Java中关键字介绍

关键字：public，private，final ，implements，package，import

1. 关键字 final ：可以修饰方法、属性、类。
2. final修饰一个类时，表示该类是一个终态类，不能被继承；
3. final修饰一个方法时，表示该方法是个终态方法，不能被重写（override）；
4. final修饰一个属性时，表示该属性不可以被更改了。①当final修饰一个原生数据类型时，该原生数据类型的值不可以再改变；②当final修饰一个引用类型时，表示该引用不能再指向其他的对象了（不改变指向对象），但是所指向的对象的内容是可以改变的。
5. final 陷阱：对于 final类型的成员变量，一般情况下有两种赋初值的方法（虽然成员变量有一个默认的值，但是必须显式地赋予初值才可以）：
6. 在声明final 成员变量时，就赋予初值；
7. 在声明 final 成员变量时，如果没有赋予初值，需要在此类中的所有方法中都赋予初值。
8. 静态代码块只在类加载的时候只会执行一次，因为类只会被加载一次，而构造方法在生成一个新对象时就会执行一次。注意：静态代码块只会执行一次，而构造方法可以执行多次。如果存在多级继承时，首先执行最顶层的静态代码块，一直执行到最底层的静态代码块，然后再执行最顶层的构造方法，一直执行到最底层的构造方法。
9. 不能在静态方法中调用非静态的成员变量。因为静态方法可以直接利用 类名.方法名 直接调用，但是静态方法中对属性的改变到底属于哪个对象的不知道，矛盾。可以在非静态的方法中访问静态成员变量。静态的可以访问静态的，非静态的可以访问一切。不能在静态方法中使用this 关键词。
10. 关键字 instanceof 作用：判断某个对象是否是某个类的实例。形式：引用名 instanceof 类名（或接口）；返回一个布尔值。
11. 包（package）：命名规则：域名倒着写；包名：都用小写字母。如 package com.shengsiyuan；注意分号不可以少。子包：如a.b.c 是a.b是的子包。

如果定义类的时候没有定义包，则就认为其类存在于默认的包里面。

1. 带有package的编译方法：利用命令 javac –d . Test.java 这样的话，编译器自动会建立起文件结构。
2. 导入(import)：（import的落脚点是类，如果整个包中的类都导入，落脚点是.\*）将使用package分离的各个类导入回来，让编译器能够找到所需要的类。

Import的用法：import a.com.test.Test; 最后Test是个类，不需要加上.class；

如果需要导入某个文件夹中的所有类，利用星号 \* 代替即可，如 import a.com.test.\*;表示导入a.com.test包中的所有的类。但是如果是import a.com.\*;并不能把 a.com.test子包的的类导入，这是需要两次导入; 如：

import a.com.\*;

import a.com.test.\*;

如果两个或者多个类在同一个包下面，就无需导入，直接使用即可。

1. 关于 package、import、class/interface 的顺序问题：
2. 首先，是package；
3. 其次是import；
4. 再是定义类或者接口。
5. public 和 private 的区别：相对于自身的类而言的，private只能在类的内部使用，在类的外部不可以调用。
6. 接上，JDK给了类System，由于其在java.lang包下的，所以不需要导入声明。
7. JDK给定的包和类，至与java.lang.\*里面的类不需要导入直接调用即可，至于不是java.lang.\*，其他的必须导入。在引用方法时，先判断此方法是否是静态（static）的，如果是，则直接用类名.方法名调用即可，否则必须先new出一个类的对象，然后利用对象.方法名进行调用。
8. 接口中的方法都是抽象的，接口中的成员变量都是public的，都是静态的，都是final的。所以public、static、final都可以省略不写。接口中的属性很少使用，知道即可。
9. 一个类不能即是final（说明该类不能被继承）又是abstract的（定义一种约定，让子类去实现约定，即用来被继承的）。矛盾，因此不可以。
10. **final 既可以修饰类、方法，也可以修饰变量（属性）。**
11. 修饰类时，表示终态类，不可以被继承；
12. 修饰方法时，表示终态方法，不可以被重写；
13. 修饰属性时，表示了修饰的是一个常量，只能被赋值一次，给常量起个名字，提高程序的可读性。

当final修饰原生数据类型时，表示此变量值不会发生变化，当final修饰引用类型时，表示此引用不可以再指向其他对象了，但是所指向的对象的内容是可以变化的。

1. 遇到一些固定不变的量，一定要起名字并用final修饰，名字全部大写，单词之间利用下划线连接，**否则不专业**。
2. final 修饰的属性只能固定显式初始化值，不能固定默认初始化值，否则会报错，因为固定一个默认初始化值没有意义。
3. 成员（包括属性和方法）一旦定义为final，通常都会定义为static，因为既然是final的，不可更改了，那么对于任何一个此类对象都一样了，直接利用类名访问就可以了。至于是public还是private等等，随需求而定。
4. 用 public static 修饰的变量叫作全局变量；利用public static final修饰的叫作全局常量。
5. **private：定义为private就是不让类外的其他程序进行直接访问，一般本类中会提供方法让其他程序进行调用，这样可以提高可控性。一般情况下都会把类中的属性定义为private的，起到了保护的作用，通过构造方法进行初始化，通过setXxx（）和getXxx()方法设置和获取变量的值。例子：**
6. **package：包。**

**（1）作用：对类文件进行分类管理；**

**（2）给类进行多层命名（名称）空间；**

**（3）写在程序文件的第一行；**

**（4）类名的全称是 包名.类名。**

**（5）包的命名规范是所有字母都要小写；**

**（6）包也会一种封装形式。**

1. **import：导入。（类）**
2. **一定要注意：导入的是包中的具体的类，平常所说的导包只是简称，本质是导入类；导包其实是导类。**
3. **通配符： \* ，可以代表某个包下面的所有类，如果需要导入另外包中的多个类，可以直接利用\*代替所有的类，但是注意：\*前面必须是完整的类名，否则导入不进来。如果某个包中既含有类又含有其他的包，此时\*只能代表类，其中的包（中的类）不能导入，必须把包名指定完整才可以，且\*这可以代表当前包下面的所有类。**
4. **在工程实际开发过程中，建议是用到一个类就导入一个类，尽量不要利用\*导入包下的所有类。**
5. **jar : java 的压缩包。利用jar命令可以创建jar文件。**