
Table of Contents

Introduction	1.1
第一天	1.2
第二天	1.3
第三天	1.4
第四天	1.5
第五天	1.6
第六天	1.7
第七天	1.8
第八天	1.9
第九天	1.10
第十天	1.11

JavaSE小目标集训营实训体系

小目标集训营，是卡尔信息为了提高大学生的动手能力、贴近企业实战。专门打造的集中实训体系。

介绍java应用领域以及发展前景

DOS常用命令

安装JDK及配置开发环境

DOS编程及编写java程序

java注释

标识符及变量命名规范

java数据类型

java变量

java运算符

运算符优先级

类型转换

条件分支 if...else

switch

if...else if else

分支嵌套

switch与if区别

java循环 while do...while

java for循环

break, continue, return

循环嵌套

方法的定义

方法形参和实参

方法返回值

方法重载

数组的定义

数组赋值

数组取值

数组循环遍历

arrays类的应用

面向对象编程

面向对象与面向过程区别

类

对象

属性

方法

成员变量及成员方法

构造方法定义及使用

构造方法重载

封装

this的使用

变量作用域

封装

继承

super关键字

java中访问控制权限

方法重写

包机制

抽象类、接口

多态机制

java常用类

String和Stringbuffer

Date

Math

Random

System

异常处理

异常处理概念机分类

异常处理方式

异常处理使用

自定义异常

集合框架

集合框架体系

Collection

List

ArrayList

LinkedList

Set

Collections工具类

IO

BufferedReader

文件操作

多线程

单例模式

项目实战

知识总结

卡尔信息图书管理系统

Books.java

```
package com.carlinfo.dao;

/**
 * @authoryangguojun
 * 图书集合信息
 */

public class Books {

    // 名称

    String []name= new String[50];

    // 是否借出 0 已经借出。1 可借

    int []state= newint[50];

    // 借出日期

    String []date= new String[50];

    // 借出次数

    int []count = newint[50];

}
```

BookManager.java

```
package com.carlinfo.dao;

import java.text.SimpleDateFormat;

import java.util.Date;

import java.util.Scanner;

/**
 * @authoryangguojun
 *
 */

public class BookManager {

    Books book = new Books();

    public void initial() {

        book.name[0]="Hadoop构建数据仓库实践";

        book.count[0]=10;

        book.state[0]=1;

    }

}
```

```
book.name[1]="平台战略";

book.count[1]=20;

book.state[1]=0;

book.date[1]="2017-7-1";

book.name[2]="Python编程";

book.count[2]=10;

book.state[2]=1;

}

public void StartMenu() {

    System.out.println("欢迎使用图书管理系统");

    System.out.println("-----");

    System.out.println("0. 借阅排行榜");

    System.out.println("1. 增加图书");

    System.out.println("2. 查看图书");

    System.out.println("3. 删除图书");

    System.out.println("4. 借书图书");

    System.out.println("5. 归还图书");

    System.out.println("6. 退出 ");

    System.out.print("-----\n");

    System.out.println("请选择");

    Scanner input = new Scanner(System.in);

    int choice = input.nextInt();

    switch(choice){

        case 0:

            list();

            break;

        case 1:

            add();

            break;

        case 2:

            search();

            break;

        case 3:

            delete();
```

```
break;

case 4:

lend();

break;

case 5:

returnBook();

break;

case 6:

System.out.println("\n谢谢 使用! ");

break;

}

}

/**

 * 返回主菜单

 */

public void returnMain(){

Scanner input = new Scanner(System.in);

System.out.print("输入0返回: ");

if(input.nextInt() == 0){

StartMenu();

}else{

System.out.println("输入错误, 异常终止! ");

}

}

public void add() {

System.out.println("增加书籍");

Scanner input = new Scanner(System.in);

System.out.println("请输入书籍名称");

String name = input.next();

for (int i = 0; i < book.name.length; i++) {

if(book.name[i]==null)

{

book.name[i]=name;

book.state[i]=1;

}
```

```
System.out.println("图书<<"+book.name[ij]+">>增加成功");

break;
}
}

System.out.println("*****");

returnMain();
}

public void list() {
// 定义新数组，用来存放排序后book信息

String[] newname = new String[50]; //
int[] newcount = newint[50];

for (int k = 0; k < book.name.length; k++) {
newname[k] = book.name[k];
newcount[k] = book.count[k];
}

// 利用冒泡排序算法进行排序

for (int i = 0; i < newname.length; i++) {
for (int j = i + 1; j < newname.length ; j++) {

if (newcount[i] > newcount[j]) {
int tempc = newcount[i];
newcount[i] = newcount[j];
newcount[j] = tempc;

String tempn = newname[i];
newname[i] = newname[j];
newname[j] = tempn;
}
}
}

System.out.println("---> 排行榜\n");

System.out.println("*****");

System.out.println("次数\t名称");

// 显示排行榜信息

for (int i = newname.length - 1; i >= 0; i--) {

if (newname[i] != null) {
```

```
System.out.println(newcount[i] + "\t<<" + newname[i] + ">>");

}

}

System.out.println("*****");

returnMain();

}

public void lend(){

System.out.println("----> 借阅图书\n");

Scanner input = new Scanner(System.in);

System.out.print("请输入书籍名称: ");

String want = input.next();//要借出的Book名称

for(int i = 0; i < book.name.length; i++){

if(book.name[i] == null){//无匹配

System.out.println("没有找到匹配信息! ");

break;

}elseif(book.name[i].equals(want)&&book.state[i]==1){//找到匹配可借

book.state[i] = 0;

SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd hh:mm:ss");

String date = format.format(new Date());

book.date[i]=date;

System.out.println("借出 《"+want+"》 成功!");

book.count[i]++;

break;

}elseif(book.name[i].equals(want)&&book.state[i]==0){//找到匹配已被借出

System.out.println("《"+want+"》 已被借出! ");

break;

}

}

System.out.println("*****");

returnMain();

}

public void delete() {

Scanner input = new Scanner(System.in);

boolean flag = false;//标识删除成功与否
```



```
System.out.println("---> 删除书籍\n");

System.out.print("请输入书籍名称: ");

String name = input.next();

// 遍历数组, 查找匹配信息

for (int i = 0; i < book.name.length; i++) {

    // 查找到, 每个元素前移一位

    if (book.name[i] != null && book.name[i].equalsIgnoreCase(name)

        && book.state[i] == 1) {

        int j = i;

        while (book.name[j + 1] != null) {

            book.name[j] = book.name[j + 1];

            book.state[j] = book.state[j + 1];

            book.date[j] = book.date[j + 1];

            j++;

        }

        // 最后一个不为空的元素置空

        book.name[j] = null;

        book.date[j] = null;

        System.out.println("删除 《" + name + "》 成功! ");

        flag = true; // 置位, 表示删除成功

        break;

    } elseif (book.name[i] != null

        && book.name[i].equalsIgnoreCase(name) && book.state[i] == 0) {

        System.out.println("《" + name + "》 为借出状态, 不能删除! ");

        flag = true; // 置位

        break;

    }

}

if (!flag) {

    System.out.println("没有找到匹配信息! ");

}

System.out.println("*****");

return Main();

}
```



```
book.state[i] = 1;

SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd hh:mm:ss");

String date = format.format(new Date());

System.out.println("\n归还 《"+want+"》 成功!");

System.out.println("借出日期为: "+book.date[i]);

System.out.println("归还日期为: "+date);

break;

}elseif(book.name[i].equals(want) &&book.state[i]==1){ //找到匹配但没有借出

System.out.println("该书籍没有被借出! 无法进行归还操作。");

break;

}

}

System.out.println("*****");

returnMain();

}

}
```

Test.java

```
package com.carlinfo.dao;

public class Test {

    public static void main(String[] args) {

        // TODO Auto-generated method stub

        BookManager bm = new BookManager();

        bm.initial();

        bm.StartMenu();

    }

}
```