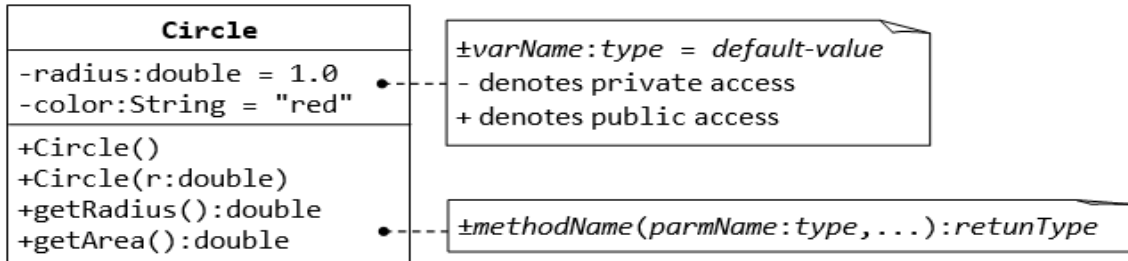


TUẦN 2: CƠ BẢN VỀ HƯỚNG ĐỐI TƯỢNG TRONG JAVA

Bài 1:

Ví dụ: Ví dụ dưới đây sẽ minh họa cho những khái niệm cơ bản của OOP



Một lớp (class) Circle được thiết kế như Class Diagram trên. Yêu cầu đề bài:

- Hai private attribute (hoặc instance variable): radius (kiểu dữ liệu là double) và color (kiểu dữ liệu là String), với 2 giá trị mặc định tương ứng là 1.0 và “red”.
- Hai overload constructor – một constructor mặc định không có đối số, và một constructor có một đối số là radius.
- Hai public method: getRadius() và getArea(), trả về radius và area của class Circle.

Mã nguồn của Circle.java là:

```
public class Circle {
    // private instance variable, not accessible from outside this class
    private double radius;
    private String color;

    // The default constructor with no argument.
    // It sets the radius and color to their default value.
    public Circle() {
        radius = 1.0;
        color = "red";
    }

    // 2nd constructor with given radius, but color default
    public Circle(double r) {
        radius = r;
        color = "red";
    }

    // A public method for retrieving the radius
    public double getRadius() {
        return radius;
    }
}
```

```
// A public method for computing the area of circle
public double getArea() {
    return radius*radius*Math.PI;
}
}
```

Sinh viên thử compile “Circle.java”, xem chương trình có thực thi được không? Lớp Circle không có hàm main(). Nên chương trình không được thực thi được.

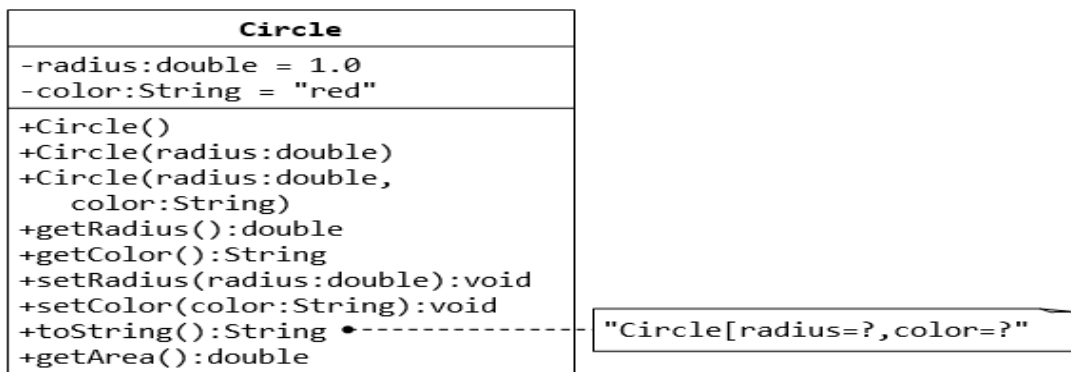
Lớp Circle là một “building block” và có nghĩa phải được sử dụng trong một lớp khác có hàm main(). Ví dụ lớp TestCircle có hàm main sử dụng lớp Circle.

Mã nguồn của TestCircle là:

```
public class TestCircle {
    public static void main(String[] args) {
        // Declare an instance of Circle class called c1.
        // Construct the instance c1 by invoking the "default" constructor
        // which sets its radius and color to their default value.
        Circle c1 = new Circle();
        // Invoke public methods on instance c1, via dot operator.
        System.out.println("The circle has radius of "
            + c1.getRadius() + " and area of " + c1.getArea());

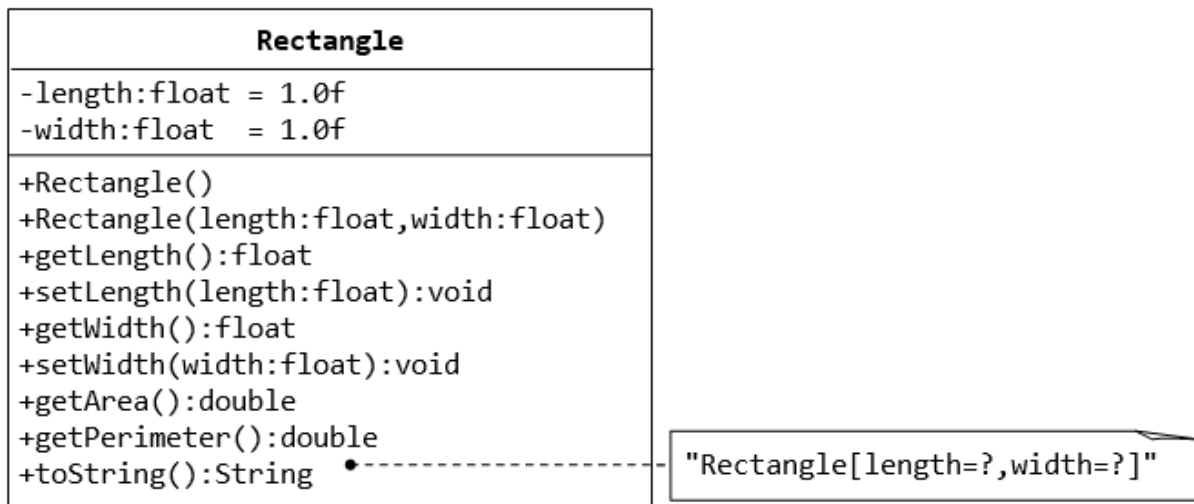
        // Declare an instance of class circle called c2.
        // Construct the instance c2 by invoking the second constructor
        // with the given radius and default color.
        Circle c2 = new Circle(2.0);
        // Invoke public methods on instance c2, via dot operator.
        System.out.println("The circle has radius of "
            + c2.getRadius() + " and area of " + c2.getArea());
    }
}
```

Bài tập (+0.25): Hãy viết một lớp Circle và lớp TestCircle trong Class Diagram sau:



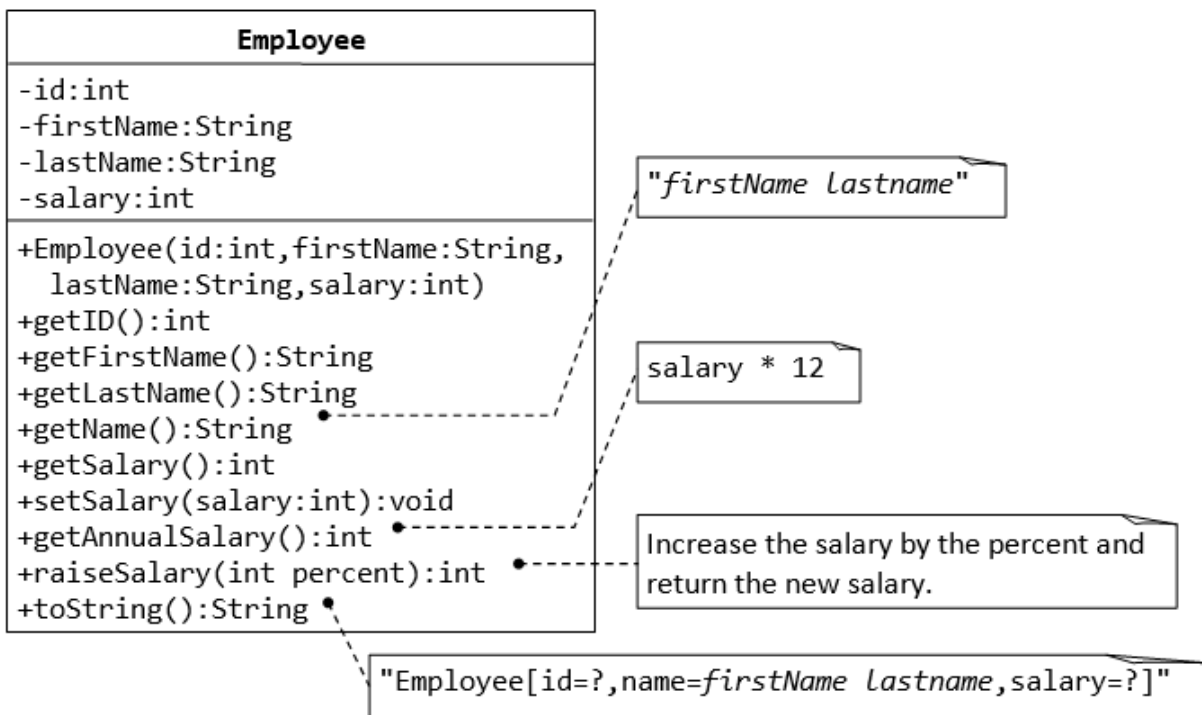
Bài 2:

Bài tập (+0.25): Hãy viết một lớp Rectangle và lớp TestRectangle trong Class Diagram sau:



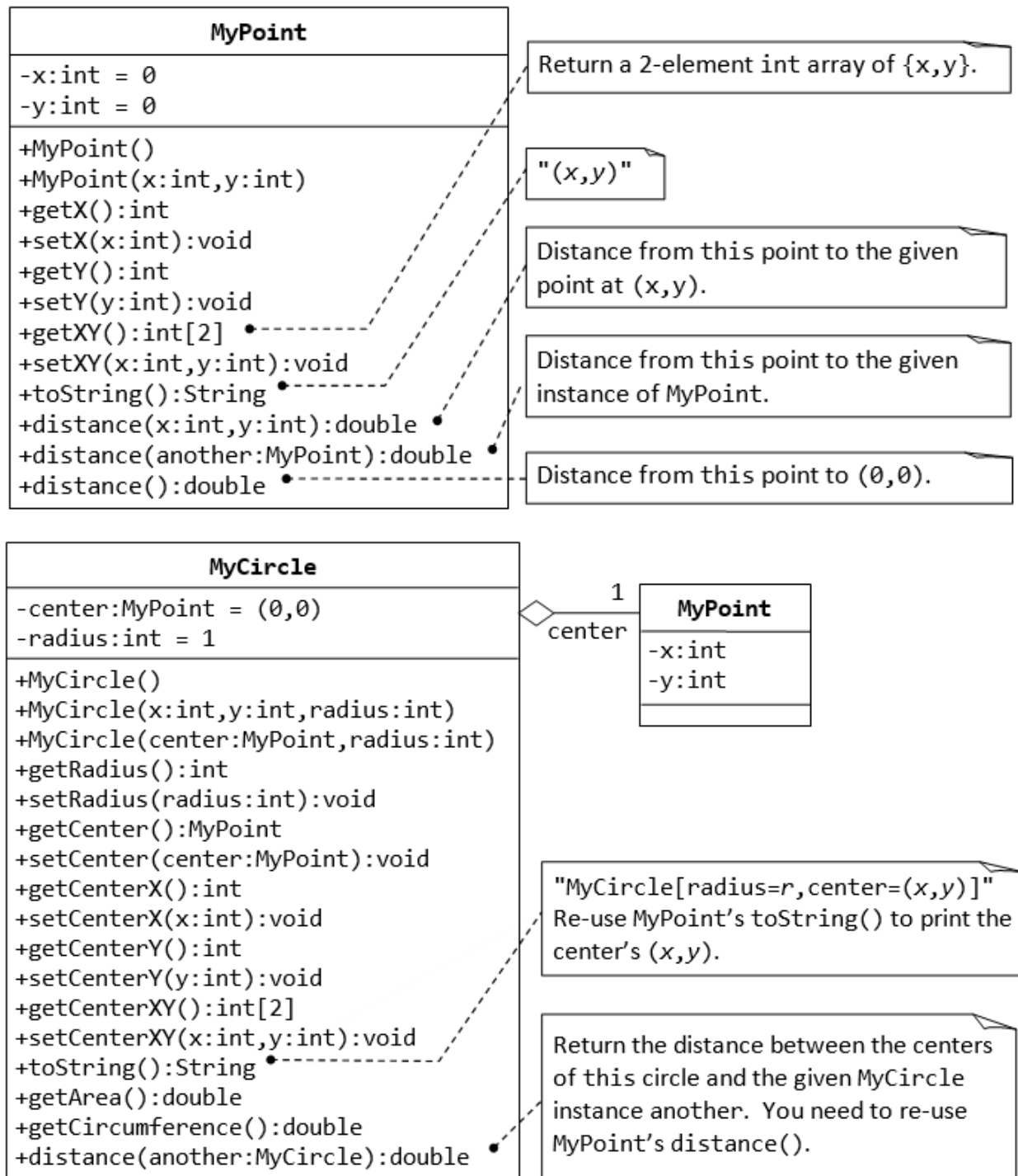
Bài 3:

Bài tập (+0.25): Hãy viết một lớp Employee và lớp TestEmployee trong Class Diagram sau



Bài 4:

Bài tập (+0.25): Ta có 2 lớp MyPoint và class MyCircle như sau



Hãy viết lớp MyCircle và MyPoint, đồng thời viết lớp TestMyCircle.