้ตัวอย่าง 1 การใช้ JFrame สร้างวินโดว์อย่างง่าย

```
1: import javax.swing.*;
2:
3: class JFrame1 {
4:   public static void main(String [ ] args) {
5:     JFrame f = new JFrame("My First Java Frame");
6:     f.setSize(250, 150); // กำหนดให้มองเห็นได้
7:     f.setVisible(true); // กำหนดให้มองเห็นได้
8:  }
9: }
```

ตัวอย่าง ๑ เป็นการสร้างโปรแกรมที่ทำงานแบบวินโคว์อย่างง่าย โดยการนำคลาส JFrame มาสร้างเป็นอ็อบเจกต์ เมื่อกอมไพล์แล้วสั่งรันด้วยกำสั่ง java JFrame1 แล้วจะได้วินโคว์หน้าตาดังต่อไปนี้



สามารถเปลี่ยนขนาด ย่อเล็กสุด (Minimize) ขยายเต็มจอ (Maximize) และปิด (close) ได้

ทดลองปฏิบัติ

- 1. ให้นักศึกษาทคลองลบบรรทัคที่ 6 ออก แล้วคอมไฟล์และรันอีกครั้ง สังเกตุผลว่าเป็นอย่างไร
- ให้นักศึกษาทคลองลบหรือบรรทัคที่ 7 ออก แล้วคอม ใฟล์และรันอีกครั้ง สังเกตุผลว่าเป็นอย่าง ไร ข้อสังเกตุที่ได้จาการทคลองนี้คือ

ตัวอย่าง 2 การใช้ JFrame สร้างวินโดว์ และมีการกำหนดคุณสมบัติบางอย่างเพิ่มเติม

```
1: import javax.swing.*;
2:
  import java.awt.Color;
3:
4: class JFrame2 {
5:
      public static void main(String [ ] args) {
6:
            JFrame f = new JFrame();
            f.setTitle("My second JFrame"); // ตั้งชื่อเฟรม
7:
8:
            f.setSize(250, 150);
            f.setResizable(false); // ไม่ให้เปลี่ขนขนาดของเฟรม
9:
            f.getContentPane().setBackground(Color.yellow);
10:
11:
            f.setLocation (400,300); // กำหนดให้เฟรมอยู่ที่ดำแหน่ง x=400 y=300 ของหน้าจอ
12:
            f.setVisible(true);
13:
      }
14: }
```

้ตัวอย่างนี้แตกต่างจากตัวอย่างแรกในประเด็นต่างๆ ดังต่อไปนี้

1

```
    IS311 Programming Concepts: การใช้ JFrame (ปรับปรุง ๑๑ กพ. ๒๕๕๔) 2
    วิธีกำหนดชื่อวินโดว์แตกต่างจากตัวอย่างแรก ตัวอย่างนี้ตั้งชื่อวินโดว์โดยใช้เมท็อด setTitle() แทนที่จะส่ง
ชื่อไปให้ตัวสร้างในตอนสร้างอีอบเจกต์ของ JFrame
```

- สั่งห้ามเปลี่ยนขนาดของวินโดว์ด้วยเมท็อด setResizable (false)
- เปลี่ยนสีพื้นของ content pane เป็นสีเหลืองด้วยเมท็อด setBackgound (Color.yellow) content pane เป็น container สำหรับใส่และแสดงผล component ต่าง ๆ เป็นส่วนประกอบของ JFrame ไม่นับรวมพื้นที่ของ แถบ เมนู งำเป็นต้องสอบถามก่าอ้างอิงถึง content pane จาก JFrame โดยใช้เมท็อด getContentPane ก่อนแล้วจึง ใช้เมท็อด setBackgound ได้
- มีการบอกตำแหน่งที่ให้วินโดว์ปรากฏบนหน้าจอด้วยเมท็อด setLocation()

้ตัวอย่าง 3 สร้างวินโดว์โดยเขียนเป็น subclass ของคลาส JFrame

```
1: import javax.swing.*;
2:
  import java.awt.Color;
3:
4: class JFrame3 {
5:
       public static void main(String [ ] args) {
6:
          MyFrame mf = new MyFrame();
7:
          mf.setVisible(true);
8:
       }
9: }
10:
11: class MyFrame extends JFrame {
12:
       MyFrame() {
13:
           setTitle("My JFrame Subclass");
14:
           setSize(250, 150);
15:
           setResizable(false);
16:
           getContentPane().setBackground(Color.cyan);
17:
           setLocation(400,300);
18:
       }
19: }
```

ตัวอย่างนี้เขียนคลาสใหม่ให้เป็น subclass ของ JFrame แล้วนำคลาสใหม่นี้ไปใช้ ผลลัพธ์จากการทำงานได้ผลทำนอง เดียวกับตัวอย่างที่ 2 นักศึกษาคิดว่าแบบใดน่าใช้กว่ากัน?

ตัวอย่างที่ 4 ใส่ JLabel และ JButton ลงใน container

```
1: import javax.swing.*;
2: import java.awt.Color;
3: import java.awt.*;
4:
5: class JFrame4{
6:
       public static void main(String [ ] args) {
7:
          MyFrame mf = new MyFrame();
8:
          mf.setVisible(true);
9:
       }
10: }
11:
12: class MyFrame extends JFrame {
13:
       MyFrame() {
14:
           Container contentPane;
```

Wanchai Khanti

Department of Management Information Systems Faculty of Commerce and Accountancy, Thammasat University

3

IS311 Programming Concepts: การใช้ JFrame

15:		<pre>setTitle("Add Button");</pre>
16:		setSize(450, 150);
17:		<pre>contentPane = getContentPane();</pre>
18:		<pre>contentPane.setBackground(Color.cyan);</pre>
19:		<pre>contentPane.setLayout(new FlowLayout(FlowLayout.RIGHT));</pre>
20:		<pre>contentPane.add(new JLabel("These are my buttons"));</pre>
21:		<pre>contentPane.add(new JButton("button 1"));</pre>
22:		<pre>contentPane.add(new JButton("Click me please"));</pre>
23:		setLocation(400,300);
24:	}	
25: }		

ตัวอย่างนี้มีการนำ JLabel และ JButton ไปใส่ใน content pane ซึ่งเป็น container และมีการกำหนดให้ content pane ใช้ LayoutManager แบบ FlowLayout เป็นเมื่อรันโปรแกรมนี้แล้วจะได้วินโคว์ดังภาพ

🛓 Add Button		
These are my buttons	button 1	another button
		- 9/

ลองปรับขนาดของวินโดว์ให้เล็กลงมาก ๆ ผลจะเป็นอย่างไร? ลองคลิกที่ปุ่มทั้งสองดูว่ามีปฏิกริยาใดเกิดขึ้นหรือไม่ อย่างไร?

ให้ทุดถองเปลี่ยนก่าพารามิเตอร์ที่ส่งให้ FlowLayout จาก FlowLayout.RIGHT เป็น FlowLayout.LEFT แล้วลองคอมไฟล์แล้วรันอีกครั้ง สังเกตุข้อแตกต่าง อธิบายความหมายของก่า FlowLayout.RIGHT และ FlowLayout.LEFT

หลังจากนั้นให้ ลบหรือ comment บรรทัค ๑៩ แล้วคอมไพล์และรันอีกครั้งสังเกตุผลว่าต่างไปจากเดิมอย่างไร หรือไม่ แล้วสรุปผลการทคลอง ว่าถ้าไม่มีการกำหนคชนิคของ Layout Manger ให้กับ container แล้วละก็ จะยังมี Layout Manager จัดการให้อยู่หรือไม่ ถ้ามีสรุปได้ว่าเป็น Layout Manager แบบใด

ตัวอย่างที่ 5 แสดงการรับมือเหตุการณ์อย่างง่าย

```
1: import javax.swing.*;
2: import java.awt.event.*;
3: import java.awt.*;
4:
5: public class SimpleEventDemo extends JFrame {
    public SimpleEventDemo() {
6:
7:
     JButton jbtOK = new JButton("OK");
8:
     add(jbtOK);
9:
     ActionListener listener = new OKListener();
10:
     jbtOK.addActionListener(listener);
11:
    }
12:
    /** Main method */
13:
    public static void main(String[] args) {
14:
     JFrame frame = new SimpleEventDemo();
15:
     frame.setVisible(true);
```

Wanchai Khanti

Department of Management Information Systems Faculty of Commerce and Accountancy, Thammasat University

16:	<pre>frame.setSize(300, 120);</pre>
17:	<pre>frame.setTitle("SimpleEventDemo");</pre>
18:	<pre>frame.setLocationRelativeTo(null); // Center the frame</pre>
19:	<pre>frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);</pre>
20:	}
21: }	
22:	
23: cl	ass OKListener implements ActionListener {
24: p	public void actionPerformed(ActionEvent e) {
25:	System.out.println("It is OK");
26:	}
27 · }	

ตัวอย่างที่ ๔ ไม่มีการจัดการใด ๆ ถ้ามีการคลิกที่ปุ่ม แต่ปุ่มมันมีปฏิกริยาตามหน้าที่ของมัน ตัวอย่างโปรแกรมที่ ๕ นี้ จะแสดงให้เห็นว่าจะจัดการกับเหตุการณ์เมื่อมีการคลิกที่ปุ่มได้อย่างไร ตัวอย่างนี้มปุ่มซึ่งสร้างจากคลาส JButton 1 ตัวแสดงบน JFrame โดยมีข้อความ OK บนปุ่ม ถ้าคลิกที่ ปุ่มนี้ จะพิมพ์ข้อความ It is OK ในวินโดว์ MS-DOS บรรทัดที่ 23 ถึง 27 เป็นคลาส OKListener ซึ่งเขียนขึ้นมาให้เป็น event listener สำหรับดำเนินการเมื่อคลิกที่ปุ่ม

OK จึงต้อง implement ActionListener นั่นคือต้องเขียนเมท็อด actionPerformed ให้ event source ซึ่งในที่นี้ คือ ปุ่ม OK เรียกเมื่อมีการคลิกที่ปุ่ม OK

🛃 SimpleEventDemo 📃 🔲 🚺	C:\WINDOW 🗆 🗙
ОК	It is OK It is OK It is OK It is OK It is OK It is OK It is OK

ภาพแสดงผลลัพธ์ของโปรแกรม จะแสดงข้องความ It is OK ทุกกครั้งที่มีการคลิกที่ปุ่ม OK

ตัวอย่างที่ 6 แสดงการรับมือเหตุการณ์

```
1:
    import javax.swing.*;
2:
    import java.awt.*;
3:
    import java.awt.event.*;
4:
    class EventDemo extends JFrame implements ActionListener {
5:
    EventDemo() {
6:
         Container contentPane;
7:
         setTitle("Add Button");
8:
         contentPane = getContentPane();
9:
         contentPane.setBackground(Color.cyan);
10:
         contentPane.setLayout(new FlowLayout());
11:
         contentPane.add(new JLabel("These are my buttons"));
12:
         JButton button1 = new JButton("Click me please");
13:
         button1.addActionListener(this);
14:
         contentPane.add(button1);
         JButton button2 = new JButton("Click me too");
15:
16:
         contentPane.add(button2);
17:
         button2.addActionListener(this);
18:
         JButton button3 = new JButton("Don\'t click me");
19:
         contentPane.add(button3);
20:
         button3.addActionListener(this);
21:
        pack();
```

Wanchai Khanti

IS311 Programming Concepts: การใช้ **JFrame**

```
22:
         setLocation(450, 250);
23:
    }
24:
    public void actionPerformed(ActionEvent event) {
         System.out.println("[" + event.getActionCommand() + "] clicked");
25:
26:
         System.out.println(event.paramString());
27:
         if (event.getActionCommand().equals("Don\'t click me")) {
28:
             System.out.println("Oops!!!!! I am disabled, bye bye");
29:
              ((JComponent)event.getSource()).setEnabled(false);
30:
         }
31:
    }
32:
    public static void main(String [ ] args) {
33:
         EventDemo mf = new EventDemo();
34:
         mf.setVisible(true);
35:
         System.out.println("Hello");
         mf.setDefaultCloseOperation(EXIT ON CLOSE);
36:
37:
38:
```

ตัวอย่างนี้ให้กลาส EventDemo เป็น subclass ของ JFrame และมีเมท็อคสำหรับจัดการเหตุการณ์อยู่ใน กลาสเดียวกัน

บรรทัดที่ 13 17 และ 20 เป็นการลงทะเบียนกับ button ทั้ง 3 แจ้งว่าอ็อบเจกต์ปัจจุบันเป็น event listener แสดงว่า เรา จะใช้ event listener ตัวเดียวกันกับทั้ง 3 ปุ่ม

บรรทัดที่ 24 ถึง 31 เป็นเมท็อดที่คอยจัดการกับเหตุการณ์ที่เกิดขึ้นจากการคลิกที่ปุ่มทั้ง 3 ถ้ามีการคลิกที่ปุ่ม จะเกิด เหตุการณ์ประเภท action ซึ่ง event source จะเรียกเมท็อด actionPerformed() ของ event listener ดังนั้นตัวจัดการเหตุการณ์ (event handler) จะต้องมีเมท็อด actionPerformed() ให้เรียกใช้ ซึ่งถูกบังกับให้ implement เมท็อดนี้อยู่แล้วตามที่กำหนดไว้ในอินเทอร์เฟส ActionListener

เมื่อเข้ามาทำงานที่เมท็อด actionPerformed () เมท็อดนี้จะรับอ็อบเจกต์กลาส ActionEvent ซึ่ง event source ได้จัดเตรียมส่งมาให้ เพื่อให้ event listener สามารถทราบรายละเอียดเกี่ยวกับเหตุการณ์ที่ เกิดขึ้นได้

การเรียกเมท็อด getActionCommand () ที่บรรทัด 25 จะใด้ command string ซึ่งกรณีนี้คือข้อความที่ ปรากฏบนปุ่มนั่นเอง

การเรียกเมท็อด paramString () ที่บรรทัด ๒๖ จะใด้สตริงที่ระบุรายละเอียดของเหตุการณ์ เมท็อคนี้มี ประโยชน์ในการเก็บบันทึกรายละเอียดเกี่ยวกับเหตุการณ์ หรือใช้ในการ debug ตอนรันโปรแกรมถ้าคลิกที่ปุ่มแล้ว สังเกตุว่ามันให้ข้อมูลอะไรบ้าง

บรรทัดที่ 27 ใช้ข้อความสั่ง if เพื่อตรวจสอบว่าเหตุการณ์ที่เกิดขึ้นนั้นเกิดที่ปุ่มที่มีข้อความ "don't click me" ใช่หรือไม่ ถ้าใช่จะทำการ disable ปุ่มนี้ไม่ให้กลิกได้อีกต่อไป โดยใช้เมท็อด setEnable() กำหนด enable ให้เป็น false แต่การเรียกเมท็อด setEnable() จะต้องรู้ก่าอ้างอิงของอีอบเจกต์ปุ่ม "don't click me" เสียก่อน ซึ่งสามารถหาได้โดยใช้เมท็อด getSouce() ซึ่งจะคืนก่าเป็นก่าอ้างอิงถึงอีอบเจกต์ที่เป็นต้นกำเนิดของ เหตุการณ์

การเรียกเมท็อค setDefaultCloseOperation (EXIT_ON_CLOSE) ที่บรรทัค 36 เพื่อเปลียนค่าโดย ปริยาย (default) ที่ JFrame จะดำเนินการถ้าคลิกปุ่ม close ของวินโดว์ ซึ่งค่าโดยปริยาย คือ HIDE_ON_CLOSE

IS311 Programming Concepts: การใช้ JFrame (ปรับปรุง ๑๑ กพ. ๒๕๕๔) 6 หมายถึงให้ซ่อนวินโคว์ไว้โคยที่โปรแกรมยังคงทำงานอยู่เพียงแต่ไม่แสคงให้เห็นเท่านั้น การเปลี่ยนค่าโคยปริญายให้เป็น EXIT_ON_CLOSE จะทำให้ JFrame สั่งจบโปรแกรมด้วยกำสั่ง System.exit()

เมื่อรันโปรแกรมนี้จะปรากฏวินโคว์ดังนี้

🛃 Add Button			
These are my buttons	Click me please	Click me too	Don't click me

ส่วนในวินโคว์ MS-DOS จะแสดงข้อกวามต่าง ๆ ของโปรแกรมที่สั่งแสดงด้วยเมท็อด

System.out.println() ให้สังเกตุว่าแสดงข้อมูลอะไรออกมาบ้าง ให้ ลองกดปุ่ม Shift หรือ Ctrl หรือ Alt ด้างไว้ ขณะที่คลิกที่ปุ่มต่าง ๆ ดู ว่าข้อความที่แสดงแตกต่างจากเดิมหรือไม่ อย่างไร ให้ทดลองกดปุ่ม Shift Ctrl และ Alt ทีละปุ่มและหลายปุ่มผสมกันด้วย

นักศึกษาเปิดดูกำอธิบายของคลาส ActionEvent ว่าคลาสนี้ให้ข้อมูลอะไรบ้าง ผลลัพธ์ที่แสดงเป็นตัวเลขหลัง ข้อความ when น่าจะเป็นตัวเลขที่บอกอะไร มี type เป็นอะไร?

ให้ _{COPY} บรรทัดต่อไปนี้ไปแทรกไว้หลังบรรทัด 26 แล้วคอมไพลและรันโปรแกรมอีกครั้ง คราวนี้บอกได้หรือ ยังว่า _{when} เป็นข้อมูลเกี่ยวกับอะไร

System.out.println(event.getWhen() + "==>" + new java.util.Date(event.getWhen()));

ตัวอย่างที่ 7 แสดงการใช้ GridLayout

```
1: import java.awt.*;
2:
    import javax.swing.*;
3:
    public class ButtonGrid extends JFrame {
4:
        ButtonGrid() {
5:
            setLayout(new GridLayout(3,2, 10, 10));
6:
            add(new Button("1"));
7:
            add(new Button("2"));
8:
            add(new Button("3"));
9:
            add(new Button("4"));
10:
            add(new Button("5"));
11:
            add(new Button("6"));
12:
       setSize(600,300);
13:
       setVisible(true);
14:
15:
     public static void main(String [] args) {
16:
     new ButtonGrid();
17:
    }
18: }
```

ตัวอย่างนี้มีการสร้างปุ่มจำนวน ๖ ตัว แล้วใส่ใน JFrame โดยกำหนดให้ใช้ Layout Manager เป็น GridLayout บรรทัดที่ ๕ ใช้ตั้งก่าให้ใช้ Layout Manager เป็น GridLayout ส่งพารามิเตอร์ให้ตัวสร้างเป็น 3,2, 10 และ 10 ตามลำดับ หมายความว่าให้แบ่งกริดเป็น ๓ row ๒ column ให้มีช่องว่างในแนวนอน และ แนวตั้ง อย่างละ ๑๐ pixel จะทำให้แสดงผลดังรูปด้านล่าง สำหรับพารามิเตอร์ตัวที่ ๓ และ ๔ จะไม่ใส่ก็ได้ ถ้าไม่ใส่ถือว่าเป็น ศูนย์ จะไม่เว้นช่องว่างระหว่าง component ที่อยู่ในกริด

IS311 Programming Concepts: การใช้ JFrame



ตัวอย่างที่ 8 แสดงการใช้ GridLayout และมีการจัดการเหตุการณ์

```
1: import java.awt.*;
2: import java.awt.event.*;
3: import javax.swing.*;
4: public class GridDemo extends JFrame implements ActionListener {
5:
   private JTextField input = new JTextField();
    private JLabel outSum = new JLabel();
6:
7:
    private JLabel outMax = new JLabel();
8:
    private JLabel outAvg = new JLabel();
    private JLabel outMin = new JLabel();
9:
10: private int count;
11:
    private double sum;
12:
    private double max = Double.NEGATIVE INFINITY;
13:
    private double min = Double.POSITIVE INFINITY;
14:
15:
    GridDemo () {
16:
       setLayout(new GridLayout(5,2));
17:
       add(new JLabel("Enter a number"));
18:
19:
       add(input);
20:
       input.addActionListener(this);
21:
       add(new JLabel("Sum"));
22:
23:
       add(outSum);
24:
25:
       add(new JLabel("Max"));
26:
27:
       add(outMax);
28:
       add(new JLabel("Average"));
29:
30:
       add(outAvg);
31:
32:
       add(new JLabel("Min"));
33:
34:
       add(outMin);
35:
36:
       setDefaultCloseOperation(EXIT ON CLOSE);
37:
       setSize(380, 400);
38:
       setVisible(true);
```

Wanchai Khanti Department of Management Information Systems Faculty of Commerce and Accountancy, Thammasat University

IS311 Programming Concepts: การใช้ JFrame

```
39:
        }
40:
41:
    public void actionPerformed(ActionEvent event) {
42:
      String s = input.getText();
43:
      double value = Double.parseDouble(s);
44:
      count++;
45:
      sum+=value;
46:
      outSum.setText(""+sum);
47:
      outAvg.setText(""+sum/count);
48:
      if (value > max) {
49:
        max = value;
50:
        outMax.setText(""+max);
51:
      }
52:
      if (value < min ) {
53:
        min = value;
54:
        outMin.setText(""+min);
55:
      }
56:
      input.setText(""); // clear input text field
57:
58:
     }
59:
60:
     public static void main(String [] args) {
61:
      new GridDemo();
62:
     }
63:
   }
```

โปรแกรมนี้เป็นตัวอย่างแสดงให้เห็นการรับข้อมูลเข้าด้วย JTextField เพื่อนำไปคำนวณแล้วแสดงผลด้วย JLabel

ใช้ GridLayout เป็นตัวกวบคุมการแสดงผลในรูปแบบของกริด มีการสร้าง "ป้ายข้อกวาม" ด้วย JLabel ให้มีข้อกวามตามที่กำหนดตั้งแต่เริ่มสร้างวัตถุจำนวน ๕ ป้ายข้อกวาม คือ "Enter a number" "Sum" "Max" "Average" และ "Min" ใส่ไว้ด้านซ้ายของกริดตามลำดับ และมีการสร้าง "ป้ายข้อกวาม" ว่าง ที่ยังไม่ได้ใส่ข้อกวาม ในตอนสร้างวัตถุอีกจำนวน ๔ ตัว เพื่อใช้ในการแสดงผลลัพธ์จากการกำนวณ โปรแกรมจะแสดงผลช่วงเริ่มต้นโปรแกรมดัง รูป

<u>ک</u>	
Enter a number	
Sum	
Max	
Average	
Min	

ถ้าผู้ใช้กรอกตัวเลขใน JTextField แล้วเกาะ Enter จะเกิดเหตุการณ์ Action ตัวจัดการเหตุการณ์จะรับ ข้อมูลจาก JTextField ซึ่งเป็นสตริงนำไปแปลงเป็นตัวเลขแล้วกำนวณหาก่าทางสถิติต่าง ๆ คือ Sum Max Average

IS311 Programming Concepts: การใช้ JFrame (ปรับปรุง ๑๑ กพ. ๒๕๕๔)

9

และ Min แล้วแสดงผลใน JLabel ด้านขวาของกริด ผู้ใช้สามารถกรอกตัวเลขชุดใหม่เพิ่มเติมได้ โปรแกรมจะคำนวณ ้ก่าสถิติต่าง ๆ ให้ใหม่ทุกกรั้งที่เกาะ Enter แล้วนำไปแสดงผลอีกกรั้ง ดังตัวอย่างในรูปด้านล่าง

<u>گ</u>	
Enter a number	
Sum	290.0
Max	200.0
Average	96.66666666666667
Min	40.0

บรรทัดที่ ๕ สร้างวัตถุ JTextField อ้างถึงด้วยตัวแปร input นำ JTextField ตัวนี้ใส่ในกริดที่บรรทัด ๑៩ หลังจากนั้นทำการลงทะเบียน event listener เข้ากับ input ด้วยกำสั่ง input.addActionListener(this) ส่วนเมท็อดที่คอยจัดการเห็นการณ์กือ actionPerformed() เริ่มต้นที่ บรรทัก ๔๑ ในเมท็อดจะมีการอ่านค่าข้อมูลจาก JTextField input ด้วยเมท็อด getText () ซึ่งได้เป็นสตริง ้นำไปคำนวณไม่ได้จึงต้องแปลงค่าให้เป็น double แล้วจึงนำไปคำนวณหาค่าผลรวาม ค่าเฉลี่ย สูงสุด ต่ำสุด แล้วแสดงผล ใน JLabel ต่อไป การส่งข้อมูลไปแสดงผลด้วย JLabel จะใช้เมท็อค setText () ซึ่งรับพารามิเตอร์เป็นสตริง แต่ ้ข้อมูลที่คำนวณได้เป็นชนิด double จึงต้องแปลงให้เป็นสตริงก่อนด้วยวิธีง่าย ๆ คือสร้างเป็นนิพจน์สตริง ด้วยรูปแบบเช่น "" + sum การขึ้นต้นนิพจน์ด้วย "" จะทำให้นิพจน์นี้เป็นนิพจน์สตริง คอมไพเลอร์จะแปลงข้อมูลจากชนิดอื่นให้เป็น สตริงก่อนที่จะนำมาต่อท้าย "" ซึ่งเป็นสตริงว่างเปล่า

์ โปรแกรมนี้มีข้อควรระวังในการใช้คือ ถ้ากรอกข้อมูลที่ไม่ใช่ตัวเลขจะเกิดปัญหาในแปลงข้อมูลให้เป็นตัวเลข ทคลองป้อนข้อมูลที่ไม่ใช่ตัวเลขแล้วสังเกตุผลว่าเป็นอย่างไร จะแก้ปัญหานี้อย่างไร?