**Java Web开发**

**实验八**

**作业报告**

|  |  |
| --- | --- |
| 学 院 | 计算机学院 |
| 所在系 班 级 | 2017软工7班 |
| 学 号 姓 名 | 04171520童煜东 |
| 指 导 教 师 | 郭晓燕 |
| 完 成 时 间 | 2019年5月2 |

# 实验题目

MVC模式下的登陆

# 设计与分析

MVC：1、servlet（业务层）

2、DAO（数据访问层）

3、jsp（视图层）

# 关键源码（部分）

**数据访问层**

public User findUser(User user) throws Exception{

Connection conn = null;

PreparedStatement =null;

ResultSet rs =null;

User u =null;

try {

conn=DBUtils.getConnection();

ps = conn.prepareStatement(“select \*from user where username=? and passWd=?”);

ps.setString(1,user.getUserName());

ps.set String(1,user.getpassWd());

rs =ps.executeQuery();

if(rs.next()){

u = new User();

u.setUserId(rs.getInt(1)) ;

u.setUserName(rs.getString(2)) ;

u.setpassWd(rs.getString(3))

u.setSex(rs.getString(4)) ;

u.setBirthday(rs.getDate(5)) ;

}

}catch(Exception e){

e.printStackTrace();}

finally{

DBUtils.closeAll(rs,ps,conn);}

Return u;

}

**Service**

public User login(User user){

User u=null;

try{

u = user.Dao.findUser(user);}

catch(Exception e){

e.printStackTrace();}

return u;

}

**Jsp登陆**

<form action=*"${pageContext.request.contextPath }/servlet/loginServlet"* method=*"post"*>

用户名：<input type=*"text"* name=*"username"* id=*"username"*/><br>

密　码：<input type=*"password"* name=*"pwd"* /><br>

<input type=*"submit"* value=*"登录"* /> </form>

**LoginServlet.java**

**protected** **void** doPost(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {

// **TODO** Auto-generated method stub

response.setContentType("UTF-8");

User user =new User();

try{

BeanUtils.populate(user,request.getParameterMap());

UserService us = new UserServiceImpl();

User u = us.login(user);

If(u!=null){

request.getSession().setAttribute(*"u",user*);

request.getRequestDispatcher(*"/index.jsp"*).forward(request,response);

}else{

Request.sendRedirect(request.getContextPath()+*"/login.jsp"*);

}

}catch(Exception e)

{e.printStackTrace();

}

}

1. 运行贴图





1. **收获与问题**

还是不是很懂MVC，还得继续努力啊。

# 一、实验题目

MVC模式的购物车

# 分析与设计

MVC：a. servlet（业务层）

      b. DAO（数据访问层）

 c. JSp（表示层）

# 关键源码（部分）

**Jsp首页**

<html>

<head>

<title>网上商城</title>

<meta http-equiv=*"pragma"* content=*"no-cache"*>

<meta http-equiv=*"cache-control"* content=*"no-cache"*>

<meta http-equiv=*"expires"* content=*"0"*>

<meta http-equiv=*"keywords"* content=*"keyword1,keyword2,keyword3"*>

<meta http-equiv=*"description"* content=*"This is my page"*>

<!--

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

-->

</head>

<body>

<hr/>

<h2 >欢迎进入网上购物商城</h2>

<hr>

<a href=*"*${pageContext.request.contextPath}*/ListBookServlet"*>进入购物页</a> <br>

</body>

</html>

**javabean**

**public** **class** Book

{

**private** String id;

**private** String name;//书名

**private** String author;//作者

**private** **int** price;

**public** Book()

{

**super**();

// **TODO** Auto-generated constructor stub

}

**public** Book(String id, String name, String author, **int** price)

{

**super**();

**this**.id = id;

**this**.name = name;

**this**.author = author;

**this**.price = price;

}

**public** String getId()

{

**return** id;

}

**public** **void** setId(String id)

{

**this**.id = id;

}

**public** String getName()

{

**return** name;

}

**public** **void** setName(String name)

{

**this**.name = name;

}

**public** String getAuthor()

{

**return** author;

}

**public** **void** setAuthor(String author)

{

**this**.author = author;

}

**public** **int** getPrice()

{

**return** price;

}

**public** **void** setPrice(**int** price)

{

**this**.price = price;

## 用Map集合来模拟数据库

**private** **static** Map<String , Book> *books*=**new** LinkedHashMap<String, Book>();

**static**{

*books*.put("1", **new** Book("1", "《水浒传》", "施耐庵", 48));

*books*.put("2", **new** Book("2", "《西游记》", "吴承恩 ", 58));

*books*.put("3", **new** Book("3", "《三国演义》", "罗贯中", 78));

*books*.put("4", **new** Book("4", "《红楼梦》", "曹雪芹", 28));

*books*.put("5", **new** Book("5", "《平凡的世界》", "路遥", 18));

}

**public** **static** Collection<Book> getAll() {

**return** *books*.values();

}

**public** **static** Book find(String id) {

**return** *books*.get(id);

}

**Javabean的购物车对象**

package cn.itcast.cart.domain;

public class ShoppingcartItem

{

//购物车项，每一本书买了多少本，总共多少钱

private Book book;

private int quantity;

private int price;

public Book getBook()

{

return book;

}

public void setBook(Book book)

{

this.book = book;

}

public int getQuantity()

{

return quantity;

}

public void setQuantity(int quantity)

{

this.quantity = quantity;

this.price=this.book.getPrice()\*quantity;

}

public int getPrice()

{

return price;

}

public void setPrice(int price)

{

this.price = price;

}

}

**购物车**

public class Shoppingcart

{

private Map<String, ShoppingcartItem> items=new HashMap<String, ShoppingcartItem>();

private int price;//总价

public Map<String, ShoppingcartItem> getItems()

{

return items;

}

public void setItems(Map<String, ShoppingcartItem> items)

{

this.items = items;

}

public int getPrice()

{

//计算总价

int price=0;

for(ShoppingcartItem item:items.values())

{

price+=item.getPrice();

}

return price;

}

public void setPrice(int price)

{

this.price = price;

}

}

**获取图书**

ublic class ListBookServlet extends HttpServlet

{

//从DB中查询所有的图书

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException

{

//查询所有的商品

Collection<Book> books = DB.getAll();

// 转发给jsp显示

request.setAttribute("books", books);

request.getRequestDispatcher("/WEB-INF/pages/listbook.jsp").forward(request, response);

}

public void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException

{

doGet(request, response);

}

}

**展示商品jsp**

</head>

<body style="text-align: center">

<h1 >商品列表</h1>

<table border="1" width="400px">

<tr>

<td>图书名称</td>

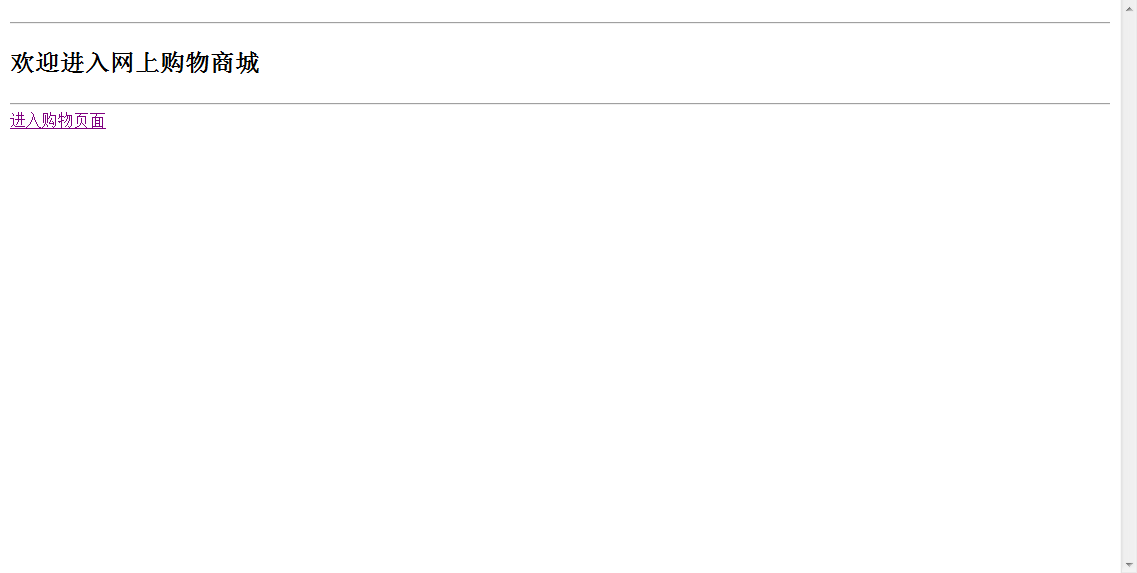
<td>作者</td>

<td>价格</td>

<td>购买</td>

</tr>

1. **运行贴图**





**五、收获与问题**

继续努力