

# Objects, memory models, and scope

TOTAL POINTS 5

1. Consider the following code:

1 point

```
1 public class MyClass
2 {
3     private int a;
4     public double b;
5
6     public MyClass(int first, double second)
7     {
8         this.a = first;
9         this.b = second;
10    }
11    public static void main(String[] args)
12    {
13        MyClass c1 = new MyClass(10, 20.5);
14        MyClass c2 = new MyClass(10, 31.5);
15        System.out.println(c1.a + ", " + c1.b);
16    }
17 }
```

What would this code print?

Note that this might seem like a compiler error to be accessing c1.a and c1.b from main, but because the method main is defined in the class "MyClass", it will compile just fine (feel free to check it for yourself).

- ☒ 10, 20.5
- ☐ 10, 31.5

2.

1 point

```
1 public class MyClass
2 {
3     private int a;
4     public double b;
5
6     public MyClass(int first, double second)
7     {
8         this.a = first;
9         this.b = second;
10    }
11    public static void main(String[] args)
12    {
13        MyClass c1 = new MyClass(10, 20.5);
14        MyClass c2 = new MyClass(10, 31.5);
15        // lines below are changed from the question above
16        c2 = c1;
17        c1.a = 2;
18        System.out.println(c2.a);
19    }
20 }
```

What is the result of running the code above? Hint - draw a memory model!

- ☒ 2
- ☐ 10

3. In the code from question 2 above, after executing c2 = c1, how can you get back the object which c2 previously pointed to (the one with a as 10 and b as 31.5)?

1 point

- ☐ c2--;
- ☐ c2.restore();
- ☒ You cannot get that object back

4. Please review the code below:

1 point

```
1 public class MyClass
2 {
3     private int a;
4     public double b;
5
6     public MyClass(int first, double second)
7     {
8         this.a = first;
9         this.b = second;
10    }
11
12    // new method
13    public static void incrementBoth(MyClass c1) {
14        c1.a = c1.a + 1;
15        c1.b = c1.b + 1.0;
16    }
17
18    public static void main(String[] args)
19    {
20        MyClass c1 = new MyClass(10, 20.5);
21        MyClass c2 = new MyClass(10, 31.5);
22        // different code below
23        incrementBoth(c2);
24        System.out.println(c1.a + ", " + c2.a);
25    }
26 }
```

What would be the output from running main?

- ☒ 10, 11
- ☐ 11, 11
- ☐ 10, 10
- ☐ 11, 10

5. Please review the code below:

1 point

```
1 public class MyClass
2 {
3     private int a;
4     public double b;
5
6     public MyClass(int first, double second)
7     {
8         this.a = first;
9         this.b = second;
10    }
11
12    public static void incrementBoth(MyClass c1) {
13        c1.a = c1.a + 1;
14        c1.b = c1.b + 1.0;
15    }
16
17    // new method
18    public static void incrementA(int first)
19    {
20        first = first + 1;
21    }
22
23    // new method
24    public static void incrementB(double second)
25    {
26        second = second + 1.0;
27    }
28
29    public static void main(String[] args)
30    {
31        MyClass c1 = new MyClass(10, 20.5);
32        MyClass c2 = new MyClass(10, 31.5);
33        // different code below
34        incrementA(c2.a);
35        incrementB(c2.b);
36        System.out.println(c2.a + ", " + c2.b);
37    }
38 }
```

What is the output from running this code?

- ☐ 11, 32.5
- ☒ 10, 31.5