**软件需求分析说明书**

**选题：《学校教材订购系统》**

姓 名：钱多 唐昊东 王衡 徐奥 廖文轩 潘志红

指导老师：黄云洁

日 期：2022年6月 13 号

目录

[**第一章** **引言** 4](#_Toc105946386)

[**第二章** **系统工程** 4](#_Toc105946387)

[**2.1系统简介** 4](#_Toc105946388)

[**2.2技术要求和限制条件** 4](#_Toc105946389)

[**2.3系统可行性分析** 4](#_Toc105946390)

[2.3.1项目背景 5](#_Toc105946391)

[2.3.2可行性分析 5](#_Toc105946392)

[2.3.3结论意见 6](#_Toc105946393)

[**第三章** **需求分析** 6](#_Toc105946394)

[**3.1系统目标** 6](#_Toc105946395)

[**3.2功能目标** 6](#_Toc105946396)

[**3.3性能目标** 7](#_Toc105946397)

[**3.4系统功能分析** 7](#_Toc105946398)

[**3.5系统需求结构分析** 8](#_Toc105946399)

[**3.6系统性能分析** 9](#_Toc105946400)

[**3.7运行环境分析** 9](#_Toc105946401)

[**第四章** **概要设计** 9](#_Toc105946402)

[**4.1概述** 9](#_Toc105946403)

[**4.2用况建模** 9](#_Toc105946404)

[4.2.1执行者的简要描述 9](#_Toc105946405)

[4.2.2用况的简要描述 10](#_Toc105946406)

[**4.3静态建模** 10](#_Toc105946407)

[4.3.1标识候选对象 10](#_Toc105946408)

[4.3.2筛选候选对象 10](#_Toc105946409)

[4.3.3标识属性和操作 10](#_Toc105946410)

[**4.4动态建模** 11](#_Toc105946411)

[4.4.1顺序图 11](#_Toc105946412)

[**4.5系统数据库设计** 11](#_Toc105946413)

[4.5.1实体抽取 12](#_Toc105946414)

[4.5.2局部E-R图 12](#_Toc105946415)

[**第五章** **详细设计** 12](#_Toc105946416)

[ 用户登录界面 12](#_Toc105946417)

[ 师生登录界面 13](#_Toc105946418)

[ 订书/缺书单界面 13](#_Toc105946419)

[ 订购/缺书单数据记录界面 14](#_Toc105946420)

[ 管理员登录界面 14](#_Toc105946421)

[ 图书库界面 15](#_Toc105946422)

[ 添加书籍界面 16](#_Toc105946423)

[ 添加书籍后的图书库界面 16](#_Toc105946424)

[ 图书修改界面 16](#_Toc105946425)

[ 修改后的图书库界面 17](#_Toc105946426)

[ 订购/缺书单信息表界面 17](#_Toc105946427)

[**第六章** **编码** 18](#_Toc105946428)

[ 登录代码： 18](#_Toc105946429)

[ 师生主界面代码： 19](#_Toc105946430)

[ 订书/缺书单填写代码： 20](#_Toc105946431)

[ 订书/缺书单数据记录代码： 21](#_Toc105946432)

[ 管理员主界面代码： 22](#_Toc105946433)

[ 书籍添加代码： 24](#_Toc105946434)

[ 书籍修改代码： 25](#_Toc105946435)

[ 查看图书库代码： 26](#_Toc105946436)

[**第七章** **总结** 27](#_Toc105946437)

[**附录** 27](#_Toc105946438)

1. **引言**

随着计算机的发展和网络技术的应用，越来越多的以往靠人来手工完成的工作被计算机代替。日常管理工作也从以前繁琐的事务中解放出来，从而提高了工作效率。过去的教材订购模式大多以手工为主，图书采购后的入库，图书的销售，图书的查询大多停留在纸质手工层面。这种模式存在着效率低下，人工成本巨大，数据容易缺失。随着市场上书籍的种类和数量的增多，更新的速度越来越快，传统管理模式弊端出现：图书采购、库存、销售和核算的手工信息管理存在工作量大，效率低，图书的市场、库存、销售、读者反馈信息不能及时获取等问题。同时，由于不可避免的人为因素，造成数据的遗漏、误报等。为了更好地适应当前学校的业务需求，缓解手工管理存在的弊端，同时向师生提供更优质的服务，需要开发一个完整的信息系统将图书采购和销售管理链接起来，完成图书的进/出库管理、库存管理、销售管理、用户管理等一系列详尽、全面的控制和管理。

1. **系统工程**

**2.1系统简介**

学校教材订购系统可以细化为两个子系统：销售系统和采购系统。

销售系统的主要工作过程为：首先由教师或学生提交购书单，经教材发行人员审核是有效购书单后，开发票、登记并返给教师或学生领书单，教师或学生可以到书库领书。

采购系统的主要工作过程为：若是教材脱销，则登记缺书，发缺书单给书库采购人员；一旦新书入库后，即发进书通知给教材发行人员。

**2.2技术要求和限制条件**

当书库中的各种书籍数量发生变化（包括进书和出书）时，都应修改相关的书库记录，如库存表或进/出库表。

在实现上述销售和采购的工作过程时，需考虑有关的合法性验证。

系统的外部项至少包括：教师、学生和教材工作人员。

系统的相关数据存储至少包括：购书表、库存表、缺书登记表、待购教材表、进库表和出库表。

**2.3系统可行性分析**

可行性分析是指在软件项目计划阶段，用最小的代价在尽可能短的时间内研究并确定客户提出的问题是否有行得通的解决办法。本节将研究“学校教材订购系统”项目的解决方案可行性以论证实现本项目的可能性和一些前期的准备工作及工作条件，为工程的下一步设计做铺垫。

2.3.1项目背景

过去的教材订购模式大多以手工为主，图书采购后的入库，图书的销售，图书的查询大多停留在纸质手工层面。这种模式存在着效率低下，人工成本巨大，数据容易缺失。随着市场上书籍的种类和数量的增多，更新的速度越来越快，传统手工销售和人工管理模式弊端初现：图书采购、库存、销售和核算的手工信息管理存在工作量大，效率低，图书的市场、库存、销售、读者反馈信息不能及时获取等问题。开发适合学校的教材订购系统势在必行。

2.3.2可行性分析

技术可行性

* 数据库技术。
* JSP技术。
* 软件工程。

经济可行性

成本估算

* 人员费用:系统开发期2周，试运行期2周，开发人员3人。开发需3人30天，按每人5千/月工资计算，人员费用为15000元人民币。
* 硬件设备费:服务器1台10000元，微机3台12000元，打印机2台3000元，不间断电源1台2000元。合计27000元人民币。
* 软件费。Windows系统，SQL server，C环境。合计8000元人民币。
* 系统设备维护费:假定本系统的运行期为1年。系统维护费5000元，设备维护费6000元，材料费8000元。合计19000元人民币。
* 总计69000元人民币。

收益分析

* 减少工资。本系统累计可以综合提高工作效率达30%，可以减少现有20%的工作人员，现有人员按10人计算，可减少两人。每人月平均工资按4000元计算，节约人员工资4000\*12\*2=96000元。
* 增加销量。假定在原有基础上可以增加10%的销售量。可以增加2万元左右收入。
* 提高决策。本系统的建设可以及时获取图市场信息，读者反馈信息，畅销滞销图书的信息。估计每年可以增加1万元左右的收入。
* 库存积压。通过系统的计算机管理，可以及时获取信息，争取最有库存，提高资金的周转率。每年可以因此减少库存积压浪费在3万元左右。
* 总计156000元

累计1年获经济效益：156000 – 69000 = 87000元。

法律可行性

正版合法。

2.3.3结论意见

通过对项目整体进行可行性分析，该项目无论在操作可行行、技术可行性、经济可行性及社会可行性上均满足要求，因此，开发此系统的构想是可行的，可着手进行。

1. **需求分析**

需求分析的主要任务是确定系统“要做什么”，通过软件人员与用户的交流和讨论，进行细致的调查分析，准确理解用户的功能需求、性能需求、运行环境需求和操作界面需求。本节将在业务需求调查的基础上研究“教材订购管理系统”项目客户的功能需求、性能需求、运行环境需求和操作界面需求，为设计阶段提供设计依据。

**3.1系统目标**

在业务需求调查的基础上，结合现有的技术条件和力量，根据用户的需求，把系统划分为高内聚、低耦合的相应功能的子模块。对系统的功能和性能进行更翔实的需求分析，为设计阶段提供设计依据。以便系统能够对书店业务进行全面有效的管理，使得最终所确定系统能够满足用户的需求和实际应用需要。

**3.2功能目标**

系统功能目标如下。

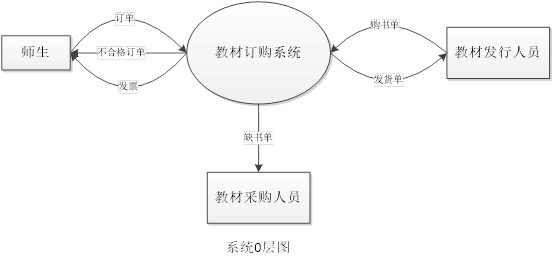
* 用户登录管理，具有权限检查机制，各级用户只能看到允许查看的系统消息。管理员拥有最高权限，拥有系统的所有操作权限。采购员拥有采购模块、查阅库存、查询书目信息的权限；发行员拥有销售模块、查阅库存、维护书目信息的权限。
* 基础信息管理，对教材订购系统业务流程中的基础数据进行维护，涉及图书信息、师生信息、图书供应商信息。
  + - 书目信息：书目信息的增加、删除、修改、查询。
    - 用户管理：用户信息的增加、删除、修改、查询。
    - 供应商管理：供应商信息的增加、删除、修改、查询。
* 采购管理，也称进货管理，负责处理从供应商采购图书的相关事务，完成图书采购和采购查询。
  + - 图书采购：从供应商那里采收新书，或是补充旧书库存。
    - 采购查询：查询历史采购记录。
    - 采购统计：按时间段对采购信息及到货情况进行查询。
* 销售管理，完成图书的销售相关事务，包括销售处理、退书处理、销售台账记录、销售记录查询和统计。
  + - 图书采购：完成图书销售和结算及销售记录的存储。
    - 退书处理：师生完成购买后，在合理的期限内可以退书并重新购买。
    - 销售查询：查询销售记录详单。
    - 销售统计：按时间、书名统计销售金额及销售情况。
* 库存管理，图书采购回来后，需要入库，图书报损或是销售后需要出库，该模块完成图书库存出入库管理，并提供库存报警设置和库存查询功能。
  + - 图书出入库管理：完成图书进入销售和图书报损的出库处理。
    - 库存查询：查询缺货详细情况。
    - 库存报警设置：设置商品缺货预警最低值。
    - 库存统计查询：统计缺货或货品充足的详细信息，生成缺货详单。

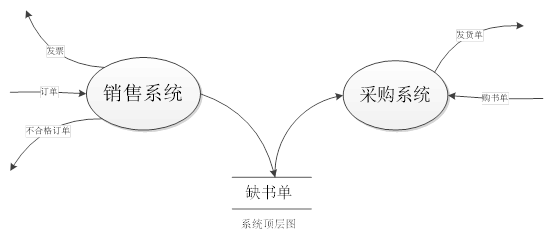
**3.3性能目标**

* 用户可以通过计算机进入系统查询所有在售图书、办理正常业务，系统界面清晰，操作方便。
* 系统可以进行联机进行采购、入库、出库、销售、统计等处理。
* 系统反应时间最慢不能低于2s。
* 一般职工经过简单培训就可以使用系统。
* 具有高可靠性和容错能力，不允许在工作时间停机，不允许丢失图书信息。
* 具有安全检查机制，非法用户不得使用系统。
* 具有权限检查系统，各级用户只能看到允许查看的系统信息。

**3.4系统功能分析**

系统设有采购、发行、管理等业务部门。教材订购管理系统数据流图如下。





**3.5系统需求结构分析**

教材订购管理系统功能目标划分为4大部分：登录管理、订购管理、销售管理和采购管理。系统功能结构图如下。





**3.6系统性能分析**

系统性能分析是对客户提出的各种性能需求进行综合分析，确定出合理、可信的系统性能要求。学校教材订购系统的性能方面，尽可能地提高系统的处理速度和能力，缩短响应时间和处理时间，提供简单、便捷的处理方式，保证系统正常工作，增加系统抗故障、抗干扰、防止非法用户访问的能力。

**3.7运行环境分析**

设备包括：服务器1台、微机3台、打印机2台、不间断电源1台、工作台4台、网络设备和布线。

支撑软件：Windows、SQL Server、JSP。

1. **概要设计**

概要设计解决“怎么做”的问题，从软件去求规格说明书出发，将需求分析产生的模型等分析结论进行转换，由此产生设计结论过程。本节将在需求分析的基础对“教材订购系统”中系统各部分模块将要实现的功能进行详细的分析，同时确定各个模块功能之间的逻辑结构，确定系统与用户间的交互关系，操作顺序和交互界面的组成，完成的数据库进行设计、概要设计结果将作为后期详细设计的基本依据。

**4.1概述**

经过前期的分析，对需要开发的系统已经有了一个清楚的把握，对系统的总体结构也有了充分的了解。该阶段要做的工作：在需求分析的基础上对待开发的系统各部分模块将要实现的功能进行详细的分析，同时确定各个模块功能之间的逻辑结构，确定系统与用户间的交互方式、操作顺序和交互界面的组成，最后对数据库进行设计。

**4.2用况建模**

用况建模的执行者和用况的简要描述如下。

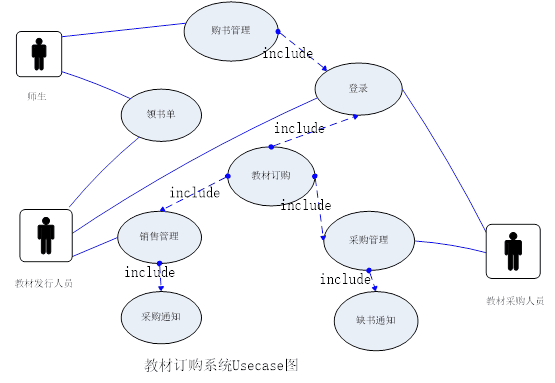
4.2.1执行者的简要描述

* 师生，向系统提交订购单的主要人群。
* 教材发行人员，处理订购单和维护图书信息的人。
* 教材采购人员，负责采购图书的人。

4.2.2用况的简要描述

* 购书管理，在订单中添加图书、删除图书、显示图书。
* 登录，对学校师生进行用户登录，对管理人员进行管理员登录。
* 教材订购，在线订购教材，包括销售管理、采购管理、填写购书单信息等。
* 销售管理，对购书单进行审核，审核通过即发送领书单。
* 采购管理，查阅缺书单并采购缺少的相应图书。

学校教材订购系统的用况图如下。



**4.3静态建模**

静态建模主要是构建类图，其步骤如下。

4.3.1标识候选对象

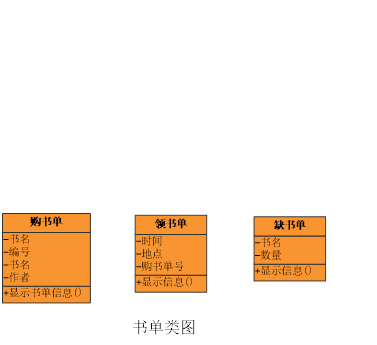
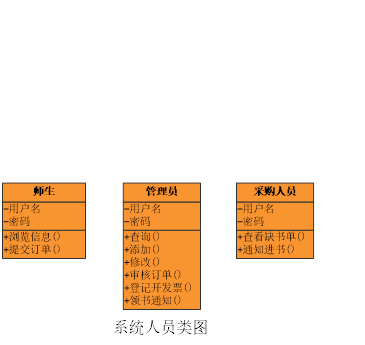
候选对象如下。

* 外部实体有销售系统、采购系统、登录系统、信息管理系统、师生、教材发行人员、教材采购人员。
* 需要存储、处理的信息有教材的书名、作者、单价、出版社，购书单，领书单，缺书单、进书单、进/出库表。

4.3.2筛选候选对象

在外部实体中，销售系统和采购系统同属于教材订购系统，登录系统和信息管理系统只是作为外部执行者参与创建和维护师生信息和教材信息。师生 、发行人员、采购人员、表单都有明确的属性和操作，显然应该成为最终的对象。考虑到表单又可分为购书单，领书单，缺书单、进书单、进/出库表等，因此增加这些对象。综合上述分析，最终得到教材订购系统包含以下类：师生、教材发行人员、教材采购人员、购书单，领书单，缺书单、进书单、进/出库表。

4.3.3标识属性和操作

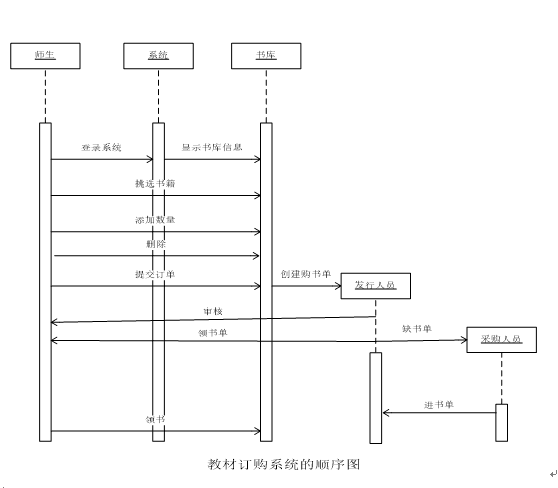


**4.4动态建模**

动态建模用来描述系统的动态行为，显示对象在系统运行期间不同时刻的动态交互。UML中用状态机、活动图、顺序图、通信图和协作图来建立动态模型。这里选用顺序图来描述教材订购系统中对象的交互行为。

4.4.1顺序图

顺序图关注于消息的顺序，即对象间消息的发送和接收的顺序。该图只描述了形成一张购书单的消息发送顺序，省略了形成一张购书单后再次购书等复杂情况。



**4.5系统数据库设计**

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，该阶段的主要工作是把前一阶段的成果转化为具体的数据库。

4.5.1实体抽取

* 抽取的实体有师生、管理员、图书、供应商。实体属性如下：
* 师生（教工号/学号、姓名、用户名、密码、学院、职位、联系电话）；
* 管理员（管理员账号、密码、姓名、岗位、联系电话）；
* 图书（图书编号、书名、作者、单价）；
* 供应商（供应商号、名称、地址、联系电话）。

4.5.2局部E-R图

销售系统E-R图如下



采购系统E-R图如下



1. **详细设计**

详细设计是软件工程中对概要设计的一个细化，主要是为软件结构图中的每一个模块确定他采用的算法和模块内数据结构，并用某种表达工具给出清晰的描述。本节将在需求分析的概要设计的基础上对待开发的系统各部分模块要实现的功能结构进行详细的设计，确定模块内部程序设计，并对数据库、用户界面进行详细设计。详细设计结果将作为后期系统实现的基本依据。

* 用户登录界面

界面说明：用户登录时需要输入的信息。



【师生/管理员登录】按钮触发的处理：验证用户的合法性，做不同处理。

处理流程：

取得用户输入的用户名、密码。

加密口令，传送到数据库并与用户信息表进行一致性验证：

If登录名and密码完全匹配

以对应身份进入系统主控界面获得相应的权限

Else

提示“登录失败”

* 师生登录界面

师生登录界面是师生进入系统后的主控界面，通过主界面的【订书/缺书单】和【注销】模块，用户在该界面中通过点击选择菜单进入到相应的子模块，同时也能浏览书库中相应的教材信息。



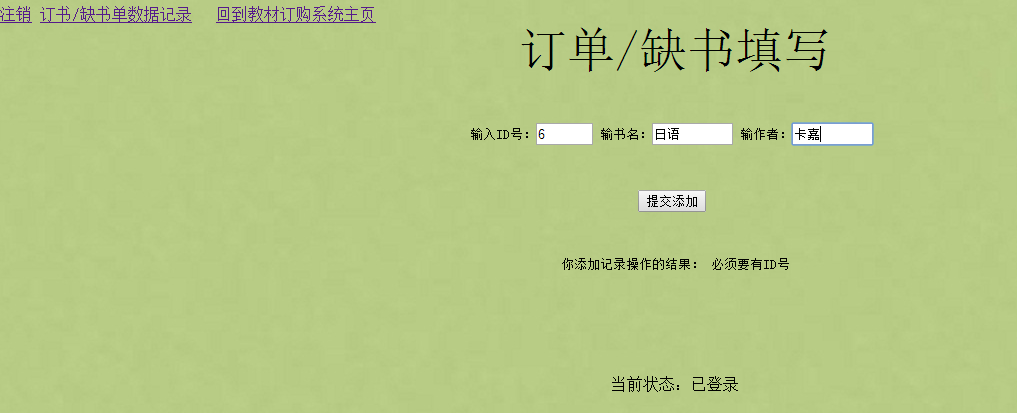
【上一页】按钮触发的处理：跳转至该界面的上一条记录。

【下一页】按钮触发的处理：跳转至该界面的下一条记录。

【提交】按钮触发的处理：跳转至文本框中填写值的页码。

* 订书/缺书单界面

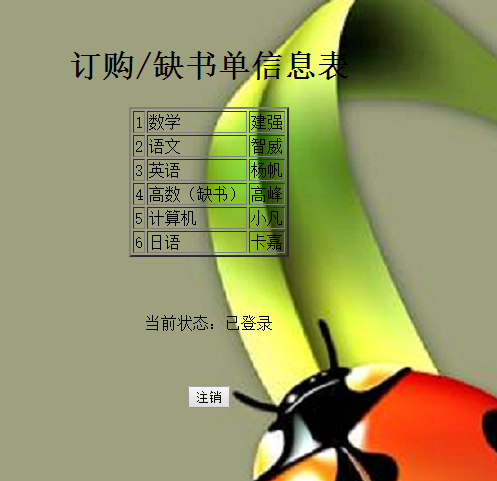
师生浏览完全部的图书信息，可填写订书/缺书单，在“输入ID号”文本框中填写图书ID号，在“输书名”文本框中填写对应的图书名称，并在“输作者”文本框中填写作者姓名。若在文本框末尾处添加“（）”符号，系统将自动标注为缺书状态。



【提交添加】按钮触发的处理：用于保存新增或修改的数据。

* 订购/缺书单数据记录界面

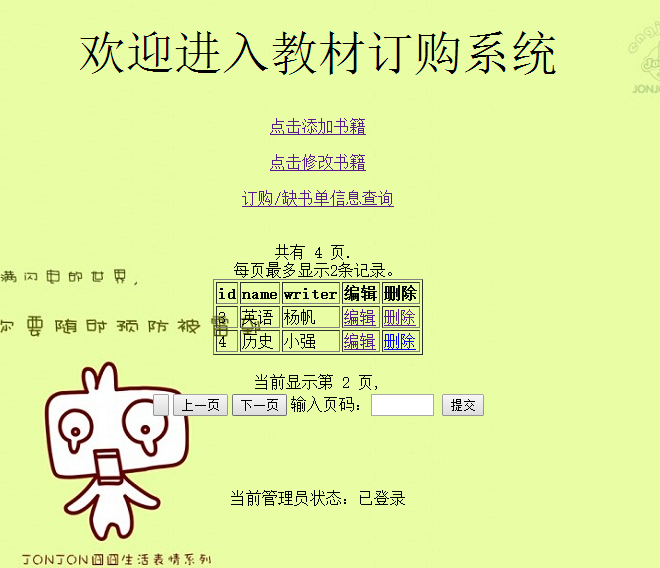
本界面可查看修改的教材信息是否成功的更新在数据库中。查阅完毕后可点击【回到教材订购系统主页】或【注销】退出登录。



【注销】按钮触发的处理：退回到系统登录主界面。

* 管理员登录界面

管理员登录界面是管理人员进入系统后的主控界面，通过主界面正中间的【点击添加图书】、【点击修改书籍】和【订购/缺书单信息查询】模块来管理教材订购系统的图书信息，管理员通过点击模块进入到相应的子模块，同时也能浏览书库中教材的所有图书信息。



【上一页】按钮触发的处理：跳转至该界面的上一条记录。

【下一页】按钮触发的处理：跳转至该界面的下一条记录。

【提交】按钮触发的处理：跳转至文本框中填写值的页码。

* 图书库界面

图书库界面可以完整的显示所有的图书信息，但该界面仅对管理员可见。



* 添加书籍界面

管理人员可在“输入ID号”文本框中填写图书ID号，在“输书名”文本框中填写相应的图书名称，并在“输作者”文本框中填写作者姓名。就可以添加新的教材图书信息到书库中并在图书库界面显示。



【提交添加】按钮触发的处理：用于保存新增或修改的数据。

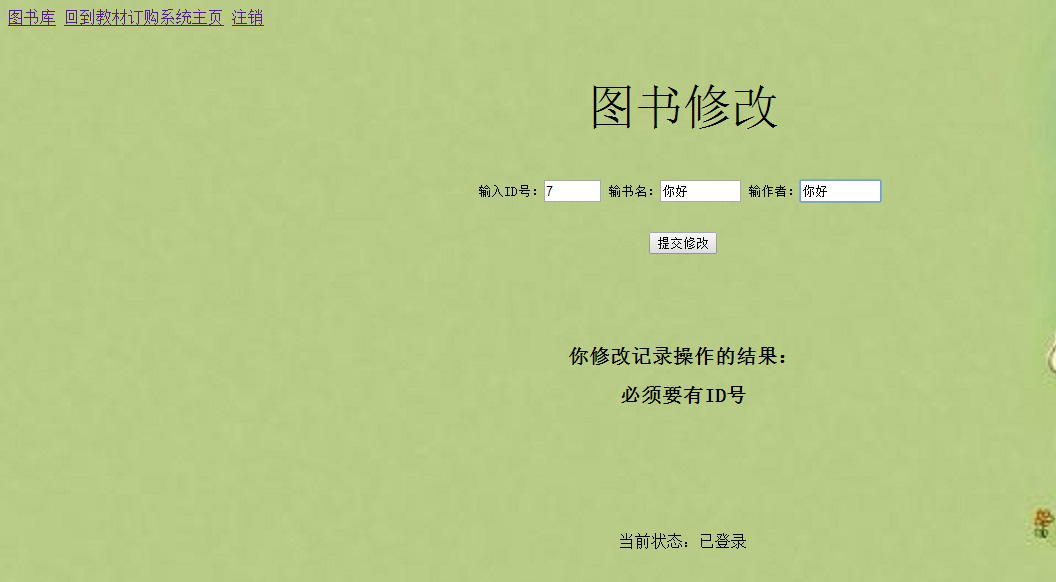
* 添加书籍后的图书库界面



【注销】按钮触发的处理：退回到系统登录主界面。

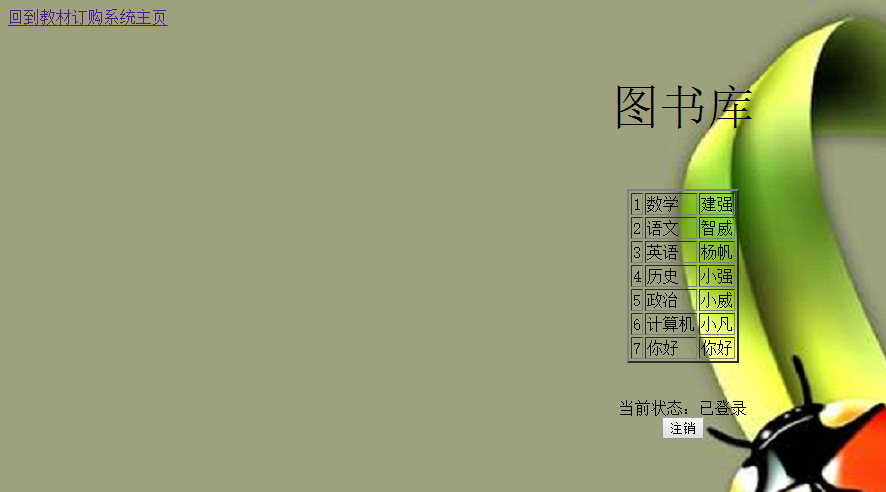
* 图书修改界面

管理人员可在“输入ID号”文本框中填写图书ID号，在“输书名”文本框中填写相应的图书名称，并在“输作者”文本框中填写作者姓名。就可以修改相对应的教材图书信息到书库中并在图书库界面显示。



【提交修改】按钮触发的处理：用于保存新增或修改的数据。

* 修改后的图书库界面



【注销】按钮触发的处理：退回到系统登录主界面。

* 订购/缺书单信息表界面

在本界面管理员可以查看用户提交的订购/缺书单，审核通过即可发送领书单给用户，也可以将缺书单发送给采购人员，通知采购人员尽快补充相应的教材到书库中。操作完毕后可点击【回到教材订购系统主页】或【注销】退出登录。



【注销】按钮触发的处理：退回到系统登录主界面。

1. **编码**

编码阶段的任务是根据详细设计说明书编写程序，程序设计语言的特性和程序设计风格会深刻地影响软件的质量和可维护性。为了保证程序编码的质量，程序员必须深刻理解、熟练掌握并正确地运用程序设计语言的特性。此外，还要求源程序具有良好的结构性和良好的程序设计风格。

* 登录代码：

Login.jsp

<%@page contentType="text/html;charset=gb2312"%>

<HTML>

<HEAD>

<title>欢迎使用教材订购系统</title>

<link rel="stylesheet" type="text/css" href="images/my.css">

</HEAD>

<body >

<div class="login\_logo">

<h2 id=logo1>

欢迎来到教材订购系统

</h2>

</div>

<div class=zhong>

<div id=zuobian></div>

<div id=denglu>

<table width=100% border="0" >

<form action="valid.jsp" method="post" name="form">

<tr> <h2>用户名:<input type="text" name="username"><br/></h2></tr>

<tr><h2>密码:<input type="password" name="password"><br/></h2></tr>

<tr> <h2><input type="submit" value="老师/管理员登陆" style="background-color:pink"><h1></tr>

<h2><input type="reset" value="重置" style="background-color:pink"><h2>

</form>

</table></div>

</div>

<div class=dibu>

</div>

</body>

</HTML>

* 师生主界面代码：

showByPage1.jsp

<%@ page contentType="text/html;charset=GB2312" %>

<%@ page import="java.sql.\*" %>

<%@ page import="xmut.\*" %>

<jsp:useBean id="look" class="xmut.Show" scope="request" />

<jsp:useBean id="login" class="xmut.Login" scope="session" />

<jsp:setProperty name="look" property="databaseName" value="xmut" />

<jsp:setProperty name="look" property="tableName" value="stu" />

<jsp:setProperty name="look" property="user" value="root" />

<jsp:setProperty name="look" property="password" value="root" />

<jsp:setProperty name="look" property="pageSize" value="2" />

<HTML>

<HEAD>

<title>欢迎使用教材订购系统</title>

&nbsp &nbsp <a href=addd.jsp?>订书/缺书单</a>&nbsp &nbsp

<a href=exit.jsp?>注销</a>

<div align="center">

<font size=8> 教材订购系统 </font>

</div></HEAD>

<BODY background="images/a1.jpg"><div align="center">

<br>共有 <jsp:getProperty name="look" property="pageAllCount"/> 页,

每页最多显示<jsp:getProperty name="look" property="pageSize" />条记录。 <BR> <BR>

<jsp:setProperty name= "look" property="showPage" />

<jsp:getProperty name= "look" property="presentPageResult" />

<BR>当前显示第 <jsp:getProperty name= "look" property="showPage" /> 页,

<BR> <BR>

<Table>

<tr>

<td><FORM action="">

<Input type=hidden name="showPage" value="<%=look.getShowPage()-1 %>" >

<Input type=submit name="g" value="上一页">

</FORM>

</td>

<td><FORM action="">

<Input type=hidden name="showPage" value="<%=look.getShowPage()+1 %>" >

<Input type=submit name="g" value="下一页">

</Form>

</td>

<td> <FORM action="">

输入页码：<Input type=text name="showPage" size=5 >

<Input type=submit name="g" value="提交">

</FORM>

</td>

</tr>

</Table>

</div>

<div align="center">

<br><br><br>

当前客户状态：<jsp:getProperty name="login" property="backNews" />

</div>

</BODY></HTML>

* 订书/缺书单填写代码：

Add.jsp

<%@ page contentType="text/html;charset=GB2312" pageEncoding='GB2312'%>

<%@ page import="java.sql.\*" %>

<%@ page import="xmut.\*" %>

<jsp:useBean id="login" class="xmut.Login" scope="session" />

<jsp:useBean id="addd" class="xmut.AddBeann" scope="session" />

<HTML>

<HEAD>

<title>欢迎使用教材订购系统</title>

<a href=exit.jsp?>注销</a>

<a href="http://127.0.0.1:8080/Myjspq/queshu.jsp">订书/缺书单数据记录</a>

&nbsp <a href=showByPage1.jsp?>回到教材订购系统主页</a>

<div align="center">

<font size=8> 订单/缺书填写</font>

</div></HEAD>

<BODY background="images/6.jpg"><Font size=2><div align="center"><br><br><br>

<FORM action="" method='get'>

输入ID号：<Input type="text" name="id" size=4>

输书名：<Input type="text" name="name" size=8>

输作者：<Input type="text" name="writer" size=8>

<br/><br/><br/>

<BR><Input type="submit" value="提交添加">

<jsp:setProperty name= "addd" property="\*" />

<br/><br/><br/>

<br>你添加记录操作的结果：

<jsp:getProperty name= "addd" property="addMessage" />

</div>

</Font><br/><br/><br/>

<div align="center"><br><br>

<form action="exit.jsp" method="post" name="form">

当前状态：<jsp:getProperty name="login" property="backNews" /><br>

</form>

</div>

</BODY></HTML>

* 订书/缺书单数据记录代码：

Queshu.jsp

<%@ page contentType="text/html;charset=GB2312" %>

<%@ page import="java.sql.\*" %>

<jsp:useBean id="login" class="xmut.Login" scope="session" />

<HTML>

<HEAD>

<title>欢迎使用教材订购系统</title>

</HEAD><br><br><br>

<BODY background="images/3.jpg"><div align="center">

<h1>订购/缺书单信息表<h1>

<% Connection con;

Statement sql;

ResultSet rs;

try { Class.forName("com.mysql.jdbc.Driver");

}

catch(Exception e)

{ out.print(e);

}

try { String uri= "jdbc:mysql://localhost/xmut";

con=DriverManager.getConnection(uri,"root","root");

sql=con.createStatement();

rs=sql.executeQuery("SELECT \* FROM stuu");

out.print("<table border=2>");

while(rs.next())

{ out.print("<tr>");

out.print("<td >"+rs.getString(1)+"</td>");

out.print("<td >"+rs.getString(2)+"</td>");

out.print("<td >"+rs.getString(3)+"</td>");

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

}

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

con.close();

}

catch(SQLException e1)

{ out.print(e1);

}

%> </div>

<div align="center"><br><br>

<form action="exit.jsp" method="post" name="form">

当前状态：<jsp:getProperty name="login" property="backNews" /><br><br><br><br>

<input type="submit" value="注销" name="exit">

</form>

</div>

</BODY></HTML>

* 管理员主界面代码：

showByPage.jsp

<%@ page contentType="text/html;charset=GB2312" %>

<%@ page import="java.sql.\*" %>

<%@ page import="xmut.\*" %>

<jsp:useBean id="look" class="xmut.ShowRecordByPage" scope="request" />

<jsp:useBean id="login" class="xmut.Login" scope="session" />

<jsp:setProperty name="look" property="databaseName" value="xmut" />

<jsp:setProperty name="look" property="tableName" value="stu" />

<jsp:setProperty name="look" property="user" value="root" />

<jsp:setProperty name="look" property="password" value="root" />

<jsp:setProperty name="look" property="pageSize" value="2" />

<HTML>

<HEAD>

<title>欢迎使用教材订购系统</title>

<a href=exit.jsp?>注销</a>

<div align="center">

<font size=8> 欢迎进入教材订购系统 </font>

</div></HEAD><br/><br/>

<BODY background="images/7.jpg"><div align="center">

<div>

<a href=add.jsp?>点击添加书籍</a></div><br/>

<div>

<a href=edit.jsp?>点击修改书籍</a>

</div><br/>

<div >

<a href=queshu.jsp?>订购/缺书单信息查询</a>

</div><br/>

<div align="center">

<br>共有 <jsp:getProperty name="look" property="pageAllCount"/> 页.

<br>每页最多显示<jsp:getProperty name="look" property="pageSize" />条记录。

<jsp:setProperty name= "look" property="showPage" />

<jsp:getProperty name= "look" property="presentPageResult" />

<BR>当前显示第 <jsp:getProperty name= "look" property="showPage" /> 页,

<Table>

<tr>

<td><FORM action="">

<Input type=hidden name="showPage" value="<%=look.getShowPage()-1 %>" >

<Input type=submit name="g" value="">

</FORM>

</td>

<td><FORM action="">

<Input type=hidden name="showPage" value="<%=look.getShowPage()-1 %>" >

<Input type=submit name="g" value="上一页">

</FORM>

</td>

<td><FORM action="">

<Input type=hidden name="showPage" value="<%=look.getShowPage()+1 %>" >

<Input type=submit name="g" value="下一页">

</Form>

</td>

<td> <FORM action="">

输入页码：<Input type=text name="showPage" size=5 >

<Input type=submit name="g" value="提交">

</FORM>

</td>

</tr>

</Table>

</div>

<div align="center">

<br><br><br>

<form action="exit.jsp" method="post" name="form">

当前管理员状态：<jsp:getProperty name="login" property="backNews" /><br>

</form>

</div>

</BODY></HTML>

* 书籍添加代码：

add.jsp

<%@ page contentType="text/html;charset=GB2312" pageEncoding='GB2312'%>

<%@ page import="java.sql.\*" %>

<%@ page import="xmut.\*" %>

<jsp:useBean id="login" class="xmut.Login" scope="session" />

<jsp:useBean id="add" class="xmut.AddBean" scope="session" />

<HTML>

<HEAD>

<title>欢迎使用教材订购系统</title>

<a href="http://127.0.0.1:8080/Myjspq/linkMySql.jsp">图书库</a>

<a href=showByPage.jsp?>回到教材订购系统主页</a>

<div align="center"><br><br><br>

<font size=8> 添加书籍 </font>

</div>

</HEAD>

<BODY background="images/6.jpg"><Font size=2><div align="center"><br><br><br>

<FORM action="" method='get'>

输入ID号：<Input type="text" name="id" size=4>

输书名：<Input type="text" name="name" size=8>

输作者：<Input type="text" name="writer" size=8>

<br/><br/><br/>

<BR><Input type="submit" value="提交添加">

<jsp:setProperty name= "add" property="\*" />

<br/><br/><br/>

<br>你添加记录操作的结果：

<jsp:getProperty name= "add" property="addMessage" />

</div>

</Font><br/><br/><br/>

<div align="center"><br><br>

<form action="exit.jsp" method="post" name="form">

当前状态：<jsp:getProperty name="login" property="backNews" /><br>

</form>

</div>

</BODY></HTML>

* 书籍修改代码：

Edit.jsp

<%@ page contentType="text/html;charset=GB2312" pageEncoding='GB2312'%>

<%@ page import="java.sql.\*" %>

<%@ page import="xmut.\*" %>

<jsp:useBean id="login" class="xmut.Login" scope="session" />

<jsp:useBean id="edit" class="xmut.Edit" scope="session" />

<HTML>

<HEAD>

<title>欢迎使用教材订购系统</title>

<a href="http://127.0.0.1:8080/Myjspq/linkMySql.jsp">图书库</a>

<a href=showByPage.jsp?>回到教材订购系统主页</a>

<a href=exit.jsp?>注销</a>

<div align="center"><br><br><br>

<font size=8> 图书修改 </font>

</div></HEAD>

<BODY background="images/6.jpg"><Font size=2><div align="center"><br><br><br>

<FORM action="" method='get'>

输入ID号：<Input type="text" value="<%=request.getParameter("id")%>" name="number" size=4>

输书名：<Input type="text" name="name1" size=8>

输作者：<Input type="text" name="writer" size=8>

<br><br>

<BR><Input type="submit" name="b" value="提交修改"><br><br>

<jsp:setProperty name= "edit" property="number" param="number" />

<jsp:setProperty name= "edit" property="name1" param="name1" />

<jsp:setProperty name= "edit" property="writer" param="writer" />

<br><br><br><br><h2>你修改记录操作的结果：<h2>

<jsp:getProperty name= "edit" property="addMessage" />

</div>

</Font><br><br><br><br>

<div align="center"><br><br>

<form action="exit.jsp" method="post" name="form">

当前状态：<jsp:getProperty name="login" property="backNews" /><br><br><br>

</form>

</div>

</BODY></HTML>

* 查看图书库代码：

linkMySql.jsp

<%@ page contentType="text/html;charset=GB2312" %>

<%@ page import="java.sql.\*" %>

<jsp:useBean id="login" class="xmut.Login" scope="session" />

<HTML>

<HEAD>

<title>欢迎使用教材订购系统</title>

<a href=showByPage.jsp?>回到教材订购系统主页</a>

<div align="center"><br><br><br>

<font size=8> 图书库 </font>

</div></HEAD><br><br><br>

<BODY background="images/3.jpg"><div align="center">

<% Connection con;

Statement sql;

ResultSet rs;

try { Class.forName("com.mysql.jdbc.Driver");

}

catch(Exception e)

{ out.print(e);

}

try { String uri= "jdbc:mysql://localhost/xmut";

con=DriverManager.getConnection(uri,"root","root");

sql=con.createStatement();

rs=sql.executeQuery("SELECT \* FROM stu");

out.print("<table border=2>");

while(rs.next())

{ out.print("<tr>");

out.print("<td >"+rs.getString(1)+"</td>");

out.print("<td >"+rs.getString(2)+"</td>");

out.print("<td >"+rs.getString(3)+"</td>");

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

}

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

con.close();

}

catch(SQLException e1)

{ out.print(e1);

}

%> </div>

<div align="center"><br><br>

<form action="exit.jsp" method="post" name="form">

当前状态：<jsp:getProperty name="login" property="backNews" /><br>

<input type="submit" value="注销" name="exit">

</form>

</div>

</BODY></HTML>

1. **总结**

经过两周的课程设计，我们虽然遇到了些困难，但经过我们不懈努力，坚持不懈的精神，我们解决了困难。我们分工很明确，但同组员有不懂的问题，我们互相帮助思考，互相提供解决思路。最大的收获是我们通过这次课程设计学到了课堂上学不到的东西。这是最有价值的。然而这次的课程设计我们或许做的不完美，但我们已经尽了最大的努力了。知识是一点一点增长的，以后我们会做的更好！

**附录**

.java文件

AddBean.java

package xmut;

import java.sql.\*;

public class AddBean

{ String id="", //号

name="", //姓名

writer="",

addMessage="";

public AddBean()

{ try{ Class.forName("com.mysql.jdbc.Driver");

}

catch(Exception e){ }

}

public void setId(String s)

{

try{id=s.trim();}

catch(Exception e){}

}

public void setName(String s)

{ name=s.trim();

/\*

try{ byte bb[]=name.getBytes("ISO-8859-1");

name=new String(bb,"utf-8");

}

catch(Exception e){}

\*/

}

public void setWriter(String s)

{ writer=s.trim();

/\*

try{ byte bb[]=writer.getBytes("ISO-8859-1");

writer=new String(bb,"utf-8");

}

catch(Exception e){}

\*/

}

public String getAddMessage()

{

String insertCondition="INSERT into stu VALUES('"+id+"', '"+name+"','"+writer+"')";

String str="";

Connection con;

Statement sql;

try {

String uri= "jdbc:mysql://localhost/xmut?useUnicode=true&characterEncoding=GB2312";

con=DriverManager.getConnection(uri,"root","root");

sql=con.createStatement();

if(id.length()>0){

int m=sql.executeUpdate(insertCondition);

if(m!=0){

str="对表中添加"+m+"条记录成功"+writer+name;

}else{

str="添加记录失败";

}

}else{

str="必须要有ID号";

}

con.close();

}

catch(SQLException e) {

str="输入的ID不允许重复"+e;

}

return str;

}

}

AddBeann.java

package xmut;

import java.sql.\*;

public class AddBeann

{ String id="", //号

name="", //姓名

writer="",

addMessage="";

public AddBeann()

{ try{ Class.forName("com.mysql.jdbc.Driver");

}

catch(Exception e){ }

}

public void setId(String s)

{ id=s.trim();

}

public void setName(String s)

{ name=s.trim();

/\*

try{ byte bb[]=name.getBytes("ISO-8859-1");

name=new String(bb,"utf-8");

}

catch(Exception e){}

\*/

}

public void setWriter(String s)

{ writer=s.trim();

/\*

try{ byte bb[]=writer.getBytes("ISO-8859-1");

writer=new String(bb,"utf-8");

}

catch(Exception e){}

\*/

}

public String getAddMessage()

{

String insertCondition="INSERT into stuu VALUES('"+id+"', '"+name+"','"+writer+"')";

String str="";

Connection con;

Statement sql;

try {

String uri= "jdbc:mysql://localhost/xmut?useUnicode=true&characterEncoding=GB2312";

con=DriverManager.getConnection(uri,"root","root");

sql=con.createStatement();

if(id.length()>0){

int m=sql.executeUpdate(insertCondition);

if(m!=0){

str="对表中添加"+m+"条记录成功"+writer+name;

}else{

str="添加记录失败";

}

}else{

str="必须要有ID号";

}

con.close();

}

catch(SQLException e) {

str="输入的ID不允许重复"+e;

}

return str;

}

}

Edit.java

package xmut;

import java.sql.\*;

public class Edit

{ String number="", //号

name1="", //姓名

writer="",

addMessage="";

public Edit()

{ try{ Class.forName("com.mysql.jdbc.Driver");

}

catch(Exception e){ }

}

public void setNumber(String s)

{ number=s.trim();

}

public void setName1(String s)

{ name1=s.trim();

/\*

try{ byte bb[]=name1.getBytes("ISO-8859-1");

name1=new String(bb,"gb2312");

}

catch(Exception e){}

\*/

}

public void setWriter(String s)

{ writer=s.trim();

/\*

try{ byte bb[]=name.getBytes("ISO-8859-1");

name=new String(bb,"gb2312");

}

catch(Exception e){}

\*/

}

public String getAddMessage()

{ String insertCondition="UPDATE stu SET name = '"+name1+"' , writer = '"+writer+"' WHERE id='"+number+"'"; //表单名字

String str="";

Connection con;

Statement sql;

try { String uri= "jdbc:mysql://localhost/xmut?useUnicode=true&characterEncoding=GB2312"; //数据库名字

con=DriverManager.getConnection(uri,"root","root");

sql=con.createStatement();

if(number.length()>0)

{ int m=sql.executeUpdate(insertCondition);

if(m!=0)

{ str="对表中修改"+m+"条记录成功"+writer+name1;

}

else

{ str="修改记录失败";

}

}

else

{ str="必须要有ID号";

}

con.close();

}

catch(SQLException e)

{ str="输入的ID不允许重复"+e;

}

return str;

}

}

ShowRecordByPage.java

package xmut;

import java.sql.\*;

import com.sun.rowset.\*;

public class ShowRecordByPage

{ int pageSize=10; //每页显示的记录数

int pageAllCount=0; //分页后的总页数

int showPage=1 ; //当前显示页

StringBuffer presentPageResult; //显示当前页内容

CachedRowSetImpl rowSet; //用于存储ResultSet对象

String databaseName=""; //数据库名称

String tableName=""; //表的名字

String user="" ; //用户

String password="" ; //密码

String word[]=new String[100] ;

int len=0;

public ShowRecordByPage()

{ presentPageResult=new StringBuffer();

try{ Class.forName("com.mysql.jdbc.Driver").newInstance();

}

catch(Exception e){}

}

public void setPageSize(int size)

{ pageSize=size;

len=0;

String uri="jdbc:mysql://localhost/"+databaseName;

try{ Connection con=DriverManager.getConnection(uri,user,password);

DatabaseMetaData metadata=con.getMetaData();

ResultSet rs1=metadata.getColumns(null,null,tableName,null);

int k=0;

while(rs1.next())

{ len++;

word[k]=rs1.getString(4); //获取word的名字

k++;

}

Statement sql=con.createStatement(ResultSet.TYPE\_SCROLL\_SENSITIVE,ResultSet.CONCUR\_READ\_ONLY);

ResultSet rs=sql.executeQuery("SELECT \* FROM "+tableName);

rowSet=new CachedRowSetImpl(); //创建行集对象

rowSet.populate(rs);

con.close(); //关闭连接

rowSet.last();

int m=rowSet.getRow(); //总行数

int n=pageSize;

pageAllCount=((m%n)==0)?(m/n):(m/n+1);

}

catch(Exception exp){}

}

public int getPageSize()

{ return pageSize;

}

public int getPageAllCount()

{ return pageAllCount;

}

public void setShowPage(int n)

{ showPage=n;

}

public int getShowPage()

{ return showPage;

}

public StringBuffer getPresentPageResult()

{ if(showPage>pageAllCount)

showPage=1;

if(showPage<=0)

showPage=pageAllCount;

presentPageResult=show(showPage);

return presentPageResult;

}

public StringBuffer show(int page)

{ StringBuffer str=new StringBuffer();

str.append("<table border=1>");

str.append("<tr>");

for(int i=0;i<len;i++)

{ str.append("<th>"+word[i]+"</th>");

}

str.append("<th>编辑</th>");

str.append("<th>删除</th>");

str.append("</tr>");

try{ rowSet.absolute((page-1)\*pageSize+1);

for(int i=1;i<=pageSize;i++)

{ str.append("<tr>");

String delete="<td><a href='delete.jsp?number="+rowSet.getString(1)+"'>删除</a></td>";

String edit="<td><a href='edit.jsp?id="+rowSet.getString(1)+"'>编辑</a></td>";

for(int k=1;k<=len;k++)

{

str.append("<td>"+rowSet.getString(k)+"</td>");

}

str.append(edit);

str.append(delete);

str.append("</tr>");

rowSet.next();

}

}

catch(SQLException exp){

str.append(exp);

}

str.append("</table>");

return str;

}

public void setDatabaseName(String s)

{ databaseName=s.trim();

}

public String getDatabaseName()

{ return databaseName;

}

public void setTableName(String s)

{ tableName=s.trim();

}

public String getTableName()

{ return tableName;

}

public void setPassword(String s)

{ password=s.trim();;

}

public void setUser(String s)

{ user=s.trim();

}

public String getUser()

{ return user;

}

}

DeleteBean.java

package xmut;

import java.sql.\*;

public class DeleteBean

{ String id="", //号

name="", //姓名

bookname="",

addMessage="";

public DeleteBean()

{ try{ Class.forName("com.mysql.jdbc.Driver");

}

catch(Exception e){ }

}

public void setId(String s)

{ id=s.trim();

}

public void setName(String s)

{ name=s.trim();

try{ byte bb[]=name.getBytes("ISO-8859-1");

name=new String(bb,"gb2312");

}

catch(Exception e){}

}

public void setBookName(String s)

{ bookname=s.trim();

try{ byte bb[]=bookname.getBytes("ISO-8859-1");

bookname=new String(bb,"gb2312");

}

catch(Exception e){}

}

public String getAddMessage()

{ String insertCondition="DELETE FORM stu WHERE number='id'"; //表单名字

String str="";

Connection con;

Statement sql;

try { String uri= "jdbc:mysql://localhost/xmut"; //数据库名字

con=DriverManager.getConnection(uri,"root","root");

sql=con.createStatement();

if(id.length()>0)

{ int m=sql.executeUpdate(insertCondition);

if(m!=0)

{ str="删除"+m+"条记录成功";

}

else

{ str="删除记录失败";

}

}

else

{ str="该ID号不存在";

}

con.close();

}

catch(SQLException e)

{ str="该ID号不存在"+e;

}

return str;

}

}

DeleteBeann.java

package xmut;

import java.sql.\*;

public class DeleteBean

{ String id="", //号

name="", //姓名

bookname="",

addMessage="";

public DeleteBean()

{ try{ Class.forName("com.mysql.jdbc.Driver");

}

catch(Exception e){ }

}

public void setId(String s)

{ id=s.trim();

}

public void setName(String s)

{ name=s.trim();

try{ byte bb[]=name.getBytes("ISO-8859-1");

name=new String(bb,"gb2312");

}

catch(Exception e){}

}

public void setBookName(String s)

{ bookname=s.trim();

try{ byte bb[]=bookname.getBytes("ISO-8859-1");

bookname=new String(bb,"gb2312");

}

catch(Exception e){}

}

public String getAddMessage()

{ String insertCondition="DELETE FORM stu WHERE number='id'"; //表单名字

String str="";

Connection con;

Statement sql;

try { String uri= "jdbc:mysql://localhost/xmut"; //数据库名字

con=DriverManager.getConnection(uri,"root","root");

sql=con.createStatement();

if(id.length()>0)

{ int m=sql.executeUpdate(insertCondition);

if(m!=0)

{ str="删除"+m+"条记录成功";

}

else

{ str="删除记录失败";

}

}

else

{ str="该ID号不存在";

}

con.close();

}

catch(SQLException e)

{ str="该ID号不存在"+e;

}

return str;

}

}

Login.java

package xmut;

import java.util.\*;

public class Login

{ String logname,

password,

backNews="未登录";

boolean success=false;

public void setBackNews(String s)

{ backNews=s;

}

public void setSuccess(boolean b)

{ success=b;

if(success){

backNews="已登录";

}else{

backNews="未登录";

}

}

public boolean getSuccess()

{ return success;

}

}

Show.java

package xmut;

import java.sql.\*;

import com.sun.rowset.\*;

public class Show

{ int pageSize=10; //每页显示的记录数

int pageAllCount=0; //分页后的总页数

int showPage=1 ; //当前显示页

StringBuffer presentPageResult; //显示当前页内容

CachedRowSetImpl rowSet; //用于存储ResultSet对象

String databaseName=""; //数据库名称

String tableName=""; //表的名字

String user="" ; //用户

String password="" ; //密码

String 字段[]=new String[100] ;

int 字段个数=0;

public Show()

{ presentPageResult=new StringBuffer();

try{ Class.forName("com.mysql.jdbc.Driver").newInstance();

}

catch(Exception e){}

}

public void setPageSize(int size)

{ pageSize=size;

字段个数=0;

String uri="jdbc:mysql://localhost/"+databaseName;

try{ Connection con=DriverManager.getConnection(uri,user,password);

DatabaseMetaData metadata=con.getMetaData();

ResultSet rs1=metadata.getColumns(null,null,tableName,null);

int k=0;

while(rs1.next())

{ 字段个数++;

字段[k]=rs1.getString(4); //获取字段的名字

k++;

}

Statement sql=con.createStatement(ResultSet.TYPE\_SCROLL\_SENSITIVE,

ResultSet.CONCUR\_READ\_ONLY);

ResultSet rs=sql.executeQuery("SELECT \* FROM "+tableName);

rowSet=new CachedRowSetImpl(); //创建行集对象

rowSet.populate(rs);

con.close(); //关闭连接

rowSet.last();

int m=rowSet.getRow(); //总行数

int n=pageSize;

pageAllCount=((m%n)==0)?(m/n):(m/n+1);

}

catch(Exception exp){}

}

public int getPageSize()

{ return pageSize;

}

public int getPageAllCount()

{ return pageAllCount;

}

public void setShowPage(int n)

{ showPage=n;

}

public int getShowPage()

{ return showPage;

}

public StringBuffer getPresentPageResult()

{ if(showPage>pageAllCount)

showPage=1;

if(showPage<=0)

showPage=pageAllCount;

presentPageResult=show(showPage);

return presentPageResult;

}

public StringBuffer show(int page)

{ StringBuffer str=new StringBuffer();

str.append("<table border=1>");

str.append("<tr>");

for(int i=0;i<字段个数;i++)

{ str.append("<th>"+字段[i]+"</th>");

}

str.append("</tr>");

try{ rowSet.absolute((page-1)\*pageSize+1);

for(int i=1;i<=pageSize;i++)

{ str.append("<tr>");

for(int k=1;k<=字段个数;k++)

{ str.append("<td>"+rowSet.getString(k)+"</td>");

}

str.append("</tr>");

rowSet.next();

}

}

catch(SQLException exp){}

str.append("</table>");

return str;

}

public void setDatabaseName(String s)

{ databaseName=s.trim();

}

public String getDatabaseName()

{ return databaseName;

}

public void setTableName(String s)

{ tableName=s.trim();

}

public String getTableName()

{ return tableName;

}

public void setPassword(String s)

{ password=s.trim();;

}

public void setUser(String s)

{ user=s.trim();

}

public String getUser()

{ return user;

}

}