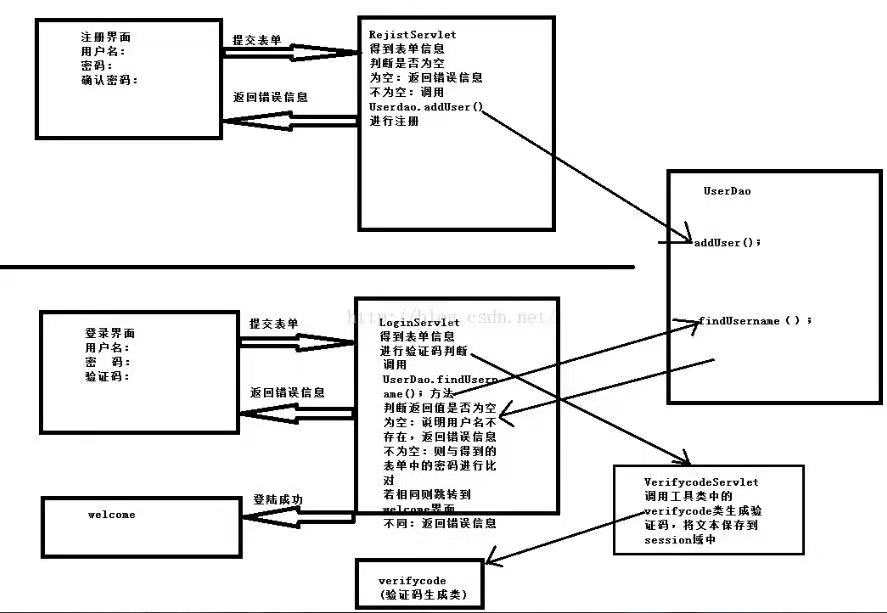
11月11日

建模回复：



数据库回复：

学习了，超认真

网络通信回复：

...就，莫得进展

程序员回复：

我今天学习了界面设计，初步了解了

11月12日

建模回复：

数据表多对多关联

（点赞功能实现）

借助第三张表中间表middle来进行查询

中间表middle有如下字段 id，picture\_id，product\_id

两个需要多对多关联的表为picture，product表

这样product表只需查询中间表middle就能找所有指定product\_id的picture\_id

再通过查询到picture\_id 去找到指定图片。

同样要知道一张图片属于哪些商品

只需在middle表找到和这张图片的picture\_id相等的所有记录，然后获取这些记录的product\_id，

再通过这些product\_id就能找到这张图片被哪些商品共有

数据库回复：

经过进一步的学习

我感觉

我的知识储备

更近一步了

网络通信回复：

我感觉很难，但还是有那么一点点进展

程序员回复：

我今天复习了可视化，还在准备

11月13日

建模回复：

1、send() 向远程服务器发送数据

　　　　2、close() 关闭该websocket链接

　　websocket同时还定义了几个监听函数

　　　　1、onopen 当网络连接建立时触发该事件

　　　　2、onerror 当网络发生错误时触发该事件

　　　　3、onclose 当websocket被关闭时触发该事件

　　　　4、onmessage 当websocket接收到服务器发来的消息的时触发的事件，也是通信中最重要的一个监听事件。msg.data

　　websocket还定义了一个readyState属性，这个属性可以返回websocket所处的状态：

　　　　1、CONNECTING(0) websocket正尝试与服务器建立连接

　　　　2、OPEN(1) websocket与服务器已经建立连接

　　　　3、CLOSING(2) websocket正在关闭与服务器的连接

　　　　4、CLOSED(3) websocket已经关闭了与服务器的连接

消息推送以及socket通信

数据库回复：

初步了解SQL

网络通信回复：

通过学习，感觉自己升华了，就很厉害，懂不？

程序员回复：

我今天复习了监听器，又了解了一点，开始弄了

11月14日

建模回复：

1、在数据库表中新添加一列名为sort，初始值与各条数据的id相同，初始list排序按照sort的值排序。

2、思路是向上移动的时候，查询出当前数据的上一条数据的sort值，两者交换sort值，同理向下移动的时候，查询出下一条数据的sort值，两者交换。

3、ssm框架中的sql及代码实现

帖子列表中

帖子上下移动

置顶还没找到办法

数据库回复：

进展比较顺利，几个数据库已经分好

网络通信回复：

有一丢丢进度

程序员回复：

我今天复习了sql，看了聊天室的代码

11月15日

建模回复：

本周工作基本完成

数据库回复：

数据库主键等已经弄好，晚点发给你看

程序员回复：

明天要报告的内容基本弄好了

网络通信回复：

这周的差不多了吧...

11月25日

建模回复：

添加评论代码（获取到评论相关的参数，然后进行向数据库表插入）

数据库回复：

public class UserDao {  
   
    // 查询用户信息  
    public List<Users> selectAll() {  
        // 生成查询用户信息的SQL语句  
        String sql = "select \* from ceshi2";  
        // 创建加载连接数据库的对象  
        DbTools db = new DbTools();  
        // 创建Connection对象  
        Connection conn = db.getConn();  
        // 创建一个集合  
        List<Users> ls = new ArrayList<Users>();  
        try {  
   
            Statement st = conn.createStatement();  
            // 执行查询语句  
            ResultSet rs = st.executeQuery(sql);  
            // 遍历结果集  
            while (rs.next()) {  
                // 判断是否是管理员  
                if (rs.getInt(4) != 1) {  
                    // 创建用户信息的实体类对象  
                    Users us = new Users();  
                    // 为id赋值  
                    [us.setId(rs.getInt(1));](https://wx2.qq.com/cgi-bin/mmwebwx-bin/webwxcheckurl?requrl=http%3A%2F%2Fus.setId(rs.getInt(1))%3B&skey=%40crypt_4c8b685_589d95a9c085a919ef73ffbef723dc12&deviceid=e854673648395497&pass_ticket=undefined&opcode=2&scene=1&username=@13fcde29fa2d412762cdab8c470ffa111c33a4bdc51bae2024d4e080170f359a)  
                    // 为用户名赋值  
                    us.setUsername(rs.getString(2));  
                    // 为用户密码赋值  
                    us.setPassword(rs.getString(3));  
                    // 为权限赋值  
                    us.setPopedom(rs.getString(4));  
                    // 把用户对象添加到集合中  
                    [ls.add(us);](https://wx2.qq.com/cgi-bin/mmwebwx-bin/webwxcheckurl?requrl=http%3A%2F%2Fls.add(us)%3B&skey=%40crypt_4c8b685_589d95a9c085a919ef73ffbef723dc12&deviceid=e854673648395497&pass_ticket=undefined&opcode=2&scene=1&username=@13fcde29fa2d412762cdab8c470ffa111c33a4bdc51bae2024d4e080170f359a)  
                }  
            }  
        } catch (SQLException e) {  
            e.printStackTrace();  
        }  
        // 返回集合  
        return ls;  
    }

程序员回复：

上网看了有关增删查改内容的代码

网络通信回复：

还行

11月26日

建模回复：

添加回复代码（前台会传来评论的ID，然后，封装成回复对象进行插入，一个评论ID会对应很多回复）

数据库回复：

用户登录  
   
   
    public int loginuser(String name, String pwd) {  
        // 生成SQL语句  
        String sql = "select popedom from ceshi2 where userid='" +id  
                + "' and password='" + pwd + "'";  
        DbTools db = new DbTools();  
        Connection conn = db.getConn();  
        try {  
            Statement st = conn.createStatement();  
            // 执行SQL语句  
            ResultSet rs = st.executeQuery(sql);  
            // 判断集合级是否有值  
            if (rs.next()) {  
                // 判断登录用户的权限值是否为1  
                if (rs.getInt(1) == 1) {  
                    // 1 表示的权限为管理员  
                    return 1;  
                } else {  
                    // 2 表示的权限为普通用户  
                    return 2;  
                }  
            }  
        } catch (SQLException e) {  
            e.printStackTrace();  
        }  
        // 3 表示无此用户  
        return 3;  
    }  
   
     
   
   
 用户注册  
         public void addUser(String name, String pwd, int id, String Umail , int Usex ) {  
        // 生成添加的SQL语句  
        String sql = "insert into ceshi2 (username,password,uid ,umail ,usex )values('" +name + "','" + pwd + "','" + id + "' ,'" + mail + "','" + sex + "')";  
        DbTools db = new DbTools();  
        Connection conn = db.getConn();  
        try {  
            Statement st = conn.createStatement();  
            // 执行SQL语句  
            st.executeUpdate(sql);  
            // 关闭Connection对象  
            [conn.close();](https://wx2.qq.com/cgi-bin/mmwebwx-bin/webwxcheckurl?requrl=http%3A%2F%2Fconn.close()%3B&skey=%40crypt_4c8b685_589d95a9c085a919ef73ffbef723dc12&deviceid=e854673648395497&pass_ticket=undefined&opcode=2&scene=1&username=@13fcde29fa2d412762cdab8c470ffa111c33a4bdc51bae2024d4e080170f359a)  
        } catch (SQLException e) {  
            e.printStackTrace();  
        }  
    }

程序员回复：

看了有关sql内容

网络通信回复：

还行

11月27日

建模回复：

思考topic热度问题，觉得可以通过优先级排列来实现热度功能

数据库回复：

我今天休息（表情）

程序员回复：

上网看了回复贴的代码内容

网络通信回复：

还行

11月28日

建模回复：

简单编写管理员类

数据库回复：

已完成本周任务

程序员回复：

回去看了上传头像部分的代码

网络通信回复：

还行

11月29日

建模回复：

摸鱼

程序员回复：

回去看了下管理员代码

网络通信回复：

还行