

第7章 控制逻辑

上一章介绍了类和对象的概念，以及成员变量、局部变量和方法的概念。本章将介绍如何通过修饰符来控制变量的访问。首先介绍包的概念，后面将介绍各个控制权限的修饰符。

7.1 包 (package)

包就好比日常生活中的箱子，是一个存放东西的空间。在Java中，包的概念就好比Windows里的 **文件夹** 的概念，是一层一层的关系。实际开发中会使用到很多的类，使用方便高效的方法将这些类进行管理也具有比较重要的意义，Java中提供包将不同的类组织起来紧系管理。类按照其功能分别存放在各个包里。

7.1.1 创建一个包

使用包是为了更好地将代码进行管理，首先介绍如何创建一个包，如果想要将类存放指定的包中，必须要使用 `package` 语句，语法如下：

```
package 包名;
```

例如下面的程序语句。

```
package a.b;
```

7.1.2 如何使用包

当创建了一个包时就要引入一个包，引入一个包的关键字为 `import`，语法如下：

```
import 包名.*;
```

```
import 包名.类名;。
```

7.1.3 什么是静态引入

静态引入就是引入包中的静态成员变量和静态方法。静态引入的关键字为static，静态的其他内容将在后面章节进行讲解。需要将import和包名之间加入static，这样不单可以导入类，还可以导入静态方法和静态成员变量。静态引入的语法如下：

```
import static 包名.aaa.*;
```

```
import static 包名.aaa.方法名称;
```

7.2 类的访问级别

类的访问级别，就好比日常生活中常见的大树，要想到达树顶，要从树底下慢慢地爬上去。树底下的树枝能看到旁边的树枝，但看不到树顶的树枝。在Java中，类的访问也是有一种级别关系的。下面介绍类的访问级别和成员变量的访问级别。本节所提到的修饰符请读者先行了解，将在第8章对其含义做详细说明。

7.2.1 公开的访问级别

带有“public”修饰符的类被声明为公有的访问级别，即在类的名称前面带有public修饰符。用public修饰符修饰该类，表示在任何包中的任何类都能访问该类。但要注意不同包的问题。虽然可以被任何包中的任何类访问，但是在不同的包下还需要使用引用语句。下面代码演示如何用public修饰符修饰一个类。

//test类描述的是用修饰符修饰类

```
public class test
```

```
{
```

```
...//方法体
```

```
}
```

7.2.2 默认访问级别

具有默认访问级别的类在声明时前面不加任何的修饰符，可以将默认访问级别看作是包级的访问，只能被同一个包下的类使用。默认访问级别和公开的访问级别很相似，不同点就是默认访问级别不能访问不同包下的类。只能访问同包下的类。

7.3 什么是封装

封装就好比用一个盒子把一些东西装起来，让外界能看到这些东西。在Java中，封装就是在一个类里定义了一些成员变量和方法，通过限制其成员变量和方法的可见性，使得外界不能访问它们。因此封装展现了接口，隐藏了细节。

封装允许把成员变量标识为public，但是在实际的应用中最好把所有的变量都保持为private。本节所提到的修饰符请读者先行了解，将在第8章对其含义进行详细的说明。

7.4 最终修饰符

所谓最终修饰符，在字面上可以说为最终的，不变的意思。修饰符final可以修饰很多类型的数据，被其修饰的数据所具有的含义也各有不同。下面将分别介绍修饰成员变量、局部变量、方法，以及基本类型所具有的含义。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/857152146125006030>