**Java** **SE** 第九讲

**1.** 面向对象程序设计的三大基本特征： 继承（**Inheritence**）、封装（**Encapsulation**）、多态 （**Polymorphism**）

**2.** 封装： 类包含了数据与方法，将数据与方法放在一个类中就构成了封装。

**3.** 如何定义类？

修饰符 **class** 类的名字

**{**

**//**类的内容（包含了属性与方法）

**}**

**4.** 方法。如何定义方法？

修饰符 返回类型 方法名称（**[**参数 **1** ， 参数 **2** ， 参数 **3** **…]**）

**{**

**//**方法体

**}**

**5.** **main** 方法是整个**Java** 程序的入口点，如果类的定义中没有 **main** 方法，则程序无法执行。

**6.** 方法定义不能嵌套， 也就说不能在一个方法中定义另外一个方法。方法只能定义在类中。

**7.** 关于方法的执行： 首先需要定义方法，接下来就可以使用方法（调用方法），当方法调 用完毕后，方法可以返回值。 方法到底是否返回值是由方法的定义决定的。

**8.** 如何生成对象？通过类来生成对象（通常使用 **new** 关键字来生成对象）。

**public** **class** **Person**

**{**

**}**

类名 变量名 **=** **new** 类名**();**

**Person** **person** **=** **new** **Person();**

**Person** **person2** **=** **new** **Person();**

**Person** **person3** **=** **new** **Person();**

**9.** 方法调用需要通过对象来完成，方法调用的形式是：

对象变量**.**方法名**([**参数值 **1** ， 参数值 **2** ， 参数值 **3** **….]);**

**10.** 关于方法的注意事项：

**1**） 在方法定义中，方法的返回类型与 **return** 后面的变量或常量类型保持一致。

**2**） 在方法调用时， 给方法传递的参数需要与方法定义时的参数保持一致（参数个数一 致，参数类型一致）。

**3**） 方法定义时的返回类型与接收方法返回值的变量类型保持一致。

**11.** **public** **int** **add(int** **a,** **int** **b)** **{**

**return** **a** **+** **b;**

**}**

方法定义时的参数叫做形式参数。

**int** **a** **=** **test.add(8,** **3);**

方法调用时所赋予的具体值叫做实际参数。

**12.** 关键字 **void** 表示方法不返回值。

**13.** 如果方法不返回值， 那么声明方法的时候使用 **void** 关键字， 在方法定义中可以有两种 情况实现不返回值：

**a)** 不使用 **return** 语句。

**b)** 使用 **return**，但 **return** 后面没有任何值或者变量， **return** 后面只有一个分号， 表示

退出方法，返回到方法的调用端。

使用方式：

**return;**