**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10班 |
| 学 号 姓 名 | 06171522吴小杰 |
| 指 导 教 师 | 罗永升 |
| 完 成 时 间 | 2019年4月23日 |

# 实验题目

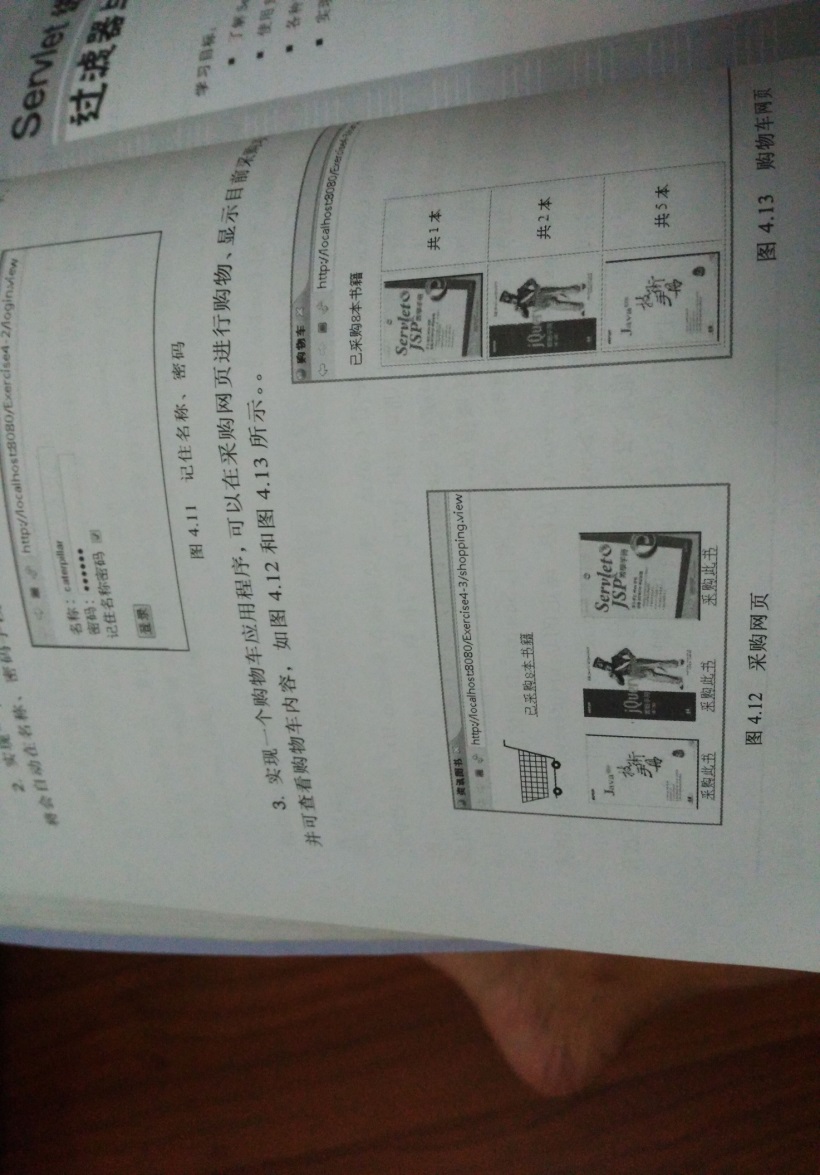
1. 练习老师在理论课上演示的购物车实例，理解购物功能如何实现，理解如何利用session对象保存购物信息，如何设计封装数据类。（请参考“第四章会话管理课上的购物车实例代码.docx”文档）；

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。

# 分析与设计

采用MVC2模式，Shopping.jsp, CarServlet.java, InitServlet.java, ShopingServlet.java,Book.java

# 关键源码（部分）

**Shopping.jsp:**

<%@ page language=*"java"* import=*"java.util.\*"* pageEncoding=*"UTF-8"*%>

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

<%@page import=*"cc.test.vo.\*"* %>

<!--

程序运行的首页是Example\_4/index.do

-->

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html>

<body>

<img src=*"img/cart.jpg"* height=*"60px"* width=*"60px"*>

<a href=*"cart.view"*>已采购<%=session.getAttribute("sumNum") %>本书籍</a><br>

<br>

<table border=*"1"*>

<%

Book[] bookList=(Book[])session.getAttribute("bookList");

**for** (Book book:bookList){

out.println("<tr>") ;

out.println("<td> <img src='"+book.getBookImg()+ "'> </td>");

String id=book.getBookId();

out.println("<td> <a href=shopping.view?id="+id+">采购此书</a></td>");

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

}

%>

</table></body>

</html>

**CarServlet.java:**

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

import cc.test.vo.Book;

public class CarServlet extends HttpServlet {

public void doGet(HttpServletRequest request,HttpServletResponse response)

throws ServletException, IOException {

HttpSession session=request.getSession();

response.setContentType("text/html;charset=utf-8");

PrintWriter out=response.getWriter();

Book[] bookList=(Book[])session.getAttribute("bookList");

out.println("<table border='1'>");

if (bookList!=null){

for (Book book:bookList){

System.out.println(book.getBookNum());

if(book.getBookNum()>0){

out.print("<tr>");

out.print("<td><img src='"+book.getBookImg()+"'></td>");

out.print("<td>"+book.getBookNum()+"</td>");

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

}

}

}

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

out.print("总共"+(String)session.getAttribute("sumNum")+"本");

}

public void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

doGet(request,response);

}

}

**InitServlet.java:**

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

import javax.servlet.annotation.WebServlet;

import cc.test.vo.\*;

/\*\*

\* 程序运行的首页

\* 在此对书库内容进行初始化

\* @author Administrator

\*

\*/

@WebServlet("/index.do")

public final class InitServlet extends HttpServlet {

/\*\*

\* Constructor of the object.

\*/

public InitServlet() {

super();

}

public void destroy() {

super.destroy(); // Just puts "destroy" string in log

// Put your code here

}

/\*\*

\* The doGet method of the servlet. <br>

\*

\* This method is called when a form has its tag value method equals to get.

\*

\* @param request the request send by the client to the server

\* @param response the response send by the server to the client

\* @throws ServletException if an error occurred

\* @throws IOException if an error occurred

\*/

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

HttpSession session=request.getSession();

Book[] BookList=null;//获取图书的相关信息

//ArrayList Vector Map<>

int num=0;//保存图书的本数

//当第一次访问采购网页，session对象中还没有book购物车信息时，进行初始化。

if (session.getAttribute("bookList")==null){

//在session对象中存储图书采购信息,也可以换成从数据库中获取

BookList = new Book[4];

BookList[0] = new Book("10101", "Java核心技术", "img\\10101.jpg", 0);

BookList[1] = new Book("10102", "Java经典实例", "img\\10102.jpg", 0);

BookList[2] = new Book("10103", "Java7", "img\\10103.jpg", 0);

BookList[3] = new Book("10104", "JSP&Servlet", "img\\10104.jpg",0);

//存入session对象中

session.setAttribute("bookList", BookList);

session.setAttribute("sumNum", String.valueOf(num));

}

response.sendRedirect("Shopping.jsp");

}

/\*\*

\* The doPost method of the servlet. <br>

\*

\* This method is called when a form has its tag value method equals to post.

\*

\* @param request the request send by the client to the server

\* @param response the response send by the server to the client

\* @throws ServletException if an error occurred

\* @throws IOException if an error occurred

\*/

public void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

doGet(request,response);

}

/\*\*

\* Initialization of the servlet. <br>

\*

\* @throws ServletException if an error occurs

\*/

public void init() throws ServletException {

// Put your code here

}

}

**ShopingServlet.java:**

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

import cc.test.vo.Book;

public class ShopingServlet extends HttpServlet {

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

Book[] bookList=null;

int bookNum=0;

int sumNum=0;

HttpSession session=request.getSession();

//获取从Shopping.jsp传来的request对象中的ID号

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

//在ID号不为空的情况下，获取Session对象中的购物车信息，

//并将信息保存在Book数组。

try{

if (id!=null){

bookList=(Book[])session.getAttribute("bookList");

sumNum=Integer.parseInt((String)session.getAttribute("sumNum"));

if (bookList!=null){

//找到数组中与ID对应的Book对象，将其属性bookNum数字加1

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

if(id.equals(bookList[i].getBookId())){

bookNum=bookList[i].getBookNum();

bookList[i].setBookNum(bookNum+1);

System.out.print("shop::"+bookNum);

}

}

//将数组写回session对象中去。

session.setAttribute("bookList", bookList);

session.setAttribute("sumNum", String.valueOf(sumNum+1));

}

}

}catch(Exception ex){

System.out.println(ex);

}

//跳转回Shopping.jsp

response.sendRedirect("Shopping.jsp");

}

public void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

doGet(request,response);

}

}

**Book.java:**

**import** java.io.Serializable;

**public** **class** Book **implements** Serializable{

**private** String bookId;

**private** String bookName;

**private** String bookImg;

**private** **int** bookNum;

**public** Book(String bookId, String bookName, String bookImg, **int** bookNum) {

**this**.bookId = bookId;

**this**.bookName = bookName;

**this**.bookImg = bookImg;

**this**.bookNum = bookNum;

}

**public** **void** setBookNum(**int** bookNum) {

**this**.bookNum = bookNum;

}

**public** String getBookId() {

**return** bookId;

}

**public** String getBookName() {

**return** bookName;

}

**public** String getBookImg() {

**return** bookImg;

}

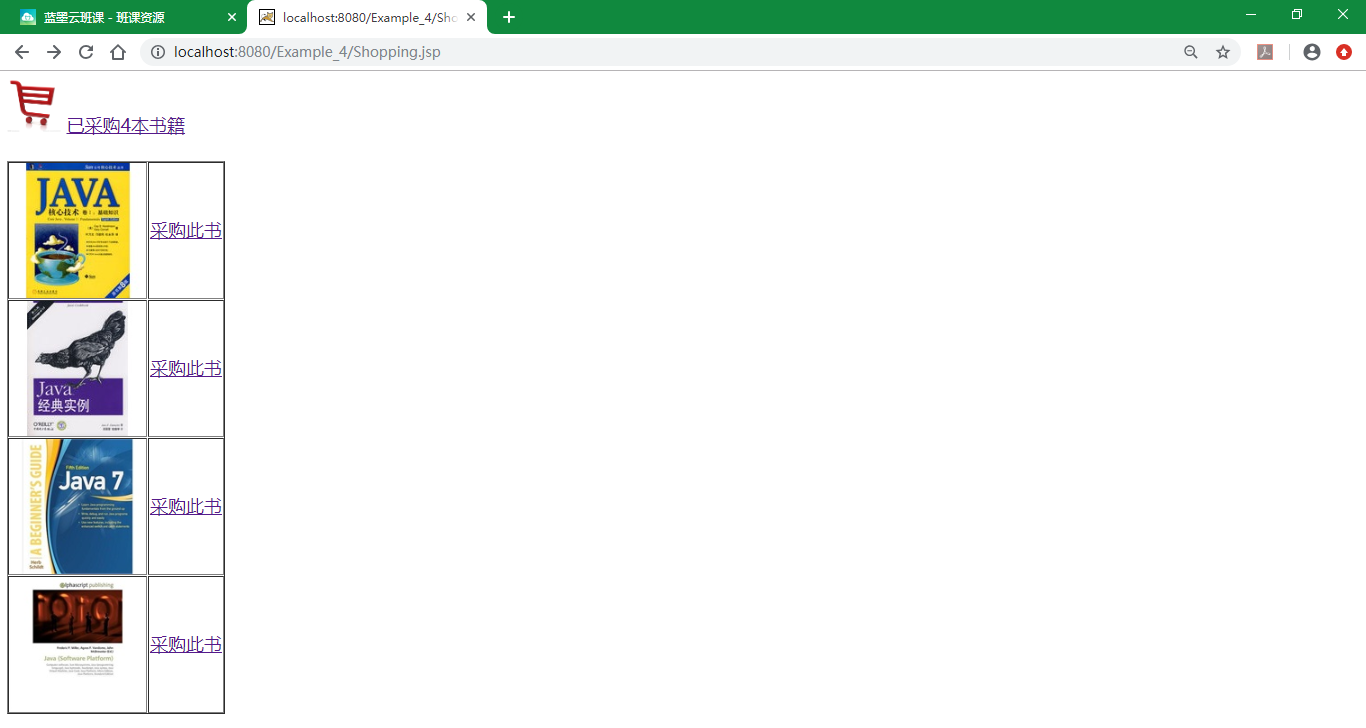
**public** **int** getBookNum() {

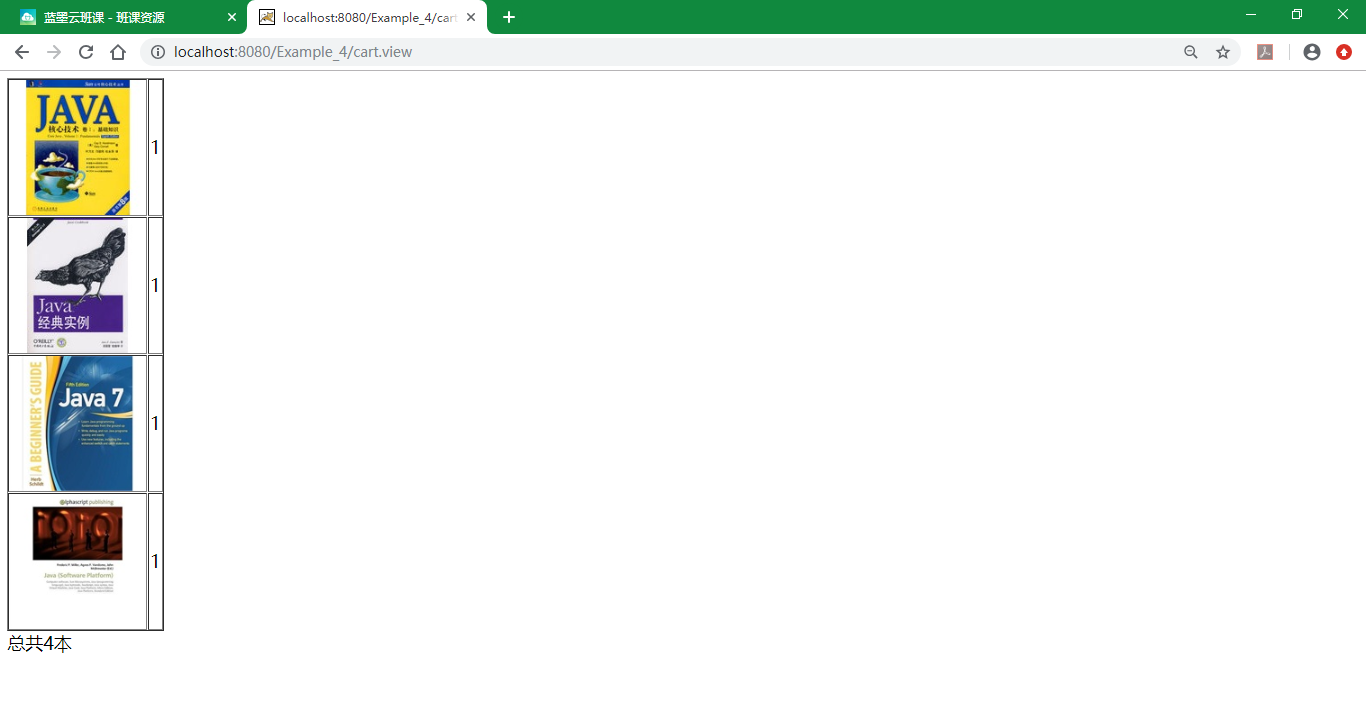
**return** bookNum;

}

}

# 运行效果图





# 收获与问题

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