**Java Web开发**

**实验六**

**作业报告**

|  |  |
| --- | --- |
| 学 院 | 计算机 |
| 所在系 班 级 | 2017软件工程系7班 |
| 学 号 姓 名 | 04161126梁彬 |
| 指 导 教 师 | 郭晓燕 |
| 完 成 时 间 | 2018年5月- |

# 实验题目

1、上周没有完成作业的同学请继续完成，完成了作业的同学请为购物车增加功能：

2、参考成熟的购物系统，为购物车增加功能，完善购物车：删除商品，查看商品具体信息，计算商品总价，修改商品购买件数等等。界面尽量美观。

提示：Java文件即Book.java的设计需要重构，需要分离商品基本信息和购买信息，需要有类分别来保存单件商品的购买信息和所有商品的购买信息。所以至少要三个Java类才能完成购物的基本功能。

Servlet也需要响应更多的操作，可以在一个Servlet中也可以分多个文件响应。并在适当的地方调用Java方法。

将题目贴上去

# 分析与设计

ListCartServlet.java: 转发到listCart.jsp，这个网页去显示已经购买的书的信息。

BusinessService.java:业务类，统一web层提供服务。

CartItem.java:代表某个商品，以及其数量，价格。

Cart.java:记住购物车所有商品多少钱，总价。看看购物车有没有要添加的书对应的购书项。

Book.java:两个构造方法，一个带参数，一个不带参数。

DB.java:储存书的各项信息。

BookDao.java:返回购物信息及id。

listServlet.java:转到listbook.jsp，这个网页去显示可以购买的书的信息。

BuyServlet.java:完成书籍的购买，得到用户的购物车并把书添加到购物车上。

index.jsp:购物车欢迎页面。

listbook.jsp:显示可以购买的书的信息。

listCart.jsp:显示已经购买的书的信息。

根据题目要求，计划设计几个文件来完成实验，每个文件的名字、作用。如果是类，要求写出类的变量和方法

# 关键源码（部分）

**ListCartServlet.java:**

package com.web;

import java.io.IOException;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

public class ListCartServlet extends HttpServlet {

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

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

}

public void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

doGet(request,response );

}

}

**BusinessService.java:**

package com.service;

import java.util.Map;

import com.dao.BookDao;

import com.domain.Book;

//业务类，统一对web层提供服务

public class BusinessService {

private BookDao dao = new BookDao();

public Map<String, Book> getAllBook(){

return dao.getAll();

}

public Book findBook(String id){

return dao.find(id);

}

}

**CartItem.java:**

package com.domain;

import java.math.BigDecimal;

//代表某个商品，以及其数量

public class CartItem {

private Book book;

private int quantity;

private double price; //小计

BigDecimal b = new BigDecimal(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()\*this.quantity;

}

public double getPrice() {

return price;

}

public void setPrice(double price) {

this.price = price;

}

}

**Cart.java:**

package com.domain;

import java.util.LinkedHashMap;

import java.util.Map;

public class Cart {

private Map<String,CartItem> map = new LinkedHashMap<String, CartItem>();

private double price; //记住购物车所有商品多少钱，总价

public void add(Book book){

//看看购物车有没有要添加的书对应的购书项

CartItem item = map.get(book.getId());

if(item==null){

item = new CartItem();

item.setBook(book);

item.setQuantity(1);

map.put(book.getId(), item);

}else{

item.setQuantity(item.getQuantity()+1);

}

}

public Map<String, CartItem> getMap() {

return map;

}

public void setMap(Map<String, CartItem> map) {

this.map = map;

}

public double getPrice() {

double totalPrice = 0;

for(Map.Entry<String, CartItem> entry:map.entrySet()){

CartItem item = entry.getValue();

totalPrice += item.getPrice();

}

this.price = totalPrice;

return price;

}

public void setPrice(double price) {

this.price = price;

}

}

**Book.java:**

package com.domain;

public class Book {

private String id;

private String name;

private String author;

private double price;

private String description;

//两个构造方法，一个带参数，一个不带参数

public Book() {

super();

}

public Book(String id, String name, String author, double price,

String description) {

super();

this.id = id;

this.name = name;

this.author = author;

this.price = price;

this.description = description;

}

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 double getPrice() {

return price;

}

public void setPrice(double price) {

this.price = price;

}

public String getDescription() {

return description;

}

public void setDescription(String description) {

this.description = description;

}

}

**DB.java:**

package com.db;

import java.util.LinkedHashMap;

import java.util.Map;

import com.domain.Book;

public class DB {

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

static{

map.put("1", new Book("1","JSP&Servlet学习笔记","林信良",78,"JavaWeb开发技术"));

map.put("2", new Book("2","数据结构","翁惠玉",52,"计算机基础"));

map.put("3", new Book("3","英语","吴晓真",48,"学习英语的书"));

}

public static Map<String, Book> getAll(){

return map;

}

}

**BookDao.java:**

package com.dao;

import java.util.Map;

import com.db.DB;

import com.domain.Book;

public class BookDao {

public Map<String, Book> getAll(){

return DB.getAll();

}

public Book find(String id){

return (Book) DB.getAll().get(id);

}

}

**listServlet.java:**

package com.Controller;

import java.io.IOException;

import java.util.Map;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import com.domain.Book;

import com.service.BusinessService;

public class listServlet extends HttpServlet {

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

BusinessService service = new BusinessService();

Map<String, Book> map = service.getAllBook();

request.setAttribute("map", map);

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

}

public void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

doGet(request,response);

}

}

**BuyServlet.java:**

package com.Controller;

import java.io.IOException;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import com.domain.Book;

import com.domain.Cart;

import com.service.BusinessService;

public class BuyServlet extends HttpServlet {

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

//完成书籍的购买

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

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

BusinessService service = new BusinessService();

Book book = service.findBook(id);

//得到用户的购物车

Cart cart = (Cart) request.getSession().getAttribute("cart");

if(cart ==null){

cart = new Cart();

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

}

//把书添加到用户的购物车上

cart.add(book);

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

}

public void doPost(HttpServletRequest request,HttpServletResponse response)

throws ServletException,IOException {

doGet(request,response);

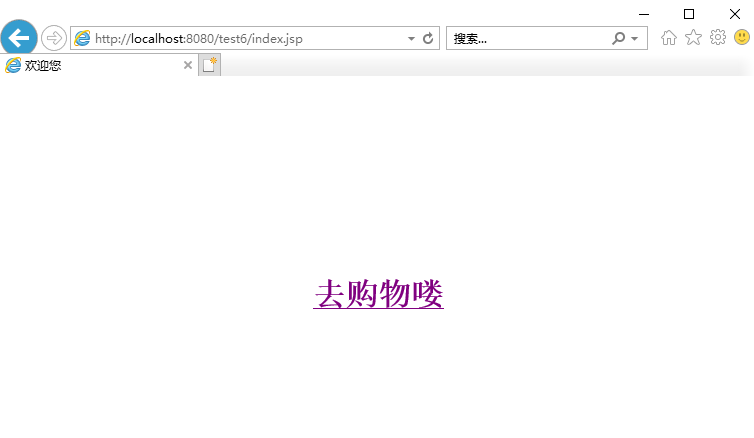
}

}

贴上这道题中最关键的部分源码。也是你实验练习的主要知识点相关的源码。并在代码旁附上详细的解释

# 运行效果图

首页欢迎页面：



购物车首页页面：



购买书籍：



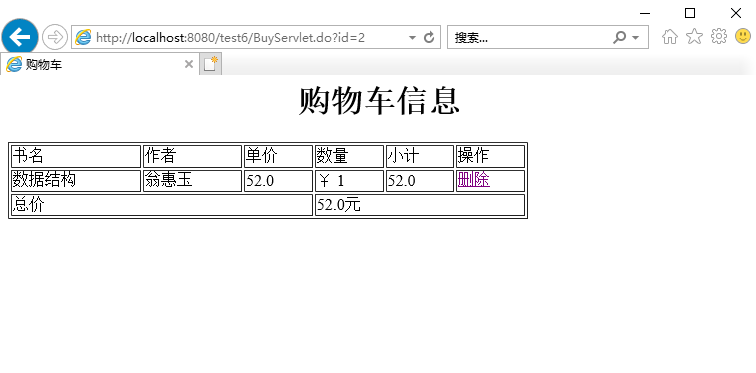
再次购买同一本书籍：



购买其它书籍：



删除书籍：



购物车中显示已经购买的书的信息，包括书名，作者，单价和数量。还有计算功能如小计和总价。还有删除操作。

实验在浏览器下运行的结果截图，并进行解释。

# 收与问题

通过session实现购物车的各项功能。进一步了解控制层的各项功能。把类写成java bean，学习如何处理数据。

谈一谈你在本实验中所收获的内容及问题。