《Java 程序设计》

实 验 指 导 手 册

授课教师：崔毅东，高慧

实验指导教师：崔毅东，高慧

教学对象：软件工程专业

二年级本科生

开课时间：　　秋季学期

北京邮电大学计算机学院

（国家示范性软件学院）

2023 年 09 月

一、实验二：Java 程序编制（应用练习）

1.1 实验目的

学生通过使用 Java 语言进行基本程序的开发，练习继承、多态、接口、异常处理、多线程同步、Socket 通信等。

1.2 实验内容（详见【二、实验内容说明】）

1.3 实验环境

1. Eclipse 或者 NetBean
2. 你**应该使用 git 管理你的代码**。建议你使用【国内】的免费代码托管网站托管你的代码。比如阿里云一站式 devops (http://devops.aliyun.com) 、码云等。**不建议使用 github 或者 gitlab 等【国外】的代码托管网站**。
3. 你在托管网站上的代码仓库应该设置为【**私有**】，不然会导致别人抄袭你的代码，从而使得你的实验成绩为 0

1.4 实验要求

1. 独立完成实验内容要求
2. 熟练使用 Java 常用 IDE 进行编程
3. 上交源程序文件（纸质版或者电子版，以课程指导教师要求为准）
4. 上交实验报告（纸质版或者电子版，以课程指导教师要求为准。标准格式见附件二）

1.5 实验步骤

1. 启动 Java 常用 IDE。
2. 建立 project。
3. 编辑源程序。
4. 编译、链接并执行源程序，看结果是否正确。
5. 如果报错或告警，做必要修改，重复 3－5 步骤直到没有错误和告警。

二、实验内容说明

本次实验一共 3 个 Project。

2.1 Access database

This project will exercise how to add / retrieve / update records in database. Print the source code and the screen shot of a sample run.

Write a program that views, inserts, and updates book information stored in a database. You may use just command line to view/insert/update the information. No GUI needed. When the user selects "view" option, the program displays a record with a specified ID. When the user selects "insert" option, the program inserts a new record. When the user selects "update" option, the program updates the record for the specified ID.
The Book table is created as follows:

```
create table Book (
  id int(5) not null,
  title varchar(100) not null,
  pubDate date not null,
  author char(50),
  primary key (id)
);
```

You can use any of your favorite database management systems, such as Microsoft Access, mysql, sqlite, etc.
You also need to create a database to put the previous Book table.
You MUST show the snapshot images of your dbms in your lab report.

2.2 Multi-Threading programming

Airline Reservation System

Simulate an airline reservation system with customer threads handling bookings. Use wait/notify/notifyall to handle booking conflicts and seat availability.

Some hints:

The goal of this project is to learn thread synchronization concepts using wait() and notifyAll(). Students will model multiple threads accessing a shared resource (the flight) concurrently.

(1) Core Classes:

1) Flight - Represents the shared resource. Contains a number of seats and booked reservations.
2) Customer - Represents the threads. Try to book seats on the flight.

(2) Key Steps:

1) Create a Flight with limited seats (e.g. 2 seats)
2) Create multiple Customer threads (e.g. 3 customers)
3) Customers repeatedly try to book a seat on the Flight
    3a. If seats available, book the reservation
    3b. If no seats available, call wait() on Flight object to wait for seat
    3c. Flight calls notifyAll() when a seat becomes available

Add synchronization around seat booking logic to avoid race conditions

Observe how Customers are blocked on wait() when no seats, and notified when seats become free

2.3 Socket programming

Weather Query System.

2.3.1 Overview:

Client-server project where client queries for weather data and server provides it. Focuses on socket communication and data transfer between programs.

2.3.2 Server Design:

(1) Opens TCP socket and listens for connections
(2) Accepts connections and receives query city name from client
(3) Looks up weather data for that city (can hardcode samples to start)
(4) Sends weather data back to client as string
(5) Can handle multiple sequential client connections

2.3.3 Client Design:

(1) Opens socket and connects to server
(2) Sends query city name entered by user
(3) Receives weather data string from server
(4) Prints out weather data for user

2.3.4 Example Usage:

Server starts and listens on port 5000
Client connects to server on port 5000
Client enters "Beijing" and sends it
Server receives "Beijing", looks up weather data
Server sends "Beijing: 31C and sunny"
Client prints "Beijing: 31C and sunny"

三、附件二：

实验报告模版，参见 Word 文档《实验报告模版》