# Database System Lab 5

1. **实验目的**

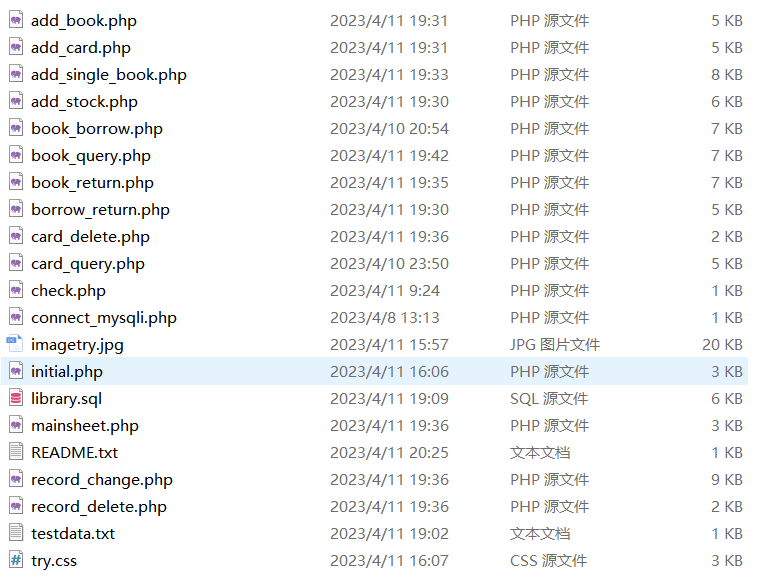
（1）掌握数据库应用开发程序设计方法。

（2）设计并实现一个精简的图书管理程序，要求具有图书入库、查询、借书、还书、借书证管理等基础功能。

1. **实验内容**

1.系统整体架构

本次“图书管理系统”的设计，我主要采用了 php + html 语言的网页实现方式。该程序主要分为六大独立功能模块：登录界面、管理员主页、图书管理、借书证管理、借还管理与切换管理账户；且在图书管理、借书证管理、借还管理模块下，分别设计了图书查询、图书修改、更改库存、单本入库、批量入库，借书证查询、借书证添加，借还信息查询、新的借阅、归还图书等子模块，在代码实现的基础之上，本图书管理系统也运用了一些简单的css美化设计，使交互界面更为美观舒适。

以下是本图书管理系统整体架构图，各文件对映具体模块实现见README.txt ：

2.应用技术

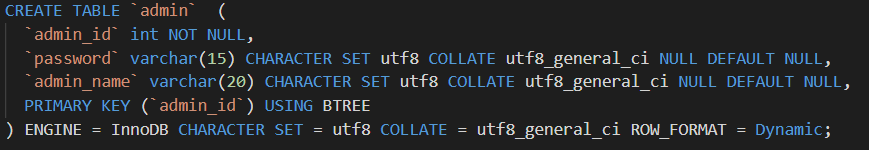
本次“图书管理系统”应用了 HTML + CSS + PHP + APache + MySQL等技术。

其中 html 主要用于网页的编写， css 则在 html 编写的基础上进行网页的布局与美化；在 Apache 服务器建立的基础上，依赖于 php 语言，链接数据库与网页，将需求转换为 mysql 语句，从而协调数据库，呈现不同的网页内容。

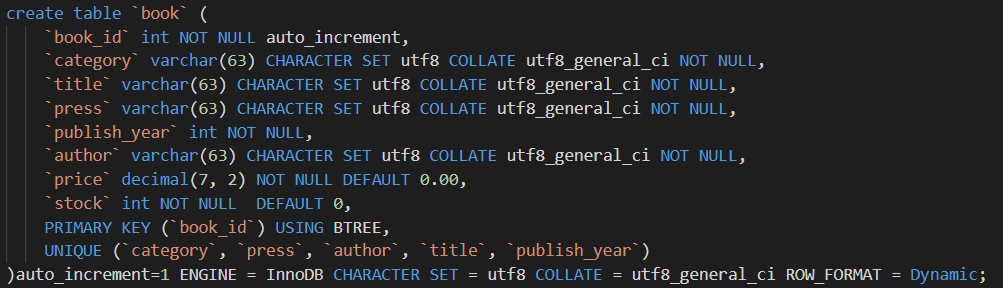
3.数据库表结构

本“图书管理系统”中，主要建立了四张数据表：admin、book、card与borrow.

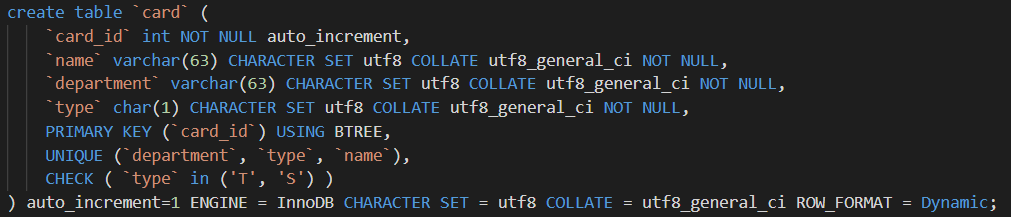
其中，admin表记录了“图书管理系统”不同管理员的管理员序号、登录密码及姓名，其详细结构设计代码见下图：



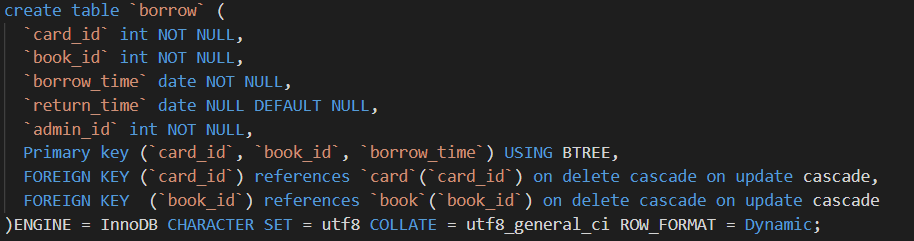
book表记录了“图书管理系统”中书籍的存储情况，包含了不同书籍的序号、类别、书名、出版社、出版时间、作者、价格及当前库存，其详细结构设计代码见下图：



card表主要负责“图书管理系统”中已注册借书证的管理，其中包含了借书证序号、借书证拥有者姓名、部门及借书证类别，其详细结构设计代码见下图：



borrow表主要记录“图书管理系统”中借还书籍的情况，其包括借换所用借书证序号、借还书籍序号、借书时间、还书时间及处理管理员序号等属性，其详细结构设计代码见下图：



4.模块功能演示

（1）登录界面

登录界面UI见下图，用户可在该界面对映横线处输入账号与密码，点击登录后即可登陆“图书管理系统”。



对映响应操作：在点击登陆按钮后，网页记录用户输入的账号与密码，提交含有信息的表单，在check.php文件中读取admin数据库中对映内容并进行比较；若账号密码匹配成功，则成功登录；反之则登录失败。

css优化：背景载入，横线式输入框设计，渐变色按钮设计，输入框选中高亮，信息呈现模块阴影......

1. 管理员主页

登陆成功后，进入管理员主页，交互UI见下图。其中显示了本次登录的管理员姓名与序号，页面中上部存有五个按钮（其中三个为下拉式菜单），鼠标选取并点击需要进入的模块即可进行页面的切换。

对映响应操作，点击对映按钮后，网页界面根据url进行指向性跳转。

css 优化：下拉式菜单，鼠标触碰按钮的颜色及手型改变......



1. 图书管理模块

该模块按钮本身不可点击，但其下五个子模块可以点击并进行相应的跳转。

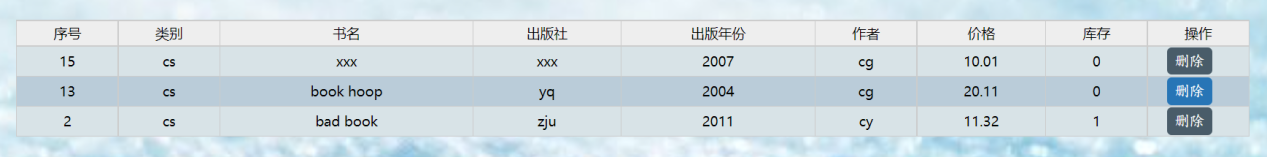
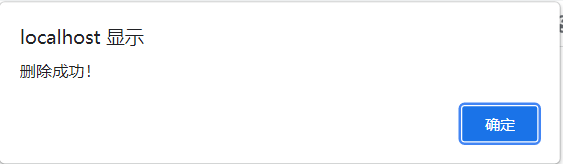
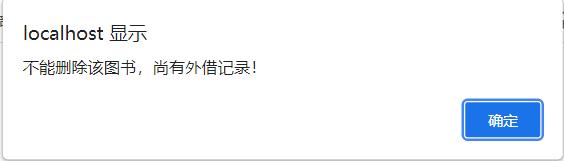
3.1 图书查询模块

在该模块中，用户可以根据已有信息进行对映书籍的查询，在该模块对映的交互UI中，并不需要填写所有输入框，只需有选择的填入已知的部分信息即可，系统便会进行相应的查询操作。

其中，类别仅支持精确查询，书名、出版社、作者信息支持模糊查询，而出版年份、对映价格则支持范围查询；在输入全部信息后，我们可以使用英语的输入方式，要求系统按照指定属性进行排序，点击查询按钮后，结果便会在页面下部呈现。

如上图操作中，我们使用筛选条件 类别为cs，年份不超2015年，价格高于10元，且结果按照库存量进行排序，筛选结果。

另外，在本查询模块中，我们在每一条记录的末尾都添加了“删除”操作按钮；若对映书籍没有任何外借记录，则在按下按钮后，该条记录便被成功删除；若该书存在外借记录，则删除失败。



对映响应操作：对查询部分而言，在点击查询按钮后，填入输入框的信息被提交，在文件对映php语言部分进行数据库的连接筛选与比较，最后将筛选所得结果呈现于UI交互界面；对于删除按钮而言，每次点击后，则通过php语句连接数据库，询问书籍是否存在对映的外借记录，若不存在，则将数据库中对映数据进行删除。

css 优化：文本框的选中高亮，文本框内提示字体的样式改变，记录的选中高亮，删除按钮的鼠标经过改变颜色与手型......

3.2 图书修改模块

在该模块中，用户可以通过书籍的序号搜寻对映的唯一记录，并可以有选择的输入需要修改的信息（直接对映输入需要修改的信息即可）；注意，输入修改信息时一定要同时输入对映的书籍序号，否则会造成修改失败的情况。

在本模块中，不支持修改图书的现有库存，具体交互UI界面见右图：

若仅输入书籍序号，或仅输入修改信息均会引发相应的错误提示窗口。

如下图中，我们选定书籍序号1，并在作者对映的输入框中输入 xxx ，点击修改按钮后，得到如下结果；我们可以看到，书籍序号1对映的作者栏信息已经从myx被更改为 xxx.



对映响应操作：在点击修改按钮后，表单提交已记录的数据，在本页面相应的php语言中，连接数据库并验证相应记录的合法性后，使用update语句更改数据库中的内容，再通过select语句输出新的查询结果。

css 优化：基本同前风格。

3.3更改库存模块

在本模块中，管理员可以对书籍的库存进行修改。

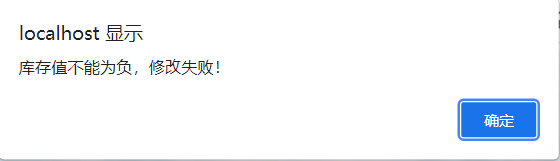
首先，此模块也支持书本序号的唯一记录查询，管理员可以通过输入书籍序号 + 库存变化量（带符号整数）来更改库存；注意，库存最终记录值不能为负，因而系统会进行校验，对非法的库存修改作出拒绝，UI界面如下：



对此功能模块，我们选择书籍序号为1的书籍，向其中加入十本库存，点击修改按钮，可以得到如下结果；我们可以看出，书籍序号为1的书籍对映的库存，确实增加了10本，来到了12册。



此时我们尝试减少书籍序号为1的书籍对映的库存13本，我们收到以下的错误信息，说明系统拒绝了此次修改。



对映响应操作：与其他图书信息修改方式相似。

css 优化：基本同前风格。

3.4单本入库模块

在该模块中，系统支持全信息输入的单本入库，其中输入信息必须全部包含类别、书名、出版商、出版年份、作者、价格以及初始库存量；注意，对于已有的书籍，系统会拒绝重复加入；若成功入库，则系统会在页面的中下部返回该书籍对映的唯一书籍序号，具体UI界面实现如下：

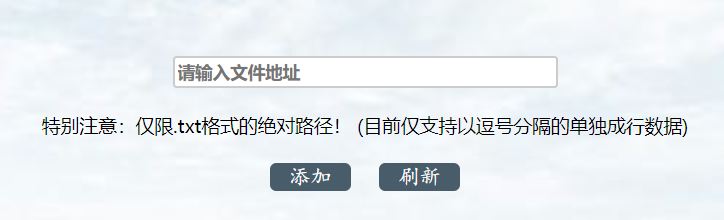


对映响应操作：点击刷新按钮，则页面被重置；若点击添加按钮，则系统获得输入框中已有的信息（通过表单的形式），在确认提交数据的完整性（内容）、无重复性及正确性（类型）后，使用php语言将相应的数据记录添加进数据库系统中，并返回相应的书籍序号。

css 优化：基本同前风格。

3.5批量入库模块

在该模块中，支持以文件形式的大规模书籍信息的导入；但其中存有对映的多条限制，首先文件仅支持.txt格式，且需要输入绝对路径；再者，.txt文件中内容数字正常输入，字符串信息需要用’’标识，且各元素之间需要以’,’分割；最后，每一条插入语句都需要包含单本入库的全部信息内容，其对映的UI界面如下：



对映响应操作：点击刷新按钮，则页面被重置；若点击添加按钮，则系统获得输入框中已有的文件地址，并前往相应的地址打开文件，从文件中按行读入数据，在php语言部分将相应的数据记录逐条添加进数据库系统中；按文件添加启动了mysql的事务概念：若全部插入成功则提交事务；若其中一条插入数据存在错误，则会触发rollback机制，返回本次事务的所有操作。

css 优化：基本同前风格。

1. 借书证管理

该模块按钮本身不可点击，但其下两个子模块可以点击并进行相应的跳转。

4.1借书证查询

在该模块中，管理员可以通过借书证卡号或借书证持有者姓名对借书证信息进行查询，其中信息包括借书证序号，借书证持有者姓名，借书证所属学科（部门）以及借书证类型；在每一条借书证记录后，我们也相应的添加了删除操作按钮，具体功能（类似于）见图书查询模块“删除”按钮功能，其对映UI界面如下：



若我们在姓名栏输入C，点击查询后，可以得到以下结果；说明我们确实经过筛选，得到姓名仅为C的记录。

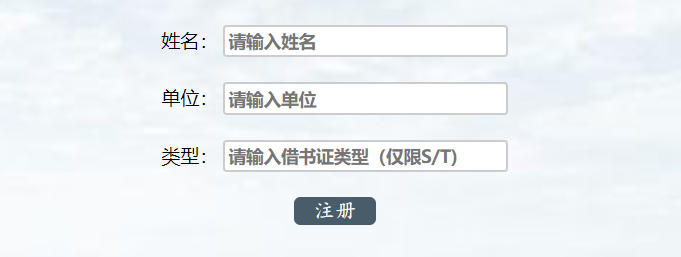


对映响应操作：同3.1图书查询模块。

css 优化：基本同前风格。

4.2借书证添加

在该模块中，管理员可以使用姓名、部门以及借书证类型三类信息创建新的借书证（借书证序号会自动生成，可通过二次查询获得）；添加借书证并不会一定成功，若相同信息的借书证已存在，则系统会拒绝借书证的二次注册，并且返回错误信息；其UI交互界面如下：



本界面的注册按钮功能相似于单本入库模块的添加按钮，均会在校对输入信息的正确性、无重复性后，在数据库对映card表中添加相应的记录。

对映响应操作：同3.4单本入库模块。

css 优化：基本同前风格。

1. 借还管理

该模块按钮本身不可点击，但其下三个子模块可以点击并进行相应的跳转。

5.1借还信息查询

在该模块中，管理员可以通过借书证序号或者借书证拥有者姓名查询相应的借还记录，具体查询方式与4.1借书证查询模块相似；在筛选后得到的记录中，若还书时间一项为（空），则表示该书处于外借而未还的状态；若记录所有信息均完善，则说明关于此书的借还过程已完成。

记录显示顺序为优先借书时间倒序，其次书籍序号正序排列；该模块具体的UI交互界面见下：

我们向卡号查询输入框中输入借书证序号1，点击查询后，可以得到以下结果；通过结果可知，我们确实只筛选了借书证序号为1的结束记录。

对映响应操作：基本同4.1借书证查询模块。

css 优化：基本同前风格。

5.2新的借阅

在本模块中，系统支持用户输入完备的信息以新增借书记录。

管理员所填信息必须包括借书证序号、所借图书序号以及借书日期，系统在检验输入信息不具备重复性后，会通过php语句自动向数据库中添加此条记录，并将相应书籍库存减少，并将还书时间暂且置为（空）。

另外，本功能模块提供了系统所有持有书籍的部分信息，方便管理员作出正确的信息填写，具体UI交互界面见下：



我们在相应的输入框中输借书证序号6，书籍序号1，借阅时间2023-4-12，点击借书后，可以得到以下结果；从结果可以分析得知，book2被成功借阅，库存减少1，且借书记录相应增加。



对映响应操作：基本同3.4单本入库模块。

css 优化：基本同前风格。

5.3归还图书

在该模块中，管理员可以通过输入借书证序号，图书序号，借书日期锁定到唯一的借书记录，并通过同时输入的还书日期，对数据库中borrow表指定记录进行一定的修改，进而完成一次完整的借还操作。

在输入框中，日期输入格式要求以 ” - ” 相连，以匹配数据库中信息；若书籍未归还，则归还日期栏显示为（空），具体UI交互界面如下



我们在输入栏中输入借书证序号为1，书籍序号为4，借阅日期为2023-04-08，归还日期为2023-04-12，点击还书，则可以得到以下结果；对映记录的还书日期进行了相应的填补：



对映响应操作：功能相似于3.2图书修改模块。

css 优化：基本同前风格。

1. 切换管理账户

该模块便于管理员快速的进行管理用户的变换。

对映响应操作：点击对映“切换管理账户”按钮后，网页返回原始登陆界面。

css 优化：基本同前风格。

1. **实验感想**

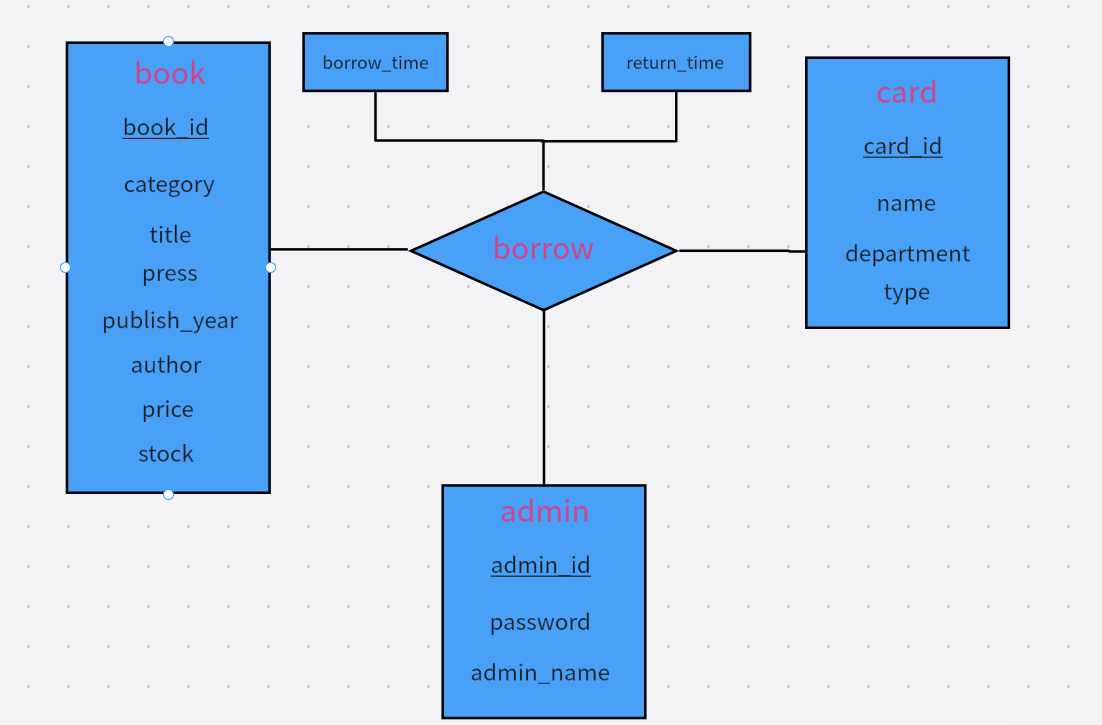
图书管理系统完结撒花，不得不说，一个人做lab5，工程量还是有一丢丢大的；感觉实验本身需要的数据库知识仅仅占了冰山一角，想要完全实现图书管理系统这一功能或者说在线应用模块，其与课本之间的跨度还是颇为明显。

当然，这次图书管理系统的实现我主要使用了html + php 的方式，虽然这样的方式可以一体化的解决前端和后端的问题，在实现数据库连接的同时也完成前端网页的图形化绘制，可以很好的满足实验的基本要求与进阶需要；但是这种前端加后端融合编写的方式，确乎不太合理，可能在后续的实际应用中未必能得到较大的使用。

因为本身是个小白，所以也算是借助本次图书管理系统的设计与编写，学习到了很多新的知识；像 Apache 服务的创立端口连接，php 的一些基础语法及前后端连接功效，html 编写网站与 css 进行适当美化等等，也算是对所学知识进行了一定的项目落实应用，因而还是成就感满满。

1. **思考题**

4.1 绘制该图书管理系统的E-R图



4.2 述SQL注入攻击的原理。在图书管理系统中，哪些模块可能会遭受SQL注入攻击？如何解决？

原理：当web应用向后台数据库传递[SQL语句](https://so.csdn.net/so/search?q=SQL%E8%AF%AD%E5%8F%A5&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/DM766924/article/details/_blank)进行数据库操作时，如果对用户输入的参数没有经过严格的过滤处理，那么攻击者就可以构造特殊的SQL语句，直接输入数据库引擎执行，获取或修改数据库中的数据。

如对于账号与密码的输入栏，如不对输入的账号密码字符串进行检验而直接并入SQL查询语句，则攻击者可以构造 “;drop table users;--”等语句，强行中断正在执行的查询语句，且对数据库进行攻击（删除整个用户表），并使用--符号注释其后的任何SQL语句，从而依赖于账号密码输入框对数据库进行攻击。

在本实验中，无论是登录模块，还是图书证查询、新增或图书查询、单本入库模块等等，都有可能作为SQL注入的对象，且批量入库模块更是支持.txt文件的读入，同样为SQL注入提供了更大的便利。

防范：

1.限制访问数据库帐号的权限

对于数据库的任何操作都是以某种特定身份和相应权限来完成的，SQL语句执行前，在数据库服务器端都有一个用户权限验证的过程，只有具备相应权限的帐号才可能执行相应权限内的SQL语句。因此，限制数据库帐号权限，实际上就阻断了某些SQL语句执行的可能。

2.参数化使用命令

参数化查询目前被视作是预防 SQL 注入攻击最有效的方法。参数化查询是指在设计与数据库连接并访问数据时，在需要填入数值或数据的地方，使用参数（Parameter）来给值。

在使用参数化查询的情况下，数据库服务器不会将参数的内容视为 SQL 语句的一部分来进行处理，而是在数据库完成 SQL 语句的编译之后，才套用参数运行。因此就算参数中含有破坏性的指令，也不会被数据库所运行。