章节练习题及面试题

java 语言的特点是什么?

>面向对象性：两个基本概念：类、对象；三大特性：封装、继承、多态
>健壮性：吸收了C/C++语言的优点，但去掉了其影响程序健壮性的部分（如指针、内存的申请与释放等），提供了一个相对安全的内存管理和访问机制
>跨平台性：通过Java语言编写的应用程序在不同的系统平台上都可以运行。“Write once , Run Anywhere”

问题：System.out.println()和 System.out.print()什么区别呢?

以下代码的运行效果是什么?

System.out.println();打印完后，会换行。

System.out.print();打印完后，不会换行。

一个".java"源文件中是否可以包括多个类（不是内部类）？有什么限制?

答：可以。但最多只有一个类名声明为 public，与文件名相同。

Something 类的文件名叫 OtherThing.java

```
class Something {
    public static void main(String[] something_to_do) {
        System.out.println("Do something ...");
    }
}
```

这个很明显。
答案：正确。从来没有人说过 Java 的 class 名字必须和其文件名相同。但 public class 的名字必须和文件名相同。

为什么要设置 path（或者说，设置 path 的目的是什么）？

目的是为了在控制台的任何文件路径下，都可以调用jdk指定目录下的所有指令。

JDK,JRE 和 JVM 的关系是什么?

JDK包含JRE，JRE包含JVM.

源文件名是否必须与类名相同？如果不是，那么什么情况下，必须相同?

程序中若只有一个 public 修饰的类，且此类含 main 方法。那么类名与源文件名可否不一致?

Java 的注释方式有哪几种，格式为何?

自己使用 java 文档注释的方式编写程序，并用 javadoc 命令解析

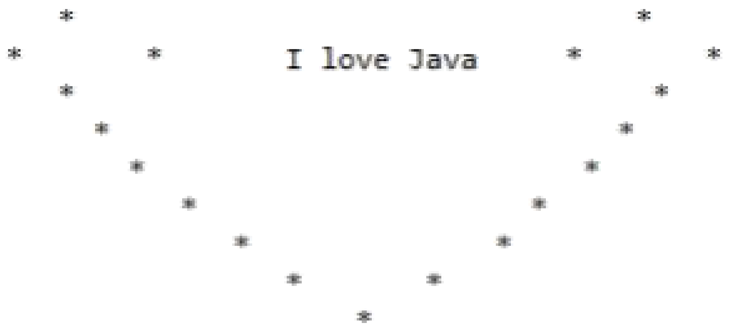
超纲题目：GC 是什么？为什么要有 GC

答：GC 是垃圾收集的意思（Gabage Collection）,内存处理是编程人员容易出现问题的地方，
忘记或者错误的内存回收会导致程序或系统的不稳定甚至崩溃，Java 提供的 GC 功能可以自动监测对象是否超过作用域从而达到自动回收内存的目的，Java 语言没有提供释放已分配内存的显示操作方法。

超纲题目：垃圾回收器的基本原理是什么？垃圾回收器可以马上回收内存吗？有什么办法主动通知虚拟机进行垃圾回收

答：对于 GC 来说，当程序员创建对象时，GC 就开始监控这个对象的地址、大小以及使用情况。通常，GC 采用有向图的方式记录和管理堆(heap)中的所有对象。通过这种方式确定哪些对象是"可达的"，哪些对象是"不可达的"。
当 GC 确定一些对象为"不可达"时，GC 就有责任回收这些内存空间。可以。程序员可以手动执行 System.gc()，通知 GC 运行，但是 Java 语言规范并不保证 GC 一定会执行。

输出：心形



```
   *                                *
*       *       I love Java     *       *
   *                                 *
     *                             *
       *                         *
          *                   *
             *              *
                *       *
                    *
```

方式一：

```
class PrintHeart {

    public static void main(String[] args) {



        System.out.print("\t");

        System.out.print("*");

        System.out.print("\t");

        System.out.print("\t");

        System.out.print("\t");

        System.out.print("\t");

        System.out.print("\t");

        System.out.print("\t");

        System.out.print("\t");



        System.out.println("*");
        System.out.print("*");
        System.out.print("\t");
        System.out.print("\t");
        System.out.print("\t");
        System.out.print("\t");
        System.out.print("I love java");
        System.out.print("\t");
        System.out.print("\t");
        System.out.print("\t");
        System.out.print("\t");

        System.out.println("*");


        System.out.print("\t");
        System.out.print("*");
        System.out.print("\t");
        System.out.print("\t");
        System.out.print("\t");
        System.out.print("\t");
        System.out.print("\t");
        System.out.print("\t");

        System.out.print("\t");



        System.out.println("*");



        System.out.print("\t");

        System.out.print("\t");

        System.out.print("*");

        System.out.print("\t");

        System.out.print("\t");

        System.out.print("\t");

        System.out.print("\t");

        System.out.print("\t");



        System.out.println("*");



        System.out.print("\t");

        System.out.print("\t");

        System.out.print("\t");

        System.out.print("*");

        System.out.print("\t");
        System.out.print("\t");
        System.out.print("\t");

        System.out.println("*");

        System.out.print("\t");
        System.out.print("\t");
        System.out.print("\t");
        System.out.print("\t");
        System.out.print("*");
        System.out.print("\t");

        System.out.println("*");

        System.out.print("\t");
        System.out.print("\t");
        System.out.print("\t");
        System.out.print("\t");
        System.out.print("      ");
        System.out.print("*");
```

```
    }

}
```

方式二：

```
class PrintHeart1 {

    public static void main(String[] args) {


        System.out.print("\t" + "*" + "\t\t\t\t\t\t\t\t\t\t\t\t" + "*" + "\t" + "\n");

        System.out.print("*" + "\t\t" + "*" + "\t\t\t\t" + "I love Java" + "\t\t\t\t\t" + "*" + "\t\t" + "*" + "\n");

        System.out.print("\t" + "*" + "\t\t\t\t\t\t\t\t\t\t\t\t" + "*" + "\t" + "\n");

        System.out.print("\t\t" + "*" + "\t\t\t\t\t\t\t\t\t\t" + "*" + "\t\t" + "\n");

        System.out.print("\t\t\t" + "*" + "\t\t\t\t\t\t\t\t" + "*" + "\t" + "\n");

        System.out.print("\t\t\t\t" + "*" + "\t\t\t\t\t\t" + "*" + "" + "\t" + "\n");

        System.out.print("\t\t\t\t\t" + "*" + "\t\t\t\t" + "*" + "" + "\t\t" + "\n");

        System.out.print("\t\t\t\t\t\t" + "*" + "\t\t" + "*" + "" + "\t\t" + "\n");

        System.out.print("\t\t\t\t\t\t\t" + "*" + "\n");


    }


}
```