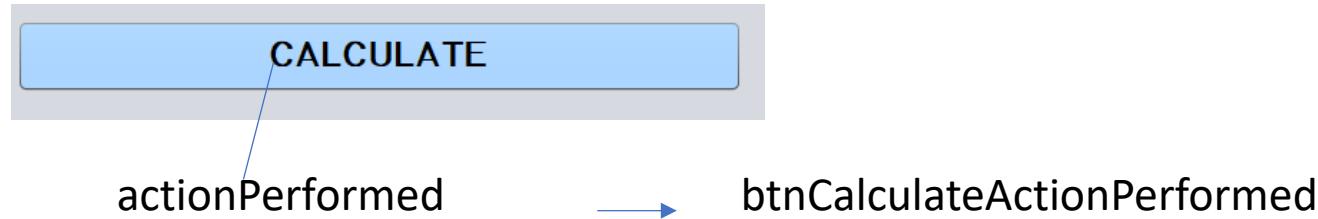
Light exercise 1-3 Days/Week

Moderate exercise 3-5 Days/Week

Hard exercise 6-7 Days/Week

Very Hard Exercise 6-7 Days/Week & Highly Physical Job

# Set Event



# สร้างคลาส Health

Health
-gender:char
-age:int
-height:double
-weight:double
-act_level:int
+Health()
+Health(char,int,double,double,int)
+calBMI(): double
+calBMR():int
+calTDEE(int) : int

$$bmi = \frac{\text{น้ำหนัก(kg.)}}{\text{ส่วนสูง(m.)}^2}$$

$$bmr(\text{ชาย}) = 66 + (13.7 \times \text{น้ำหนัก กก.}) + (5 \times \text{ส่วนสูง ซม.}) - (6.8 \times \text{อายุ ปี})$$

$$bmr(\text{หญิง}) = 665 + (9.6 \times \text{น้ำหนัก กก.}) + (1.8 \times \text{ส่วนสูง ซม.}) - (4.7 \times \text{อายุ ปี})$$

tdee

- Little or No Exercise (= bmr \* 1.2)
- Light exercise 1-3 Days/Week (= bmr \* 1.375)
- Moderate exercise 3-5 Days/Week (=bmr \* 1.55)
- Hard exercise 6-7 Days/Week (=bmr \* 1.725)
- Very Hard Exercise 6-7 Days/Week & Highly Physical Job (=bmr \* 1.9)

# จบการนำเสนอ

# END.

# Q & A