1\_新建demo01.html页面,实现如下效果



代码如下:

<table border="1" width="50%" align="center">

<tr><td>商品名称</td><td>查看</td></tr>

<tr>

<td>雷柏机械键盘</td>

<td><a href='/day38\_homework/ServletDemo04?pid=p001'>查看</a></td></tr>

<tr>

<td>联想笔记本电脑</td>

<td><a href='/day38\_homework/ServletDemo04?pid=p002'>查看</a></td></tr>

<tr>

<td>小米6手机</td>

<td><a href='/day38\_homework/ServletDemo04?pid=p003'>查看</a></td></tr>

<tr>

<td>华为荣耀8手机</td>

<td><a href='/day38\_homework/ServletDemo04?pid=p004'>查看</a></td></tr>

<tr>

<td>小米手环</td>

<td><a href='/day38\_homework/ServletDemo04?pid=p005'>查看</a></td></tr>

<tr>

<td colspan='2'>

<a href="/day38\_homework/ServletDemo06">查看购物车内容</a></td>

</tr>

</table>

2\_实现Product类,代表各个商品信息

public class Product {

private String pid;//商品编号

private String pname;//商品名称

private String pdesc; //商品描述

private double price; //价格

private int num;//商品数量

//getter and setter....

}

3\_实现类DB,在本类中我们用一个MAP来模拟了仓库中的各个商品信息,

并在其中实现一个根据PID获取商品信息的方法

public class DB {

private static Map<String,Product> map=new HashMap<String,Product>();

static{

//public Product(String pid, String pname, String pdesc, double price) {

Product p1=new Product("p001","雷柏机械键盘","雷柏的机械键盘是目前市面上性价比最好的键盘,好用",199);

Product p2=new Product("p002","联想笔记本电脑","雷柏的机械键盘是目前市面上性价比最好的键盘,好用",7999);

Product p3=new Product("p003","小米6手机","雷柏的机械键盘是目前市面上性价比最好的键盘,好用",1999);

Product p4=new Product("p004","华为荣耀8手机","雷柏的机械键盘是目前市面上性价比最好的键盘,好用",2001);

Product p5=new Product("p005","小米手环","雷柏的机械键盘是目前市面上性价比最好的键盘,好用",149);

map.put("p001", p1);

map.put("p002", p2);

map.put("p003", p3);

map.put("p004", p4);

map.put("p005", p5);

}

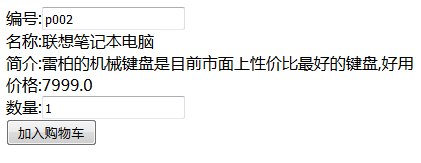
public static Product getProductByPid(String pid){

return map.get(pid);

}

}

4\_实现ServletDemo04,获取各个商品的PID,向客户端输出商品详情信息,效果如下



代码如下:

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

PrintWriter out = response.getWriter();

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

Product p = DB.getProductByPid(pid);

out.println("<form action='/day38\_homework/ServletDemo05' method='post'>");

out.println("编号:<input type='text' name='pid' value='"+p.getPid()+"'/><br/>");

out.println("名称:"+p.getPname()+"<br/>");

out.println("简介:"+p.getPdesc()+"<br/>");

out.println("价格:"+p.getPrice()+"<br/>");

out.println("数量:<input type='text' name='num' value='1'/><br/>");

out.println("<input type='submit' value='加入购物车'/>");

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

5\_实现ServletDemo05,接收商品编号和数量,将其放入到MAP中(这个MAP代表购物车),将

Map放入session域对象内,并向客户端生成相应信息

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

PrintWriter out = response.getWriter();

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

int num = Integer.parseInt(request.getParameter("num"));

Map<String, Product> map =

(Map<String, Product>) request.getSession().getAttribute("cart");

if (null == map) {

map = new HashMap<String, Product>();

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

}

if (!map.containsKey(pid)) {

Product pro = DB.getProductByPid(pid);

pro.setNum(1);

map.put(pid, pro);

} else {

Product pro = map.get(pid);

pro.setNum(pro.getNum() + num);

}

out.println("商品加入购物车成功<br/>");

out.println("<a href='/day38\_homework/demo01.html'>请继续浏览商品</a><br/>");

out.println("<a href='/day38\_homework/ServletDemo06'>浏览购物车</a><br/>");

6\_实现ServletDemo06代码,实现点击浏览购物车的链接时,实现的功能

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

PrintWriter out = response.getWriter();

Map<String,Product> map=(Map<String,Product>)request.getSession().getAttribute("cart");

if(null==map||map.size()==0){

String str=”<a href='/day38\_homework/demo01.html'>浏览商品</a>”;

out.println("购物车中暂无商品信息"+str);

}else{

for(Product pro:map.values()){

out.println("商品名称: "+pro.getPname()+"<br/>");

out.println("商品价格: "+pro.getPrice()+"<br/>");

out.println("商品数量"+pro.getNum()+"<br/><br/>");

}

}

out.println("<a href='/day38\_homework/demo01.html'>继续浏览商品</a>");