

Tutorial of Composition and static field

Based on the tutorial of "2020S-Java-A" designed by teaching group in SUSTech

Designed by ZHU Yueming in 2024. April. 8th

Objective

- Learn to static class members

Part 1: Static class member

Question Description:

Continue to use the `Circle` class we used before.

(1) Adding a new field `id`, which represents the created order of the current `Circle` object. For example, if one circle is the first created, its `id` is 1, and if one circle is the second created, its `id` is 2.

(2) The instance method `toString()` returns a `String` which include all the data field value as desired format.

```
Circle #[id]: radius = [radius], x = [x], y = [y]
```

The sample main method would be:

```
public static void main(String[] args) {
    Random random = new Random();
    Circle[] circles = new Circle[random.nextInt(3) + 3];
    for (int i = 0; i < circles.length; i++) {
        double radius = random.nextDouble() * 2 + 1;
        double x = random.nextDouble() * 10 - 5;
        double y = random.nextDouble() * 10 - 5;
        circles[i] = new Circle(radius, x, y);
    }
    for (Circle c: circles) {
        System.out.println(c);
    }
}
```

The output would be:

```
Circle #1: radius = 2.33, x = 3.77, y = 0.31
Circle #2: radius = 2.62, x = 0.43, y = 2.41
Circle #3: radius = 2.94, x = -0.04, y = -2.07
Circle #4: radius = 2.38, x = 3.13, y = 2.42
Circle #5: radius = 2.49, x = 1.70, y = 3.22
```

1. Add static class number

Add a static class number `count`, which records how many circle has been created.

```
private static int count = 0;
```

Add a privated data field `id`, which represents the id of current object.

```
private int id;
```

2. Modify Constructor:

When creating a circle object, increasing the count, and giving its value to id.

```
public Circle() {
    this.id = ++count;
}

public Circle(double radius, double x, double y) {
    this.id = ++count;
    this.radius = radius;
    this.x = x;
    this.y = y;
}
```

3. Modify toString Method:

```
public String toString() {
    return String.format("Circle #%d: radius = %.2f, x = %.2f, y = %.2f", id,
radius, x, y);
}
```

4. How to visit the value of static field?

We know if we want to visit a private member field, we can use getter and setter method. If the field is a static, how can we design the getter and setter method?

Design:

```
public static int getCount() {  
    return count;  
}  
  
public static void setCount(int count) {  
    Circle.count = count;  
}
```

Invoke:

```
System.out.printf("There are %d Circles:\n",Circle.getCount());
```