重庆机电职业技术学院

课程设计报告

名 称： 图书馆管理系统设计与制作

二级学院： 信息工程学院

年 级： 2017级

专 业： 计算机应用技术

班 级： 01班

学 号： 1260720200668

姓 名： 韩潇

指导教师： 张永志

日 期： 2019 年 1 月 1 日

重庆机电职业技术学院

课程设计任务书

二级学院： 信息工程学院 专业： 计算机应用技术 年级： 2017级 班： 1班

1. **课程设计目的**

《综合实训》作为独立的教学环节，是计算机类相关专业实践性环节系列之一。其目的在于加深对JSP程序设计，数据库相关操作及web前端等开发等相关知识和技能的理解和巩固。要求学生熟悉java程序设计的一般方法和步骤，能使用JSP 语言开发可视化界面管理数据库和文件，具备初步的web软件开发编程能力，从中提高学生分析问题、解决问题的实际能力，培养学生的专项技能、职业素养和团队协作精神。

本系统将会涉及到图书馆日常管理工作的基本常见细节，诸如新图书的入库登记，图书馆所有书籍的分类管理，图书的查询，图书的借阅、退还手续的登记，费旧图书的清理撤除等，本系统所要实现的这些功能基本上涵盖了图书馆的日常管理工作，基本能够满足校园图书馆的工作人员的管理需要。

1. **课程设计要求**

**课题:**

1. **需求概述及简要分析**

**简述系统设计的目的及相关功能的描述，学生所学是否能够满足其基本功能**

1. **用户界面和功能设计（界面采用截图，具体功能通过图表或者文字描述，附相关的代码）**

（1）登录界面和功能（用户和管理面）

（2）管理员主界面和功能

（3）用户管理界面和功能

用户的添加，修改，删除和查找

（4）图书管理界面和功能

图书的入库，图书信息的更改，删除及图书的查找

（5）借阅管理界面和功能

借书管理和还书管理

（6）图书入库界面和功能

（8）用户查询功能

用户借还书的查找，图书馆图书的查找

**3、数据库设计**

（1）数据库

采用何种数据库，有无特别的技术或者靓点，附上创建数据库的相关代码。

网页是如何连接到数据库的，附相关代码

（2）数据表

表的功能及结构，附上创建表的SQL代码

（3）关键表之间的关系

说明：以上内容应有相关对象的说明，关键代码应有注释；**相同项目的必要、测试数据应该不同，如相同视为抄袭。**

**4、课程设计体会**

要求从：在该设计中遇到了哪些问题？如何解决的问题？通过该设计学习到了什么？团队成员协作之间碰到些什么问题？如何解决？从以上几个方面出发，写出项目在设计过程中的心得体会。(缺少程序设计体会，该课程设计题目视作0分处理。)

1. **成绩评定**

采用“形成性”考核方式，注重实际能力的考察与对过程的监控。具体是采用项目开发过程中的阶段性检查与作品评价相结合的考核方式。阶段性检查的实施能有效地监控学生的开发进度，并能为考察学生的实际开发能力提供客观、可靠的依据；作品评价则考察学生最终开发的项目和提交的相关报告以及最后答辩为准。

**课程成绩 = 过程成绩（30%）+ 答辩成绩（70%）**

1、过程成绩

占30分，主要包括实训时的态度、出勤等成绩。

2、答辩成绩

占70分，其比例分配如下：

（1）系统功能的实现（35分）

（2）程序的开发调试能力（10分）

（3）创新能力（15分）

（4）实训报告（10分）

表3《应用程序开发高级实训》答辩成绩评分表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **评分项** | **评分标准** | **分值** | **评分** |
| 1 | 系统功能的实现 | 系统功能是否完整、合理并符合项目需求 | 35 |  |
| 2 | 程序的开发调试能力 | 程序是否拥有遵循通用的编程规范，并作好注释；程序是否具备易读性和易维护性等。 | 10 |  |
| 3 | 创新能力 | 程序设计中是否具有新颖的思路和方法。 | 15 |  |
| 4 | 实训报告等文档 | 文档：是否有详细的文字说明，是否符合格式规范  （文档严重不符合规范，实训不合格） | 10 |  |
| 合计得分 | |  | | |

指导教师 签名 日期 年 月 日

系主任 审核 日期 年 月 日

目录

[一、 需求概述及简要分析 1](#_Toc534315709)

[二、 系统详细设计 4](#_Toc534315710)

[三、 系统测试与发布 38](#_Toc534315711)

[四、 课程设计体会 42](#_Toc534315712)

## 需求概述及简要分析

1. 需求分析

图书馆管理系统是图书馆管理工作中补课缺少的部分，对于图书馆的管理者和使用者来说都非常重要，但长期以来，人们使用传统的手工方式或性能较低的图书馆管理系统管理图书馆的日常事务，操作流程比较繁琐，效率相当低。而一个成功的图书馆管理系统应提供快速的图书馆信息检索功能，快捷的图书借阅，归还流程，为管理者提供充足的信息和快捷的数据处理手段。从读者与图书馆管理员的角度出发，本着以读者借书，还书快捷，方便的原则。

通过对图书的了解，确定此次实训系统具有以下功能：

A.学生用户端：查询图书，学生可以进行简单的数据查询。

B.管理员端：超级管理员能够对普通管理员进行增加和删除；对学生进行增加和删除；图书的增加、查询、修改和删除；图书外借和图书归还的管理，最后退出系统。

通过系统分析进行相关探讨，确定系统最终目标如下：

a.实现需求分析阶段A、B及客户提出的功能。

b.提高键盘的易用性，减少鼠标的操作功能。

c.使用较为流行的树状列表控件作为导航，直观的现实系统结构。

d.洽谈地使用监听器进行动作时间的监听。

f.在适当的场合灵活运用swing的各种控件，如面板、标签等。

1. 开发环境

系统是以MyEclipse 开发环境作为主要开发工具，使用Microsoft SQL server 2008作为关系数据库，配合功能强大的SQL查询语言实现建立关系数据库，访问数据库，对数据库的更新，较好地实现了预定的需求功能。使用了客户端/服务器结构，并使用Microslft SQL Server 2000创建中心数据库。

1. 可行性分析

通过对图书的各个方面进行了具体的了解之后，现对开发图书管理系统的可行性进行分析。

技术上可行：系统所需硬件设备，市场上销售且价格较低，甚至可以使用原有的设备，软件上，操作系统采用Windows系列操作系统，系统开发工具采用java，后台数据库采用SQL Server 2000这些软件在MIS开发中已被大量应用，技术上都比较成熟，因此在技术上是可行的。

经济上可行：由于中小型企业急需采用计算机来进行图书的管理，因此对开发图书管理系统大力支持。

管理上可行：中小型企业对图书有很多的管理体制，每位借书人均有会员卡号，同时图书馆还为每位借书人建立了档案。因此管理上可行。另外，同时制定了培训计划，并进行了实施MIS的物质准备和工作准备。

综上所述，开发图书管理系统在技术上、经济上、管理上都是可行的。

**系统总体设计**

1. 业务流程

根据需求分析的得到的现行业务处理流程，在反复研究后，首先确定目标系统的业务流程，其处理流程如图图 1。

图书登记和管理

用户验证

管理员

登录 合法用户

登记后

的图书

借书或归还图书

管理员

图 1图书管理系统的业务流程

针对图书管理业务流程，现具体描述其功能如下：

1．用户登录：系统对用户合法性进行检查。

2．图书入库：对新购进的图书分类进行编号，把图书基本信息录入计算机。

3．图书借阅：对已入库的图书，会员可以借阅，并进行登记。

4．图书归还：对会员已归还的图书进行归还登记。

5．信息查询：即对所有图书的借阅情况，或会员的基本情况和借阅情况进行查询。

6．管理措施：定义系统规定的借阅图书超期时间、超期处罚办法。

系统功能模块图

依据需求分析阶段得到的数据流图，在保证系统基本功能要求的前提下，结合系统业务流程确定系统必须具备的所有功能，由此给出图书馆管理系统的系统功能模块图如图图 2所示：

功能模块图：

图 2图书管理系统功能模块图

## 系统详细设计

* 1. 程序流程图

依据详细设计阶段的根本任务是确定应该怎样具体实现所要求的系统，经过这个阶段的设计工作，应该得出对目标系统的精确描述，从而在系统实现阶段可以把这个描述直接翻译成用某种程序设计语言书写的程序。所以在此给出系统的主程序流程图，如图图 3所示。

开始

管理员登陆

是否合法

No

Yes

显示主界面

选择操作

学生管理

图书管理

借阅管理

管理设置

归还管理

管理员注销

结束

图 3主程序流程图

* 1. 数据库的设计
  2. 数据库设计概述

计算机信息系统以数据库为核心，在数据库管理系统的支持下，进行信息的收集、整理、存储、检索、更新、加工、统计和传播等操作。

为了使现实世界的信息流计算机化，并对计算机化的信息进行各种操作，就是如何利用数据库管理系统、系统软件和相关硬件系统，将用户的要求转化成有效的数据结构，并使数据库结构易于实现用户新的要求的过程。 确切的说，数据库设计是指对于一个给定的应用环境，提供一个确定最优数据模型与处理模式的逻辑设计，以及一个确定数据库存储结构与存取方法的物理设计，建立起既能反映现实世界信息和信息联系，满足用户数据要求和加工要求，又能被某个数据库管理系统所接受，同时能实现系统目标，并有效存取数据的数据库。

根据软件工程的思想，数据库设计的周期可以划分为六个阶段： 规划阶段、需求分析阶段 、设计阶段、程序编制阶段 、调试阶段、运行和维护阶段 。数据库设计是要在一个给定的应用环境（DBMS）中，通过合理的逻辑设计和有效的物理设计，构造较优的数据库模式、子模式，建立数据库和设计应用程序，满足用户的各种信息需求。物理结构设计的原则如下：（1） 尽可能的减少数据冗余和重复（2） 结构设计与操作设计相结合（3） 数据结构具有相对的稳定性。

基于以上设计原则，系统设计了一个数据库，包含基本信息表（如图书信息表、系统用户表等）及几个实体联系建模后的表（如借阅信息表、会员信息表等）。为了加快系统访问的速度把这些表放在一个数据库中。

* 1. 数据字典

有了系统的数据流图后，还有相当多的数据信息如图书、借阅、归还等信息需要进一步描述，这就是需要定义数据字典，才能把现有的系统描述清楚。列出系统的主要数据字典。

名称：管理员

描述：记录管理员信息

定义：管理员信息=管理员ID+管理员密码

输入数据:管理员信息

输出数据:管理员信息

名称：图书信息表结构

描述：记录所有图书的基本情况

定义：图书信息=图书编号+ISBN号+索引号+图书类别+图书名称+语种+原名+页码+作者+价格+出版社+出版时间+登记日期+备注

输入数据：图书信息

输出数据：图书信息

名称：读者信息表结构

描述：记录所有读者的基本情况

定义：读者信息=读者编号+性别+姓名+类别+身份证号+办证日期+证有效期+工作单位+E-mail+联系电话

输入数据：读者信息

输出数据：读者信息

名称：借书登记表结构

描述：记录所有图书的借阅情况

定义：借书信息=读者编号+读者编号+读者类别+图书编号+图书类别+图书名称+原名+作者+出版社+借出日期+应还日期

输入数据：借书信息

输出数据：借书信息

名称：记录归还登记表结构

描述：记录所有图书的归还情况

定义：归还信息=读者编号+读者编号+读者类别+图书编号+图书类别+图书名称+原名+作者+出版社+借出日期+应还日期+实还日期+罚款

输入数据：归还信息

输出数据：归还信息

* 1. 数据库的逻辑设计

在需求分析阶段已完成该系统所有的数据分析。根据该阶段所建立的概念模型，已经得出满足系统设计要求的几个关系描述，该阶段的主要工作就是把前一阶段的成果转化为具体的数据库。下面给出概念结构设计得E-R图。

1：管理员实体E-R图：

管理员

图 4管理员实体E-R图

2：会员信息实体E-R图：

读者id 用户

密码

图 5读者实体E-R图

3：图书信息实体E-R图：

图书编号 图书

价格

图 6图书实体E-R图

4：借书信息实体E-R图：

读者编号 借书

书籍作者

图 7借书实体E-R图

6：还书信息实体E-R图：

读者编号 还书

书籍作者

图 8还书实体E-R图

7：总体E-R图：

读者管理

管理员

图书管理

读者

图书

归还

借阅

1 1

N N

M M N

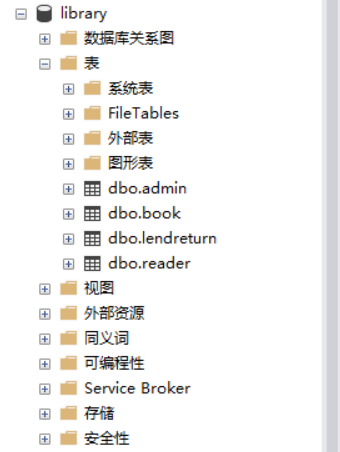
N N

图 9总体E-R图

* 1. 数据库物理设计

数据库

SQL Server 2008 具有很强的完整性与课伸缩性，具有较低的价格与较高的性能。数据库的树形结构图如图所示：



数据库图

数据库在java中的主要代码如下：

package db;

import java.\*;

import java.beans.Statement;

import java.sql.Driver;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import javax.jms.Connection;

public class Conndb {

private ResultSet rs=null;//定义rs为空

private java.sql.Connection conn=null;//定义conn为空

private java.sql.Statement stmt=null;//定义stmt为空

public Conndb(){

String user="sa"; //数据库用户名

String databaseName="tsgdb";

String url="jdbc:sqlserver://localhost:1433;"+databaseName;//定义jdbc的url对象

String password="";//数据库密码

//System.out.println(url);

String driverClas="com.microsoft.sqlserver.jdbc.SQLServerDriver";//定义驱动程序class对象

try {

Class.forName(driverClas);//加载驱动

conn=DriverManager.getConnection(url, user, password);//创立对象con，驱动程序连接url,用户，密码

conn.setAutoCommit(true);//每执行一条sql语句，就提交一次，默认为true

} catch (Exception e){//异常处理

System.out.println("-----------");

System.out.println("数据库连接错误");//输出打印数据库连接错误

System.out.println("-----------");

e.printStackTrace();

}

}

/\*\*

\* 用于执行数据库查询操作

\* @param sql

\* @return

\*/

public ResultSet myExcuteQuary(String sql){

try {

//System.out.println(sql);

stmt=conn.createStatement();//定义查询数据对象

System.out.println("sql=+sql");

rs= stmt.executeQuery(sql);//执行查询，得到查询表的结果集

while(rs.next()){rs.getString("State");}

} catch (SQLException e) {

System.out.println("-----------");

System.out.println("数据查询出错");//输出打印数据库连查询出错

System.out.println("-----------");

e.printStackTrace();//

}

return rs;//返回rs

}

public int myExcuteUpdate(String sql){

//System.out.println(sql);//测试输出

try {

stmt=conn.createStatement();//定义查询数据对象

System.out.println("sql=+sql");

} catch (Exception e) {

System.out.println("-----------");

System.out.println("数据查询出错");

System.out.println("-----------");

//e.printStackTrace();

}

return 0;//返回false

}

/\*\*

\* 释放与数据库的相关资源

\*/

public void close(){ //释放资源

try {

if(rs!=null){

rs.close();

}

if(stmt!=null){

stmt.close();

}

if(conn!=null){

conn.close();

} //关闭数据库，释放所占用资源

} catch (Exception e) {

System.out.println("数据库关闭报错");

}

}

}

Admin表是用户记录用户的信息（包括：管理员id，用户名，密码），如表表 1

|  |  |  |
| --- | --- | --- |
| **字段名称** | **数据类型** | **数据长度** |
| adminid | 管理员id | 20 |
| password | 密码 | 50 |
| name | 姓名 | 30 |

表 1

Lendreturn表用于记录图书的全部类型，（包括：自动id，读者id，书名id，管理员id，状态，借书时间，还书时间），如表表 2

|  |  |  |
| --- | --- | --- |
| **字段名称** | **数据类型** | **数据长度** |
| id | 读者编号 | 20 |
| autoid | 自动 | 长整形 |
| bookid | 书名id | 20 |
| adminid | 管理员id | 20 |
| state | 状态 | 10 |
| lendtime | 借书时间 | 日期时间类型 |
| returntime | 还书时间 | 日期时间类型 |

表 2

Book表用于记录书籍的详细信息( 包括：书籍编号，书名，ISBN，价格)，如表表 3

|  |  |  |
| --- | --- | --- |
| **字段名称** | **数据类型** | **数据长度** |
| bookid | 书籍编号 | 20 |
| bookname | 书名 | 50 |
| isbn | ISBN | 20 |
| price | 价格 | 浮点型 |

表 3

**reader表用于记录用户登录的详细信息（包括：用户id，用户名，密码，性别，电话），如表表 4**

|  |  |  |
| --- | --- | --- |
| **字段名称** | **数据类型** | **数据长度** |
| id | 管理员id | 长整形 |
| name | 姓名 | 50 |
| Pw | 密码 | 50 |
| sex | 性别 | 50 |
| tel | 电话 | 50 |

表 4

数据库代码

--create database library

/\* use library

create table admin

(id int not null primary key,

name varchar(50),

sex varchar(50) not null,

pw varchar(50) not null,

tel varchar(50) not null

) \*/

/\*create table book

(bookid int not null primary key,

ISBN varchar(20),

bookname varchar(50),

state varchar(20),

price float

)\*/

/\*create table lendreturn

(

id varchar(20) primary key,

adminid varchar(20),

readerid varchar(20) not null,

bookid varchar(20) not null,

bookname varchar(20) not null,

ISBN varchar(20) not null,

state varchar(20) not null,

lendtime date ,

returntime date

)\*/

/\*create table reader

(id int not null primary key,

name varchar(50) not null,

pw varchar(50) not null,

sex varchar(50) not null,

tel varchar(50) not null

)\*/

* 1. 登陆界面

当系统登陆时，首先出现的是一个用户权限登陆的界面, 权限设置主要是维护系统的安全性和完整性。拥有管理员权限的操作员能对其他操作员进行相应的权限设置，没有权限的操作员不能对相应的窗口进行操作。如下图图 10所示：

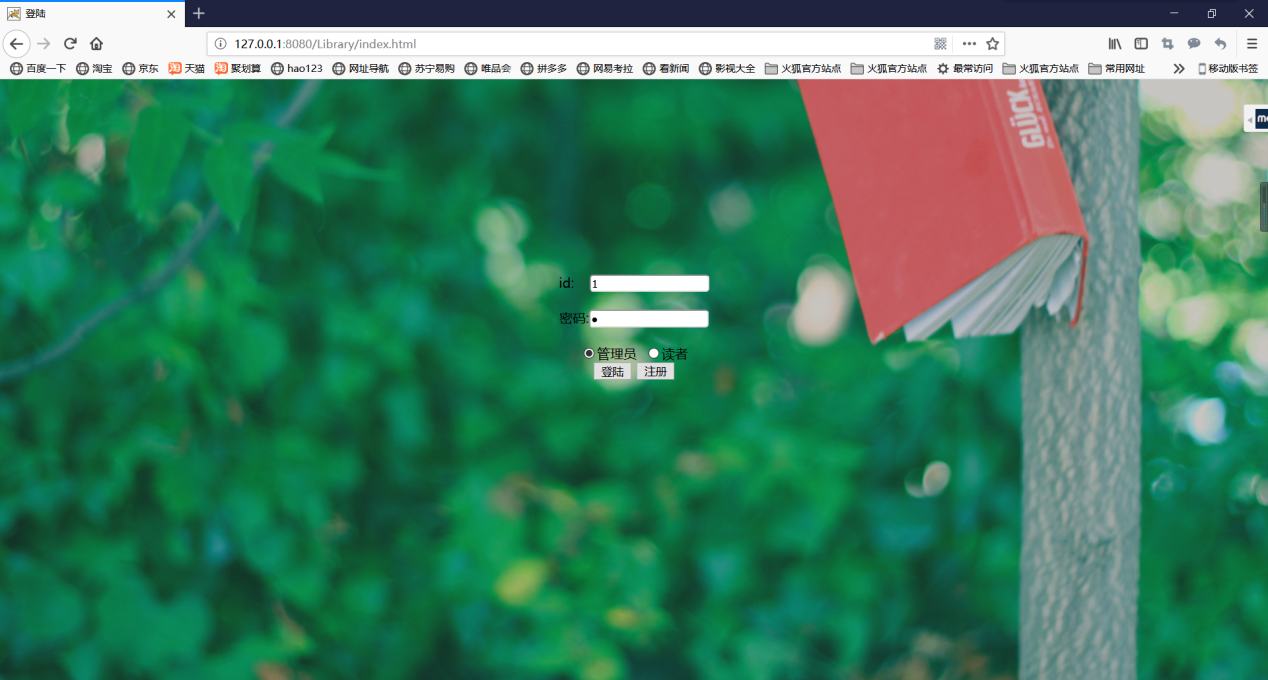


图 10

这是进入系统时的身份验证，用户首先要从软件开发者那里申请用户名和密码，才可以进入。

代码如下：

<title>登陆</title>

<style type=*"text/css"*>

*#a1* {

height: *100%*;

width: *100%*;

text-align: *center*;

padding-top: *15%*;

}

**body** {

background-image: *url(images/11.jpg)*;

background-repeat: *no-repeat*;

background-position: *center center*;

}

*#id* *#pw* {

border-radius: *0.25em*;

}

**a***:link* {

text-decoration: *none*;

}

**a***:visited* {

text-decoration: *none*;

}

**a***:hover* {

text-decoration: *none*;

}

**a***:active* {

text-decoration: *none*;

}

</style>

<meta http-equiv=*"Content-Type"* content=*"text/html; charset=utf-8"*>

</head>

<body>

<div id=*"a1"*>

<form action=*"login.jsp"* method=*"post"*>

id: &nbsp;&nbsp;&nbsp;<input type=*"text"* style="border-radius: *4px*" name=*"id"* id=*"id"*><br><br>

密码:<input type=*"password"* name=*"pw"* style="border-radius: *4px*" id=*"pw"*><br><br>

<input type=*"radio"* id=*"gl"* name=*"gl"* value=*"0"* checked=*"checked"*>管理员&nbsp;

<input type=*"radio"* id=*"gl"* name=*"gl"* value=*"1"*>读者<br>

<input type=*"submit"* id=*"login"* value=*"登陆"*>

<a target=*"\_top"* href=*"reader/add.html"*> <input type=*"button"* id=*""* value=*"注册"*></a>

</form>

</div>

</body>

</html>

* 1. 添加图书

身份验证通过以后，点击可以使用系统的基本信息管理界面，这是管理员主要的输入信息部分，它即可以对数据进行输入。填写好各项信息后，单击保存按钮，系统将对这些信息进行处理。界面见下图图 11所示：



图 11

<body>

<div align=*"center"*>

<h2>添加图书</h2>

<form name=*"form2"* method=*"post"* action=*"selectsbn.jsp"*>

<font color=*"pink"*> <br></font> 请输入准备添加的图书的ISBN <input

type=*"text"* name=*"ISBN2"* id=*"isb2"*>

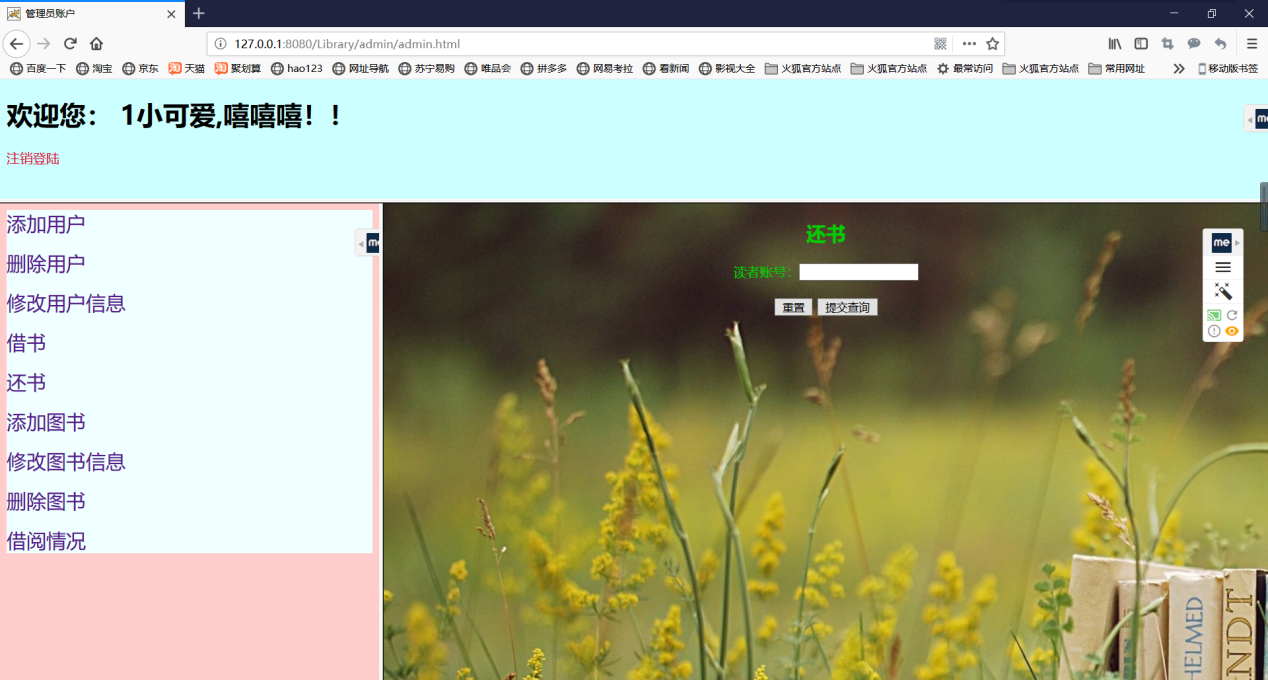
<input type=*"submit"* name=*"button"* id=*"button"* value=*"查询ISBN"*>

</form>

</div>

</body>

* 1. 还书

图 12

<body>

<div align=*"center"*>

<h2>还书</h2>

<form action=*"returnbook2.jsp"* method=*"post"* name=*"returnbook"*>

读者账号：<input name=*"readerid"* type=*"text"*><br>

<br> <input name=*"reset"* type=*"reset"* value=*"重置"*>

<input name=*"查询相关信息"* type=*"submit"*><br>

</form>

</div>

</body>

* 1. 借书



图 13

<body>

<div align=*"center"*>

<h2>借书记录查询</h2>

<form name=*"brrowbook1"* action=*"brrowbook1.jsp"* method=*"post"* >

读者账号：<input name=*"readerid"* type=*"text"*><br>

图书编号：<input name=*"bookid"* type=*"text"*><br>

<input type=*"reset"* value=*"重置"*>

<input type=*"submit"* value=*"查询相关信息"*>

</form>

</div>

</body>

* 1. 借阅情况



图 14

<body>

<div align=*"center"*>

<p>借阅情况查询 </p>

<hr>

<form name=*"form1"* method=*"post"* action=*"selectlendandreturn2.jsp"*>

<p>查询：<input type=*"text"* name=*"find"* id=*"find"*>

读者账号 <input type=*"radio"* name=*"radio"* value=*"readerid"* checked=*"checked"*> &nbsp;&nbsp;&nbsp;&nbsp;

书籍编号 <input type=*"radio"* name=*"radio"* value=*"bookid"*> &nbsp;&nbsp;&nbsp;&nbsp;

书名 <input type=*"radio"* name=*"radio"* value=*"bookname"*> &nbsp;&nbsp;&nbsp;&nbsp;

ISBN <input type=*"radio"* name=*"radio"* value=*"isbn"*> &nbsp;&nbsp;&nbsp;&nbsp;

<input name=*"sb"* type=*"submit"* value=*"查询借阅情况 "*>

</p>

</form>

</div>

</body>

* 1. 删除图书



图 15

<body>

<div align=*"center"*>

<p>请输入您要修改的信息的基本情况进行查询，注意查询项的选择</p>

<form name=*"form1"* method=*"post"* action=*""*>

<label for=*"find"*> 请输入您要查询的信息:</label>

<input type=*"text"* name=*"find"* id=*"find"*>

<input type=*"radio"* name=*"fd"* value=*"bookname"* id=*"fd\_2"* checked=*"checked"*>

书名&nbsp;&nbsp;&nbsp;&nbsp;

<input type=*"radio"* name=*"fd"* value=*"id"* id=*"fd-0"*>

id&nbsp;&nbsp;&nbsp;&nbsp;

<input type=*"radio"* name=*"fd"* value=*"ISBN"* id=*"fd-1"*>

ISBN&nbsp;&nbsp;&nbsp;&nbsp;

<input name=*"submit1"* type=*"submit"* value=*"查询"*>

<br><br><br>

</form>

<form action=*"deletebook.jsp"* method=*"post"* name=*"form2"*>

<%

String sb="";

String find="";

String fdtype="";

sb=request.getParameter("submit");

String sql="select \* from book";

**if**(sb!=""){

find=request.getParameter("find");

fdtype=request.getParameter("fd");

// out.print("测试 "+find+" "+fdtype+"<br>");

**if**(find==**null**||"".equals(find)){

out.print("<script type='text/javascript'>window.alert('查找的值不能为空')</script>");

}**else**{

**if**(fdtype.equals("bookname")){

sql=sql+" where bookname like '%"+find+"%'";

System.out.print("<br>"+sql+"<br>");

// out.print("<br>"+sql+"<br>");

}

ConnDB conndb=**new** ConnDB();

ResultSet rs=conndb.myExcuteQuary(sql);

out.print("<table border='1' width='98%'>");

out.print("<tr><td>编号</td><td>书名</td><td>价格</td><td>ISBN</td><td>删除</td></tr>");

**while**(rs.next()){

String bookid=rs.getString("bookid");

String bookname=rs.getString("bookname");

**float** price=rs.getFloat("price");

String ISBN=rs.getString("ISBN");

%>

<tr>

<td><%=bookid %></td>

<td><%=bookname %></td>

<td><%=price%></td>

<td><%=ISBN%></td>

<td><input name=*"id"* type=*"checkbox"* value=<%=rs.getString("bookid") %>></td>

</tr>

<%

}

%>

<tr>

<td colspan=*"7"* align=*"right"*>

<br><br><input type=*"submit"* value=*"提交"*>

</td>

<%

out.print("</table>");

rs.close();

conndb.close();

}

}

%>

<p>&nbsp;</p>

</form>

</div>

</body>

* 1. 删除用户

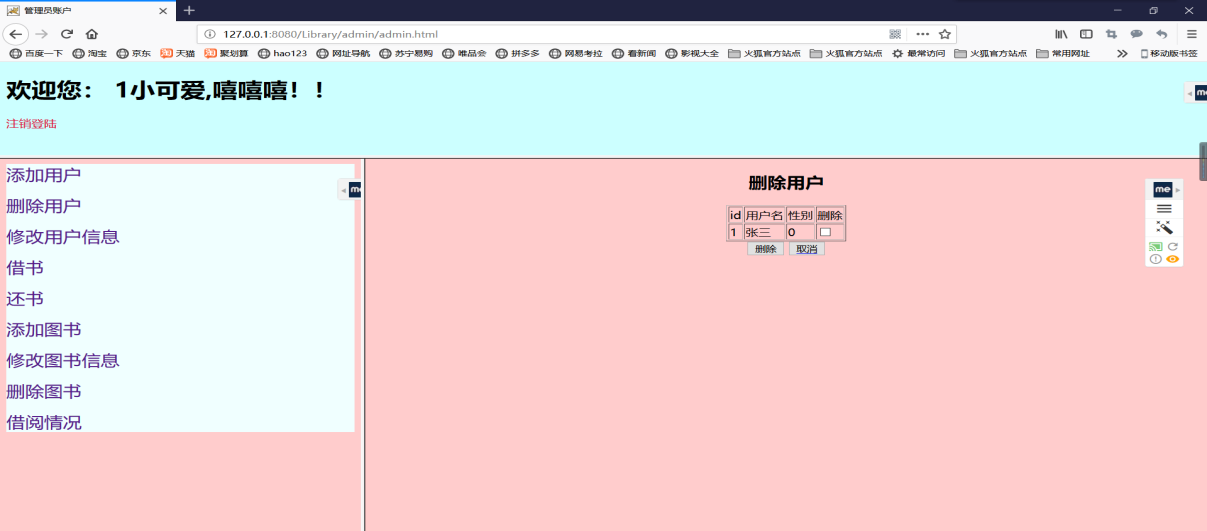


图 16

<body>

<%

String[] bookid=request.getParameterValues("id");

ConnDB conndb=**new** ConnDB();

**for**(**int** i=0;i< bookid.length;i++){

String sql="delete from book where bookid='"+bookid[i]+"'";

conndb.myExcuteUpdate(sql);

}

conndb.close();

// response.sendRedirect("selecteanddeletebook.jsp");

%>

</body>

* 1. 添加用户



图 17

<body>

<%

String[] bookid=request.getParameterValues("id");

ConnDB conndb=**new** ConnDB();

**for**(**int** i=0;i< bookid.length;i++){

String sql="delete from book where bookid='"+bookid[i]+"'";

conndb.myExcuteUpdate(sql);

}

conndb.close();

// response.sendRedirect("selecteanddeletebook.jsp");

%>

</body>

* 1. 修改图书



图 18

<body>

<div align=*"center"*>

<p>请输入您要修改的信息的基本情况进行查询,注意查询的选择</p>

<br>

<form action=*""* method=*"post"* name=*"form1"*>

<label>请输入您要查询的信息:</label>

<input type=*"text"* name=*"find"* id=*"find"*>

<input type=*"radio"* name=*"fd"* value=*"bookname"* id=*"fd\_2"* checked=*"checked"*>

书名&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;

<input type=*"radio"* name=*"fd"* value=*"id"* id=*"fd\_0"*>

id&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;

<input type=*"radio"* name=*"fd"* value=*"ISBN"* id=*"fd\_1"*>

ISBN&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<br>

<br> <br> <br> <br> <br>

<input type=*"submit"* name=*"submit1"* value=*"查询"*><br>

</form>

<form action=*""* method=*"post"* name=*"form2"*>

<%

String sb = "";

String find = "";

String fdtype = "";

sb = request.getParameter("submit1");

String sql = "select \* from book "; //查询数据库中表名为book中的信息

**if** (!(sb == **null** || "".equals(**null**))) {

find = request.getParameter("find");

fdtype = request.getParameter("fd");

}

**if** (fdtype.equals("bookname")) {

sql = sql + "where bookname like '%" + find + "%'";

// out.print("<br>" + sql + "<br>");

}

**if** (fdtype.equals("id")) {

sql = sql + "where bookid like '%" + find + "%'";

// out.print("<br>" + sql + "<br>");

}

**if** (fdtype.equals("ISBN")) {

sql = sql + "where ISBN like '%" + find + "%'";

// out.print("<br>" + sql + "<br>");

}

System.out.print(sql+"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

ConnDB conndb = **new** ConnDB();

ResultSet rs = conndb.myExcuteQuary(sql);

out.print("<table border='1' width='90%'>");

%>

<tr>

<td>id</td>

<td>书名</td>

<td>价格</td>

<td>ISBN</td>

<td>状态</td>

</tr>

<%

**int** i = 0;

**while** (rs.next()) {

String bookid = rs.getString("bookid");

String ISBN = rs.getString("ISBN");

String bookname = rs.getString("bookname");

**float** price = rs.getFloat("price");

String statei = "state" + bookid;

%>

<tr>

<td><input name=*"bookid"* type=*"text"* readonly=*"readonly"* value=<%=bookid%>></td><!-- readonly="readonly" -->

<td><input name=*"bookname"* type=*"text"* value=<%=bookname%>></td>

<td><input name=*"price"* type=*"text"* value=<%=price%>></td>

<td><input name=*"ISBN"* type=*"text"* value=<%=ISBN%>></td>

<%

String state = rs.getString("state");

**if** (state.equals("已还")) {

%>

<td><input name=<%=statei%> type=*'radio'* value=*'已还'*

checked=*"checked"* />已还</td>

<td><input name=<%=statei%> type=*'radio'* value=*'已借'*>已借</td>

<%

} **else** {

%>

<td><input name=<%=statei%> type=*'radio'* value=*'已还'*>已还</td>

<td><input name=<%=statei%> type=*'radio'* value=*'已借'*

checked=*"checked"* />已借</td>

</tr>

<%

}

%>

<%

}

rs.close();

conndb.close();

%>

<tr>

<td><br> <br> <input type=*"submit"* value=*"提交"*></td>

</tr>

</form>

</div>

<body>

* 1. 修改用户



图 19

<body>

<div align=*"center"*>

<h2>修改用户信息</h2>

<form action=*"alterreader.jsp"* metood=*"post"*>

<input type=*"text"* name=*"find"*>

<input type=*"radio"* name=*"radio"* value=*"1"* checked>用户名

<input type=*"radio"* name=*"radio"* value=*"2"*>id

<input type=*"submit"* value=*"查找"*>

<input type=*"reset"* value=*"取消"*>

</form>

</div>

</body>

* 1. 用户登录界面



图 20

<body>

<%

response.setCharacterEncoding("utf-8");

%>

<%

String id = request.getParameter("id");

String pw = request.getParameter("pw");

String login = request.getParameter("gl");

System.out.print(pw+"\*\*\*");

**if** (id == "" || pw == "") {

%>

<script type=*"text/javascript"*>

window.alert("输入不能为空，请重新输入");

window.history.back("login.html");

</script>

<%

}

**else**{

**if** (login.equals("0")) {

**if** (!admin.isadmin(id)) {

%><script type=*"text/javascript"*>

window.alert("账号不存在，请重新输入");

window.history.back("login.html");

</script>

<%

}

**else** {

String sql = "select \* from admin where id="+id+"";

ResultSet rs =admin.myExcuteQuery(sql);

**while** (rs.next()) {

String id1 = rs.getString("id");

String name = rs.getString("name");

String pw1 = rs.getString("pw");

System.out.print(pw1+"\*\*\*\*\*\*\*\*\*\*\*\*");

**if** (!pw.equals(pw1)) {

%>

<script type=*"text/javascript"*>

window.alert("密码不正确，请重新输入");

window.history.back("login.html");

</script>

<%

} **else** {

session.setAttribute("id", id);

session.setAttribute("name", name);

response.sendRedirect("admin/admin.html");

**return**;

}

%>

<%

}

}

}

**if** (login.equals("1")) {

**if** (!myBean.isUser(id)) {

%><script type=*"text/javascript"*>

window.alert("用户不存在，请重新输入");

window.history.back("login.html");

</script>

<%

}

**else** {

String sql = "select \* from reader";

ResultSet rs = myBean.find(sql);

**while** (rs.next()) {

String id1 = rs.getString("id");

String name = rs.getString("name");

String pw1 = rs.getString("pw");

**if** (!pw.equals(pw1)) {

%>

<script type=*"text/javascript"*>

window.alert("密码不正确，请重新输入");

window.history.back("login.html");

</script>

<%

} **else** {

session.setAttribute("id", id);

session.setAttribute("name", name);

response.sendRedirect("reader/reader.html");

**return**;

}

}

}

}

}

%>

</body>

* 1. 用户借书记录查询



图 21

<body>

<div align=*"center"*>

<%

String id = (String) session.getAttribute("id");

//out.print(id+"------");

String sql = "select \* from lendreturn where readerid=" + id + "";

ResultSet rs = db.bookmessage(sql);

%>

<table border=*"1"*>

<h3>借阅记录</h3>

<td>读者编号</td>

<td>读者姓名</td>

<td>书的编号</td>

<td>书名</td>

<td>ISBN</td>

<td>状态</td>

<td>借书时间</td>

<td>还书时间</td>

</tr>

<%

**while** (rs.next()) {

String adminid = rs.getString("adminid");

String readerid = rs.getString("readerid");

String bookid = rs.getString("bookid");

String bookname = rs.getString("bookname");

String ISBN = rs.getString("ISBN");

String zt = rs.getString("state");//状态

String jstime = rs.getString("lendtime");

String hstime = rs.getString("returntime");

%>

<tr>

<td><%=adminid%></td>

<td><%=readerid%></td>

<td><%=bookid%></td>

<td><%=bookname%></td>

<td><%=ISBN%></td>

<td><%=zt%></td>

<td><%=jstime%></td>

<td><%=hstime%></td>

</tr>

<%

**if**(bookid.equals("")){

%>

<script type=*"text/javascript"*>

window.alert('未查到相关记录');

window.history.back();

</script>

<%

}

}

%>

</table>

</div>

</body>

* 1. 用户图书查询

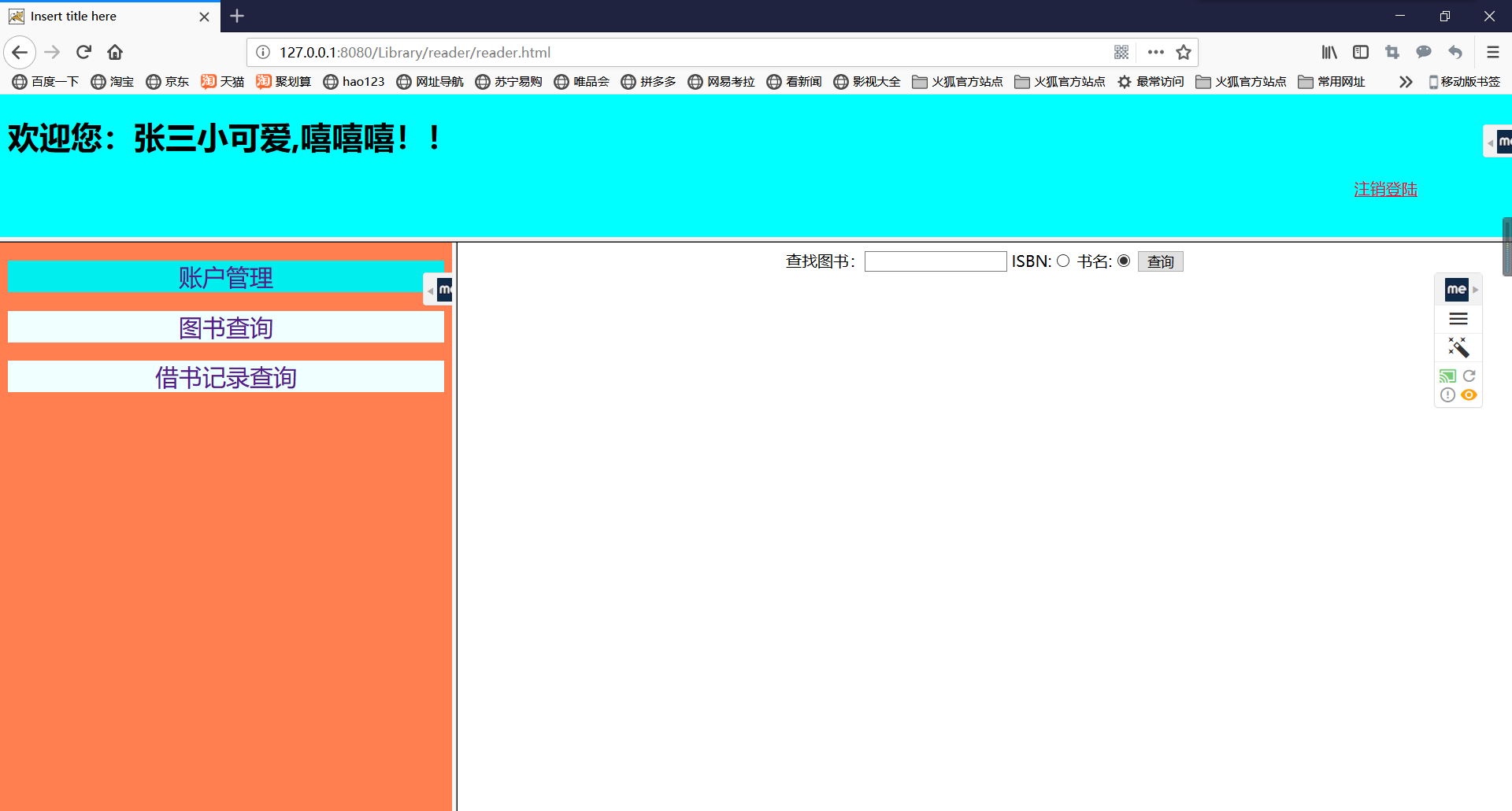


图 22

<body>

<div id=*"div1"*>

<form action=*"selectbook.jsp"* method=*"post"*>

查找图书：<input type=*"text"* name=*"find"*>

ISBN:<input type=*"radio"* name=*"radio"* value=*"0"*>

书名:<input type=*"radio"* name=*"radio"* value=*"1"* checked=*"checked"*>

<input type=*"submit"* name=*"tj"* value=*"查询"*>

</form>

</div>

</body>

* 1. 用户账号管理



图 23

<body>

<%@ include file=*"tool.jsp"*%>

<!-- 引用tool里面的id -->

<%

String id1 = (String) session.getAttribute("id");

String sql = "select \* from reader where id=" + id1 + "";

ResultSet rs = db.find(sql);

**while** (rs.next()) {

String id2 = rs.getString("id");

String name1 = rs.getString("name");

String pw = rs.getString("pw");

%>

<div id=*"div1"* align=*"center"* >

<table border=*"0"* width=*"100px"*>

<caption>我的信息</caption>

<tr>

<td>id</td>

<td>姓名</td>

<td>密码</td>

</tr>

<tr>

<td>

<%=id2%>

</td>

<td>

<%=name1%>

</td>

<td>

<%=pw%>

</td>

</tr>

</table>

<a href=*"alterpw.jsp"*><button>修改密码</button></a>

</div>

<%

session.setAttribute("pw", pw);

session.setAttribute("id", id);

session.setAttribute("name", name1);

}

%>

</body>

* 1. 注册

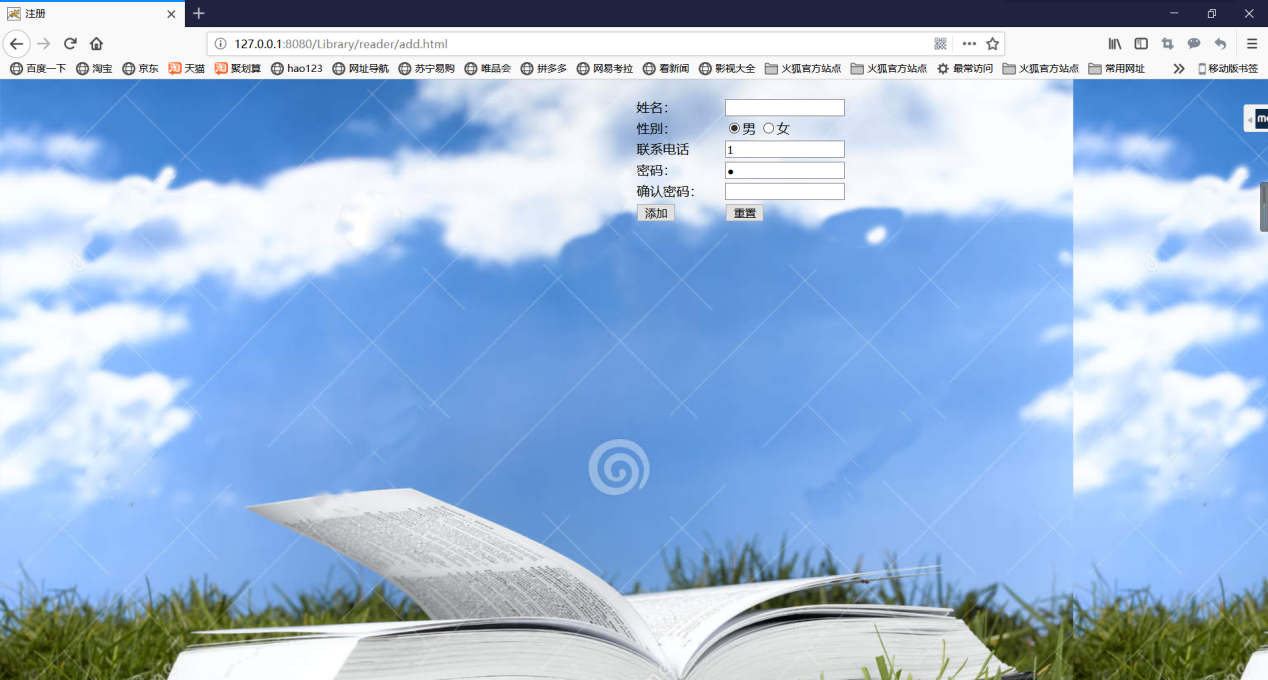


图 24

<body id=*"bd"*>

<div id=*"div"*>

<form action=*"add2.jsp"* metood=*"post"* onsubmit="return login(form);" name=*"form"*><table width=*"300"* border=*"0"*>

<tr><td> 姓名：</td> <td><input type=*"text"* name=*"name"* id=*"name"*></td> </tr> <tr> <td>性别：</td> <td><input type=*"radio"* name=*"sex"* value=*"男"* checked>男 <input type=*"radio"* name=*"sex"* value=*"女"*>女</td></tr> <tr> <td>联系电话 </td><td><input type=*"text"* name=*"tel"* id=*"tel"*></td></tr> <tr><td>密码：</td><td> <input type=*"password"* name=*"pw"* id=*"pw"*></td></tr> <tr> <td>确认密码：</td> <td><input type=*"password"* name=*"pw1"* id=*"pw1"*></td></tr> <tr <td><input type=*"submit"* value=*"添加"*></td><td><input type=*"reset"* value=*"重置"*></td </tr>

</table </form>

</div></body>

## 系统测试与发布

* 1. 系统测试的定义

系统测试是将经过测试的子系统装配成一个完整系统来测试。它是检验系统是否确实能提供系统方案说明书中指定功能的有效方法。为了发现缺陷并度量产品质量,按照系统的功能和性能需求进行测试。对于模块之间交互性比较强的软件,还会有单独的集成测试,用来发现模块接口之间的错误。

* 1. 图书管理系统测试的内容及方法

（1）图书管理系统测试的主要内容包括：

1）功能测试。功能测试主要包括：链接测试、表单测试、Cookies测试、设计语言测试、数据库测试。

2）性能测试。性能测试主要包括：连接速度测试、负载测试、压力测试。

3）可用性测试。可用性测试主要包括：导航测试、图形测试、内容测试、整体界面测试。

4）兼容性测试。兼容性测试主要包括：平台测试、浏览器测试。

5）安全测试。

（2）图书管理系统测试的方法：

1）功能测试：对图书管理系统的所有页面都进行了链接测试。所有的页面都能链接到指定要链接到的页面，所链接的页面均存在，而且系统中不存在有孤立的页面；对表单进行测试，如果输入信息不正确或者不输入信息就直接提交，会出现出错提示；通过互动用户填写的表单信息和调查信息的提交对数据库的存入和读取操作进行测试，数据库运行正常。

2）性能测试：通过宽带上网,无线上网,拨号上网的方式对图书管理系统的连接速度测试进行测试，都在可接受的范围内（少于三秒）。用图书管理系统压力测试Web-CT软件，对测试频率，持续时间、平均流量、平均响应时间进行测试，图书管理系统压力测试结果较好。

3）可用性测试：图书管理系统导航结构清晰、目的明确，经测试每个导航下的模块都能准确链接；页面字体使用CSS控制，确保了风格的一致，页面使用较少颜色组合，有较好的视觉效果；对图书管理系统内容进行抽样式调查未发现语法、拼写错误以及其他错误信息；图书管理系统整体界面布局合理，页面风格简洁，颜色搭配协调，保证了图书管理系统可用性良好。

4）兼容性测试：在安装了Windows、Unix、Linux的不同客户端上进行测试，系统能够正常运行；在安装了IE、腾讯TT、Maxthon Browser等不同浏览器的客户端进行测试，系统中JavaScript均可正常运行，框架和层次结构风格差别不大，所以图书管理系统兼容性较好。

5）安全测试：系统对管理员的帐号和密码的有效性进行了测试。只有管理员的帐号和密码确认有效后，才可以对图书管理系统进行维护工作。保证了本电图书管理系统的安全性。

* 1. 设计测试的方法

设计测试方案是测试阶段的关键技术问题。所谓测试方案包括预定要测试的功能，应该输入的测试数据和预期的结果。其中，最困难的问题是设计测试用的输入数据。

不同的测试数据发现程序错误的能力差别很大，为了提高测试效率降低测试成本，应该选用高效的测试数据。因为不可能进行穷尽的测试，选用少量“最有效的”测试数据，做到尽可能完备的测试就更重要了。

设计测试方案的基本目标是：确定一组最可能发现某个错误或某类错误的测试数据，已经研究出许多设计测试数据的技术，这些技术各有优缺点，没有哪一种是最好的，更没有哪一种可以代替其余的所有技术；同一种技术在不同的应用场合效果可能相差很大，因此，通常需要联合使用多种设计测试数据的技术。

* 1. 系统测试案例与分析

1 测试案例的内容 ：

周期:测试时间域。

层次:测试的层面。

类型:测试的分类。

系统:测试何系统。

分系统: 测试何分系统。

模块: 测试何模块。

平台: 测试的环境。

描述: 对测试问题说明与叙述。

目的: 测试的目标与期望。

此外，还包括测试文档号、测试设置、输入条件、测试结果和期望结果。

2 测试案例与脚本

1）测试案例与脚本设计主要包括以下内容:检查集成测试策略，制定测试数据目的和性能目标，找出关键测试条件，检查已有的在线测试脚本，输入或抓取在线测试脚本，

检查已有的用于测试的数据，生成测试数据，检查重要的数据，执行测试数据和测试脚本。

2）系统测试案例如下:

方 法： 功能模块测试方法

目 的： 测试录入功能的正确性

预期结果： 有出错警告

过 程： 在姓名栏中，没有选择任何选项，直接提交

输 入： 无输入

测试结果： 无法保存，有出错警告

比 较： 与预期结果相当

结 论： 正确

只有系统的测试工作在用户的协助下，不断的修改，才能达到完善的预期目标。本系统经过初步测试，基本上达到目标。

* 1. 图书管理系统发布

首先安装JDK，需要配置一些环境变量，再通过一个简单的java程序来测试JDK已安装成功。然后下载tomcat5.0，因为它不需要设置太多的系统变量，右击“我的电脑”，选择“属性”->“高级”->“环境变量”->“系统变量”，新建一个TOMCAT\_HOME，值设置成tomcat所在的路径，tomcat在安装之前会自动找到JDK的安装路径，一路点击"下一步"，经过一段时间的文件复制，最后"close"，完成tomcat的安装。

在外网服务器配置好tomcat相关服务，数据库链接等等。然后将工程放在tomcat/webapp目录下，重启tomcat, 敲入外网地址及配置好的目录就可以成功的发布图书管理系统了。值得强调的是必须确认所有的配置都要认真配置好后，方可重启tomcat。在本地站点测试成功后，就可以上传到服务器上。

## 课程设计体会

通过这次实训我了解到了我对有些代码不太熟悉 ，但我们小组加深了对软件开发流程的深入了解。

 在实习的这段时间内,我们小组终于完成了自己的图书管理系统,看着我们编辑的图书管理系统编译成功和运行时,心里的那种喜悦和无比的成就感简直就是爆棚。

 实训才是最能帮助我们检验自己的最好方法，能够让我们认识到自己的不足和缺点。