

# Java 课程实验

## 1 整数累加器：JAccumulator

### 【实验目的】:

1. 了解 Java 程序设计的基本结构：如何编写程序主函数，以及主函数在 Java 中的位置与在 C/C++ 中的区别（在 Java 中，主函数放在类的内部，这体现了 Java 完全面向对象的思想）；
2. 掌握 Java 的基本语法：条件语句、循环语句、判断语句。

### 【实验要求】:

1. 设计一个 Java 程序，要求：能够接收从客户输入的任意一个正整数  $n$ ，并输出从 1 累加到  $n$  的结果。

### 【注意事项】:

1. 关于 Java 输入参数的方式请参考课本 p.52~p.56；
2. 请用 Eclipse 开发环境编写 Java 代码。

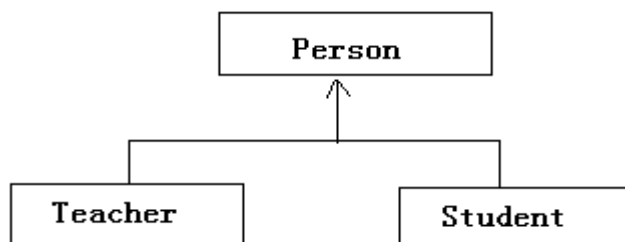
## 2 Java 类的继承和多态

### 【实验目的】:

1. 理解 Java 类的继承机制是如何实现的；
2. 理解 Java 类的动态多态机制是如何实现的。

### 【实验要求】:

1. 尽量少写相同的代码，编写 Java 程序，描述如下图所示的类层次，其中 **Person** 为父类，其属性包括姓名、性别、出生日期等，方法为 **printInfo()** 打印信息；**Teacher** 和 **Student** 为子类，**Teacher** 类还包括学校和工号属性；**Student** 类还包括学校、学号、专业、年级和班级等属性；编写一个测试类 **TestPerson**，在 **main** 方法中创建 1 名教师和 1 名学生对象，输出对象的所有属性信息。
2. 静态方法的多态：在 **Person** 类中，加入一个静态方法（任意的，没有限制），并在 **Teacher** 和 **Student** 类覆盖该静态方法，体会这种静态方法的多态有什么样的问题。



### 3 接口的定义与实现

**【实验目的】：**

1. 理解 Java 中的接口是如何定义和被其子类实现的；
2. 能够用 Java 语法描述简单的数据结构，如：队列。

**【实验要求】：**

1. 定义一个接口 `DataStructure`，包括以下方法：  
`Boolean isFull()`  
`Boolean isEmpty()`  
`void addElement()`  
`Object removeElement()`
2. 设计一个队列类（`MyQueue`）实现 `DataStructure` 接口。队列的大小由其构造方法指定。要求实现的方法体现出队列的先进先出特性。
3. 设计一个测试类，在其主要方法中用 `DataStructure` 类型的引用变量引用一个大小为 10 的 `MyQueue` 对象，使用 `addElement()` 方法增加 “0”、“1”、... “9” 共 10 个字符串对象，再用 `removeElement()` 方法取出这些元素并打印出来。