

ใบงานการทดลองที่ 7

เรื่อง Overloading Method และ Overriding Method

1. จุดประสงค์ทั่วไป

- 1.1. รู้และเข้าใจการพ้องรูปในการเขียนโปรแกรมเชิงวัตถุ
- 1.2. รู้และเข้าใจการสืบทอดของวัตถุ โครงข่ายของวัตถุ โครงสร้างของโปรแกรมเชิงวัตถุ

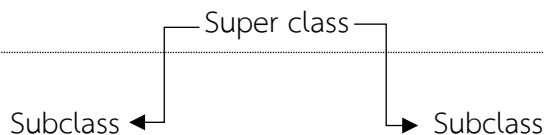
2. เครื่องมือและอุปกรณ์

เครื่องคอมพิวเตอร์ 1 เครื่อง ที่ติดตั้งโปรแกรม Eclipse

3. ทฤษฎีการทดลอง

- 3.1. Super Class คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

คือ class ที่อยู่บนสุดเป็น class แม่ของ subclass



- 3.2. การพ้องรูป(Polymorphism) คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

คือการที่ class แม่แปลงร่างเป็น class ลูกเพื่อใช้ method และ property ของ class ลูกได้

- 3.3. Overloading Method คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

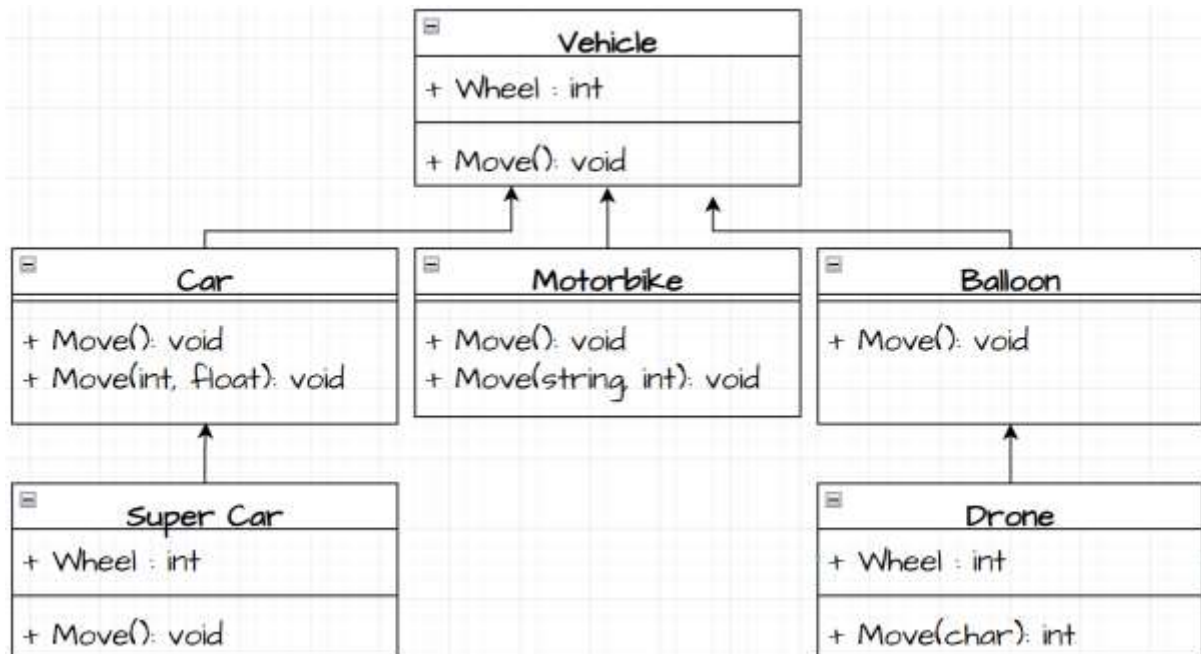
คือ Method ที่มีชื่อเหมือนกัน แต่มี Parameter ต่างกัน

- 3.4. Overriding Method คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

คือ method ที่มีชื่อเหมือนกันกับ class แม่ แต่มีรายละเอียดที่ต่างกัน

4. ลำดับชั้นการปฏิบัติการ

- 4.1. จงสร้างคลาสต่างๆ และทำการสืบทอดกันตามโครงสร้างดังต่อไปนี้ โดยแต่ละ Method จะต้องสร้างด้วยชื่อเดียวกัน แต่มีกระบวนการทำงานที่แตกต่างกัน พร้อมทั้งแสดงผลการทำงานของการทำงานภายในแต่ละคลาสให้ดูเพื่อแสดงถึงความแตกต่าง ตามหลักการของ Overloading Method และ Overriding Method



- 4.2. จงเขียนโค้ดโปรแกรมจากผังงานดังกล่าว ที่แสดงให้เห็นว่าเป็น Overloading Method

โค้ดโปรแกรม

```
package LAB;

public class vehicle {
    public int wheel ;

    public vehicle() {
        System.out.println("\n | Vehicle start |");
    } //end method

    /**
     * Start move vehicle
     */
    public void move() {
        System.out.print(" | Vehicle Move  |");
    } //end method

    /**
     * Just moving vehicle
     * @param name user input name Driver
     */
    public void move( String name) {
        System.out.println("--|My Driver ::" + name + " --->");
    } //end method
}
```

โค้ดโปรแกรม

```
package LAB;

public class vehicle {
    public int wheel ;

    public vehicle() {
        System.out.println("\n | Vehicle start |");
    } //end method

    /**
     * Start move vehicle
     */
    public void move() {
        System.out.print(" | Vehicle Move  |");
    } //end method

    /**
     * Just moving vehicle
     * @param name user input name Driver
     */
    public void move( String name) {
        System.out.println("--|My Driver ::" + name + " --->");
    } //end method
}
```

4.4. จงเขียนโค้ดโปรแกรมแสดงการเรียกใช้งานในการสร้างวัตถุทั้งหมดเพื่อมาทดสอบในฟังก์ชันหลัก

โค้ดโปรแกรม

```
package LAB;

public class main {
    public static void main (String[] args) {
        vehicle vehicle = new vehicle();
        vehicle.move();
        vehicle.move("Saksit Sanchai");
    }
}
```

```

car car = new car();
car.move(80, 40);

supercar supercar = new supercar();
supercar.move();

motobike motobike = new motobike();
motobike.move("honda", 120);

ballon ballon = new ballon();
ballon.move();

drone drone = new drone();
drone.move('a', 120);
}

```

4.5. ผลลัพธ์การทำงานของโปรแกรม

ผลลัพธ์การทำงาน (พร้อมคำอธิบายประกอบ)

```

| Vehicle start |
| Vehicle Move  |--|My Driver ::Saksit Sanchai --->

| Vehicle start |
| car start ! |
My Driver , drive at speed 80 km/hr form home 40.0 kilometer

| Vehicle start |
| car start ! |
| Supercar start ! |
there is Supercar, So verey Speed drive

| Vehicle start |
| motobike start ! |
This honda motorcycle at speed on the roads 120 km/hr

| Vehicle start |
| balloon start ! |
The balloon floats 500 feet above the ground

| Vehicle start |
| balloon start ! |
| drone start ! |
Mr. a ,farmer flies a drone at 120 above sea level. to survey the vegetable growing area

```

5. สรุปผลการปฏิบัติการ

จากการปฏิบัติพบว่าการส่งข้อมูลผ่าน class จะต้องระวังเรื่องการเรียกใช้งานถ้าประกาศผิดก็จะไม่สามารถเรียกใช้งานได้

6. คำถามท้ายการทดลอง

6.1. ข้อแตกต่างระหว่าง Overloading method และ Overriding method คืออะไร?
แตกต่างกันตรงที่ Overloading method เป็น Method ที่มีชื่อเหมือนกัน แต่มี Parameter ต่างกัน
ส่วน Overriding method จะเป็น method ที่มีชื่อเหมือนกันกับ class แม่ แต่มีรายละเอียดที่ต่างกัน

6.2. คุณคิดว่าหลักการของ Polymorphism จะสามารถนำมาประยุกต์ใช้ในการเขียนโปรแกรมของคุณได้
อย่างไรบ้าง ?
ประยุกต์ใช้ในการที่เราต้องการสร้าง class แม่ แล้วเราต้องการเรียกใช้งาน method ของ class ลูก
