

SA第九次作业

1、针对Iterator的例子，将存储Book用的数组换成其他Collection并运行。

实现方式：

重新编写BookShelf类，将private Book[] books;改为private List books;，使用ArrayList替代原数组，同理修改其方法即可。

```
public class BookShelf implements Aggregate {
    private List<Book> books; // 使用ArrayList替换数组

    public BookShelf(int maxsize) {
        this.books = new ArrayList<>(maxsize); // 初始化ArrayList，设置初始容量
    }

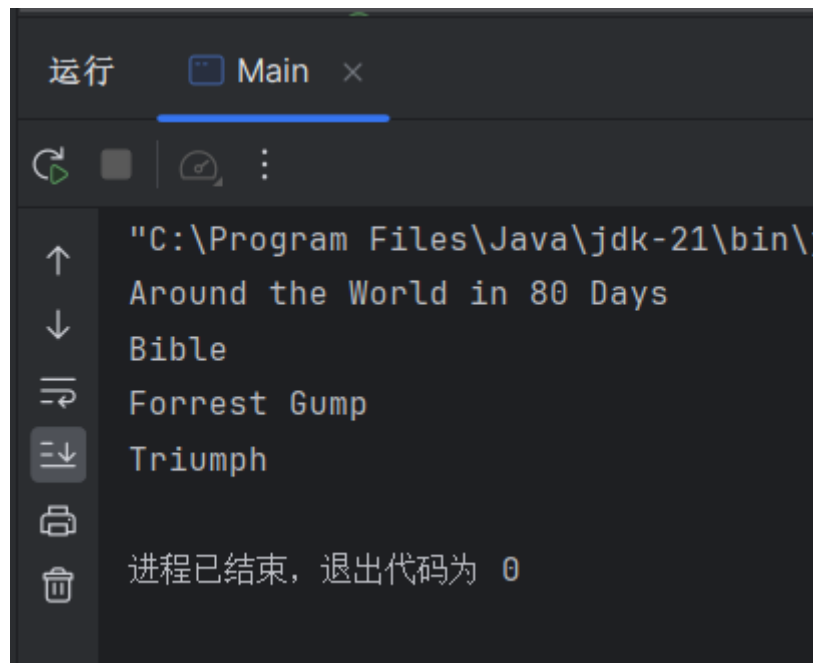
    public Book getBookAt(int index) {
        return this.books.get(index); // 使用ArrayList的get方法
    }

    public void appendBook(Book book) {
        this.books.add(book); // 使用ArrayList的add方法
    }

    public int getLength() {
        return this.books.size(); // 使用ArrayList的size方法
    }

    public Iterator iterator() {
        return new BookShelfIterator(this);
    }
}
```

运行结果：



2、针对Iterator的例子，设计一个Specified的Iterator并运行。

实现方式：

本例编写一个只迭代包含特定关键词的书籍名称的迭代器。

为迭代器添加一个keyword字段，每次迭代通过while循环找到下一个含该关键字的书籍再返回。

```
public class MyBookShelfIterator implements Iterator {
    private BookShelf bookShelf;
    private String keyword;
    private int index;

    public MyBookShelfIterator(BookShelf bookShelf) {
        this.bookShelf = bookShelf;
        this.keyword = "o";
        this.index = 0;
    }

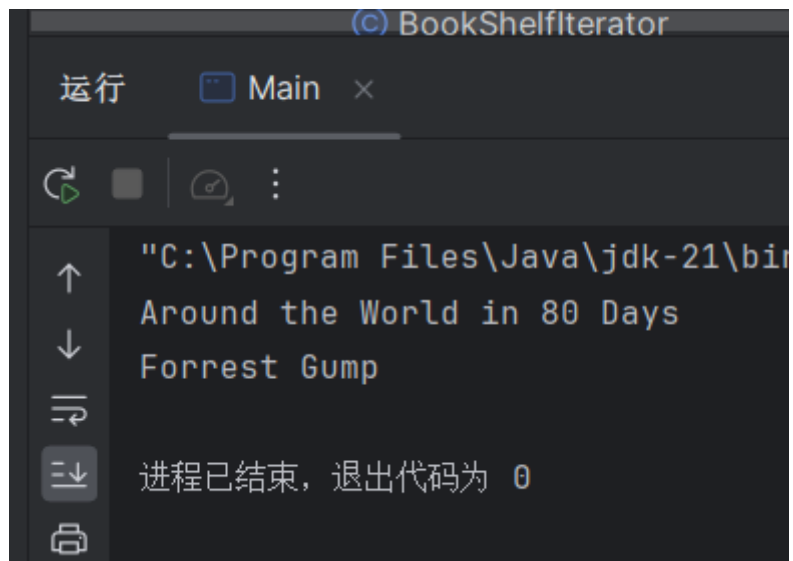
    public boolean hasNext() {
        while (index < bookShelf.getLength()) {
            if (bookShelf.getBookAt(index).getName().contains(keyword)) {
                return true;
            }
            index++;
        }
        return false;
    }

    public Object next() {
        while (index < bookShelf.getLength()) {
            Book book = bookShelf.getBookAt(index);
            index++;
            if (book.getName().contains(keyword)) {
                return book;
            }
        }
        return false;
    }
}
```

```
}
```

运行结果：

成功返回书籍名字中含“o”的书籍



```
"C:\Program Files\Java\jdk-21\bin
Around the World in 80 Days
Forrest Gump
进程已结束，退出代码为 0
```