



مسئله‌ی ۱. سوال‌های تشریحی [۳۰]

به هر یک از سوال‌های زیر در چند خط پاسخ دهید.

۱. در جاوا چه سطوح دسترسی وجود دارد (توابع یا متغیرهای کلاس)؟ آن‌ها را نام برده و میزان دسترسی در هر کدام را توضیح دهید. [۸]

۲. فرق بین Stack و Heap چیست و چه تفاوتی دارند؟ داده‌های برنامه در چه حالتی در کدام نوع از حافظه ذخیره می‌گردند؟ [۸]

۳. Static method و Default method را توضیح دهید (در interface به آن اشاره شد) و چرا به زبان جاوا اضافه شدند؟ [۸]

۴. final در جاوا چه کاربردهایی دارد؟ [۶]

مسئله‌ی ۲. خروجی برنامه [۷۰]

به ازای هر یک از برنامه‌ها، خروجی خواسته شده را بنویسید.

۱. کدام یک از خطوط زیر خطای کامپایل دارد؟ [۸]

```
public class TypeCasting {  
    public static void main(String[] args){  
        int intValue = 2;  
        Integer classIntValue = 2;  
        long longValue = 12;  
        long secondLongValue = 12L;  
        Long classLongVlaue = 13L;  
  
        float floatValue = 0.12f;  
        boolean booleanValue = true;  
        char charValue = 'd';  
  
        intValue = secondLongValue;  
        secondLongValue = intValue;  
        intValue = (int)secondLongValue;  
  
        floatValue = intValue;  
        intValue = floatValue;  
        classLongValue = classIntValue;  
    }  
}
```

```

        intValue = booleanValue;
        floatValue = secondLongValue;

        floatValue = charValue;
    }
}

```

۲. در خروجی استاندارد چه چیزی چاپ می‌شود؟ [۱۲]

```

public class StringOperation {
    public static void main(String[] args){
        String input = "Hello World! How Are You???? (1397)";
        String backup = input;
        System.out.println(input.replace("a", "b"));
        System.out.println(backup);

        input = input.replace("!", "?");
        System.out.println(input);
        System.out.println(backup);

        System.out.println(input.split(" ")[1]);
        System.out.println(input.split(" ")[2]);

        System.out.println(input.indexOf('l') + input.indexOf('Y'));
        System.out.println(input.replaceAll("[a-zA-Z]+", "c"));

        System.out.println(input.replaceFirst("[\\d]+", "Y"));
        System.out.println(input.replaceFirst("\\d", "Y"));
    }
}

```

۳. در خروجی استاندارد چه چیزی چاپ می‌شود؟ [۸]

```

public class ParameterPassing {
    public static void main(String[] args){
        Integer age = 10;
        int oldAge = 70;
        Person first = new Person(age);
        age += 10;
        Person second = new Person(age);
        Person third = new Person(oldAge);
        oldAge += 10;
        Person forth = new Person(oldAge);

        Person[] persons = {first, second, third, forth};
        operation(age, oldAge, persons);

        for(Person person: persons){
            person = new Person(100);
        }
    }
}

```

```

    }

    for(Person person: persons){
        System.out.println(person.age);
    }

    for(Person person: persons){
        if(person != null){
            System.out.println(person.age);
        }else{
            System.out.println(person);
        }
    }
}

}

public static void operation(Integer age, Integer oldAge, Person[] persons){
    age += 10;
    oldAge += 10;
    persons[0].age += 15;
    persons = new Person[] {null, persons[0], persons[1], persons[2]};
    persons[2].age += 15;
}

}

class Person {
    int age = 0;

    public Person(int age){
        this.age = age;
    }
}
}

```

۴. در خروجی استاندارد چه چیزی چاپ می‌شود؟ (اگر خطوطی دارای خطای کامپایل دارد، آن را مشخص و برای اجرا حذف کنید.) [۱۲]

```

public class Override {
    public static void main(String[] args){
        C c = new C();
        D d = new D();
        C e = d;

        Integer i = 10;
        Object j = i;

        c.a(i);
        c.b(i);
        c.b(j);
        c.c(i);
    }
}

```

```

        d.a(i);
        d.b(i);
        d.b(j);
        d.c(i);
        ((C)d).c(i);

        e.a(i);
        e.b(i);
        e.b(j);
        e.c(i);
    }
}

class C {
    public void a(Integer num){
        System.out.println("C(a)");
    }

    public void b(Object num){
        System.out.println("C(b)");
    }

    private void c(Integer num){
        System.out.println("C(c)");
    }
}

class D extends C{
    public void a(Integer num){
        System.out.println("D(a)");
    }

    public void b(Integer num){
        System.out.println("D(b)");
    }

    public void c(Integer num){
        System.out.println("D(c)");
    }
}

```

۵. در خروجی استاندارد چه چیزی چاپ می‌شود؟ [۱۸]

```

public class Initialization {
    public static void main(String[] args){
        new B(1397);
        new A(1369);
    }
}

```

```

class A {
    public static int a = log(6);
    private int c = log(2);

    static {
        a = log(5);
    }

    {
        a = log(a + 1);
        c = log(9);
    }

    public A(int num){
        c = log(num);
    }

    public static int log(int num){
        System.out.println(num);
        return num;
    }
}

class B extends A {
    public static int b = log(3);
    private int d = log(4);

    static {
        b = log(3);
        a = log(10);
    }

    {
        a = log (a + 1);
        b = log (b * b);
        d = log (100);
    }

    public B(int num){
        super(num + 1);
        d = log(num - 1);
    }
}

```

۶. در خروجی استاندارد چه چیزی چاپ می‌شود؟ [۱۲]

```

public class Fibbo
{
    public static void main(String[] args)
    {
        FibCalculator calculator = new SubFibCalculator();
        calcFib(calculator);
    }

    private static void calcFib(FibCalculator calculator)
    {
        System.out.println("Super");
        calculator.fib(3);
    }

    private static void calcFib(SubFibCalculator calculator)
    {
        System.out.println("Sub");
        calculator.fib(3);
    }

    private static class FibCalculator
    {
        public int fib(int n)
        {
            System.out.println("Super");
            if (n == 1 || n == 0)
                return 1;
            return fib(n - 1) + fib(n - 2);
        }
    }

    private static class SubFibCalculator extends FibCalculator
    {
        public int fib(int n)
        {
            System.out.println("Sub");
            if (n == 1 || n == 0)
                return 1;
            if (n % 2 == 0)
                return super.fib(n);
            else
                return fib(n - 1) + fib(n - 2);
        }
    }
}

```