แบบฝึกหัดปฏิบัติการคาบที่ 2: Class and Object

1. ให้ศึกษาและทดลองพิมพ์ Code ตัวอย่างการสร้างคลาสจากตัวอย่างต่อไปนี้

ต้องการสร้างคลาสของเลขเชิงซ้อนที่ประกอบไปด้วยส่วนจริง(r) และส่วนจินตภาพ (i) เช่น 3+5i

- โดยตัวเลขเชิงซ้อน ตัวจะถือว่าเป็น 1 1Object
- การทำงานของตัวเลขเชิงซ้อนที่เราจะสามารถเรียกเพื่อให้ทำงานได้คือ
    - การบวก(add) โดยการส่งตัวเลขเชิงซ้อนอีก ตัวเข้ามาบวก 1
    - และการแสดงผลตัวเลข(print) โดยนำส่วนจริงและส่วนจินตภาพมาแสดงผลในรูปแบบ r+ci

เมื่อต้องการสร้างตัวเลขเชิงซ้อนขึ้นมา 1ตัวจะต้องระบุรูปแบบของตัวเลขที่จะสร้าง(Type/class) และตามด้วยตัวแปรอ้างอิงวัตถุ (ตัวแปร) ที่จะเก็บที่อยู่ของวัตถุจริง ๆ และตามด้วยการสร้างวัตถุในหน่วยความจำที่ประกอบด้วยแอทริบิวต์และเมธอดตามที่กำหนดไว้ในคลาส โดยการใช้คำสั่ง new และตามด้วย Constructor ที่ทำหน้าที่กำหนดค่าเริ่มต้นให้วัตถุ เช่น

```
Complex a = new Complex(1.0, 2.0);
```

เมื่อสร้างวัตถุด้วยคำสั่งดังกล่าวจะมีรูปแบบการเก็บข้อมูลในหน่วยความจำคือ

```
// ComplexTest.java
class Complex {
   private double r, i;
   Complex(double r, double i) {
      this.r = r; this.i = i;
   }
   Complex(Complex c) {
      this(c.r, c.i);
   }
   public void add(Complex c) {
      r += c.r;
      i += c.i;
   }
   public void print() {
     System.out.println(r + "+ i" + i);
   }
}
class ComplexTest {
   public static void main(String args[]) {
      Complex a = new Complex(1.0, 2.0);
      Complex b = new Complex(3.0, 4.0);
      Complex c = new Complex(a);
      c.add(b);
      c.print();
   }
}
```

จากโปรแกรมดังกล่าวให้เพิ่มความสามารถของ Object ให้สามารถลบ คูณ และหารได้

```
// ComplexTest.java
class Complex {
   private double r, i;
   Complex(double r, double i) {
      this.r = r; this.i = i;
   }
   Complex(Complex c) {
      this(c.r, c.i);
   }
   public void add(Complex c) {
      r += c.r;
      i += c.i;
   }
   public void subtract(Complex c) {
        r -= c.r;
        i -= c.i;
    }
    public void multiply(Complex c) {
        double tempR = this.r;
        double tempI = this.i;
        r = tempR * c.r - tempI * c.i;
        i = tempR * c.i + tempI * c.r;
    }
    public void divide(Complex c) {
        double temp = c.r * c.r + c.i * c.i;
        Complex tempC = new Complex(c.r,-c.i);
        if(temp == 0)
                System.out.println("Error: Divided by 0");
        else{
                this.multiply(tempC);
                r /= temp;
                i /= temp;
        }
    }
   public void print() {
     System.out.println(r + "+ i" + i);
   }
}
class ComplexTest {
   public static void main(String args[]) {
      Complex a = new Complex(1.0, 2.0);
      Complex b = new Complex(3.0, 4.0);
      Complex c = new Complex(a);
      c.add(b);
      c.print();
   }
}
```

2. จงหาข้อผิดพลาดของส่วนของโปรแกรมต่อไปนี้ โดยวงกลมตำแหน่งที่ผิด เขียนหมายเลขกำกับแต่ละตำแหน่งและอธิบายเหตุผลด้านล่างว่าเหตุใดตำแหน่งดังกล่าวจึงผิด ถ้าไม่มีข้อผิดพลาดให้ตอบว่าไม่มีข้อผิดพลาด

ส่วนของโปรแกรม	ผลลัพธ์

`public class ShowErrors{`
`  public static void main(String[] args){`
`  ShowErrors t= new ShowErrors(5);`
`  }`
`}`
	ไม่มี constructor ที่รับ argument

`public class ShowErrors{`
`  public static void main(String[] args){`
`  ShowErrors t= new ShowErrors();`
`  t.x();`
`  }`
`}`
	ไม่มี method x

`public class ShowErrors{`
`  public void method1(){`
`  Circle c;`
`  System.out.println("What is radius "+`
`c.getRadius());`
`  c=new Circle;`
`  }`
`}`
	Syntax error, แก้: `c=new Circle();`
	
	ไม่สามารถเรียกใช้ `c.getRadius()` ได้เพราะยังไม่เคย assign new object ให้ตัวแปร c

`public class ShowErrors{`
`  public static void main(String[] args){`
`  C c = new C(5.0);`
`  System.out.println(c.value);`
`  }`
`}`
`class C{`
`  int value=2;`
`}`
	ไม่มี constructor ที่รับ argument

3. จงอธิบายการทำงานของโปรแกรมต่อไปนี้

```
public class Test{
   public static void main(String[] args){
      Count myCount = new Count();
      int times=0;
      for (int i=0; i<100 ;i++)
         increment(myCount, times);
      System.out.println("count is "+myCount.count);
      System.out.println("times is "+times);
   }
   public static void increment(Count c, int times){
      c.count++;
      times++;
   }
}
public class Count{
   public int count;

   public Count(int c){
     count =c;
   }
   public Count(){
     count =1;
   }
}
```

แสดงผลลัพธ์จากโปรแกรม

```
count is 101
times is 0
```

3.1 ข้อแตกต่างของการส่งค่าพารามิเตอร์ของ Primitive type และการส่งค่าพารามิเตอร์ของ reference type จากโปรแกรมต่อไปนี้

Primitive type เป็นการส่งแค่ค่าของข้อมูล หากมีการแก้ไขค่าของข้อมูล จะเกิดการเปลี่ยนแปลงแค่ใน method นั้น ๆ ไม่ส่งผลกับข้อมูลต้นฉบับ ส่วน reference type เป็นการส่งที่อยู่ของข้อมูล การแก้ไขค่าของข้อมูลจะส่งผลถึงตัวข้อมูลต้นฉบับด้วย

3.2 ระบุจุดที่มีการส่งค่าพารามิเตอร์ของ Primitive type และการส่งค่าพารามิเตอร์ของ reference type

increment(myCount, times);

- myCount เป็นการส่งค่าแบบ reference type
- times เป็นการส่งค่าแบบ primitive type

4. จงแสดง output จากโปรแกรมต่อไปนี้พร้อมอธิบายการทำงานในแต่ละบรรทัด

	ส่วนของโปรแกรม	ผลลัพธ์

1
2
3
4
5
6
7
8
9
10
11
12
13
	`public class Test {`
	`  public static void main(String[] args) {`
	`  int[] a = {1, 2}; //สร้าง array a`
	`  swap(a[0], a[1]); //ส่งค่าแบบ Primitive type ให้ method`
	`  System.out.println("a[0] = " + a[0] + " a[1] = " + a[1]); //output`
	`  }`
	`  public static void swap(int n1, int n2) { //สลับค่าของ n1, n2 ภายใน method`
	`  int temp = n1; //ให้ตัวแปร temp มีค่าเท่ากับ n1`
	`  n1 = n2; //จากนั้นให้ n1 มีค่าเท่ากับ n2`
	`  n2 = temp; //และให้ n2 มีค่าเท่ากับ temp ซึ่งมีค่าเท่ากับ n1 ตอนแรก`
	`  }`
	`}`
		a[0] = 1 a[1] = 2

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
	`public class Test {`
	`  public static void main(String[] args) {`
	`  T t1 = new T(); //สร้าง objects ใหม่ พร้อมรันโค้ดใน T( ): i เพิ่มจาก 0 เป็น 1`
	`  T t2 = new T(); // i เพิ่มจาก 1 เป็น 2 (i ไม่ได้เริ่มจาก 0 เนื่องจากเป็นตัวแปร static ของ class T)`
	`  System.out.println("t1's i = " + t1.i +" and j = " + t1.j);`
	`  System.out.println("t2's i = " + t2.i +" and j = " + t2.j);`
	`  } //output ค่า i และ j ของ t1 และ t2 ตามลำดับ`
	`}`
	`class T {`
	`  static int i = 0; //ประกาศตัวแปร`
	`  int j = 0;`
	`  T() { //constructor T`
	`  i++; //เพิ่มค่า i โดย 1`
	`  j = 1; //ให้ j = 1`
	`  }`
	`}`
		t1's i = 2 and j = 1
		t2's i = 2 and j = 1