**Java语言的重要特点**

1. **Java语言是面向对象的oop**
2. **Java语言是健壮的。Java的强类型机制、异常处理、垃圾的自动收集等是Java程序健壮性的重要保证。**
3. **Java语言是跨平台性的**
4. **Java语言是解释型的**

**解释性语言：JavaScript、PHP、java 编译性语言：c/c++**

**区别是：解释性语言，编译后的代码，不能直接被机器执行，需要解释器来执行，编译性语言，可以直接被机器执行，c/c++**

**Java运行机制及运行过程**

**Java核心机制：java虚拟机【JVM】**

1. **JVM是一个虚拟的计算机，具有指令集并使用不同的存储区域。负责执行指令，管理数据、内存、寄存器，包含在JDK中。**
2. **对于不同的平台，有不同的虚拟机。**
3. **Java虚拟机机制屏蔽了底层运行平台的差别，实现了“一次编译，到处运行”**

**Jdk基本介绍**

1. **jdk的全称 Java development kit java开发工具包**

**jdk = jre+Java开发工具【Java、javac、javadoc、Javap等】**

1. **jdk是提供给Java开发人员使用的，其中包含了java的开发工具，也包括了jre。**

**Jre基本介绍**

1. **jre Java runtime environment Java运行环境**

**jre = jvm +Java的核心库【类】**

1. **包括java虚拟机和java程序所需的核心库等，如果想运行一个开发好的Java程序，计算机中只需要安装jre即可。**

**Jdk、jre和JVM的包含关系**

1. **JDK = JRE + 开发工具集（例如javac、java编译工具等）**
2. **JRE = JVM + Java SE标准类库（java核心类库）**
3. **JDK = JVM + Java SE标准类库 + 开发工具集**
4. **如果想要运行开发好的.class文件 只需要JRE**

**什么是运行**

1. **有了可执行的java程序（hello.class）**
2. **通过运行工具java.exe 对字节码文件进行执行，本质就是.class 装载到jvm机执行**

**Java程序开发注意事项**

**对修改后的Hello.java源文件需要重新编译，生成新的class文件后，再进行执行，才能生效。**

**Java开发注意事项和细节说明**

1. **Java源程序以.java为扩展名。源文件的基本组成部分是类（class），如本类中的hello类。**
2. **Java应用程序的执行入口是main( )方法。它有固定的书写编写方式。**
3. **Java语言严格区分大小写**
4. **Java方法由一条条语句构成，每个语句以“；”结束。**
5. **大括号都是成对出现的，缺一不可 [习惯，先写{} 再写代码]**
6. **一个源文件中最多只能有一个public类。其它类的个数不限。**
7. **如果源文件中包含一个public类，则文件名必须按改类名命名！**
8. **一个源文件中最能只能有一个public类。其它类的个数不限，也可以将main方法写在非public类中，然后指定运行非public类，这样入口方法就是非public 的 main 方法。**

**Java转义字符**

**1、\t 一个制表位，实现对齐的功能**

**2、\n 换行符**

**3、\\ 一个\**

**4、\” 一个”**

**5、\’ 一个’**

**6、\r 一个回车 System.out.println(“Angelczack\r上海”)；**

**初学Java易犯错误**

1. **找不到文件**

**解决方法：源文件名不存在或者写错，或者当前路径错误**

1. **主类名和文件名不一致**

**解决方法：声明为public的主类应与文件名一致，否则编译失败**

1. **缺少分号**

**解决方法：编译失败，注意错误出现的行数，再到源代码中指定位置改错。**

**注释（comment）**

**介绍：**

**用于注解说明结束程序的文字就是注释，注释提高了代码的阅读性；注释是一个程序员必须具有的良好编程习惯。将自己的思想通过注释先整理出来，再用代码去体现。**

**Java中的注释类型**

**1、单行注释**

**基本格式 //注释文字**

**2、多行注释**

**基本格式 /\*注释文字\*/**

**3、文档注释**

**注释内容可以被JDK提供的工具javadoc所解析，生成一套以网页为文件形式体现的该程序的说明文档，一般写在类。**

**基本格式 Javadoc -d 文件夹名（目录） -xx -yy Demo.java**

**Javadoc 标签**

**@author 标识一个类的作者**

**@version 指定类的版本**

**@return 说明返回的类型**

**@param 说明一个方法的参数**

**@see 说明一个到另一个主题的链接**

**使用细节**

**1）被注释的文字，不会被JVM解释执行**

**2）多行注释里面不允许有多行注释嵌套**

**Java代码规范**

**1、类、方法的注释，要以javadoc的当时来写**

**2、非Java Doc的数值，往往是给代码的维护者看的，着重告诉读者为什么这样写，如何修改，注意什么问题等**

**3、使用tab操作，实现缩进，默认整体向右边移动，时候用shift+tab整体向左移**

**4、运算符和 = 两边习惯性各加一个空格。 比如：2 + 4 \* 5 + 345 -89**

**5、源文件使用utf-8编码**

**6、行宽度不要超过80字符**

**7、代码编写次行风格和行尾风格**

**DOS命令**

**DOS介绍：Disk Operating System 磁盘操作系统，简单说一下windows的目录结构**

**常用的dos命令**

**1、查看当前的目录有什么**

**dir dir d:\abc2\test200**

**2、切换到其他盘下**

**Cd /D c:**

**3、切换到当前盘的其他目录下(相对路径和绝对路径的展示方法)**

**1) cd d:\abc2\test200 cd ..\..\abc2\test200**

**4、切换到上一级**

**cd ..**

**5、切换到根目录**

**cd \**

**变量原理**

**为什么需要变量：变量是程序的基本组成单位**

**变量的介绍**

**变量相当于内存中一个数据存储空间的表示，你可以把变量看作是一个房间的门牌号，通过门牌号我们可以找到房间，而通过变量名可以访问到变量值**

**变量使用的基本步骤**

**1）声明变量**

**int a；**

**2) 赋值**

**a = 60; //应该这么说：把60赋给a**

**3) 使用System.out.println(a);**

**//也可以一步到位[int a =60; 通常我们是一步完成]**

**变量使用注意事项**

**1、变量表示内存中的一个存储区域 [ 不同的变量，类型不同，占用的空间大小不同，比如：int 4个字节，double 就是 8个字节]**

**2、该区域有自己的名称 [变量名] 和类型 [数据类型]**

**3、变量必须先声明，后使用，即有顺序**

**4、该区域的数据可以在同一类型范围内不断变化**

**5、变量在同一个作用域内不能重名**

**6、变量 = 变量名 + 值 + 数据类型，这是变量的三要素**

**Java程序中 +号的使用**

**1、当左右两边都是数值型时，则做加法运算**

**2、当左右两边有一方为字符串，则做拼接运算**

**数据类型**

**每一种数据都定义了明确的数据类型，在内存中分配了不同大小的内存空间(字节)**

**一、基本数据类型**

**1、数值型**

**1）整数类型，存放整数(byte[1],short[2],int[4],long[8])**

**2）浮点（小数）类型(float[4],double[8])**

**2、字符型(char[2]),存放单个字符 ‘a’**

**3、布尔型(boolean[1]),存放true，false**

**二、引用数据类型**

**1、类 class**

**2、接口 interface**

**3、数组 [ ]**