**暨南大学本科实验报告专用纸**

课程名称 数据库系统原理实验 成绩评定

实验项目名称 数据库的设计与应用开发 指导教师 朱蔚恒

实验项目编号 7、8 实验项目类型 综合 实验地点 N116

学生姓名 陈宇 学号 2020101642

学院 信息科学技术学院 系 计算机科学系 专业 软件工程

实验时间 2022 年 12 月 4 日 上 午～ 12 月4 日 上 午

# 实验目的

掌握基于JDBC驱动的数据库应用开发方法；

掌握综合运用数据库原理、方法和技术进行数据库应用系统分析、设计和开发的能力。

# 实验内容

基于JDBC驱动的数据库连接方法，实现数据库数据应用等开发常见功能；

设计一个图书馆管理系统。

# 实验过程

设计一个图书管理系统：

1. 需求分析

①设计目的：

基于图书馆的基本业务提供一个服务平台，按服务对象划分可分为管理员和用户两个角色。旨在减轻管理压力并为用户提供更便捷服务。

②功能需求：

管理员能实现对书籍和用户的统一管理；用户能通过系统更快更方便地定位或借阅图书。

管理员功能

人员信息：对系统内人员信息（系统内编号、学号、姓名、联系方式以及是否为管理员）的增删改查；

书籍管理：对馆内图书按书名、作者以及分类进行查询，可维护书籍当前信息（如封面图、简介、位置、数量等）；

事务操作：

对图书借还操作的实现，查询管理借还书用户的记录，包括书名、姓名、日期、时长等。

用户功能

个人中心：

个人足迹的记录（借阅图书和发表过的书评）

图书馆浏览

可按分类、书籍名、作者名寻找书籍（支持模糊查询）

得到书籍位置、馆藏等基本信息后可去阅读或借阅图书。

书评社区

设计开放的评价平台，可根据其他用户的打分和评论了解书籍。

1. 概念结构的设计：

识别出管理员admin、图书book、用户user、书籍类别category、书评信息comment等5个实体，每个实体的属性和码如下：

管理员：编号，学号，姓名，昵称，密码，联系方式，头像，个签

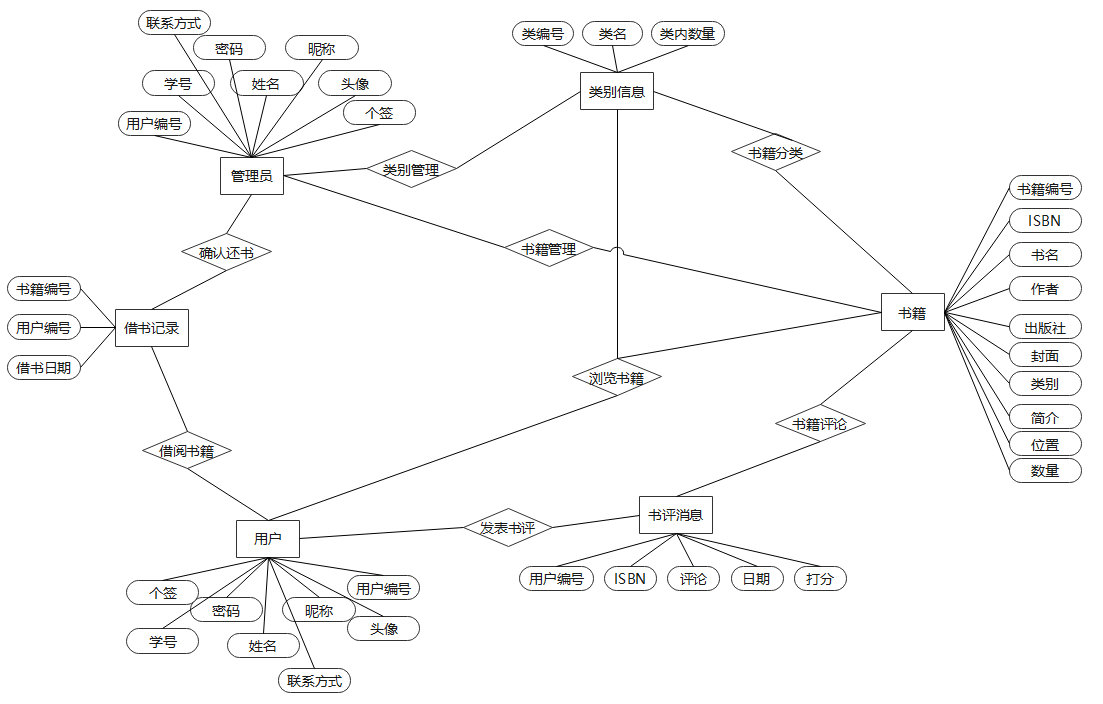
用户：编号，学号，姓名，昵称，密码，联系方式，头像，个签

图书：书名，作者，ISBN码，出版日期，出版社，简介，类别，位置

书籍类别：类编号，类名，类内数量

书评信息：用户编号，ISBN，内容，日期，评分

用E-R图描述概念结构：



1. 逻辑结构的设计：

将上述概念结构转化为数据模型

user关系（包括了user实体和admin实体）：

uid：用户编号

username：用户名（学号）

password：密码

name：姓名

nick：昵称

img：头像

motto：个性签名

phone：联系方式

isAdmin：管理员选项（1是0否）

book关系：

bid：书籍编号

ISBN：国际通用标准书号

title：书名

author：作者

publishDate：出版社

cover：封面

abs：简介

cid：类别序号

location：位置信息

nums：藏书数量

借阅关系borrow：

bid：书籍编号

uid：用户编号

borrowDate：借书日期

书籍类别category：

cid：类别编号

cname：类别名

icon：同类书籍数量

书评信息comment：

uid：用户编号

ISBN：国际通用标准书号

review：评论内容

commdate：发表日期

level：打分

1. 物理结构的设计：

数据库物理结构首先根据逻辑结构生成，然后根据应用需求设计数据库的索引结构、存储结构

数据库模式SQL语句的生成（采用mysql）

相应的建表语句：

建user表：

create table user (

uid int primary key auto\_increment ,

username varchar(10) not null unique,

password varchar(32) not null default '123456',

name varchar(10) not null,

nick varchar(10) not null,

img varchar(255) default 'http://inews.gtimg.com/newsapp\_match/0/15077040964/0',

motto varchar(10) default '我爱读书',

phone varchar(11) not null unique,

isAdmin tinyint default false

)ENGINE=InnoDB DEFAULT character set = gbk;

建book表：

create table book (

bid int primary key AUTO\_INCREMENT,

title varchar(20) not null,

author varchar(20) not null,

publishDate date not null,

publisher varchar(20) not null,

cover varchar(255) default 'https://shadow.elemecdn.com/app/element/hamburger.9cf7b091-55e9-11e9-a976-7f4d0b07eef6.png',

abs varchar(255),

cid int references category(cid),

location varchar(20) not null ,

nums int default 0

)ENGINE=InnoDB DEFAULT character set = gbk;

建borrow表：

create table borrow (

bid int references book(bid),

uid varchar(10) references user(uid),

borrowDate date not null,

primary key (bid, uid)

)ENGINE=InnoDB DEFAULT character set = gbk;

建category表：

create table category (

cid int primary key AUTO\_INCREMENT,

cname varchar(10) not null

)ENGINE=InnoDB DEFAULT character set = gbk;

建comment表：

create table comment(

uid int references user.uid,

ISBN int references book.ISBN,

Review varchar(300),

commdate date not null,

level float(2,1)

) ENGINE=InnoDB DEFAULT character set = gbk;

相关视图的建立：

create view borrow\_info as select borrowDate , user.uid as uid, book.bid as bid, name, username, title, utc\_date()-borrowDate as days\_used from user, borrow, book where borrow.uid = user.uid && borrow.bid = book.bid;

create view user\_info as select user.\*, count(borrow.uid) as borrownums

from user left join borrow on borrow.uid = user.uid

group by user.uid;

create view category\_info as  
select c.cid,c.cname,*count*(b.bid) as nums  
from category c left join book b on c.cid=b.cid  
group by c.cid;

create view comment\_info as  
select nick,img,title,review,commdate,level  
from user,book,comment  
where comment.uid=user.uid && comment.ISBN=book.ISBN;

create view user\_comment\_info as  
select nick,title,review,username,commdate  
from user,comment,book  
where comment.uid=user.uid && comment.ISBN=book.ISBN;

系统的建立：

该图书管理系统的实现由前端、后端和数据库组成，前端为网页，后端为前端、数据库提供接口并且处理一些业务逻辑，数据库主要勇于存放数据。本系统使用的框架和数据库所使用到的语言如下说明。

**前端框架（AJAX + Vue）**

AJAX(Asynchronous JavaScript And XML)，用于与服务器进行数据交换。

Vue是一套用于构建用户界面的渐进式框架。通过简单的API可实现响应式的数据绑定和可组合的视图组件。可以简化javaScript书写。

**后端框架（Springboot + Mybatis）**

Spring框架是一个轻量级控制反转和面向切面的容器框架。使用Spring框架时只需要专注于编写运用的业务逻辑从而大大提高了开发效率

Mybatis是一个基于java的持久层框架，支持普通 SQL查询，存储过程和高级映射的优秀持久层框架。MyBatis 消除了几乎所有的JDBC代码和参数的手工设置以及结果集的检索。MyBatis使用简单的 XML或注解用于配置和原始映射，将接口和Java对象映射成数据库中的记录。

**数据库的相关操作（MySQL）**

建表：

建user表：

create table user (

uid int primary key auto\_increment ,

username varchar(10) not null unique,

password varchar(32) not null default '123456',

name varchar(10) not null,

nick varchar(10) not null,

img varchar(255) default 'http://inews.gtimg.com/newsapp\_match/0/15077040964/0',

motto varchar(10) default '我爱读书',

phone varchar(11) not null unique,

isAdmin tinyint default false

)ENGINE=InnoDB DEFAULT character set = gbk;

建book表：

create table book (

bid int primary key AUTO\_INCREMENT,

title varchar(20) not null,

author varchar(20) not null,

publishDate date not null,

publisher varchar(20) not null,

cover varchar(255) default 'https://shadow.elemecdn.com/app/element/hamburger.9cf7b091-55e9-11e9-a976-7f4d0b07eef6.png',

abs varchar(255),

cid int references category(cid),

location varchar(20) not null ,

nums int default 0

)ENGINE=InnoDB DEFAULT character set = gbk;

建borrow表：

create table borrow (

bid int references book(bid),

uid varchar(10) references user(uid),

borrowDate date not null,

primary key (bid, uid)

)ENGINE=InnoDB DEFAULT character set = gbk;

建category表：

create table category (

cid int primary key AUTO\_INCREMENT,

cname varchar(10) not null

)ENGINE=InnoDB DEFAULT character set = gbk;

建comment表：

create table comment(

uid int references user.uid,

ISBN int references book.ISBN,

Review varchar(300),

commdate date not null,

level float(2,1)

) ENGINE=InnoDB DEFAULT character set = gbk;

建视图：

create view borrow\_info as select borrowDate , user.uid as uid, book.bid as bid, name, username, title, utc\_date()-borrowDate as days\_used from user, borrow, book where borrow.uid = user.uid && borrow.bid = book.bid;

create view user\_info as select user.\*, count(borrow.uid) as borrownums

from user left join borrow on borrow.uid = user.uid

group by user.uid;

create view category\_info as  
select c.cid,c.cname,*count*(b.bid) as nums  
from category c left join book b on c.cid=b.cid  
group by c.cid;

create view comment\_info as  
select nick,img,title,review,commdate,level  
from user,book,comment  
where comment.uid=user.uid && comment.ISBN=book.ISBN;

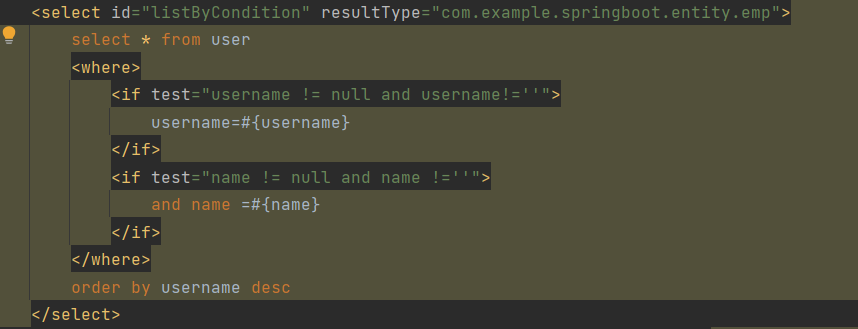
create view user\_comment\_info as  
select nick,title,review,username,commdate  
from user,comment,book  
where comment.uid=user.uid && comment.ISBN=book.ISBN;

“增删改查”功能的实现：（在后端中结合MyBatis实现）

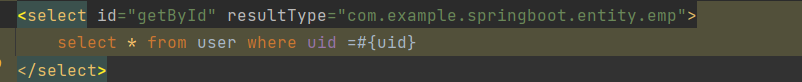
user表操作：

查找用户：

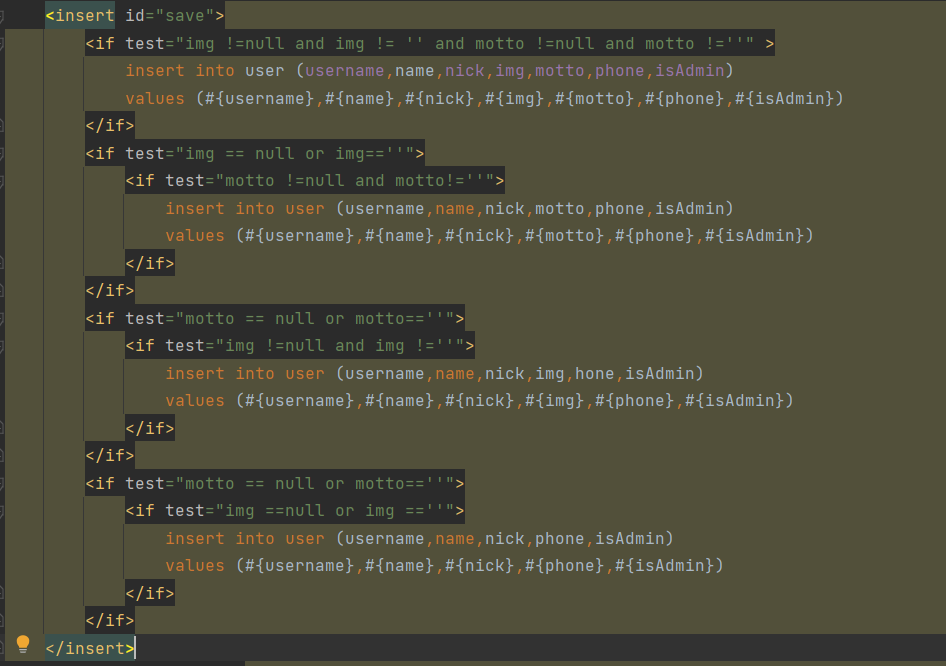
分页查找用户：



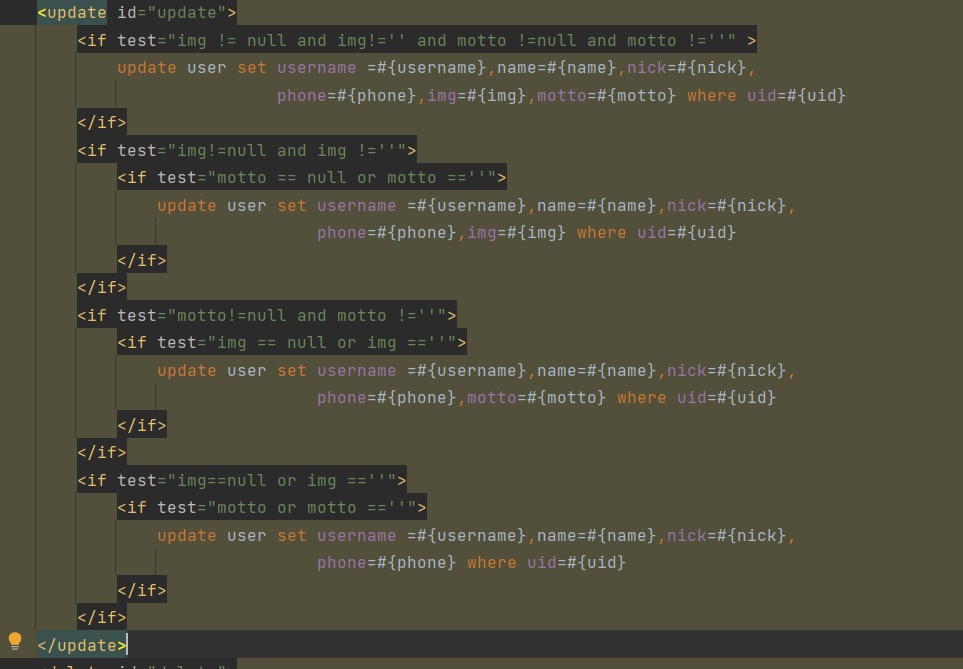
通过uid查找用户：



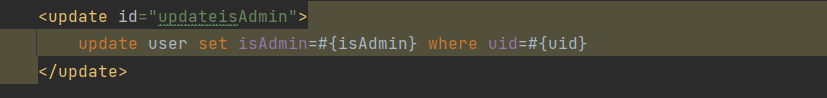
增加用户：



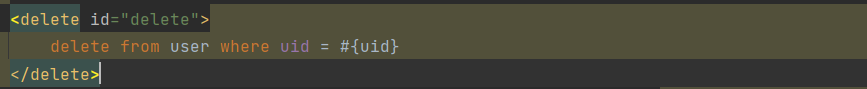
修改用户：



更改用户权限：



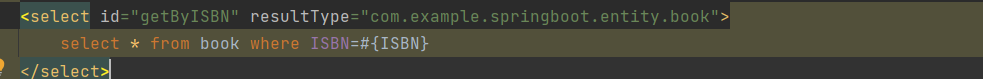
删除用户：

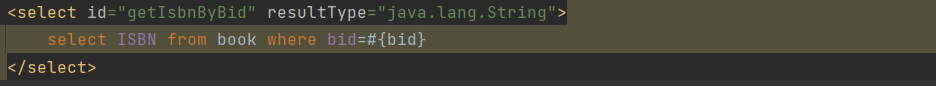


book表：

查找图书：

通过ISBN查询图书：

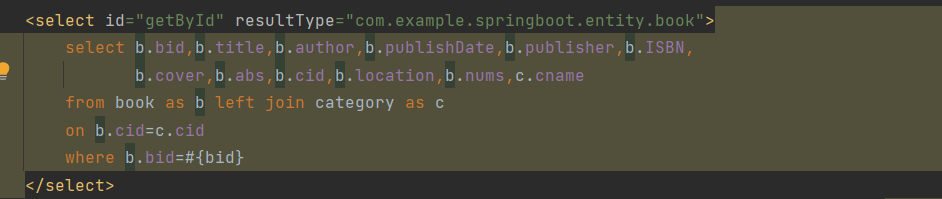


通过bid查找图书的ISBN：  


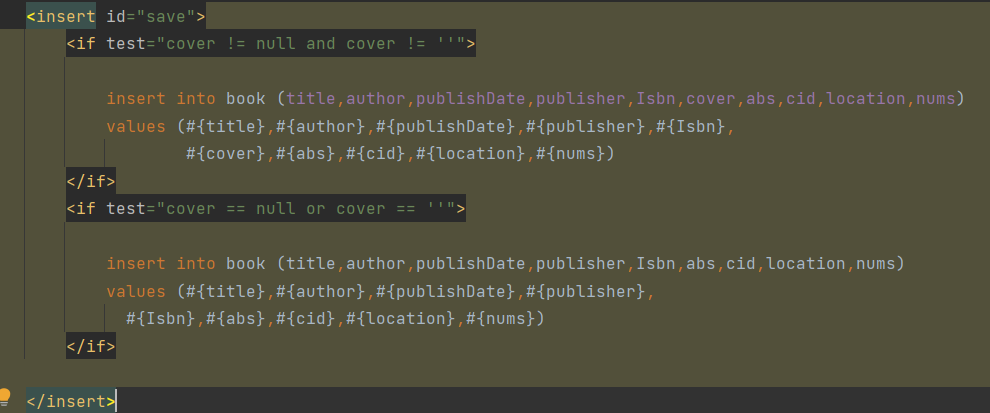
分页查询：



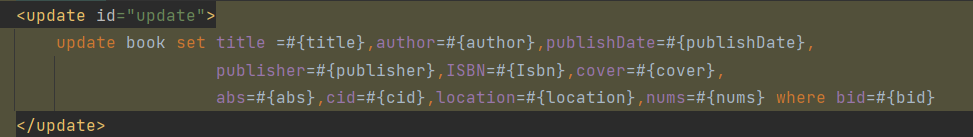
通过bid查找图书：



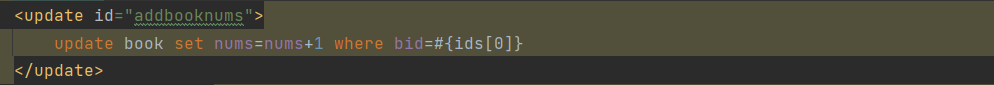
插入新书：



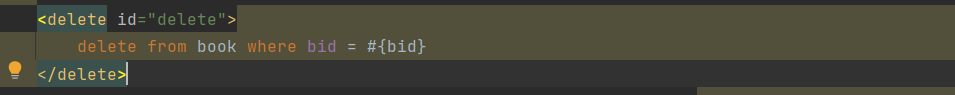
修改图书信息：



增加图书数量：

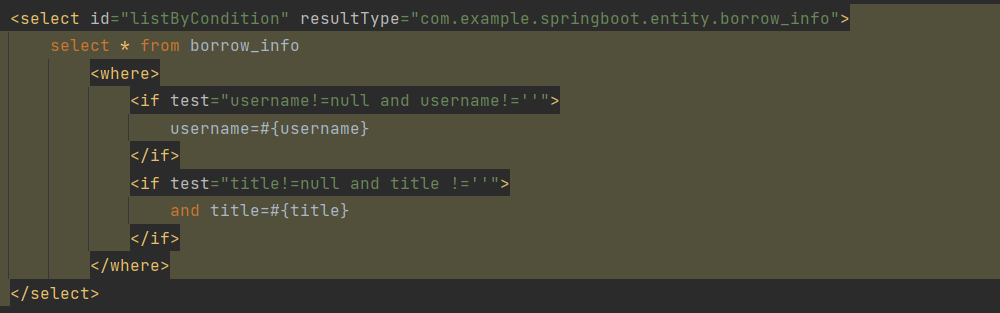


删除图书：

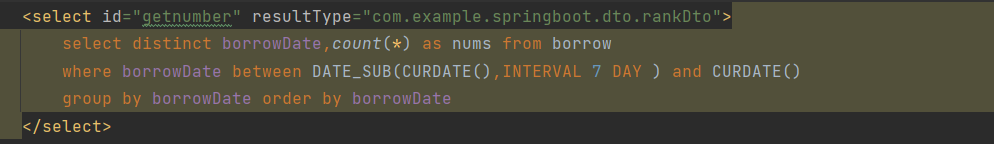


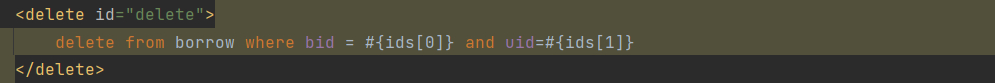
borrow表：

分页查询：



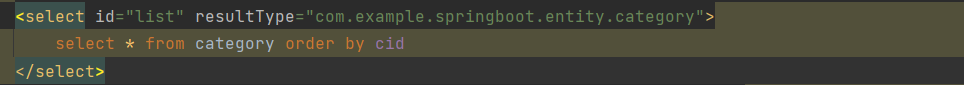
获取每一天的借阅数量：



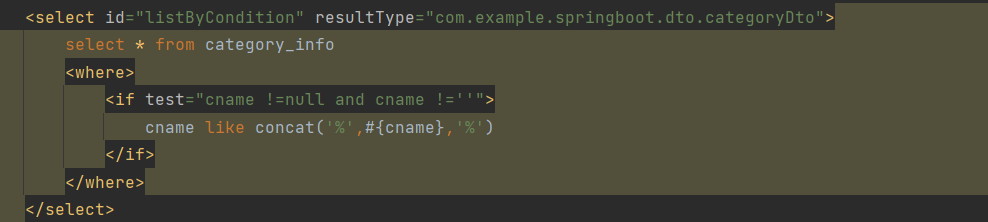
删除借阅记录：  


category表：

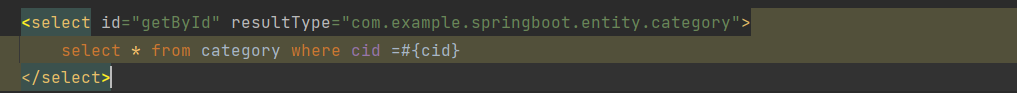
查询所有的类别：



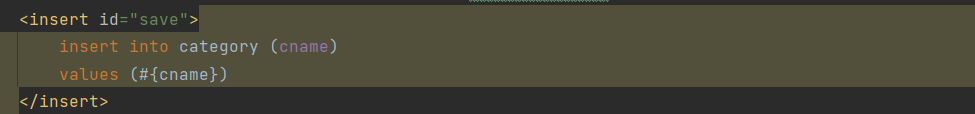
分页查询：

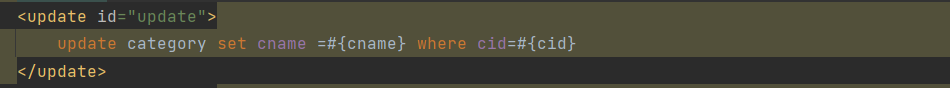


通过cid查看图书类别：



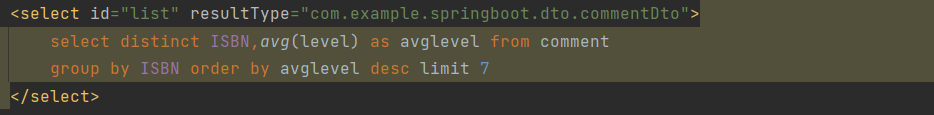
增加类别：

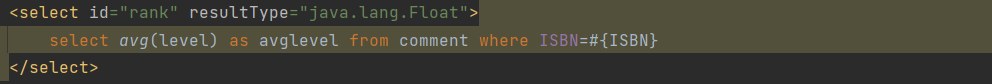


修改类别：  


comment表：

查找：



通过ISBN查找：  


四、系统展示：

先展示管理员功能：  
对用户的管理：

分页显示所有的用户：



查找功能：



支持按照学号/姓名进行查找：

按照学号查找：



按照姓名查找：



按照学号+姓名查找：



查找用户不存在的情况：



修改用户信息：

修改“木木”的信息：

修改前：



修改窗口：



修改姓名为“森”：



修改用户权限：

将上面用户的权限修改为管理员：

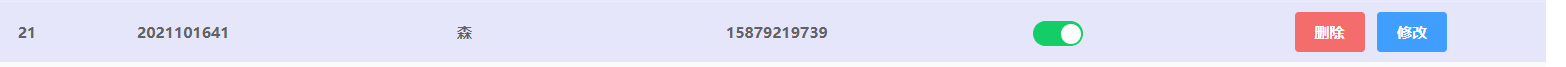
修改前：



点击修改，弹出提示框：

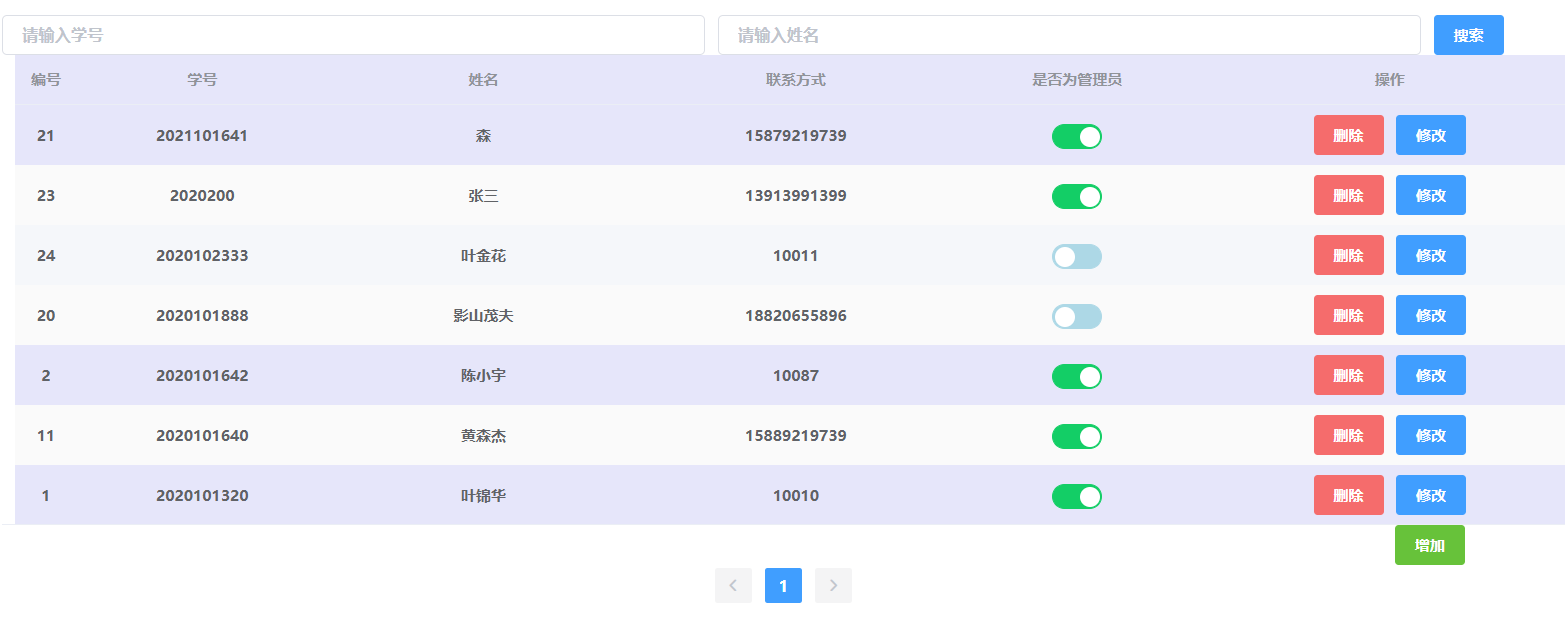


确定修改，修改后：



增加用户：

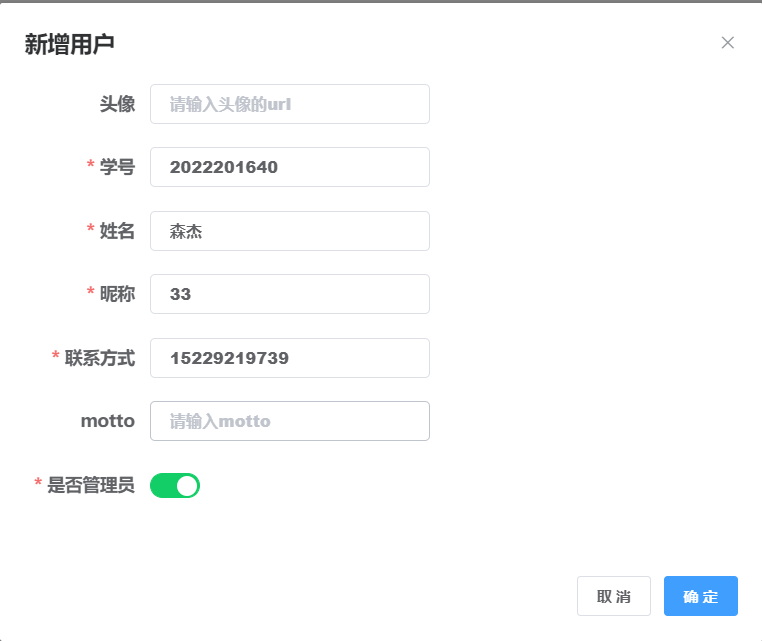
增加前：



增加页面：



填写相关信息：



增加后：



删除用户：

删除刚才增加的用户：

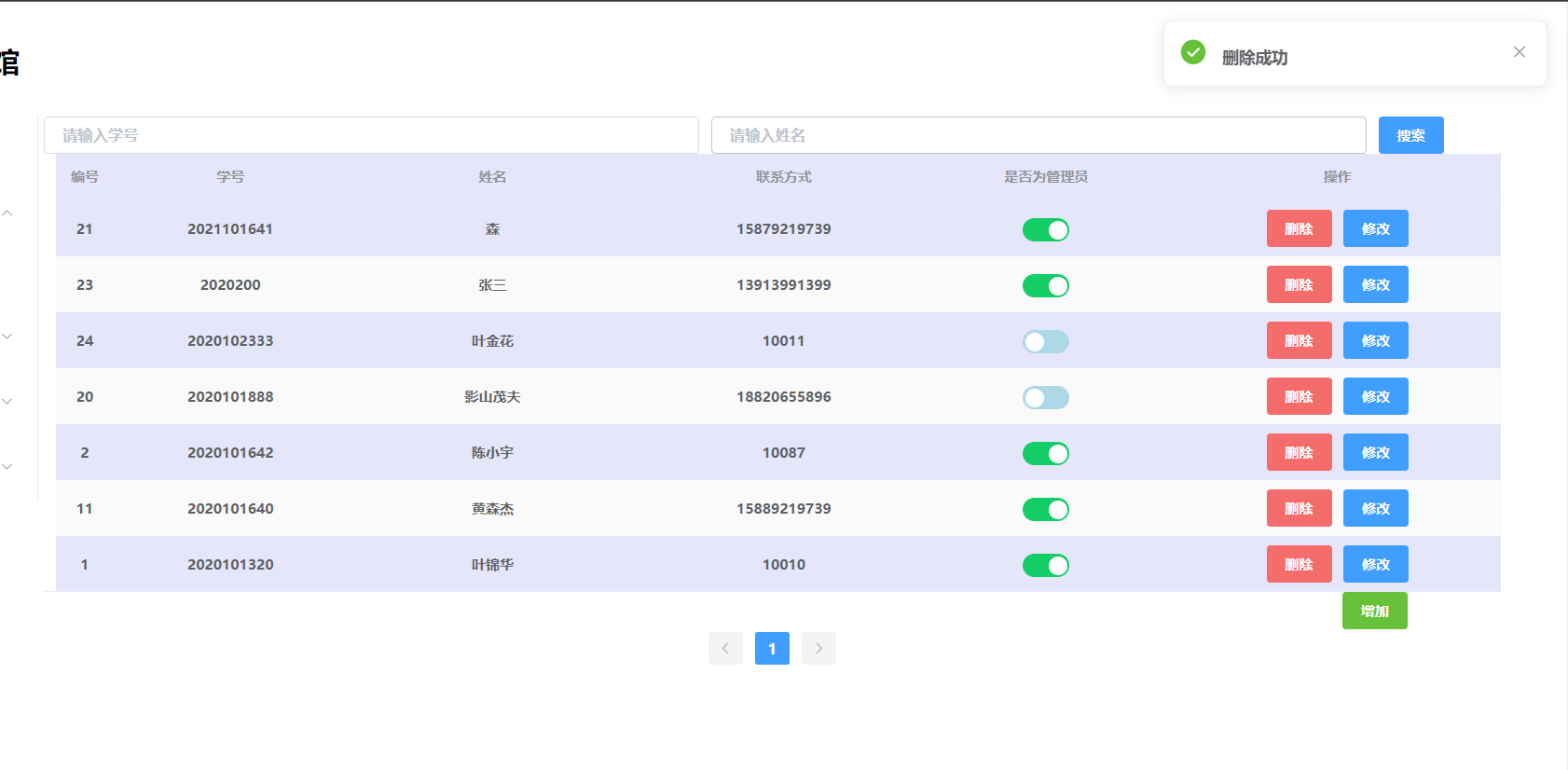
删除前：



弹出提示：



确定删除，删除后：

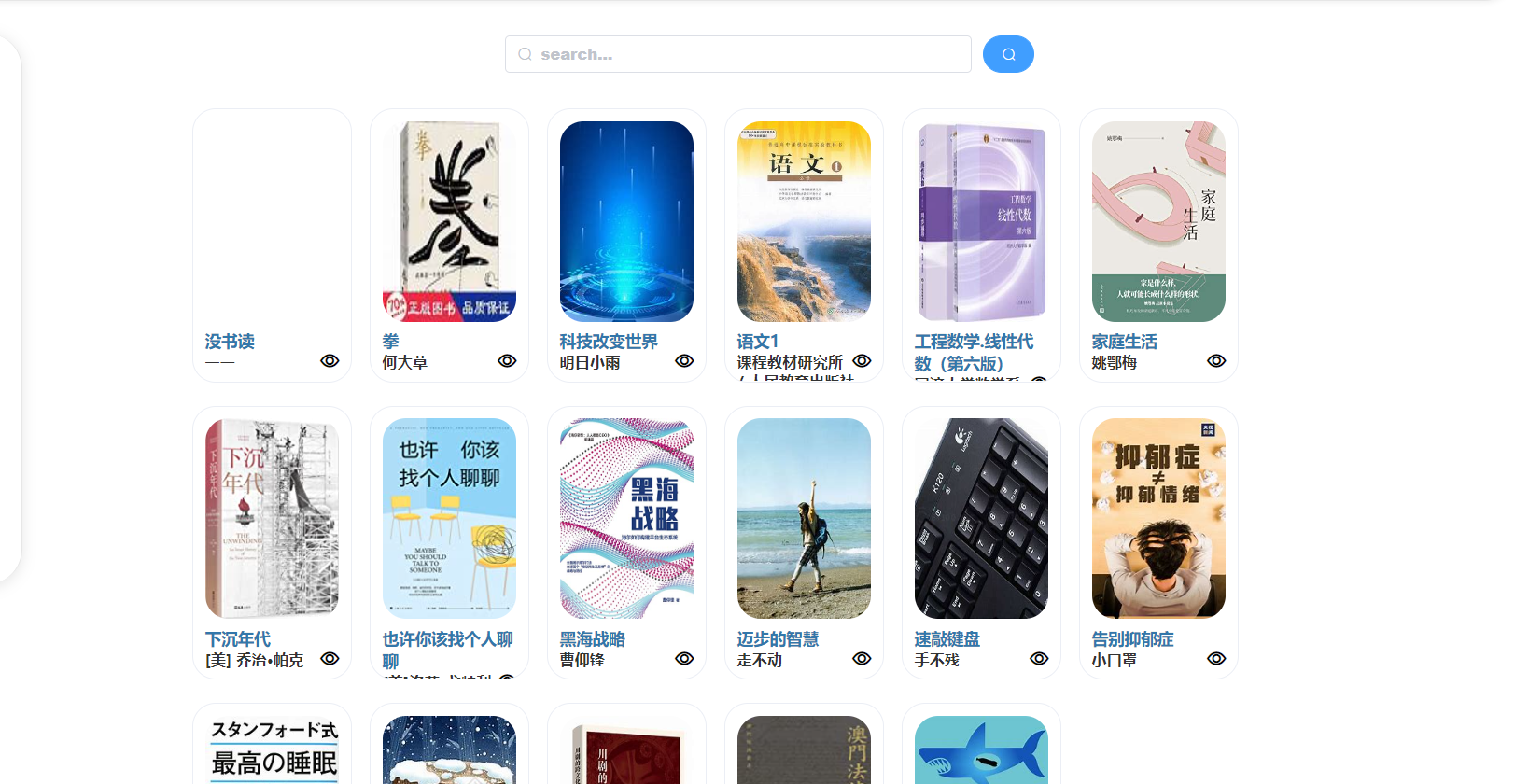


发现没有前面添加的用户，证明删除成功

管理员其它功能与上述功能类似，这里不做赘述。

下面展示用户界面的功能：

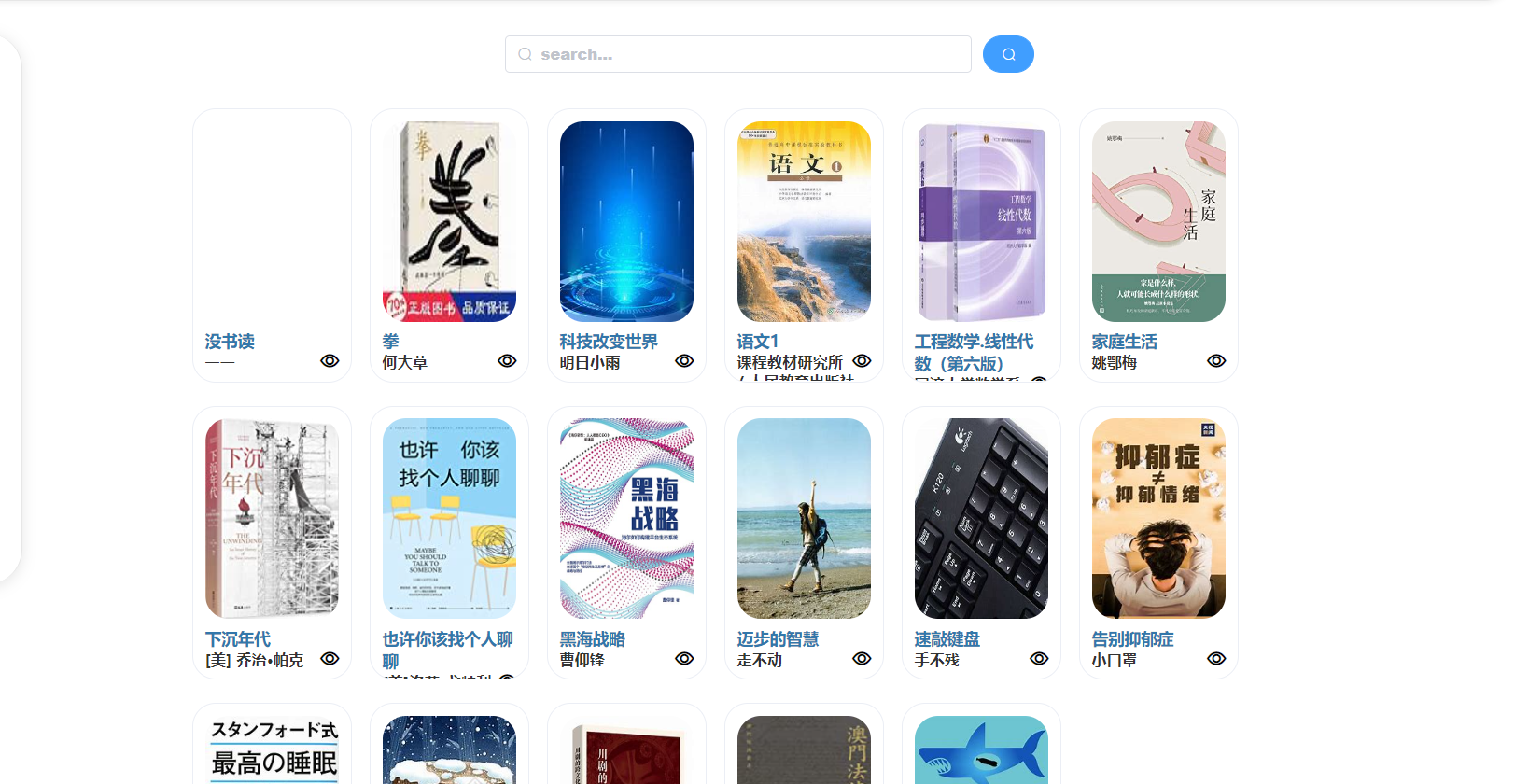
图书的展示：



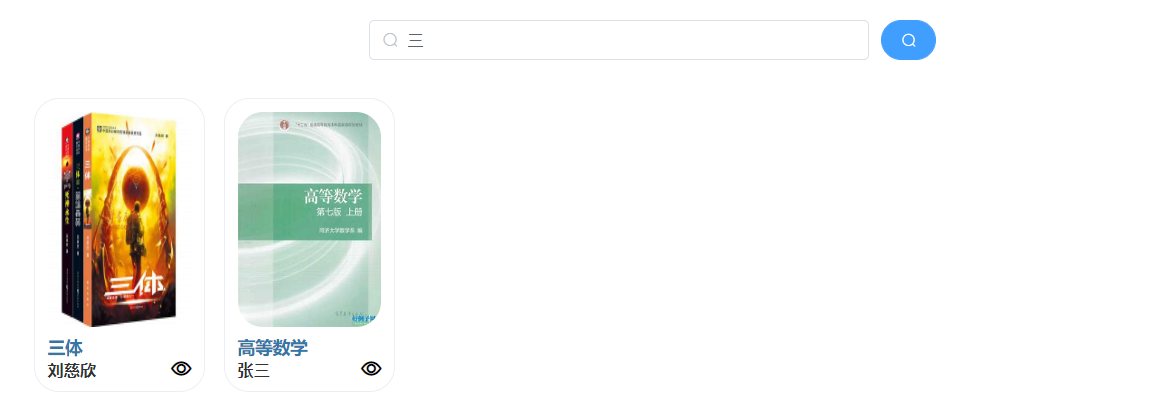
查找功能：

该功能支持模糊查询；

会同时查找作者或者书名；



输入“三”：



书的信息的展示：

将鼠标放在书的封面上会显示书的基本信息：



借书功能：

借阅一本《拳》：

借书前：

用户没有借阅记录；



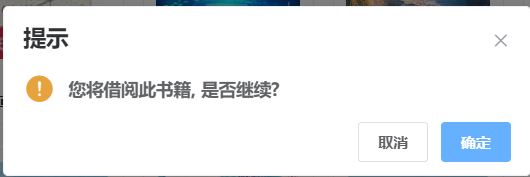
关注拳这本书的数量



借阅图书：



弹出窗口：



确定后：



查看借阅记录：



多了刚才借阅的图书，借阅成功；

查看《拳》的数量：



数量减少1，正确

下面演示借阅数量为0的书：  


借阅的时候会显示借阅失败：



分类功能：  
所有类别的展示:



查看特定类别的书：

