****

《数据库原理》课程

实验报告

学 号 09019204

姓 名 曹邹颖

东南大学计算机科学与工程学院

二0 二1 年 12 月

**目录**

**[一、 场景介绍 3](#_Toc13814)**

**[二、 数据库设计 3](#_Toc3004)**

[1. 虚拟图书馆系统的数据流图 3](#_Toc17928)

[2. 虚拟图书馆系统的E-R图 4](#_Toc16954)

[3. 虚拟图书馆系统的数据模式 5](#_Toc29336)

[①建立数据库library 5](#_Toc16399)

[②用户信息表tb\_user 5](#_Toc5027)

[③图书信息表tb\_bookinfo 5](#_Toc7660)

[④借阅信息表tb\_borrowinfo 5](#_Toc1334)

[⑤读者信息表tb\_readerinfo 6](#_Toc15704)

[⑥触发器功能 6](#_Toc2465)

**[三、 核心代码 8](#_Toc22223)**

[1. 数据库连接类DbUtil 8](#_Toc3048)

[2. 对用户信息表的数据库操作 9](#_Toc20718)

[3. 对图书信息表的数据库操作 9](#_Toc4679)

[4. 对借阅信息表的数据库操作 10](#_Toc7478)

[5. 对读者信息表的数据库操作 12](#_Toc4578)

**[四、 实验成效 13](#_Toc21578)**

[1. 登录界面（涉及数据库的用户信息表) 13](#_Toc3046)

[2. 功能界面 14](#_Toc22127)

[(1) 管理员特有功能：图书管理、读者信息查询 14](#_Toc21665)

[(2) 读者特有功能：修改个人联系方式 16](#_Toc30387)

[(3) 共有功能 17](#_Toc13515)

[①个人用户大厅 17](#_Toc6471)

[②修改密码功能 17](#_Toc11387)

[③修改用户个性信息功能 17](#_Toc6677)

[④查询借阅信息功能 18](#_Toc24597)

[⑤还书、续借功能 18](#_Toc5214)

[⑥书籍信息查询功能 19](#_Toc3224)

[⑦借书功能 19](#_Toc11485)

[⑧最受欢迎的top10图书信息显示、博览群书的top10读者信息显示 20](#_Toc6308)

**[五、 实验小结 20](#_Toc31033)**

# 场景介绍

随着智能手机的普及，虚拟图书馆这种模式能更加贴近师生的生活习惯，并且适合师生年龄段的习性，这对师生更多的接触到图书馆的信息有一定得促进作用。一套功能齐全的校内的**虚拟图书馆系统**，使其具有一般图书管理系统所具有的功能，可以实现管理员对图书信息的增删改查功能、对所有读者信息的查阅，也帮助师生读者书籍查询、书籍借阅与归还、续借等功能，方便师生搜索书籍、借阅图书。

1. **数据库设计**
2. **虚拟图书馆系统的数据流图：**

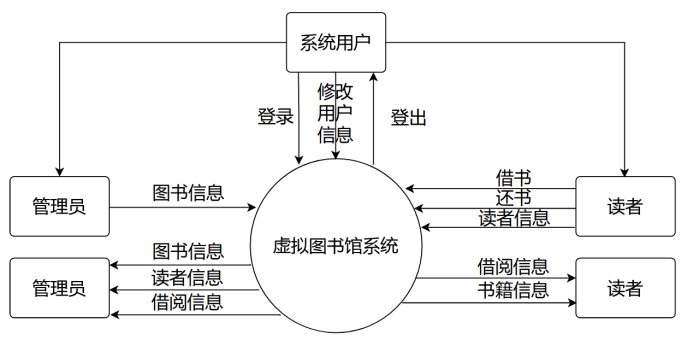


图1. 顶层数据流图

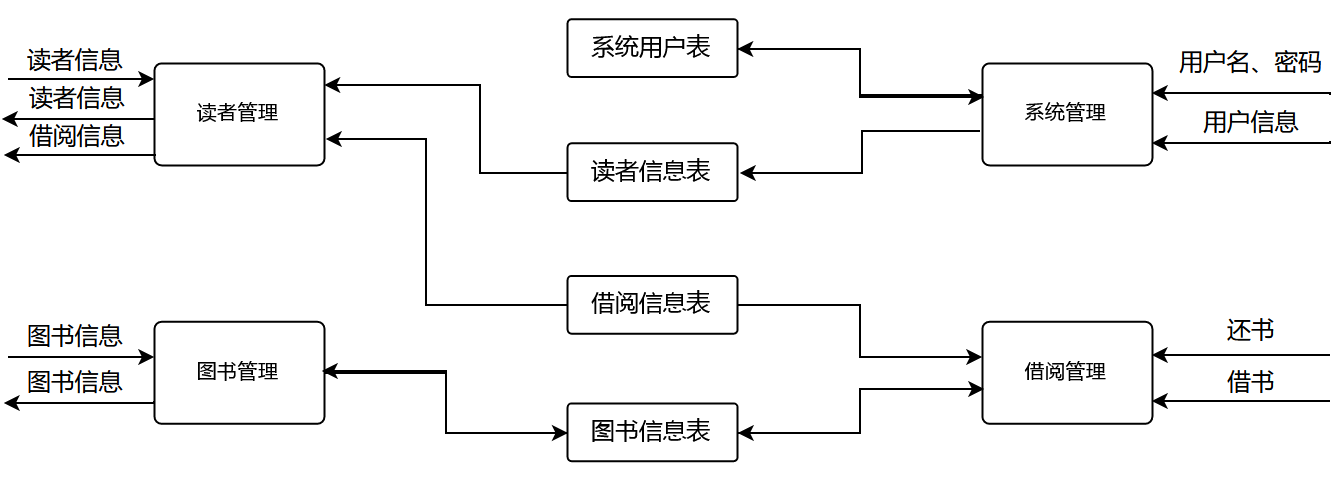


图2. 0层数据流图

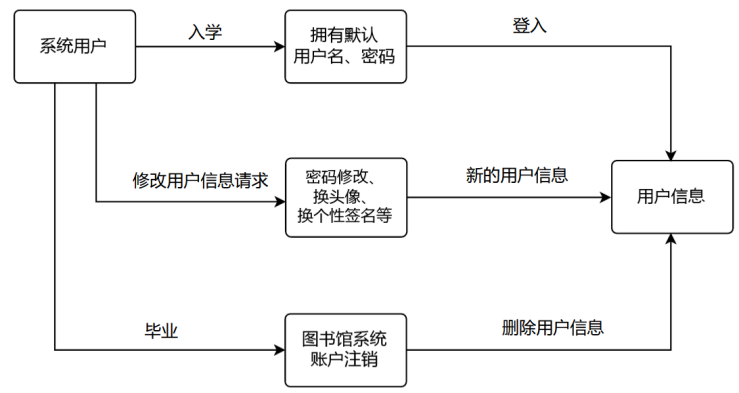


图3.1 1层数据流图之系统管理

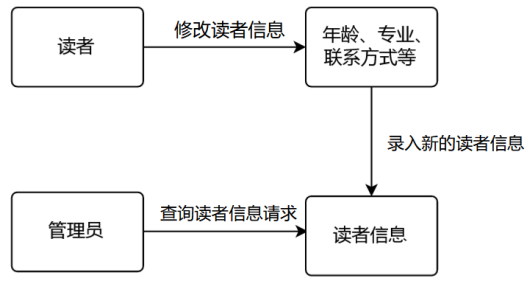


图3.2 1层数据流图之读者管理

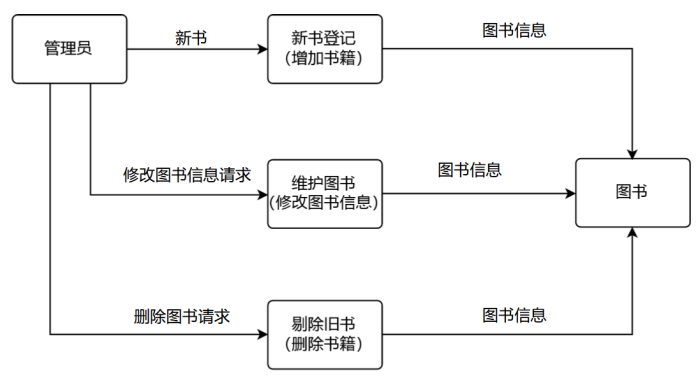


图3.3 1层数据流图之图书管理

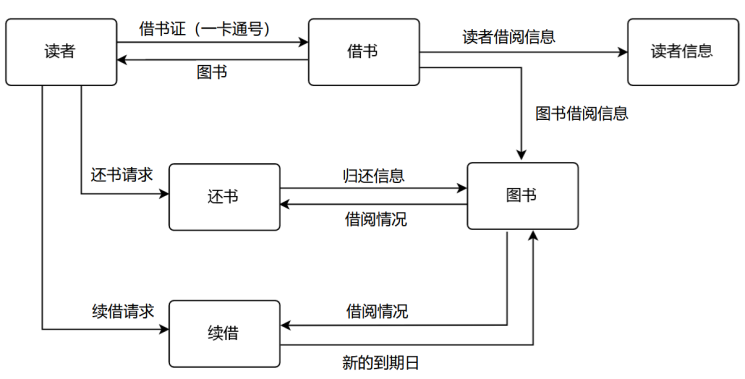
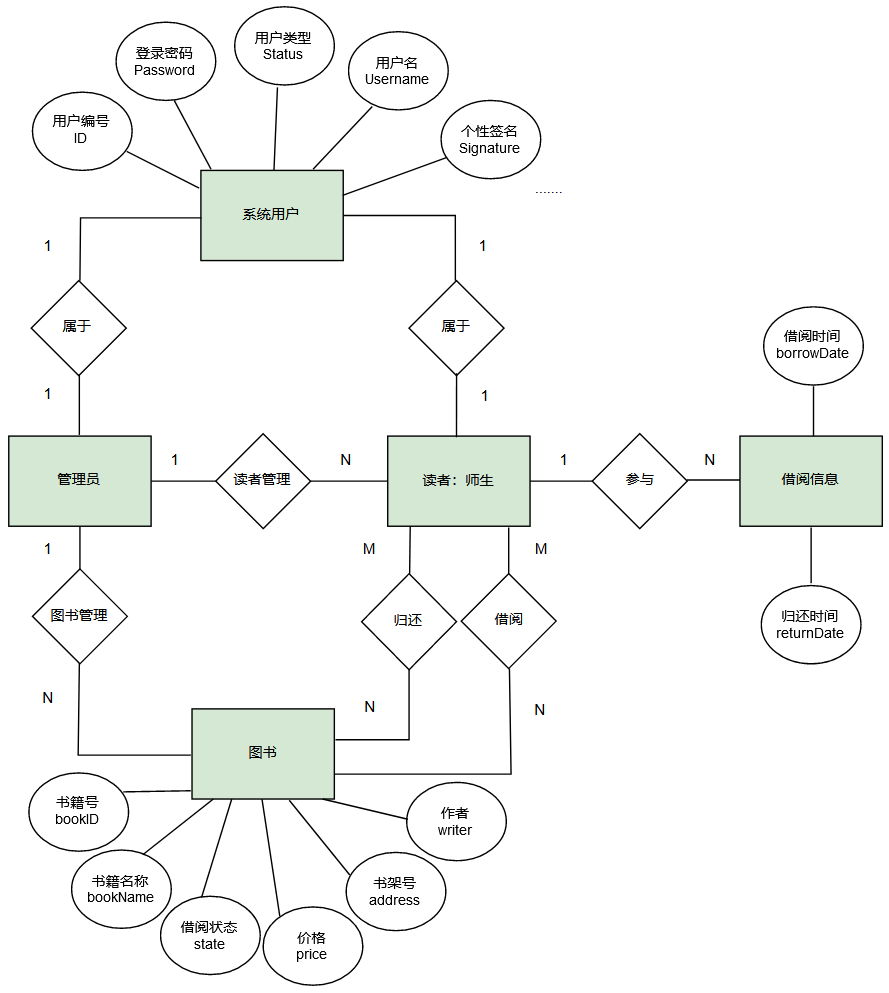


图3.4 1层数据流图之借阅管理

1. **虚拟图书馆系统的E-R图：**



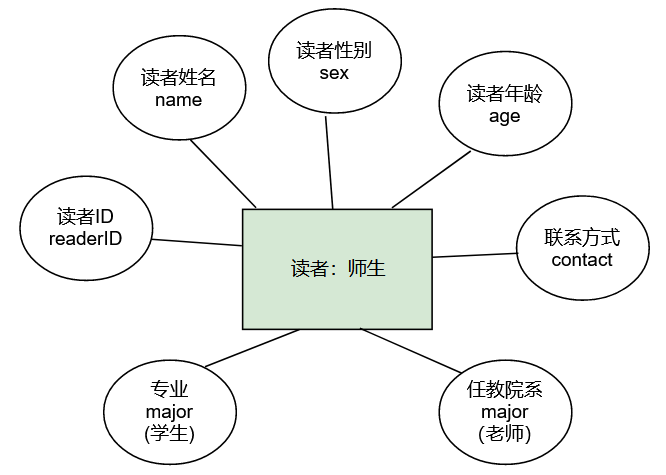


图4. 虚拟图书馆系统E-R图

1. **虚拟图书馆系统的数据模式：**

**①建立数据库library：**

CREATE SCHEMA `library` DEFAULT CHARACTER SET utf8 COLLATE utf8\_bin ;

**②用户信息表tb\_user：**

CREATE TABLE `library`.`tb\_user` (

`ID` VARCHAR(255) NOT NULL,

`Password` VARCHAR(255) NOT NULL,

`status` INT NULL DEFAULT 2,

`Username` VARCHAR(45) NULL DEFAULT NULL,

`Signature` VARCHAR(45) NULL DEFAULT NULL,

`Selfinfo` VARCHAR(45) NULL DEFAULT NULL,

`Background` VARCHAR(45) NULL DEFAULT NULL,

`Headpic` BLOB NULL DEFAULT NULL,

PRIMARY KEY (`ID`));

型：

值：

系统录入一个管理员：INSERT INTO `library`.`tb\_user` (`ID`, `Password`, `status`) VALUES ('000123456', 'guanliyuan', '0');

**③图书信息表tb\_bookinfo：**

CREATE TABLE `library`.`tb\_bookinfo` (

`bookID` VARCHAR(45) NOT NULL,

`bookName` VARCHAR(45) NULL DEFAULT NULL,

`state` TINYINT NULL DEFAULT 0,

`price` VARCHAR(45) NULL DEFAULT NULL,

`writer` VARCHAR(45) NULL DEFAULT NULL,

`address` VARCHAR(45) NULL DEFAULT NULL,

PRIMARY KEY (`bookID`));

型：

值：

**④借阅信息表tb\_borrowinfo：**

CREATE TABLE `library`.`tb\_borrowinfo` (

`bookID` VARCHAR(45) NOT NULL,

`borrower` VARCHAR(45) NULL DEFAULT '000123456',

`borrowDate` TIMESTAMP NULL DEFAULT CURRENT\_TIMESTAMP,

`returnDate` TIMESTAMP NULL DEFAULT NULL,

PRIMARY KEY (`bookID`, `borrower`, `borrowDate`, `returnDate`));

型：

值：

**⑤读者信息表tb\_readerinfo：**

CREATE TABLE `library`.`tb\_readerinfo` (

`readerID` VARCHAR(255) NOT NULL,

`sno` VARCHAR(45) NULL DEFAULT NULL,

`name` VARCHAR(45) NULL DEFAULT NULL,

`sex` VARCHAR(45) NULL DEFAULT '男',

`age` INT NULL DEFAULT NULL,

`major` VARCHAR(45) NULL DEFAULT NULL,

`contact` VARCHAR(45) NULL DEFAULT NULL,

PRIMARY KEY (`readerID`));

型：

值：

**⑥触发器功能：**

1. **对于添加用户信息后添加相应的读者信息**

DROP TRIGGER IF EXISTS `library`.`tb\_user\_after\_insert`;

DELIMITER $$

USE `library`$$

CREATE DEFINER = CURRENT\_USER TRIGGER `library`.`tb\_user\_after\_insert`

AFTER INSERT ON `tb\_user` FOR EACH ROW

BEGIN

INSERT INTO tb\_readerinfo(readerID) VALUES (new.ID);

END$$

DELIMITER ;

1. **对于删除用户信息成功后删除相应的读者信息和借阅信息**

DROP TRIGGER IF EXISTS `library`.`tb\_user\_AFTER\_DELETE`;

DELIMITER $$

USE `library`$$

CREATE DEFINER = CURRENT\_USER TRIGGER `library`.`tb\_user\_AFTER\_DELETE` AFTER DELETE ON `tb\_user` FOR EACH ROW

BEGIN

delete from tb\_readerinfo where readerID=old.ID;

delete from tb\_borrowrinfo where readerID=old.ID;

END$$

DELIMITER ;

1. **对于增加借阅信息（借书）后修改bookID对应的图书状态state=1(表示已借阅)**

DROP TRIGGER IF EXISTS `library`.`tb\_borrowinfo\_after\_insert`;

DErary`$LIMITER $$

USE `lib$

CREATE DEFINER = CURRENT\_USER TRIGGER `library`.`tb\_borrowinfo\_after\_insert`

AFTER INSERT ON `tb\_borrowinfo` FOR EACH ROW

BEGIN

update tb\_bookinfo set state=1 where bookID=new.bookID;

END$$

DELIMITER ;

1. **对于修改借阅信息（还书）后修改bookID对应的图书状态state=0(表示已归还)**

DROP TRIGGER IF EXISTS `library`.`tb\_borrowinfo\_after\_update`;

DELIMITER $$

USE `library`$$

CREATE DEFINER = CURRENT\_USER TRIGGER `library`.`tb\_borrowinfo\_after\_update` AFTER UPDATE ON `tb\_borrowinfo` FOR EACH ROW

BEGIN

if not (new.returnDate <=> old.returnDate) then

update tb\_bookinfo set state=0 where bookID=new.bookID;

end if;

END$$

DELIMITER ;

由于MySQL中不支持触发器中使用rollback，所以这里呈现的两个触发器功能采用sql server 创建触发器时的语言。

1. **对于删除用户信息前检查是否存在对应借阅信息（即该用户还有未归还的图书），若存在，则不予删除该用户信息（rollback）**

create trigger tb\_user\_before\_delete on tb\_user

instead of delete

as

if not exists(select \* from tb\_borrowinfo, deleted where tb\_borrowinfo.borrower = deleted.ID and tb\_borrowinfo.returnDate>GETDATE())

delete from tb\_user where tb\_user.ID in (select ID from deleted)

1. **对于删除图书信息前检查是否存在对应借阅信息（已借阅但未归还），若存在，则不予删除（rollback）**

create trigger tb\_bookinfo\_before\_delete on tb\_bookinfo

after delete

as

if exists(select \* from tb\_borrowinfo, deleted where tb\_borrowinfo.bookID = deleted.bookID and tb\_borrowinfo.returnDate>GETDATE())

begin

rollback transaction

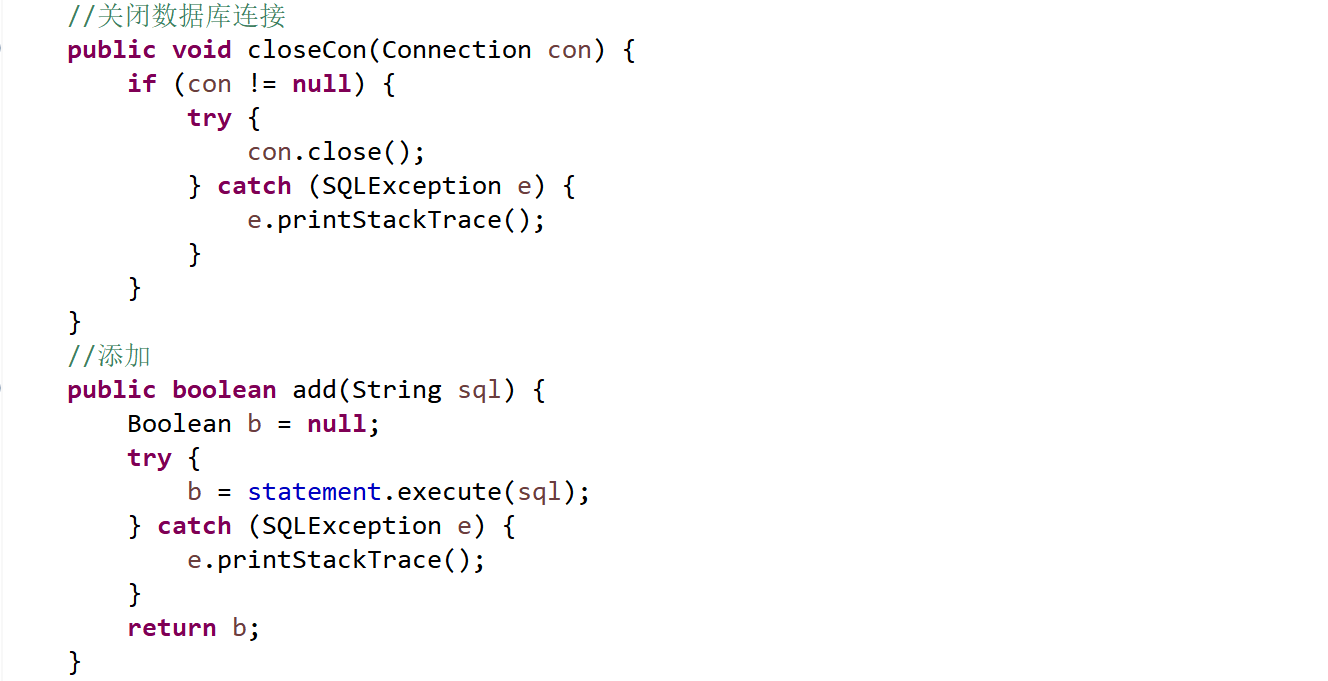
end

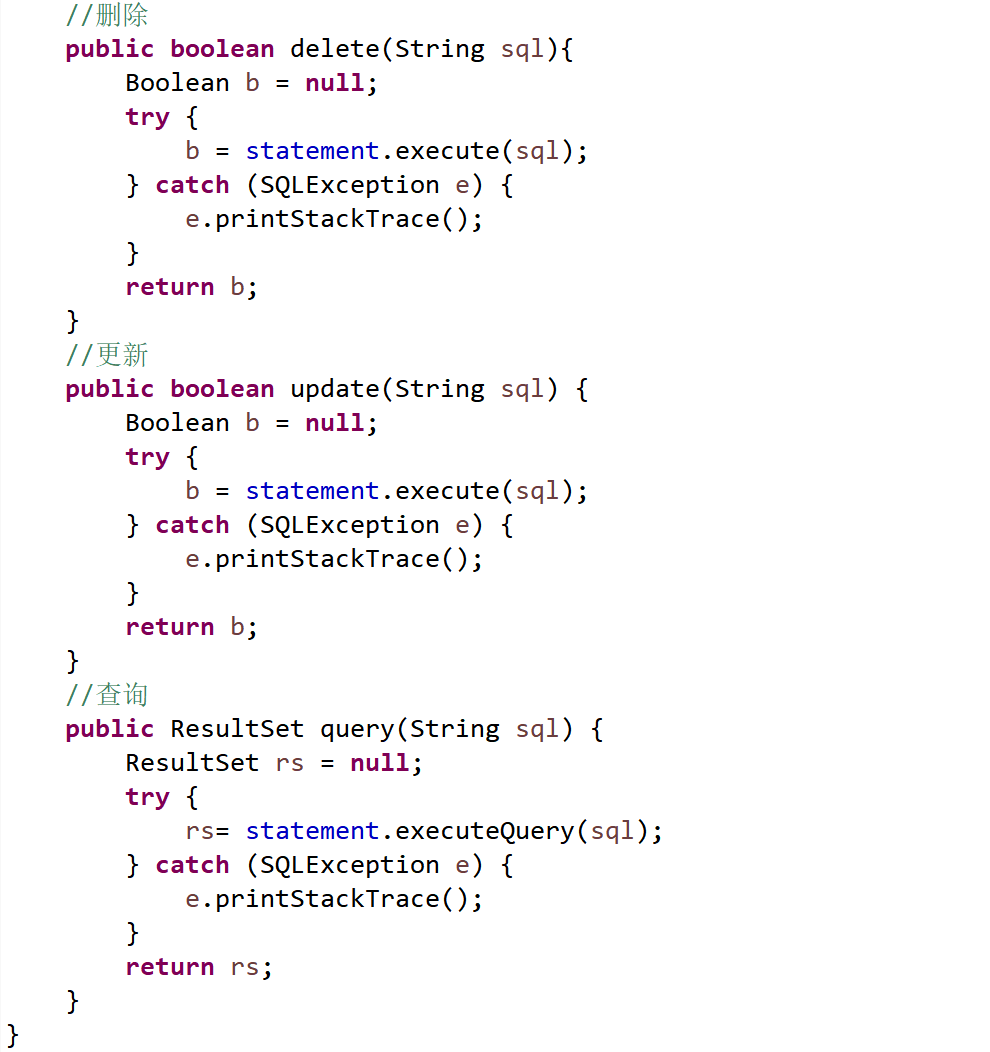
1. **核心代码**
2. **数据库连接类DbUtil**

定义一个数据库连接类DbUtil：

DbUtil类设计在于将Java中嵌入式SQL语言封装起来，使得调用数据库操作方便，尽管增删改都对应execute()函数，但封装成不同函数目的在于增加代码可读性。

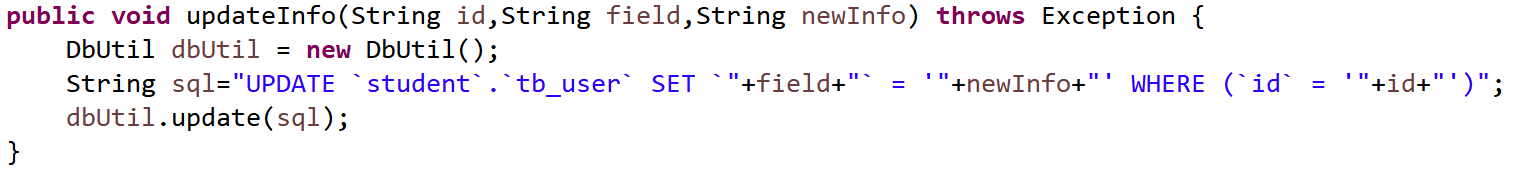






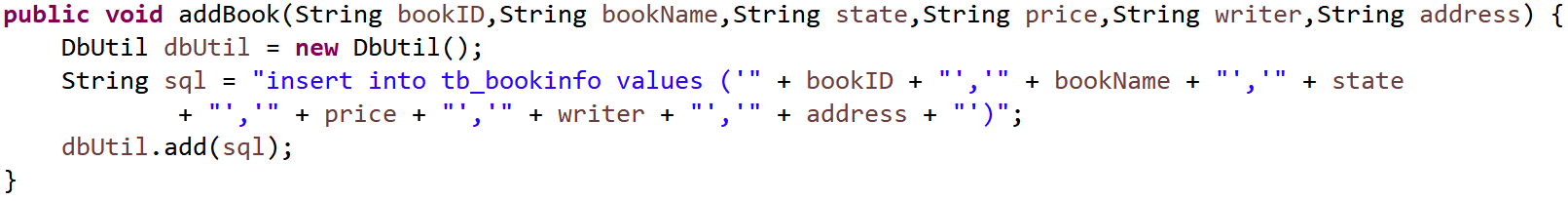
1. **对用户信息表的数据库操作**

**按Field名称修改相应用户信息(密码、个性签名等)：**

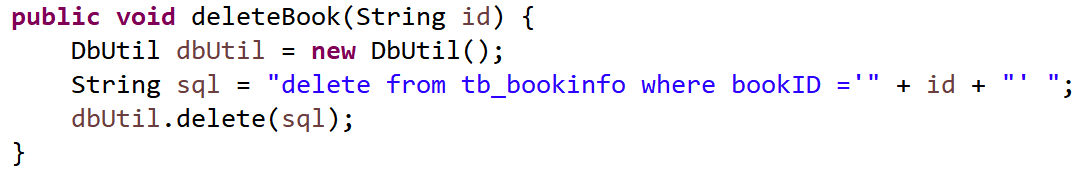


1. **对图书信息表的数据库操作**

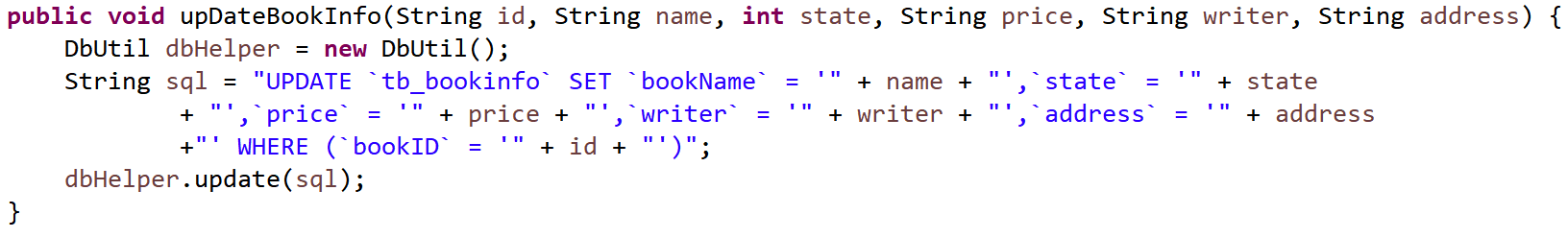
**添加书籍信息：**(**用于管理员**)



**删除书籍信息：(用于管理员)**

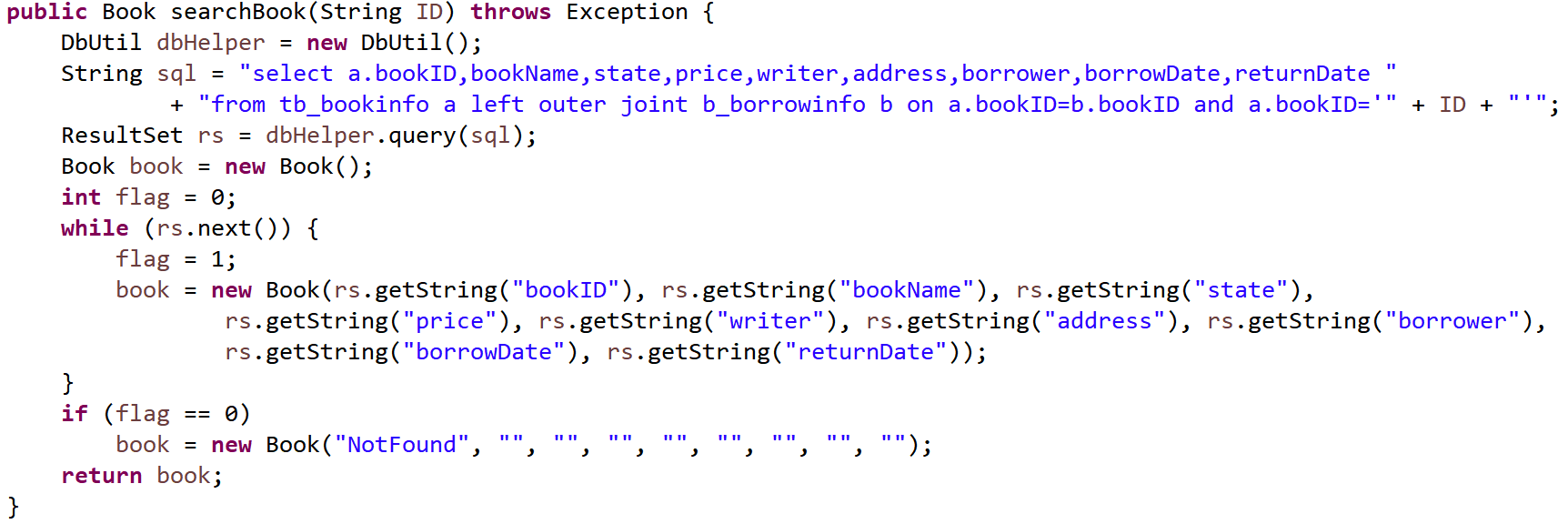


**更新书籍信息：(用于管理员)**

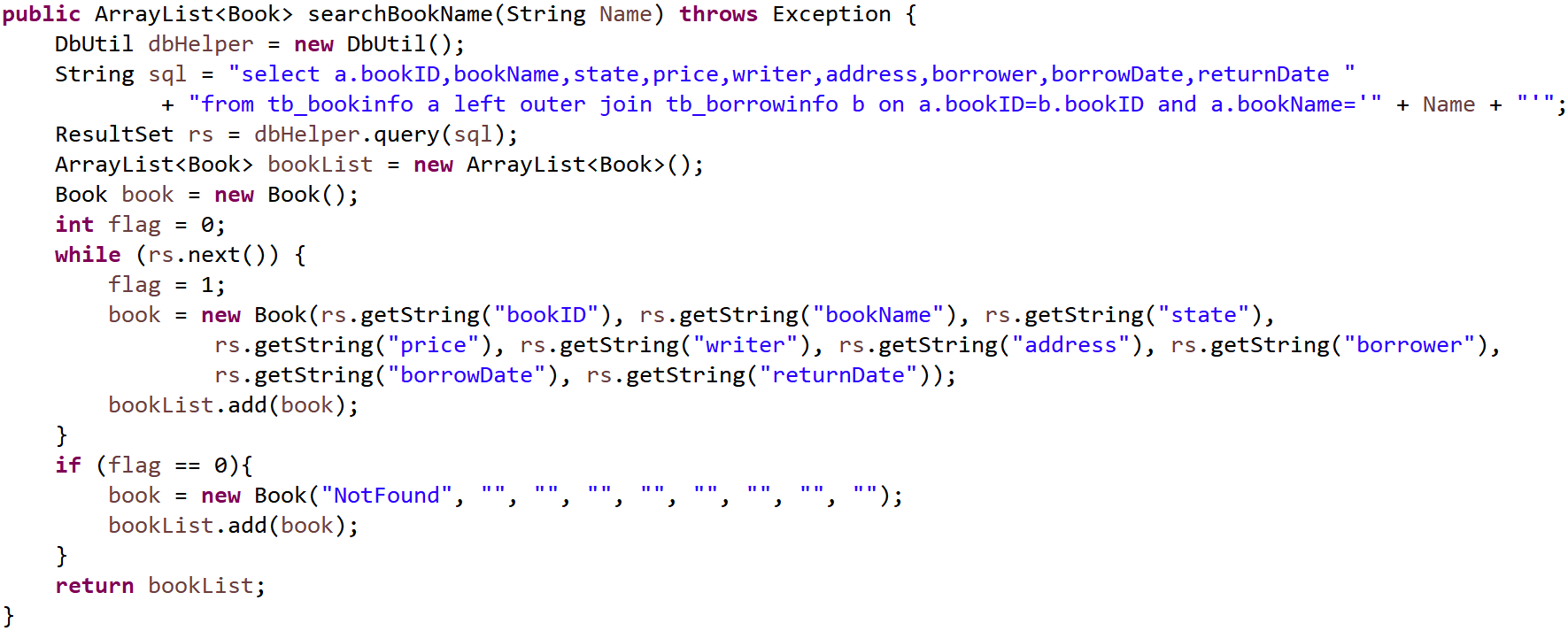


**查询书籍信息：**

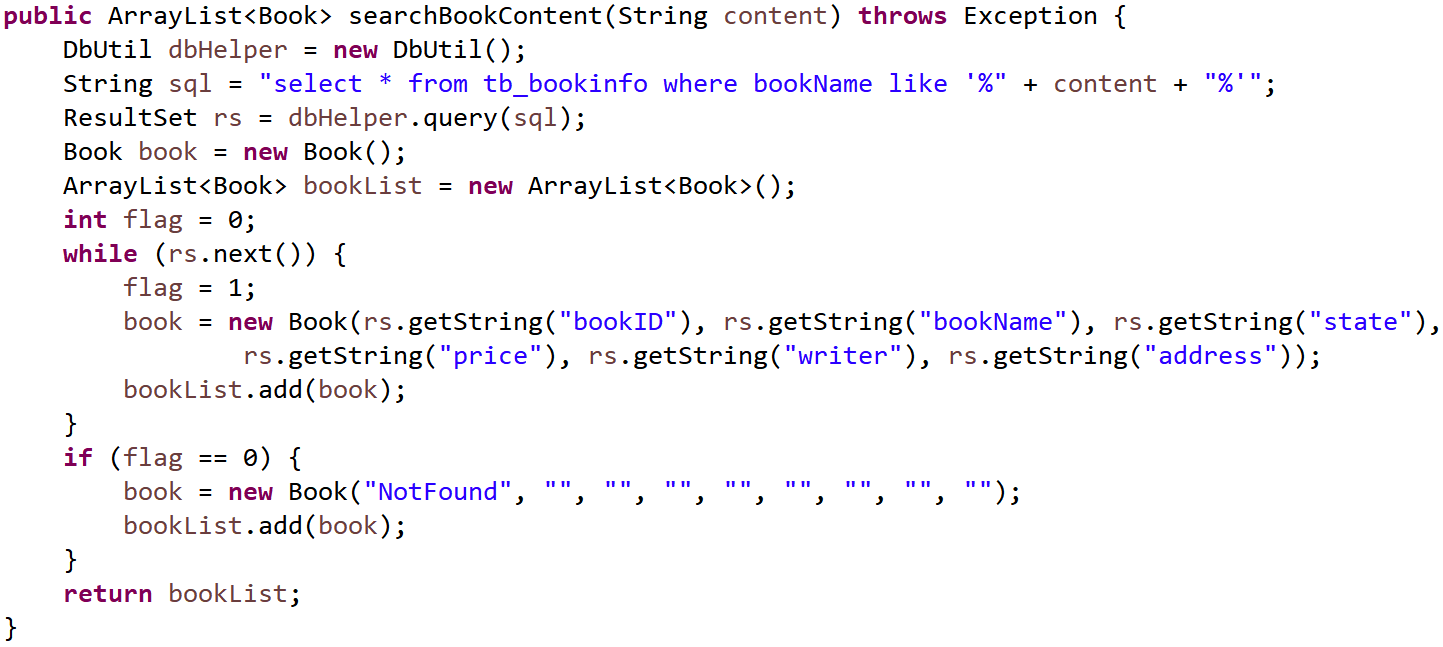
①按照book ID查询：(左连接∵可能没有借阅信息)(用于管理员查询图书信息)



②按照bookName查询：(左连接∵可能没有借阅信息)

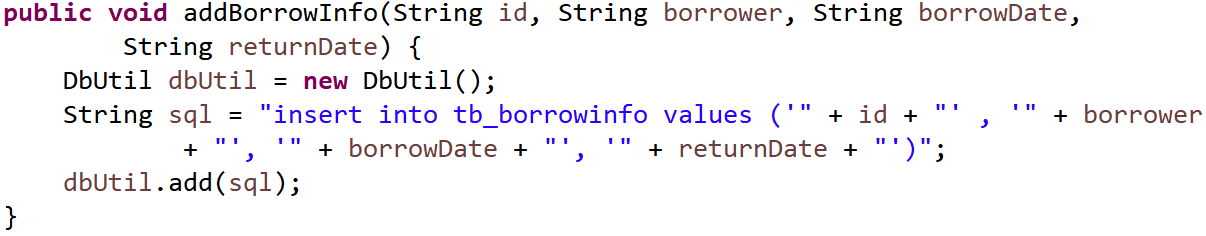


③按照bookName模糊查询：(用于读者查询图书信息)

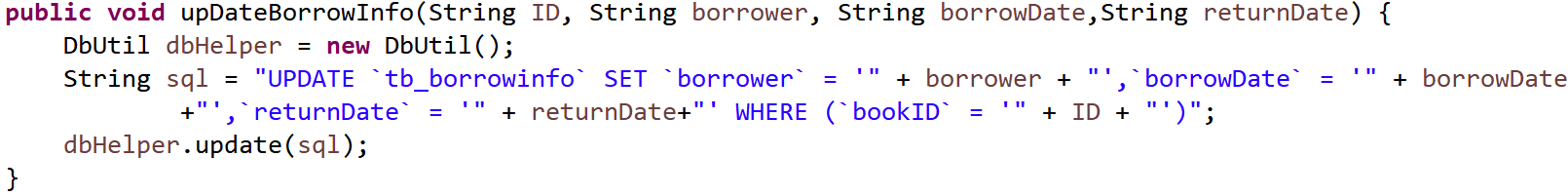


1. **对借阅信息表的数据库操作：**

**添加借阅信息：（用于借书）**



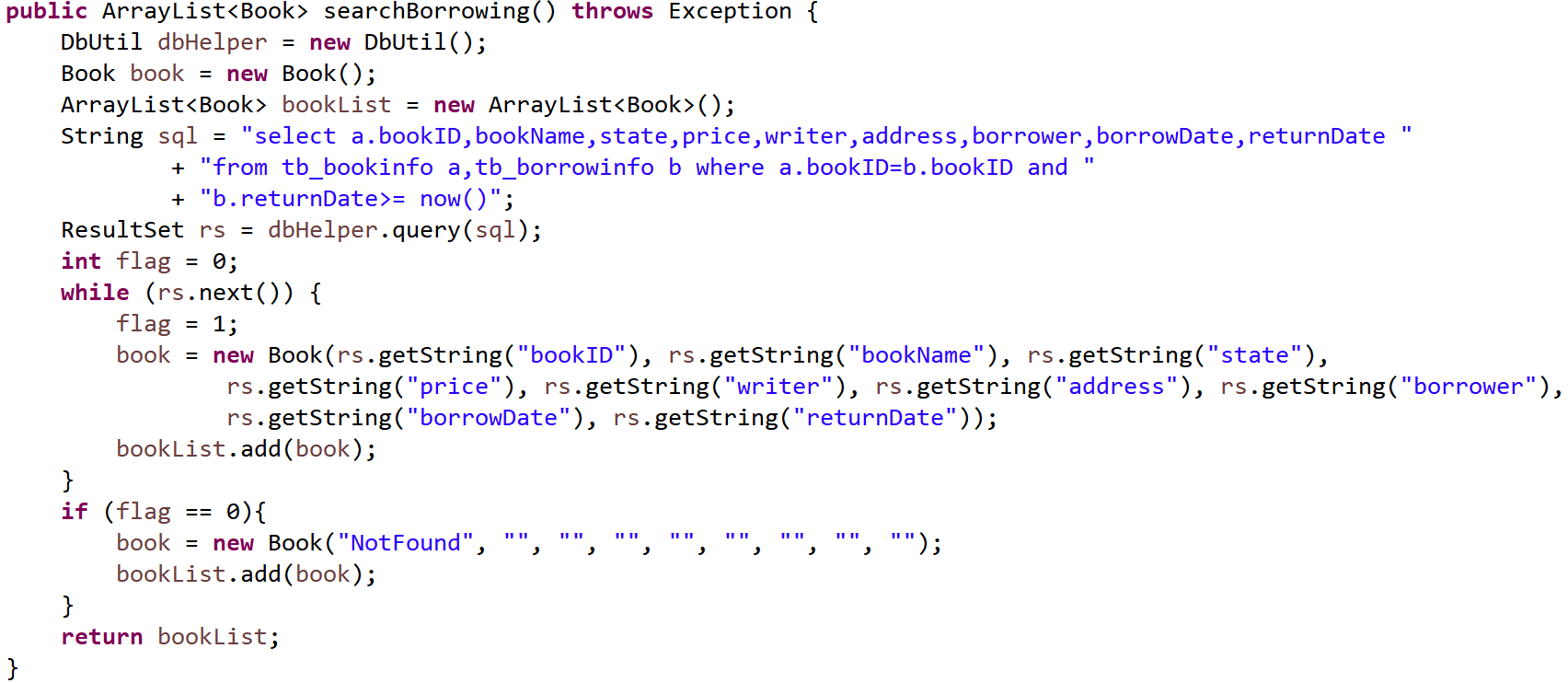
**修改借阅信息：（用于续借、还书）**



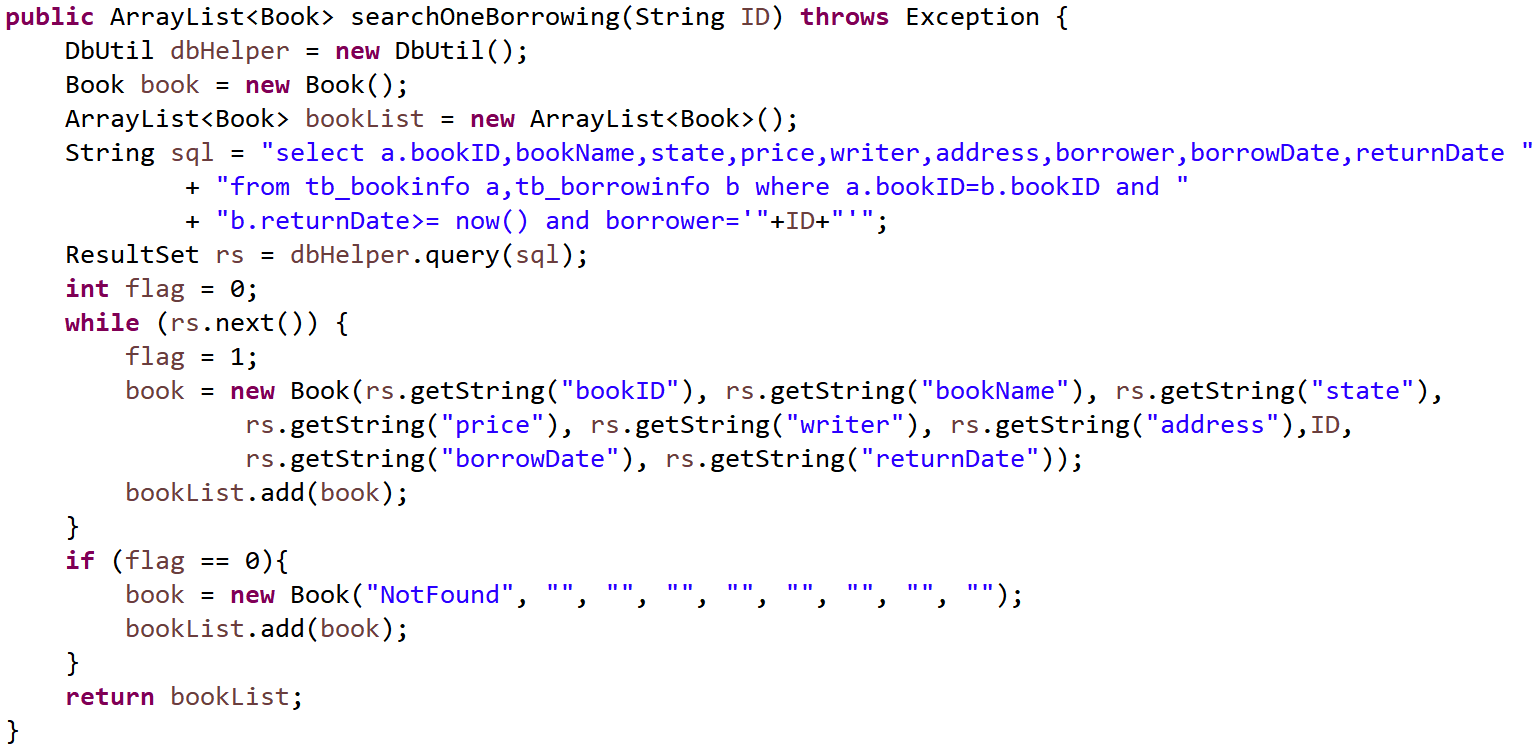
**查询借阅信息：**

①查询已借阅待归还的图书信息：

无特定借阅者：（用于管理员）

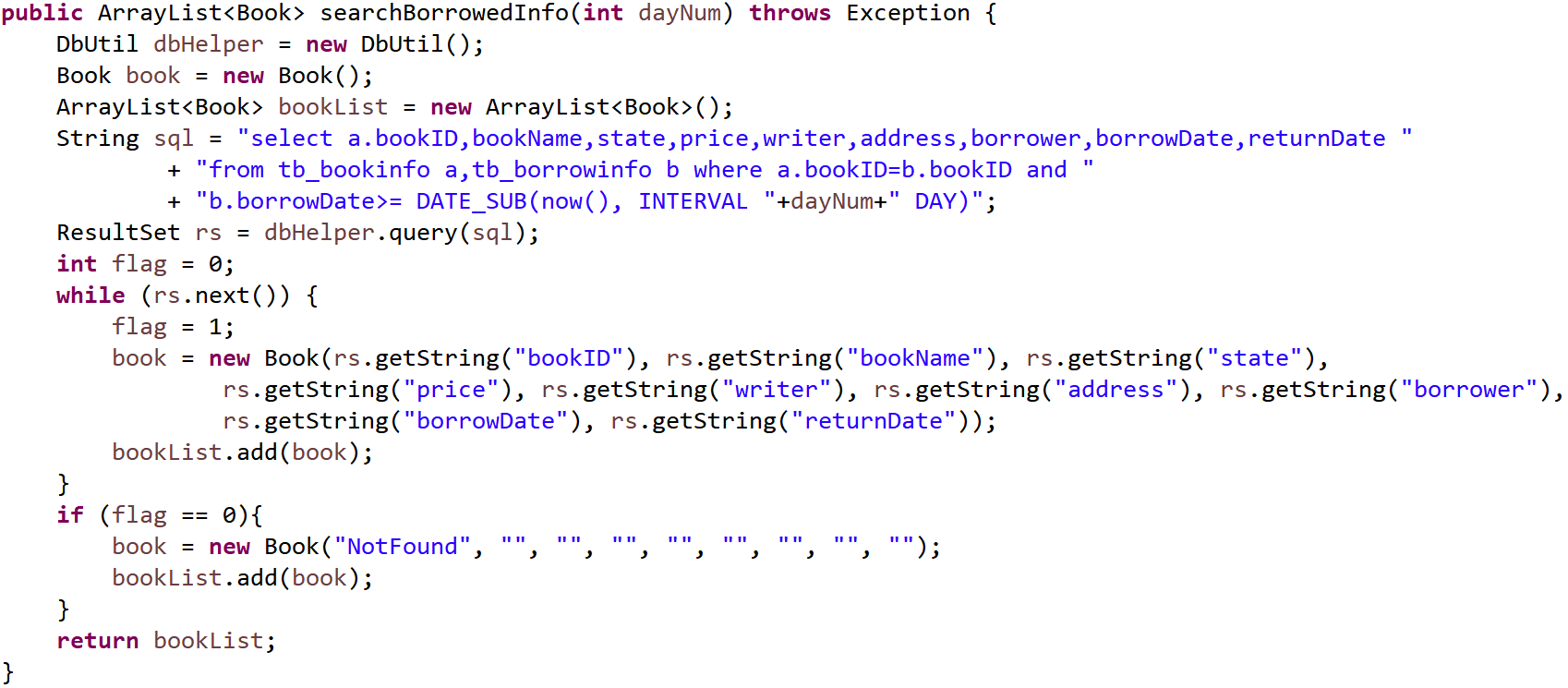


特定借阅者：（通用：读者只能特指自己，管理员可以指定借阅者）

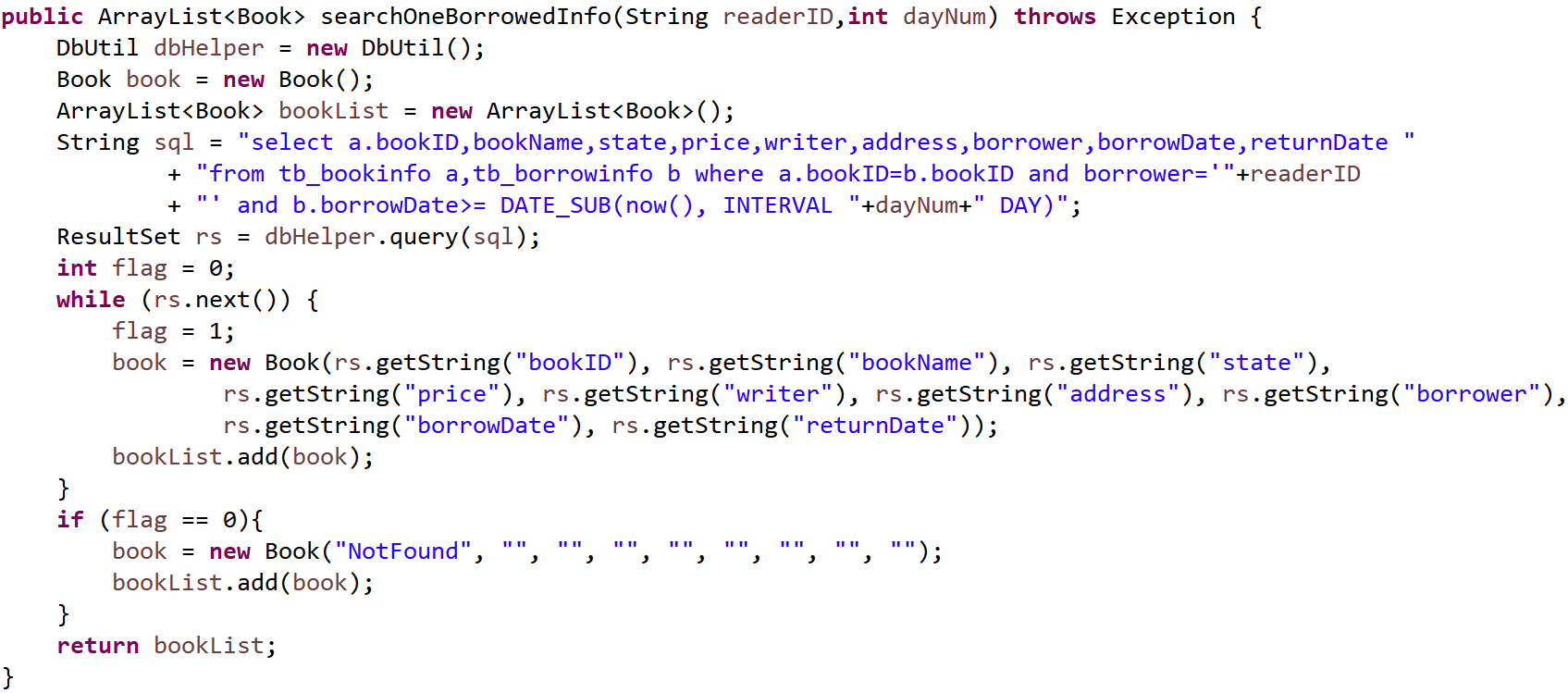


②查询dayNum天内的所有借阅信息：

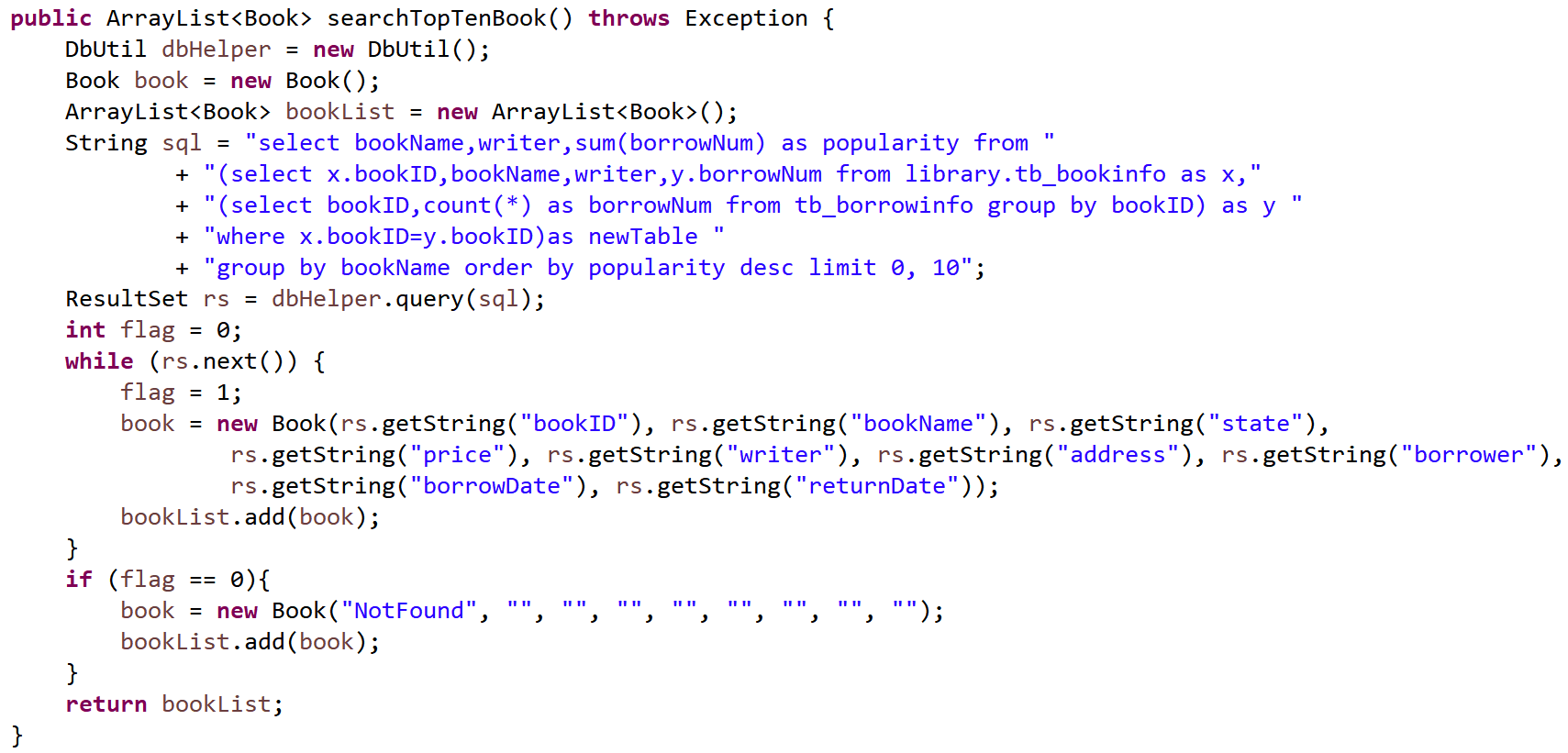
无特定借阅者：（用于管理员）



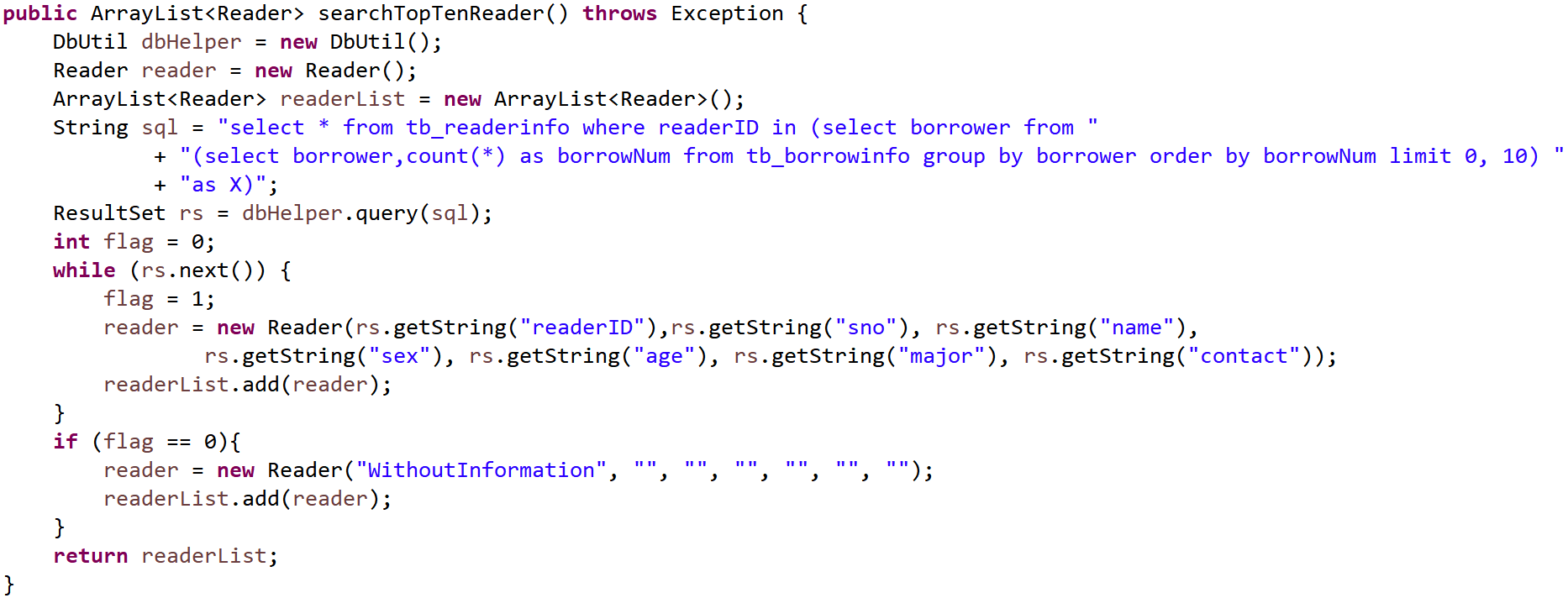
特定借阅者：（通用：读者只能特指自己，管理员可以指定借阅者）



③查询最受欢迎的top 10图书信息：（即借阅次数最多的前10本书籍）

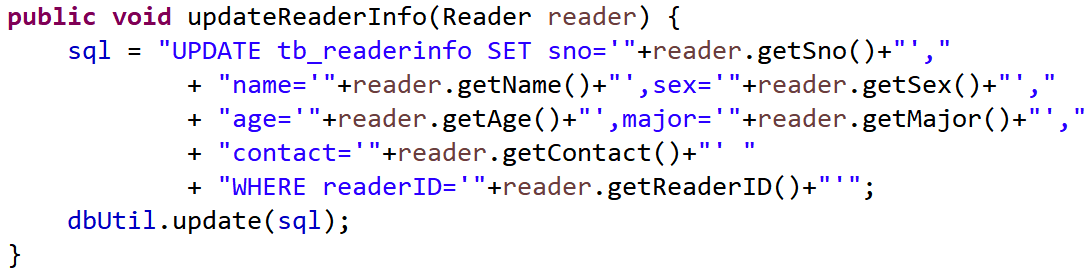


④查询博览群书的top 10读者信息：（即借阅次数最多的前10位读者）



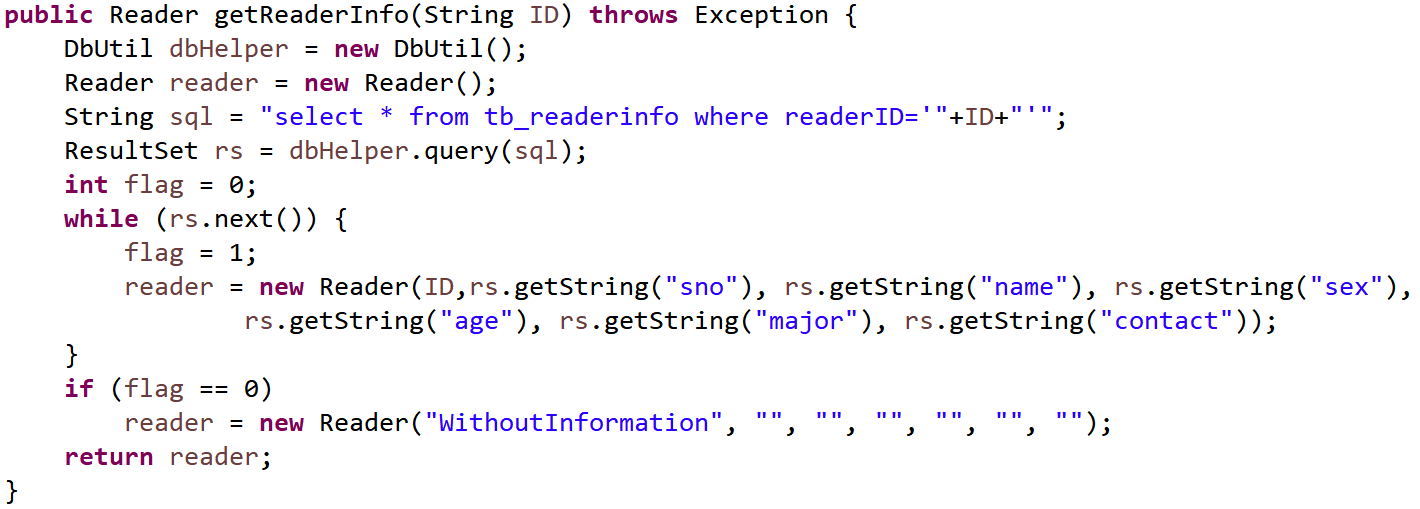
1. **对读者信息表的数据库操作：**

**修改读者信息：**(用于读者)

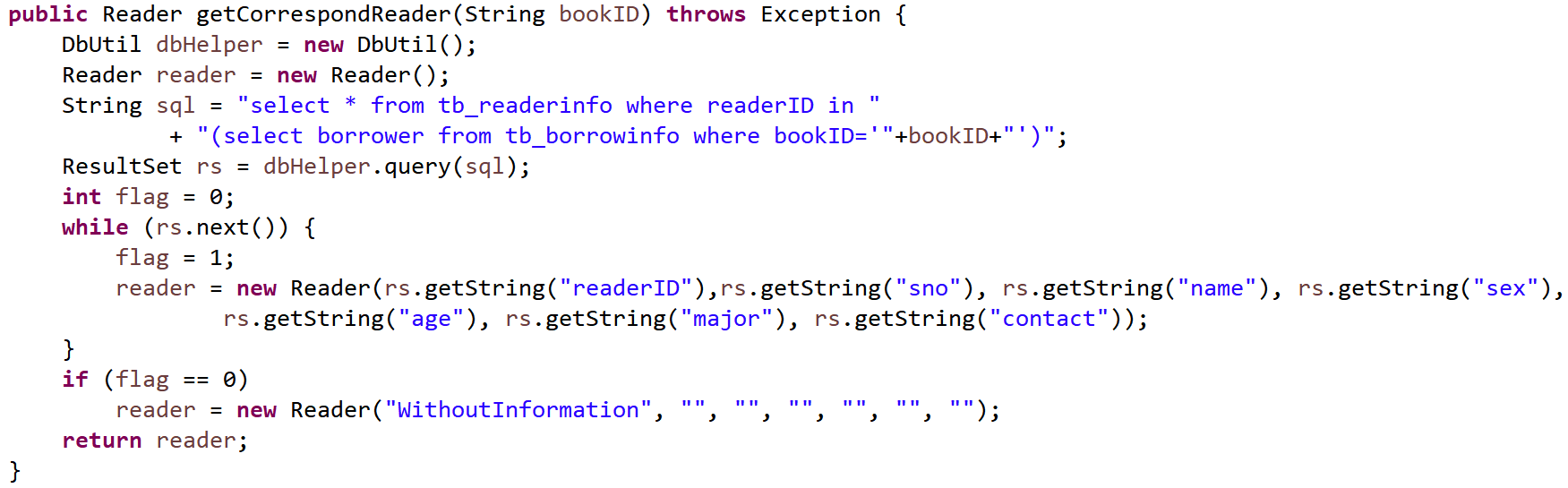


**查询读者信息：**

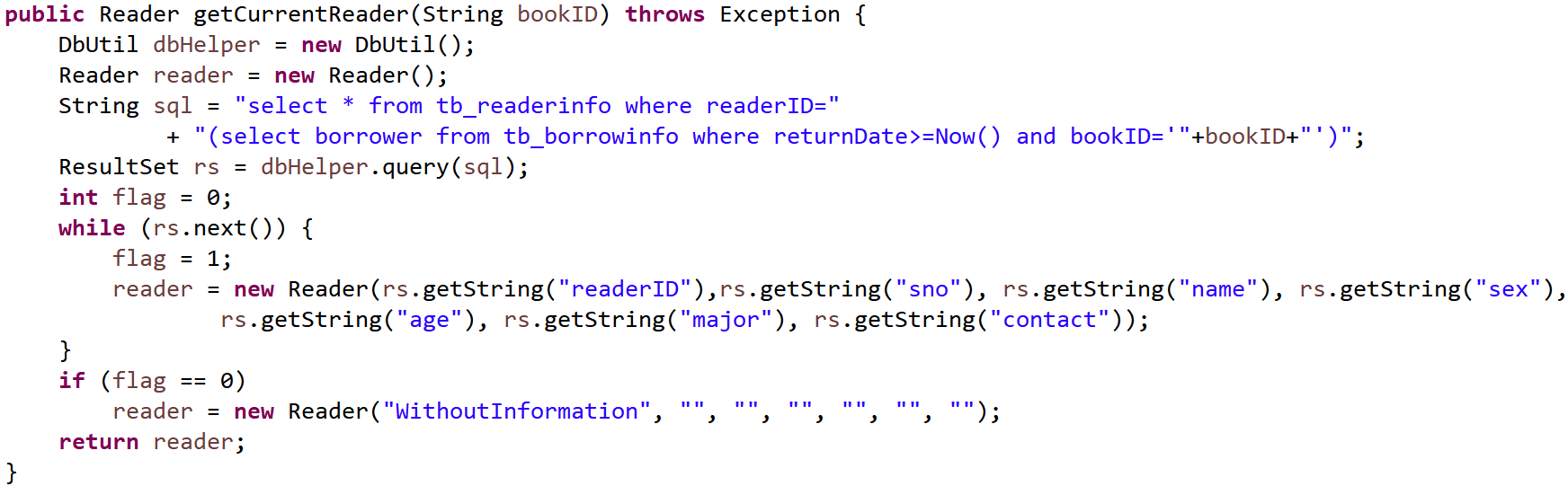
①读者查询自己信息：



②管理员查询某书号所有借阅者的信息：



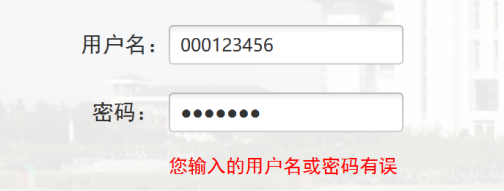
③管理员查询某书号当前借阅者的信息：



1. **实验成效**
2. **登录界面**（涉及数据库的用户信息表)

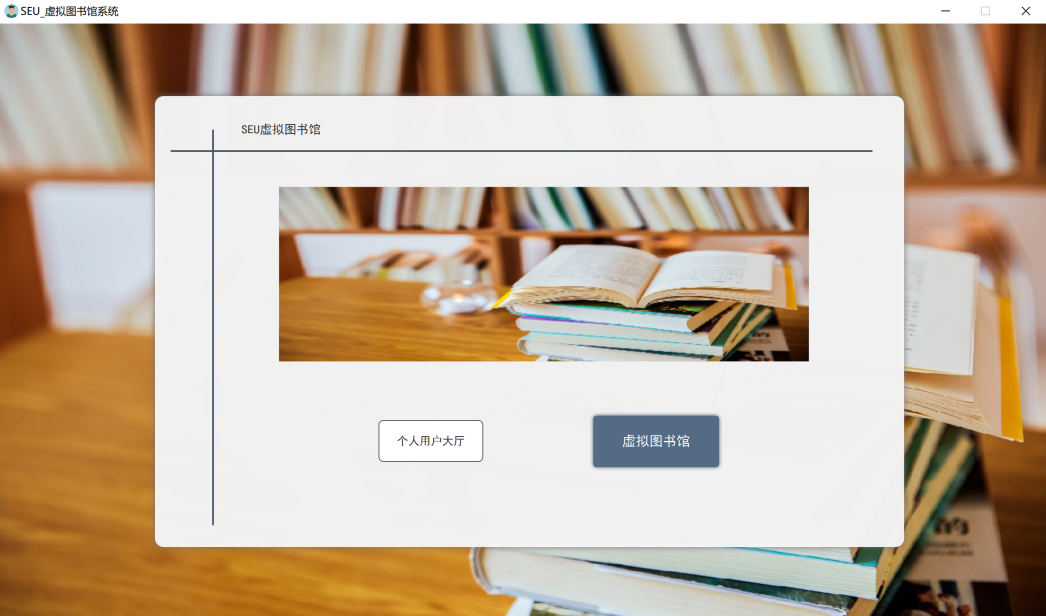


若密码与用户名不匹配会报错：



1. **功能界面**

选择进入个人用户大厅还是虚拟图书馆：



1. **管理员特有功能：**图书管理、读者信息查询
2. **图书管理：(增删改图书信息)**



当前图书信息展示如上，使用添加图书功能如下：

点击某一条图书信息修改按钮，便可在其任意信息框中修改，

例如，修改3404708书号的图书位置为“基础书库”：

修改后结果如下，同时若点击删除按钮便可将这本书籍从图书馆系统中删除。



删除后结果：



②读者信息查询：



1. **读者特有功能：修改个人联系方式**



修改结果如下：



1. **共有功能**

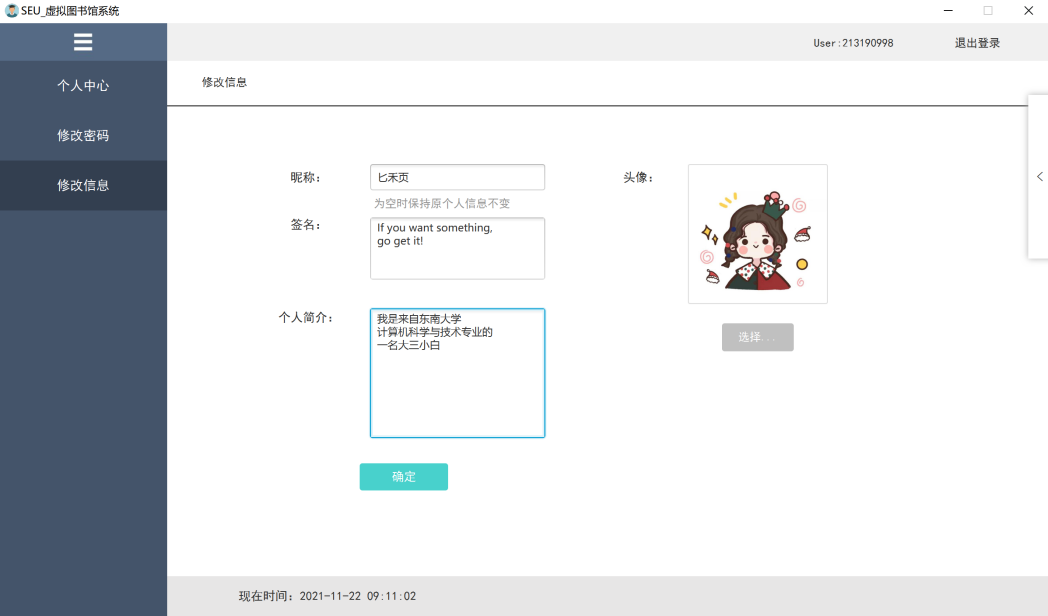
①个人用户大厅（涉及数据库的用户信息表，根据登录的用户ID会查询到TA的用户 名、系统头像、个人简介等显示）：



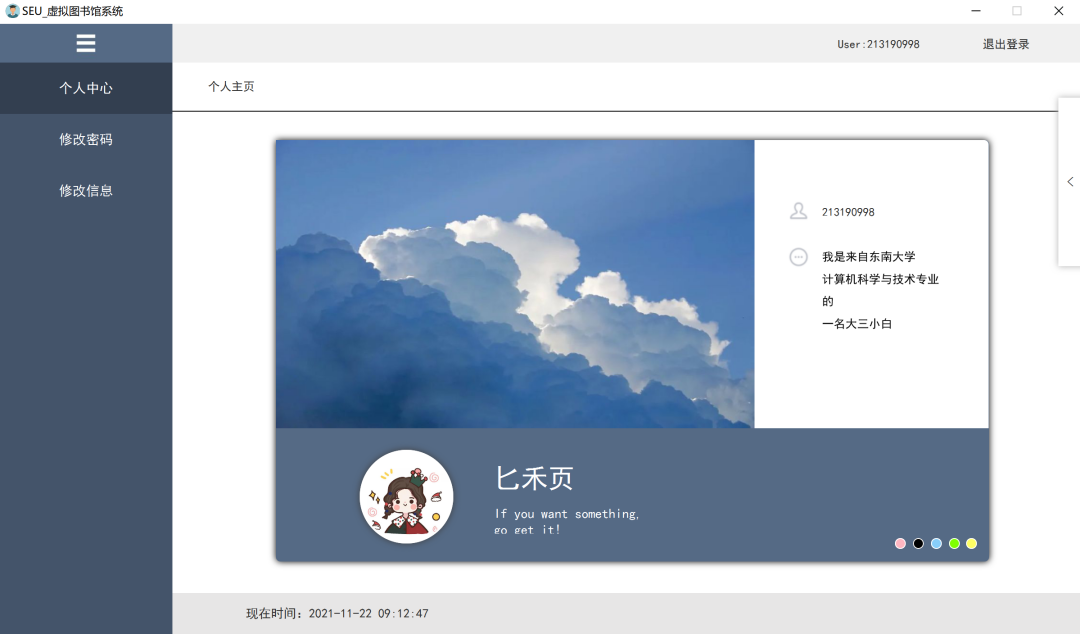
②修改密码功能：(修改完成后会需要重新登陆)



③修改用户个性信息功能：



返回个人中心便可看到个性信息的修改：



④查询借阅信息功能：



⑤还书、续借功能：



⑥书籍信息查询功能：（根据book ID准确查找、bookName模糊查找）



⑦借书功能：



点击借阅后，查看个人借阅信息界面：



⑧最受欢迎的top 10图书信息显示、博览群书的top 10读者信息显示：



1. **实验小结**

在这次数据库课程实验中，又一次运用到Java、MySQL数据库、Java FX，与暑期学校的Java实训项目不同的是，这一次完成这个虚拟图书馆系统的设计在数据库方面增加了复杂性。

对于MySQL数据库，暑期学校从零开始学习时很困难，之后是靠MySQL Workbench可视化的界面上手的，然而这次系统学习了数据库原理课程后，就是直接使用MySQL中的SQL语句去构建数据库、并在工作区输入SQL语句对相应表增删改查进行测试，便于自己SQL语句书写的纠错。UI设计关于FXML文件和CSS的内容真是复杂纷呈，我在有限时间内做出了一些UI的修改与美化，还算是较为满意的。

总体下来，自己在这个实验中得到的锻炼主要便是复杂数据库语言的实践，在数据库测试时经常会报错，虽然修改错误的过程并不难，但是要在其中吸取教训，才是有意义的，不管怎样，每次出现问题都是对所学知识加深认识的机会，都是进步的过程！