数据库课程设计报告书

**题目：图书馆管理系统**

目录

[一、选题背景 4](#_Toc10754)

[二、需求分析 4](#_Toc25181)

[2.1调查用户需求 4](#_Toc28885)

[2.1.1图书馆系统的组织机构 4](#_Toc24773)

[2.1.2各部门的业务活动 4](#_Toc14621)

[2.1.3用户对系统的要求 4](#_Toc2324)

[2.1.4确定系统边界 6](#_Toc23009)

[2.2系统功能设计 6](#_Toc594)

[2.2.1图书搜索展示子系统 6](#_Toc24995)

[2.2.2读者个人信息子系统 7](#_Toc32302)

[2.2.3管理员子系统 7](#_Toc27591)

[2.3数据字典 10](#_Toc16704)

[三、概念结构设计 15](#_Toc28217)

[四、逻辑表结构设计 18](#_Toc1682)

[4.1表的设计及优化说明 18](#_Toc10596)

[4.2关系表总体设计说明 19](#_Toc24652)

[4.3设计用户子模式 20](#_Toc29870)

[4.3.1定义用户级别 20](#_Toc1700)

[4.3.2读者查询子系统 20](#_Toc13343)

[4.3.2管理员查询子系统 21](#_Toc20366)

[五、物理设计 22](#_Toc26578)

[5.1建立索引的依据 22](#_Toc7363)

[5.2需要建立索引的属性 22](#_Toc3172)

[5.3确定数据库的存储结构 22](#_Toc15123)

[六、详细设计 22](#_Toc31623)

[6.1 图书搜索展示子系统 22](#_Toc17686)

[6.2读者个人信息子系统 33](#_Toc13908)

[6.3管理员子系统 38](#_Toc27848)

[6.4前后端接口ajax 62](#_Toc3866)

[6.5关键功能的算法设计 71](#_Toc1433)

[6.5.1 MVC 71](#_Toc4563)

[6.5.2 ThinkPHP框架 71](#_Toc2322)

[6.5.3 RBAC 72](#_Toc7233)

[6.5.4 ajax保证完整性 72](#_Toc20955)

[七、课程总结和体会 72](#_Toc5578)

一、选题背景

图书馆管理系统是最常见的数据库管理系统之一，各大图书馆都有自己的一套系统，读者和管理员通过该系统对馆藏图书进行相关的操作。图书馆信息的数字化运用于流程管理的过程，以图书馆管理人员和读者为直接受益对象。

图书馆管理系统解决的主要问题有管理员对馆藏图书的管理，管理员对读者信息的管理，读者对馆藏图书进行查询和借阅等。图书管理系统是比较简单，但是操作数量庞大的信息系统，是图书馆运作方式和业务流程的具体体现。它实施的技术手段也与目前发展的信息技术密切相关。

总的来说，图书管理系统的建立具有长期性发展的特点，它并不是封闭的软件，会随着信息技术的发展逐步深化发展的软件系统。

本小组通过参考现有的各大图书馆管理系统，结合日常的使用经验，来设计实现本次课程设计的具体内容。

二、需求分析

2.1调查用户需求

我们所要做的是一个高校图书馆藏书借阅系统，用于读者对图书的检索、浏览、借阅。因此，需要的功能包括对图书的检索查询、借阅、预约、续借、安全性保障。 而在拓展功能板块，我们的考虑为以图书为中心建立起图书与图书之间的关系，图书与人之间的关系，以图书为中心的推荐系统强调的是如何建立起有效的链接关系。本系统最终用户为图书馆的读者和管理员，我们根据图书馆的现有信息和日常生活中的经验，得出有以下的实际要求：

2.1.1图书馆系统的组织机构

图书管理主要分为两个部分，读者部分和图书管理员部分，图书馆的日常工作都围绕这两部分进行。

2.1.2各部门的业务活动

读者部分分为注册、登录、查询、借阅、预约、评论、反馈几大功能，查询包括对ISBN、书名、作者、出版社等关键字查询；借阅包括对查询对图书进行借阅、对未过期对图书进行续借、对过期的图书进行罚款等；预约包括对不在馆的书籍进行预约，等书籍在馆后进行通知；评论包括对书籍进行评分和评论。

图书管理员部分分为注册、登录、对图书和用户信息增删改查、对用户罚款、接收用户反馈、热门书籍管理等功能。包括对图书的添加、删除、信息修改，通过用户的注册申请、修改用户信息等。

2.1.3用户对系统的要求

2.1.3.1信息要求

\*对此系统有需求的用户是有浏览、借阅图书需求的读者——由教师、学生组成和图书管理员。 其中，教师、学生是有借阅/预约图书资格的用户，而且借阅权限有所不同；图书管理员根据被授予的等级不同也有不同的操作权限。

\*用户需要注册填写基本信息来使用该系统的功能，主要包括学/工号，姓名，读者类型，所在学院，联系方式等。

\*教师与学生使用此系统主要用于检索、浏览、借阅、预约图书，包括查询书目信息，借书、还书、评论、预约与续借。教师与学生为获得借阅资格，需要在此系统上进行登录，然后通过检索，并根据每本书的书目信息——其ISBN、标题、摘要、封面图、出版者、出版年份、出版社、在馆位置、图书类型、价格来确定所借阅的图书。

\*教师与学生还有对自身用户状态的信息需求，包括借阅记录、预约记录、评论表、信用值、是否被罚款、修改个人信息等。在进行图书相关操作时会得到相关的提示，比如超期还书会提示先交罚款才能还书等。同时，还有对所需图书的借阅状态的信息需求，比如只有知道此书是否已被他人预约而确定能否续借等。额外的需求还有图书推荐、期刊推荐等。

\*图书管理员有获取图书及读者的信息需求，包括全部图书的详细信息，热门图书的信息，全部读者的详细信息，全部借阅记录、预约记录、评论记录、反馈记录以及注册申请记录等。

2.1.3.2处理要求

系统应当完成以下的信息处理：

1.存储注册的用户信息，审核注册用户并通知用户，存储添加的图书信息；

2.对新增加的用户和图书进行及时的更新和统计；

3.存储热门图书的信息并及时更新；

4.对借阅记录、预约记录、评论记录、反馈记录进行及时的更新和统计；

5.预约的图书有馆藏后，更新相关的记录，对相关用户进行及时的通知；

2.1.3.3安全性要求

1. 游客只能使用查询图书功能，若要使用更多功能必须先注册为用户，并且要求设置密码，密码采用md5进行加密，保证用户身份不被盗用；
2. 系统应对不同的数据设置不同的访问级别，限制访问用户可查询和处理的数据类别和内容；
3. 系统对不同的用户类型设置不同的权限，区分不同的用户，如教师和学生的可借阅数量不同。

2.1.3.4完整性要求

1. 各种信息记录的完整性，重要信息记录内容不能为空；
2. 各种数据间联系的正确性；
3. 相同的数据在不同表中记录的一致性。

2.1.4确定系统边界

\*对此系统的人工操作包括供图书馆管理员操作的图书管理（增删改查），读者管理（新用户的等级等），图书分类管理，图书借阅，读者借阅查询，预期处理，图书归还，系统设置，审核注册申请等，而数据库管理员拥有上述的所有权限，并且能够定义图书馆管理员的权限等。 总体而言，DBA拥有对于系统的全部掌握。

\*由计算机完成的工作是对数据进行各种管理和处理；由手工完成的工作主要有：书籍数据的录入、更新，部分用户数据的删改，失效数据或无用数据的删除以及系统的日常维护等。

2.2系统功能设计

2.2.1图书搜索展示子系统

1.导航栏注册和登陆功能

游客可以任意浏览图书相关信息，但不能进行预约和借阅行为。

游客可以通过注册（分教师注册和学生注册）获得账号。

游客在注册成功之后可以通过登陆来实现预约和借阅书籍，并且查看自己的个人信息。

游客注册完成--->管理员进行审核并给予通过--->管理员向游客发送邮件---->游客登陆。

2.首页图书搜索功能

读者可以选择按书名、任意词、作者、期刊名、出版社来搜索书籍。读者不需要完全匹配搜索项也可以匹配到对应的书籍，即模糊搜索。

3.图书类型查询功能

读者可以通过选择图书或期刊的类型来查找对应的图书或期刊

4.热门图书推荐功能和热门期刊推荐功能

读者可以看到经常被借阅或者评分高的图书期刊推荐，从而选择是否要借阅该书（按评分由高到低进行排序）

5.查看图书详情功能

查看图书详情功能：读者可以任意地点击某本书的书名来查看该书籍更多的详情信息，如书籍的评论和评分，馆藏位置，简要介绍等等。

评论和评分书籍功能：读者可以在该图书详情页面下方进行对此书的评论和评分

预约书籍功能：符合预约要求的读者可以在该图书详情页面下方预约该书

借阅图书功能：符合借阅要求的读者可以在该图书详情页面下方借阅该书

6.反馈功能

已登录的用户可以在首页下方点击用户反馈，来填写对我们图书馆管理系统的意见反馈。

2.2.2读者个人信息子系统

1. 查看个人信息

读者登录后可以查看到自己的个人信息

2. 修改个人信息

读者可以在个人信息界面对联系方式、登录密码进行修改

1. 查看个人借阅、预约、评论记录

读者可以查看自己的借阅记录、预约记录、评论记录，并显示借阅、预约、评论的具体信息

1. 相关书本的续借、还书操作

读者可以在借阅记录中对借阅的书籍进行续借和还书

2.2.3管理员子系统

后台管理员登录页面：管理员通过自己的账号、密码以及输入验证码登录到后台操作界面。

2.2.3.1普通管理员

普通管理员分为三级，每级管理员权限不同，具体如下。

一级管理员

1. 图书管理

I.书籍管理

i)查看书籍

目前数据库中的所有书籍

搜索查找书籍，分为书籍的索引号和书籍名称两种查询方式

ii)可以添加新的书籍

iii)修改书籍的相关信息

iv)删除书籍，分为单独删除和批量删除

Ⅱ.期刊管理

i)查看期刊

目前数据库中的所有期刊及其信息

搜索查找期刊，分为期刊的索引号和期刊名称两种查询方式

ii)可以添加新的期刊

iii)修改期刊的相关信息

iv)删除期刊，分为单独删除和批量删除

1. 书籍借阅

Ⅰ.借阅列表

查看当前所有的借阅记录，并可以对借阅记录进行删除。

筛选逾期归还的借阅记录，得到逾期未还的借阅记录表，对相应的读者进行罚款并扣除信用值

Ⅱ.预约列表

查看图书预约记录表,删除预约记录

1. 热门管理

Ⅰ.热门图书列表

查看热门图书

对热门图书进行更新

Ⅱ.热门期刊列表

查看热门期刊

对热门期刊进行更新

1. 读者管理

Ⅰ.读者列表

查看读者信息，删除读者，对提交罚款的读者进行罚款费用的修改

Ⅱ.读者注册审核表

对读者提交的注册信息进行审核，符合要求的读者，同意其注册。

III.读者预约通知表

对该表上的读者进行邮件通知,图书预约成功

1. 查看信息

Ⅰ.反馈信息列表

查看系统的反馈信息列表，对读者的反馈进行改进

删除反馈记录

Ⅱ.图书评论

查看图书的评论信息，可对不当评论进行删除

1. 个人信息

查看个人信息

修改个人信息，密码，邮箱等

二级管理员

1. 书籍借阅

Ⅰ.借阅列表

查看当前所有的借阅记录，并可以对借阅记录进行删除。

筛选逾期归还的借阅记录，得到逾期未还的借阅记录表，对相应的读者进行罚款并扣除信用值

Ⅱ.预约列表

查看图书预约记录表,删除预约记录

1. 热门管理

Ⅰ.热门图书列表

查看热门图书

对热门图书进行更新

Ⅱ.热门期刊列表

查看热门期刊

对热门期刊进行更新

1. 查看信息

Ⅰ.反馈信息列表

查看系统的反馈信息列表，对读者的反馈进行改进

删除反馈记录

Ⅱ.图书评论

查看图书的评论信息，可对不当评论进行删除

1. 个人信息

查看个人信息

修改个人信息，密码，邮箱等

三级管理员

1. 读者管理

Ⅰ.读者列表

查看读者信息，删除读者，对提交罚款的读者进行罚款费用的修改

Ⅱ.读者注册审核表

对读者提交的注册信息进行审核，符合要求的读者，同意其注册。

III.读者预约通知表

对该表上的读者进行邮件通知,图书预约成功

1. 查看信息

Ⅰ.反馈信息列表

查看系统的反馈信息列表，对读者的反馈进行改进

删除反馈记录

Ⅱ.图书评论

查看图书的评论信息，可对不当评论进行删除

1. 个人信息

查看个人信息

修改个人信息，密码，邮箱等

2.2.3.2 超级管理员

一级管理员的功能(除去个人信息的功能和登录功能)+管理员管理的功能

管理员管理:

* 添加管理员,并设置相应的权限;
* 删除管理员
* 修改管理员的权限

2.3数据字典

注册申请

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 申请流水编号 | apply\_id | int | 15 |
| 学/工号 | reader\_id | varchar | 20 |
| 密码 | reader\_password | char | 20 |
| 读者类型 | reader\_type\_id | int | 1 |
| 姓名 | reader\_name | varchar | 20 |
| 申请时间 | apply\_date | date |  |
| 学院编号 | college\_id | int | 3 |
| 联系电话 | reader\_tel | varchar | 15 |
| 邮箱 | reader\_mail | varchar | 20 |
| 是否通过 | apply\_result | tinyint | 3 |

预约

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 预约流水编号 | app\_id | int | 15 |
| 索引号 | book\_id | varchar | 20 |
| 读者ID | reader\_id | varchar | 20 |
| 预约日期 | app\_date | datetime |  |

图书预约通知

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 预约通知流水编号 | inform\_id | int | 15 |
| 索引号 | book\_id | varchar | 20 |
| 读者ID | reader\_id | varchar | 20 |

图书

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 索引号 | book\_id | varchar | 50 |
| 书名 | book\_name | varchr | 20 |
| 图书简介 | book\_intro | text |  |
| 价格 | book\_price | double | 10,3 |
| 剩余数量 | book\_quantity | int | 3 |
| 作者 | book\_author | varchar | 100 |
| 译者 | book\_translator | varchar | 30 |
| 国家编号 | country\_id | int | 3 |
| 出版社编号 | publish\_id | int | 10 |
| 出版日期 | book\_publish\_date | date |  |
| 图书封面 | book\_cover | char | 100 |
| 借阅次数 | book\_frequency | int | 5 |
| 评分 | book\_point | double | 10,3 |
| 馆藏位置 | location\_id | int | 20 |
| 图书类型 | book\_type\_id | char | 2 |

期刊

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 索引号 | book\_id | varchar | 50 |
| 期刊名 | jou\_name | varchar | 20 |
| 出版社编号 | publish\_id | int | 10 |
| 出版日期 | jou\_publish\_date | date |  |
| 价格 | jou\_price | double | 100,3 |
| 期刊分类编号 | book\_type\_id | char | 2 |
| 剩余数量 | jou\_quantity | int | 3 |
| 馆藏位置编号 | location\_id | int | 20 |
| 附注 | Other | text |  |
| 期刊封面 | jou\_cover | varchar | 100 |
| 期刊评分 | jou\_point | double | 5,3 |

图书类型

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 书籍分类编号 | book\_type\_id | char | 2 |
| 分类名称 | book\_type\_name | varchar | 100 |
| 附注 | Other | varchar | 20 |

馆藏地

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 馆藏位置编号 | location\_id | int | 5 |
| 馆藏位置 | location | varchar | 20 |
| 馆藏名称 | location\_name | varchar | 20 |
| 功能/附注 | loc\_function | varchar | 100 |

出版社

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 出版社编号 | publish\_id | int | 10 |
| 出版社名称 | publish\_name | varchar | 20 |

热门书籍

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 索引号 | book\_id | varchar | 20 |
| 借阅次数 | book\_frequency | int | 5 |
| 评分 | book\_point | double | 5,3 |

热门期刊

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 索引号 | book\_id | varchar | 20 |
| 评分 | jou\_point | double | 5,3 |

国家

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 国家编号 | country\_id | int | 3 |
| 国家名称 | country\_name | varchar | 20 |
| 国家名称简写 | country\_short | varchar | 20 |

学院

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 学院编号 | college\_id | int | 3 |
| 学院名称 | college\_name | varchar | 20 |

借阅记录

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 借阅流水编号 | borrowed\_id | int | 10 |
| 读者ID | reader\_id | varchar | 20 |
| 索引号 | book\_id | varchar | 20 |
| 起始日期 | borrowed\_sdate | date |  |
| 到期日期 | borrowed\_edate | date |  |
| 归还日期 | borrowed\_rdate | date |  |
| 续借次数 | borrowed\_times | int | 2 |
| 书籍借阅是否到期 | is\_overed | int | 1 |

图书评论

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 评论流水编号 | com\_id | int | 15 |
| 索引号 | book\_id | varchar | 20 |
| 读者ID | reader\_id | varchar | 20 |
| 评论日期 | com\_date | date |  |
| 评分 | com\_point | double | 10,3 |
| 评论内容 | com\_content | text |  |

期刊评论

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 评论流水编号 | com\_id | int | 15 |
| 索引号 | book\_id | varchar | 20 |
| 读者ID | reader\_id | varchar | 20 |
| 评论日期 | com\_date | date |  |
| 评分 | com\_point | double | 10,3 |
| 评论内容 | com\_content | text |  |

反馈

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 反馈流水编号 | fdb\_id | int | 15 |
| 读者ID | reader\_id | varchar | 20 |
| 反馈信息 | fdb\_content | text |  |
| 反馈日期 | fdb\_date | date |  |

管理员

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 管理员ID | man\_id | varchar | 20 |
| 姓名 | man\_name | varchar | 20 |
| 密码 | man\_passord | char | 32 |
| 重复输入密码 | man\_repasswod | char | 32 |
| 邮箱 | man\_mail | varchar | 30 |
| 管理员级别 | man\_rank | int | 2 |

读者

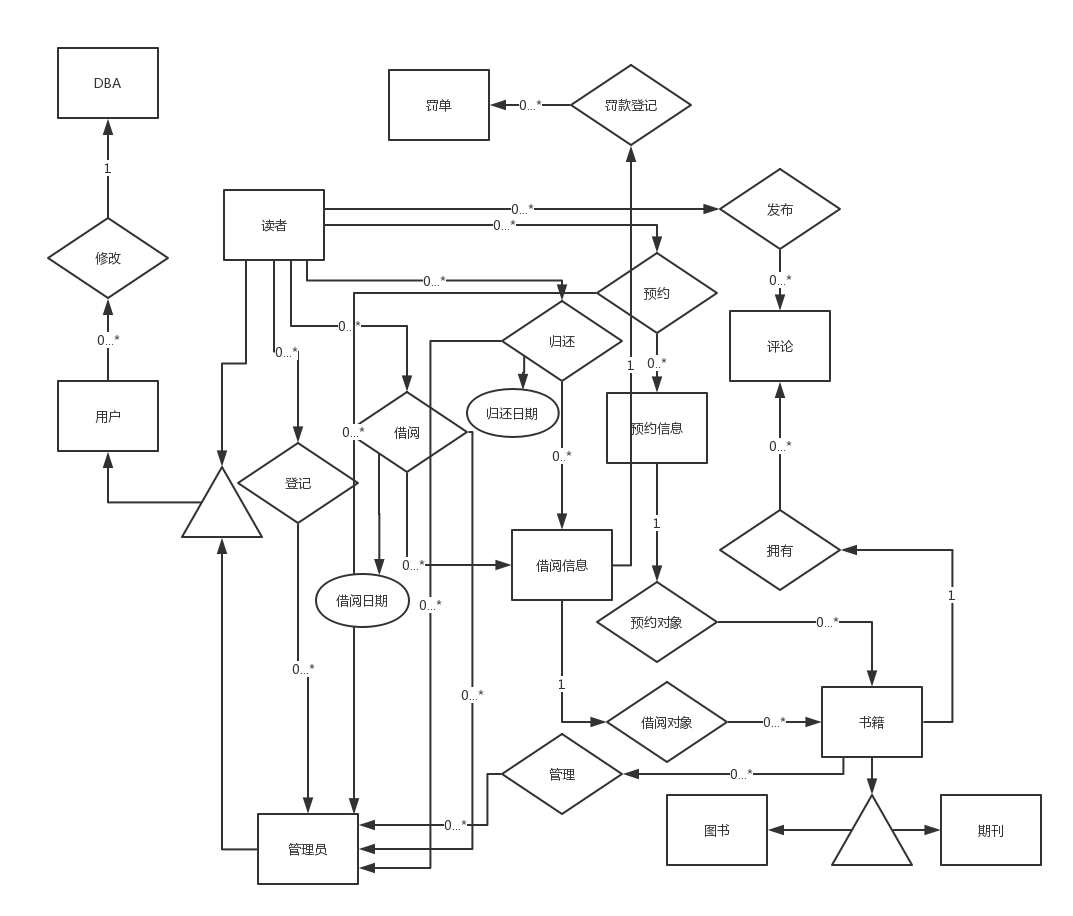
|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 读者ID | reader\_id | varchar | 20 |
| 姓名 | reader\_name | varchar | 20 |
| 密码 | reader\_password | char | 20 |
| 读者类型编号 | reader\_type\_id | int | 1 |
| 开户日期 | reader\_date | date |  |
| 联系电话 | reader\_tel | varchar | 15 |
| 邮箱 | reader\_mail | varchar | 15 |
| 已借阅数量 | reader\_bquantity | int | 3 |
| 信用分数 | reader\_credit | int | 3 |
| 延期还书的罚款 | reader\_fine | double | 200,3 |

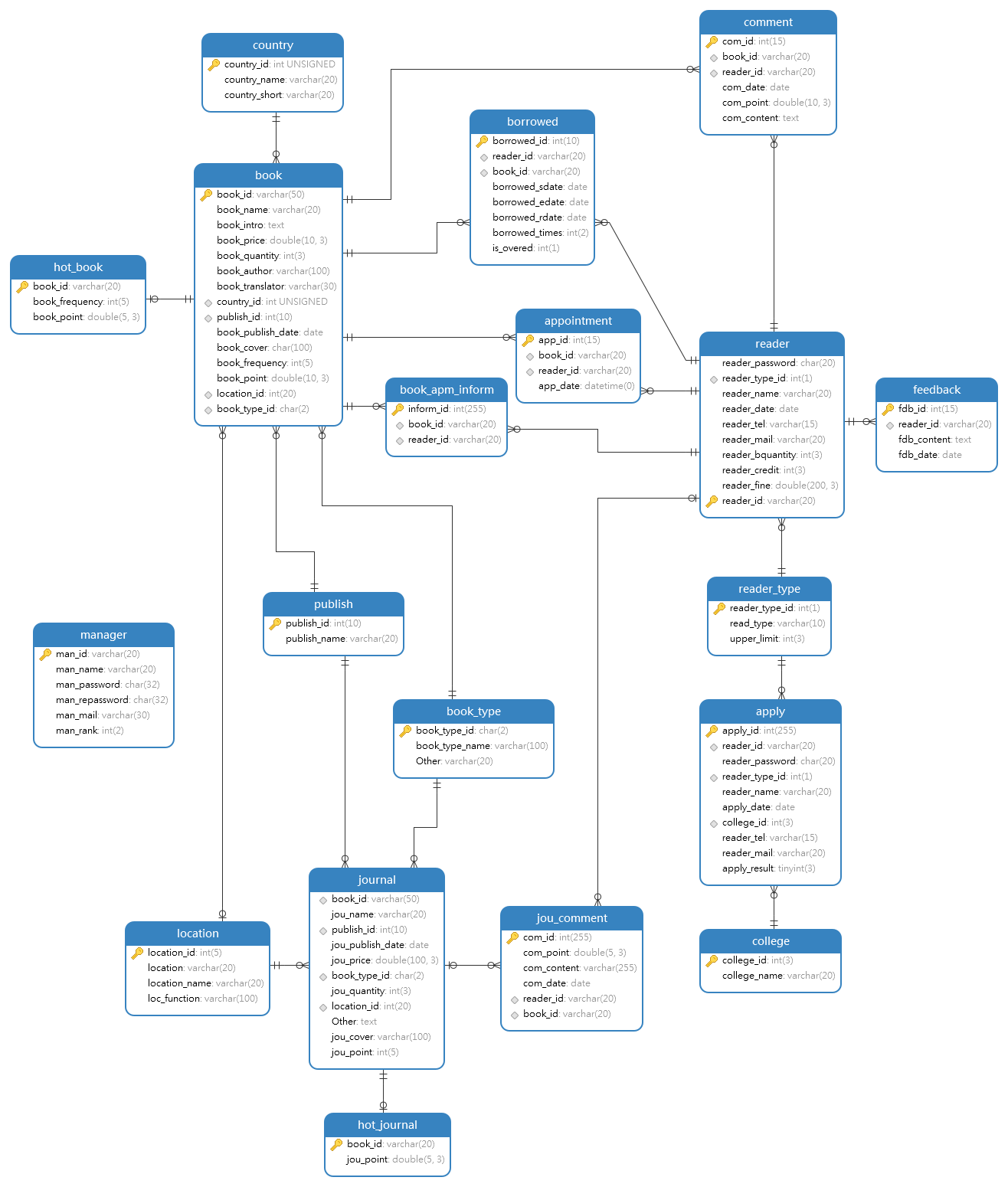
读者类型

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 存储代码 | 类型 | 长度 |
| 读者类型编号 | reader\_type\_id | int | 1 |
| 读者类型 | read\_type | varchar | 10 |
| 借阅数量上限 | upper\_limit | int | 3 |

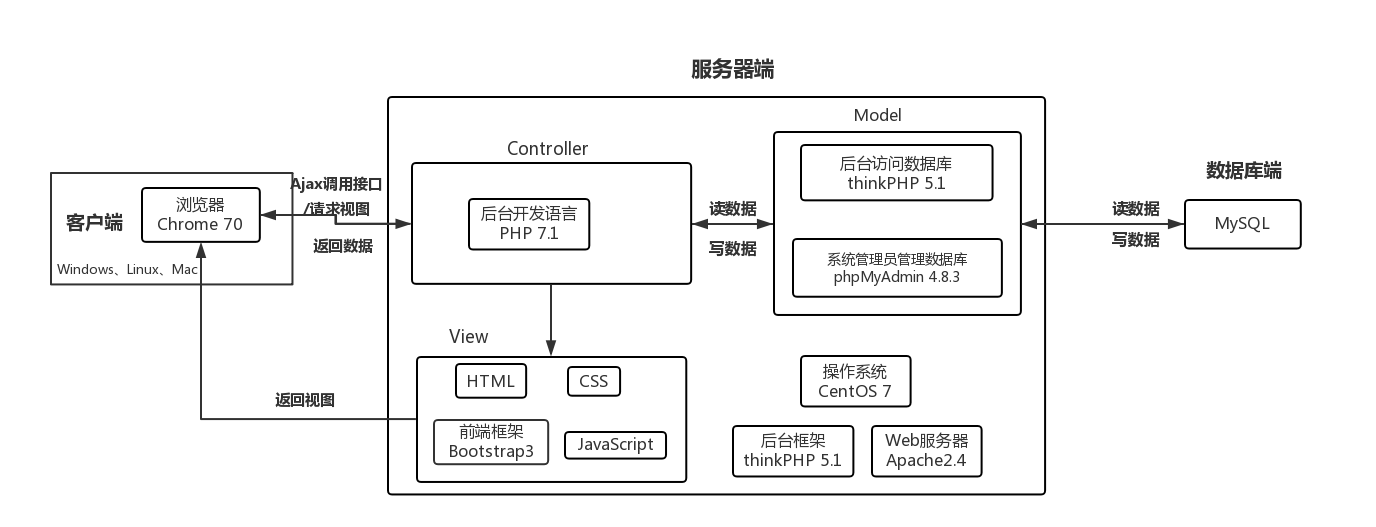
三、概念结构设计

系统的概念结构设计图。





系统逻辑架构设计：



四、逻辑表结构设计

在概念设计的基础上，根据设计得到系统总的E-R图，按照概念模式与关系表转化的一般规则，结合实际的需要进行逻辑设计，E—R图中的实体、实体的属性和实体之间的联系转化为关系模式。最后生成的关系及关系表如下（同时附优化说明）

4.1表的设计及优化说明

1.读者注册申请（申请流水编号、学/工号、密码、读者类型编号、姓名、申请时间、学院编号、联系电话、邮箱、是否通过）

说明：注册申请和读者ID、读者类型、学院都是1:1的联系，将三个主码加入到注册的关系模式中，将读者ID改为学/工号

2.读者（读者ID、姓名、密码、读者类型编号、开户日期、联系电话、邮箱、已借阅数量、信用分数、罚款）

说明：读者与读者类型的联系是1:1，将读者类型的主码读者类型编号加入到读者关系模式中

3.学院（学院编号、学院名称）

读者类型（读者类型编号、读者类型、借阅数量上限）

反馈记录（反馈流水编号、读者ID、反馈信息、反馈日期）

说明：读者和反馈记录的联系是1:n，将读者的主码读者ID加入到反馈记录中

优化说明：读者姓名可通过读者ID查询到，存在冗余，故删除

4.预约记录（预约流水编号、索引号、读者ID、预约日期）

说明：读者和预约记录的联系是1:n，图书和预约记录的联系是1:1，将读者和图书的主码加入到预约记录中

优化说明：读者姓名和书名可以通过读者ID和索引号查询到，存在冗余，故删除

5.预约图书通知（预约通知流水编号、索引号、读者ID）

说明：读者和图书与预约通知的联系是1:n，将读者ID和索引号加入预约图书通知

6.借阅记录（借阅流水编号、读者ID、索引号、起始日期、到期日期、归还日期、续借次数、书籍是否到期）

说明：读者和图书与借阅记录的联系是1：n，将读者ID和索引号加入借阅记录

7.图书评论记录（评论流水编号、图书索引号、读者ID、评论日期、评分、评论内容）

说明：读者和图书与评论的联系是1：n，将读者ID和索引号加入评论记录

优化说明：索引号改为图书索引号，与期刊进行区分

8.期刊评论记录（评论流水编号、期刊索引号、读者ID、评论日期、评分、评论内容）

说明：读者和期刊与评论的联系是1：n，将读者ID和期刊索引号加入期刊评论记录

9．图书（索引号、书名、图书简介、价格、剩余数量、作者、译者、国家编号、出版社编号、出版日期、图书封面、借阅次数、评分、馆藏位置编号、图书类型编号）

说明：国家、出版社、馆藏地、图书类型与图书的联系是1:n，将国家编号、出版社编号、馆藏位置编号、图书类型编号加入图书

优化说明：国家、出版社、馆藏位置均可通过各自编号查询出，存在冗余，故删除

10.期刊（索引号、期刊名、出版社编号、出版日期、价格、期刊分类编号、剩余数量、馆藏位置编号、附注、期刊封面、期刊评分）

说明：出版社、馆藏地、图书类型与期刊的联系是1:n，将出版社编号、馆藏位置编号、图书类型编号加入期刊

11.图书类型（书籍分类编号、分类名称、附注）

国家（国家编号、国家名称）

馆藏地（馆藏位置编号、馆藏位置、馆藏名称、功能/附注）

说明：附注指馆藏位置所藏的具体图书类型

12.出版社（出版社编号、出版社名称）

热门图书（索引号、借阅次数、评分）

说明：图书与热门图书的联系是1:1，将图书索引号加入热门图书

13.热门期刊（索引号、评分）

说明：期刊与热门期刊的联系是1:1，将期刊索引号加入热门期刊

优化说明：期刊改为不能借阅，将借阅次数删除

14.管理员（管理员ID、姓名、密码、注册时重复输入密码、邮箱、管理员等级）

4.2关系表总体设计说明

关于期刊表与图书表、期刊评论记录与图书评论记录不进行合并的说明：

1.图书（索引号、书名、图书简介、价格、剩余数量、作者、译者、国家编号、出版社编号、出版日期、图书封面、借阅次数、评分、馆藏位置编号、图书类型编号）

期刊（索引号、期刊名、出版社编号、出版日期、价格、期刊分类编号、剩余数量、馆藏位置编号、附注、期刊封面、期刊评分）

这两个关系表由于有些字段保留了不同的必要信息，比如图书表有借阅次数而期刊表没有，期刊没有作者而图书有，因此无法进行合并。这里也没有必要把两张表合在一起做总体来处理，用户分别对两类书进行操作就可以了。

2.图书评论记录（评论流水编号、图书索引号、读者ID、评论日期、评分、评论内容）

期刊评论记录（评论流水编号、期刊索引号、读者ID、评论日期、评分、评论内容）

这两个表由于外键约束不能合并在一起。由于图书和期刊的索引号都是book\_id，而一张评论记录表的book\_id不能外键链接图书表的book\_id的同时又连接期刊表的book\_id.删除图书或期刊其中一个表时就会把评论表删除，所以这两个表不能一起处理。

journal期刊不能外借。因为，我们想要模拟真实情景中，部分藏书不能外借的情况。所以journal表中没有jou\_frequency字段。

4.3设计用户子模式

在将概念模型转化为全局逻辑模型后,根据图书管理系统的局部应用需求,以下设计用户子模式：

* 1. 图书搜索展示
  2. 读者信息获取
  3. 管理员管理图书信息
  4. 超级管理员管理管理员信息

4.3.1定义用户级别

* 学生，教师：可以使用系统提供给读者的全部功能，学生和教师借阅上限不同
* 管理员：可以查看除管理员记录之外的全部表
* 超级管理员：可以查看所有表

根据分析需求，我们设计了以下几个子系统：读者查询子系统，管理员查询子系统。

4.3.2读者查询子系统

向读者提供图书馆的书籍搜索查询，读者自身的借阅查询、评论查询、预约查询，显示在相应的页面上。

在这里创建几个视图：

1.查询书目（书名/期刊名、索引号、作者、简介、在馆位置、图书类型、图书封面、检索数目）

注：如果是期刊则没有‘作者’。在馆位置和图书类型分别从馆藏地表和图书类型表获取，检索数目为查询出的记录和，其他从图书信息表中获取。

2.图书详情（作者、译者、封面、出版社、价格、索引号、国家、评分、数量、简介、出版日期、图书所在位置1、图书所在位置2、图书类型）

注：如果是期刊则没有‘作者’‘译者’。图书所在位置1用数字记录，图书所在位置2用文字记录。国家、出版社、馆藏地、图书类型分别从国家表、出版社表、馆藏地表、图书类型表中获取，其他从图书信息表中获取。

3.热门图书（书名、图书评分、被借阅次数、图书封面、索引号）

注：根据被借阅次数和评分来选出热门图书。书名、封面从图书表中获取，其他从热门图书表中获取。

4.热门期刊（期刊名、期刊评分、期刊封面、索引号）

注：根据期刊评分选出热门期刊。期刊名从期刊表中获取，其他从热门期刊表中获取。

5.个人信息（姓名、学/工号、读者类型、开户日期、联系电话、邮箱、已借阅数量、信用分数、欠费情况）

注：读者类型从读者类型表中获取、已借阅数量为用户借阅记录中未还书的和，其他从读者信息表中获取。

6.借阅记录（借阅流水号、索书号、书名、作者、馆藏地、借阅时间、应还时间、还书时间、续借次数、是否到期、借阅记录数量）

注：书名、作者、馆藏地分别从图书表、馆藏地表中获取，续借次数是用户点击续借的次数，借阅记录数量是查询记录的和，其他从借阅表中获取

7.预约记录（预约时间，书名，索书号，预约记录数量）

注：书名从图书表中获取，预约记录数量是查询记录的和，其他从预约表中获取。

8.评论记录（书名，评论时间，评分，评论内容、图书类型、评论记录数量）

注：书名、图书类型从图书表中获取，评论记录数量是查询记录的和，其他从图书评论表和期刊评论表中获取。

4.3.2管理员查询子系统

向图书管理员提供对图书、期刊表的查询，所有评论、预约、借阅记录的查询，所有读者信息的查询，读者反馈信息的查询，显示在相对应的页面上。

管理员视图与读者视图类似，不同的地方有：

1.借阅、预约、评论记录读者只能看到自己的，管理员可以看到所有用户的；

2.增加了用户注册申请视图：

注册（申请流水编号、学/工号、密码、读者类型、姓名、申请时间、学院编号、联系电话、邮箱、是否通过）

3.增加了读者信息视图：

读者信息（读者ID、密码、姓名、学/工号、读者类型、开户日期、联系电话、邮箱、已借阅数量、信用分数、欠费情况）

4.增加了反馈信息视图：

反馈信息（反馈流水号、读者ID、反馈信息、反馈日期）

五、物理设计

图书管理系统是多用户共享的系统，对同一个关系要建立多条存储路径，才能满足多用户的多种应用要求。在本次课程设计中，我们采用B树索引方法作为系统的存取方法。

5.1建立索引的依据

所谓选择索引存取方法实际上就是根据应用要求确定对关系的哪些属性列建立索引、哪些属性列建立组合索引、哪些索引要设计为唯一索引等。根据本信息系统的具体情况，建立索引的依据如下：

1.一个属性经常在查询条件下出现

2.一个属性经常在连接操作的连接条件出现

5.2需要建立索引的属性

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 对应实体 | 属性名 | 对应实体 |
| 读者类型编号 | 读者类型 | 学院编号 | 学院 |
| 图书索引号 | 图书 | 期刊索引号 | 期刊 |
| 读者ID | 读者 | 国家编号 | 国家 |
| 出版社编号 | 出版社 | 馆藏地编号 | 馆藏地 |

5.3确定数据库的存储结构

本部分主要是确定数据库物理结构，即确定数据的存放位置和存储结构，包括确定关系、索引、聚簇、日志、备份等的存储安排和存储结构，确定系统配置等。

在综合考虑存取时间、存储空间利用率和维护代价三方面的因素下，经过权衡，确定下数据的存放位置和存储结构。为了提高系统的性能，根据应用情况将数据的易变部分与稳定部分、经常存取部分与存取效率较低部分分开存放。

考虑到目前许多计算机都有多个磁盘，因此将表和索引放在不同的磁盘上，在查询时，两个磁盘驱动器并行工作，可以大大提高I/O读写的效率。

六、详细设计

6.1 图书搜索展示子系统

###### 注册功能

根据用户提交的信息，包括学号或职工号、姓名、密码、读者类型（教师注册或学生注册）、邮箱、电话、学院，进行初步的提交格式确认后，将信息存储在apply表中，待管理员进行进一步审核。

注册成功，用户会得到弹窗提示“注册成功”，然后跳转到首页；注册失败，用户会得到弹窗提示“注册失败”，并且如果是格式错误，用户会得到相关的提示。

注册的学号/教职工号需要10位

密码要六位以上

确认密码需和密码一致

邮箱必填，符合一定的邮箱格式

电话选填





###### 登录功能

根据用户提交的信息，包括学号或职工号、密码，进行初步的提交格式确认后，然后调用数据库reader表的数据，判断是否存在该用户；然后判断密码是否正确；如果密码正确，则将用户id和密码保存在session中，返回用户提示信息“登陆成功”。否则根据登陆失败原因返回用户相关提示信息。

登陆成功后，导航栏上的登陆按键会变成用户按键，注册会变成登陆，点击用户，即可跳转到用户信息首页，有个人信息、借阅记录、评论记录。用户可以在此进行个人信息查看和修改。





###### 热门图书推荐功能

在数据库hot\_book表中根据图书的借阅记录和评分记录进行统计排序，筛选出排名前5的图书展示在热门图书推荐栏目，显示热门图书的名字，借阅次数，分数。用户可以根据自己喜欢的推荐书目，点击书目，进入图书详情页面查看。

###### 热门期刊推荐功能

在数据库hot\_journal表中根据期刊的评分记录进行统计排序，筛选出排名前5的期刊展示在热门期刊推荐栏目，显示热门期刊的名字，借阅次数，分数。用户可以根据自己喜欢的推荐期刊，点击期刊，进入期刊详情页面查看。

###### 图书类型搜索功能

读者可以点击图书分类栏中的类型来选择搜索该类型的图书，展示的图书信息有书名或期刊名，索书号，出版社，作者，图书或期刊的简介，图书类型。

1. 在数据库的book表中通过连接location表、publish表、book\_type表，判断book\_type\_id字段查询图书，同时将查询得到的结果按照图书借阅频率book\_frequency进行倒序输出。由此得到用户选择的书籍类型的图书。
2. 在数据库的journal表中通过连接location表、publish表、book\_type表，判断book\_type\_id字段查询期刊。由此得到用户选择的书籍类型的期刊。
3. 判断通过两个表得到的结果是否都为空
4. 将两个表搜索出来的结果转成数组形式，分别加上代表图书和期刊的类型标签
5. 将两个数组进行合并，然后按照其评分point进行由大到小的排列
6. 返回前端需要展示的数据（如：接口文档所示）



###### 图书搜索功能



根据书名进行查找

1. 在数据库的book表中通过连接location表、publish表、book\_type表，判断book\_name字段查询图书，同时将查询得到的结果按照图书借阅频率book\_frequency进行倒序输出。由此得到根据书名查询到的图书。
2. 对查询到的数据进行数量统计操作
3. 将搜索得到的数据转为数组形式
4. 标记数据为图书类
5. 返回前端需要展示的数据（如：接口文档所示）



根据任意词进行查找

1. 在数据库的book表中通过连接location表、publish表、book\_type表，判断book\_name字段查询图书，同时将查询得到的结果按照图书借阅频率book\_frequency进行倒序输出。由此得到任意词搜索到的图书。
2. 在数据库的journal表中通过连接location表、publish表、book\_type表，判断jou\_name字段查询图书。由此得到根据任意词搜索到的图书。
3. 判断通过两个表得到的结果是否都为空
4. 将两个表搜索出来的结果转成数组形式，分别加上代表图书和期刊的类型标签
5. 将两个数组进行合并，然后按照其评分point进行由大到小的排列
6. 返回前端需要展示的数据（如：接口文档所示）



根据期刊名进行查找

1. 在数据库的journal表中通过连接location表、publish表、book\_type表，判断book\_type\_id字段查询图书。由此得到用户选择的书籍类型的期刊。
2. 对查询到的数据进行数量统计操作
3. 将搜索得到的数据转为数组形式
4. 标记数据为期刊类型
5. 然后按照其评分point进行由大到小的排列
6. 返回前端需要展示的数据（如：接口文档所示）



根据作者进行查找

1. 在数据库的book表中通过连接location表、publish表、book\_type表，判断authot字段查询图书，同时将查询得到的结果按照图书借阅频率book\_frequency进行倒序输出。由此得到根据书名查询到的图书。
2. 对查询到的数据进行数量统计操作
3. 将搜索得到的数据转为数组形式
4. 标记数据为图书类型
5. 返回前端需要展示的数据（如：接口文档所示）



根据出版社进行查找

1. 在数据库的book表中通过连接location表、publish表、book\_type表，判断publish字段查询图书，同时将查询得到的结果按照图书借阅频率book\_frequency进行倒序输出。由此得到用户选择的书籍类型的图书。
2. 在数据库的journal表中通过连接location表、publish表、book\_type表，判断publish字段查询期刊。由此得到用户选择的书籍类型的期刊。
3. 判断通过两个表得到的结果是否都为空
4. 将两个表搜索出来的结果转成数组形式，分别加上代表图书和期刊的类型标签
5. 将两个数组进行合并，然后按照其评分point进行由大到小的排列
6. 返回前端需要展示的数据（如：接口文档所示）



###### 反馈功能

已登录的用户可以在首页下方点击用户反馈，来填写对我们图书馆管理系统的意见反馈。

1. 通过session值判断用户是否已登录。
2. 将用户的反馈，用户id，当前日期插入数据库的feedback表中。
3. 根据反馈是否成功给用户弹窗提示相关信息





###### 图书详情展示功能

用户在任意板块中，包括各种类型图书搜索、热门图书推荐等，点击图书名字即可进入该图书的详情页面展示，内容包括：书籍图片、书籍名称、出版社、书籍分类、书籍简介等信息。

1. 通过索书号book\_id在数据库journal表和book表中连接location表、publish表、country表、book\_type表查找书籍数据。
2. 将搜索得到的数据转为数组形式
3. 标记数据为图书或期刊类型
4. 返回前端需要展示的数据（如：接口文档所示）



###### 搜索图书评论功能

用户进入图书详情页面后，能够看到自己曾经或其他用户对该图书的评论。

1. 根据书籍类型type分别去数据库comment表（对应book）或者jou\_comment表（对应journal）中查找数据。
2. 通过索引号book\_id来进行判断查询。
3. 将搜索得到的数据转为数组形式
4. 返回前端需要展示的数据（如：接口文档所示）



###### 图书评论功能

用户进入图书详情页面后，能够对该图书进行评论。

1. 通过session值判断用户是否已登录。
2. 获取当前的日期
3. 根据书籍类型type将用户id，索书号book\_id，用户评论com\_content，用户评分com\_point，评论日期com\_date插入comment表（对于book来说）或jou\_comment表（对于journal来说）中。
4. 计算目前该书籍所有评论的平均分数
5. 根据该分数更新book表或journal表中的book\_point字段或jou\_point字段。



###### 图书预约功能

已登录的用户可以在图书详情页面看到该书籍的索书号，馆藏位置，剩余数量。然后用户可以根据这些相关信息点击预约按钮预约该书。

1. 通过session值判断用户是否已登录。
2. 通过reader\_id到数据库的reader表中判断读者的信用值是否大于0
3. 判断读者是否还有罚金没有上缴.
4. 判断读者是否到达借阅上限.
5. 通过reader\_id和book\_id到数据库的appointment表中判断读者之前是否预约过该书籍
6. 通过reader\_id和book\_id到数据库的borrowed表中判断读者之前是否借阅过该书籍
7. 获取当前时间
8. 将索书号book\_id，reader\_id,app\_date插入appointment表中
9. 预约成功。如果上述有判断出错的，则用弹窗方式告诉读者预约失败的原因。
10. 当有其他用户归还这本书时，管理员可以通知预约该书的读者，告知其预约的书现在空闲下来了。读者即可借阅该书籍。



###### 图书借阅功能

已登录的用户可以在图书详情页面看到该书籍的索书号，馆藏位置，剩余数量。然后用户可以根据这些相关信息点击借阅按钮借阅该书。

1. 通过session值判断用户是否已登录。
2. 通过reader\_id到数据库的reader表中判断读者的信用值是否大于0
3. 判断读者是否还有罚金没有上缴.
4. 判断读者是否到达借阅上限.
5. 通过reader\_id和book\_id到数据库的borrowed表中判断读者之前是否借阅过该书籍
6. 判断该书是否在馆内仍存本可供借阅
7. 获取当前时间borrow\_sdate，计算归还时间borrow\_rdate
8. 将索书号book\_id，reader\_id,borrow\_sdate,borrow\_rdate插入borrowed表中
9. 根据reader\_id 和book\_id 到数据库的appointment表中查看该读者之前是否有预约过该书籍，如果有，则在appointment表中删除相关数据
10. 根据book\_id到book表或journal表中更新图书或期刊的frequency(+1)
11. 借阅成功。如果上述有判断出错的，则用弹窗方式告诉读者借阅失败的原因。

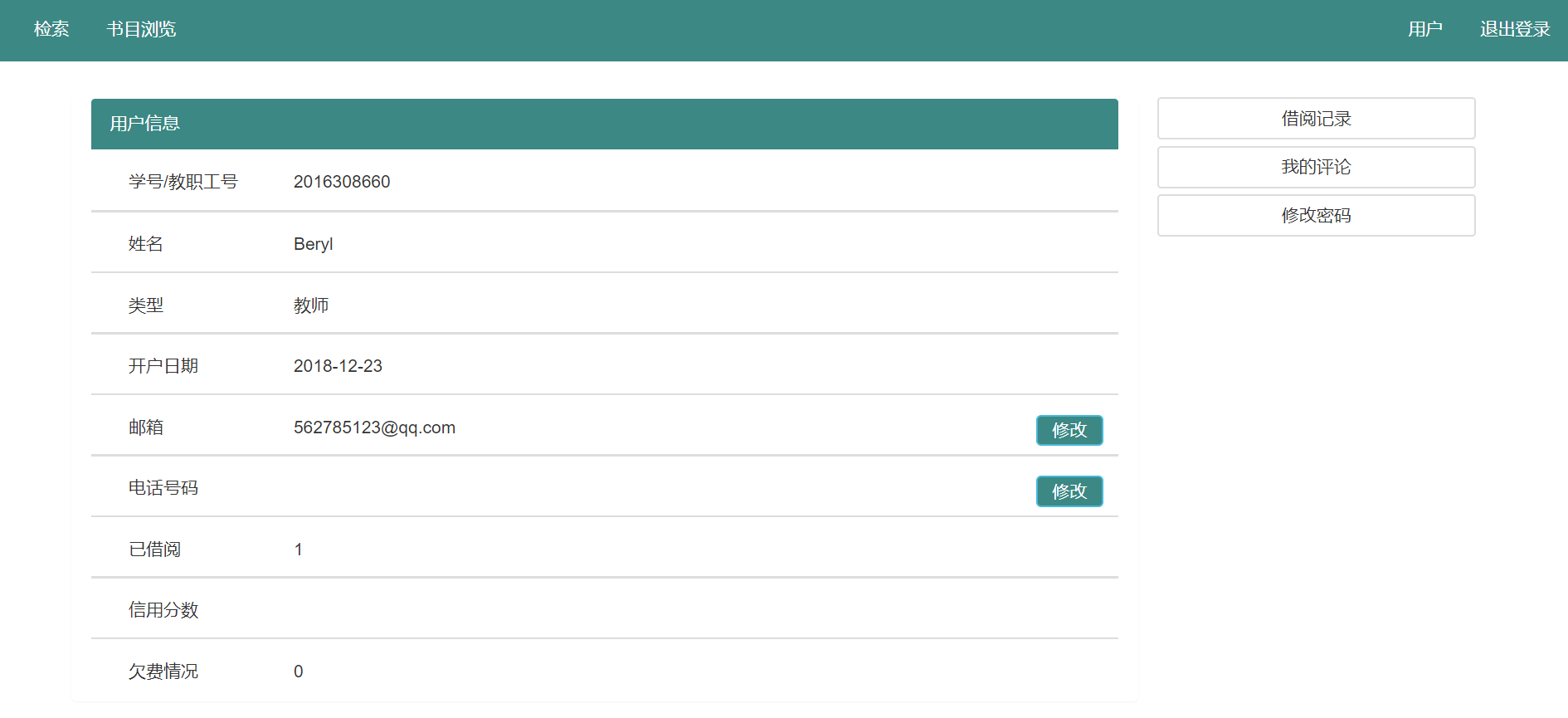


6.2读者个人信息子系统

###### 个人信息显示

用户登录后可以查看个人信息

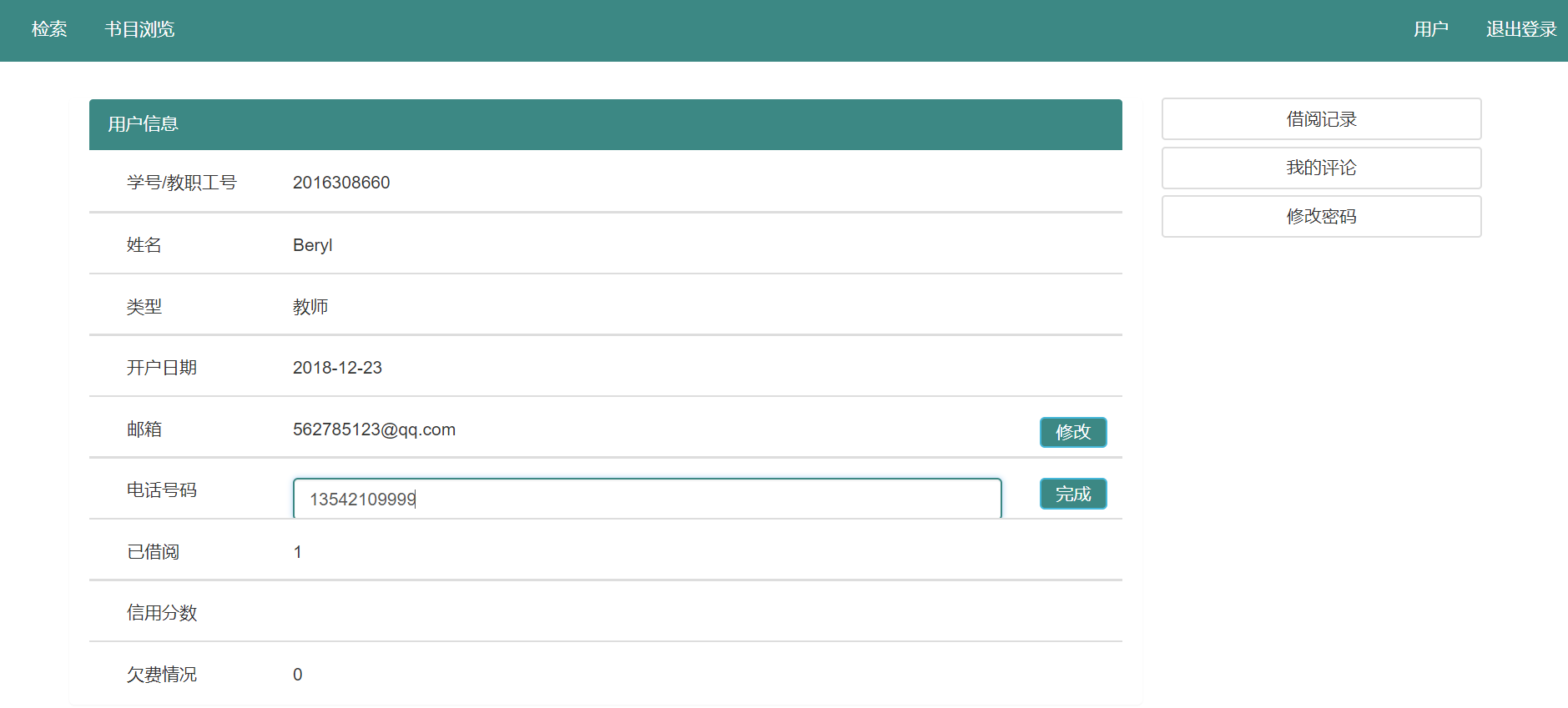
1. 通过session值判断用户是否登录
2. 连接数据库reader表和reader\_type表查询用户详细信息
3. 返回数据



###### 修改电话

用户在个人信息界面可以修改联系电话

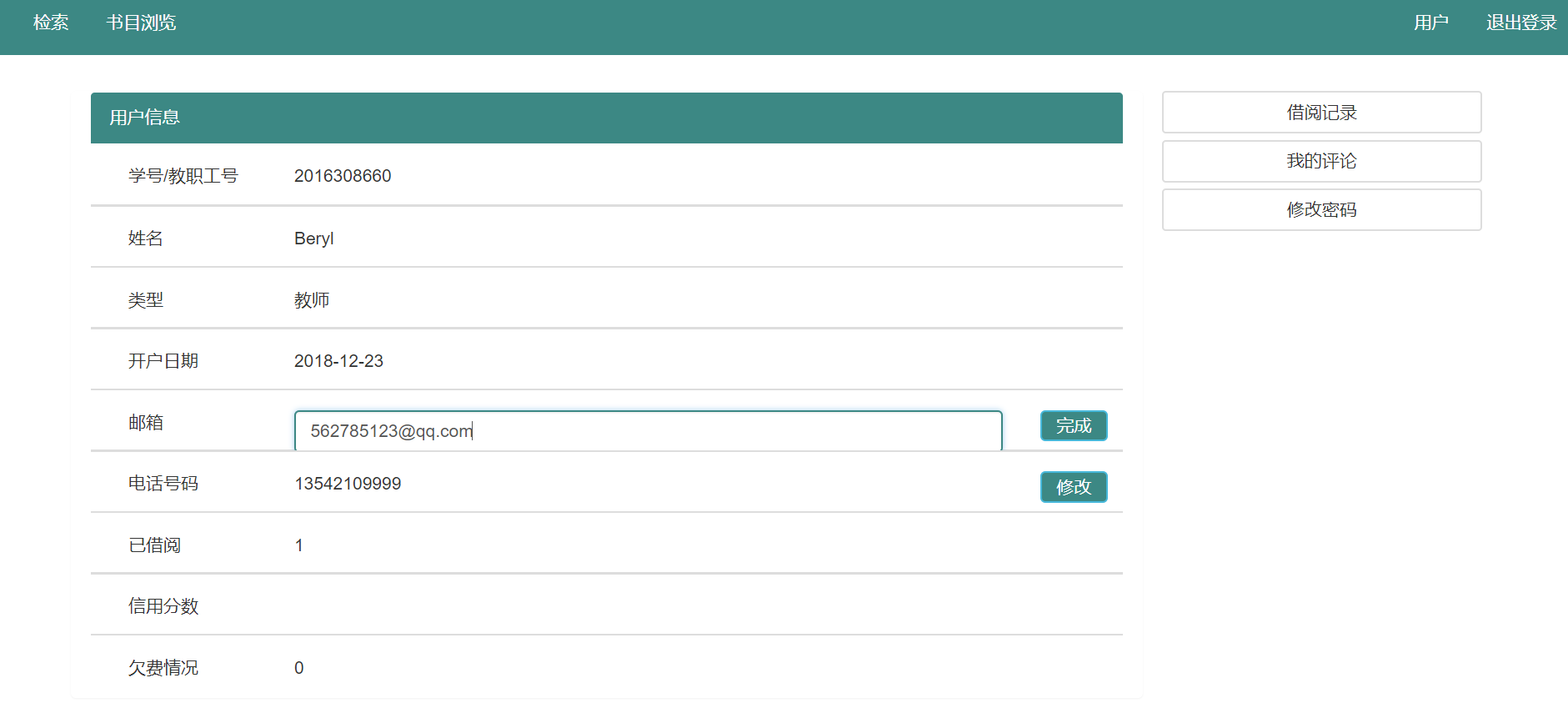
1. 通过session值判断用户是否登录
2. 获取用户输入的新电话，连接数据表reader进行数据更新
3. 修改成功



###### 修改邮箱

用户在个人信息界面可以修改邮箱

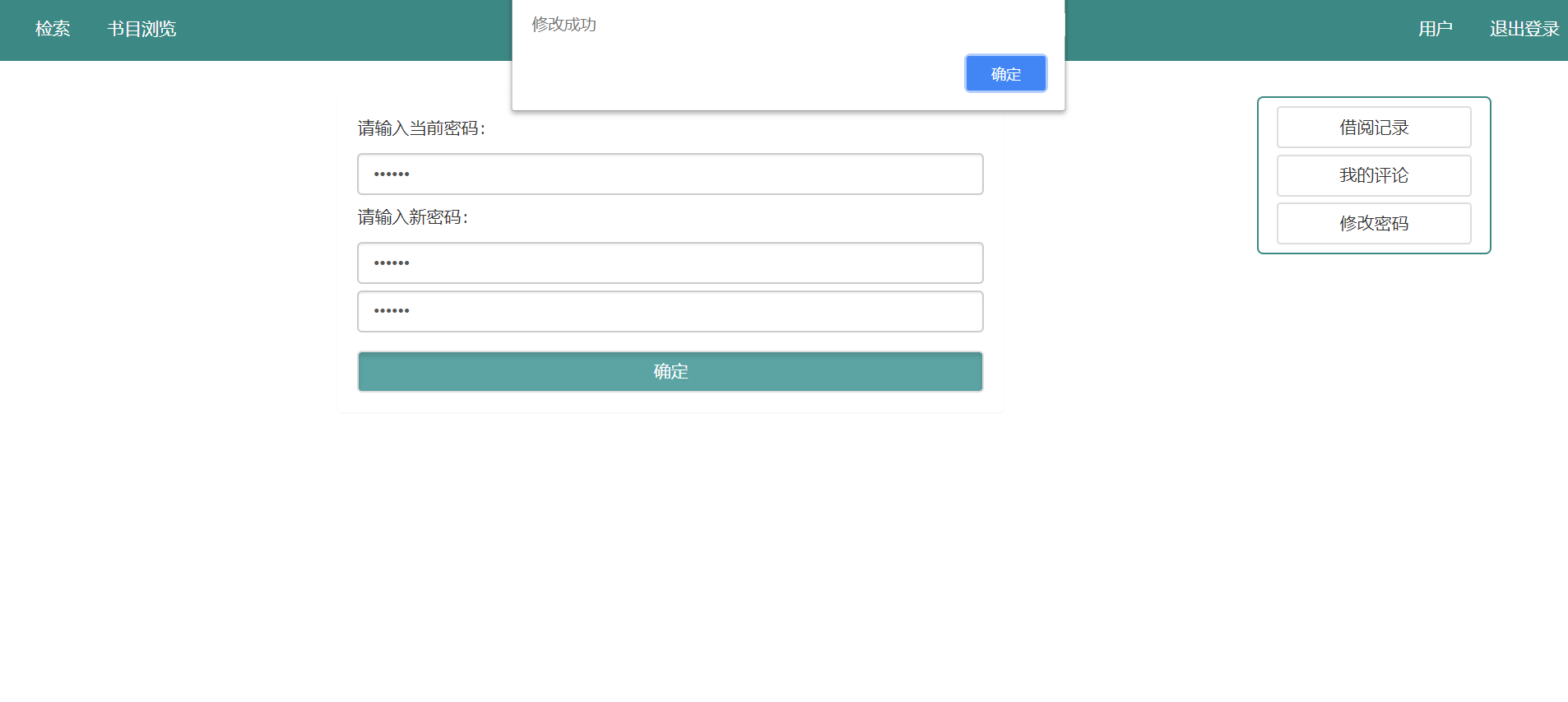
1. 通过session值判断用户是否登录
2. 获取用户输入的新邮箱，连接数据表reader进行数据更新
3. 修改成功



###### 修改密码

用户提交旧密码和要设置的新密码（2次输入）来修改登录密码

1. 通过session值判断用户是否登录
2. 获取用户输入的新旧密码
3. 连接数据表reader查询旧密码输入是否正确
4. 如果正确则更新reader中的password，错误则返回弹窗提示



###### 个人评论

用户查看个人评论记录

1. 通过session值判断用户是否登录
2. 连接数据表comment、jou\_comment查看用户的预约详情
3. 计算评论记录总数
4. 返回数据



###### 个人借阅

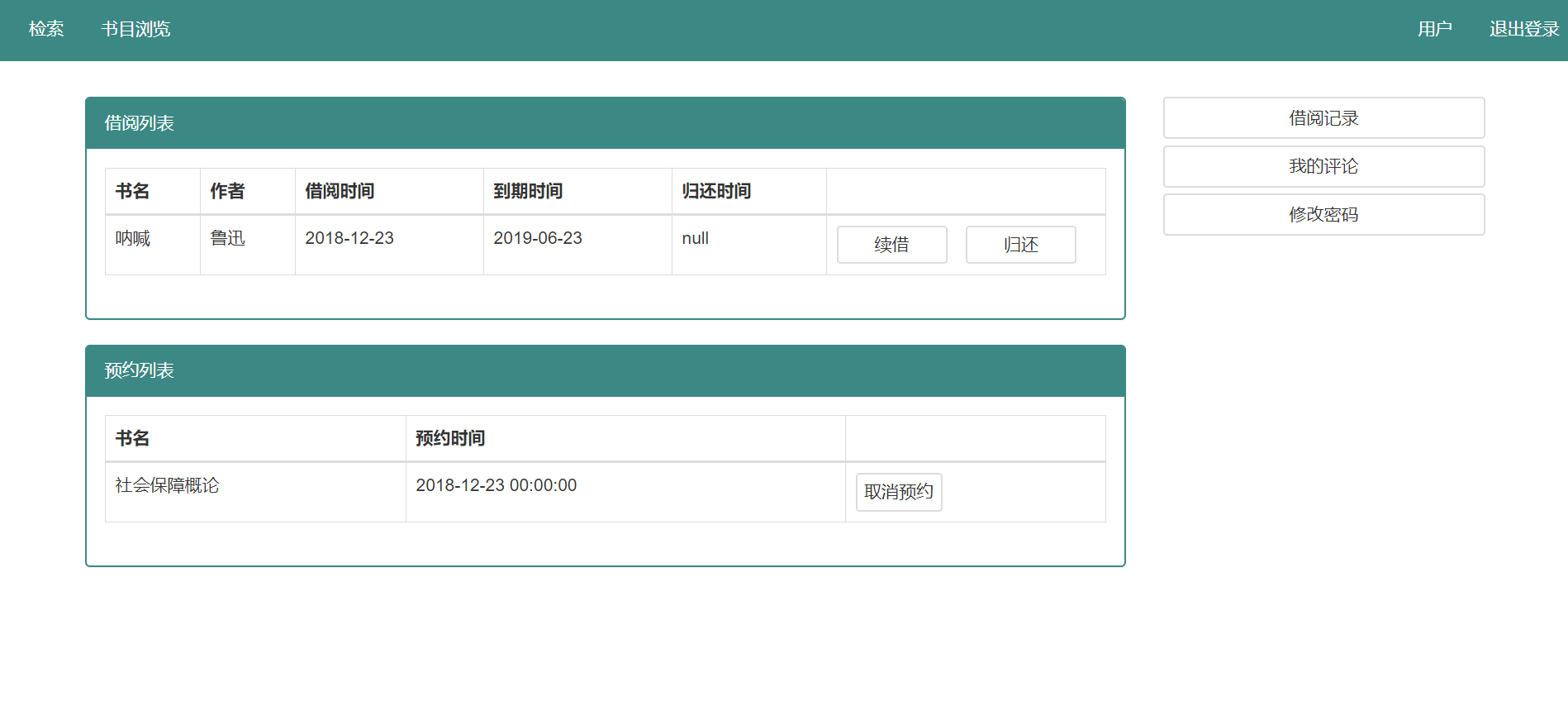
用户查看个人借阅记录

1. 通过session值判断用户是否登录
2. 连接数据表borrowed、book、location，查看当前用户的借阅信息
3. 计算借阅记录总数
4. 返回数据

###### 个人预约

用户查看个人预约记录

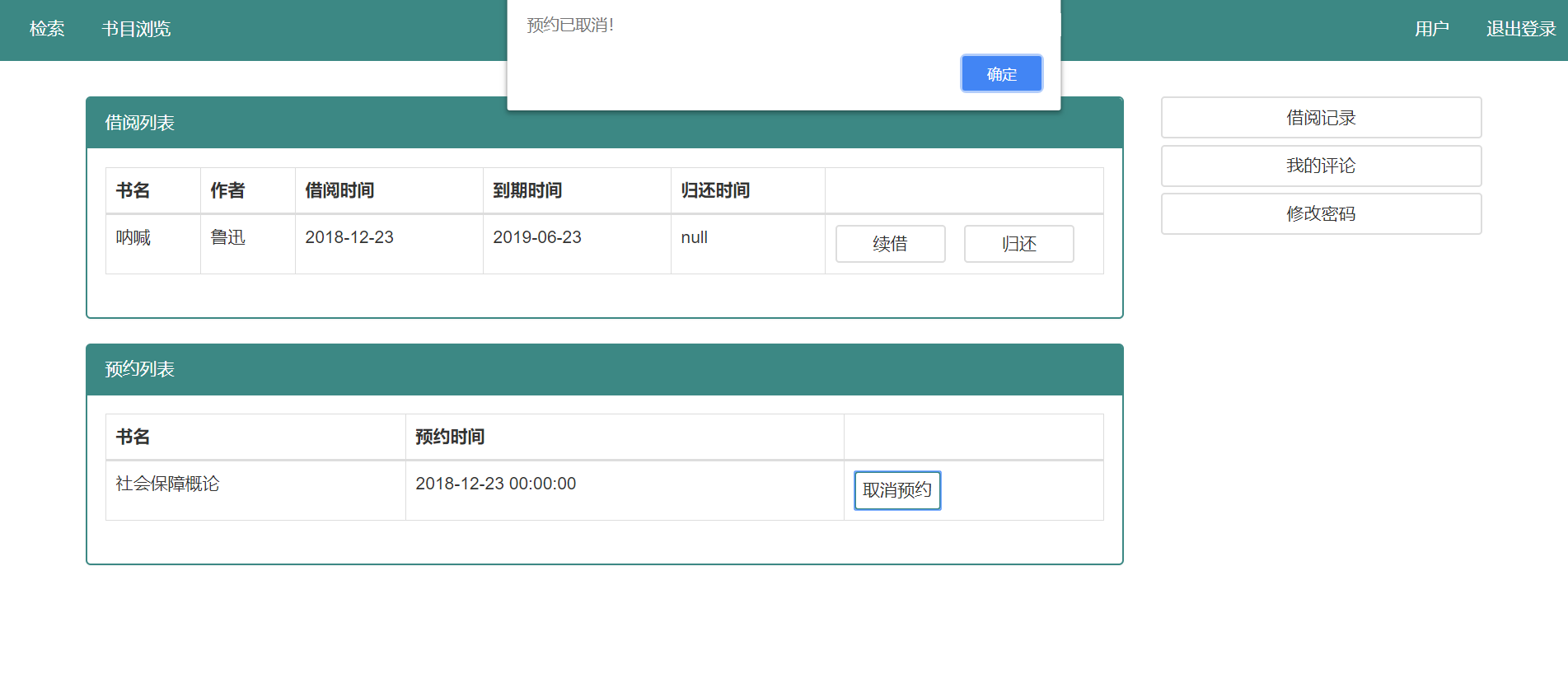
1. 通过session值判断用户是否登录
2. 连接数据表appointment、book查看用户的预约详情
3. 计算预约记录总数
4. 返回数据



###### 取消预约

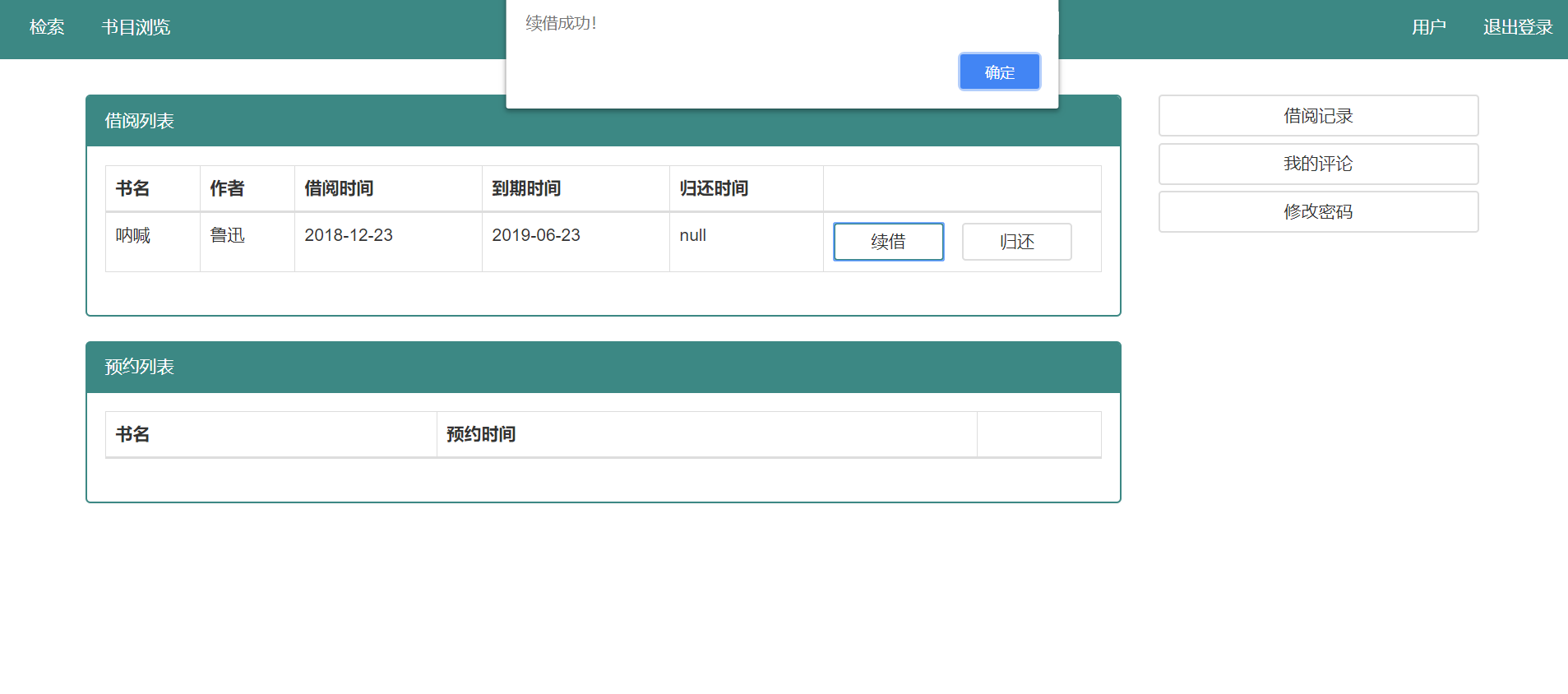
用户可以取消之前的预约

1. 通过session值判断用户是否登录
2. 连接数据表appointment，根据索引号删除该记录
3. 取消成功



###### 续借图书

用户可以续借图书，每续借一次，应还日期加30天

1. 通过session值判断用户是否登录
2. 连接appointment表，查看当前要续借的书籍是否有人预约
3. 如果有人预约则不可以续借，返回错误提示
4. 如果没有人预约则在borrowed表中查看该书是否过期
5. 如果该书过期则不可以续借，返回错误提示
6. 如果该书没有过期，则在borrowed表中将续借次数+1，应还日期+30天

###### 还书

用户可以还书

1. 通过session值判断用户是否登录
2. 连接borrowed表查看该书是否过期
3. 如果借阅过期则不能还书，需要在管理员处缴纳罚款，返回错误提示
4. 如果没有过期，更新borrowed表中的归还时间，将book表中相应书的book\_quantity+1
5. 连接appointment查询有无读者预约这本书
6. 如果有读者预约，则在预约通知表中插入该书数据

6.3管理员子系统

###### 普通管理员登录退出功能

根据管理员提交的信息，包括账号、密码和验证码，进行初步的提交格式确认后，然后调用数据库manager表的数据，判断是否存在该管理员；然后判断密码是否正确；如果密码正确，再判断输入的验证码是否正确，若正确，则将用户id保存在session中，并记录管理员的等级，返回用户提示信息“登录成功”，根据管理员等级进入相应的管理员页面。否则根据登陆失败原因返回用户相关提示信息。

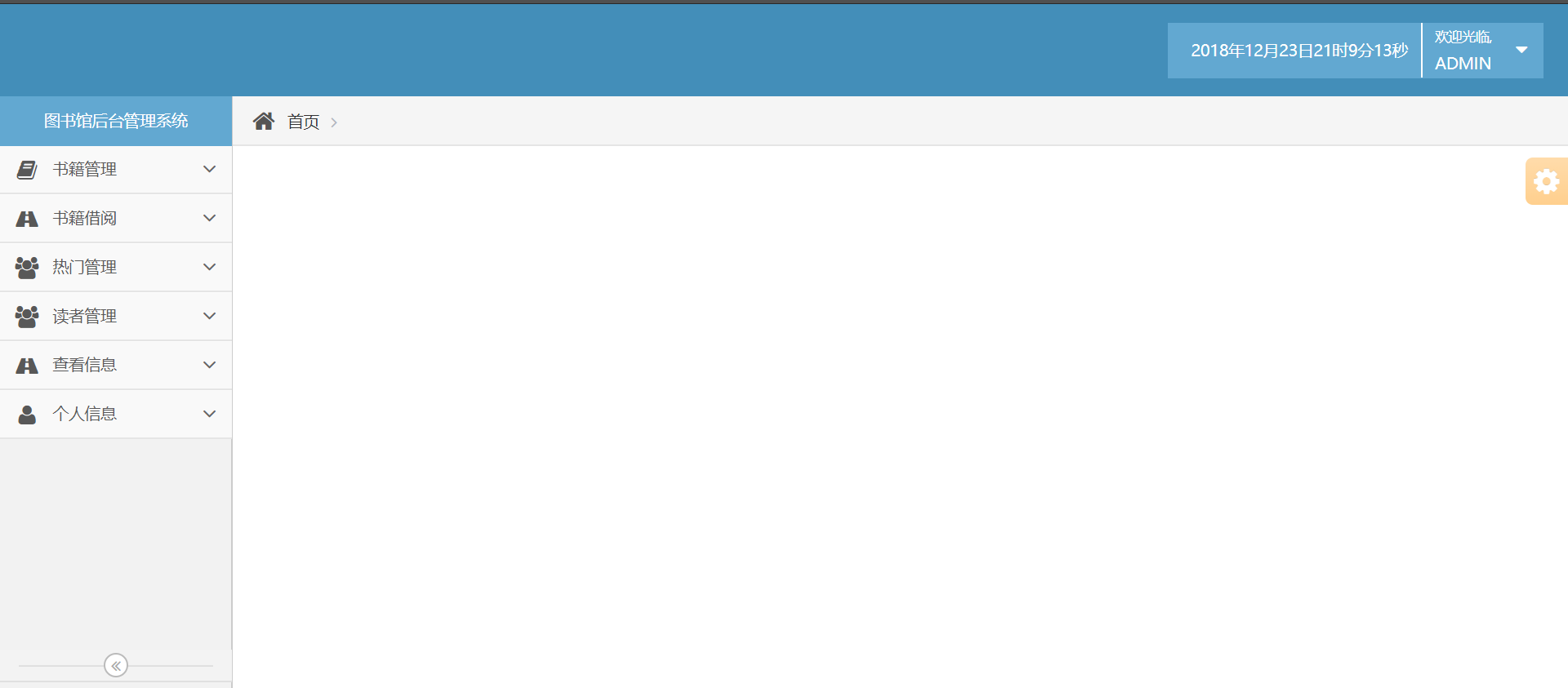
登录成功后，左侧为菜单，对应该管理员可执行的操作。上方导航栏的下拉框，点击可选择退出，清空session中的man\_id,退出系统，返回至后台的登录页。

即使是普通管理员，不同等级的普通管理员的管理权限也不一样，具体可看下方截图演示：



登录成功后进入首页

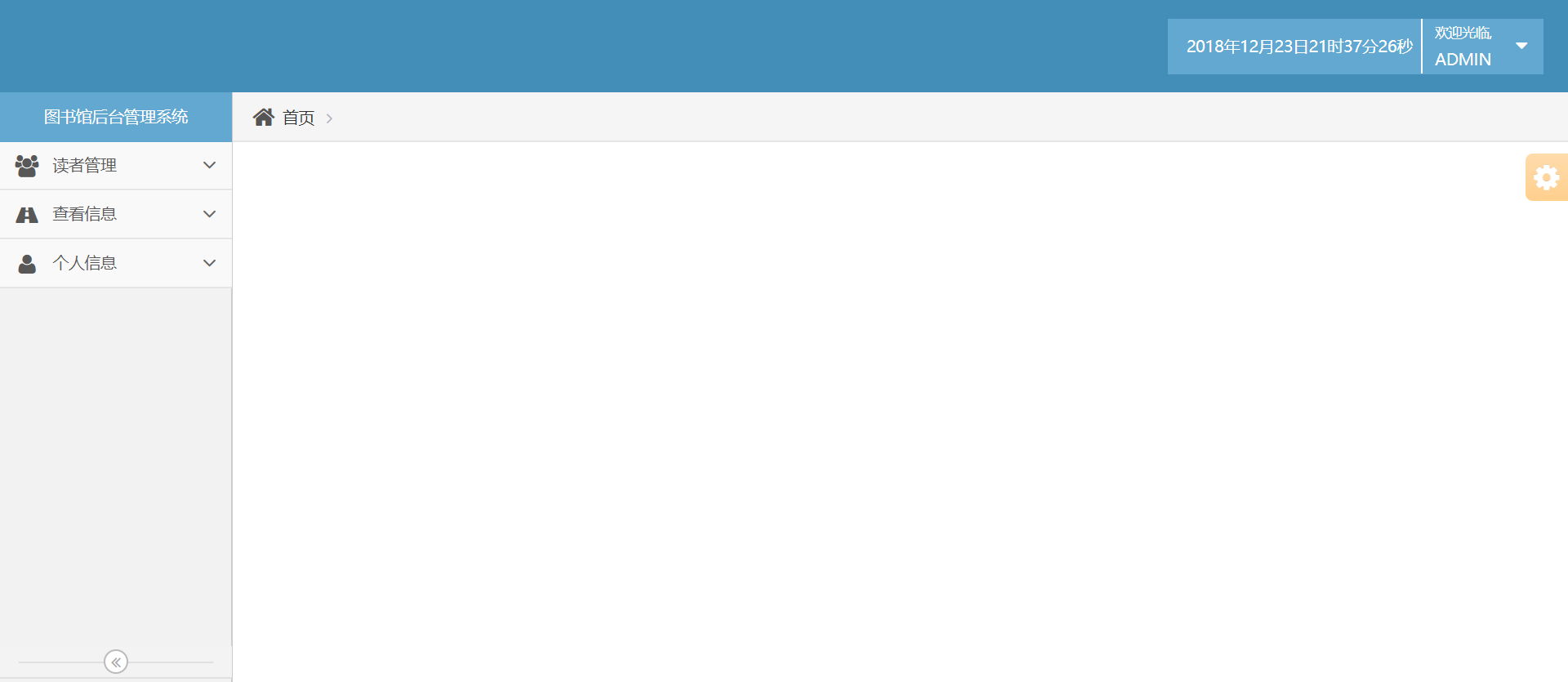
一级管理员：拥有管理书籍和读者的所有权限



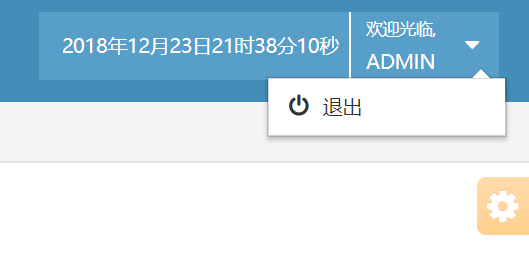
二级管理员：拥有管理书籍的所有权限



三级管理员：拥有管理读者的所有权限



退出

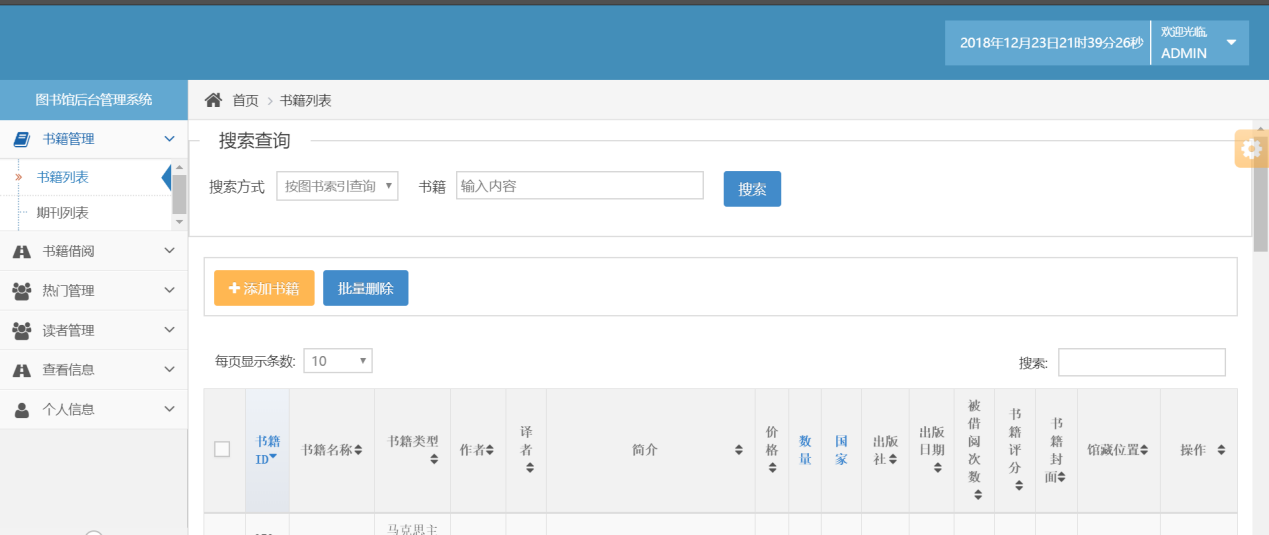


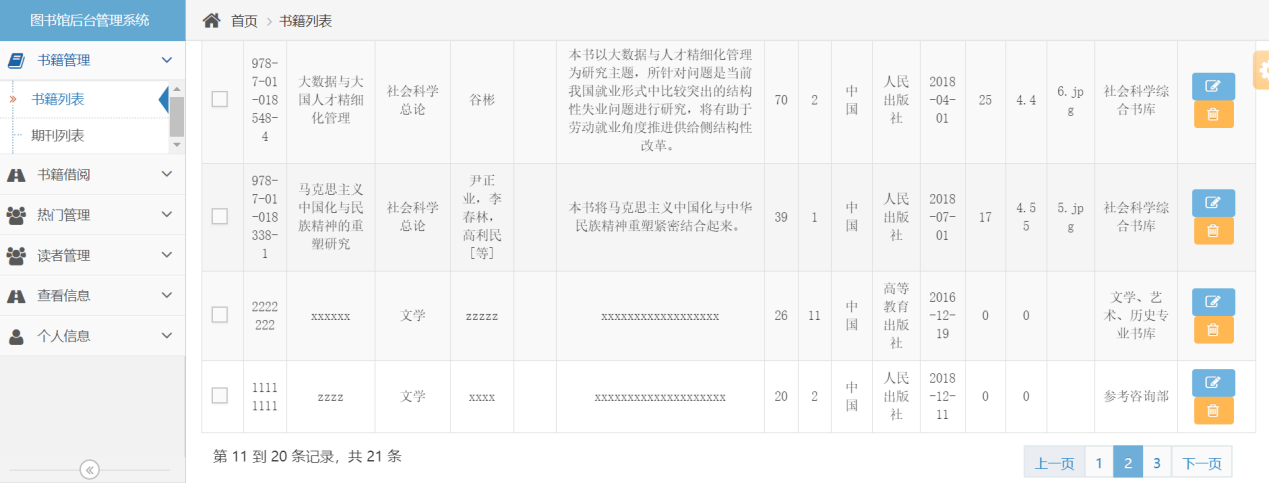
###### 管理员书籍管理功能

书籍管理，主要有添加，修改，删除，批量删除，搜索查询的功能。

* 全部图书显示功能

读取book表单的全部内容，并与book\_type表，publish表，country表，location表连接，从而显示具体的图书类型，出版社，国家，馆藏位置。将所有内容返回至前端，以表单的信息全部显示。并且具有分页显示记录的功能，并且可以自主选择每页显示的记录数目。





右下角具有翻页功能

可选择每页显示的记录数

表格各项名称那里可以点击箭头选择排序方式。比如：根据评分由上至下进行排序等。

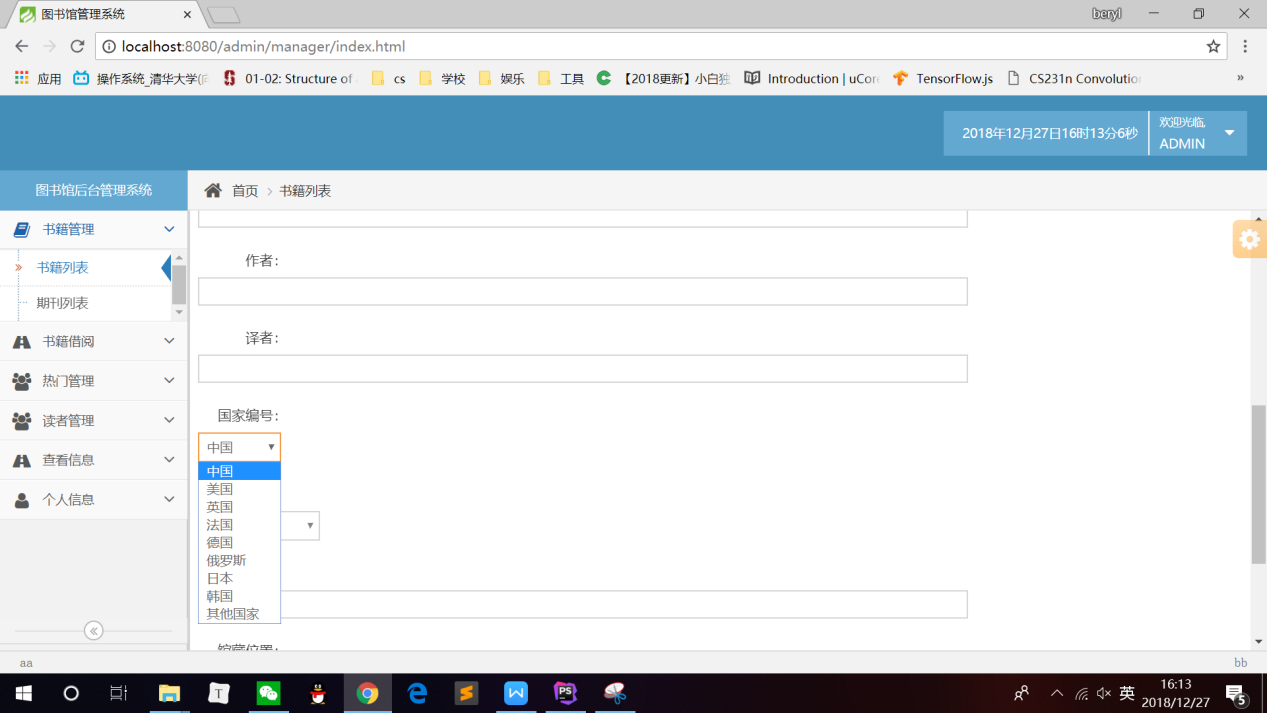
* 添加功能

在全部图书的页面，点击添加图书的按钮，进入到添加图书的页面。输入图书的信息，图书的索引号、名称、价格、数量等（译者和封面可以选择不填，其他必填），主要通过输入框和下拉框进行信息输入，前端的数据传至后台。首先判断数据是否符合要求，不符合，返回错误信息；符合，首先，根据book\_id判断，图书是否已经存在，若已存在，返回错误提示信息；不存在，则将图书信息，插入数据库book表中，返回插入成功提示信息，并回到booklist页面。

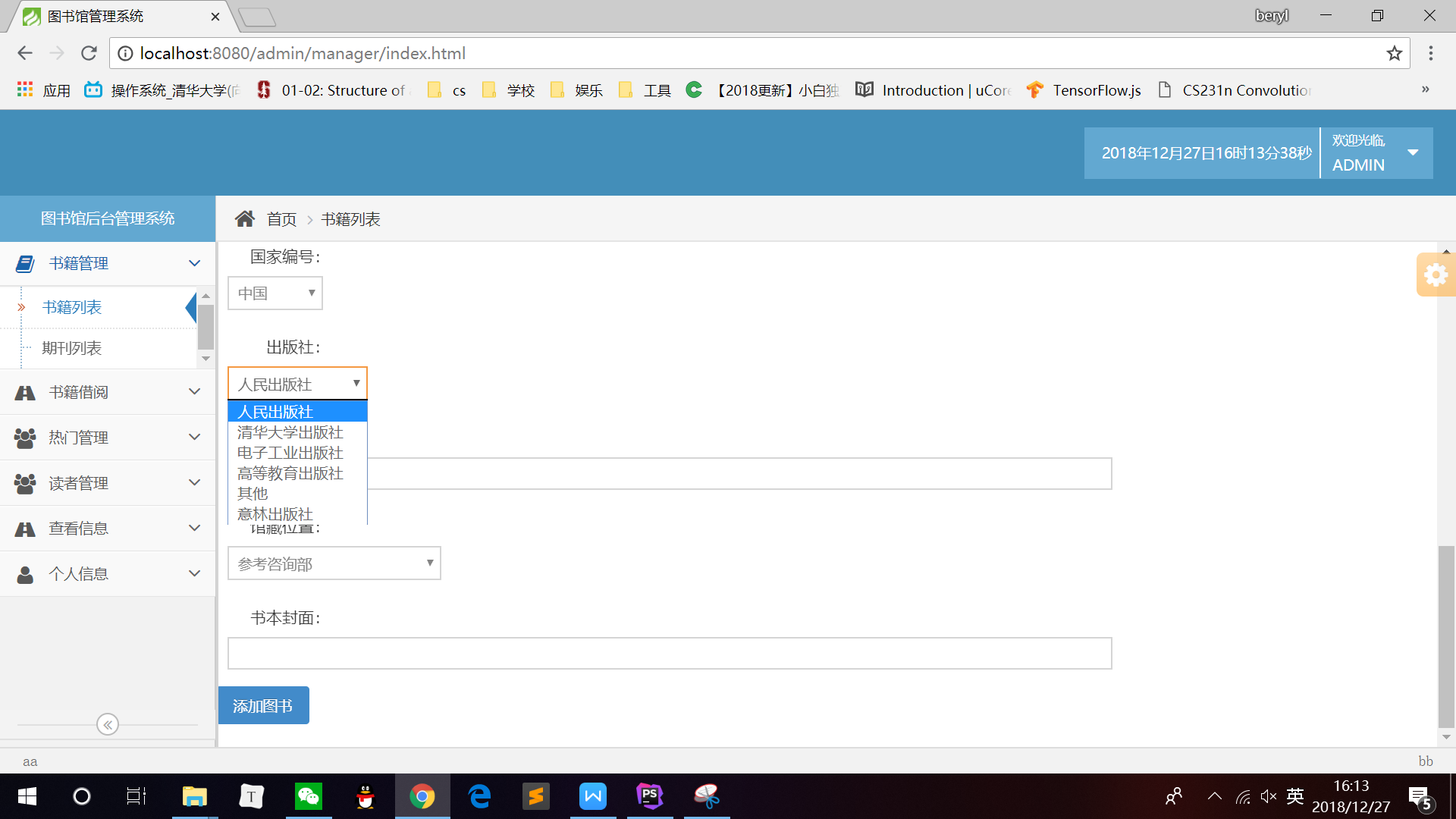
书籍类型选择：



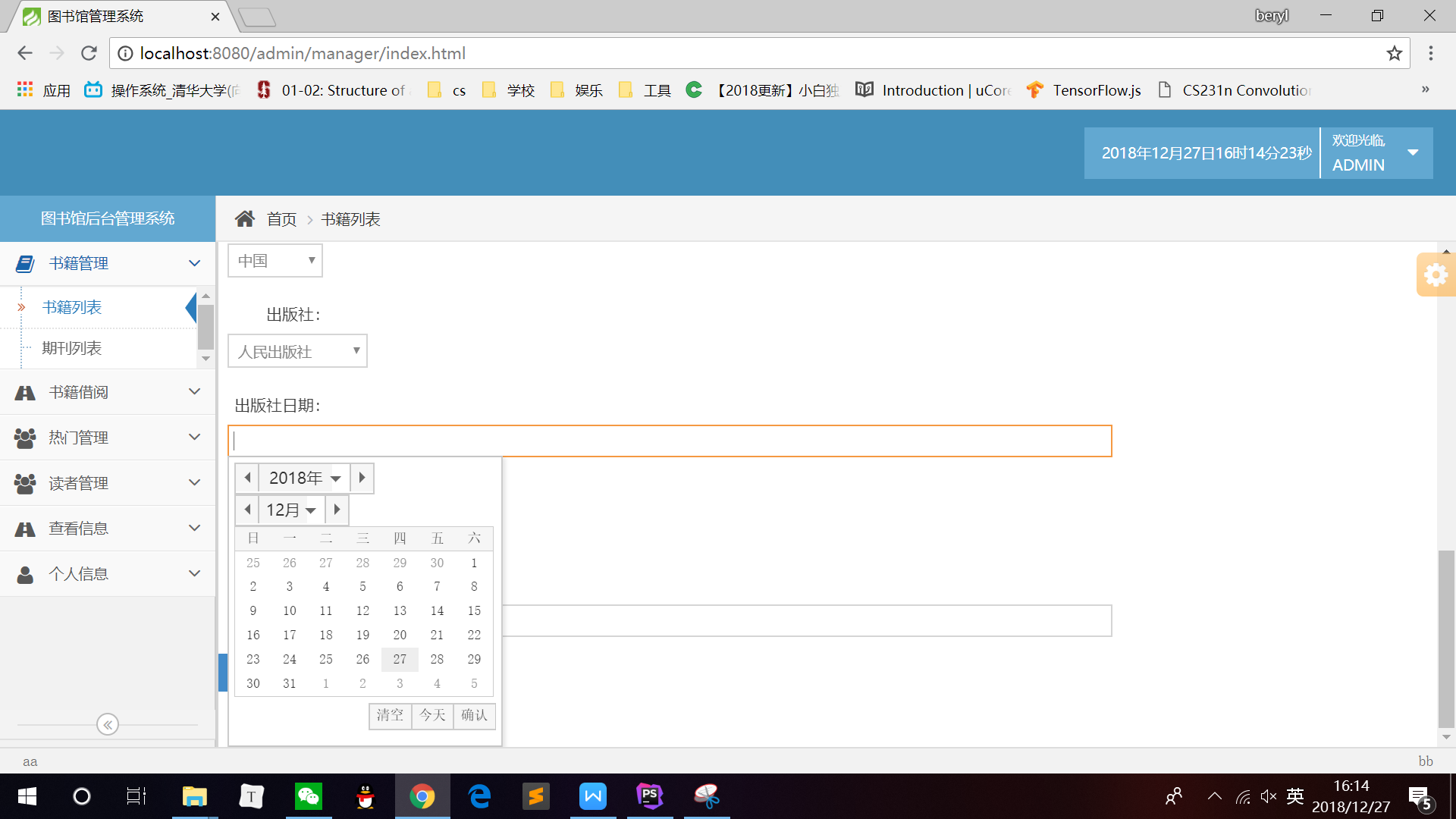
国家选择：



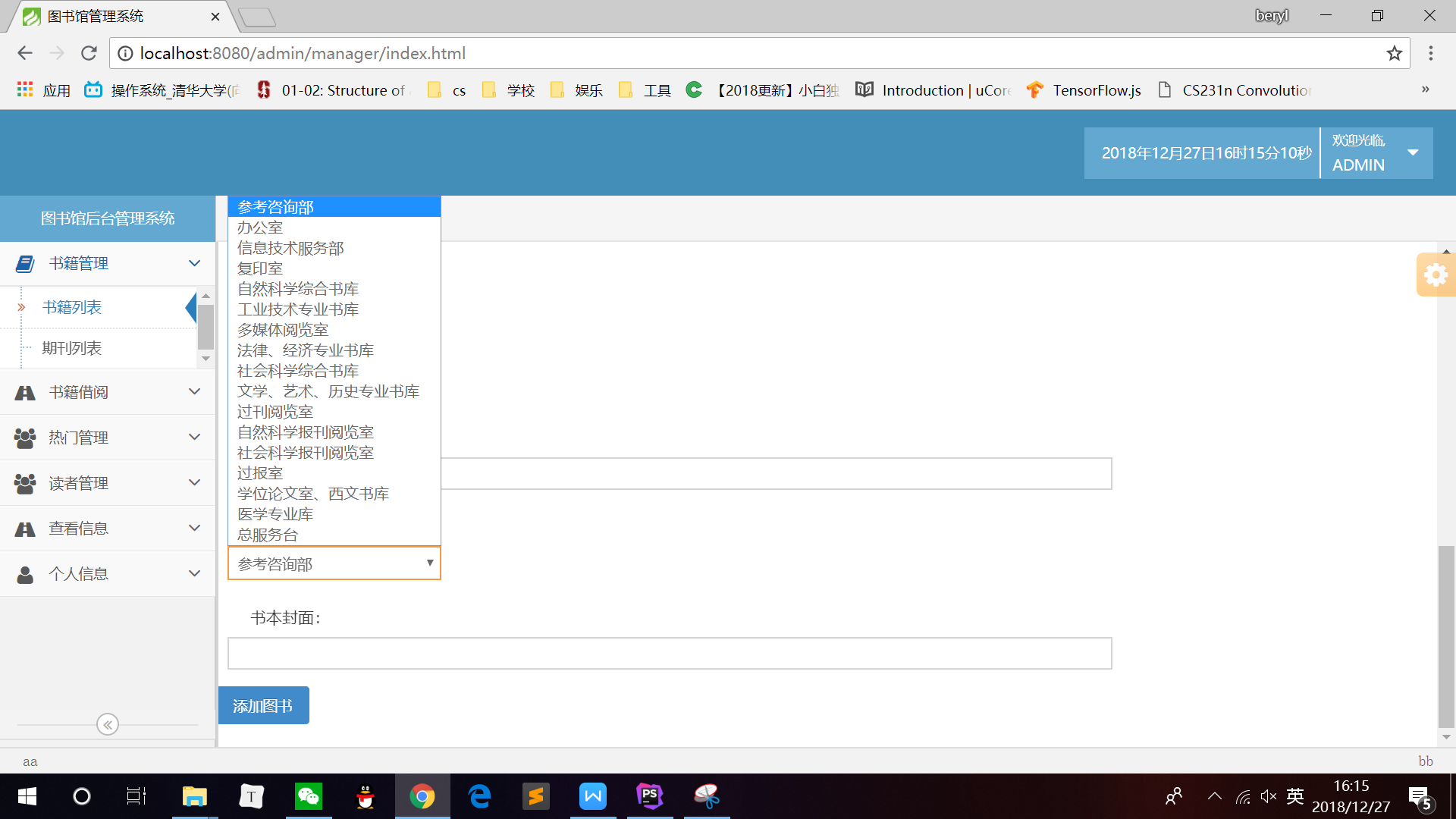
出版社选择：



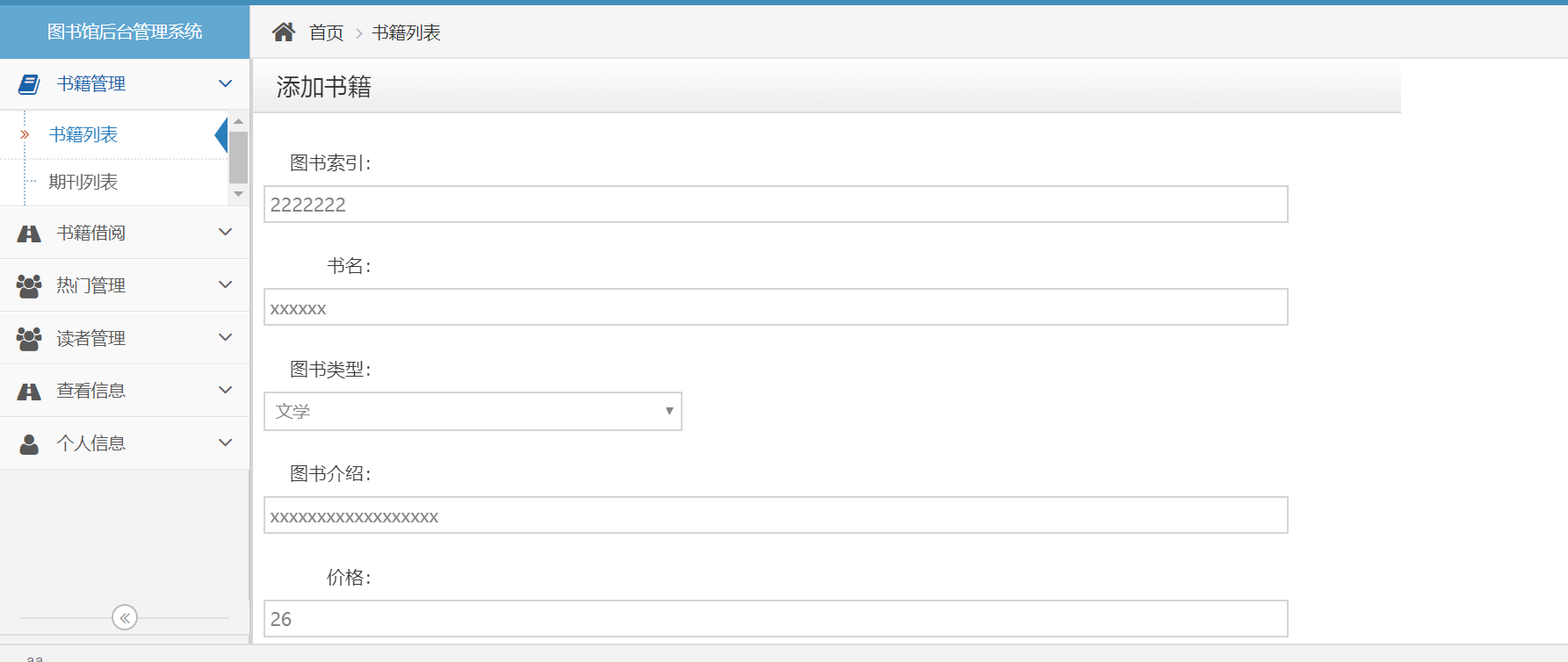
日期：



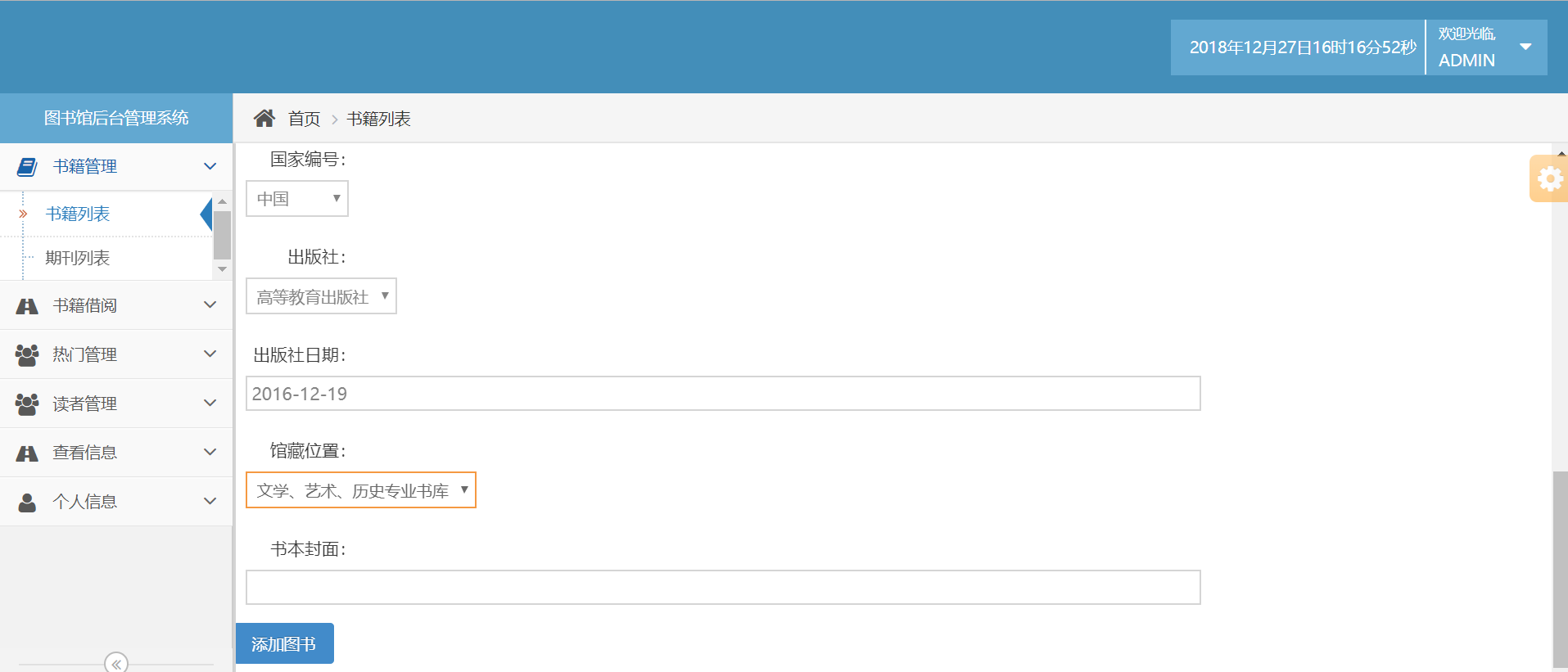
馆藏位置：



其他的信息为直接输入：



其中，也有像书本封面信息可以选择不填写，此时，系统会将默认封面补充在数据库对应的信息中。



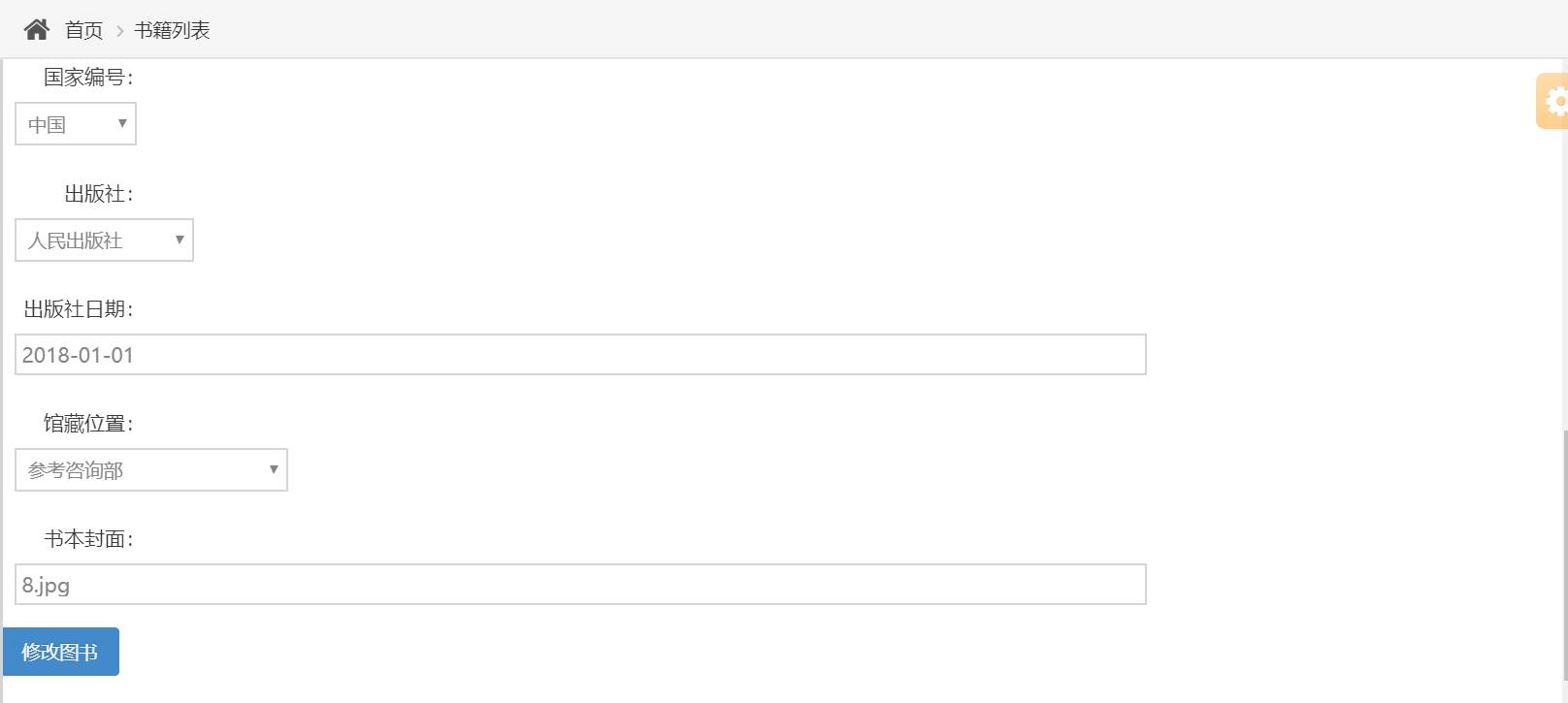
添加的记录结果展示：



* 修改功能

点击图书展示页面，每条图书记录的操作部分的编辑按钮，进入编辑页面，可以对图书除去book\_id之外的信息进行修改，点击修改按钮，将对应图书的book\_id以及输入的修改内容返回到后台，通过book\_id在book表中找到之前的图书记录，将该记录进行更新。若成功更新，返回成功提示信息，并回到booklist页面。





* 删除功能

点击图书展示页面，每条图书记录的操作部分的删除按钮，将对应图书的book\_id返回到后台，根据book\_id在book表中找到相关的图书记录，删除该记录。若成功删除，返回成功提示信息，并回到booklist页面。

* 批量删除功能

每条图书记录的最左侧有勾选框，勾选需要要删除的图书，点击批量删除，以数组形式，返回全部勾选的book\_id，将book表中的对应的图书记录全部删除。若成功删除，返回成功提示信息，并回到booklist页面。



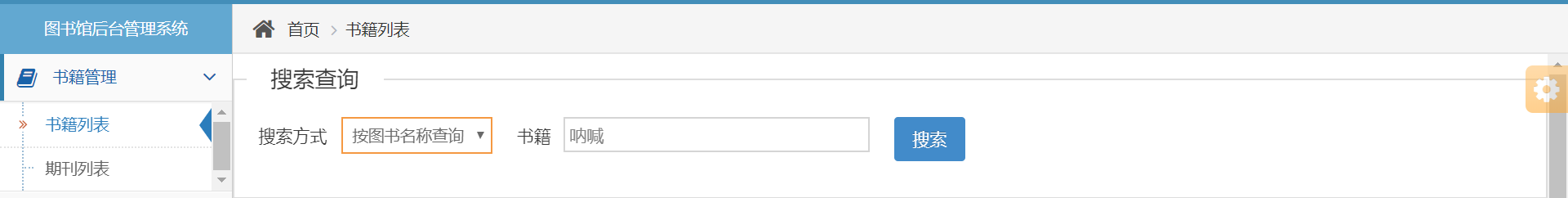
* 搜索查询功能

查询分为两种方式，按图书索引查询和按图书名称查询，输入查询的方式以及查询的图书，点击查询按钮，数据传至后台，根据输入的选择方式，进行查询。

按索引号查询，根据book\_id查找book表中的内容，若找到，返回查找的结果，显示出来，并可继续对查找到的图书进行删除，修改等操作。



ii）按书名查询，根据book\_name查找book表中的内容，若找到，返回查找的结果，显示出来，并可继续对查找到的图书进行删除，修改等操作。





###### 期刊管理功能

主要有添加，修改，删除，批量删除，搜索查询的功能。

* 全部期刊显示功能

读取book表单的全部内容，并与book\_type表，publish表location表连接，从而显示具体的期刊类型，出版社，馆藏位置。将所有内容返回至前端，以表单的信息全部显示。并且具有分页显示记录的功能，并且可以自主选择每页显示的记录数目。





右下角具有翻页功能

可选择每页显示的记录数

表格各项名称那里可以点击箭头选择排序方式。

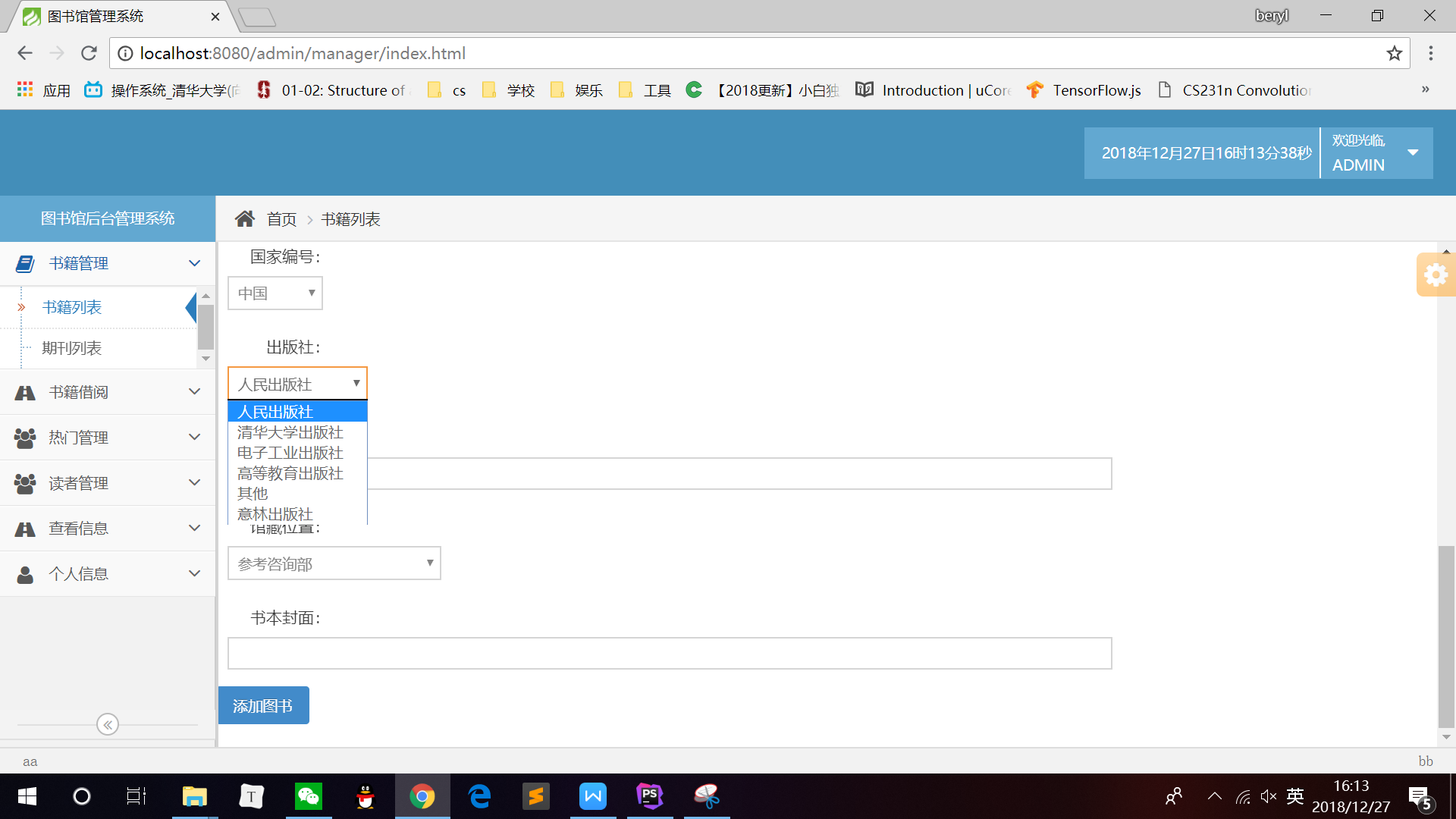
* 添加功能

在全部期刊的页面，点击添加期刊的按钮，进入到添加图书的页面。输入期刊的信息，期刊书的索引号、名称、价格、数量等（附注和封面可不填，其他信息必填），主要通过输入框和下拉框进行信息输入，前端的数据传至后台。首先判断数据是否符合要求，不符合，返回错误信息；符合，首先，根据book\_id判断，图书是否已经存在，若已存在，返回错误提示信息；不存在，则将图书信息，插入数据库journal表中，返回插入成功提示信息，并回到journallist页面。

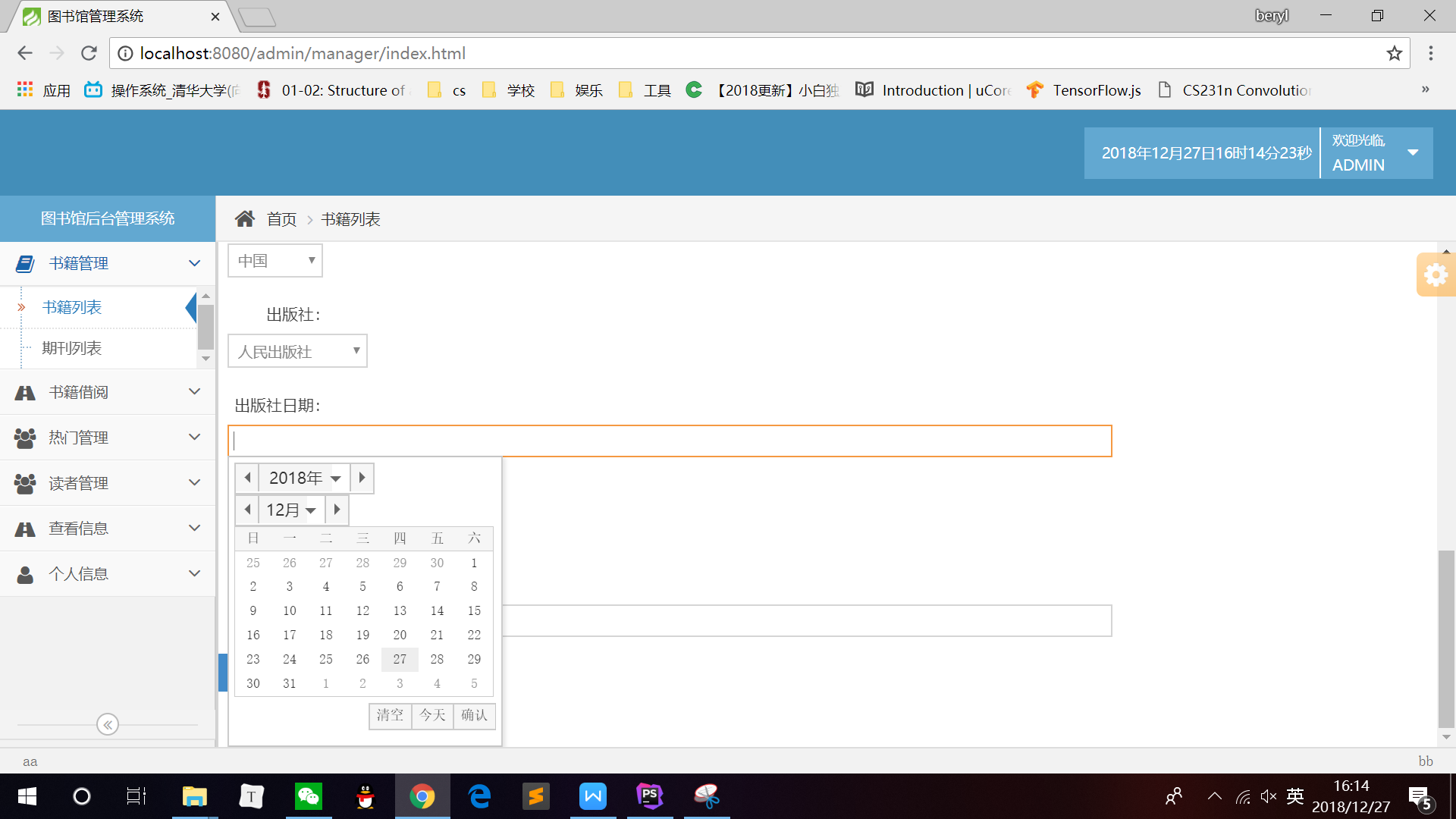
期刊类型选择：



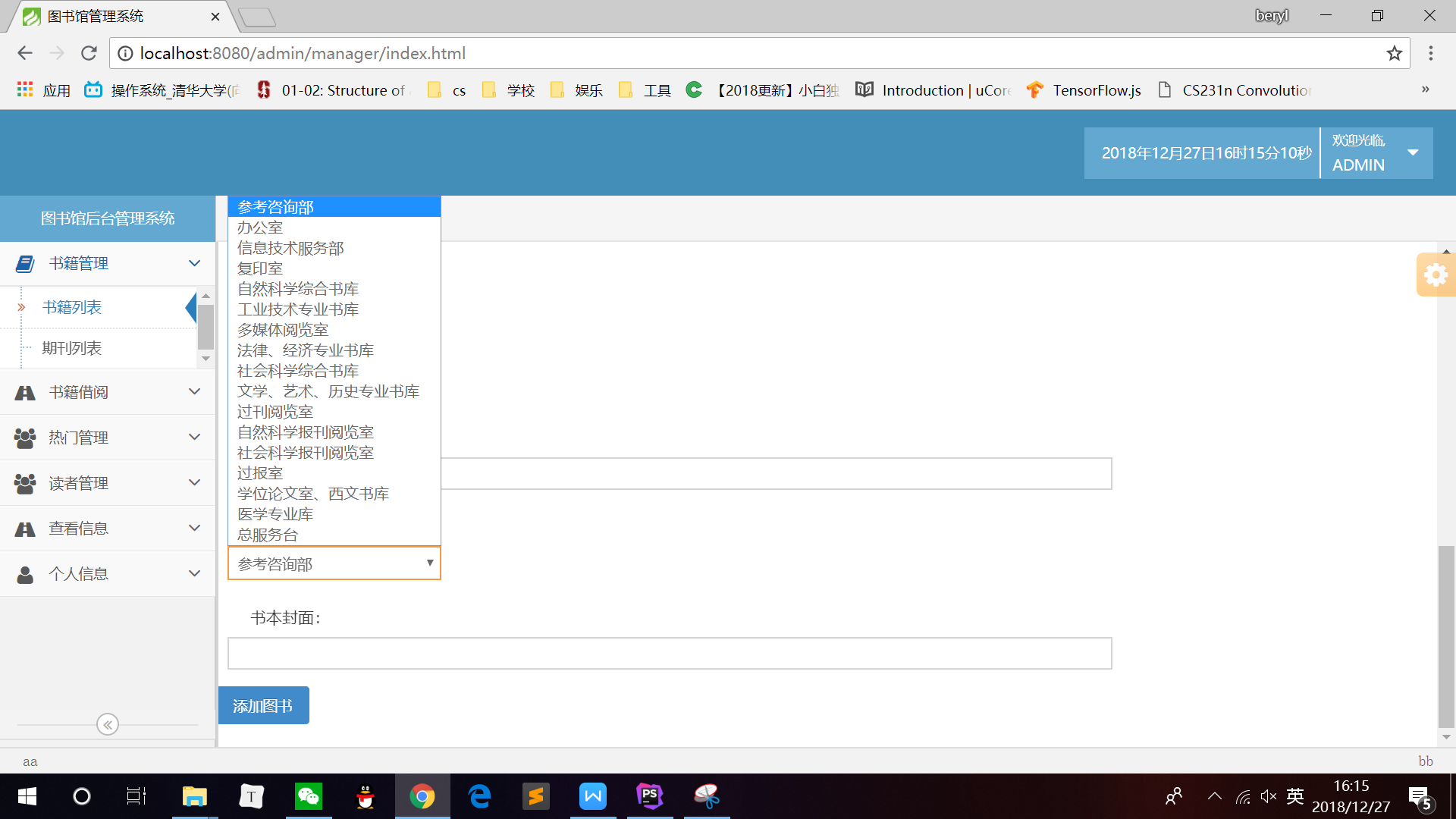
出版社选择：



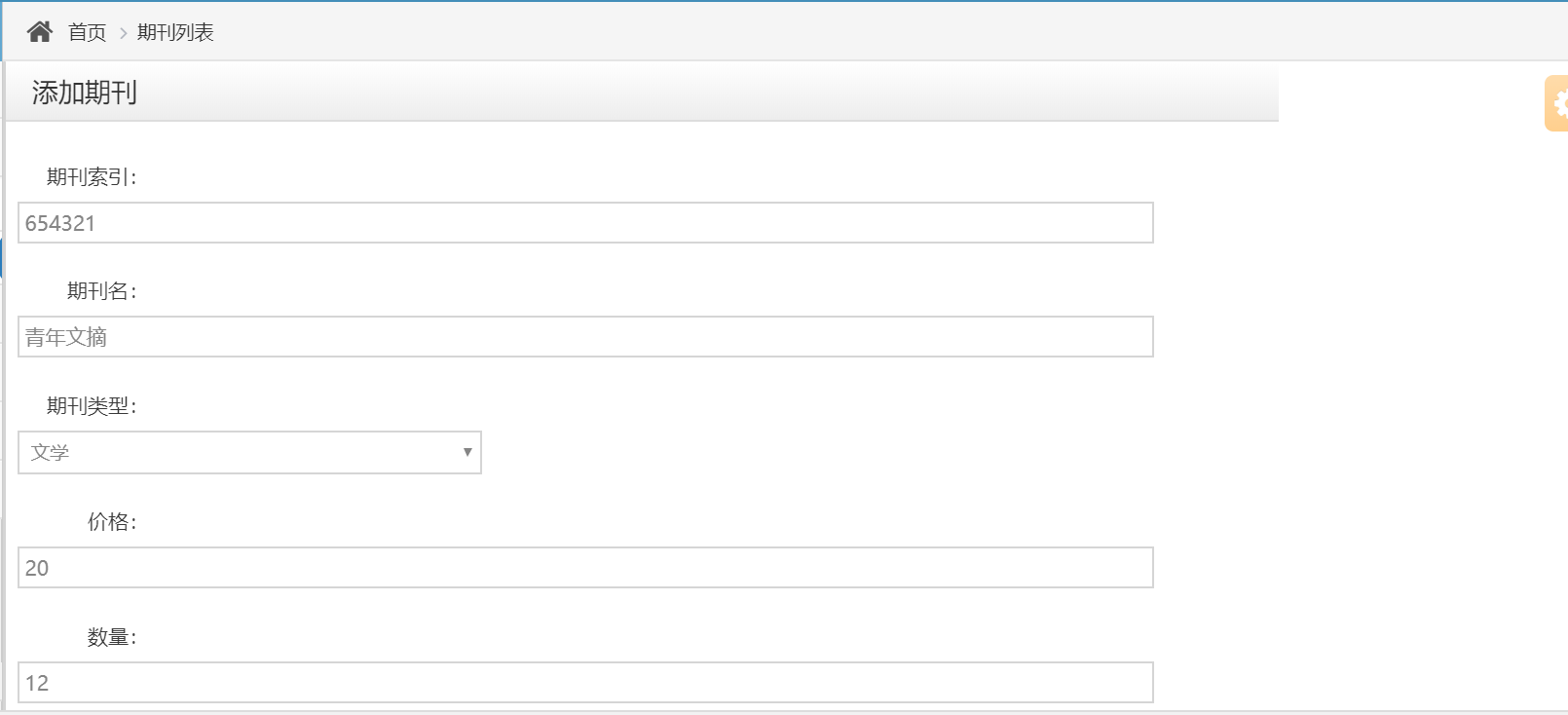
日期：

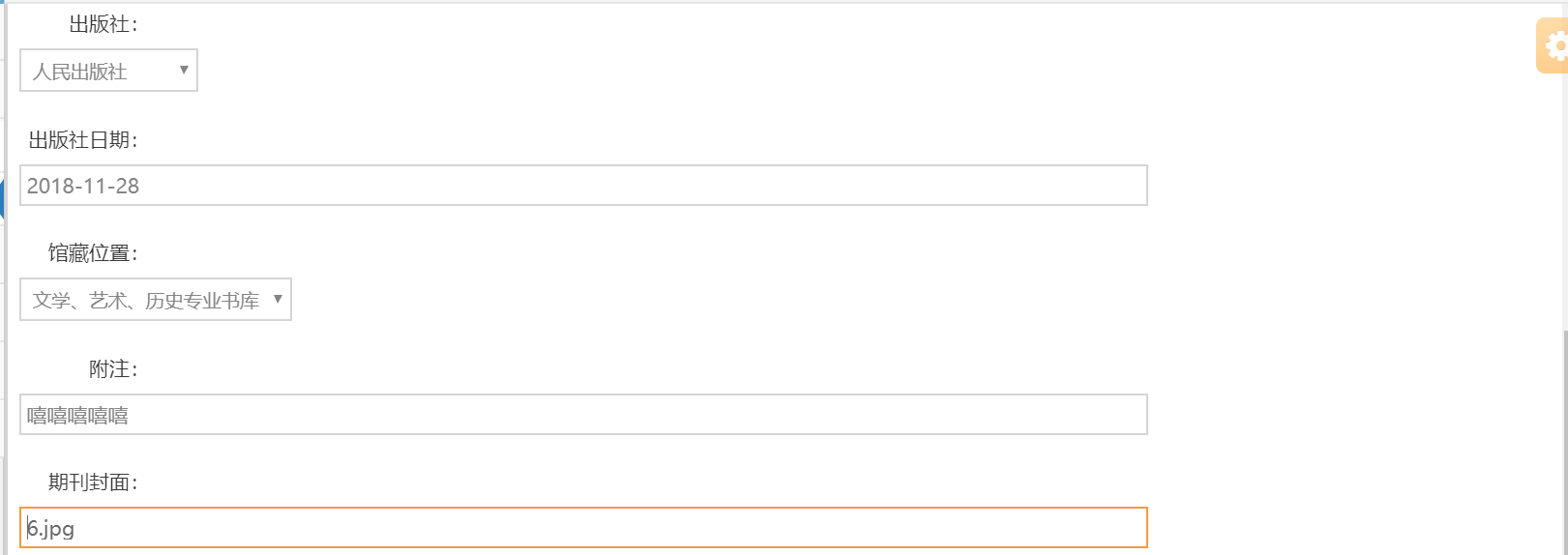


馆藏位置：



其他的信息为直接输入：



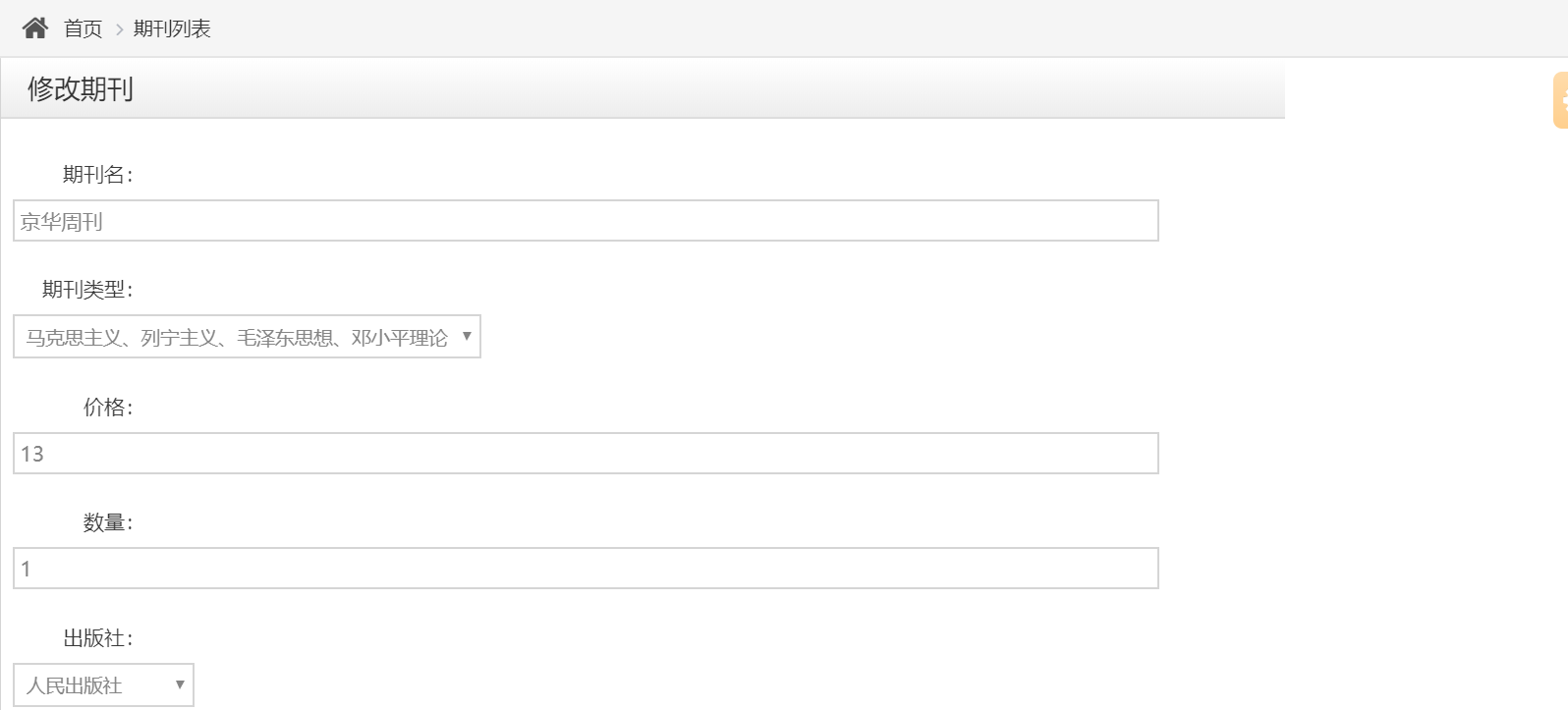


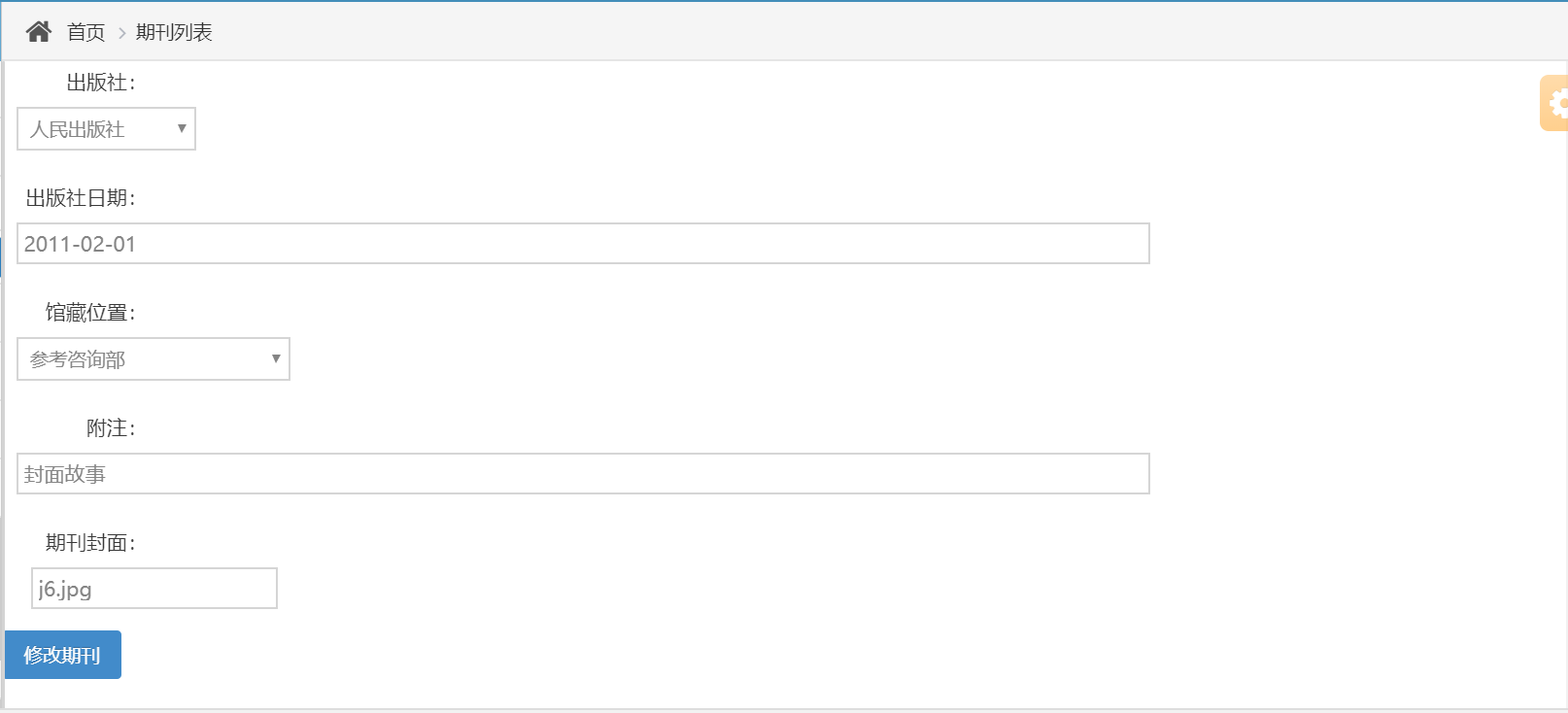
添加的记录



* 修改功能

点击期刊展示页面中每条期刊记录的操作部分的编辑按钮，进入编辑页面，可以对图书除去book\_id之外的信息进行修改，点击修改按钮，将对应期刊的book\_id以及输入的修改内容返回到后台，通过book\_id在journal表中找到之前的期刊记录，将该记录进行更新。若成功更新，返回成功提示信息，并回到journallist页面。



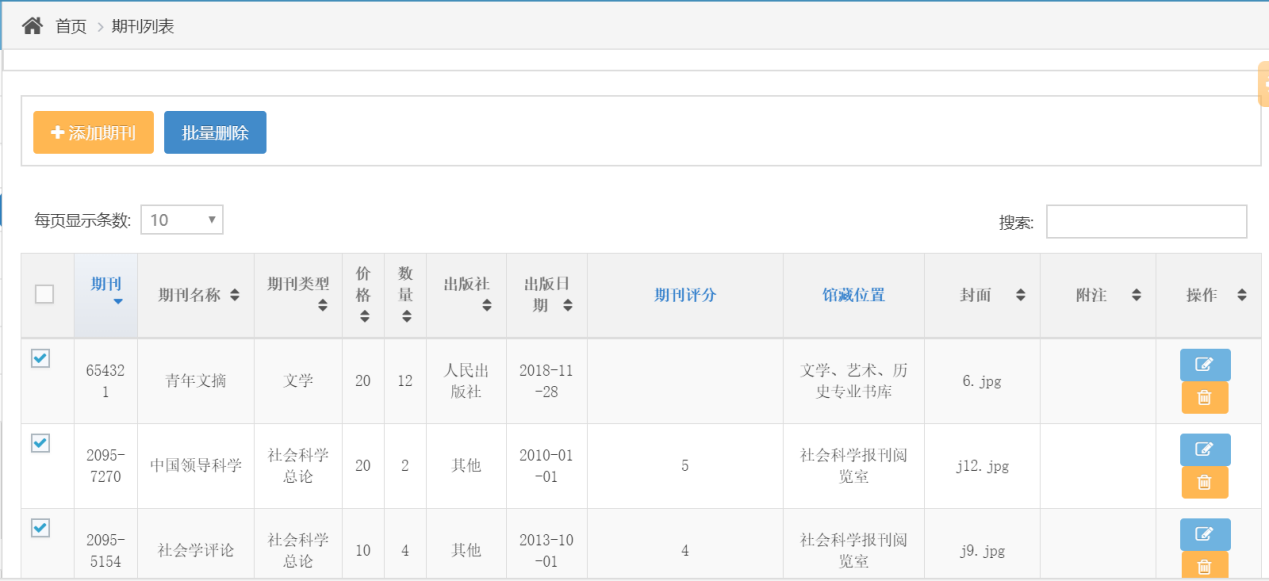


* 删除功能

点击期刊展示页面中每条期刊记录的操作部分的删除按钮，将对应期刊的book\_id返回到后台，根据book\_id在journal表中找到相关的期刊记录，删除该记录。若成功删除，返回成功提示信息，并回到journallist页面。

* 批量删除功能

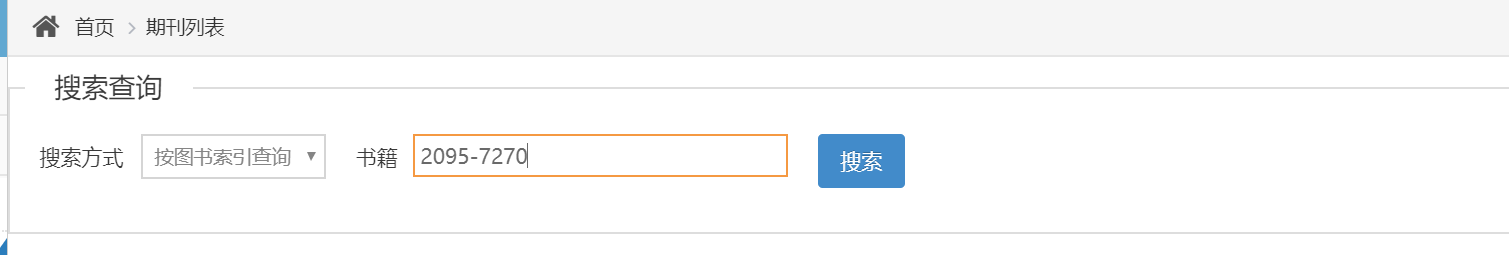
每条期刊记录的最左侧有勾选框，勾选需要要删除的期刊，点击批量删除，以数组形式，返回全部勾选的book\_id，将journal表中的对应的期刊记录全部删除。若成功删除，返回成功提示信息，并回到journallist页面。



* 搜索查询功能

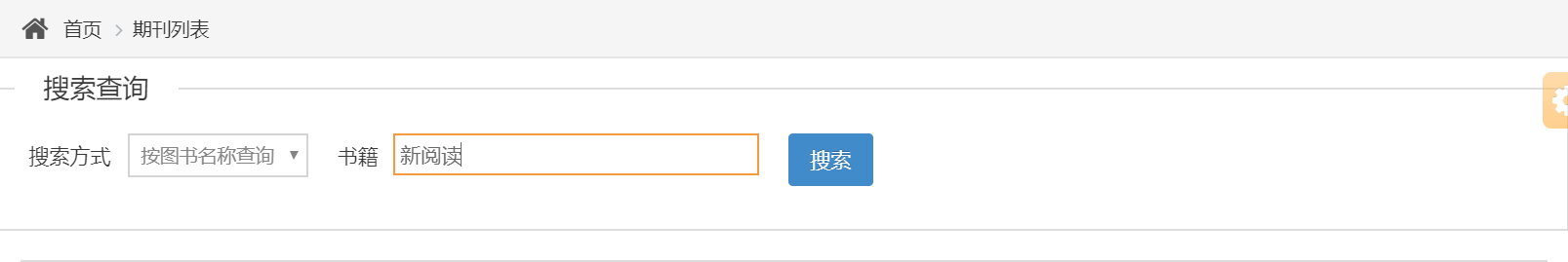
查询分为两种方式，按期刊索引查询和按期刊名称查询，输入查询的方式以及查询的期刊，点击查询按钮，数据传至后台，根据输入的选择方式，进行查询。

1. 按索引号查询，根据book\_id查找journal表中的内容，若找到，返回查找的结果，显示出来，并可继续对查找到的期刊进行删除，修改等操作。





ii）按期刊名查询，根据jou\_name查找journal表中的内容，若找到，返回查找的结果，显示出来，并可继续对查找到的期刊进行删除，修改等操作。





###### 热门图书更新功能

热门图书分为书籍和期刊管理，列表显示之前的热门图书、热门期刊表单的内容。

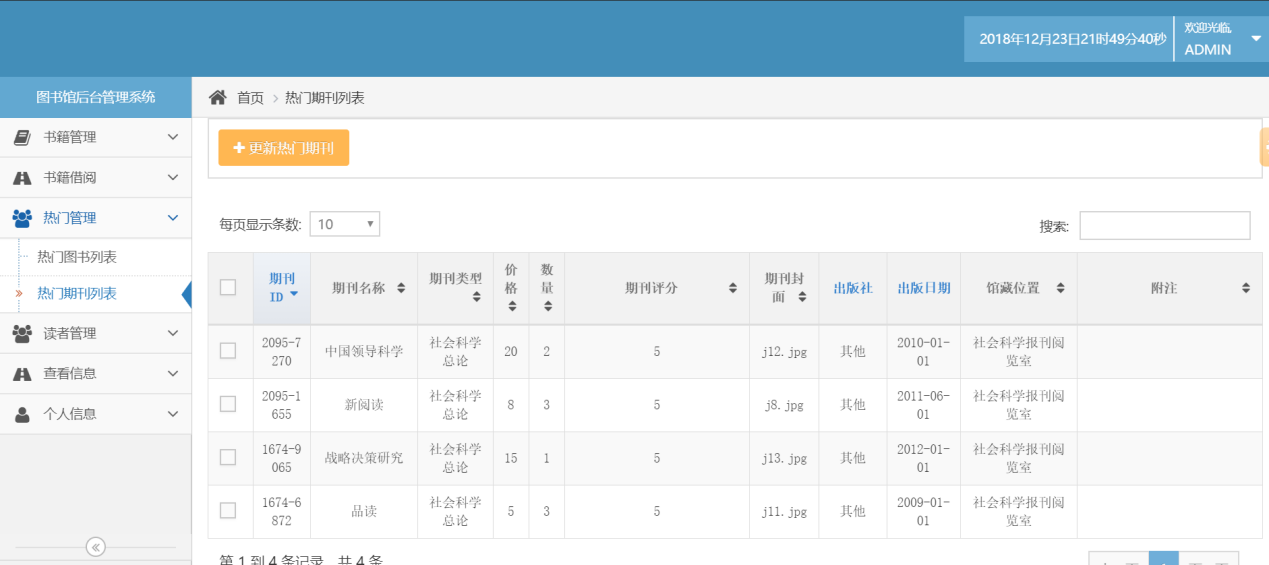
热门图书更新，根据book表中，图书的评分以及图书的借阅数量倒序排序，选择前5本作为热门图书，并更新热门图书表。

热门期刊的更新，根据journal表中，期刊的评分倒序排序，选择前5本作为热门期刊，并更新热门期刊表。

热门书籍



热门期刊



###### 书籍借阅管理功能

* 借阅记录表

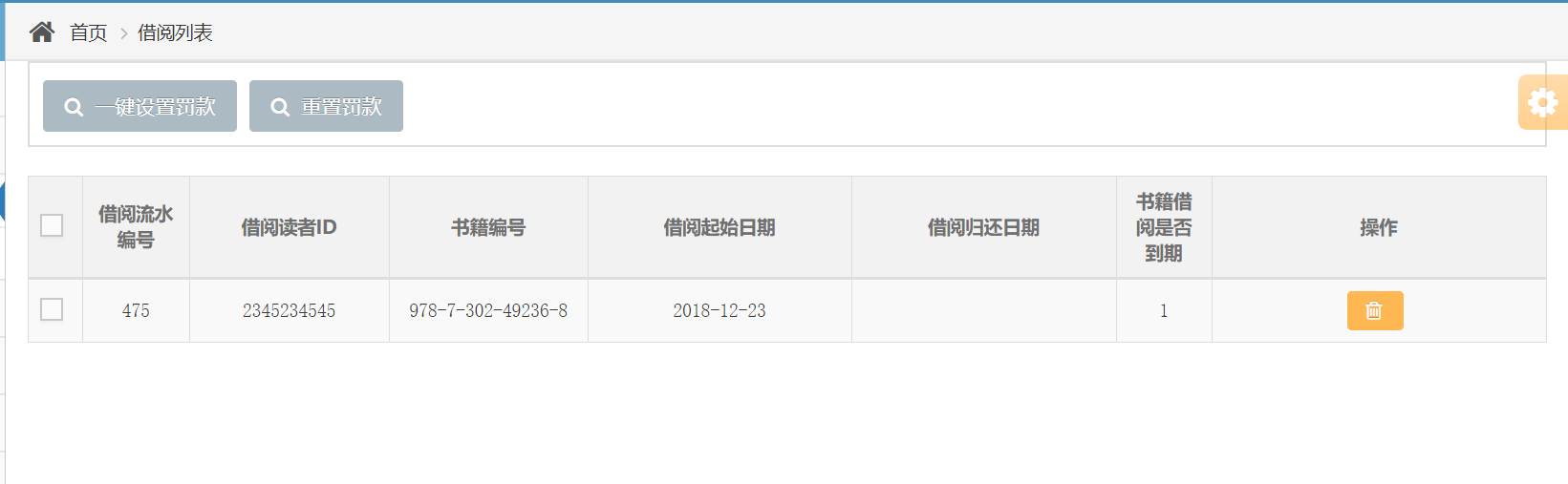
查看借阅记录表的全部信息，通过select查询，将borrowed表的内容返回至前端，并以表单的形式呈现。



* 筛选逾期未还记录

点击筛选按钮，调用后台，筛选逾期未还的方法，根据归还时间与借阅时间的差值或者未归还时，根据当前时间与借阅时间的差值，判断借阅记录是否逾期，若逾期，将更新该条记录，将result改为1，表示逾期，最后将所有逾期还书的记录返回前端显示。点击该页面上的罚款按钮，可以对逾期归还记录的读者进行扣费和扣除信用值，根据逾期归还的时间，按一天1元，5天扣除1信用值的标准进行处理，修改相应读者的罚款信息和信用值。

点击筛选逾期记录按钮后的结果：



点击设置罚款按钮将会对对应的读者进行扣费处理

点击重置会清除上一步的罚款操作，修改数据库中的内容

* 删除借阅记录信息

删除借阅记录，通过点击删除按钮，将borrowed\_id传给后端，查找对应的记录，并删除。

* 批量删除借阅记录

通过勾选框，将选中的borrowed\_id传给后台，删除记录。

* 预约记录表

查看所有预约记录，通过select查询，将appointment表的内容返回至前端，并以表单的形式呈现。



* 删预约记录信息

通过点击删除按钮，将app\_id传给后端，查找对应的记录，并删除。

* 批量删除预约记录

通过勾选框，将选中的app\_id传给后台，删除记录。

###### 读者管理功能

* 读者注册审核功能

管理员查看读者注册的审核表的所有内容，对读者的注册信息进行审核，如果符合管理员的要求，管理员同意审核，将读者的注册信息传至后台，先判断reder\_id是否存在在读者信息表中，若已存在，返回错误信息，不存在，将审核表中的读者信息插入到读者信息表中，并将审核记录表中的审核结果更新为1，表示审核成功。

管理员可以对审核记录进行删除。



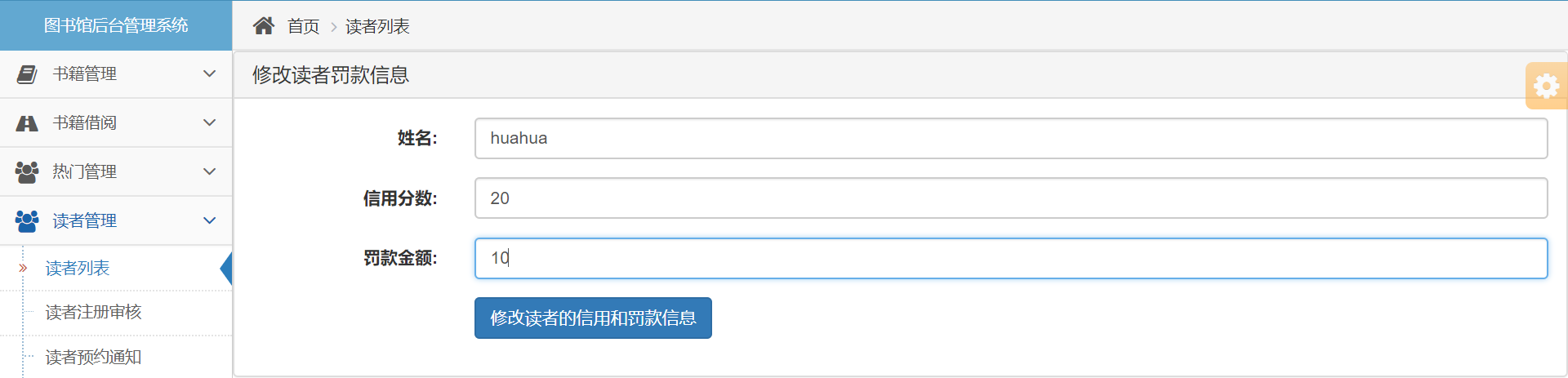
审核注册信息，将结果置为1，并加入reader表中



* 读者缴费处理功能

读者如果去管理员处上交罚款，管理员可点击编辑按钮，对读者的罚款信息进行修改。将reader\_id和输入的新的罚款信息传递给后台，根据reader\_id，查找到对应的读者，更新罚款的信息。

管理员修改罚款信息



* 读者预约成功通知表

管理员查看改表的信息，该表的内容表明，某读者预约的图书有人归还，已有库存，可以借阅。根据该表的提供的读者邮箱，管理员将发送邮件，通知读者前来借阅。



* 读者查看功能

后端直接获取reader表中的内容，以数组形式，将数据返回至前端，前端获取数据，以表单形式展示。



* 读者的删除功能

对不在使用图书馆系统的读者，管理员可以删除该类读者，分为单独删除以及批量删除两种方式。

单独删除：点击每条读者记录右侧的删除按钮，获取reader\_id，后台根据reader\_id，查找到指定记录，将该记录从数据库中删去。

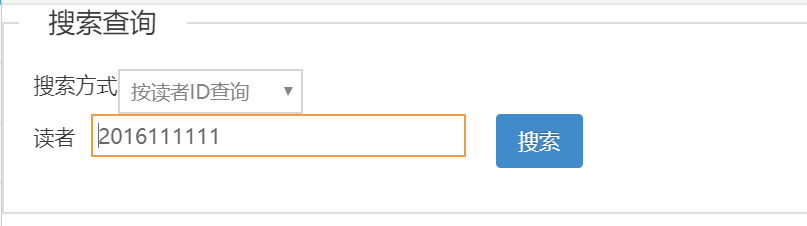
批量删除：点击勾选框，选择要删除的读者，再点击批量删除按钮，传给后台一个ids['reader\_id']数组，根据数组中包含的reader\_id,c查找reader表中相关记录，并删除。

* 读者搜索功能

前端输入搜索方式,type以及搜索的内容content,后端接收数据，判断type的值，type='1'，按索引号进行条件查询，查找相关记录，成功找到，返回数据；失败，返回错误提示信息。type='2'，按读者姓名进行条件查询，成功找到相关记录，返回数据；失败，返回错误提示信息。









###### 查看信息功能

1.图书评论

* 查看

后端直接读取comment表中的内容，以数组形式，将数据返回至前端，前端获取数据，以表单形式展示。



* 删除

单独删除：点击每条评论记录右侧的删除按钮，获取com\_id，后台根据com\_id，在comment表中查找到指定记录，将该记录从数据库中删去。

批量删除：点击勾选框，选择要删除的评论，再点击批量删除按钮，传给后台一个ids['com\_id']数组，根据数组中包含的com\_id,c查找comment表中相关记录，并删除。

2.期刊评论

* 查看

后端直接读取jou\_comment表中的内容，以数组形式，将数据返回至前端，前端获取数据，以表单形式展示。



* 删除

单独删除：点击每条评论记录右侧的删除按钮，获取com\_id，后台根据com\_id，在jou\_comment表中查找到指定记录，将该记录从数据库中删去。

批量删除：点击勾选框，选择要删除的评论，再点击批量删除按钮，传给后台一个ids['com\_id']数组，根据数组中包含的com\_id,c查找jou\_comment表中相关记录，并删除。

3.反馈信息

* 查看

后端直接获取feedback表中的内容，以数组形式，将数据返回至前端，前端获取数据，以表单形式展示。



* 删除

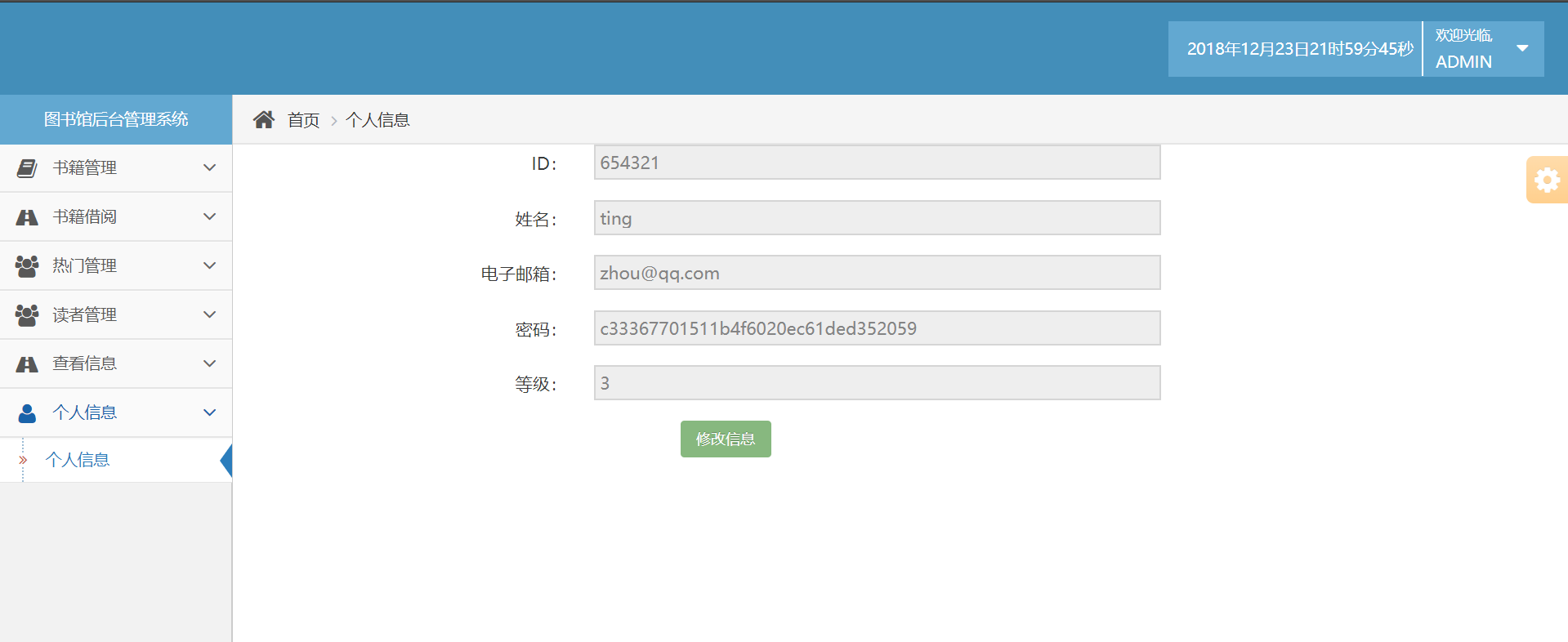
单独删除：点击每条评论记录右侧的删除按钮，获取fdb\_id，后台根据fdb\_id，在fdb\_comment表中查找到指定记录，将该记录从数据库中删去。

批量删除：点击勾选框，选择要删除的评论，再点击批量删除按钮，传给后台一个ids['fdb\_id']数组，根据数组中包含的fdb\_id,查找feedback表中相关记录，并删除。

###### 普通管理员个人信息管理功能

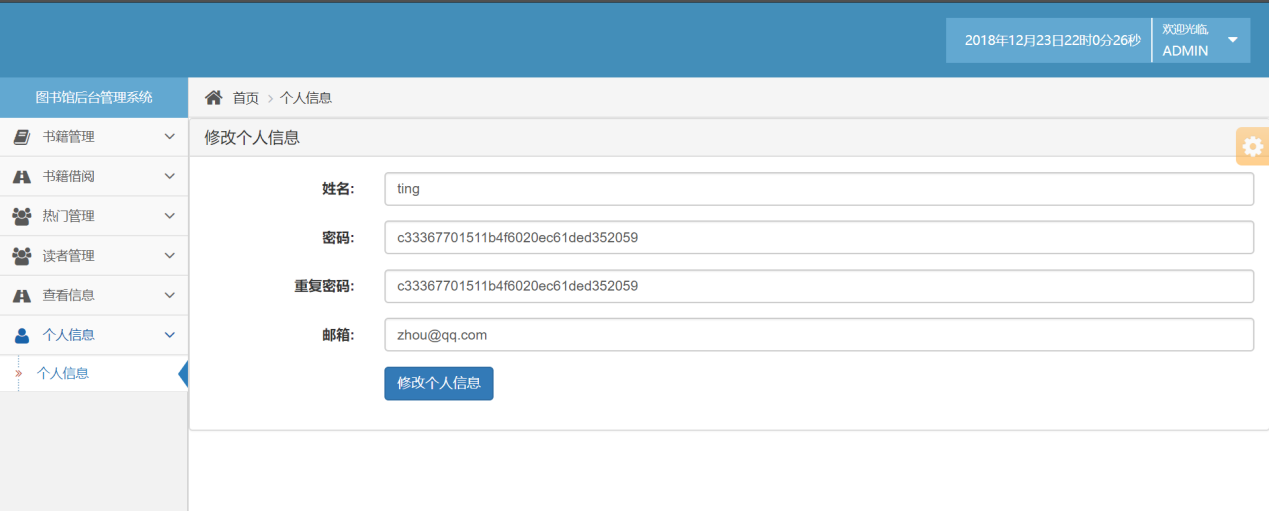
* 查看个人信息

获取登录时，session中存入的man\_id，在管理员记录表中，按man\_id，进行条件查询，找到管理员的个人信息，将数据以数组的形式返回前端，前端显示管理员的id,姓名，邮箱，密码，和等级。



* 修改个人信息

点击修改按钮键，获取登录时，session中存入的man\_id，允许修改姓名，密码和邮箱。输入需要修改的内容，另外包括重复密码，后端读取输入的数据，判断密码和重复密码是否一致，一致则根据man\_id查找到管理员记录，将信息修改，保存到数据库，返回成功修改的提示信息；不一致，返回错误提示信息。对修改的密码，自动进行加密。



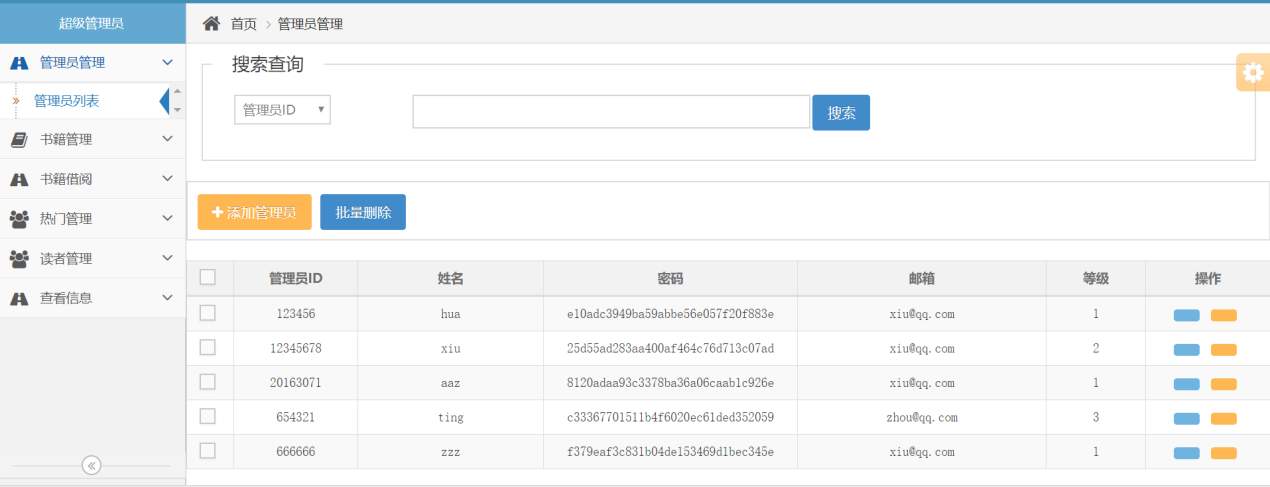
###### 超级管理员对管理员的管理功能

* 超级管理员首页



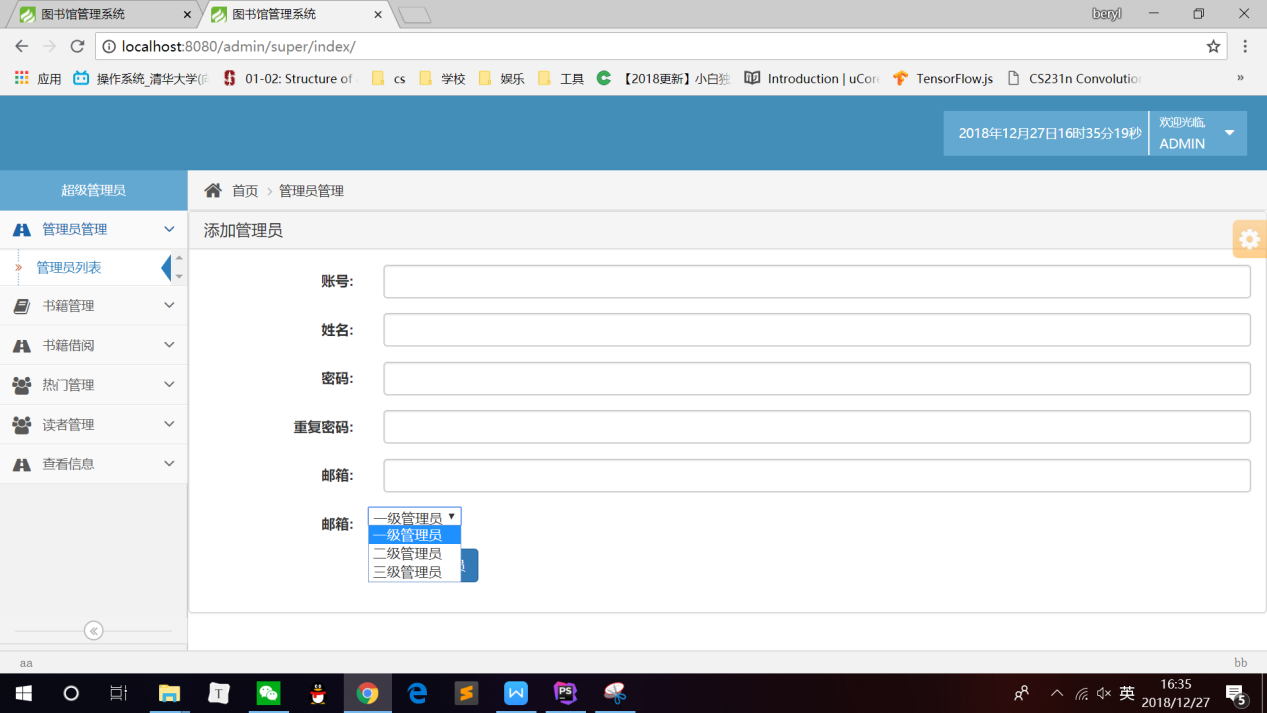
* 查看所有管理员

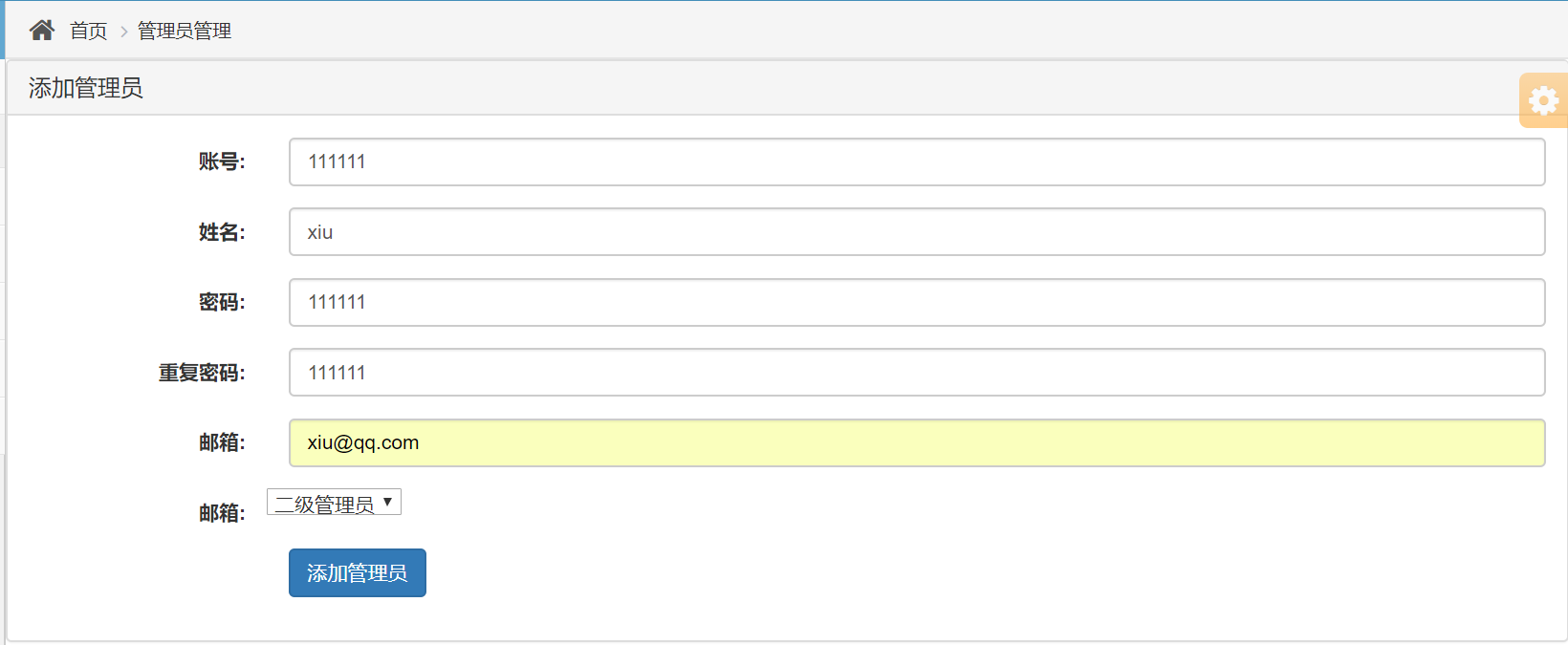
后端直接获取manager中的内容，以数组形式，将数据返回至前端，前端获取数据，以表单形式展示



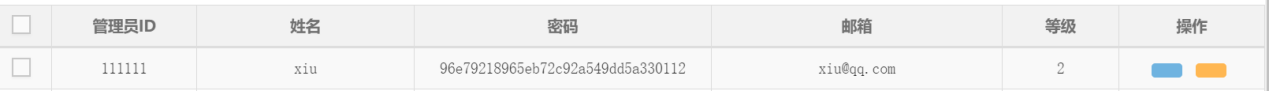
* 添加管理员

超级管理员添加管理员，在添加页面输入，管理员id、姓名、密码、重复密码、邮箱、等级，将数据传递给后端，后端用验证器先进行判断所接收的数据是否符合要求，不符合，返回错误提示信息；符合要求，进一步，判断man\_id在manager表中是否存在，不存在，则将数据，插入到manager表中，反之，返回错误提示信息，管理员已存在。同时添加管理员时，会对密码自动加密。





添加成功的记录，对密码自动加密



* 修改

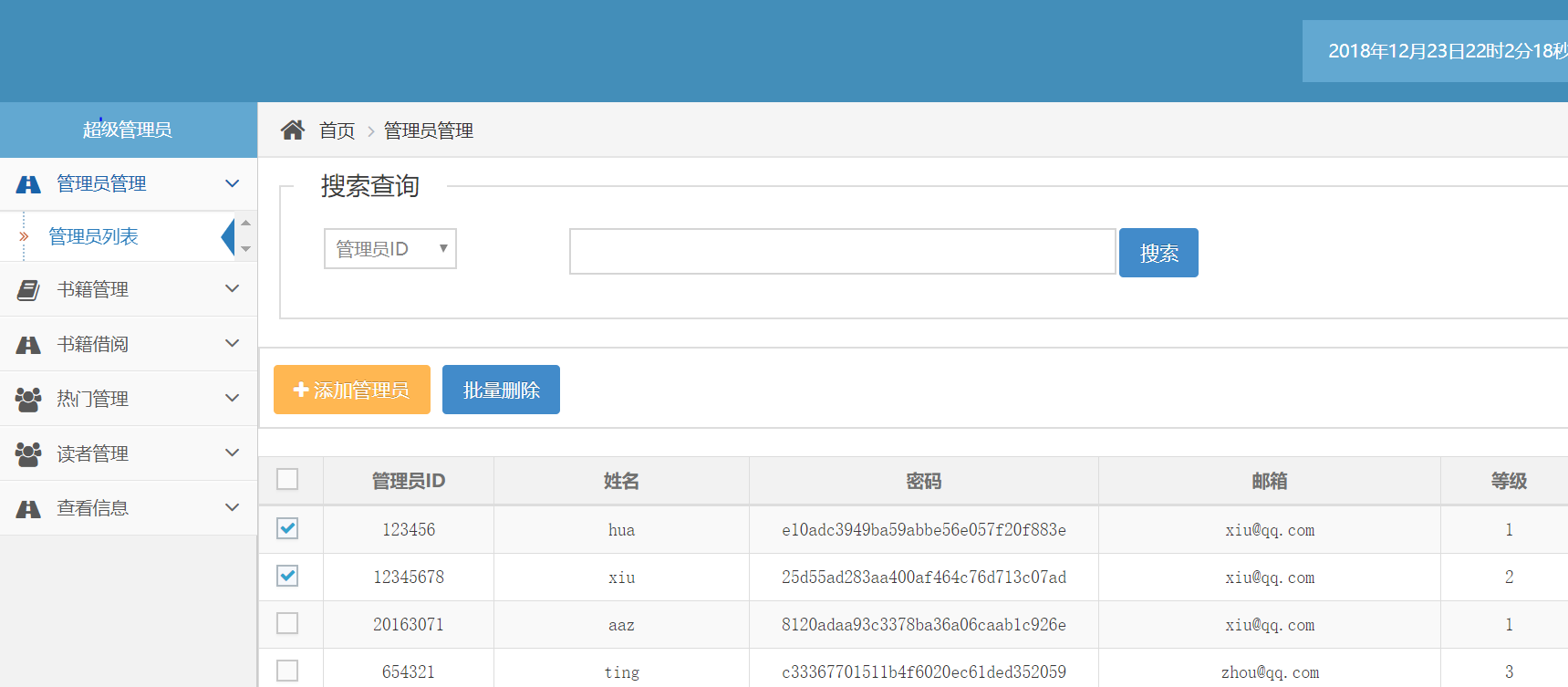
点击修改按钮，超级管理员可以修改管理员的除去id之外的全部信息，输入修改的内容，将输入的数据和man\_id传递给后端，验证数据是否符合要求，符合要求，则按man\_id查找到对应的管理员记录，按修改的内容更新该管理员信息。对密码进行加密



* 删除

单独删除：点击每条管理员记录右侧的删除按钮，获取man\_id，后台根据man\_id，在manager表中查找到指定记录，将该记录从数据库中删去。

批量删除：点击勾选框，选择要删除的管理员，再点击批量删除按钮，传给后台一个ids['man\_id']数组，根据数组中包含的man\_id,查找manager表中相关记录，并删除。



* 管理搜索功能

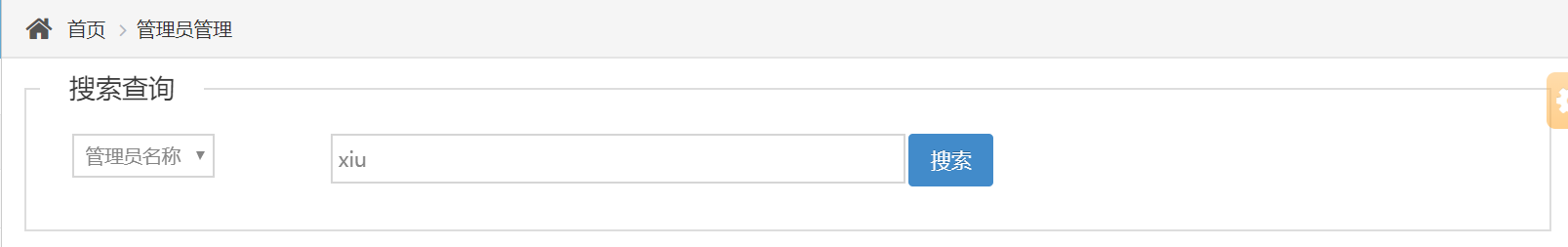
前端输入搜索方式,type以及搜索的内容content,后端接收数据，判断type的值，type='1'，按索引号进行条件查询，查找相关记录，成功找到，返回数据；失败，返回错误提示信息。type='2'，按管理员姓名进行条件查询，成功找到相关记录，返回数据；失败，返回错误提示信息。

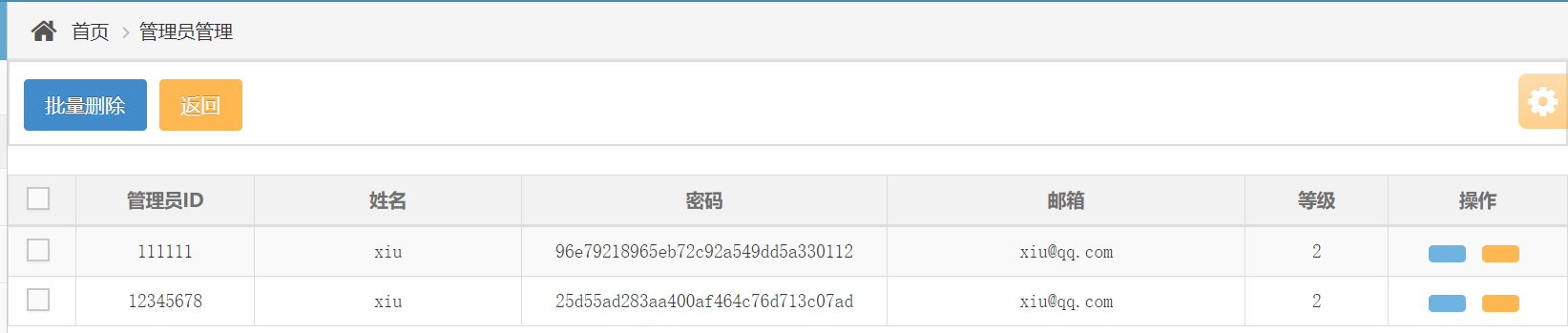
1）按id





2）按姓名





6.4前后端接口ajax

系统开发前后端分离，而前后端进行信息交互主要采用js中内置的ajax方法进行，而且ajax方法对用户十分友好，不会有翻转页面的发生，也是目前网站开发中常常采用的技术。

以下是我们网站开发前后端设计的接口，从中可以看到我们前后端数据的交互和以下错误信息返回处理等。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 功能 | 前端传入参数 | 参数描述 | 后端返回 | 参数描述 |
|  | userId,password,  type,userName,  mail,tel,college | 用户的学/工号，密码，读者类型，姓名，邮箱，电话，学院编号 | isSucceed  error | 1:成功0:不成功  1:userId已存在 |
| 用户登录 | userId,password | 用户名，密码 | isSucceed  error | 1:登录成功  1:不存在2:密码错误 |
| 搜索图书 | words,identify | 检索词和检索项编号 | data[name,bookid,publish,author,content,location,book\_type,type]；  result  picture  result\_num | data中有书名，索引号，出版社，作者，简介，馆藏位置，图书类型；  1:成功0:失败无数据  图片名字  检索数目 |
| 用户提交反馈 | content | 反馈内容 | isSucceed  error | 成功  1:用户未登录 |
| 按图书类型查询 | bookType | 图书类型编号 | data[name,bookid,publish,author,content,location,type]；  result\_num  result | data中有书名，索引号，出版社，作者，馆藏位置，图书类型;  检索数目  1:成功0:失败 |
| 读者信息显示 |  |  | readers[readername,readerid,readertype,readerdate,readertel,readermail,borrowedquantity,readercredit,arrear] | readers中有姓名，学/工号，读者类型，开户日期，联系电话，邮箱，已借阅数量，信用分数，欠费情况 |
| 借阅记录显示 |  |  | borrow\_data[borrow\_id,bookid,bookname,author,location,sdate,edate,rdate,times,overed]  borrow\_num | borrow\_data中有借阅流水号，索引号，书名，作者，馆藏地，借阅时间，应还时间，还书时间，续借次数，是否到期；  借阅记录数目 |
| 预约记录显示 |  |  | myapp[appdate,bookname,bookid,]  app\_num | myapp中有预约时间，书名；  预约记录数目 |
| 评论记录显示 |  |  | mycom[  bookname,comdate,compoint,content,type,]  com\_num | mycom中有书名，评论时间，评分，评论内容，图书类型；  评论记录数目 |
| 修改电话/邮箱 | newtele/newmail | 新电话/邮箱 | isSucceed | 0:修改成功 |
| 修改密码 | oldpass,newpass,  renewpass | 旧密码，新密码，重复新密码 | isSucceed  error | 0:修改成功  0:旧密码错误 |
| 续借图书 | bookid | 索引号 | isSucceed  error | 0:续借成功  0:过期1:有人预约 |
| 归还图书 | bookid | 索引号 | isSucceed | 1:还书成功  0:逾期未还，先交罚款 |
| 取消预约 | bookid | 索引号 | isSucceed | 1:取消成功 |
| 热门图书显示 |  |  | book\_data[name,point,frequency,  pictur,bookid] | book\_data中有书名，评分，借阅次数，封面，索引号 |
| 热门期刊显示 |  |  | journal\_data[  name,point,  picture,  bookid] | journal\_data中有期刊名，评分，封面，索引号 |
| 获取图书详情 | bookid | 索引号 | data[name,author,translator,picture,public,price,  bookID,  author\_country,  point,quantity,  intro,publish\_time,location,  location\_name,  book\_type,type];  result | data中包括书名，作者，译者，封面，出版社，价格，索引号，国家，评分，数量，简介，出版日期，馆藏位置，图书类型；  0:错误1:正常返回 |
| 图书详情中的相关评论 | bookid | 索引号 | comment[comment,time,point]  result\_num | comment中有评论内容，评论时间，评分；  评论数目 |
| 图书详情中借阅该书 | bookid | 索引号 | isSucceed  error | 是否成功  1:无信用2:罚金未交3:到达借阅上限4:已经借阅5:未登录6:书没有存货 |
| 图书详情中评论该书 | comment，book\_id，point，type | 评论内容，索引号，评分，图书类型 | isSucceed  error | 0:错误，>0成功  1:未登录 |
| 图书详情中归还该书 | bookid | 索引号 | isSucceed  error | 0:不成功1:更新借阅表成功2:更新借阅表+通知表成功  1:没有对应记录 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 功能 | 前端传入参数 | 参数描述 | 后端返回 | 参数描述 |
| 管理员登录 | man\_id,  password  code | 管理员id,  密码，  验证码 | 提示信息 | 登录成功  /管理员不存在  /密码错误  /验证码错误 |
| 查看图书 |  |  | list[book\_id,book\_name,  book\_type\_name,book\_intro,book\_price,book\_quantity,book\_author,book\_translator,  country\_name,publish\_name,book\_publish\_date,book\_cover,  location\_name,  book\_frequency,  book\_point | list中包括【  图书索引  图书名称  图书类型  图书介绍  价格  数量  作者  译者  国家  出版社  出版日期  封面  馆藏位置  借阅次数  评分】 |
| 添加图书 | book\_id,  book\_name,  book\_type\_id,  book\_intro,  book\_price,  book\_quantity,  book\_author,  book\_translator,  country\_id,  publish\_id,  book\_publish\_date,book\_cover,  location\_id | 图书索引  图书名称  图书类型id  图书介绍  价格  数量  作者  译者  国家id  出版社id  出版日期  封面  馆藏位置id | 提示信息 | 添加成功  /该图书已存在，添加失败  /图书索引不能为空  /书名不能为空  /价格不能为空  /数量不能为空  /简介不能为空  /简介内容不能少于10个字  /出版日期不能为空  /出版社编号不能为空  /国家编号不能为空 |
| 修改图书 | book\_id,  book\_name,  book\_type\_id,  book\_intro,  book\_price,  book\_quantity,  book\_author,  book\_translator,  country\_id,  publish\_id,  book\_publish\_date,book\_cover,  location\_id | 图书索引  post包括【书名称  图书类型id  图书介绍  价格  数量  作者  译者  国家id  出版社id  出版日期  封面  馆藏位置id】 | 提示信息 | 修改成功  /图书索引不能为空  /书名不能为空  /价格不能为空  /数量不能为空  /简介不能为空  /简介内容不能少于10个字  /出版日期不能为空  /出版社编号不能为空  /国家编号不能为空  /馆藏位置id不能为空 |
| 删除图书 | book\_id | 图书索引 | 提示信息 | 删除成功 |
| 批量删除图书 | ids[book\_id] | 包含book\_id的数组 | 提示信息 | 勾选框不能为空/  删除成功 |
| 搜索图书 | type  content | 搜索方式  内容 | list[book\_id,book\_name,  book\_type\_name,book\_intro,book\_price,book\_quantity,book\_author,book\_translator,  country\_name,publish\_name,book\_publish\_date,book\_cover,  location\_name,  book\_frequency,  book\_point | list中包括【  图书索引  图书名称  图书类型  图书介绍  价格  数量  作者  译者  国家  出版社  出版日期  封面  馆藏位置  借阅次数  评分】 |
| 查看期刊 |  |  | list[book\_id,  jou\_name,  book\_type\_name,  jou\_price,  jou\_quantity,  publish\_name,  book\_publish\_date,jou\_cover,location\_name,jou\_point,  other] | list中包括【  期刊索引  期刊名称  类型  价格  数量  出版社  出版日期  封面  馆藏位置  评分  附注】 |
| 添加期刊 | book\_id  jou\_name  book\_type\_id  jou\_price  jou\_quantity  publish\_id  book\_publish\_date  jou\_cover  location\_id  other | 期刊索引  期刊名称  图书类型id  价格  数量  出版社id  出版日期  封面  馆藏位置id  附注 | 提示信息 | 添加成功  /该期刊已存在，添加失败  /期刊索引不能为空  /期刊名不能为空  /价格不能为空  /数量不能为空  /出版日期不能为空  /出版社编号不能为空  /馆藏位置id不能为空 |
| 修改期刊 | book\_id  jou\_name  book\_type\_id  jou\_price  jou\_quantity  publish\_id  book\_publish\_date  jou\_cover  location\_id  other | 期刊索引  期刊名称  图书类型id  价格  数量  出版社id  出版日期  封面  馆藏位置id  附注 | 提示信息 | /修改成功  /期刊索引不能为空  /期刊名不能为空  /价格不能为空  /数量不能为空  /出版日期不能为空  /出版社编号不能为空  /馆藏位置id不能为空 |
| 搜索期刊 | type  content | 搜索方式  内容 | list[book\_id,  jou\_name,  book\_type\_name,  jou\_price,  jou\_quantity,  publish\_name,  book\_publish\_date,jou\_cover,location\_name,jou\_point,  other] | list中包括【  期刊索引  期刊名称  类型  价格  数量  出版社  出版日期  封面  馆藏位置  评分  附注】 |
| 删除期刊 | book\_id | 图书索引 | 提示信息 | 删除成功 |
| 批量删除期刊 | ids[] | 包含book\_id的数组 | 提示信息 | 勾选框不能为空/  删除成功 |
| 查看（更新）热门图书 |  |  | list[book\_id,book\_name,  book\_type\_name,book\_intro,book\_price,book\_quantity,book\_author,book\_translator,  country\_name,publish\_name,book\_publish\_date,book\_cover,  location\_name,  book\_frequency,  book\_point | list中包括【  图书索引  图书名称  图书类型  图书介绍  价格  数量  作者  译者  国家  出版社  出版日期  封面  馆藏位置  借阅次数  评分】 |
| 查看（更新）热门期刊 |  |  | list[book\_id,  jou\_name,  book\_type\_name,  jou\_price,  jou\_quantity,  publish\_name,  book\_publish\_date,jou\_cover,location\_name,jou\_point,  other] | list中包括【期刊索引  期刊名称  类型  价格  数量  出版社  出版日期  封面  馆藏位置  评分  附注】 |
| 查看借阅记录 |  |  | list[borrowed\_id,  reader\_id,  book\_id,  borrowed\_sdate,  borrowed\_rdate,  is\_overed] | list中包括【  借阅流水编号  借阅读者id  图书id  借阅起始日期  借阅归还日期  书籍借阅是否到期】 |
| 筛选逾期记录 |  |  | list[borrowed\_id,  reader\_id,  book\_id,  borrowed\_sdate,  borrowed\_rdate,  is\_overed] | list中包括【  借阅流水编号  借阅读者id  图书id  借阅起始日期  借阅归还日期  书籍借阅是否到期】 |
| 设置罚款 | borrowed\_id | 借阅流水编号 | 提示信息 | 设置成功 |
| 重置罚款 | borrowed\_id | 借阅流水编号 | 提示信息 | 回退成功 |
| 删除借阅记录 | borrowed\_id | 借阅流水编号 | 提示信息 | 删除成功 |
| 批量删除借阅记录 | ids[borrowed\_id] | 包含借阅流水编号的数组 | 提示信息 | 勾选框不能为空/  删除成功 |
| 查看预约记录 |  |  | list[app\_id,book\_id,reader\_id,app\_date] | list包括【预约流水编号，图书id，读者id,预约日期】 |
| 删除预约记录 | app\_id | 预约流水编号 | 提示信息 | 删除成功 |
| 批量删除预约记录 | ids[app\_id] | 包含预约流水编号的数组 | 提示信息 | 勾选框不能为空/  删除成功 |
| 查看读者 |  |  | list[reader\_id,reader\_name,read\_type\_id,reader\_tel,reader\_mail,reader\_credit,reader\_fine] | list包括【读者id,读者姓名，读者类型id,读者电话，读者邮箱，读者信用值，读者罚款】 |
| 修改读者罚款信息 | reader\_id  reader\_fine | 读者id  读者罚款 | 提示信息 | 修改成功 |
| 搜索读者 | type  content | 搜索方式  内容 | list[reader\_id,reader\_name,read\_type\_id,reader\_tel,reader\_mail,reader\_credit,reader\_fine] | list包括【读者id,读者姓名，读者类型id,读者电话，读者邮箱，读者信用值，读者罚款】 |
| 删除读者 | reader\_id | 读者id | 提示信息 | 删除成功 |
| 批量删除读者 | ids[reader\_id] | 包含读者id的数组 | 提示信息 | 勾选框不能为空/  删除成功 |
| 查看读者注册信息 |  |  | list[  apply\_id,reader\_id,read\_type,read\_name,apply\_date,college\_id,reader\_tel,reader\_mail,apply\_result] | list包括【申请流水编号，读者id，读者类型id，读者姓名，申请日期，学院id，电话，邮箱，申请结果】 |
| 审核读者注册信息 | apply\_id,reader\_id,read\_type,read\_name,apply\_date,college\_id,reader\_tel,reader\_mail | 申请流水编号，读者id，读者类型id，读者姓名，申请日期，学院id，电话，邮箱 | 提示信息 | 审核通过  /审核不通过 |
| 删除申请记录 | apply\_id | 申请流水编号 | 提示信息 | 删除成功 |
| 批量删除申请记录 | ids[apply\_id] | 包含申请流水编号的数组 | 提示信息 | 勾选框不能为空  /删除成功 |
| 查看预约通知表 |  |  | list[inform\_id,book\_id,book\_name,reader\_ireader\_name,reader\_mail] | list包括【通知流水编号，图书id，图书名称，读者id，读者姓名，读者邮箱】 |
| 发送通知 | inform\_id  reader\_id  reader\_mail | 通知流水编号，读者id，读者邮箱 | 提示信息 | 发送成功 |
| 查看书籍评论 |  |  | list[  com\_id,book\_id,reader\_id,com\_content,com\_point,com\_date] | list包括【评论流水编号，图书id，读者id，评论内容，评分，评论日期】 |
| 删除书籍评论 | com\_id | 评论流水编号 | 提示信息 | 删除成功 |
| 批量删除书籍评论 | ids[com\_id] | 包含评论流水编号的数组 | 提示信息 | 勾选框不能为空/删除成功 |
| 查看期刊评论 |  |  | list[  com\_id,book\_id,reader\_id,com\_content,com\_point,com\_date] | list包括【评论流水编号，图书id，读者id，评论内容，评分，评论日期】 |
| 删除期刊评论 | com\_id | 评论流水编号 | 提示信息 | 删除成功 |
| 批量删除期刊评论 | ids[com\_id] | 包含评论流水编号的数组 | 提示信息 | 勾选框不能为空/  删除成功 |
| 查看反馈信息 |  |  | list[fdb\_id,reader\_id,fdb\_content,fdb\_date] | list包括【反馈流水编号，读者id,反馈信息，反馈日期】 |
| 删除反馈信息 | fdb\_id | 反馈流水编号 | 提示信息 | 删除成功 |
| 批量删除反馈信息 | ids[fdb\_id] | 包含反馈流水编号的数组 | 提示信息 | 勾选框不能为空/删除成功 |
| 查看个人信息 |  |  | list[man\_id,man\_name,man\_mail，password,,man\_man\_rank] | list包括【管理员id,姓名，邮箱，密码，等级】 |
| 修改个人信息 | man\_id,man\_name,man\_mail，password,  man\_repassord | 管理员id,姓名，邮箱，密重复密码 | 提示信息 | 修改成功/管理员名不能为空/密码不能为/密码长度不少于6位/两次密码不一致/邮箱不能为空 |
| 查看所有管理员 |  |  | list[man\_id,man\_name,man\_password,man\_mail,man\_rank] | list包括【管理员id,管理员密码，管理员邮箱，管理员等级】 |
| 添加管理员 | man\_id  man\_name  man\_password  man\_repassword  man\_rank | 管理员id  姓名  密码  重复密码  管理员等级 | 提示信息 | 添加成功/该管理员已存在/管理员id不能为空  用户名不能为空  密码不能为空/  密码长度不少于6位/两次密码不一致/邮箱不能为空/等级不能为空 |
| 修改管理员 | man\_id  man\_name  man\_password  man\_repassword  man\_rank | 管理员id  姓名  密码  重复密码  管理员等级 | 提示信息 | 修改成功/  用户名不能为空  密码不能为空/  密码长度不少于6位/两次密码不一致/邮箱不能为空/等级不能为空 |
| 删除管理员 | man\_id | 管理员id | 提示信息 | 删除成功 |
| 批量删除管理员 | ids[man\_id] | 包含管理员id的数组 | 提示信息 | 勾选框不能为空/  删除成功 |
| 搜索管理员 | type  content | 搜索方式  内容 | list[man\_id,man\_name,man\_password,man\_mail,man\_rank] | list包括【管理员id,管理员密码，管理员邮箱，管理员等级】 |

6.5关键功能的算法设计

6.5.1 MVC

我们采用了MVC软件构建模式， MVC全名是Model View Controller，是模型(model)－视图(view)－控制器(controller)的缩写。 其中： Model（模型）是应用程序中用于处理应用程序数据逻辑的部分。通常模型对象负责在数据库中存取数据。 View（视图）是应用程序中处理数据显示的部分。通常视图是依据模型数据创建的。 Controller（控制器）是应用程序中处理用户交互的部分。可以接受访问并对访问做出应答。通常控制器负责从视图读取数据，控制用户输入，并向模型发送数据。

MVC的主要优点有 ：

1. MVC分层有助于管理复杂的应用程序。这是因为可以在一个时间内专门关注一个方面。例如，可以在不依赖业务逻辑的情况下专注于视图设计。同时也让应用程序的测试更加容易。
2. MVC分层同时也简化了分组开发，不同的开发人员可同时开发视图、控制器逻辑和业务逻辑。

6.5.2 ThinkPHP框架

ThinkPHP是快速兼容简单的php开发框架，使用面向对象的开发结构和MVC模式，融合了Struts的思想和标签库、ROR的ORM映射和ActiveRecord模式。ThinkPHP能解决应用开发中的大多数需要，包含了底层架构、兼容处理、基类库、数据库访问层、模版引擎、缓存机制、插件机制、角色认证、表单处理等常用组件。 在使用ThinkPHP框架时，我们能将更多的精力放在图书馆系统的逻辑设计、功能实现上，因为繁琐的基本功能或需要多次使用的程序段都能直接从组件中调用。在图书管理系统中，图书管理员的身份相对于用户和图书记录来说是长期身份，在保存到数据库后一段时间内不会变动。 ThinkPHP中的静态模型功能能在访问数据库后生成缓存，之后再次查看管理员记录表时就不需要再次链接数据库，而直接从缓存文件中查看，能节省时间提高效率。 另外，它还提供了灵活和方便的数据操作方法，实现了对数据库的创建、读取、更新和删除，方便在数据库构建过程中使用。

6.5.3 RBAC

\*RBAC是基于角色的访问控制。在RBAC中，权限与角色相关联，用户通过成为适当角色成员来获得这些角色的权限，能极大地简化权限的管理。在一个组织中，角色为了完成某种工作而创造，用户则依据他的责任和资格被委派相应的角色，并可以很容易的在角色之间转换。同时，角色可以因实际需要被赋予新的权限，也能随时收回被赋予的权限。

\*在我们的图书馆系统中，使用者被分为了四个主要权限：超级管理员、不同级别的图书管理员、学生和老师。他们分别使用不同的权限。 如教师和学生都是直接用户，他们的权限包括浏览、借阅、预定和续借等； 图书管理员还需要对图书进行登记，在归还时改变图书的状态，因而有操作图书表的权限；超级管理员对整个数据库进行维护，其权限范围更广。 通过对不同的角色赋权，再将角色赋予用户，能极大地减少管理员工作，提高效率。

6.5.4 ajax保证完整性

\*在设计注册、借还等页面时，我们使用了异步javascript和XML技术，保证局部刷新时不影响整体页面，也不需要用户来回登录注册页面。

\*Ajax是用于创建快速动态网页的技术，通过在后台和服务器进行少量数据交换，Ajax可以使网页实现异步更新。这意味着不需要重新载入整个网页来显示新内容，而只需要进行局部的刷新来获得变更内容。 在我们的系统中，用户注册时会被要求输入用户名，因为实际情况中可能使用的用户是非常多的，难免造成用户名一致的问题。当用户名与之前注册的用户一致时，会直接显示错误提示，要求用户更换用户名，而不需要在注册提交时再告知重复，让用户重新填写所有内容。 另外，在借书时系统会要求用户登录自己的帐号，此时如果用户未注册，却输入了已经存在的用户名，也会提示用户名被注册。 这种设计主要目的在于为用户节省时间，一方面保证了系统中的注册用户是唯一的，实现了系统完整性的要求；另一方面方便用户注册，不需要一次次输入全部注册信息。

七、课程总结和体会

数据库建表时经验避坑：

1. 引擎设置为InnoDB。
   1. MyISAM类型不支持事务处理等高级处理，而InnoDB类型支持。MyISAM类型的表强调的是性能，其执行速度比InnoDB类型更快，但是不提供事务支持，而InnoDB提供事务支持以及外部键等高级数据库功能。
   2. 而我使用的Mysql默认引擎是MyISAM，而MyISAM不支持事务处理，所以我需要更换他的引擎。
2. 设置外键的名字、类型、是否能为空等都要一致
3. Mysql命名、设计及使用规范：
   1. 主键一般设置为表名+id，并且设定为自增
   2. 所有字段都设置为not\_null,同时提供默认值，比如：0
   3. 表命名中，有info或者record后缀的全部删掉
   4. 库、表、字段全部采用小写，不要使用驼峰式命名
   5. 布尔意义的字段以“is\_”作为前缀，后接动词过去分词。
4. 如果随机地创建表而不是一定要所有父表早于子表建立，需要设置需求：set foreign\_key\_checks=0;
5. 对数据库的数据和结构进行备份，防止意外发生
6. 