**Java Web开发**

**实验五**

**作业报告**

|  |  |
| --- | --- |
| 学 院 | 计算机 |
| 所在系 班 级 | 2017软件工程系9班 |
| 学 号 姓 名 | 10171421钟质昌 |
| 指 导 教 师 | 罗永升 |
| 完 成 时 间 | 2019年4月- |

**实验五 会话管理-购物车**

**（此实验四个课时）**

实验目标：

1、理解会话的原理和HttpSession会话期；

2、掌握如何使用URL重写；

3、掌握如何使用HttpSession会话管理

实验内容：

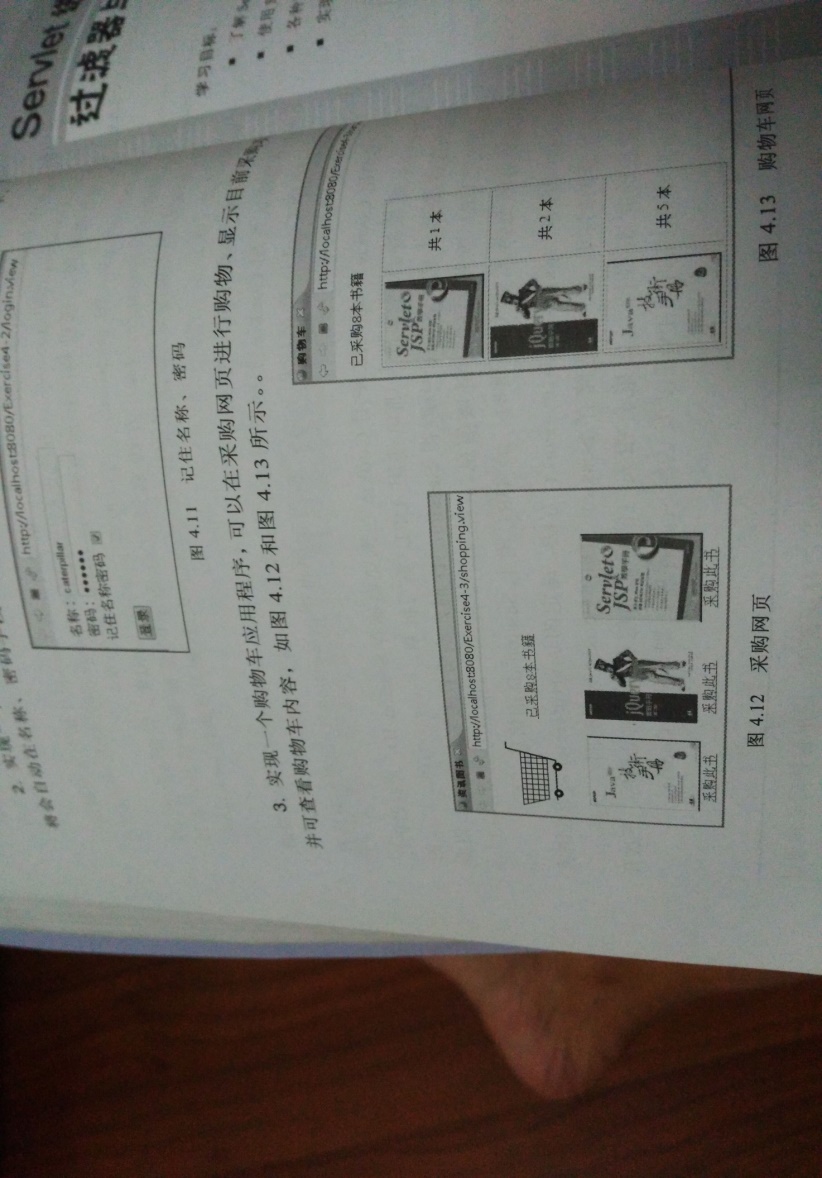
1. 练习老师在理论课上演示的购物车实例，理解购物功能如何实现，理解如何利用session对象保存购物信息，如何设计封装数据类。（请参考“第四章会话管理课上的购物车实例代码.docx”文档）；

如果上课听懂了购物车的原理或者部分理解，可以参考我的实例完成第2题，此题可忽略不写。

1. 第四章课后练习3购物车，实现一个购物车应用程序，可以购物，显示购物数量并查看购物车内容。（此题模版test4E，提供给大家参考。）

提示：采购此书做成超链接，并用URL重写带关键信息

* 1. Shopping.view?bookID=‘0001’，其中0001是你选中图书对应的ID号，每点击一次就在相应的购书数量上加一，并返回当前的选购页面。
  2. 除了要保存图书的ID号、书名，还要保存购书数量，所以session对象保存的信息需要考虑清楚，最好是一个对象数组或列表。



购物车的效果图

1. 对第三章动态验证码程序实现动态验证。（选做）

提示：在生成动态验证码的同时将其放入HttpSession对象中，在网页上提取出来进行比对。

实验报告：

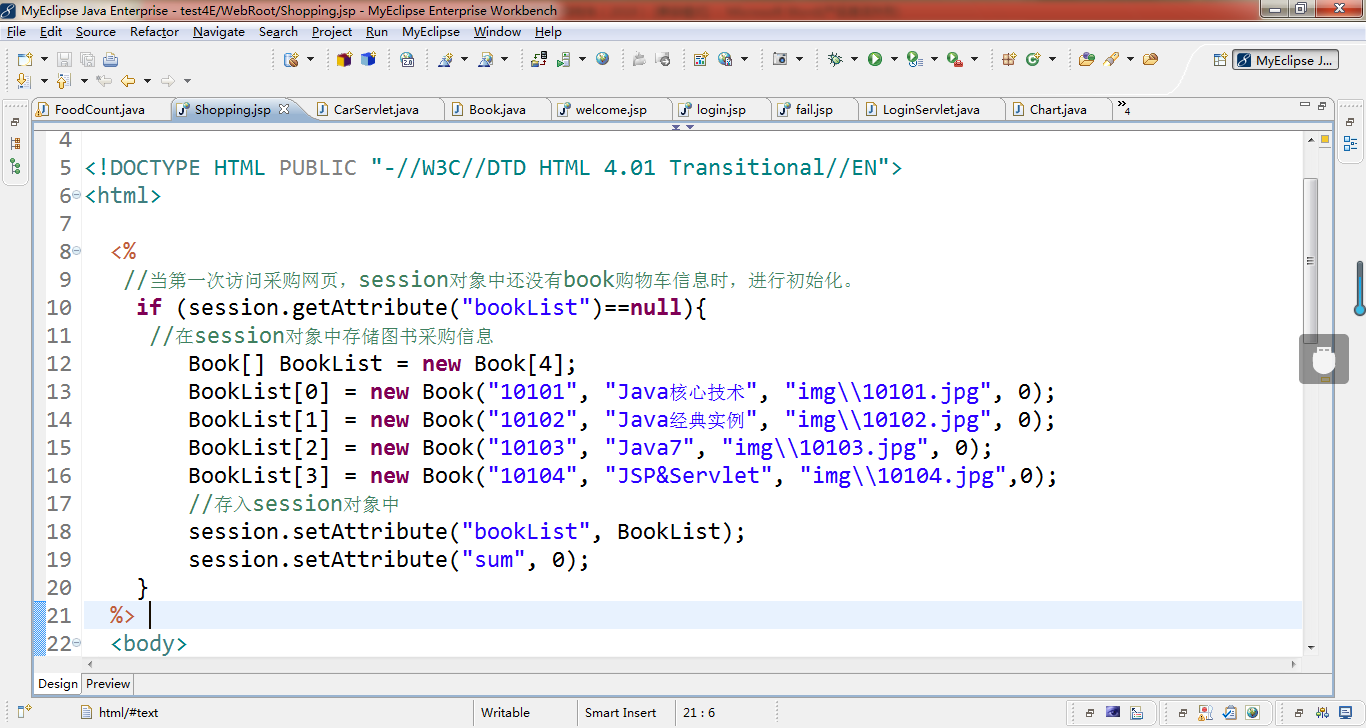
提交相关程序代码并回答问题。

**附：Test4E涉及的文件内容及功能作用**

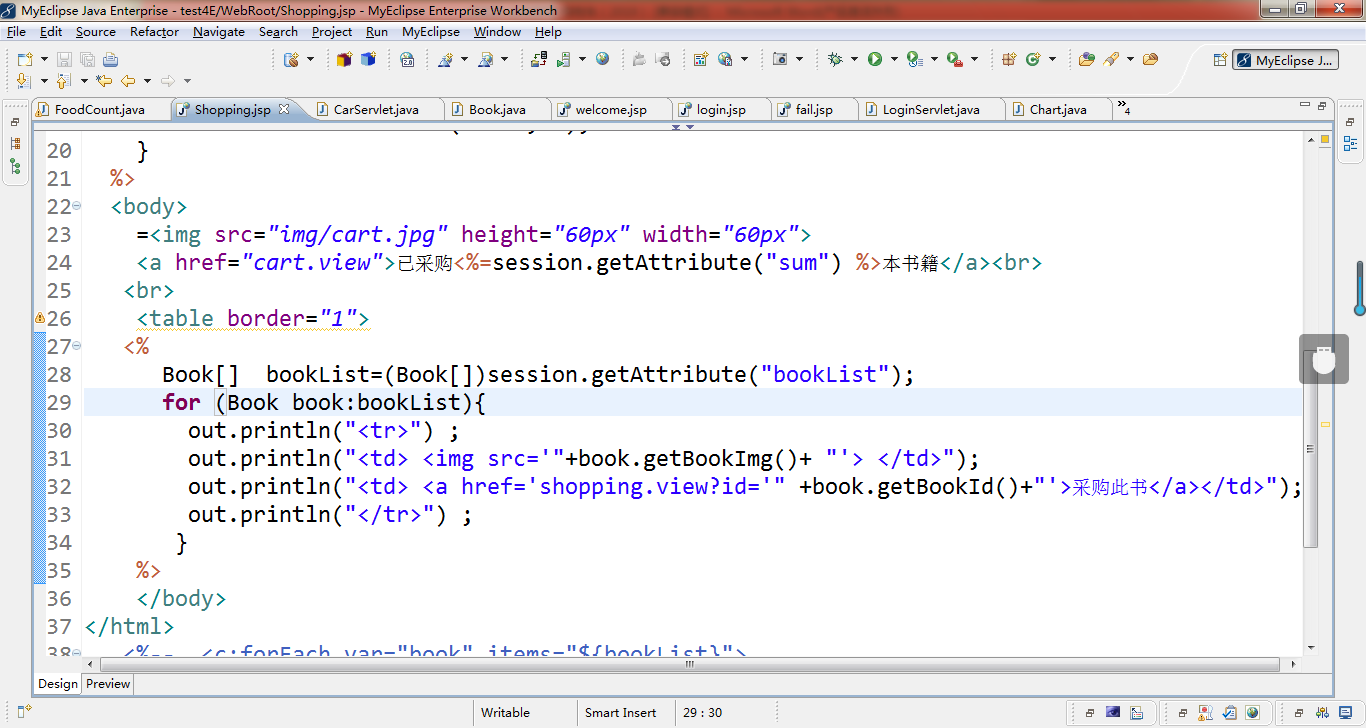
Book.java:数据类，用来保存图书的基本信息（ID、书名、价格）和购书信息（购书数量）。

Shopping.jsp:购书页面，通过此页面用户选购图书。即效果图片的左图。

1、初始化图书信息，将所有提供给用户选购的图书信息、购书信息初始化后保存在一个图书对象数组中，再将此内容保存在session对象中（注：session对象在JSP页面中可以无需定义直接使用，与在Servlet中获取的session对象是同一个）



2、列出所有商品，提供给用户选择购买。请注意此处URL重写的应用，另外读取商品信息也是借助循环自动生成，而不是写死。



ShoppingServlet.java（访问URL地址为shopping.view）

基本思路：是获取到用户所点击的ID号（即书号），同时从Session对象中将图书数组取出来。在图书数组中进行遍历，找到和用户点击ID号相同的那本图书，将其购买数量+1，同时将总本数+1，最后将更新好的图书数组和总本数放回session对象中，对应的key不变。

CarServlet.java(访问URL地址为cart.view)

基本思路：从Session对象中取出图书数组，对其进行遍历，凡图书购买本数大于0的图书，将其基本信息显示在页面上。

# 关键源码（部分）

import com.Book;

@WebServlet("/shopping.view")

public class ShoppingServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

//Map<Integer,String> books = new HashMap<Integer,String>();

//books.put(1,book0);

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

int sum=(Integer) request.getSession().getAttribute("sum");

Book[] BookList = (Book[])request.getSession().getAttribute("BookList");

for(Book book:BookList){

int count = 0;

if(id.equals(book.getId())){

sum=sum+1;

count =book.getCount()+1;

book.setCount(count);

}

}

request.getSession().setAttribute("sum",sum);

request.getSession().setAttribute("BookList", BookList);

request.getRequestDispatcher("Shopping.jsp").forward(request, response);

}

}

<%@ page import= *"com.Book"* import=*'java.io.PrintWriter'* %>

<%@ page language=*"java"* contentType=*"text/html; charset=UTF-8"*

pageEncoding=*"UTF-8"*%>

<%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<%

**if**(session.getAttribute("BookList")==**null**){

Book[] BookList = **new** Book[3];

BookList[0] = **new** Book("01","思修","//img//sixiu.jpg",17.5,0);

BookList[1] = **new** Book("02","毛概","//img//maogai.jpg",16.0,0);

BookList[2] = **new** Book("03","高数","//img//gaoshu.jpg",15.0,0);

session.setAttribute("BookList",BookList);

session.setAttribute("sum",0);

}

%>

<body>

<img src=*"*${pageContext.request.contextPath}*//img//cart.jpg"*>

<a href=*'cart.jsp'*>已采购<%=session.getAttribute("sum")%>本</a>

<br>

<table border=*'1'*>

<c:forEach var=*"book"* items=*"*${BookList}*"*>

<tr>

<td><img src =*"*${pageContext.request.contextPath}*//*${book.getImg()}*"*></td>

<td><a href=*'shopping.view?id=*${book.getId()}*'*>采购此书</a></td>

</tr>

</c:forEach>

</table>

</body>

</html>

<%@ page import= *"com.Book"* import=*'java.io.PrintWriter'* %>

<%@ page language=*"java"* contentType=*"text/html; charset=UTF-8"*

pageEncoding=*"UTF-8"*%>

<%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<body>

<img src=*"*${pageContext.request.contextPath}*//img//cart.jpg"*>

<a href=*'Shopping.jsp'*>已采购<%=session.getAttribute("sum")%>本</a>

<br>

<table border=*'1'*>

<c:forEach var=*"book"* items=*"*${BookList}*"*>

<tr>

<td><img src =*"*${pageContext.request.contextPath}*//*${book.getImg()}*"*></td>

<td>共采购${book.getCount()}本</td>

</tr>

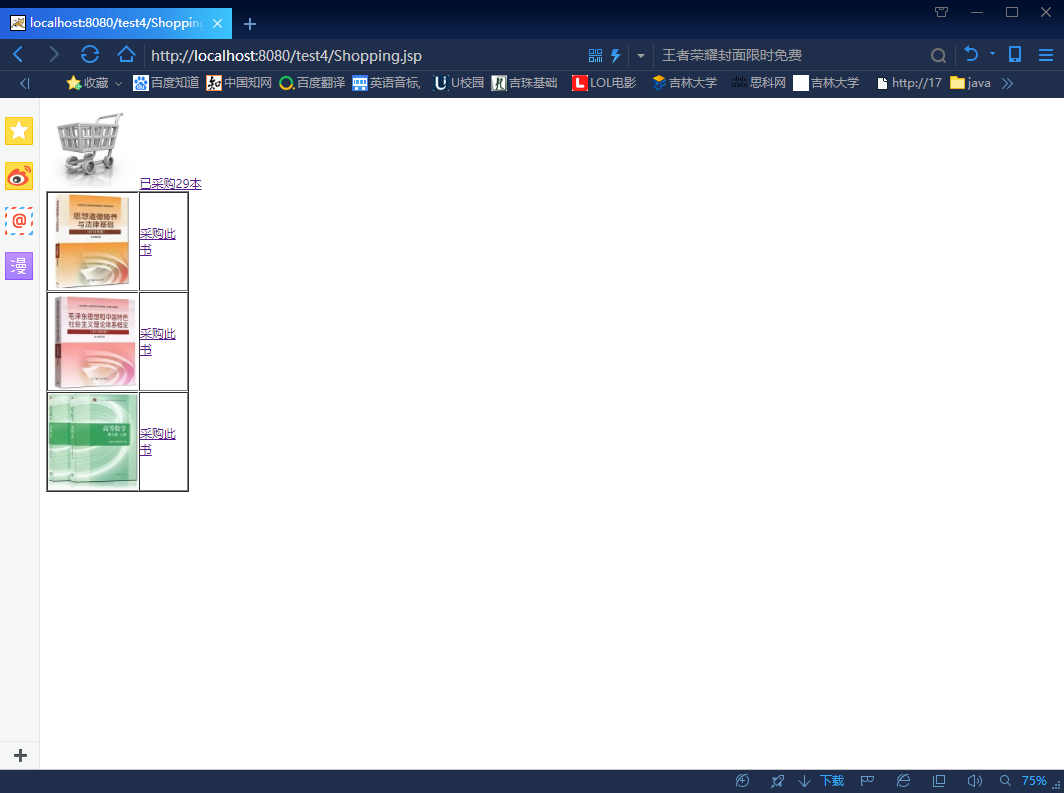
</c:forEach>

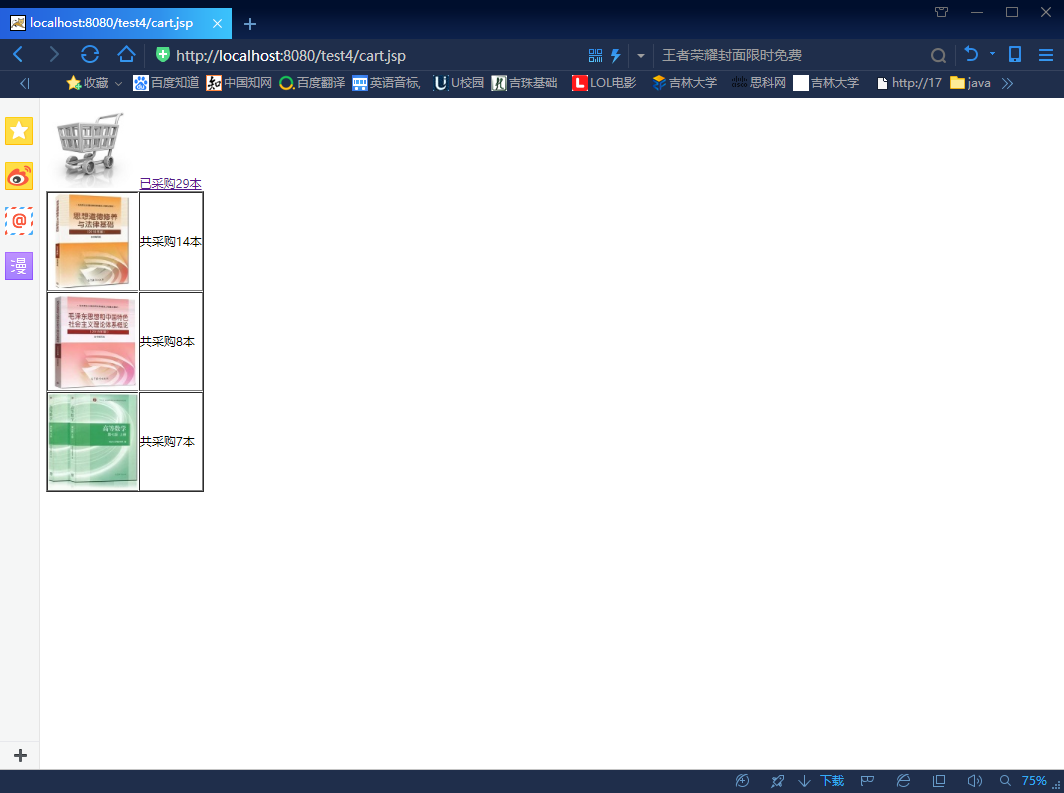
</table>

</body>

</html>

# 运行效果图





# 收获与问题

收获很多学会了session的基本使用方法