**实验7集合类**

**一、实验时间**

**二、实验目的**

1、了解集合类框架的体系结构；

2、掌握Collection、List、Set、ArrayList、HashSet、TreeSet的常用方法，进一步理解List和Set的区别；

3、掌握Map、HashMap的常用方法，进一步理解Collection和Map的区别；

4、掌握迭代器的使用；

5、能够使用上述类、接口完成实际应用问题的程序实现。

**三、实验内容**

（1）定义Book类，定义其属性bID、bName、author、publishDate，其中包含方法：

无参和带参构造方法；

各属性的get和set方法；

toString方法；

（2）定义一个BookManagement类，定义一个集合类，用于保存图书信息，并实现如下方法：

addBook，向集合类中添加书籍信息；

deleteBook、从集合类中删除指定编号的图书信息；

searchBookByID，根据编号查找书籍信息；

searchBookByName，根据书名查找书籍信息，要求可以实现模糊查找，例如查找书名包含“计算机”字样的书籍信息；

sortByID，根据书的编号进行排序，可以根据参数指定升序或降序排序方式。

printBooksInfo，打印输出所有书籍的信息；

（3）定义BookTest类，定义main方法：

新建BookManagement类的对象，添加8本以上的书籍信息，分别调用上述方法实现对书籍的各种管理操作。

PS：

各类的属性和方法不限于实验内容要求，可由各位同学自行思考分析增加；

BookManagement类中的集合类可以尝试使用ArrayList、HashSet、TreeSet、HashMap实现。

**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）

**集合类是Java语言中用于存储和管理数据的核心框架。它包括了Collection、List、Set、ArrayList、HashSet、TreeSet等接口和类，这些接口和类构成了集合类框架的体系结构。还有Java下util包的Date类。**

2、实验设计（写出实验的算法设计）（25分）

**从book.java中没有任何特殊的算法方面的设计，重写了toString方法，方便输出每本书的信息。只进行了一些set，get的声明，在BookManagement里主要运用了哈希BookMap对Book进行管理，其key值主要就是ID，因为在实际生产生活中，book的ID一般是不会重复的，所以利用ID充当其key是最合适的。搜索里，主要调用字符串的contains方法判断是否包含了标题，这只模糊搜索，而且返回的是list。其他部分都是对collections或list的调用。**

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）

**我发现对于sort过后用hashmap不会是排序后的顺序存储，所以我选择在函数中输出排序好的BookList用来实现输出的方法。**

４、实验源码（20分）

//Book.java

package 源代码存储.Java.实验7.Code;

import java.util.Date;

public class Book {

    private String bID;

    private String bName;

    private String author;

    private Date publishDate;

    // 无参构造方法

    public Book() {

    }

    // 带参构造方法

    public Book(String bID, String bName, String author, Date publishDate) {

        this.bID = bID;

        this.bName = bName;

        this.author = author;

        this.publishDate = publishDate;

    }

    // get和set方法

    public String getbID() {

        return bID;

    }

    public void setbID(String bID) {

        this.bID = bID;

    }

    public String getbName() {

        return bName;

    }

    public void setbName(String bName) {

        this.bName = bName;

    }

    public String getAuthor() {

        return author;

    }

    public void setAuthor(String author) {

        this.author = author;

    }

    public Date getPublishDate() {

        return publishDate;

    }

    public void setPublishDate(Date publishDate) {

        this.publishDate = publishDate;

    }

    // toString方法，用于打印书籍信息

    @Override

    public String toString() {

        return "Book [bID=" + bID + ", bName=" + bName + ", author=" + author + ", publishDate=" + publishDate + "]";

    }

}

//BookManagement.java

package 源代码存储.Java.实验7.Code;

import java.util.\*;

public class BookManagement {

    private Map<String, Book> bookMap;

    public BookManagement() {

        bookMap = new HashMap<>();

    }

    // 添加书籍信息

    public void addBook(String ID, String Name, String author, Date publishDate) {

        Book book = new Book(ID,Name,author,publishDate);

        bookMap.put(ID, book);

    }

    // 删除书籍信息

    public void deleteBook(String ID) {

        bookMap.remove(ID);

    }

    // 根据ID搜索书籍信息

    public Book searchBookByID(String ID) {

        return bookMap.get(ID);

    }

    // 根据书名搜索书籍信息

    public List<Book> searchBookByName(String Name) {

        List<Book> result = new ArrayList<>();

        for (Book book : bookMap.values()) {

            if (book.getbName().contains(Name)) {

                result.add(book);

            }

        }

        return result;

    }

    // 根据ID排序书籍信息

    public void sortBooksByID() {

        List<Book> bookList = new ArrayList<>(bookMap.values());

        Collections.sort(bookList, Comparator.comparing(Book::getbID)); //调用排序算法

        for (int i = 0; i < bookList.size(); i++) {

            bookMap.put(bookList.get(i).getbID(), bookList.get(i));

        }

        for(Book bk:bookList) System.out.println(bk);

    }

    // 根据书名排序书籍信息

    public void sortBookByName() {

        List<Book> bookList = new ArrayList<>(bookMap.values());

        Collections.sort(bookList, Comparator.comparing(Book::getbName));

        for (int i = 0; i < bookList.size(); i++) {

            bookMap.put(bookList.get(i).getbID(), bookList.get(i));

        }

        for(Book bk:bookList) System.out.println(bk);

    }

    // 打印所有书籍信息

    public void printBooksInfo() {

        for (Book book : bookMap.values()) {

            System.out.println(book);

        }

    }

}

//BookTest.java

package 源代码存储.Java.实验7.Code;

import java.util.Date;

public class BookTest {

    public static void main(String[] args) {

        // 创建BookManagement对象

        BookManagement bookManagement = new BookManagement();

        // 添加书籍信息

        bookManagement.addBook("9875631", "《Java程序设计》", "某老师", new Date());

        bookManagement.addBook("9875632", "《C primer plus》", "author 2", new Date());

        bookManagement.addBook("9875633", "Book3", "Author3", new Date());

        bookManagement.addBook("9875634", "Book4", "Author4", new Date());

        bookManagement.addBook("9875635", "Book5", "Author5", new Date());

        bookManagement.addBook("9875636", "Book6", "Author6", new Date());

        bookManagement.addBook("9875637", "Book7", "Author7", new Date());

        bookManagement.addBook("9875638", "Book8", "Author8", new Date());

        // 调用方法管理书籍信息

        bookManagement.printBooksInfo(); // 打印所有书籍信息

        // 根据ID排序

        System.out.println("After sorting by ID:");

          bookManagement.sortBooksByID();

        // 根据书名排序

        System.out.println("After sorting by Name:");

        bookManagement.sortBookByName();

        //查找

        System.out.println("调用查找算法"+(bookManagement.searchBookByName("《Java程序设计》")));

        System.out.println("调用查找算法"+(bookManagement.searchBookByID("9875632")));

    }

}

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）

**Book [bID=9875631, bName=《Java程序设计》, author=某老师, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875634, bName=Book4, author=Author4, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875635, bName=Book5, author=Author5, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875632, bName=《C primer plus》, author=author 2, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875633, bName=Book3, author=Author3, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875638, bName=Book8, author=Author8, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875636, bName=Book6, author=Author6, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875637, bName=Book7, author=Author7, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**After sorting by ID:**

**Book [bID=9875631, bName=《Java程序设计》, author=某老师, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875632, bName=《C primer plus》, author=author 2, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875633, bName=Book3, author=Author3, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875634, bName=Book4, author=Author4, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875635, bName=Book5, author=Author5, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875636, bName=Book6, author=Author6, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875637, bName=Book7, author=Author7, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875638, bName=Book8, author=Author8, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**After sorting by Name:**

**Book [bID=9875633, bName=Book3, author=Author3, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875634, bName=Book4, author=Author4, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875635, bName=Book5, author=Author5, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875636, bName=Book6, author=Author6, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875637, bName=Book7, author=Author7, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875638, bName=Book8, author=Author8, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875632, bName=《C primer plus》, author=author 2, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**Book [bID=9875631, bName=《Java程序设计》, author=某老师, publishDate=Sun Nov 12 20:35:12 CST 2023]**

**调用查找算法[Book [bID=9875631, bName=《Java程序设计》, author=某老师, publishDate=Sun Nov 12 20:35:12 CST 2023]]**

**调用查找算法Book [bID=9875632, bName=《C primer plus》, author=author 2, publishDate=Sun Nov 12 20:35:12 CST 2023]**

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）

通过本次实验，我对集合类框架的体系结构有了更深入的了解，掌握了Collection、List、Set、ArrayList、HashSet、TreeSet的常用方法，进一步理解了List和Set的区别，同时也掌握了Map、HashMap的常用方法，进一步理解了Collection和Map的区别。在实验过程中，我尝试使用HashMap实现BookManagement类中的集合类，并对书籍信息进行了各种管理操作，包括添加书籍信息、删除书籍信息、根据编号查找书籍信息、根据书名模糊查找书籍信息、根据编号排序书籍信息等。

通过本次实验，我不仅掌握了集合类的常用方法和使用技巧，还学会了如何使用迭代器来遍历集合类中的元素。同时，我也发现了自己在编程中容易忽略的一些细节问题，例如在定义类的属性时需要同时提供get和set方法，否则在调用这些属性时会报错。此外，我也学会了如何对测试部分的输入和输出进行设计，以确保测试的正确性和完整性。

总之，本次实验让我对集合类有了更深入的了解和掌握，同时也让我在编程中更加注意细节问题。

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 |  | | | |