

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 คลาสแม่ หรือ คลาสต้นแบบ

```
class Parent {  
    void show()  
    {  
        System.out.println("Parent's show()");  
    }  
}
```

- 3.2. การพ้องรูป(Polymorphism) คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

คลาสแม่แปลงร่างกลายเป็นคลาสลูก และเรียกใช้ Method ของคลาสลูกได้

```
// A Simple Java program to demonstrate  
// method overriding in java  
  
// Base Class  
class Parent {  
    void show()  
    {  
        System.out.println("Parent's show()");  
    }  
}  
  
// Inherited class  
class Child extends Parent {  
    // This method overrides show() of Parent  
    @Override  
    void show()  
    {  
        System.out.println("Child's show()");  
    }  
}
```

```

}

// Driver class
class Main {
    public static void main(String[] args)
    {
        // If a Parent type reference refers
        // to a Parent object, then Parent's
        // show is called
        Parent obj1 = new Parent();
        obj1.show();

        // If a Parent type reference refers
        // to a Child object Child's show()
        // is called. This is called RUN TIME
        // POLYMORPHISM.
        Parent obj2 = new Child();
        obj2.show();
    }
}

```

3.3. Overloading Method คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ
 คือ Method ที่มีชื่อเหมือนกัน แต่มี Parameter ต่างกัน

```

public void Move() {
    System.out.println("Vehicle Move by");
}

/**
 * Just moving Vehicle like a pro.
 * @param Name String Input name of user
 */

public void Move(String Name) {
    System.out.println "[" + Name + ""];
}

```

3.4. Overriding Method คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ
 คือ Method ที่มีชื่อเหมือนกันกับ Method ในคลาสแม่ แต่มีรายละเอียดที่แตกต่างกัน

```

คลาสแม่ public void Move() {
    System.out.println("Vehicle Move by");
}

```

```

คลาสลูก public class Car extends Vehicle {
    /*public Car() {
        System.out.println( "!!!!-Car-!!!!" );
    }//end Method*/

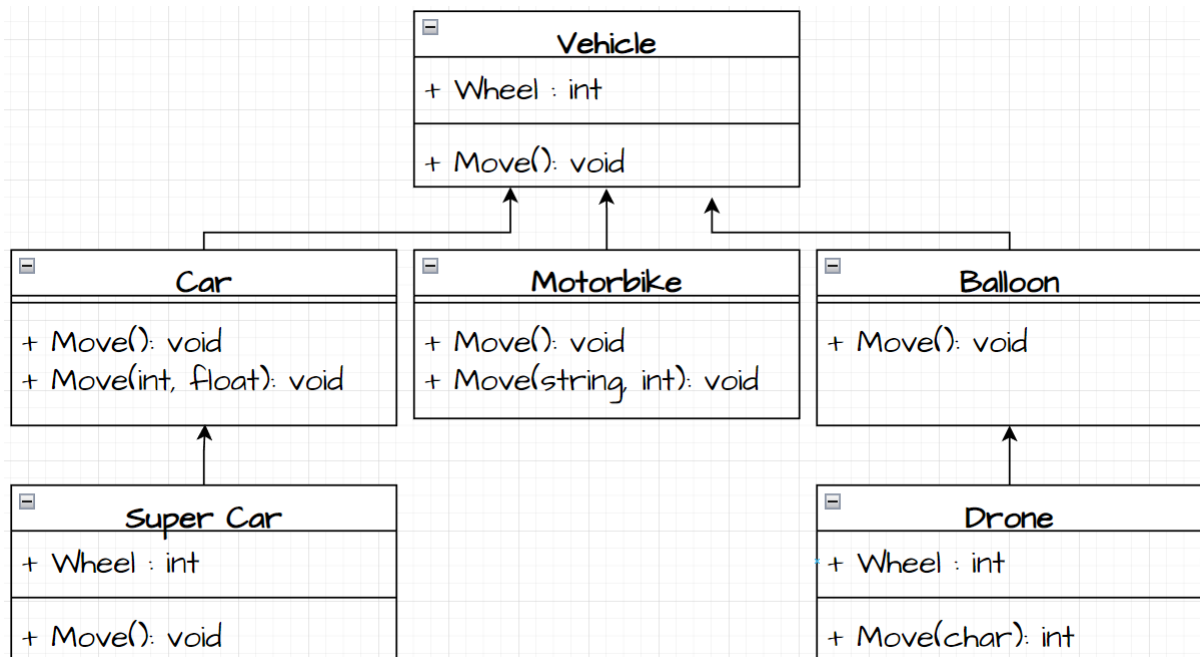
    public void Move() {
        System.out.println ("Car Move");
    }
}

```

```
}//end method
```

4. ลำดับขั้นการปฏิบัติการ

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



- 4.2. จงเขียนโค้ดโปรแกรมจากผังงานดังกล่าว ที่แสดงให้เห็นว่าเป็น Overloading Method

โค้ดโปรแกรม

```
1 package Lab7;
2
3 public class Vehicle {
4     public int Wheel = 4 ;
5     /* public Vehicle() {
6         System.out.println("--| Vehicle Start." );
7     }//end method
8     */
9
10    /**
11     * This is Move like a normal people
12     */
13    public void Move() {
14        System.out.println("Vehicle Move by");
15    }
16
17    /**
18     * Just moving Vehicle like a pro.
19     * @param Name String Input name of user
20     */
21
22    public void Move(String Name) {
23        System.out.println "[" + Name + "]");
24    }
25
26 }
27
```

```

1 package Lab7;
2
3 public class Car extends Vehicle {
4     /*public Car() {
5         System.out.println( "!!!!-Car-!!!!" );
6     } //end Method*/
7
8     public void Move() {
9         System.out.println ( "Car Move" );
10    } //end method
11
12    public void Move(int x, float y) {
13        System.out.println("The car have "+ x +" wheels and has weight "+ y);
14    } //end method
15
16
17 }
18

```

```

1 package Lab7;
2
3 public class Motorbike extends Vehicle {
4     /*public Motorbike() {
5         System.out.println("^^^^-Motorbike-^^^^");
6     } //end Method*/
7
8     public void Move() {
9         System.out.println ( "Motorbike Move" );
10    } //end method
11
12    public void Move(String Name ,int x) {
13        System.out.println("The Motorbike drove by "+ Name +" and MotorBike have "+ x +" wheels");
14    } //end method
15
16 }
17

```

4.3. จงเขียนโค้ดโปรแกรมจากผังงานดังกล่าว ที่แสดงให้เห็นว่าเป็น Overriding Method

โค้ดโปรแกรม

```
1 package Lab7 ;
2
3 public class Main {
4
5     public static void main(String[] args) {
6         Vehicle V = new Vehicle() ;
7
8         int x = V.Wheel ;
9         float y = (float) 456.4;
10
11         V.Move();
12         V.Move("User");
13
14         Car C = new Car();
15         C.Move();
16         C.Move(x, y);
17
18         Motorbike M = new Motorbike();
19         M.Move();
20         M.Move("pram", x - 2);
21
22         Balloon B = new Balloon();
23         B.Move();
24
25         SuperCar S = new SuperCar();
26         S.Move();
27
28         Drone D = new Drone();
29         D.Move();
30
31     }
32
33
34
35 }
```

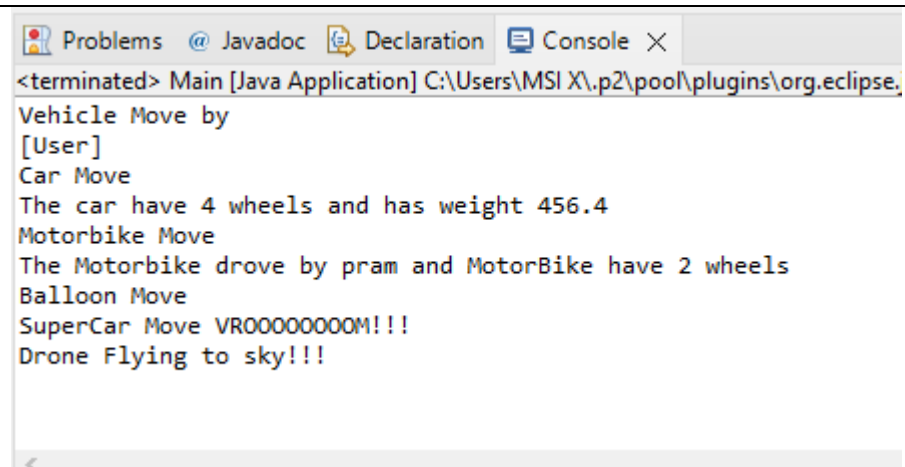
4.4. จงเขียนโค้ดโปรแกรมแสดงการเรียกใช้งานในการสร้างวัตถุทั้งหมดเพื่อมาทดสอบในฟังก์ชันหลัก

โค้ดโปรแกรม

```
1 package Lab7 ;
2
3 public class Main {
4
5     public static void main(String[] args) {
6         Vehicle V = new Vehicle() ;
7
8         int x = V.Wheel ;
9         float y = (float) 456.4;
10
11         V.Move();
12         V.Move("User");
13
14         Car C = new Car();
15         C.Move();
16         C.Move(x, y);
17
18         Motorbike M = new Motorbike();
19         M.Move();
20         M.Move("pram", x - 2);
21
22         Balloon B = new Balloon();
23         B.Move();
24
25         SuperCar S = new SuperCar();
26         S.Move();
27
28         Drone D = new Drone();
29         D.Move();
30
31     }
32
33
34
35 }
```

4.5. ผลลัพธ์การทำงานของโปรแกรม

ผลลัพธ์การทำงาน (พร้อมคำอธิบายประกอบ)



```
<terminated> Main [Java Application] C:\Users\MSI X\p2\pool\plugins\org.eclipse.  
Vehicle Move by  
[User]  
Car Move  
The car have 4 wheels and has weight 456.4  
Motorbike Move  
The Motorbike drove by pram and MotorBike have 2 wheels  
Balloon Move  
SuperCar Move VR00000000M!!!  
Drone Flying to sky!!!
```

5. สรุปผลการปฏิบัติการ

สามารถเรียกใช้งานคลาสแม่ คลาสลูกได้ง่ายยิ่งขึ้น ดูตัวโค้ดโปรแกรมได้สะดวกยิ่งขึ้น

6. คำถามท้ายการทดลอง

6.1. ข้อแตกต่างระหว่าง Overloading method และ Overriding method คืออะไร?

Overriding Method คือ Method ที่มีชื่อเหมือนกันกับ Method ในคลาสแม่ แต่มีรายละเอียดที่แตกต่างกัน

ส่วน Overloading Method คือ Method ที่มีชื่อเหมือนกัน แต่มี Parameter ต่างกัน

6.2. คุณคิดว่าหลักการของ Polymorphism จะสามารถนำมาประยุกต์ใช้ในการเขียนโปรแกรมของคุณได้อย่างไรบ้าง ?

นำมาใช้เกี่ยวกับการเรียกใช้คลาสแม่คลาส คลาสลูกสะดวกยิ่งขึ้น และในคลาสเมนก้จะดูโค้ดยิ่งขึ้น
