中南大学

Web应用开发技术

实验报告

|  |  |
| --- | --- |
| 姓 名： | XXX,XXX |
| 学 号： | XXXXXX, XXXXXXX |
| 班 级： | 软件工程190X |
| 指导老师： | 宋 铁 |
| 完成时间： |  |
| 分 数： |  |
| 批阅日期： |  |

1. **实验内容——基本任务。**
2. 商品展示业务模块，包括大类Category、小类Product和具体商品Item的展示和搜索功能。
3. 用户管理业务模块，包括用户注册、登录、修改用户信息、查询用户相关订单等业务功能。
4. 订单管理模块，包括购物车管理、新增订单、地址信息等业务功能。

**//以上三个业务模块，用运行界面截图和核心代码展示功能完成情况。**

1. **实验内容——扩展任务。**
2. 添加验证码功能：在用户注册和用户登录模块中添加验证码功能。
3. 购物车功能：原项目中购物车为内存购物车，即将用户相关的购物车存储在session中；在数据库中添加购物车表，将购物车数据持久化，以便用户下次登录时可查看或修改自己的购物车数据。
4. 日志功能：数据库中添加日志信息表，给项目添加日志功能，用户登录后记录用户行为，比如浏览了哪些商品、将商品添加进购物车、生成订单等。

**//以上两个扩展任务，除用界面截图和核心代码展示功能完成情况外，验证码功能还需说明验证码的基本原理和实现方案，日志功能需描述项目所完成的功能情况（能记录哪些日志、怎么记录等）。**

1. **小组成员分工及实验总结**
2. 小组分工：

XXX为组长负责XXX，XXX负责XXX，XXX负责XXX。。。

1. 实验总结

实验总结记录实验过程遇到的问题和解决的方法，以及心得体会，字数不少于500字。

一、实验内容------基本内容:

**1、商品展示模块，包括大类Category、小类Product和具体商品Item的展示和搜索功能：**

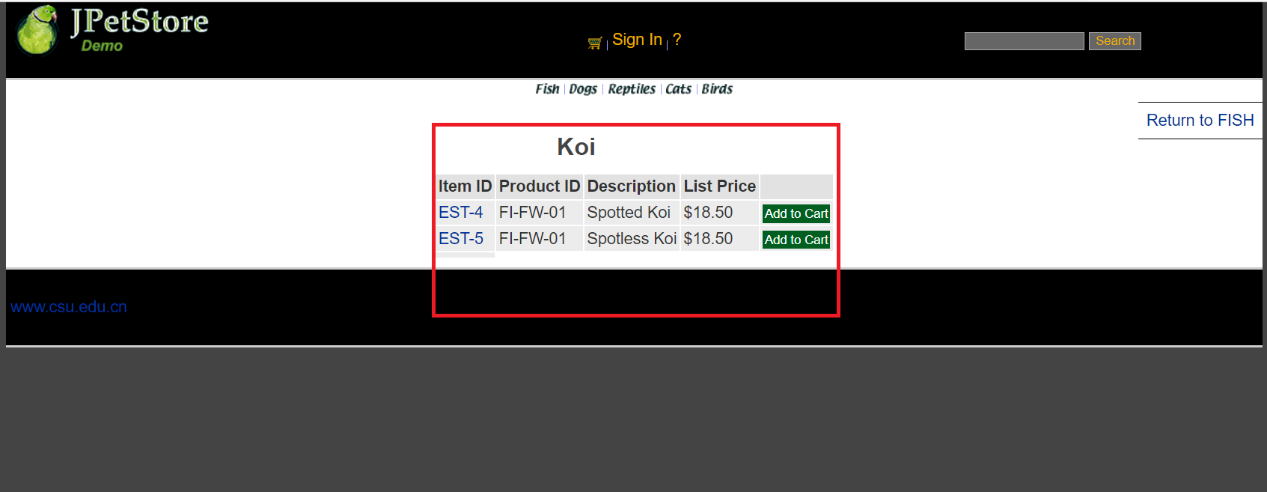
1、运行界面截图：

（1）大类Category分为：Fish、Dogs、Reptiles、Cats、Birds，此处选取Fish为例子，运行截图如下：



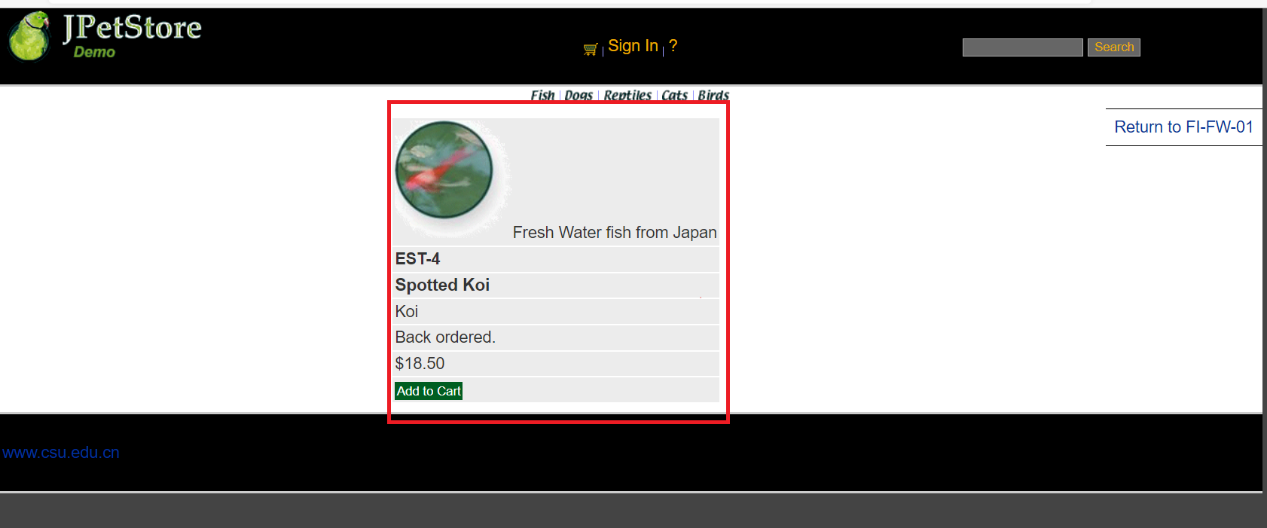
（上图中红框圈住的均为大类Category，点击即可进入大类所包含的product部分的展示，且上方黑框包含的部分为各个页面通用的，即每个页面都存在，包含上方Category种类列表）

（2）小类product展示如下（此处以Koi为例子）：



该界面展示了名为Koi的Product所包含的全部Item信息，点击Item ID，即可跳转至相应的Item

（3）Item展示：

此处以ID为EST-4的Item为例，点击其ID，展示其详细信息

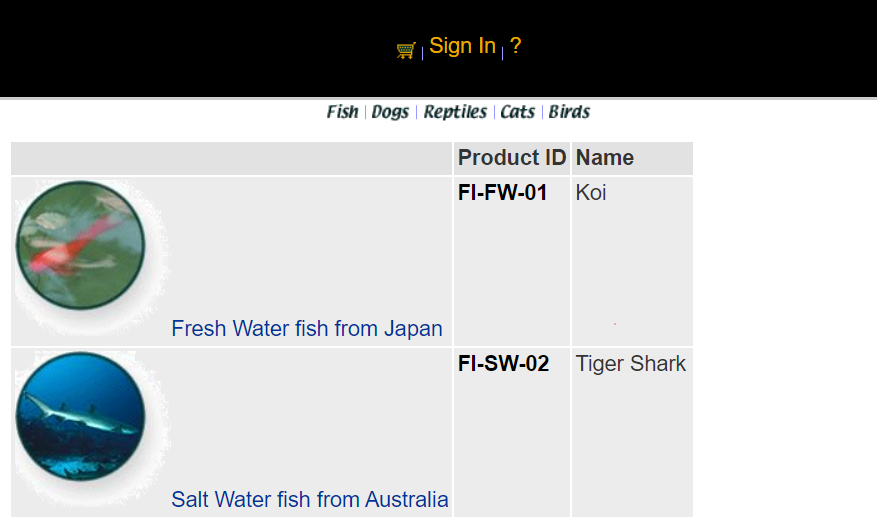
该商品的详细内容如上图红框中内容所示.

(4)商品的搜索功能:

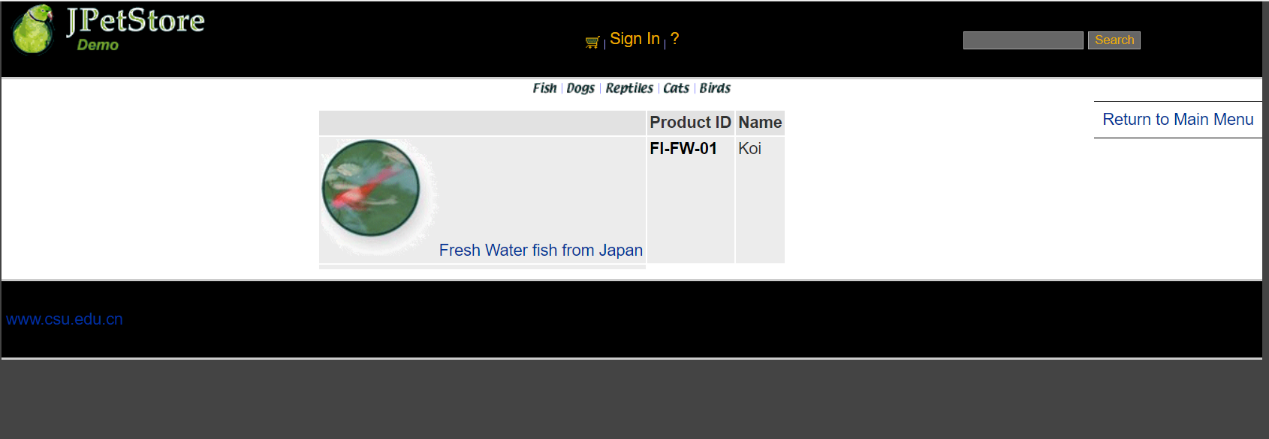
本程序的搜索功能具体到Item,搜索的依据时ItemName,且为模糊搜索\不区分大小写,接下来我们的目的是搜索到Koi,先输入k进行尝试



点击右侧Search按钮,搜索到名字里包含k的Item,相关信息如下:



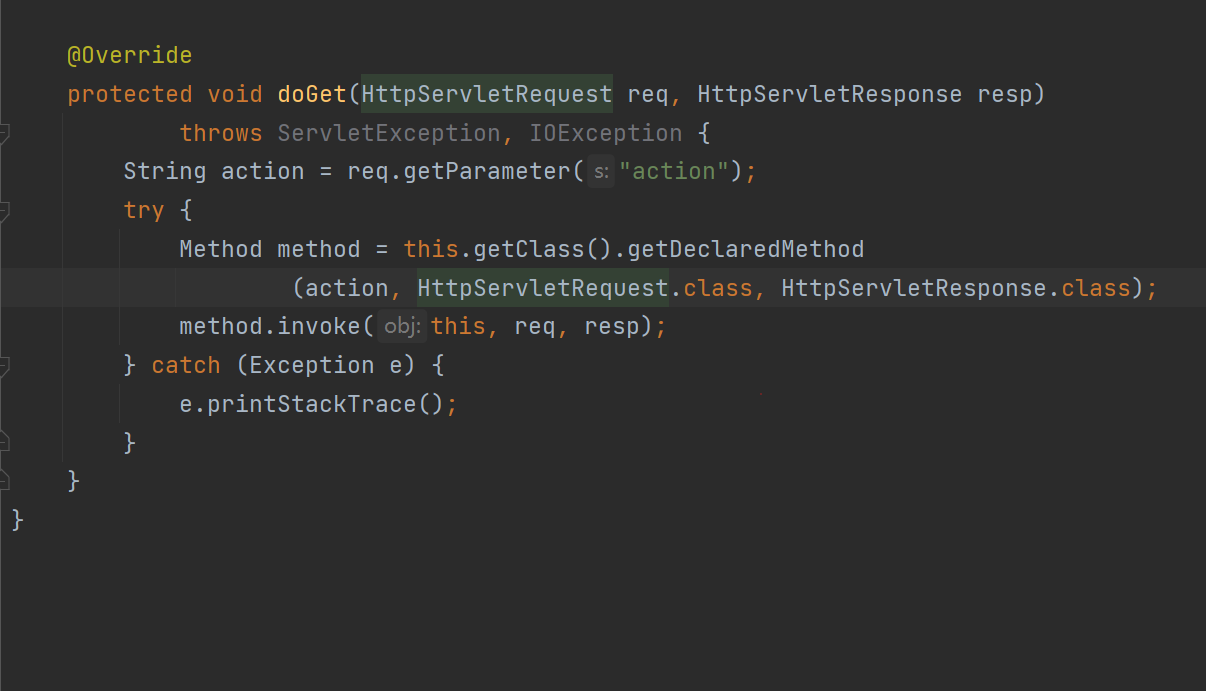
当输入koi时，点击search，展示的内容如下：



搜索成功

2、核心代码：

（1）为了简化代码，本程序在Servlet部分采取了反射的方式（由组长郑乾坤完成），作为基础的Servlet代码截图如下：



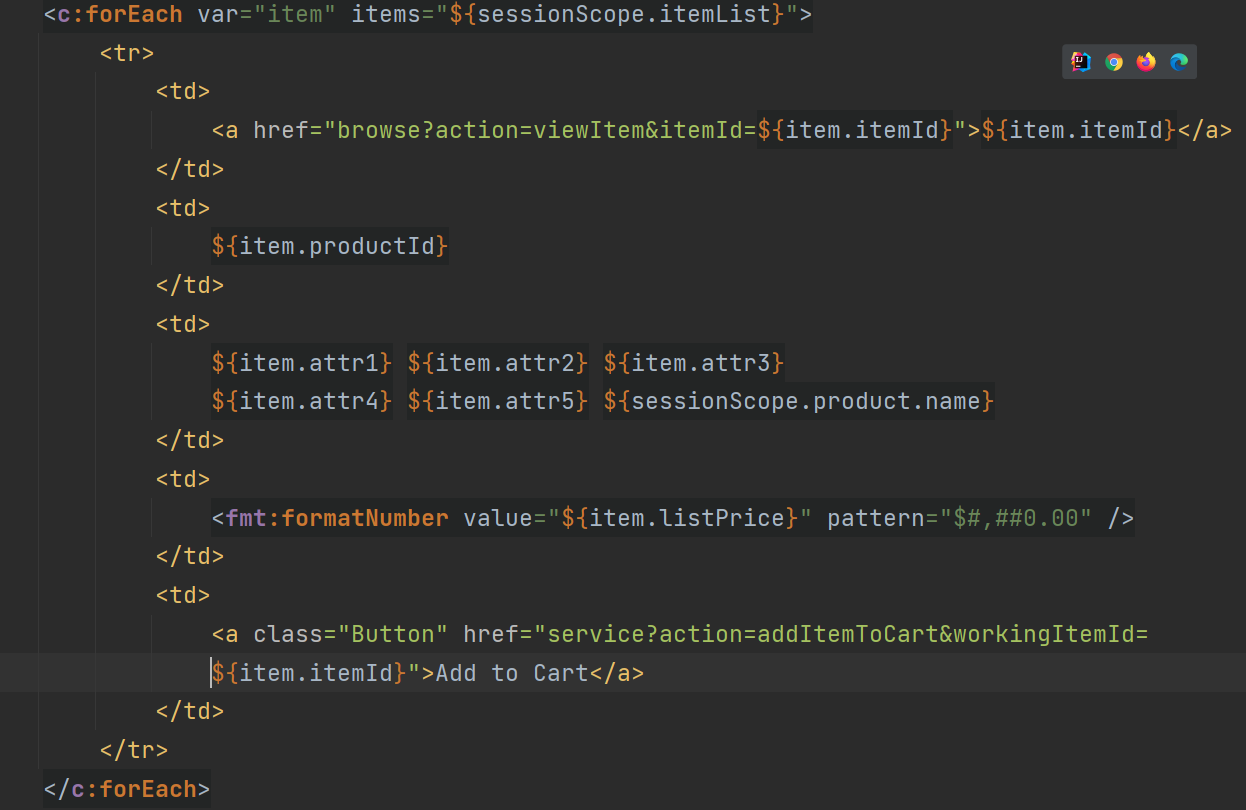
（2）展示功能Category的Servlet和Jsp部分核心代码：





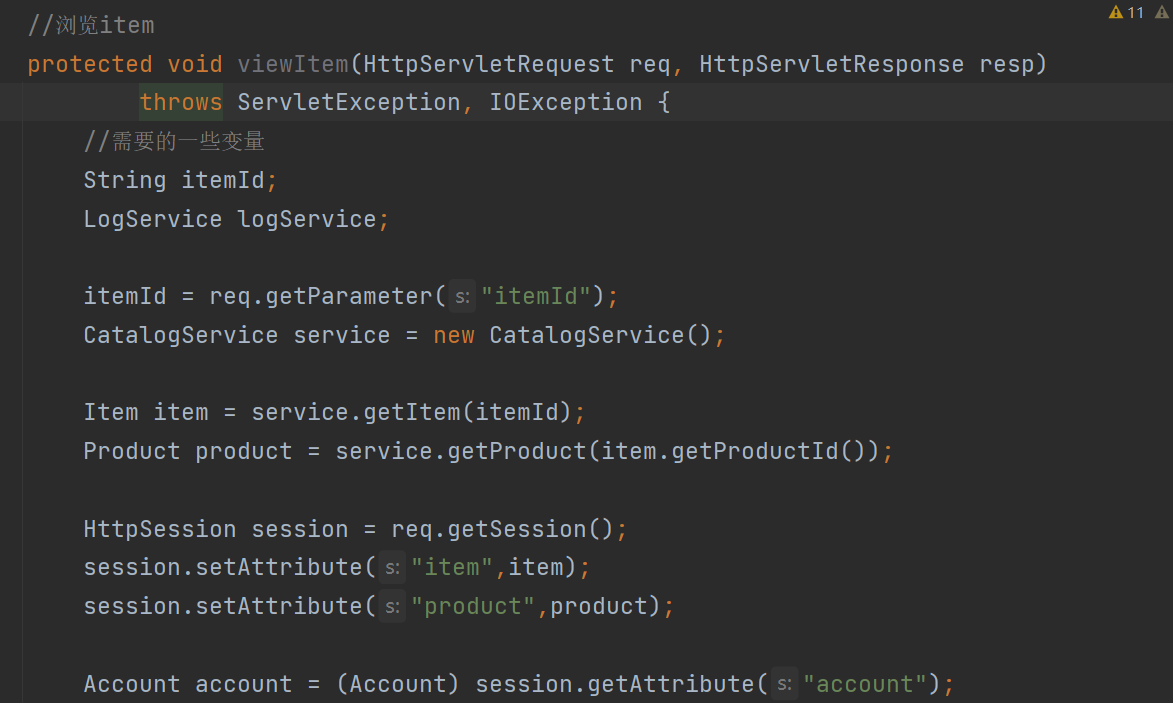
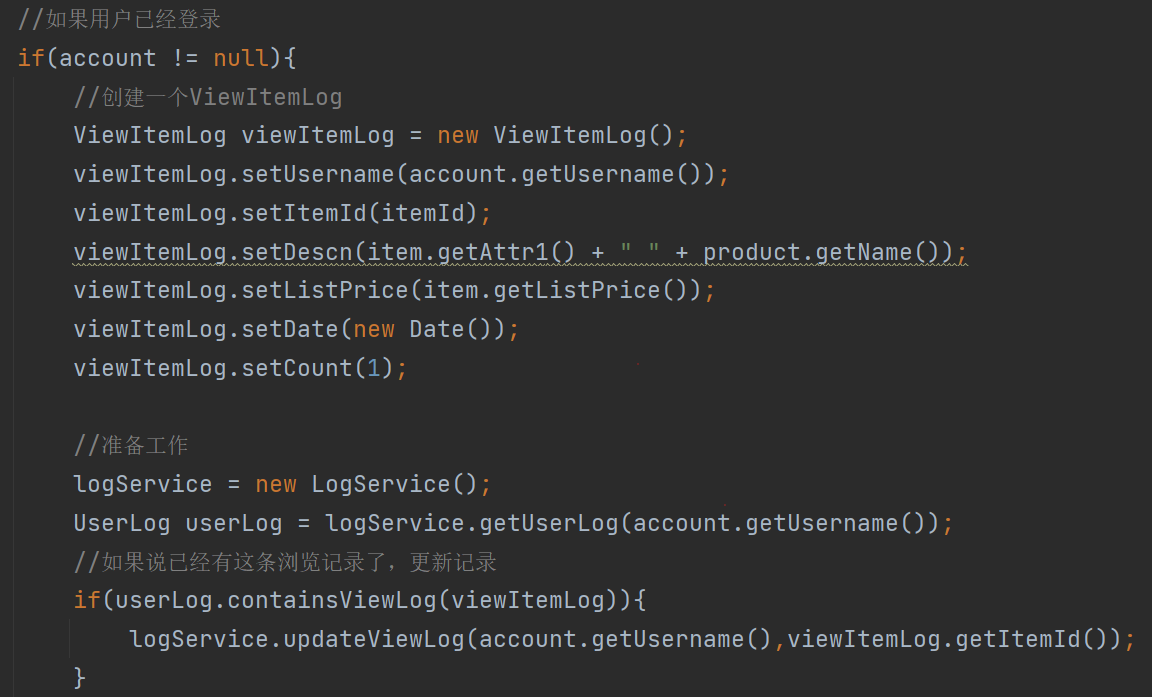
（3）浏览Product的Servlet和Jsp核心代码如下：

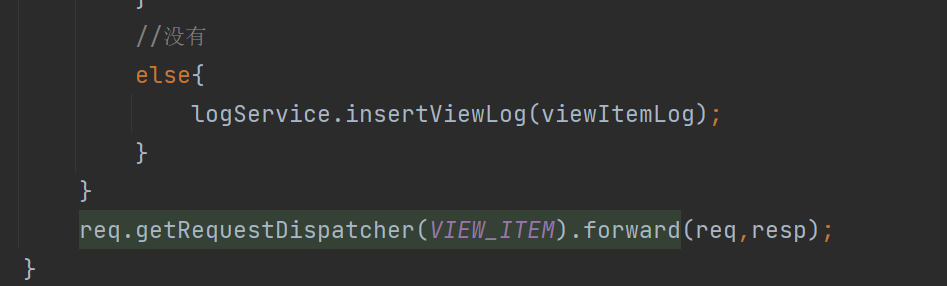




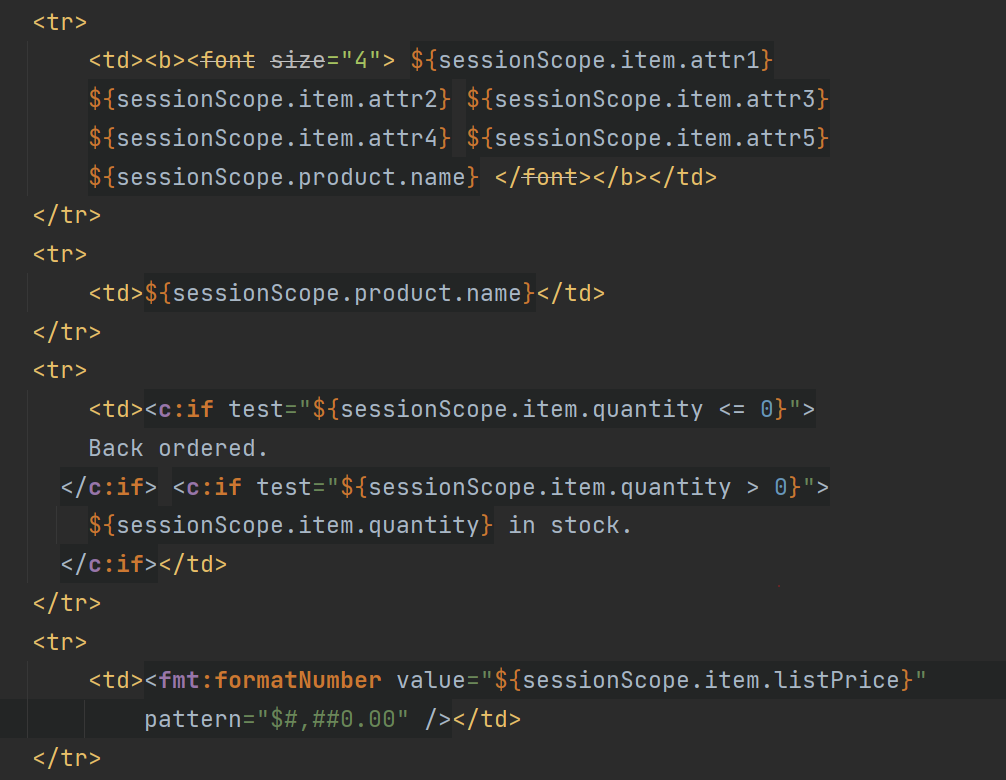
（4）浏览Item的Servlet和Jsp核心代码：

Servlet：



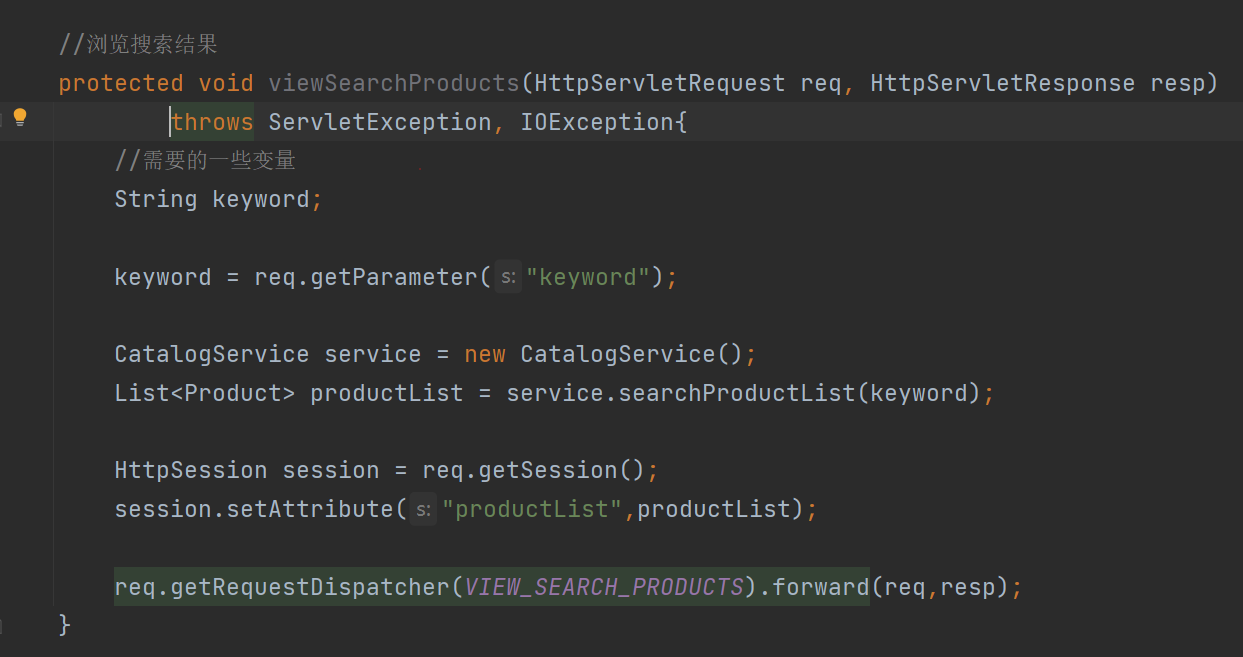


JSP：

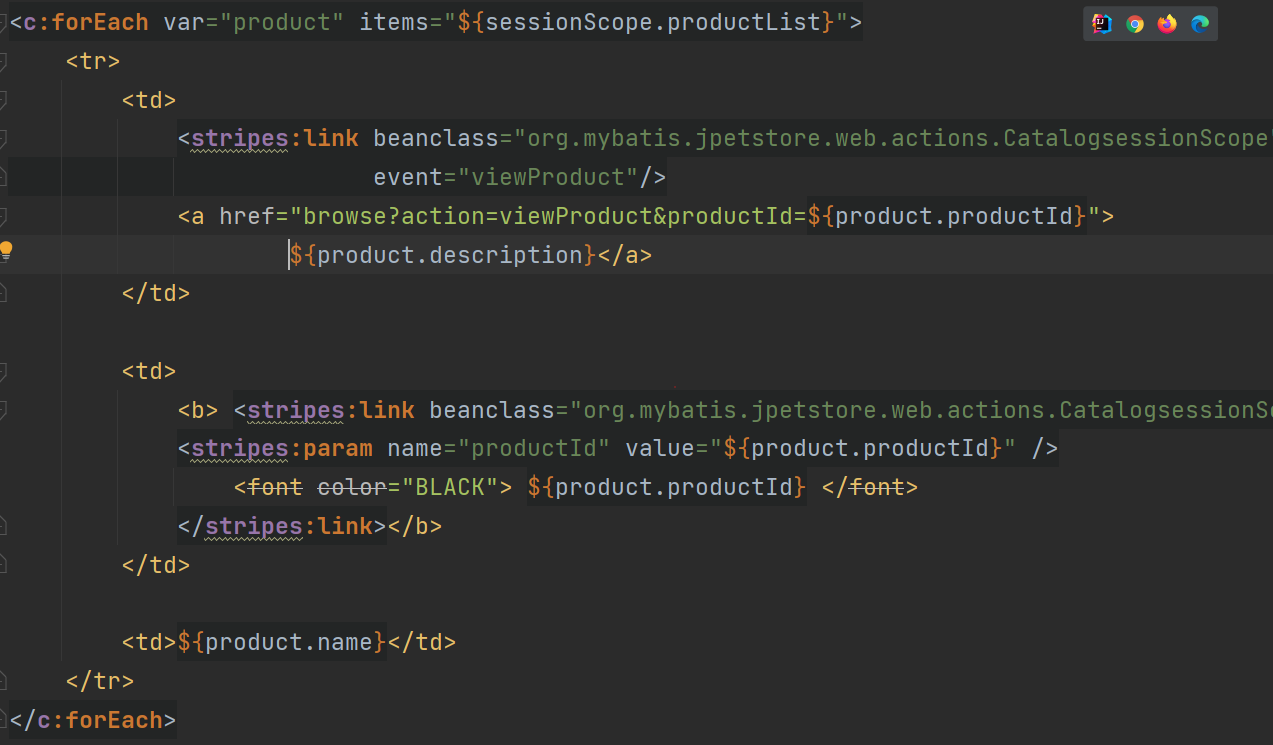


（5）浏览搜索部分的Servlet和Jsp：

Servlet：



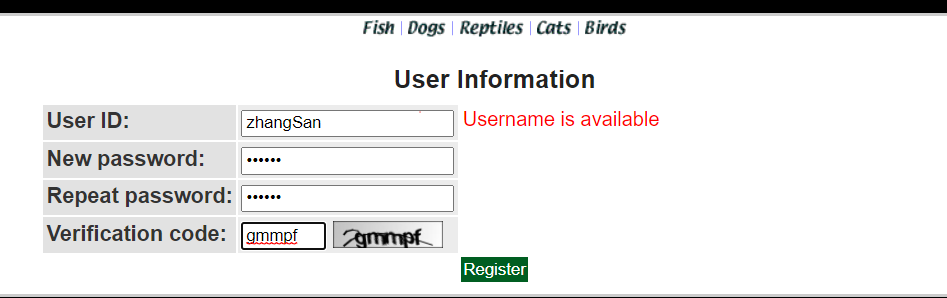
Jsp：



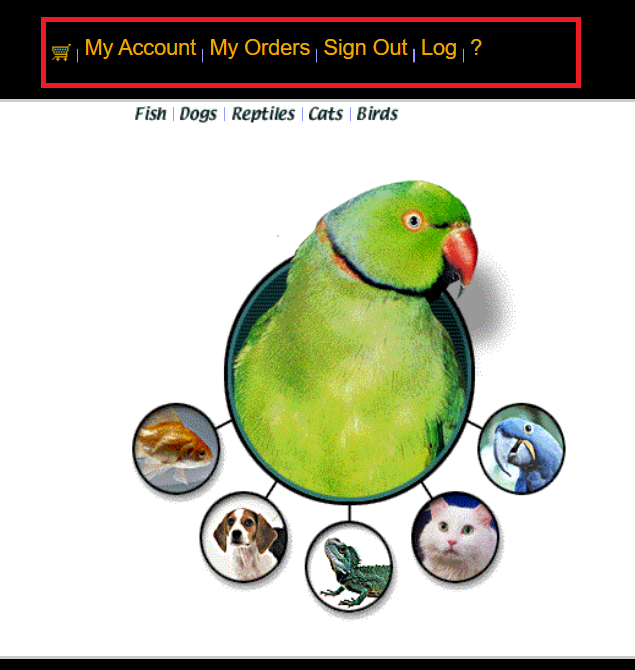
**2、用户管理业务模块，包括用户注册、登录、修改用户信息、查询用户相关订单等业务功能。**

1、运行界面截图：

（1）注册界面



点击Register，注册成功，自动登录

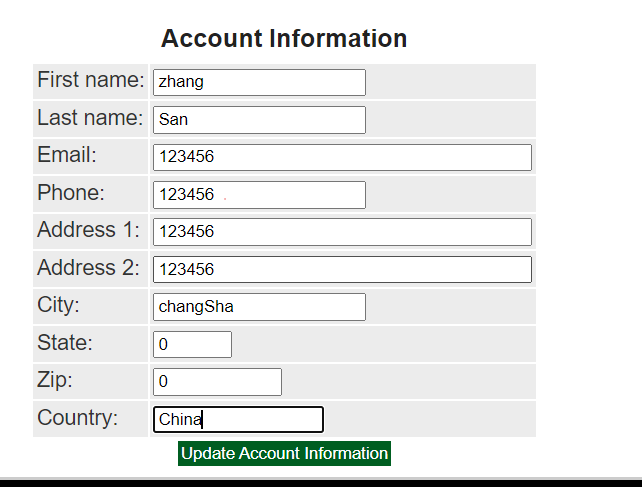


上方选择栏刷新



数据库内相关内容也刷新

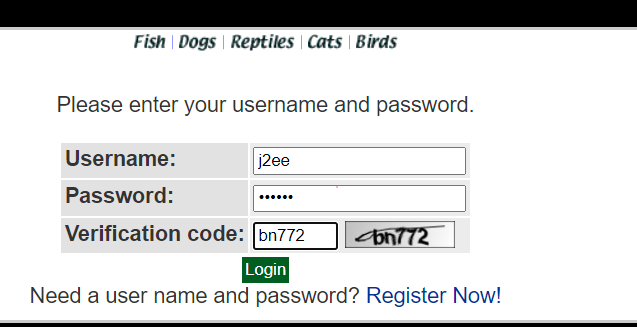
（2）更改账号详细信息：



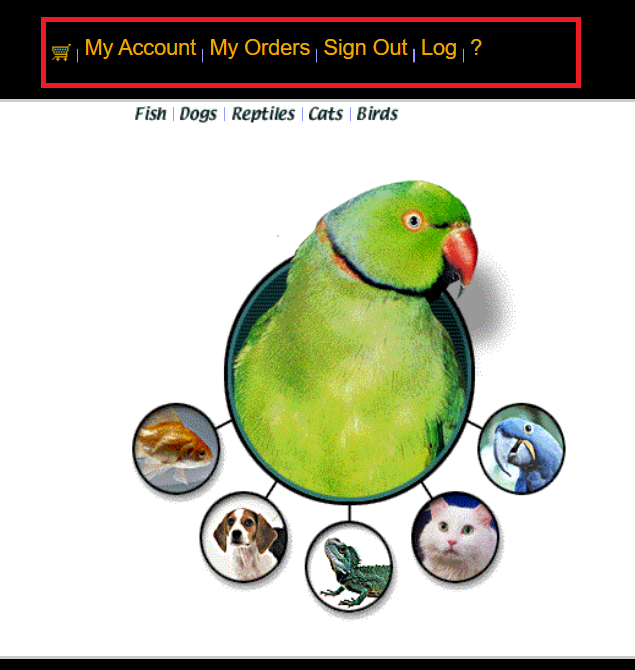
点击下方更新键，数据库内信息刷新：



（3）、登录界面：



点击登陆成功，界面跳转：



（4）查看用户订单：

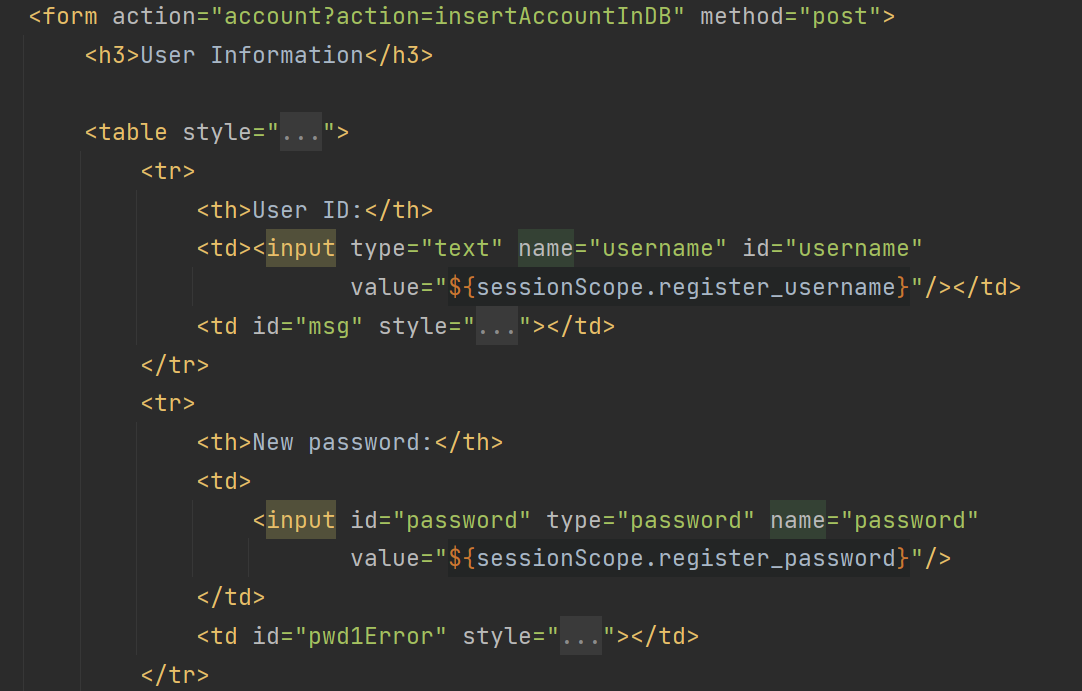


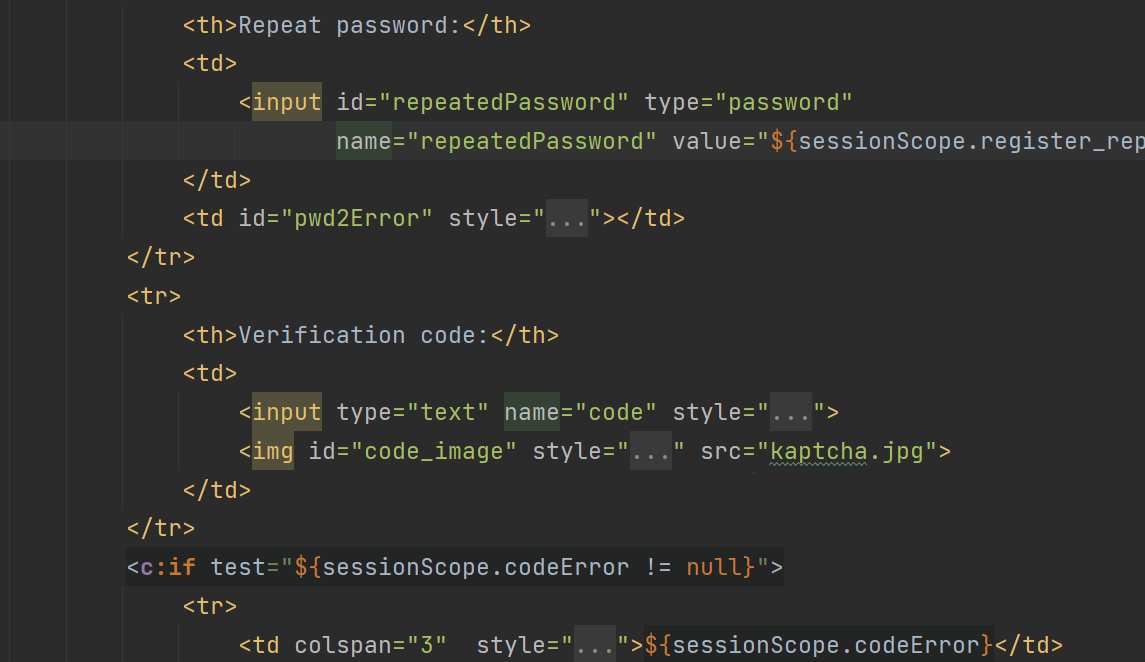
2、核心代码：

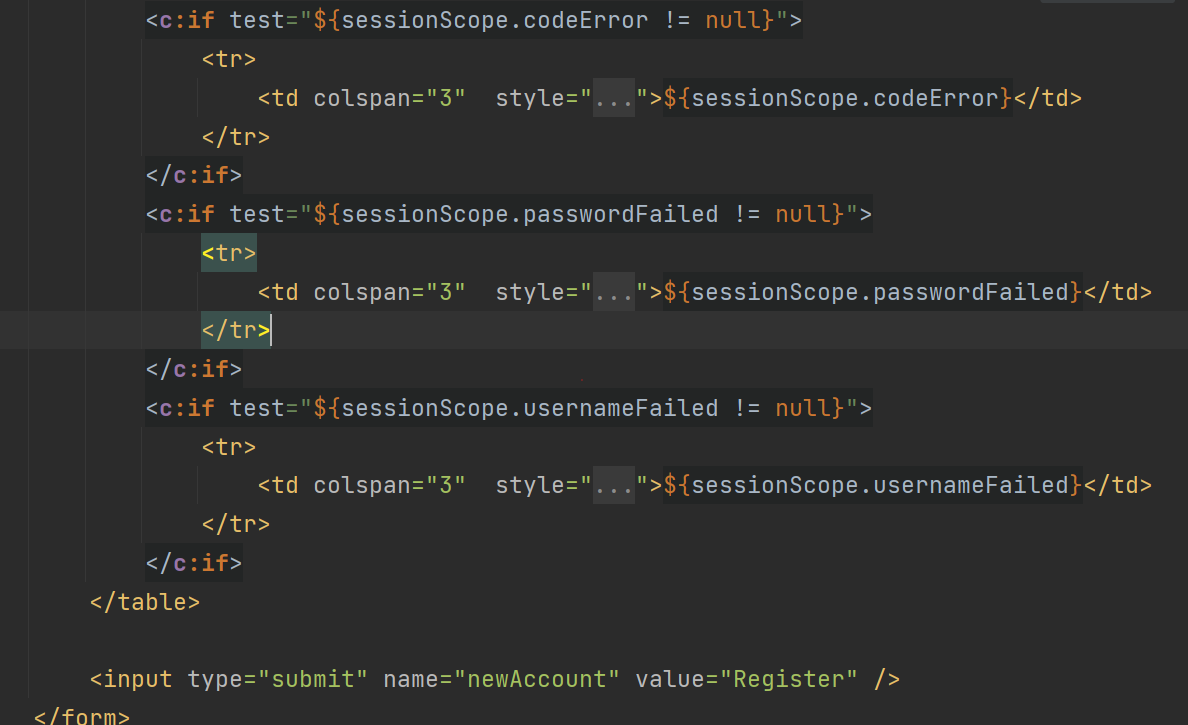
（1）注册功能的Servlet和Jsp：

Servlet：









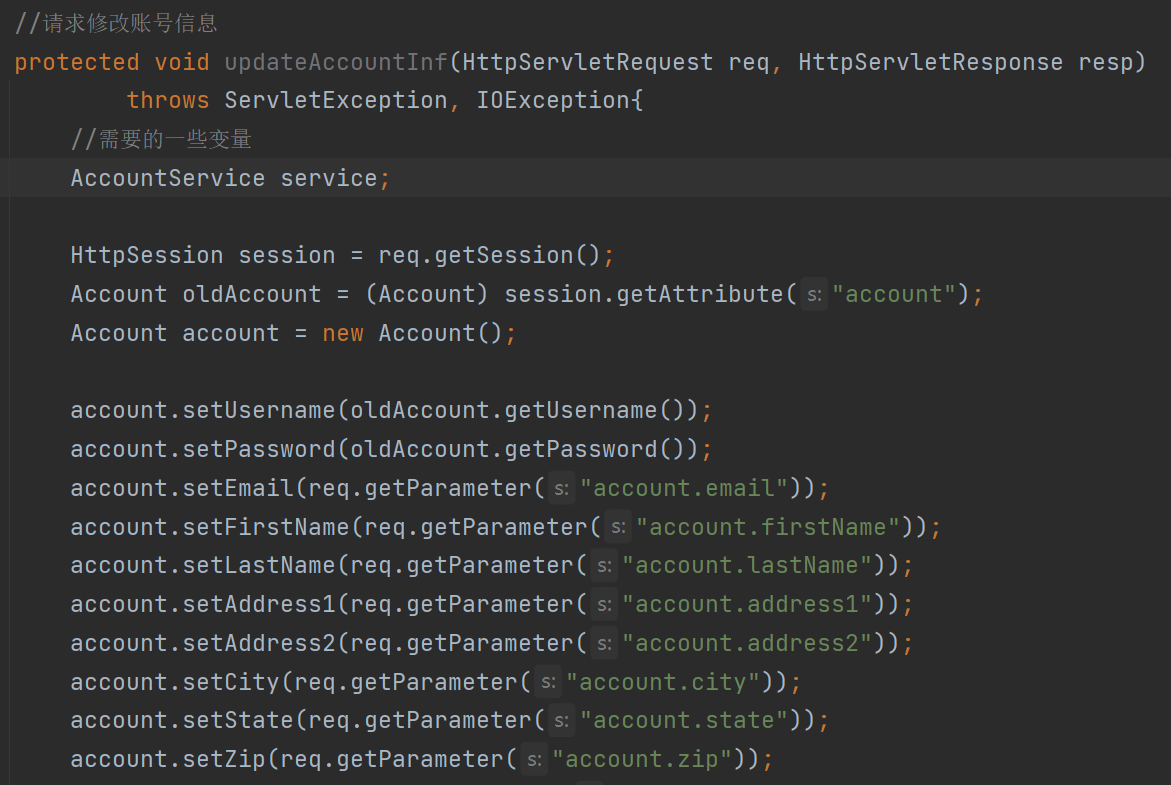
（2）登陆的Servlet和Jsp的核心代码：

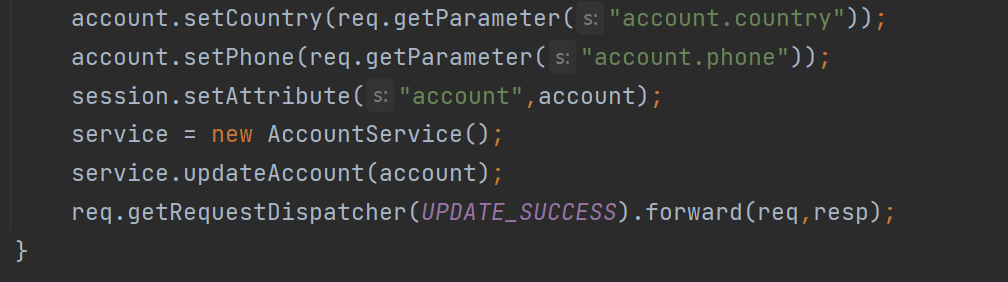
Servlet：  


Jsp：

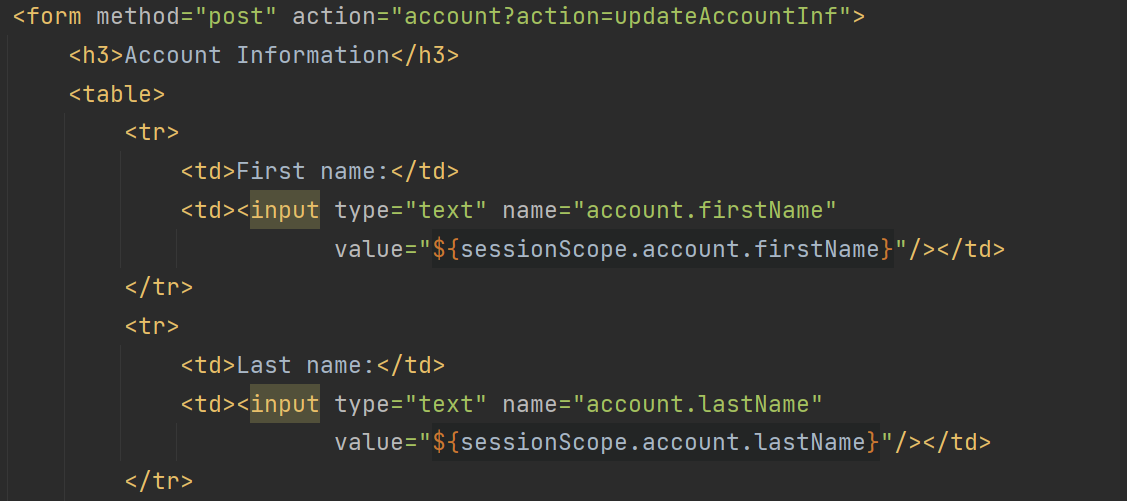


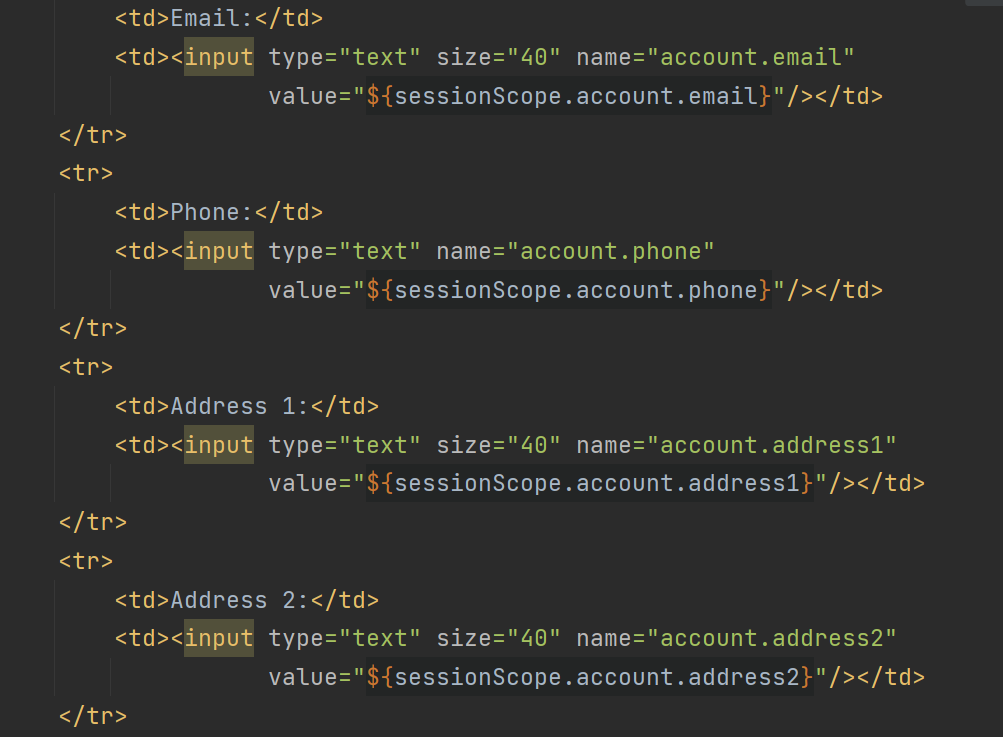
（3）编辑账号信息的Servlet和Jsp：  
Servlet：

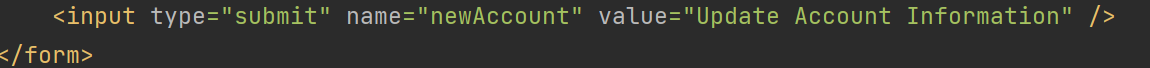




Jsp：

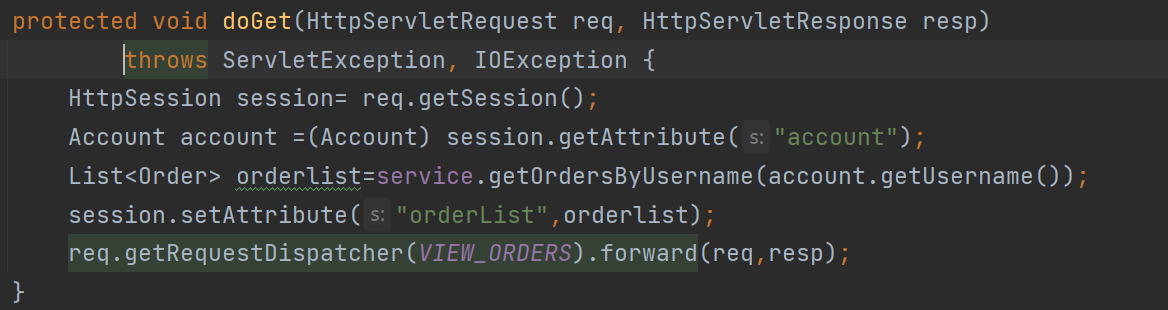




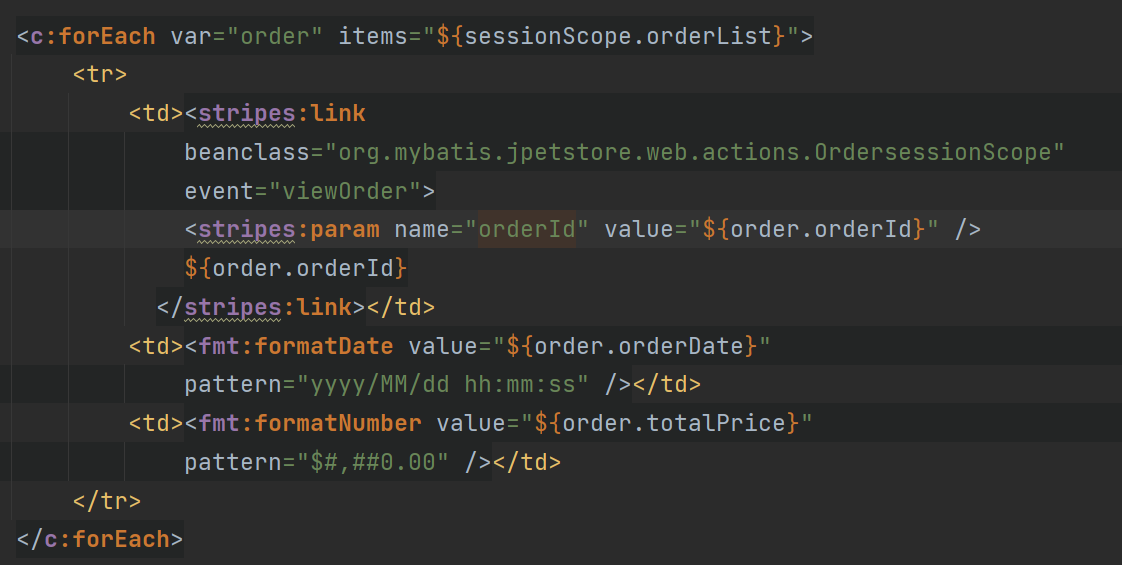


（4）查询用户购物信息的Servlet和Jsp：

Servlet



Jsp：

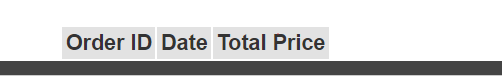


**3、订单管理模块，包括购物车管理、新增订单、地址信息等业务功能。**

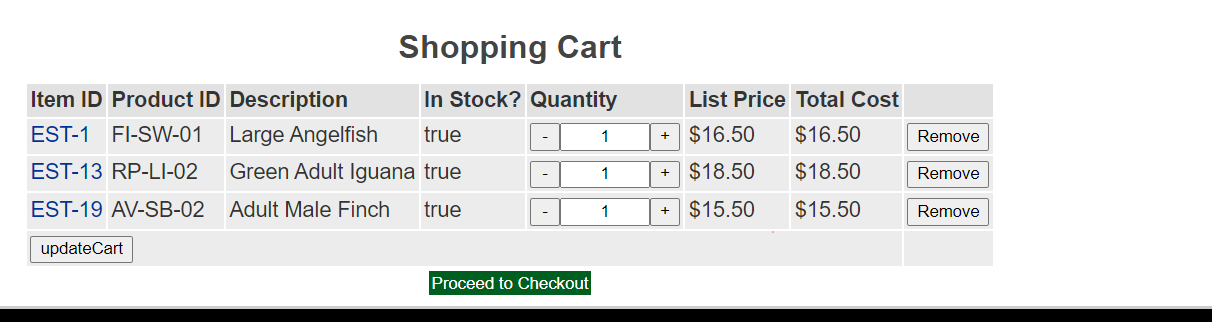
1、运行界面截图：

（1）购物车管理：

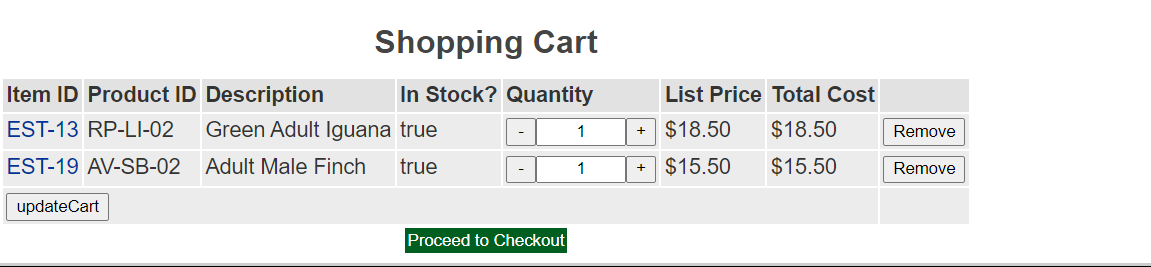
-该账号初始订单情况（为空）：



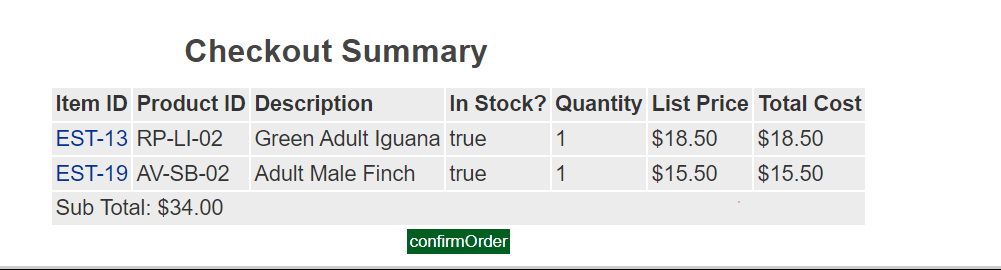
-购物车初始状况



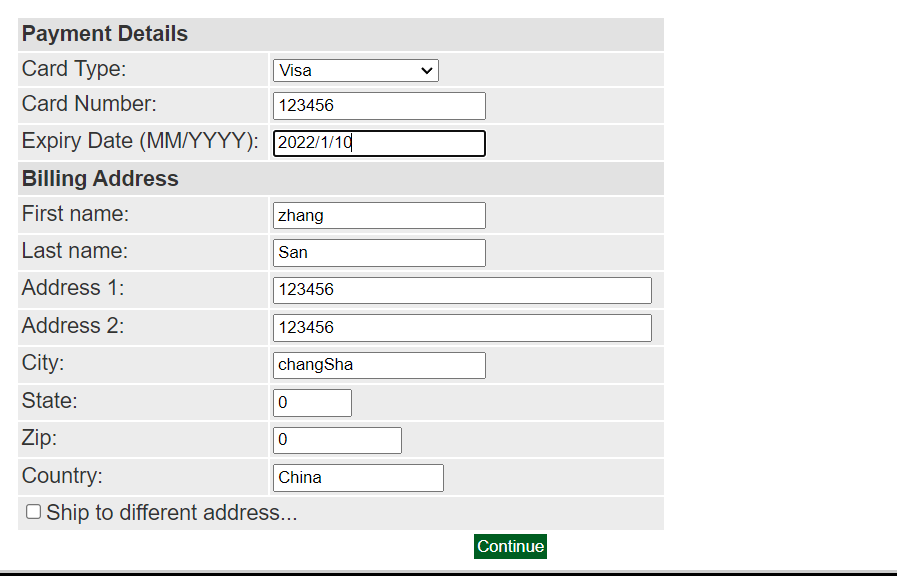
-删除EST-1这件商品：



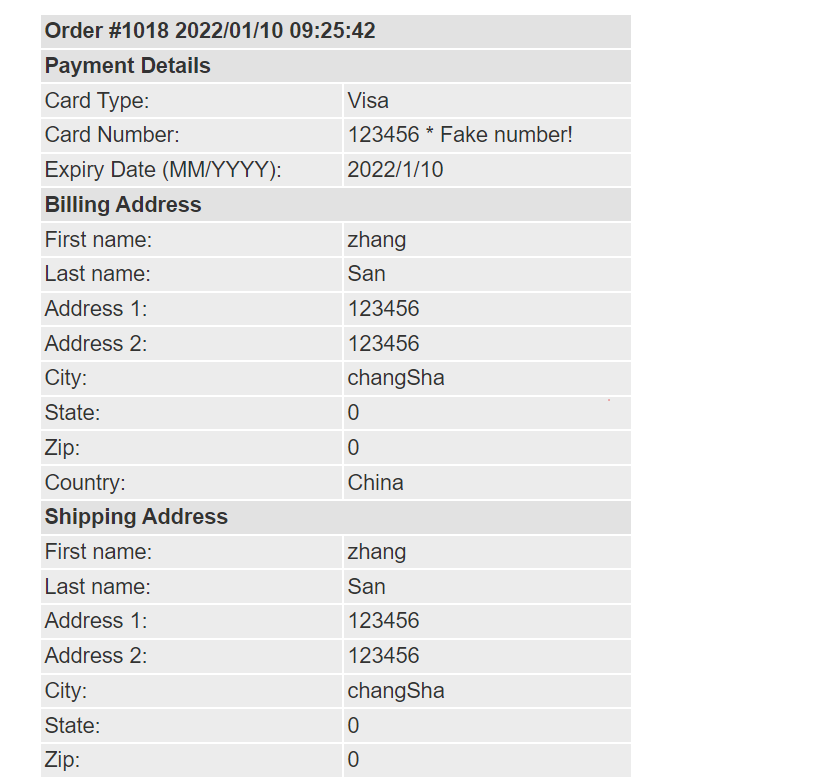
-支付订单：



（2）新增订单：s



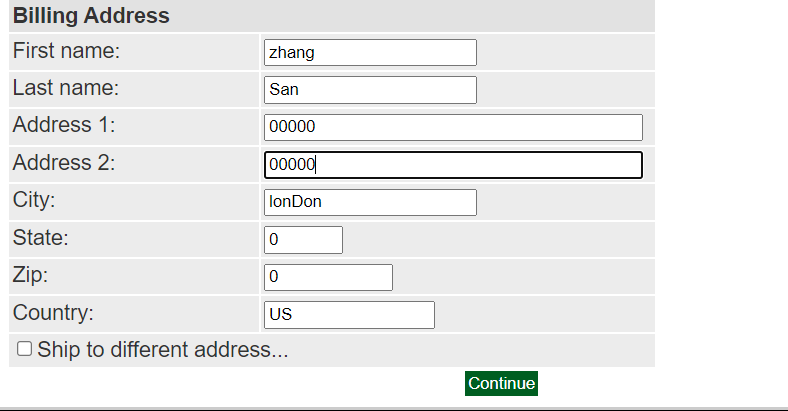
订单信息：



（3）地址信息：



修改地址：



修改地址成功，订单信息发生改变：

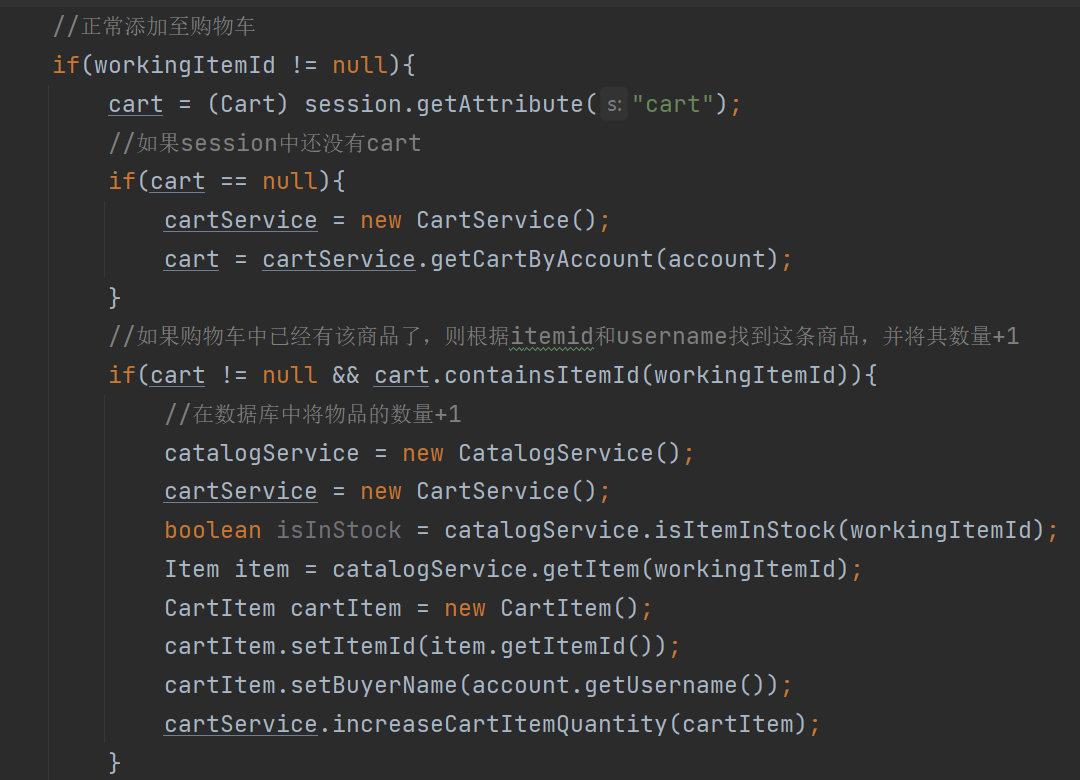


2、核心代码：

（1）、购物车添加、移除商品Servlet和JSP：

Servlet：

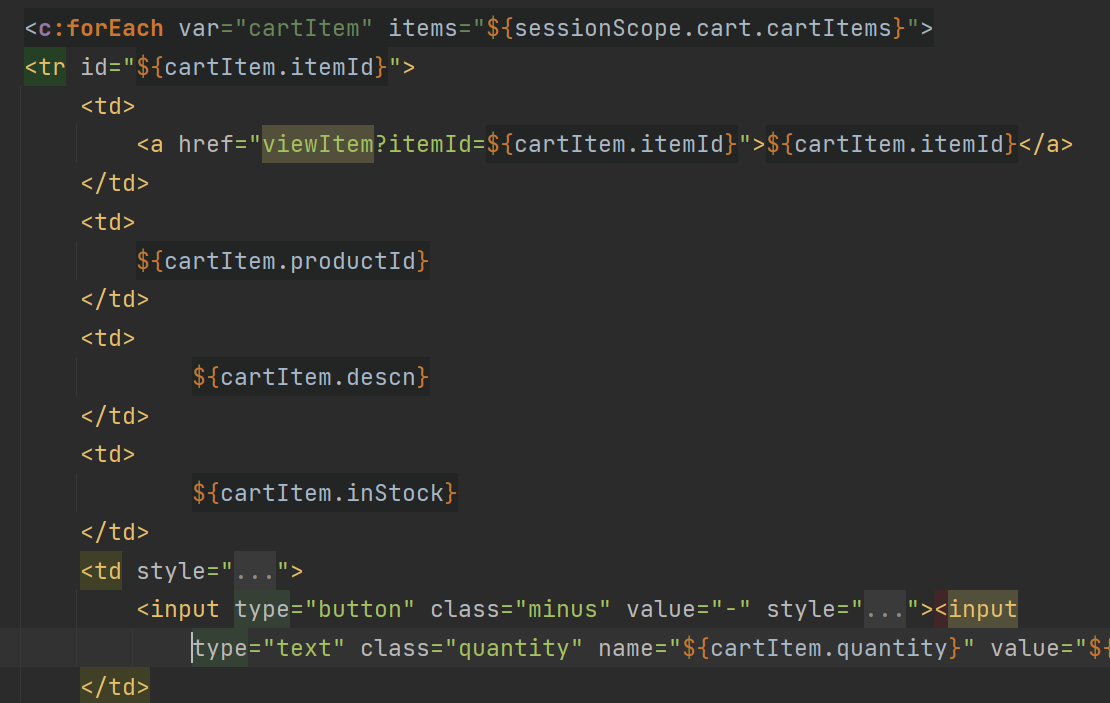


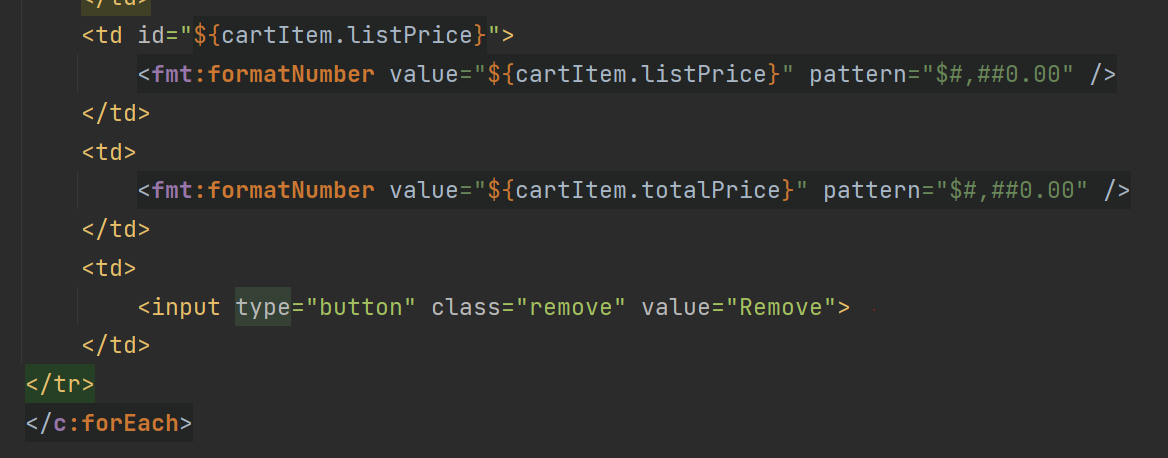






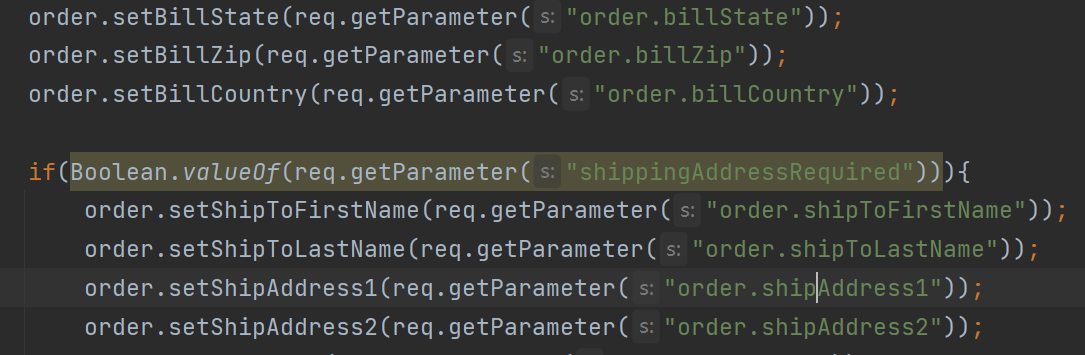
JSP：





（2）新增订单的Servlet：

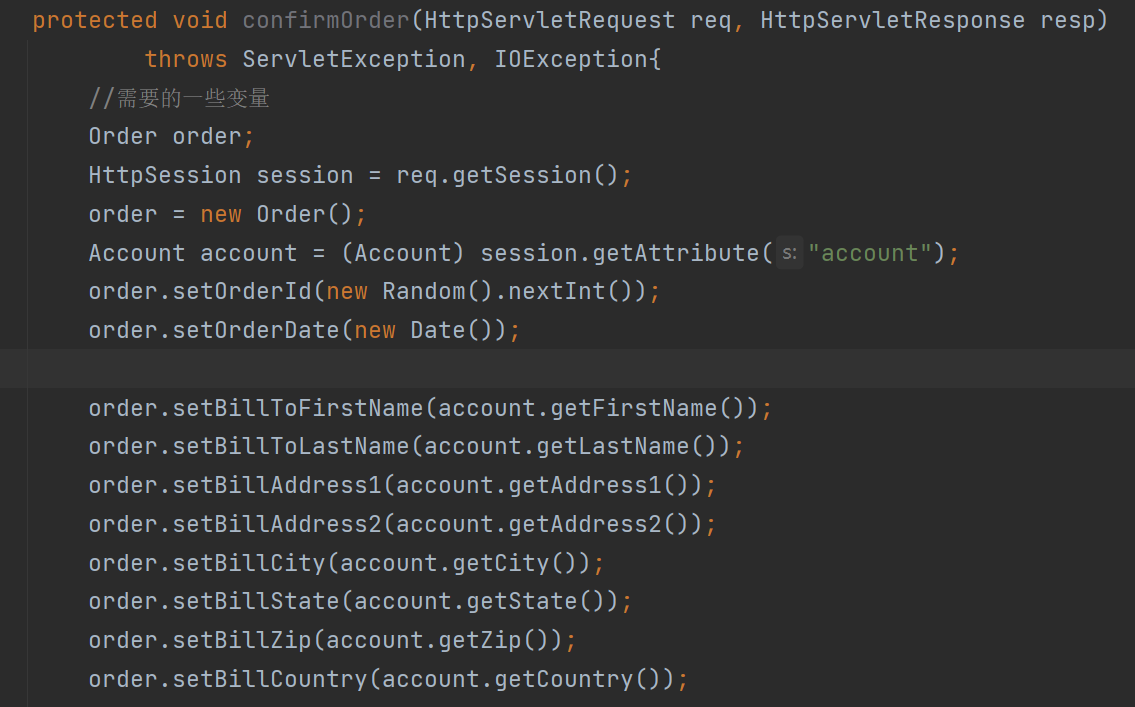


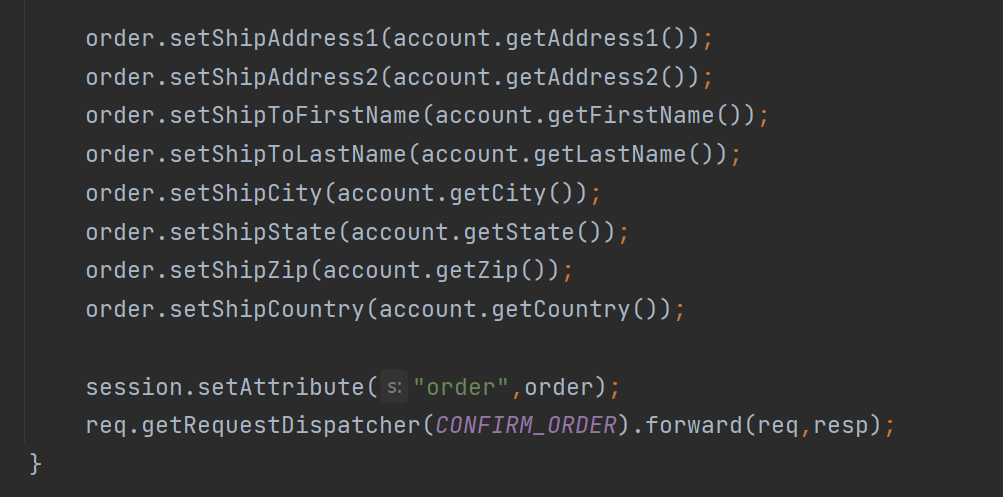






(3)地址信息的的Servlet（包含在确认订单表单之中）：





三、实验总结：

**分工：**

**（1）****商品展示业务模块；**

**（2）用户管理业务模块**

**（3）订单管理模块**

**（4）参与后端数据库的开发**

通过本次实验，我学到了许多东西。实验过程很艰难，但收获的汇报却是相当喜人。

遇到的一些问题：

（1）路径问题：

在开发中，尤其是图片显示时多次遇到无法找到资源，因此我在CSDN上专门查询WEB路径的表示方式，较好的解决了这个问题。

（2）调试问题：

本次实验成功的“逼”我学会了debug，以往开发代码遇到错误都是自己捋顺，全靠脑子，但这次团队开发成功让我意识到人的能力是有限的，单凭记忆极易出错，于是我学会了动态调试代码，从此开发能力又上一阶层。

（3）GET、POST与SESSION：

本次实验中界面多次跳转，界面间的信息传递成为重中之重，使用url进行传值变为不可能，于是我使用了Session。期间出现多次关键字输入错误（因为它本身不报错，只是取不出结果）。这让我意识到以后再开发的过程中遇到像这样的情况不能仅凭借脑子记，而应当用电脑记录下来。

一些体会

（1）我切实的体会到了按照MVC框架开发的好处，并且对其应用也愈发娴熟。

通过时使用MVC结构进行开发，不仅能使代码结构变得更加有条理，精简形式，更重要的使开发人员分工更加明确，可实现前后端分离，仅凭借接口进行交互，大大减少交接工作时的错误。

（2）写项目时应敢于并善于使用新学到的只是，这点我们组的组长郑乾坤非常值得我学习，他在开发项目的过程中自学Java的反射机制，并以此对Servlet文件进行整合，大大精简了项目结构，让我们组的工作变得更易展开。

# 实验二

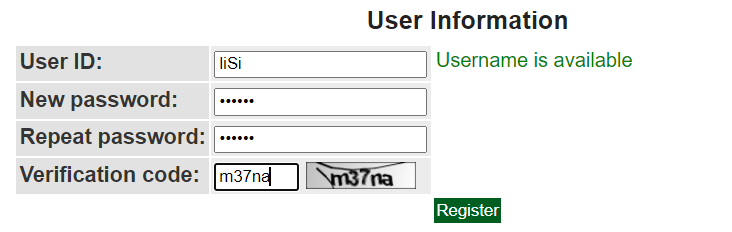
1. **实验内容——基本任务。**

**1、账号管理模块。在新建账号和修改账号信息页面上用AJAX技术进行表单验证，如“判断用户名是否已存在”等功能。**

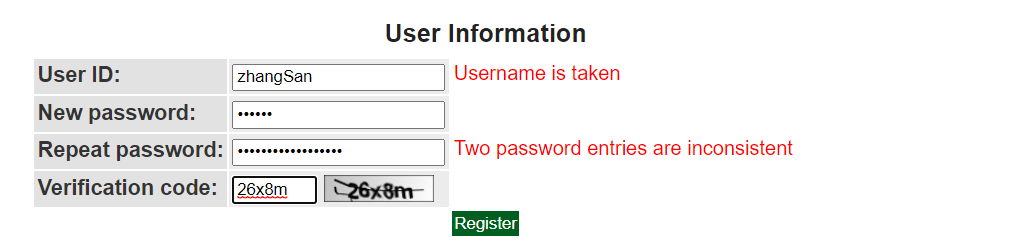
（1）运行界面：

若用户名没有被注册过，则会显示可用；

若两次密码输入一致，则不会显示错误信息。

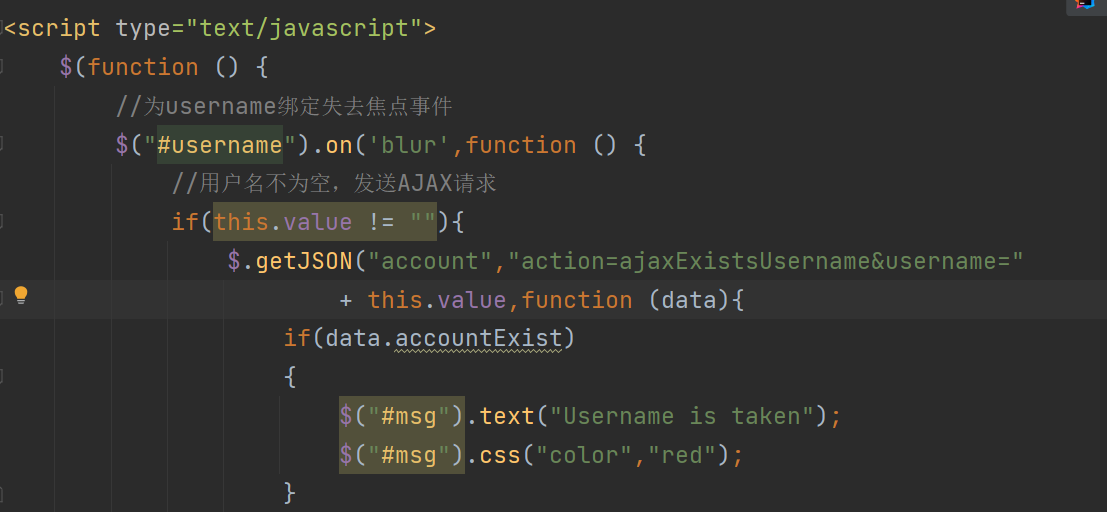


反之则会显示红色报错信息：

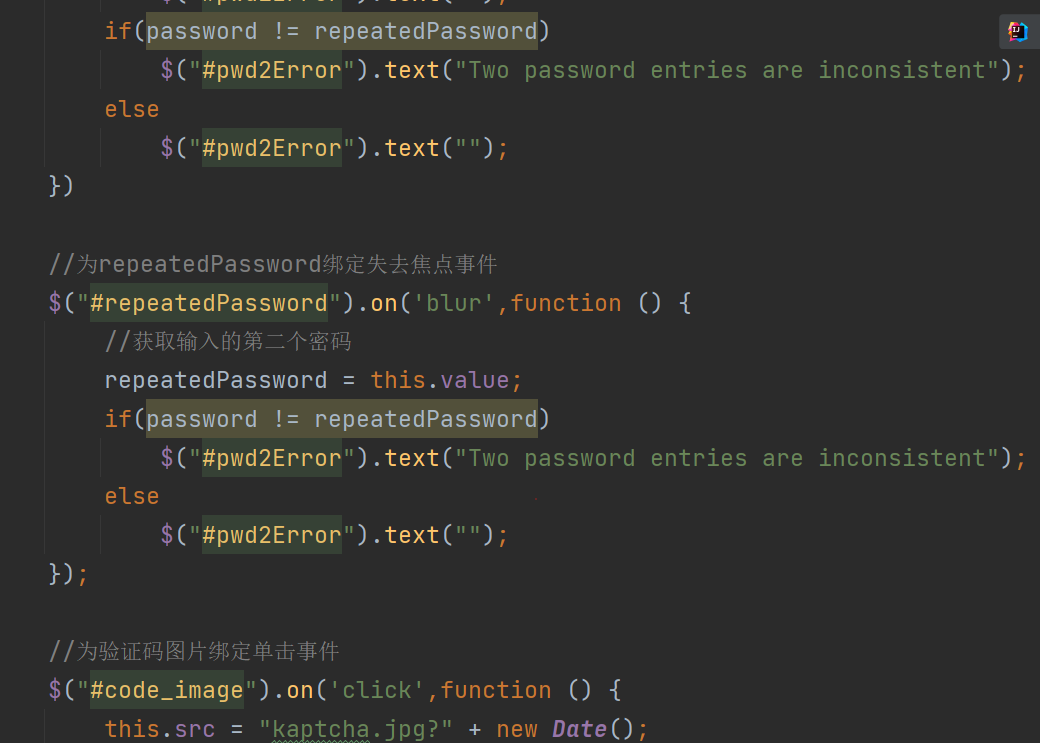


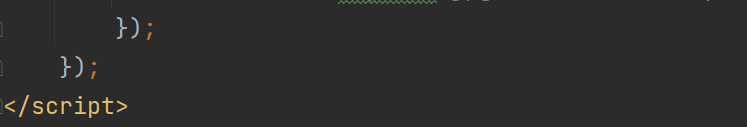
（2）核心代码：

JS部分：

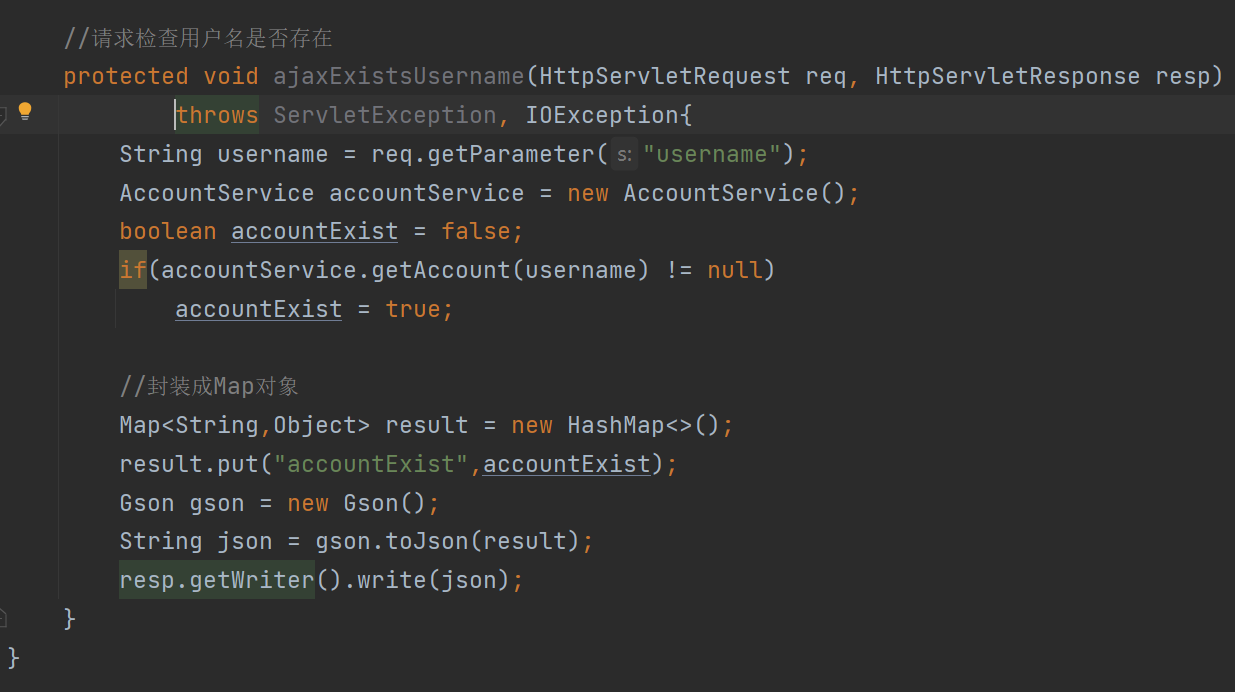








Servlet部分：

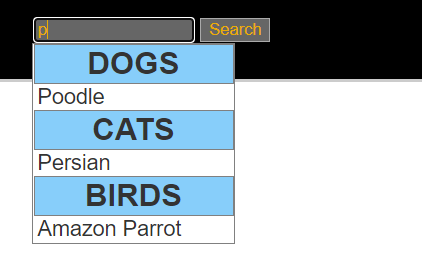


**2、查询商品模块。在主页面右上角的查询商品功能中添加自动补全功能。**

1、运行界面



查询出来的商品会自动根据商品所属的Category进行分类，从而呈现出下图的样子，蓝色背景加粗字体为大类，不可点击；白色背景为商品，可点击



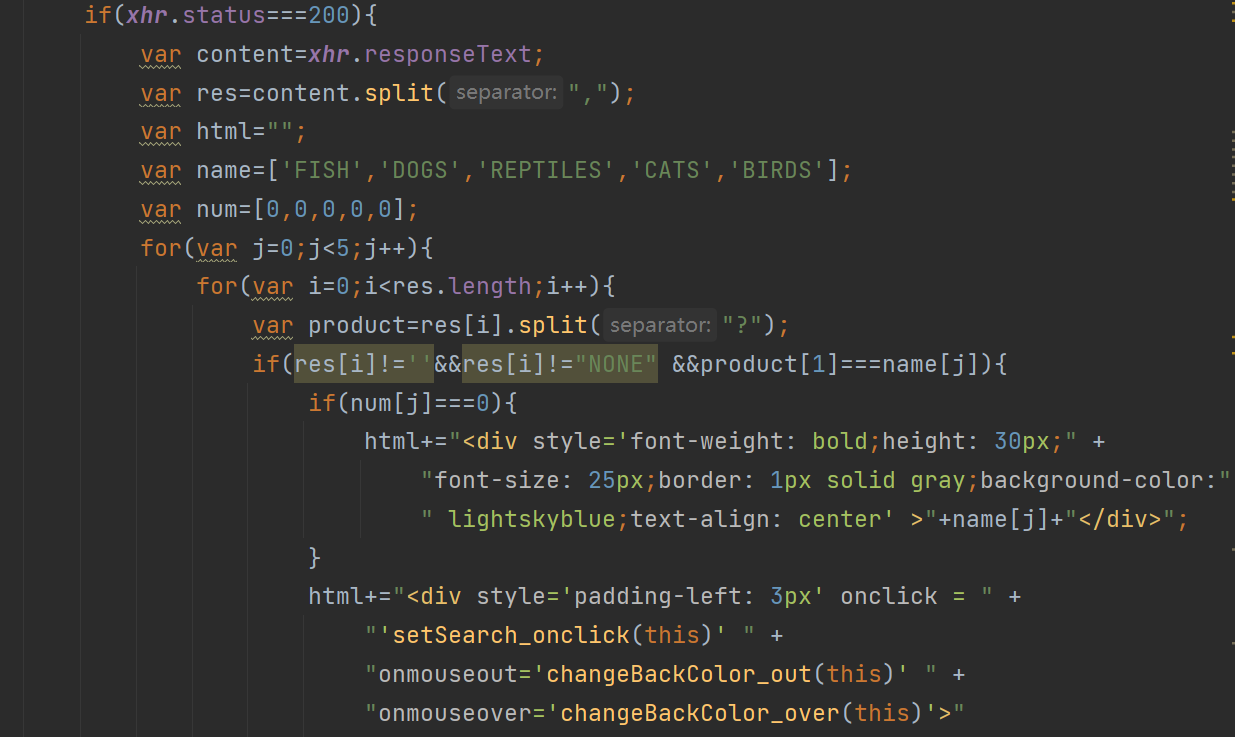
点击后跳转至商品信息展示界面（此处点击即Poodle）：



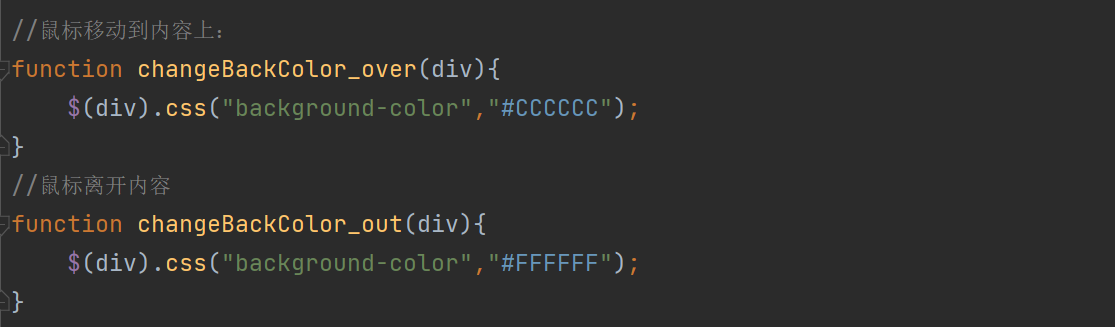
跳转成功

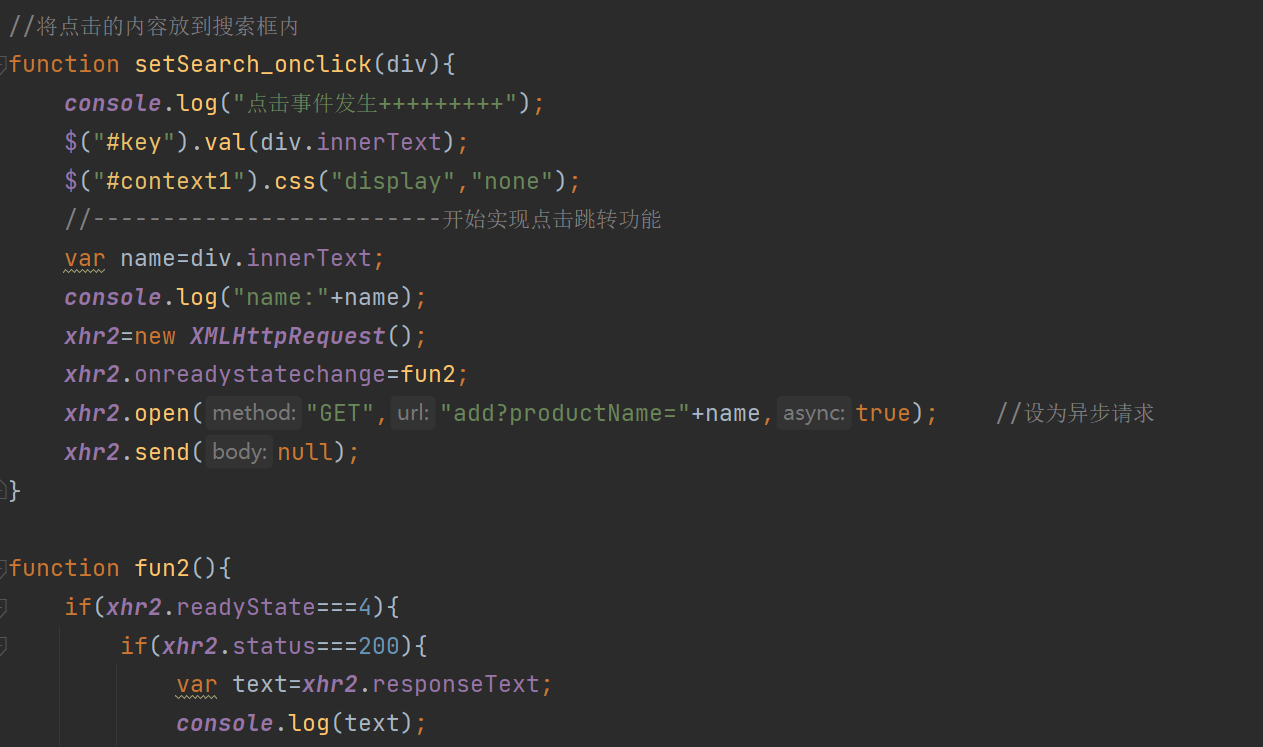
2、代码展示：

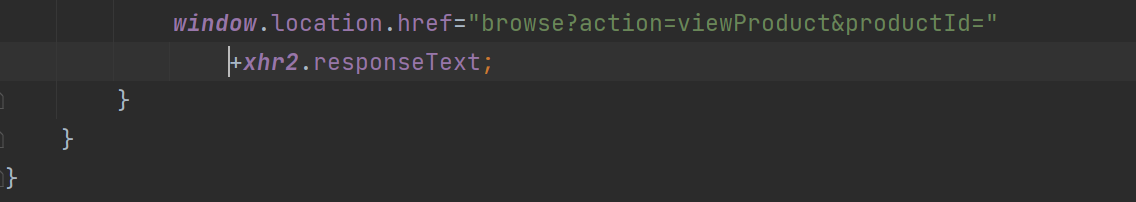
JS：  







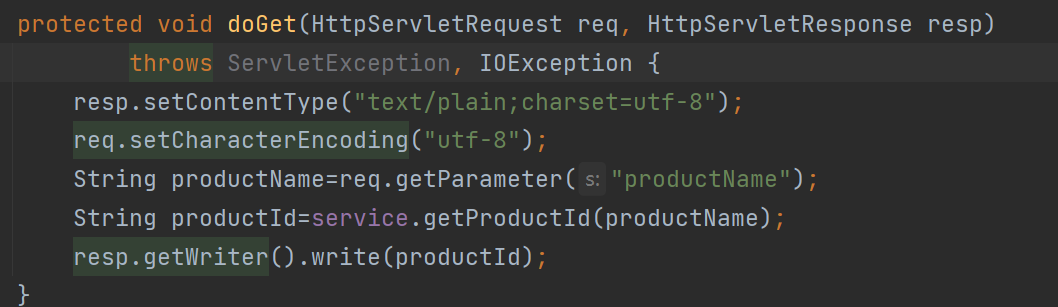


Servlet：

自动补全部分



点击下拉列表内容，页面跳转部分



# 小组分工及实验总结：

1、李文博为组员，负责：

**（1）账号管理模块**

**（2）查询商品模块**

**（3）参与css的设计**

2、实验总结：

本次实验虽然过程比较艰辛，但收获是喜人的，我的前端水平有了显著性提升，以下本次实验过程中遇到的一些问题以及一些新的：

一些问题

1、JS代码的同步问题。

本次实验中出现了JS代码已修改，但浏览器显示不变的情况。经调试发现，浏览器内的JS代码并没有发生变动，采取ctrl+f5刷新浏览器缓存，重新加载JS代码，问题解决。这让我认识到了JS的特性：脚本语言，写完就是镶嵌在浏览器里，因此必须注意代码的效率以及同步问题。

2、JS代码引入JSP（仍未确切掌握原因）：

在外部写完JS引入JSP文件，引入语句位置写在头部，但出现了一个JS能读取，另外一个404的现象，该问题的原因我到现在还没有弄清楚。但将报404的JS文件引入语句写在JSP文件下方后，JS正常读取。问题虽然解决，但疑惑没能解决，后续我将继续研究。

3、布局的相对位置问题：

搜索框自动补全，意味着要在原有JSP布局内加入新的布局。使用相对布局时导致了整个界面发生了拉伸。由于时间问题，最后我使用了相对简单的绝对布局，并将下拉框的z-index值设的极大，用来防覆盖。但这并不是一种很好的方法，仅是应急之策，后续我将继续改进。

一些心得：

1、JQuery的优越性：使用JS有时要做许多判断，JQuery包含的“语法糖”可以帮助我们精简代码。同时使用JQuery UI 使界面的设计变得更加简单，这我们减少工作量，后续我将继续练习JQuery。

2、本次实验暴漏了我许多不足，尤其是在同步与异步的处理上。如：为了实现点击自动补全的下拉框的同时实现界面跳转，我想当然的在JS里使用了同步操作。但后续的学习让我明白，异步也可以实现该操作。