**第26次作业**

# 需求分析

## 项目概述

## 目标

该项目用于实现对easybuy\_user表的用户查询，用户更新，查询结果分页，更新置于事务管理中，连接库用连接池和线程池等操作。

### 范围

需求范围：可以对数据库表中各种数据进行各种操作。

功能描述：

1. 可以实现按用户id查询，按用户姓名查询：可模糊查询以及精准查询；
2. 可以实现对用户的更新操作，例如增加新用户，删除表中已有用户，修改表中用户的信息；
3. 对用户查询结果进行分页显示。

### 运行环境

数据库系统：MySql数据库管理系统

开发平台及工具：Eclipse MySql

jsp页面显示：FireFox Chrome

模式：B/S模式

# 功能需求

## 查询功能

（1）普通查询：即输入用户id来查询用户的所有信息；

（2）模糊查询：即输入用户姓名中的个别字即可找到包含该字的所有用户的信息，例如：输入用户“李”，即可查找到小李，大李，李白这样包含李的用户；模糊查询中也包含精准查询，输入用户的全名称，即可查询指定用户的信息；

（3）全部查询：即查询数据库指定表的全部用户信息。

## 更新功能

增加新用户：通过输入新的id、loginName、password、sex、identityCode、email、mobile、type等值添加新的用户。

删除表中存在的用户：有两种方式删除，一是通过输入想要删除的用户的id,而是通过输入想要删除的用户的姓名。

修改用户信息：通过输入需要修改用户信息的用户id定位到该用户。

## 分页功能

分页功能：对于全部查询到的用户进行分页查看，例如可以分为一页显示3个用户，或者5个用户等。

### 性能需求

操作界面：要求简洁明了，布局合理，对于用户来说简单易操作。

安全保密性：只有输入正确的用户id以及name和密码才能进行相应的界面操作。

扩展性需求：可以根据用户实际需要，在不修改源代码的基础上进行功能增加。

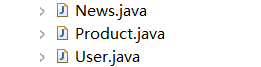
信息的完整性：输入的新信息应该是合法的，如果输入信息不符合系统要求，应该给出警告。

稳定性：在一般情况下，项目是能够正常运行的，为了避免单一界面信息量爆炸的情况，采用分页查看是十分重要的。

数据处理能力：能够快速响应用户请求，并且运行效果稳定。

# 模块划分

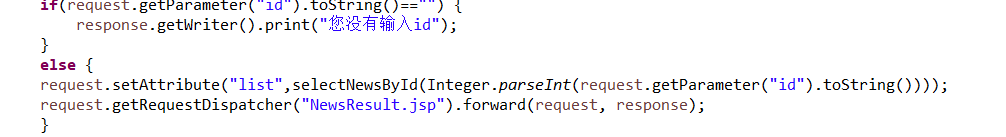
首先我们先定义好了所有可需要的Class，以便后面创建他们的对象。



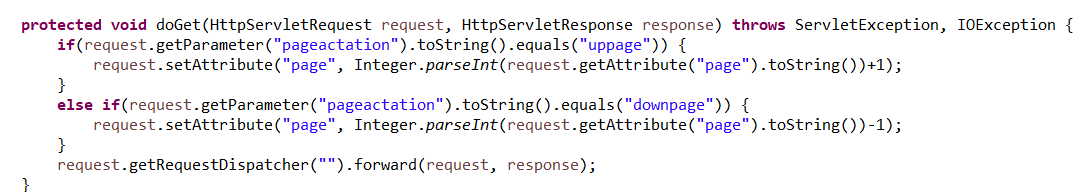
为了简单的连接数据库，我们创建了一个数据库连接，关闭和释放的静态类，使别的类在连接数据库的时候，就可以直接的使用数据库连接的类名进行连接，方便，利于编程。



我们在Servlet包里面，写了一堆关于数据库操作的Servlet，这些Servlet不紧有着增删改查数据的功能，还有将查询出来的信息，放到公共域里面，然后跳转到指定的JSP页面，使后面的JSP页面可以获取和显示这些内容。



然后Contronll是JSP和Servlet通信的中间路径，他能将JSP页面获得的信息传送给相应的Servlet来进行响应操作，使Servlet去执行任务，然后取得的数据又返回给JSP页面。



为了编码方便，我们使用了对Contronll的过滤器，使所有的Contronll在执行前都必须经过过滤器进行过滤，使其编码变为统一的utf-8字符。

# 操作设计

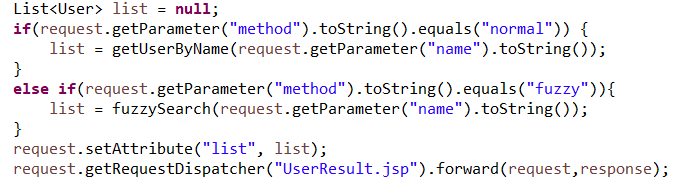
## 查找操作设计：

主要是进行用户的查找，通过不同的方式(id/用户名)在不同的条件(模糊/精准)下进行查找。

查找功能实现主要由五部分组成：用户进行输入的input界面，进行转发的Controller，进行数据库操作的NormorSelect和SexSelect ，以及用户的类User和显示结果的result界面组成。

### 按姓名查找：

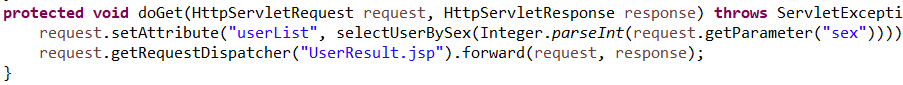
数据在Input界面输入之后，在Controller界面进行转发的判断为，Controller就将request转发到NormalSelect，在NormalSelect判断客户需要模糊查找还是精确查找。



查找完成后将获得到的list转发到UserResult.jsp进行显示。

### 按性别查找：

用户在点击Input界面的按钮并选择好了自己所需要的性别之后，数据转发到Controller界面，之后Controller在判断之后将数据转发到SexSelect。在SexSelect中进行查找操作并转发到最后的结果界面。



## 新建操作设计

在实际使用中的时候，新建一个东西是非常常见的，比如用户的注册，商品的添加等。

新建操作的实现主要包括四部分进行输入的Input界面，进行转发的Controller界面，进行数据库操作的AddUser.java,以及用户的类User。

用户在Input输入界面之后，通过request转发到AddUser之后，用一个User来保存输入的数据，之后通过一个新的User保存数据，并添加到数据库中。

## 删除操作设计

删除操作的实现主要包括两个模块：

一个是搜索模块，查询所要删除的用户。

另一个是删除模块，由进行转发的Controller界面，进行数据库操作的DeleteUser.java组成。

首先在查询页面中查询到自己想要搜索的用户信息。之后点击对应位置上的删除键将信息发送到Controller，然后再在DeleteUser中从数据库中找到要删除的信息，进行删除。

## 修改操作设计

删除操作的实现主要包括两个模块：

一个是搜索模块，查询所要修改的用户。

另一个是修改模块，由Controller以及ChangeUser.java来实现。

在查询界面查询到所想要想要修改的信息之后，将修改之后的信息传入ChangeUser，之后，在ChangeUser中对数据库中对应的信息执行更新操作。

# 具体实现

## form层

对所有需要使用到的数据库中存在的表格对象属性进行用于接收和显示的设置。

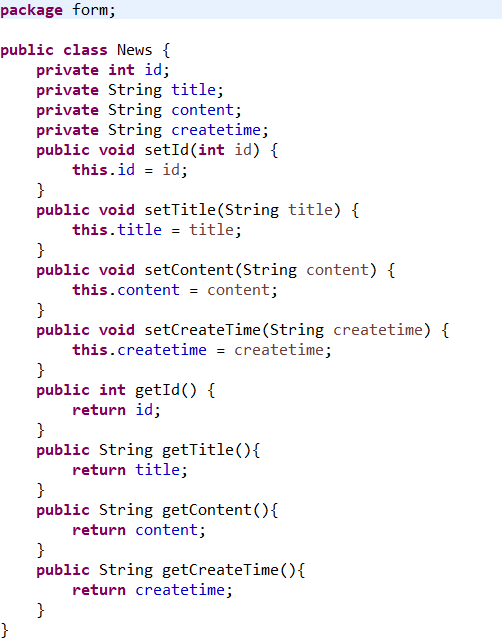


图1.News属性设置

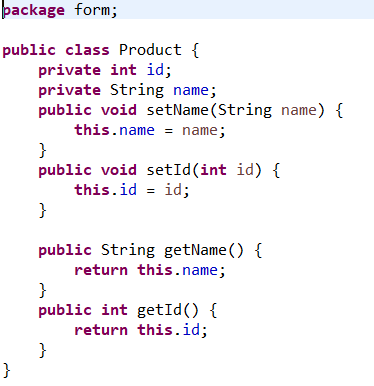


图2.Product属性设置



图3.User属性设置

## dbUtils层

进行数据库的连接操作

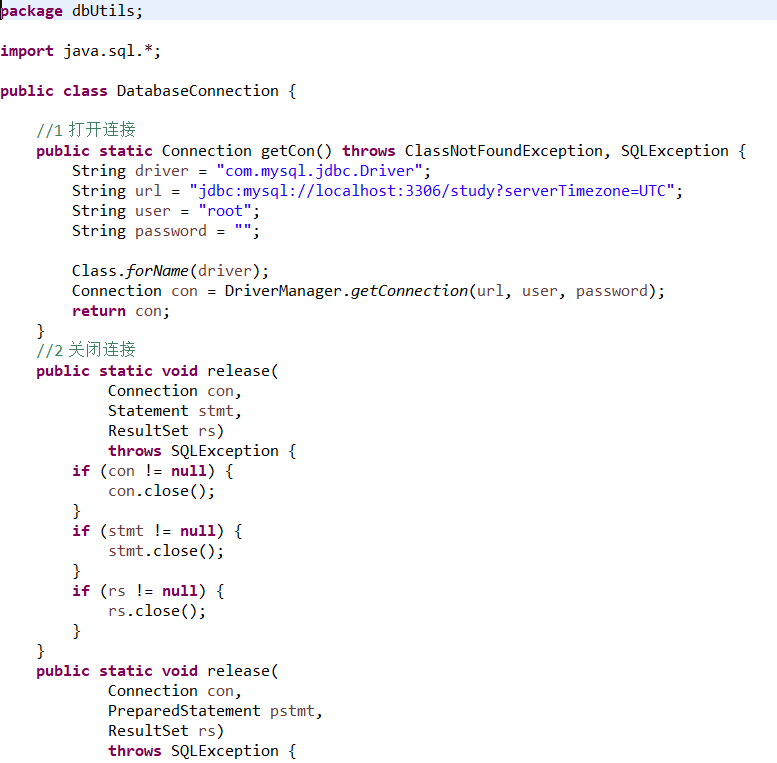


图4.进行数据库连接

## Controll层：

用于响应View层的firstPage.jsp页面的不同查询已经进行响应查询实现方法的调用。

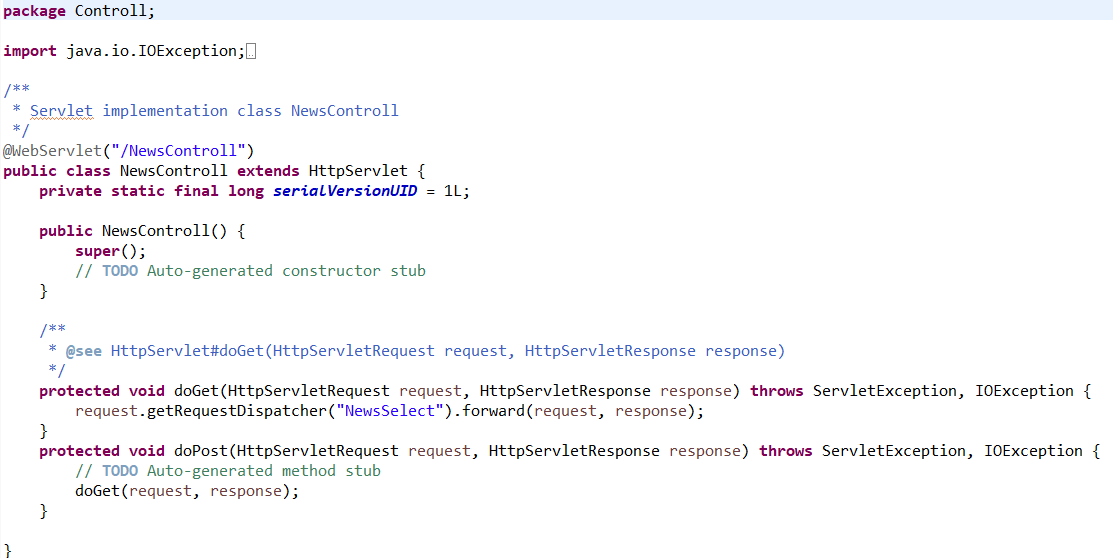


图5.响应News查询并跳转



图6.响应翻页



图7. 响应Product查询并跳转

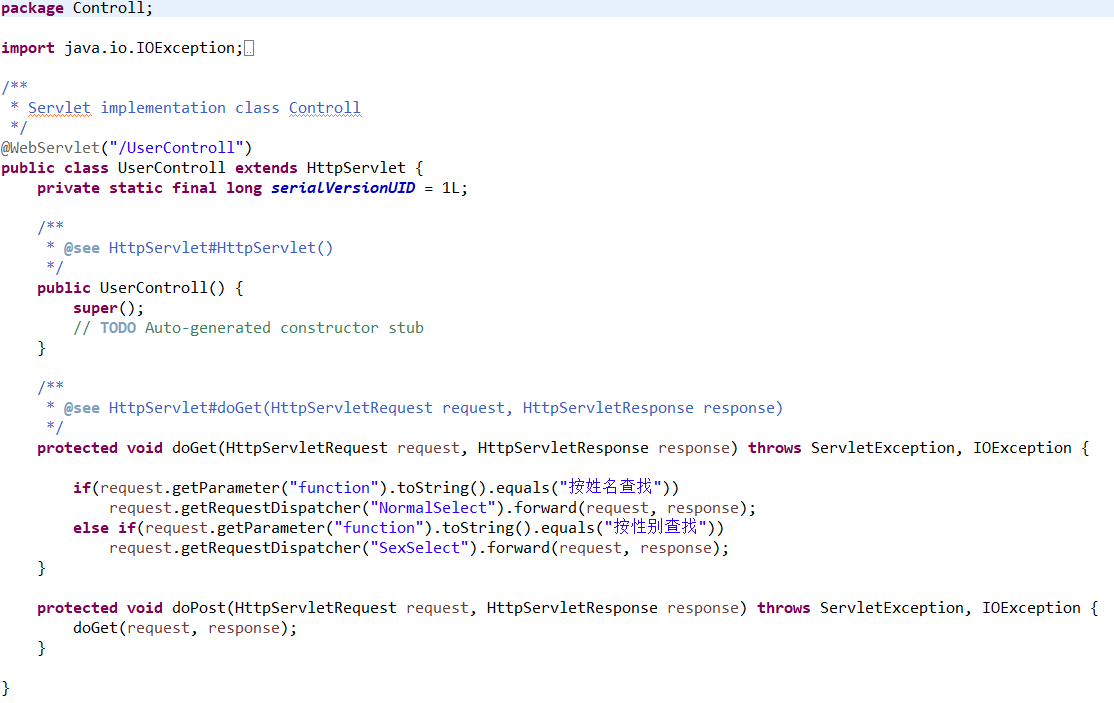


图8. 响应User查询并跳转

## servlet层

对不同的查询操作进行实现。



图9. 实现News查询



图10. 实现根据用户名精确查询User



图11. 实现根据用户名模糊查询User



图12. 实现根据性别查询User



图13. 实现Product查询

## filter层

对输入输出乱码问题进行解决。

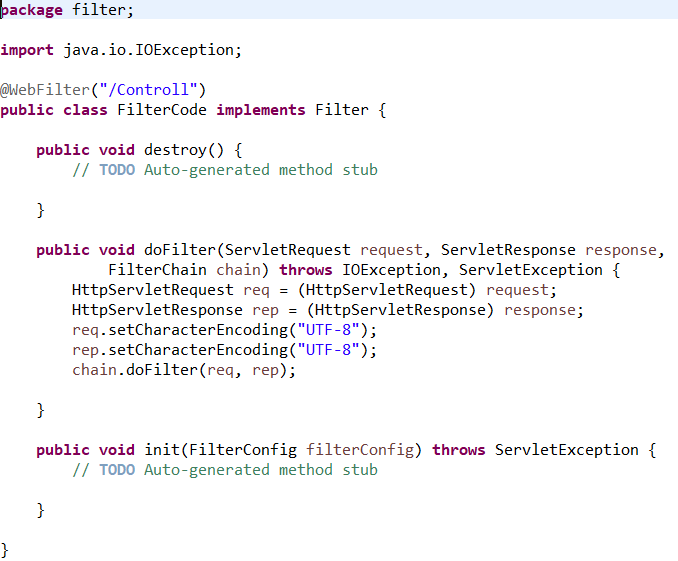


图14.对输入输出乱码问题进行解决

## View层（jsp页面）

用于进行查询的交互，引导用户输入想要查询的内容。



图15.按需求输入要查询对象的界面



图16.显示News查询结果的界面



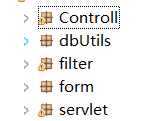
图17.显示Product查询结果的界面



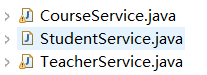
图18.显示User查询结果的界面

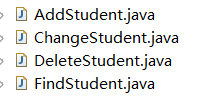
# 心得小结

在进行实验的过程中，如何能够更好的解耦合成为了一个关键，因为如果将所有的代码全部写在一个类中，就会因为要考虑的东西越来越多而使得代码的修改变成几乎不可能的一件事情。而在解耦合之后代码看起来就简单而明了许多



在这次的实验中由于任务量比较大，命名也就成为了一个问题。之前并没有考虑过命名的问题，因为之前的代码即使自己命名比较随心所欲，在需要的时候也可以比较好的进行理解，因为都是自己写的东西，并且时间比较短。所以之前我的命名会是：

之类的，虽然知道大概是什么东西，但也只是能够知道个大概。

在这次的实验中，我遵循了见名知意的原则，将名称改成了如：之类的，这样一看就可以知道这个代码的功能。

这次的实验主要的练习方面我认为不在代码，而在于整体上对于程序的规划以及框架有很大的提高。