

# 《Java 程序设计》实验指导书

人工智能与数据科学学院

## 《Java 程序设计》实验指导书

### 一、实验目的

Java 程序设计是计算机科学与技术、网络工程、软件工程、物联网工程专业的一门专业基础课程。本课程是一门实用性和综合运用性都很强的课程，实践教学环节是教学过程中必不可少的重要内容。通过实验，让学生熟悉 JDK 中的主要内容，掌握 JDK 调试和运行程序的方法、面向对象编程的思想、Java 中的基本方法和技术，能够熟练使用 Java 设计、编写程序，特别是基于 TCP/IP 的 Socket 编程，并能运用这些知识方法完成 C/S 和 B/S 结构程序的设计工作。通过实验，提高学生使用 Java 语言程序设计开发的能力，提高应用面向对象技术分析和解决实际问题的能力，并在此基础上强化学生的实践意识、提高其分析问题、解决问题的能力以及动手能力和创新能力。

### 二、实验要求

要求学生熟悉 JDK 中的主要内容，掌握 JDK 调试和运行程序的方法、面向对象编程的思想、Java 中的基本方法和技术，能够熟练使用 Java 设计、编写程序，特别是基于 TCP/IP 的 Socket 编程，并能运用这些知识方法完成 C/S 和 B/S 结构程序的设计工作。要注意培养学生良好的编程习惯，自始至终贯彻课程中所介绍的程序设计风格。为保证尽量在统一安排的上机时间内完成程序设计任务，学生应事先做问题分析，并做静态检查。学生应记录实验中所遇到的问题，并写出详细的实验报告。课前准备上机程序，上机认真调试，课后撰写实验报告，实验报告包括实验目的、实验内容、源程序、实验结果及分析。

### 三、实验设备及环境

装有 JDK、文本编辑软件的 PC 机。

### 四、实验学时分配

序号	项目名称	实验学时分配	内容提要
1	Java 基本语法	2	数据类型、变量和流程控制语句
2	面向对象编程	2	访问控制、继承、多态、包、接口
3	异常处理程序设计	2	异常的作用和处理
4	图形用户界面	4	图形界面、布局管理、事件处理
5	多线程程序设计	2	多线程的创建、运行及应用

6	输入/输出流	2	输入输出流和文件操作
7	使用 TCP 协议的 Socket 网络程序设计	2	SOCKET 编程，服务端、客户端程序设计
8	数据库应用	4	JDBC 连接数据库、执行 SQL 语句
合计		20	

带有星号题目为选做题目，在完成规定题目基础上可以根据学习进度选择相应题目完成。

## 实验一 Java 语言基础

**实验目的：**掌握 Java 语言的基本语法、基本数据类型的使用方法，掌握从键盘输入基本类型的数据，熟练运用分支、循环等语句控制程序流程。

**实验内容：**

1、编写一个 Java 应用程序，用户从键盘输入十名学生的信息，至少包括姓名、年龄、出生年月日、java 课程实验成绩，成绩使用浮点数，年龄使用整型，程序将输出年龄、java 课程实验成绩的平均值。

提示：Scanner 对象调用 nextDouble()或 nextFloat()可以获取用户从键盘输入的浮点数。

2、使用 Arrays 类实现数组排序：使用 java.util 包中的 Arrays 类的类方法 public static void sort(double a[])可以把参数 a 指定的 double 类型数组按升序排序；public static void sort(double a[], int start , int end)可以把参数 a 指定的 double 类型数组中从位置 start 到 end 位置的值按升序排序。

给定数组 int a[]={ 12,34,9,-23,45,6,90,123,19,45,34}; 从键盘读入一个整数，使用折半查找判断该整数是否在这个数组中，并将结果输出。

\*3、输出 100~200 之间的所有素数。

\*4、采用 for 循环求 1 至 1000 之内的所有“完全数”。所谓“完全数”是指一个数，恰好等于它的因子之和。例如，6 是一个完全数，因为 6 的因子为 1、2、3，而  $6=1+2+3$ 。

\*5、已知  $XYZ+YZZ=532$ ，其中 X、Y 和 Z 为数字，编程求出 X，Y 和 Z 的值。

## 实验二 面向对象程序设计

**实验目的：**通过编程和上机实验，掌握类和构造方法的定义及创建对象的方法，掌握类的封装及继承原则，正确使用重载和覆盖等多态概念设计可复用方法，熟悉包、接口的使用方法，掌握面向对象的程序设计方法。

**实验内容：**

1、编写 `MyDate` 类，完善上次实验中的人员信息录入，实现日期合法性判断，包括大小月和闰年。

\*2、声明一个 `Person` 类和派生类 `Student`，成员变量包括学号、姓名、入学时间、身份证号、学分绩点等信息，成员方法包括开户、存款、取款、查询（余额、明细）、销户等操作。

\*3、设计一个汽车类 `Vehicle`，包含的属性有车轮个数（`wheels`）和车重（`weight`）。小车类 `Car` 是 `Vehicle` 类的子类，其中包含的属性有载人数（`loader`）。卡车类（`Truck` 类）是 `Car` 类的子类，其中包含的属性有载重量（`payload`）。每一个类都有相关数据的输出。

实验要求：（1）汽车类 `Vehicle` 的构造方法带有 2 个参数，分别是 `wheels` 和 `weight`。`Car` 类的构造方法带有 3 个参数，分别为 `wheels`、`weight` 和 `loader`。`Truck` 的构造方法带有 4 个参数，分别为 `wheels`、`weight`、`loader` 和 `payload`。

4、定义接口 `Shape` 及其抽象方法 `getArea()`和 `getPerimeter()`用于计算图形和面积和周长。定义类 `Rectangle`(矩形)、类 `Circle`(圆形)、类 `Triangle`(三角形)，要求这些类继承点类 `Coordinates()`并实现接口的抽象方法。

提示：

```
class Coordinates
{
    long x;
    long y;
    Coordinate(long x, long y)
    {
        this.x=x;
```

```

        this.y=y;
    }
}

```

## 5、包的定义和使用

### ● 创建自定义包 Mypackage

在存放源程序的文件夹中建立一个子文件夹 Mypackage。例如，在“E:\java\程序”文件夹之中创建一个与包同名的子文件夹 Mypackage（E:\java\程序\Mypackage），并将编译过的 class 文件放入该文件夹中。注意：包名与文件夹名大小写要一致。再添加环境变量 classpath 的路径，例如：E:\j2sdk1.4.2\_01\lib;E:\java\程序

### ● 在包中创建类

YMD.java 程序功能：在源程序中，首先声明使用的包名 Mypackage，然后创建 YMD 类，该类具有计算今年的年份并输出一个带有年月日的字符串的功能。源代码如下：

```

package Mypackage;                //声明存放类的包
import java.util.*;               //引用 java.util 包
public class Test_YMD {
    private int year,month,day;
    public static void main(String[] arg3){
        public Test_YMD(int y,int m,int d) {
            year = y;
            month = (((m>=1) & (m<=12)) ? m : 1);
            day = (((d>=1) & (d<=31)) ? d : 1);
        }
        public Test_YMD() {
            this(0,0,0);
        }
        public static int thisyear() {
            return Calendar.getInstance().get(Calendar.YEAR); //返回当年的年份
        }
        public int year() {
            return year; //返回年份
        }
        public String toString() {
            return year+"-"+month+"-"+day; //返回转化为字符串的年-月-日
        }
    }
}

```

编译 Test\_YMD.java 文件，然后将 Test\_YMD.class 文件存放到 Mypackage 文件夹中。

### ● 编写使用包 Mypackage 中 Test\_YMD 类的程序

YMD\_2.java 程序功能：给定某人姓名与出生日期，计算该人年龄，并输出该人姓名，年龄，出生日期。程序使用了 Test\_YMD 的方法来计算年龄。源代码如下：

```
import Mypackage.KY4_1_YMD; //引用 Mypackage 包中的 KY4_1_YMD 类
public class YMD_2
{
    private String name;
    private Test_YMD birth;
    public static void main(String args[])
    {
        YMD_2 a = new YMD_2("张弛",1990,1,11);
        a.output();
    }
    public YMD_2(String n1,Test_YMD d1)
    {
        name = n1;
        birth = d1;
    }
    public YMD_2(String n1,int y,int m,int d)
    {
        this(n1,new Test_YMD(y,m,d)); //初始化变量与对象
    }
    public int age() //计算年龄
    {
        return TEST_YMD.this.year() - birth.year();
        //返回当前年与出生年的差即年龄
    }

    public void output()
    {
        System.out.println("姓名 : "+name);
        System.out.println("出生日期: "+birth.toString());
        System.out.println("今年年龄 : "+age());
    }
}
```

编译并运行程序。

## 实验三、异常处理程序设计

**实验目的：**了解 Java 中异常处理（exception）的作用及常用的异常类，掌握异常处理的设计方法。

**实验内容：**

**\*1、用 try-catch-finally 结构实现异常处理。编译并运行程序，写出程序运行结果。**

```
public class ExceptionTest{  
{  
    public static void main( String args[] )  
    {  
        int i=0;  
        String greeting[]={ "Hello", "Only", "Test"};  
        while ( i < 4 )  
        {  
            try { System.out.println(greeting[i]); }  
            catch (ArrayIndexOutOfBoundsException e)  
            {  
                System.out.println(" 数组越界 ");  
            }  
            finally{  
                System.out.println("总会运行");  
            }  
            i++;  
        }  
    }  
}
```

2、设计一个 Java 程序，自定义异常类，从命令行（键盘）输入一个字符串，如果该字符串值为“XYZ”，则抛出一个异常信息“This is a XYZ”，如果从命令行输入 ABC，则没有抛出异常。（只有 XYZ 和 ABC 两种输入）。

3、声明一个 Average 接口，继续完善学生信息录入程序，其中约定求平均值的方法；声明多个类实现 Average 接口，分别给出求平均值的方法实现。例如，在一组数值中，一种算法是，全部数值相加后求平均值，另一种算法是，去掉一个最高分和一个最低分后，再将总分求平均等；使用键盘输入数据时，对于不能转换成数值的字符串进行异常处理。

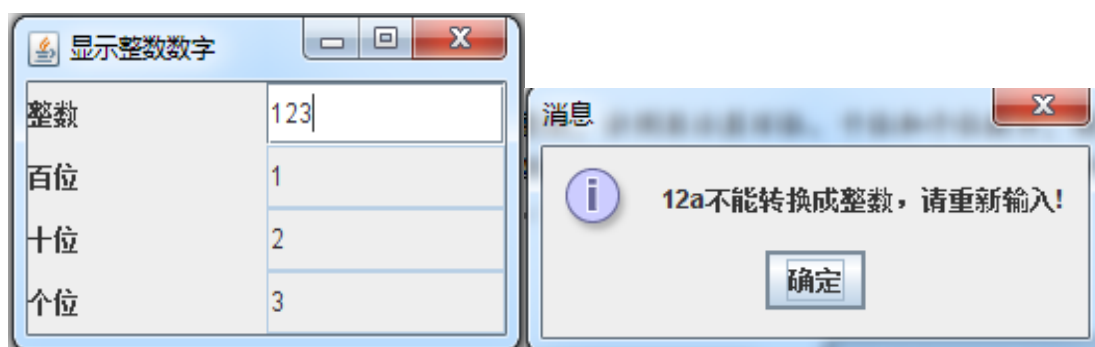


## 实验四 图形用户界面程序设计

**实验目的：**掌握组件的使用方法，理解委托事件处理模型。熟悉图形用户界面基本组件的使用方法，熟悉如何使用布局管理器对组件进行管理及如何使用 Java 的事件处理机制。

**实验内容：**

\*1、输入一个整数，分别显示其百位、十位和个位数字，图形用户界面如图 4.1 所示。要求：整数文本行可编辑且能实现事件处理，当输入数据错误时，处理异常，弹出对话框，提示重新输入信息；其他文本行仅用于显示不可编辑。



\*2、模拟实现一个可视化的简单计算器，至少提供进行加法、减法、乘法、除法等基本运算的功能，希望能支持加正负号、求平方根、清零等其他功能。

3、设计图形页面实现学生的信息录入，至少包括姓名、年龄、出生年月日、java 课程实验成绩，成绩使用浮点数，年龄使用整型，使用数据存储输入对象，程序输出按年龄排序的学生信息。

## 实验五 多线程程序设计

**实验目的：**理解多线程的概念，掌握创建、管理和控制 Java 线程对象的方法，包括创建 Java 线程对象、改变线程状态、设置线程优先级及控制线程调度等方法，掌握实现线程互斥和线程同步的方法。

**实验内容：**

1、编写一个有两个线程的程序，第一个线程用来计算1~100之间的偶数及个数，第二个线程用来计算1-100之间的偶数及个数。

2、编写一个Java应用程序，在主线程中再创建两个线程，要求线程经历四种状态：新建，运行、中断和死亡。按模板要求，将【代码1】~【代码8】替换为Java程序代码。

```
class Tortoise extends Thread
{
    int sleepTime=0, liveLength=0;

    public Tortoise(String name,int sleepTime, int liveLength)
    {
        this.sleepTime=sleepTime;
        this.liveLength=liveLength;
        【代码1】 // 设置线程的名字为name
    }

    public void run()
    {
        while (true )
        {
            liveLength--;
            System.out.println("@_@");
            try{
                【代码2】 // 让线程调用sleep()方法进入中断状态
            }
            catch (InterruptedException e) {}
            if (liveLength<=0 )
            {

```

```
System.out.println(getName()+"进入死亡状态\n");
```

【代码3】 // 结束run()方法的语句

```
    }  
    }  
}  
  
class Rabbit extends Thread  
{  
    int sleepTime=0, liveLength=0;  
    public Rabbit(String name,int sleepTime, int liveLength)  
    {  
        【代码4】 // 调用父类构造函数，设置线程的名字为name  
        this.sleepTime=sleepTime;  
        this.liveLength=liveLength;  
    }  
    public void run()  
    {  
        while (true )  
        {  
            liveLength--;  
            System.out.println("*_*");  
            try{  
                sleep( sleepTime);  
            }  
            catch (InterruptedException e) {}  
            if (liveLength<=0 )  
            {  
                System.out.println(getName()+"进入死亡状态\n");  
                break;  
            }  
        }  
    }  
}
```

```
        }  
    }  
}  
public class ThreadExample  
{  
    public static void main(String a[])  
    {  
        Rabbit rabbit;  
        rabbit = 【代码5】 // 新建线程rabbit  
        Tortoise tortoise = 【代码6】 // 新建线程tortoise  
        【代码7】 // 启动线程tortoise  
        【代码8】 // 启动线程rabbit  
    }  
}
```

\*3、编写Java应用程序模拟5个人排队买票。售票员只有1张五元的钱，电影票五元钱一张。假设5个人的名字及排队顺序是：赵、钱、孙、李、周。“赵”拿1张二十元的人民币买2张票，“钱”拿1张二十元的人民币买1张票，“孙”1张十元的人民币买1张票，“李”拿1张十元的人民币买2张票，“周”拿1张五元的人民币买1张票。

要求售票员按如下规则找赎：

- (1) 二十元买1张票，找零：1张十元；不许找零2张五元。
- (2) 二十元买1张票，找零：1张十元，1张五元；不许找零3张五元。
- (3) 十元买一张票，找零1张五元。

## 实验六 输入输出流

**实验目的：**理解文件和流的概念、Java 流的层次结构，掌握 Java 提供的各种字节流类和字符流类的功能和使用方法。

**实验内容：**

1、编写一个 Java 应用程序，将已存在的扩展名为.txt 的文本文件加密后存入另一个文本文件中。按模板要求，将【代码 1】~【代码 7】替换为 Java 程序代码。

```
import java.io.*;

class SecretExample
{
    public static void main(String a[])
    {
        File fileone=new File("hello.txt");
        File filetwo=new File("hello.secret");
        char b[]=new char[100];
        try{
            FileReader in = 【代码 1】 //创建指向 fileone 的字符输入流
            FileWriter out = 【代码 2】 //创建指向 fileontwo 的字符输出流
            int n=-1;
            while ( (n=in.read(b)) != -1 )
            {
                for (int i=0; i<n;i++)
                {   b[i]=(char)(b[i]^'a');   }
                【代码 3】      //out 将数组 b 的前 n 单元写到文件
            }
            【代码 4】      //out 关闭
            in = 【代码 5】 //创建指向 fileontwo 的字符输入流
            System.out.println("加密后的文件内容：");
            n=in.read(b);
            while ( n != -1 )
            {
                String str=new String(b,0,n);
```

```

        System.out.println("dfjakfjd"+str);
        n=in.read(b);
    }
    in=new FileReader(filetwo);
    System.out.println("解密后的文件内容: ");
    while ( (n=in.read(b)) != -1 )
    {
        for (int i=0; i<n;i++)
        {   b[i]=(char)(b[i]^'a');   }
        String str=new String(b,0,n);
        System.out.println(str);
    }
    【代码 6】    // in 关闭
}
catch(IOException e)
{
    System.out.println(e);
}
}
}

```

## 2、编程完成下列功能：

- 1) 首先建立两个文件 myfiel.txt 和 myfile2.txt。
- 2) 从标准设备中输入多名学生信息，如学号，姓名，专业，班级，家庭住址等，待输入"bye"时结束，将输入内容保存在 myfile1.txt 文件中。
- 3) 再将 myfile1.txt 文件中内容拷贝到 myfile2.txt。

## 实验七 使用 TCP 协议的 Socket 网络程序设计

**实验目的：**理解 Socket 通信原理，掌握使用 Socket 和 ServerSocket 类进行 TCP Socket 通信的程序设计方法。

**实验内容：**

1、使用 ServerSocket 类和 Socket 类实现按如下协议通信的服务器端和客户端程序。

服务器程序的处理规则如下：

- 1) 向客户端程序发送 Verifying Server!。
- 2) 若读口令次数超过 3 次，则发送 Illegal User!给客户端，程序退出。否则向下执行步骤 3)。
- 3) 读取客户端程序提供的口令。
- 4) 若口令不正确，则发送 PassWord Wrong!给客户端，并转步骤 2)，否则向下执行步骤 5)。
- 5) 发送 Registration Successful!给客户端程序。

客户端程序的处理规则如下：

- 1) 读取服务器反馈信息。
- 2) 若反馈信息不是 Verifying Server!，则提示 Server Wrong!，程序退出。否则向下执行步骤 3)。
- 3) 提示输入 PassWord 并将输入的口令发送给服务器。
- 4) 读取服务器反馈信息。
- 5) 若反馈信息是 Illegal User!，则提示 Illegal User!，程序退出。否则向下执行步骤 6)。
- 6) 若反馈信息是 PassWord Wrong!，则提示 PassWord Wrong!，并转步骤 3)，否则向下执行步骤。
- 7) 输出 Registration Successful!。

2、实现聊天室功能。

- 1) 完成服务器端和一个客户端之间的聊天功能。
- 2) 扩展部分，采用多线程技术实现一个服务器端和多个客户端之间的聊天功能。

## 实验八 数据库应用

**实验目的：**了解数据库系统、关系模型、客户-服务器模式等基本概念，掌握 SQL 的数据定义、数据操纵和数据查询等语句的语法；了解 JDBC 各种数据库驱动程序类型，熟悉 JDBC 提供的接口和类，掌握指定驱动类型、连接数据库、执行 SQL 语句、处理结果集等操作方法。

**实验内容：**

数据库应用程序

1) 首先在数据库应用程序中创建数据库 Studentinfo，按照下表的结构在数据库中建立"student"表。

字段名	Java 数据类型	宽度	SQL 数据类型
Name	String	10	Char(10)
Sex	String	2	Char(2)
Age	Int	3	Integer

2) 编写程序，a)向"student"表中填入若干数据记录；b)在"student"表中分别查询所有记录以及满足条件"age>18"的记录。c)编程创建学生成绩表，并进行数据插入、修改、删除、查询和成绩统计等操作。

注：不限定数据库类型，