1. Java开发的前提条件就是JDK的安装，然后配置环境变量，接下来就是Coding，当然开发的工具也很多。在"系统变量"中设置3项属性，JAVA\_HOME,PATH,CLASSPATH(大小写无所谓)：

说明：

JAVA\_HOME：软件靠这个变量找到jdk的安装目录，如果不设这个变量的话，这些软件包很可能找不到javac.exe在哪里。比如ant就需要通过这个环境变量来确定javac.exe的路径来执行compile task。这个环境变量的值就是你的jdk安装目录。如果你的jdk装在c:/jdk1.42，那么“c:/jdk1.42”就是JAVA\_HOME的值。

CLASSPATH：类路径，告诉 Java 应用程序去哪里查找第三方和自定义类 -- 即不是 Java 扩展或 Java 平台一部分的类。在 JDK 1.2 中，JVM 和其他 JDK 工具通过依次搜索平台库、库扩展和类路径来查找类（有关搜索策略的详细信息，参见如何查找类）。 多数应用程序的类库都会充分利用扩展机制。因此，只有想要加载某个类库 (a) 未位于当前目录或其分枝包中 且 (b) 未位于扩展机制所指定的位置时，才需设置类路径。在 CLASSPATH 列表中除了可以包含路径外，还可以包含 .jar 文件。Java 查找类时会把这个 .jar 文件当作一个目录来进行查找。通常，我们需要把 JDK 安装路径下的 jre/lib/rt.jar (Linux: jre/lib/rt.jar) 包含在 CLASSPATH 中。

PATH：指定一个路径列表，用于搜索可执行文件的。执行一个可执行文件时，如果该文件不能在当前路径下找到，则依次寻找 PATH 中的每一个路径，直至找到。或者找完 PATH 中的路径也不能找到，则报错。Java 的编译命令 (javac)，执行命令 (java) 和一些工具命令 (javadoc, jdb 等) 都在其安装路径下的 bin 目录中。因此我们应该将该路径添加到 PATH 变量中。

PATH 和 CLASSPATH 都指定路径列表，列表中的各项 (即各个路径) 之间使用分隔符分隔。在 Windows 下，分隔符是分号 (;)，而在 Linux 下，分隔符是冒号 (:)。

1、JAVA\_HOME:指名JDK的位置。

2、CLASSPATH:指名到哪里去找运行时需要用到的类代码（字节码）

3、PATH:指名可执行程序的位置。

2. 从面向过程编程到面向对象编程的思维转变

我们知道所有的计算机程序都是由两类元素组成：代码和数据。此外从概念上将讲，程序还可以以他的代码或是数据为核心进行组织编写。也就是说一些程序围绕“正在发生什么编写“，而另一些程序则围绕”谁将被影响“编写。这两种范型决定程序的构建方法。第一种方法被称为面向过程的模型，用他编写的程序都具有线性执行的特点。面向过程的模型可以认为是代码作用于数据，用C写的程序就是典型的面向过程模型。第二种方法也就是我们现在正在学习的面向对象编程，面向对象编程围绕她的数据（即对象）和为这个数据严格定义的接口来组织程序。面向对象的程序实际上就是用数据控制对代码的访问。Java就是一门纯面向对象编程的语言。

3.什么是字节码和虚拟机：

字节码是一套设计用来在Java运行时系统下执行的高度优化的指令集。该Java运行时系统称为Java虚拟机(JVM)。JVM其实就是一个字节码解释器。虚拟机将字节码解释成机器码给CPU执行，所以我们在java中通过虚拟机的这种解释执行方式来屏蔽底层操作系统的差异。

JRE = JVM+编译器

JDK= JVM+编译器+类库

4.

标识符是给变量、类、方法命名的符号，必须以字母、下划线、$开头，后面可以跟任意的字母、数字、下划线、$.所有的关键字都是小写

基本类型和引用类型

基本类型有byte,int,short,long, char,boolean, float,double;;默认浮点类型是double,,float型后面必须加f

引用类型包括类，接口和数组;引用类型变量就是一个指针。