面向对象程序设计Java实验指导书

实验六 Java多线程

# 实验目的

* 理解Java多线程相关理论知识
* 掌握两种创建多线程类的方法
* 掌握Thread类中的主要方法，掌握线程的调度

# 实验要求

* 按照下述要求编写Java代码
* 按要求回答相关问题
* 按时完成并上交上机报告

# 实验内容

|  |  |
| --- | --- |
|  | 在Eclipse中创建一个Java项目，项目命名方式为：学号+实验编号，例如：学号为2015121001的同学的实验三项目命名为：2014121001exp6 |

## 多线程编程练习

在你的项目中定义一个package，命名为：threads；在threads的包中定义下面的代码。

1. 以继承方式创建多线程类
   1. 在threads包中以继承Thread类的方式创建一个多线程类：MyThread，MyThread类覆盖父类（Thread）的run方法，在多线程任务中完成10次迭代循环，每次循环输出迭代变量(i)的值。
   2. 在threads包中定义一个可运行的主类：ThreadDemo，基本代码见下面，在在ThreadDemo的main方法中，创建两个MyThread类的对象，并启动这两个线程运行，看看程序输出的结果。

|  |  |
| --- | --- |
|  | public class ThreadDemo {  public static void main(String[] args) {  // TODO Auto-generated method stub  }  } |

1. 以实现接口的方式创建多线程类
   1. 在threads包中以实现Runnable接口的方式创建一个多线程类RunnableImpl，RunnableImpl类实现Runnable接口的抽象方法run方法，在多线程任务中完成10次迭代循环，每次循环输出迭代变量(i)的值。
   2. 同样在ThreadDemo的main方法中，创建两个基于RunnableImpl类的线程对象，并启动这两个线程运行，看看程序输出的结果，并与上一种方式创建的线程对象比较，同时掌握两种创建线程类的方式。
2. Thread类的主要方法与线程调度
   1. sleep方法的使用，按下要求修改前面定义的MyThread类。
      1. 在run方法的每次循环时增加输出当前线程的名字；
      2. 每循环一次，当前线程sleep 一次，sleep的时间为一个随机的时间（选用Math类的random()方法获取一个随机时间，可查看API文档了解random（）方法的使用）；
      3. 在ThreadDemo类的main方法中为MyThread类的两个对象命名，分别命名为“T1”和“T2”，多次重复运行ThreadDemo类看看结果，并对比题目1未添加sleep方法的运行结果。
   2. yield方法使用，基于前面的MyThread类做如下修改：
      1. 将上面a）中的sleep方法，直接替换成yield方法，同样多次重复运行ThreadDemo类查看结果，并对比与sleep方法以及题目1未添加sleep方法的运行结果。
      2. 在ThreadDemo类的main方法最后加上“System.out.println(“Main Thread End”)”,语句，运行ThreadDemo类，验证主线程的存在；在main方法中使用join()方法实现线程的调度，实现主线程让T1线程先运行完再运行。

## 课后思考

继承Thread类和实现Runnable接口是定义线程类的两种方法，比较这两种方法各自的优缺点。