Javaweb-项目

# 功能实现

## p2p\_home功能实现

### 用户管理-注册操作

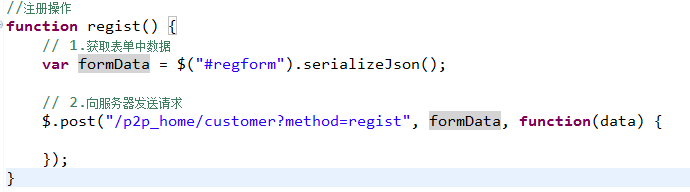
注册操作它不仅是添加客户信息，还要在注册时，为客户生成一个帐户。简单说，我们在进行客户注册操作时，要insert两条记录,一条是customer表中添加数据，代表的是客户信息，还需要向account表中插入数据，当创建一个客户时，也会为其生成一条帐户信息。

（本项中，没有使用支付宝，微信操作）去进行转账操作，只是简单的自己生成了帐户信息。

#### 步骤1:修改register.html页面，当点击注册按钮时向服务器发送请求

1.修改手机号为邮箱地址

2.给注册按钮添加事件，点击它完成向服务器发送请求操作



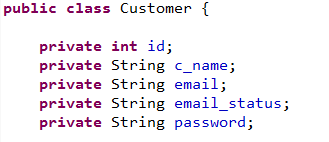
#### 步骤2:服务器端完成注册操作

注意：注册时要向customer与account表中插入数据，我们需要进行事务控制。

Servlet创建

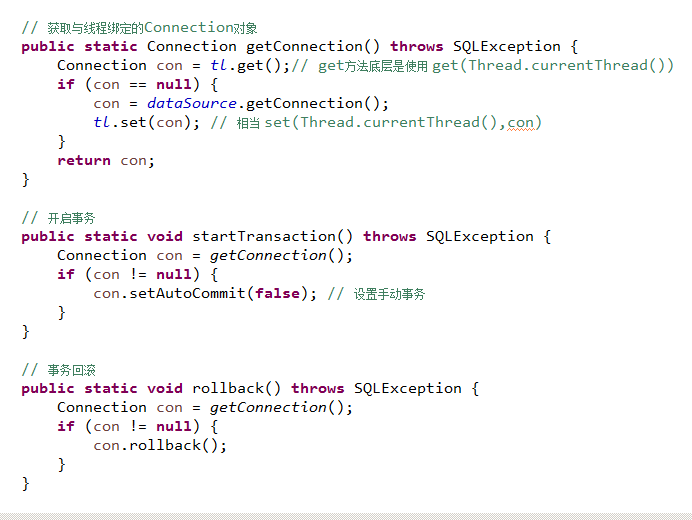


Domain



#### 步骤3:完成添加客户操作

关于JdbcUtils工具



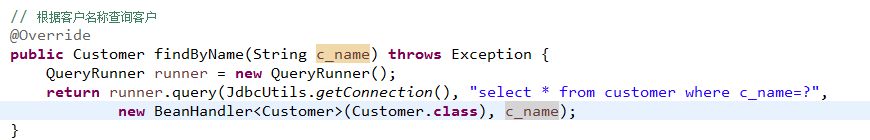
Service操作



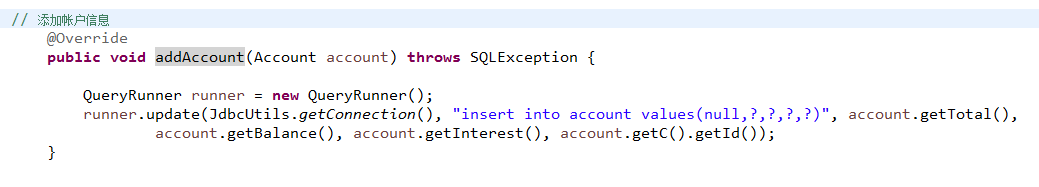
Dao操作

#### 步骤4:完成添加帐户操作

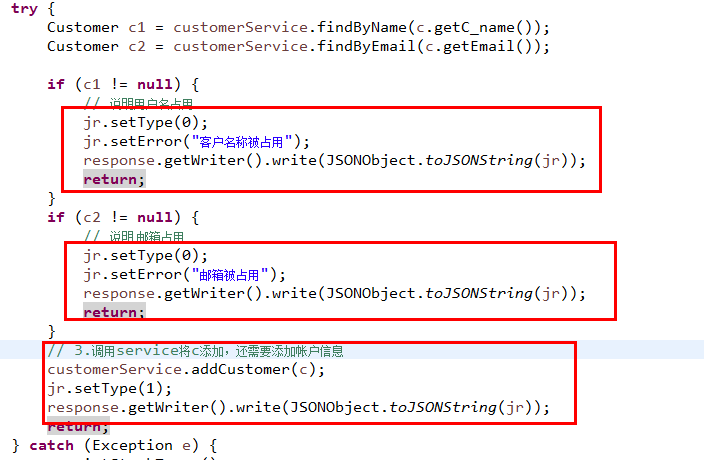
注意：添加客户时要保证它的email与c\_name是唯一的。



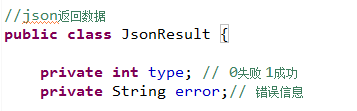
添加帐户信息

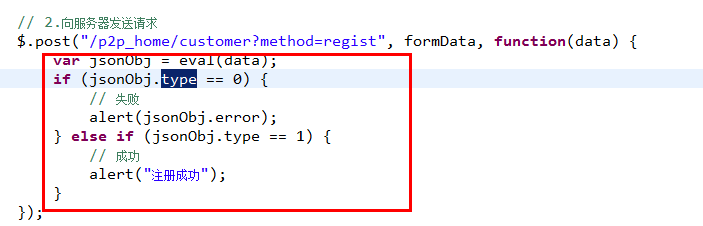


#### 步骤5:判断客户名称与邮箱唯一 ，并响应数据到浏览器



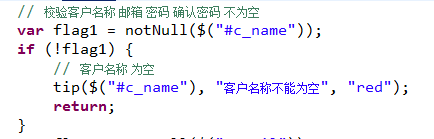
为了方便在浏览器端判断是否成功，我们可以设置一个json类型的返回结果。

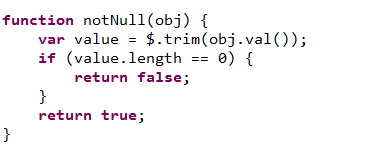




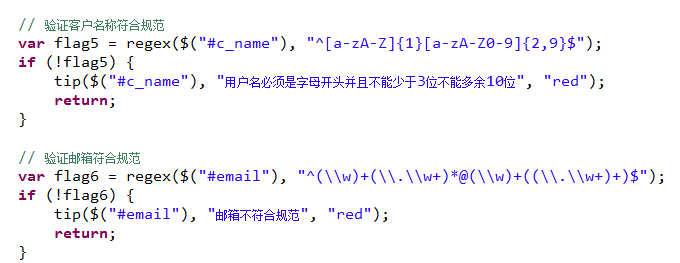
#### 步骤6:解决前端js校验

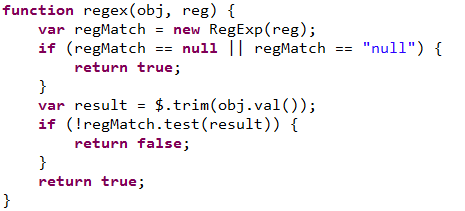
非空校验





规则校验



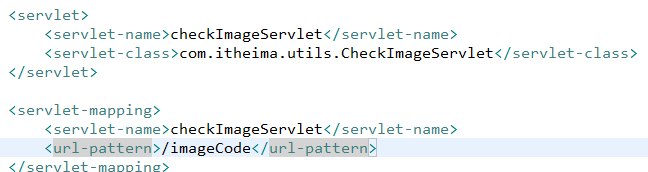


### 用户管理-登录操作

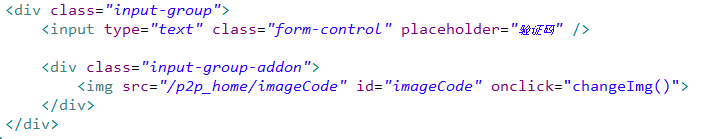
注意：在登录时，我们录制客户名称或邮箱地址都可以完成登录。

#### 步骤1：在登录页面上显示图片验证码

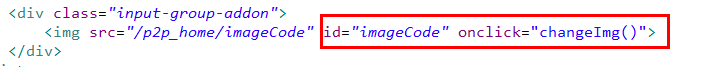
将资料中的CheckImageServlet复制到utils包下在web.xml文件中配置

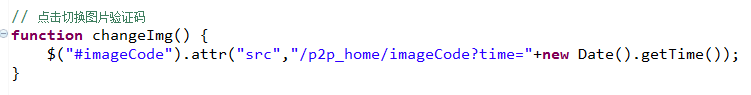


在login.html页面上添加



点击图片验证码切换图片

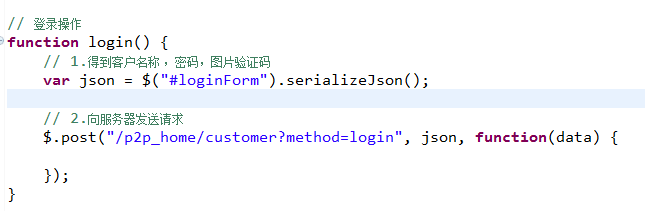




#### 步骤2：完成点击立即登录操作，将信息发送到服务器端

注意:在页面上需要给用户名密码验证码添加name属性



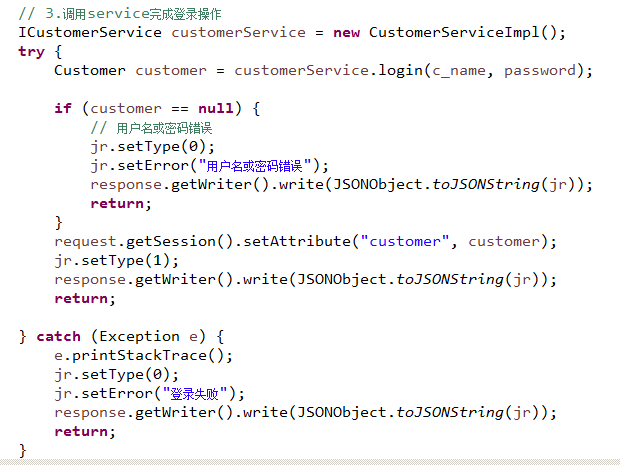


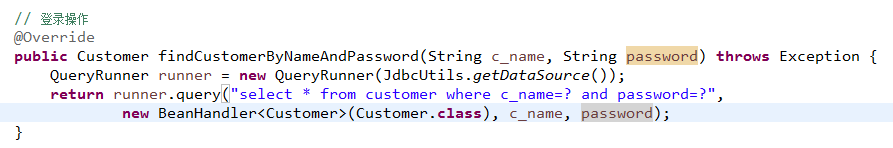
#### 步骤3：在服务器端完成登录操作，并响应数据到浏览器端

获取请求参数，判断验证码是否正确

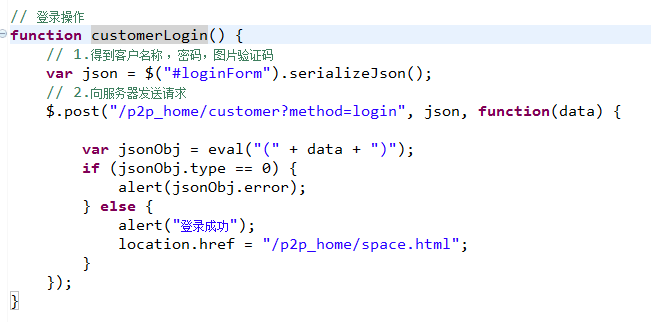


调用service完成登录操作





在浏览器端处理服务器的响应数据



注意:在login.html页面上我们进行了修改



#### 步骤4：完成邮箱登录

在登录程序中添加一段来判断是否是邮箱



#### 步骤5:在服务器端进行校验



#### 步骤6:关于密码md5加密问题

Java加密

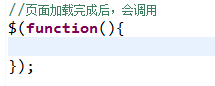


在mysql中对密码进行加密

UPDATE customer SET PASSWORD=MD5('123');

#### 步骤7 登录后，跳转到space.html页面加载客户帐户信息

给space.html页面中导入一个space.js文件，在space.js文件中

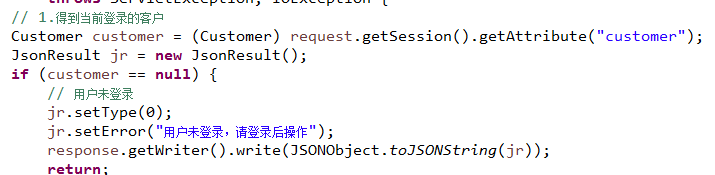


这个函数是页面加载完成后执行。我们就可以在函数中向服务器发送请求，来获取客户的帐户信息在页面上显示

1.向服务器发送请求



2.在服务器端判断客户是否登录



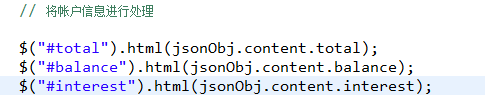
3.客户登录，查询客户的帐户信息



注意：在JsonResult中双封装了一个content信息，它主要是用于封装查询的对象，例如Account对象。

4.在页面上处理帐户信息显示

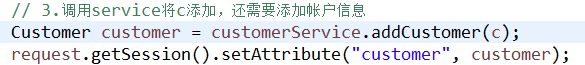
首先在space.html页面上添加了id,在js代码中进行了处理

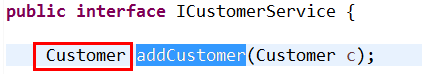


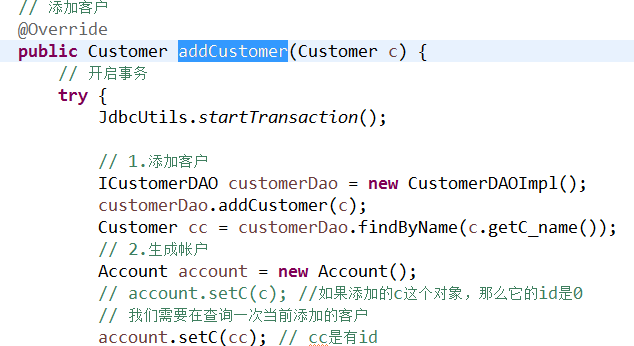
### 用户管理-注册操作bug修复

当注册成功后，要将注册的客户存储到session中。

那么我们将添加客户操作设定成有返回值，返回的就是当前添加的Customer对象。







关于注册时，我们考虑事务问题，我们创建了一个JdbcUtils工具。



报错的原因就是我们在使用JdbcUtils获取的Connection对象它是绑定到ThreadLocal中，当我们使用完，并将Connection关闭后，我们应该将Connection也从ThreadLocal中移除。

