

# ใบงานการทดลองที่ 8

## เรื่อง Wrapper Class และ Math Class

### 1. จุดประสงค์ทั่วไป

- 1.1. รู้และเข้าใจการติดต่อระหว่างงาน และการเรียกใช้งานคลาสสำคัญทางคณิตศาสตร์
- 1.2. รู้และเข้าใจการใช้งานคลาส Wrapper

### 2. เครื่องมือและอุปกรณ์

เครื่องคอมพิวเตอร์ 1 เครื่อง ที่ติดตั้งโปรแกรม Eclipse

### 3. ทฤษฎีการทดลอง

#### 3.1. Primitive Data Type คืออะไร ?

Primitive data type หรือชนิดข้อมูลพื้นฐาน

เป็นชนิดข้อมูลในโปรแกรมที่เก็บค่าเป็นค่าพื้นฐานแบบตรรกะ (boolean), ตัวเลขจำนวนเต็ม (integer)

หรือทศนิยม (floating point) และอักขระ (character) โดยที่ไม่ได้ถูกสร้างขึ้นมาจากการรวมข้อมูลชนิดอื่น

ๆ

#### 3.2. Wrapper Class คืออะไร? มีอะไรบ้าง? อธิบายพร้อมยกตัวอย่างประกอบ

Wrapper class เป็นคลาสที่ใช้สำหรับแปลงหรือแพ็คเกจข้อมูลพื้นฐาน (primitive data type) เช่น int, double, char

เป็นอ็อบเจกต์ เพื่อให้สามารถนำไปใช้กับคลาสที่ต้องการรับพารามิเตอร์เป็น Object ได้ โดยสามารถใช้ method และ property ต่าง ๆ ของ Wrapper class ได้ตามปกติ

#### 3.3. ยกตัวอย่าง Method ที่สามารถเรียกใช้งานได้ใน Wrapper Class

```
String numStr = "123";
```

```
int num = Integer.parseInt(numStr);
```

#### 3.4. คำสั่ง Math.max( x, y ); ใช้สำหรับทำอะไร? อธิบายพร้อมยกตัวอย่างประกอบ

คำสั่ง Math.max(x, y) ใช้สำหรับหาค่าสูงสุด (maximum) ระหว่าง x และ y

โดยจะคืนค่าเป็นตัวเลขที่มากที่สุดระหว่าง x และ y ออกมา

#### 3.5. คำสั่ง Math.min( x, y ); ใช้สำหรับทำอะไร? อธิบายพร้อมยกตัวอย่างประกอบ

คำสั่ง Math.min(x, y) ใช้สำหรับหาค่าต่ำสุด (minimum) ระหว่าง x และ y

โดยจะคืนค่าเป็นตัวเลขที่น้อยที่สุดระหว่าง x และ y ออกมา

3.6. คำสั่ง Math.sqrt( x ) ; ใช้สำหรับทำอะไร? อธิบายพร้อมยกตัวอย่างประกอบ

คำสั่ง Math.sqrt(x) ใช้สำหรับหาค่า square root (รากที่สอง) ของตัวเลข x โดยจะคืนค่าเป็นตัวเลขที่เป็น double ออกมา

ตัวอย่างการใช้งาน Math.sqrt(x) เพื่อหาค่า square root ของตัวเลข 16 คือ 4

3.7. คำสั่ง Math.abs( x ) ; ใช้สำหรับทำอะไร? อธิบายพร้อมยกตัวอย่างประกอบ

คำสั่ง Math.abs(x) ใช้สำหรับหาค่า absolute (ค่าสัมบูรณ์) ของตัวเลข x โดยไม่สนใจเครื่องหมาย (sign) ของ x ว่าเป็นบวกหรือลบ และจะคืนค่าเป็นตัวเลขที่เป็นจำนวนเต็มหรือจำนวนจริง (integer หรือ double) ออกมา

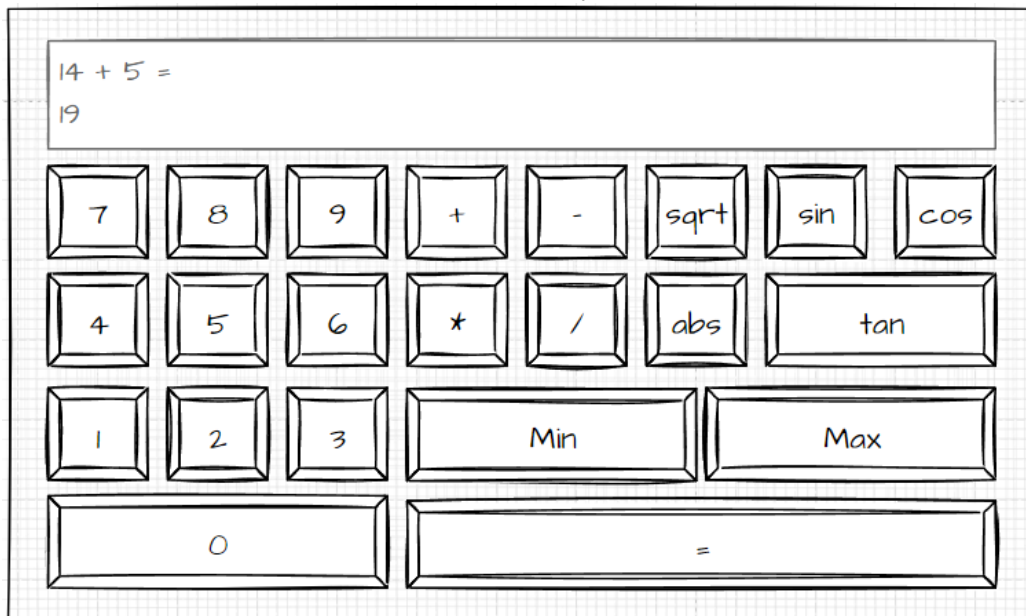
3.8. คำสั่ง Math.random() ; ใช้สำหรับทำอะไร? อธิบายพร้อมยกตัวอย่างประกอบ

คำสั่ง Math.random() ใช้สำหรับสุ่มตัวเลขที่เป็นจำนวนจริง (double) ที่มีค่าอยู่ในช่วง 0.0 ถึง 1.0 (ไม่รวม 1.0) โดยตัวเลขที่ได้จะมีการสุ่มแบบสุ่มที่ไม่ซ้ำกันและเป็นลำดับอนันต์ที่ไม่สม่ำเสมอ โดยสามารถนำผลลัพธ์ที่ได้มาใช้ในการสุ่มตัวเลขหรือสิ่งอื่นๆ ได้

#### 4. ลำดับขั้นการปฏิบัติการ

4.1. จงพัฒนาและปรับปรุงโปรแกรมเครื่องคิดเลขจากใบงานทดลองที่ 5 โดยการเพิ่มปุ่มให้กับเครื่องคิดเลขที่มีความสามารถเพิ่มเติมดังต่อไปนี้

- 4.1.1. ปุ่ม sqrt เพื่อใช้ในการหาค่ารากที่ 2 ของตัวเลขที่ผู้ใช้กรอก
- 4.1.2. ปุ่ม abs เพื่อใช้ในการหาค่าสัมบูรณ์ของตัวเลขที่ผู้ใช้กรอก
- 4.1.3. ปุ่ม sin เพื่อใช้ในการหา Sin ของตัวเลขที่ผู้ใช้กรอก
- 4.1.4. ปุ่ม cos เพื่อใช้ในการหา Cos ของตัวเลขที่ผู้ใช้กรอก
- 4.1.5. ปุ่ม tan เพื่อใช้ในการหา Tan ของตัวเลขที่ผู้ใช้กรอก
- 4.1.6. ปุ่ม Min เพื่อใช้ในการหาค่าที่น้อยที่สุดของตัวเลขที่ผู้ใช้กรอกมา 2 ค่า
- 4.1.7. ปุ่ม Max เพื่อใช้ในการหาค่าที่มากที่สุดของตัวเลขที่ผู้ใช้กรอกมา 2 ค่า



โค้ดโปรแกรมภายในปุ่มเครื่องหมายเท่ากับ (เฉพาะส่วนที่เพิ่มเข้ามาในใบงานการทดลองนี้เท่านั้น)

```

btnDot.setText(".");
btnDot.setFont(SWTResourceManager.getFont("Segoe UI", 11, SWT.BOLD));
btnDot.setBounds(79, 325, 63, 57);

Button btnEqual = new Button(shell, SWT.NONE);
btnEqual.addSelectionListener(new SelectionAdapter() {
    @Override
    public void widgetSelected(SelectionEvent e) {

        int one = Integer.parseInt(text1.getText());

        int two = Integer.parseInt(text.getText());
        int sub = Integer.parseInt(text.getText());

        actionRecived.setText("d");

        if (actionRecived.getText().equals("f"))
        { sub = one * two ;
          text.setText(String.valueOf(sub));
        }

        if (actionRecived.getText().equals("a"))
        { sub = one / two;
          text.setText(String.valueOf(sub));
        }

        if (actionRecived.getText().equals("s"))
        { sub = one - two;
          text.setText(String.valueOf(sub));
        }

        if (actionRecived.getText().equals("d"))
        { sub = one + two;
          text.setText(String.valueOf(sub));
        }
    }
});
btnEqual.setText("=");
btnEqual.setFont(SWTResourceManager.getFont("Segoe UI", 11, SWT.BOLD));
btnEqual.setBounds(148, 325, 63, 57);

Button btnPerceNr = new Button(shell, SWT.NONE);
btnPerceNr.addSelectionListener(new SelectionAdapter() {
    @Override
    public void widgetSelected(SelectionEvent e) {
        text.setText(text.getText()+"(%)");
    }
}

```

## 5. สรุปผลการปฏิบัติการ

---

---

---

## 6. คำถามท้ายการทดลอง

6.1. Wrapper Class ควรใช้เมื่อใด? เพราะอะไร?

---

---

---

---

6.2. ข้อควรระวังในการใช้ Math Class คืออะไร?

---

---

---

---