**实验3类和对象**

**一、实验时间**

2023年9月28日星期四

**二、实验目的**

1、理解Java面向对象程序设计的基本思想；

2、理解Java封装的思想，Java封装思想的实现机制；

3、掌握Java类的定义，包括构造方法的使用、get方法和set方法的定义及使用、toString方法的定义及使用；

4、掌握Java对象及对象数组的创建、方法和数据成员的访问。

**三、实验内容**

１、定义一个Book类，要求：

（1）定义该类的bookID、bookName、price、author等属性，各属性的数据类型大家根据实际进行定义，bookID的值要求自动增长（其值是每个Book对象产生时，自动让bookID加1得到）

（2）分别定义该类的无参构造函数、带bookName、author参数的构造函数、以及带除bookID之外的其他所有属性参数的构造函数；

（3）bookID的值要求在构造代码块中形成；

（4）定义Book类所有属性的set方法和get方法，bookID属性不能定义其set方法；

（5）定义Book类的toString方法；

２、定义一个BookTest类，要求：

（1）创建一个Book数组，每本书的初始化信息需要从键盘输入；

（2）定义sortByPrice方法，实现按价格对Book数组进行排序（用快速排序）；

（3）定义searchByAuthor方法，实现查找指定作者对应的Book信息，不能在该方法中打印指定的Book信息；

（4）在main方法中，分别调用上述方法，并打印排序后的Book数组对象的信息，以及指定作者的Book信息。

**四、实验过程**

1、实验原理分析（写出你采用的实验理论基础）（10分）

实验主要理论为类的封装，包括重载函数来写方法，定义所有属性的Set方法主要是对于类的方法的使用和声明。对于bookID用到了静态变量的方法的用法，同时学习了Java的函数设计。

2、实验设计（写出实验的算法设计）（25分）

快速排序算法设计：

首先在待排序数组中随便选择一个数作为节点（pivot），然后从最后面（high）往左查找比这个节点（pivot）小的数，并且从最前面（low）往右查找比这个节点（pivot）大的数（low），情况1：找到后就把这两个数进行交换，然后接着上面的查找交换直到low等于high，球后将节点（pivot）与low位置处的数进行交换，这样比pivot小的数都在其前面，比pivot大的数就在其后面，然后把数组以pivot分为两半，重复上述操作；情况2：直到low 等于 high都没有找到，就直接交换pivot和low位置的数据，然后同样的将数组以pivot分为两半重复上述操作。

3、实验过程（写出实验过程中遇到的问题及解决的思路和方法）（30分）

实验过程中遇到了JAVA语言的报错，发现数组溢出的问题。通过更改传参数的值。

４、实验源码（20分）

package 源代码存储.Java.实验3.Code;

public class Book {

    int bookID;

    String bookName;

    double price=0;

    String author;

    private static int nums=0;

    public Book()

    {

        //默认的构造函数

        {//bookID构造

            bookID=nums;

            nums++;

        }

        bookName ="未知";

        price=0.0;

        author="未知";

    }

    public Book(String bookName,double price,String author)

    {

        {//bookID构造

            bookID=nums;

            nums++;

        }

        this.bookName=bookName;

        this.price=price;

        this.author=author;

    }

    public Book(String bookName,String author)

    {

        {//ookID构造

            bookID=nums;

            nums++;

        }

        this.bookName=bookName;

        this.author=author;

    }

    public void set\_bookName(String bookName)

    {

        this.bookName=bookName;

    }

    public void set\_author(String author)

    {

        this.author=author;

    }

    public void set\_price(double price)

    {

        this.price=price;

    }

    public double get\_price()

    {

        return this.price;

    }

      public String get\_author()

    {

        return this.author;

    }

    public int  get\_bookID()

    {

        return this.bookID;

    }

    public String get\_bookName()

    {

        return this.bookName;

    }

    public String toString()

    {

        return " 本书编号"+bookID+" 名称"+bookName+" 价格"+price+" 作者"+author;

    }

}

BookTest.java

package 源代码存储.Java.实验3.Code;

import java.util.Scanner;

public class BookTest

{

    public static void main(String[] args) {

        Book []book=new Book[100];

        Scanner sc=new Scanner(System.in);

        String n,m;

        double t;

        int j=0;

        do

        {

            System.out.println("书名");

            n=sc.next();

            if(n=="1") break;

            System.out.println("价格");

            t=sc.nextDouble();

            if(t==0) break;

            System.out.println("作者");

            m=sc.next();

            book[j]=new Book(n,t,m);

            j++;

        }while (t!=0);

        for(int i=0;i<j;i++)

        {

            System.out.println(book[i]);

        }

        SortByprice(book,0,j-1);

        for(int i=0;i<j;i++)

        {

            System.out.println(book[i]);

        }

        System.out.println("查谁");

        searchByAuthor(book,sc.next(),j);

        sc.close();

    }

    public static void SortByprice(Book[] book,int s,int e)

    {

        if (s >= e) return;

        int i = s, j = e;

        double pivot = book[s].price;

        Book pivot1=book[s];

        while (i < j) {

            while (i < j && book[i].price <= pivot) {

                i++;

            }

            while (i < j && book[j].price >= pivot) {

                j--;

            }

            if (i < j) {

                Book temp = book[i];

                book[i] = book[j];

                book[j] = temp;

            }

        }

        book[s] = book[i];

        book[i] = pivot1;

        SortByprice(book, s, i - 1);

        SortByprice(book, i + 1, e);

    }

    public static void searchByAuthor(Book[] book,String author,int length)

    {

        int flag=0;

        for(int i=0;i<length;i++)

        {

            if(book[i].author.equals(author))

            {

                System.out.println(book[i]);

                flag=1;

            }

        }

        if(flag==0) System.out.println("无");

    }

}

５、运行结果（要求对结果分析，分别对输入出错和输入正确的结果进行分析）（10分）

书名

ljx

价格

2

作者

ljx

书名

wxha

价格

200

作者

ljx2

书名

0

价格

0

本书编号0 名称ljx 价格2.0 作者ljx

本书编号1 名称wxha 价格200.0 作者ljx2

本书编号1 名称wxha 价格200.0 作者ljx2

本书编号0 名称ljx 价格2.0 作者ljx

查谁

ljx2

本书编号1 名称wxha 价格200.0 作者ljx2

6、实验总结（根据实验的过程和结果分析写出你对实验原理的理解）（5分）

我学会了有关类的方法的定义和调用使用。也学会了函数重载的方式方法。这真是一次好的实验机会呢。

评分表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 项目 | 总分 | 评分标准 | 实际得分 |
| 1 | 实验原理分析 | 10 | 对实验全过程涉及的知识点应用进行有效分析 |  |
| 2 | 实验设计 | 25 | 类的封装机制、继承机制、多态机制、异常机制的设计（10分），类的属性成员的设计（5分），类的方法成员的设计（5分），测试部分输入的设计（3分）、测试部分输出的设计（2分） |  |
| 3 | 实验过程 | 30 | 对遇到的问题进行分析（10分），解决思路（5分）、方案（5分）、过程（10分）， |  |
| 4 | 实验源码 | 20 | 实验源码的书写规范性（2分）、标识符的命名规范（2分）、是否有注释（2分）；代码的正确性（14分） |  |
| 5 | 运行结果 | 10 | 是否有运行截图（5分），是否对结果进行分析（5分） |  |
| 6 | 实验总结 | 5 | 对实验是否进行了有效总结，主要体现在对知识点的理解，实验过程的总结、知识点应用总结 |  |
| 总分 |  | | | |
| 实验评价 |  | | | |