

package 和 import 的使用

1. 包 (package)

包名一般为小写，而类名的第一个字母一般为大写，这样在引用时，可以明显的分辨出包名和类名。如果在类的定义之前没有使用 `package` 定义包名，那么该类就属于缺省的包。

1.1 包的作用

可以更好的组织类，包与文件夹类似，文件夹可以将不同的文件放在同一个文件夹中，而包也可以将不同的类文件放在同一包中。

减少类名的冲突问题，这也与文件夹类似，同一文件夹中的文件不能重名，不同文件中的文件可以重名，同一包中的类名不能重复，不同包中的类名可以重复。

对包中的类起了一定的保护作用，详见 Java 的访问控制。

1.2 包的使用

使用 `package` 关键字指明类所在的包。

`package` 语句必须在文件的最前面。

编译时可使用 `javac -d . Test.java` 自动产生包需要目录。

可以使用 `java testPackage.Test` 来执行编译后的代码。

2. 导入 (import)

使用了包机制后，如果在一个类中使用了其他包中的类，需要使用 `import` 来引入。

2.1 import 的使用

直接引用指定的类，如 `import java.util.Vector`。

引用一个包中的多个类，如 `import java.awt.*`。更确切的说，它并不是引用 `java.awt` 中的所有类，而只引用定义为 `public` 的类，并且只引用被代码引用的类，所以这种引用方法并不会降低程序的性能。

号代替类名，在不能代替包名，如 `import java.awt.`，只引用 `java.awt` 下的类，而不引用 `java.awt` 下的包。

`import java.awt.F*`，这种使用方法是错误的。

`import` 语句在所有类定义之前，在 `package` 定义之后。

`import` 只告诉编译器及解释器哪里可以找到类、变量、方法的定义，而并没有将这些定义引入代码中。

2.2包中类的使用

如果要使用的类是属于 `java.lang` 包的，那么可以直接使用类名来引用指定的类，而不需要加上包名，因为包 `java.lang` 不用显示使用 `import`，它是缺省引入的。

如果要使用的类在其它包（`java.lang` 除外）中，那么可以通过包名加上类名来引用该类，如 `java.awt.Font`。

对于经常要使用的类（该类在其它包中），那么最好使用 `import` 引用指定的包，如 `java.awt.*`。

如果 `import` 引入的不同的包中包含有相同的类名，那么这些类的使用必须加上包名。

接口也可以属于某个包，也可以使用 `import` 引入其它包中的类和接口。