中南大學
CENTRAL SOUTH UNIVERSITY

JavaSocket 编程开发聊天室
JavaURL 爬取并分析网页敏感词

学生姓名	
学　　号	
专业班级	
指导教师	
学　　院	计算机学院
完成时间	2022.6.20

计算机学院

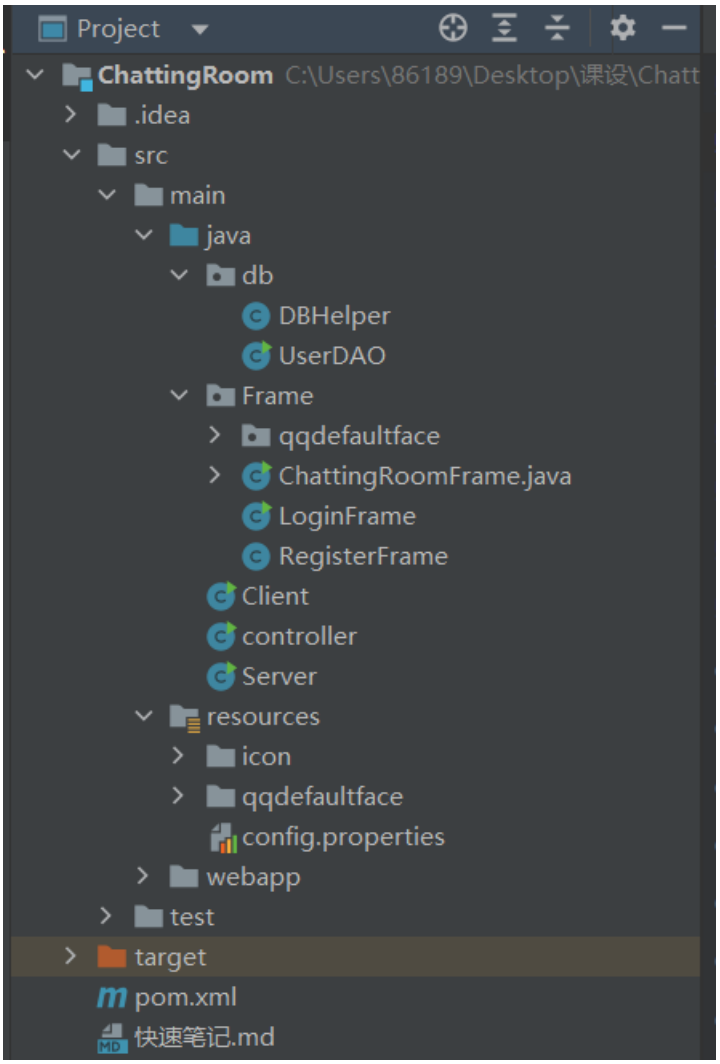
目录

题目一要求

1. 用 Java 图形用户界面编写聊天室服务器端和客户端， 支持多个客户端连接到一个服务器。每个客户端能够输入账号。
2. 可以实现群聊（聊天记录显示在所有客户端界面）。
3. 完成好友列表在各个客户端上显示。
4. 可以实现私人聊天，用户可以选择某个其他用户，单独发送信息。
5. 服务器能够群发系统消息，能够强行让某些用户下线。
6. 客户端的上线下线要求能够在其他客户端上面实时刷新。

题目一的设计思路

项目结构：
使用 maven 管理项目，db 下是数据库操作的辅助类，Frame 下是窗口类，最主要的是聊天室窗口，另外两个是登录窗口和注册窗口。Client 是客户端，Server 是服务器，controller 是服务器控制端，用来完成群发系统消息和让某些用户下线的功能。Resources 中的 icon 存一些图标，qqdefaultface 是使用到的 105 个 QQ 表情，config.properties 存放数据库连接的一些参数

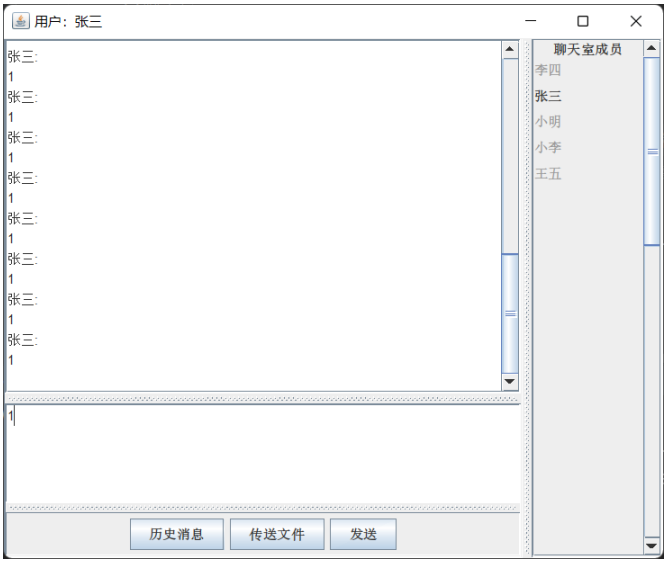


设计经历

整个聊天室总共改过三大版，第一版客户端服务器之间的通信是通过传输由信息和分隔符组成的字符串来完成的；第二版把通信部分全改成了 json 格式，操作方便也更合规范，不用担心客户输入了分隔符且更容易扩充；第三版为了能够支持发送表情功能，将 JTextArea（仅支持显示文字）都换成了 JTextPane（支持显示标签，方便信息插入）。

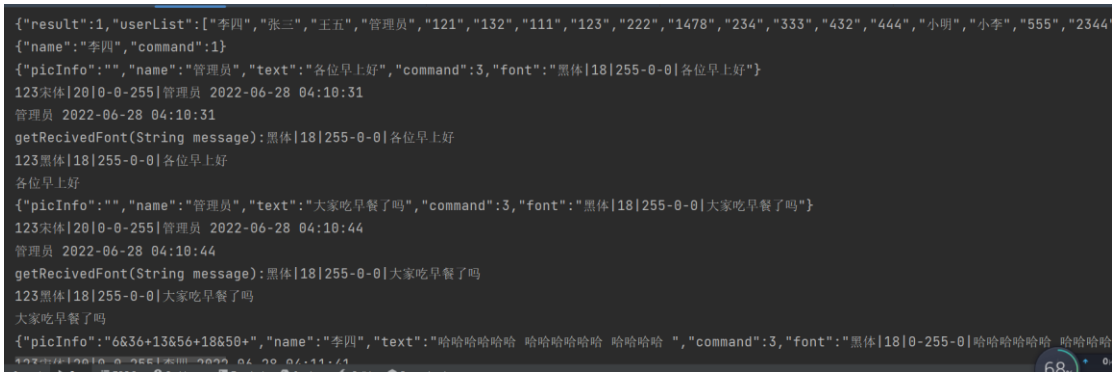
```
run():绑定输入流，在死循环中不断接收客服端发送的字符串line并处理
1."L"+用户名：登录功能，通知其他用户，修改socketMap并发送给新用户
2."R"+"\t"+用户名+"\t"+密码：注册功能，检查数据库是否重名，插入数据库，执行登录功能
3."D":注销功能，修改socketMap，通知其他用户，关闭当前线程
4."G"+消息：向socketMap中每个用户发一次line+"\t"+name（发送人）
5."S"+消息+"\t"+目标:在socketMap中找到目标，向目标发送line+"\t"+name(发送人)
6."M"+消息+"\t"+目标+"\t":在socketMap中找到目标，向目标发送line+"\t"+name(发送人)
```

1 第一版字符串传输



2 第二版的界面

3 第三版界面



4 通信时的传输格式

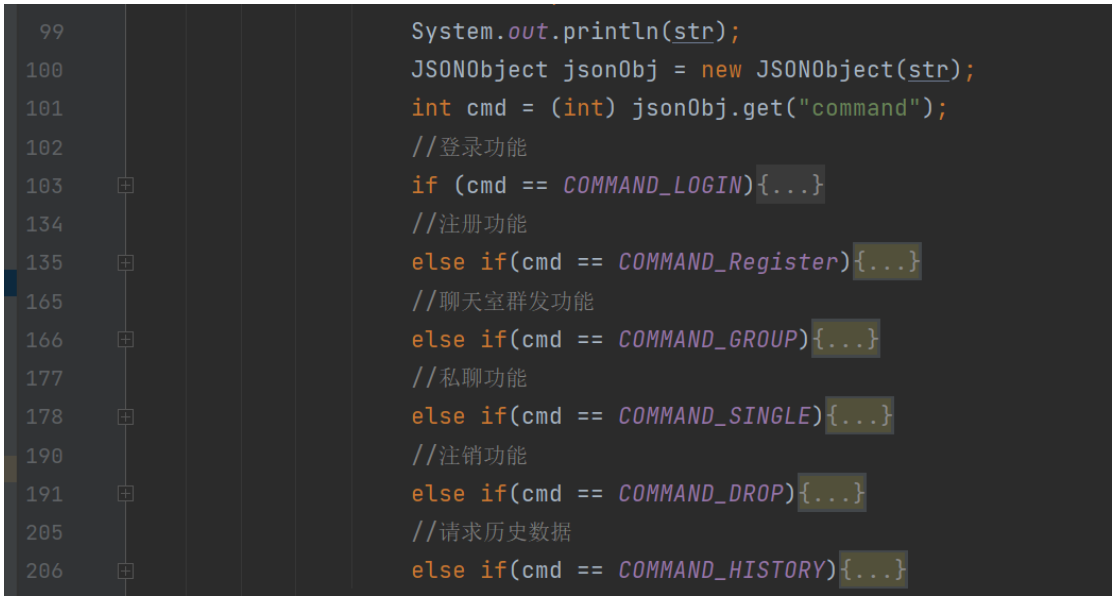
服务器：

服务器主程序在一个 while(true)中持续监听端口，每接收到一个新客户连接请求就为该客户新开一个线程 listener，负责和该客户进行通信。Listener 线程持续接收 json 数据包，用 jsonObj.get("command")取出命令，判断后执行登录、注册、群发、私聊、注销、请求历史信息等六个操作。

服务器的显示面板作为另开的一个类 controller，修改 ChattingRoomFrame 增加两个功能即可。

此外，还需要新建辅助数据库访问的 UserDAO 类封装对数据库的增删改查。

服务器编写的重点在于接收不同 command 的请求然后访问数据库请求数据并返回结果，对各种数据进行转发，对用户进行控制管理等。



5 服务器的分支处理

客户端

客户端启动弹出登录窗口，支持账户管理会校对账号密码，还可以注册账户。登陆成功后用服务器返回的数据如在线人员等信息初始化聊天室界面，同时启动新线程持续监听服务器发回的 json 包并据此更新界面，如新用户登录注册刷新在线人员表，收到群发消息显示在聊天板上，收到私聊消息根据勿扰模式开启新窗口/进行消息缓存，收到下线命令关闭 socket，收到历史消息显示在 HistoryBoard 上。

客户端最难的地方在于处理支持发送表情，此外主要动态刷新在线人员列表，对私聊消息缓存等，这些在后续详细展开，其他的都比较常规略去不谈了。

6 客户端的分支处理

题目一的实现过程

服务器

数据库处理类

比较常规，基本的连接然后用 sql 语句进行查询。这里为了方便数据库的切换，改变数据库信息如名字，密码，驱动等额外编写了一个 config.properties 文件存储这些信息

```
1. static {
2.     // 从配置文件中获得属性文件输入流
3.     InputStream input = DBHelper.class.getClassLoader()
4.         .getResourceAsStream("config.properties");
5.     try {
6.         info.load(input);
7.         url = info.getProperty("url");
8.         String driverClassName = info.getProperty("driver");
9.         Class.forName(driverClassName);
10.        System.out.println("驱动程序加载成功...");
11.    } catch (ClassNotFoundException e) {
12.        System.out.println("驱动程序加载失败...");
13.    } catch (IOException e) {
14.        System.out.println("加载属性文件失败...");
15.    }
```

数据库查询操作也很常规，数据库中的表有 user,history,friend（之前存储好友关系，后废弃不用）主要封装接口如下：

findAllUser 查询所有用户
findById 根据用户名查询用户信息
insertUser 向 user 表中插入新用户
insertMess 向 history 表中插入新的历史消息
queryHistory 根据日期在 history 表中查询历史消息

代码较多不再赘述，JAVA 访问数据库的展示基本格式如下：

```
1. String sql = "xxxxxxxxx"; //编写sql 语句
2. conn = DBHelper.getConnection();//连接数据库
3. pstmt = conn.prepareStatement(sql);//装入 sql 语句
4. pstmt.setString(1, user_name);//填写 sql 语句中的参数
```

```
5. ResultSet rs = pstmt.executeQuery();//执行 sql 语句返回结果
6. //todo 对 rs 进行一些处理得到结果 result
7. return result
```

服务器变量

服务器存储变量如下：
UserDAO dao:数据库查询辅助类
HashSet<String> userList:用户名列表
Map<String,Socket> socketMap:所有在线用户的 socket 表
int port:端口号
ExecutorService executorService:线程池
ServerSocket serverSocket:服务器 Socket

处理请求

简单封装一个 sendMessage(Socket client, String s)负责向 client 发送信息 s。服务器的 listner 总共处理六种请求：

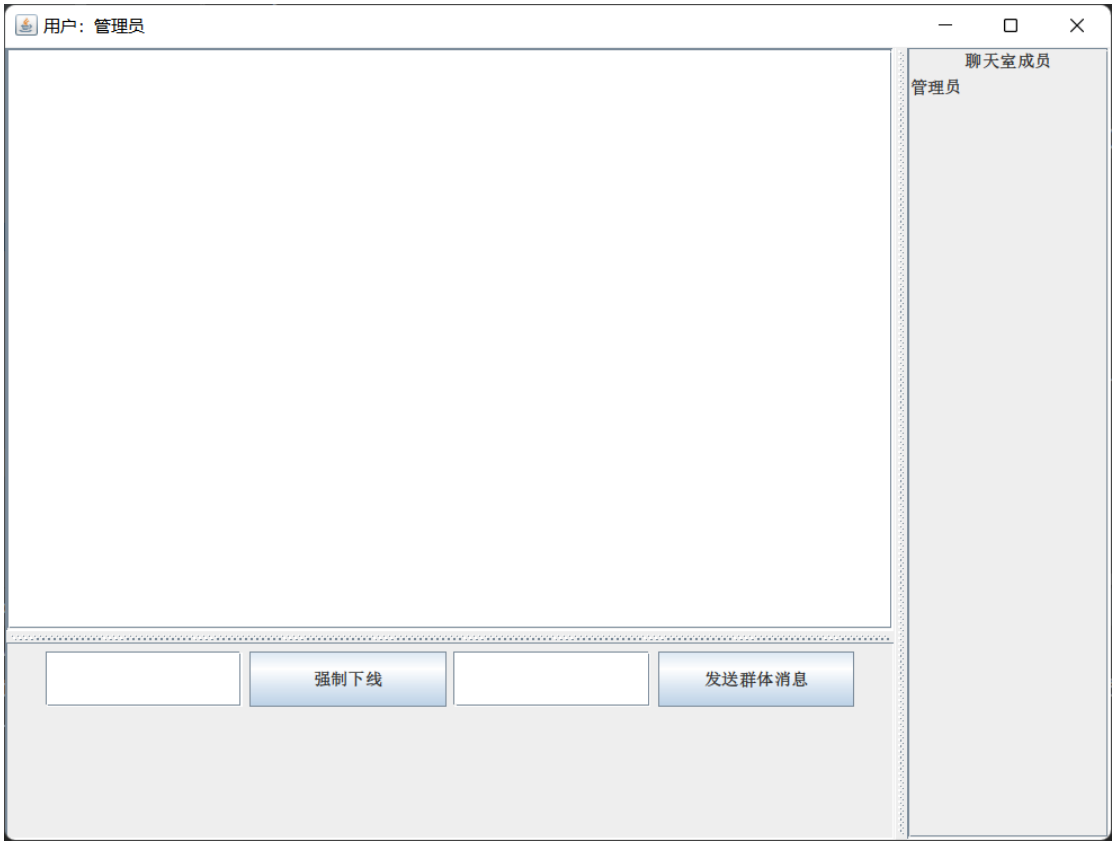
1. 群发功能：将该 json 包添加发送方名字 name 后根据 socketMap 广播即可。
2. 私聊功能：根据 json 包中的 target 字段在 socketMap 中查询目标用户的 socket 并转发该 json 包
3. 注销功能：socketMap 中移除该用户的 socket，根据 socketMap 广播该用户的下线，关闭该 socket 连接
4. 查询历史记录：根据 json 包中的 date 字段查询数据库取出当天的所有消息，封装成 json 包返回
5. 登录/注册功能较复杂如下详细展开，其中注册功能包括了登录，仅以注册功能展开：
6. 从 json 中取出姓名和密码字段，通过 dao 插入新用户数据，更新用户总表 userList，广播新的用户总成员表，至此成功注册新用户。然后根据 socketMap 广播新用户登录，再将新用户添加进 socketMap，然后将聊天室总成员名单和在线人员名单发送给新用户。这里处理的顺序不能调换，否则会出问题。（最开始编写时每个客户端会收到总成员名单和在线成员名单，显示总名单用黑色和灰色区分在线与否。后来修改为仅显示在线成员了，由于不影响功能所以未修改服务器部分代码）

```
1. else if(cmd == COMMAND_Register){
2.     name = (String) jsonObj.get("user_name");
3.     String pwd=(String) jsonObj.get("pwd");
4.     dao.insertUser(name,pwd);//插入新用户数据
5.     userList = dao.findAllUser();// 更新总用户名单信息
6.
7.     //广播更新后的总成员表
8.     JSONObject broadcast = new JSONObject();
9.     broadcast.put("command",COMMAND_Register);
10.    broadcast.put("userList",userList);
```

```
    for(Socket socket:socketMap.values()){
        sendMessage(socket,broadcast.toString());
    }
    //广播当前用户上线了
    broadcast = new JSONObject();
    broadcast.put("command",COMMAND_LOGIN);
    broadcast.put("name",name);
    for(Socket socket:socketMap.values()){
        sendMessage(socket,broadcast.toString());
    }
    //进行登录
    Map<String, String> user = dao.findById(name);
    socketMap.put(name,this.socket);//socket 表加入新用户
    JSONObject sendJsonObj = new JSONObject(user);//创建结果 json 包
    sendJsonObj.put("result", 1);// 添加 result:0 键值对，1 表示成功，0 表示失败
    Set<String> online = socketMap.keySet();// 在线人员列表
    sendJsonObj.put("online", online);
    sendJsonObj.put("userList", userList);// 聊天室成员总名单
    sendMessage(socket,sendJsonObj.toString());// 创建 DatagramPacket 对象，用于向客户端发送数据
}
```

编写控制窗口 controller

方便起见 controller 直接用管理员账号登录并实例化一个 ChattingRoomFrame control，然后用 remove 移除 control 下方输入区部分，并添加下线用户，群发消息的输入框和按钮。由于现实生活中的服务器程序是不会有可视化界面的，这里的管理员相当于一个客户端，其对服务器的控制也是通过 socket 通信，能额外发送 COMMAND_CONTROLD 强制注销和 COMMAND_CONTROLS 群发消息两个命令。服务器接收到后对于强制注销，取出 socketMap 中对应用户的 socket 发送注销命令然后关闭；对于群发消息，按正常群发消息流程走即可。代码比较简单略去，演示操作在后面的测试结果中展示，控制窗口界面展示如下：



客户端

客户端变量

客户端变量众多，展示如下，都有详尽的注释

```
//运行所需变量
int h=600;//窗口高
int w=800;//窗口宽
String username;//当前用户的名字
Socket socket;//当前用户连接的socket
SimpleDateFormat sf = new SimpleDateFormat("yyyy-MM-dd hh:mm:ss");//日期格式
FontAndText dateFont = new FontAndText("", "宋体", 20, Color.BLUE);//固定用户名+日期的信息头样式
int pos1;//全局存储当前插入的起始位置，插入表情时以此为基准
FontAndText myFont = null;//全局存储用户选择的文本样式
//运行所需容器
List<String> userlist;//用户总成员列表（修改后不再用了，可以删掉了）
Set<String> online;//在线人员名单
List<JLabel> userLabelList;//用户标签列表
Map<String,SingleChatFrame>singlewindow=new HashMap<>();//管理私聊窗口（私聊对象：私聊窗口）
Map<String,ArrayList<String>>messBuffer=new HashMap<>();//消息缓冲（私聊对象：发送信息缓冲）
List<PicInfo> myPicInfo = new LinkedList<>();//自己的表情信息
List<PicInfo> receivePicInfo = new LinkedList<>();//接收到的表情信息
//标志
boolean flag=true;//在线
//重要控件
Box menberListBox=createVerticalBox();//总成员列表面板
HistoryBoard hisbd=new HistoryBoard();//历史记录板
PicsJWindow picWindow=new PicsJWindow(this);//表情框
//聊天区
JScrollPane chatScroll = new JScrollPane();//聊天板的滚动底板
JTextPane chatBoard=new JTextPane();//聊天板
StyledDocument docChat = chatBoard.getStyledDocument();//聊天板的通用样式化文档
//输入区
JPanel inputArea = new JPanel(new BorderLayout());//输入区域
Box btnZone = Box.createHorizontalBox();//表情，历史记录，勿扰模式等开关控制区
JTextPane content = new JTextPane();//用户输入框
StyledDocument docContent = content.getStyledDocument();//输入框的通用样式化文档
```

```
1. //运行所需变量
2. int h=600;//窗口高
3. int w=800;//窗口宽
4. String username;//当前用户的名字
5. Socket socket;//当前用户连接的socket
6. SimpleDateFormat sf = new SimpleDateFormat("yyyy-MM-dd hh:mm:ss");//
   日期格式
7. FontAndText dateFont = new FontAndText("", "宋体", 20, Color.BLUE);//
   固定用户名+日期的信息头样式
8. int pos1;//全局存储当前插入的起始位置，插入表情时以此为基准
9. FontAndText myFont = null;//全局存储用户选择的文本样式
10.//运行所需容器
11.List<String> userlist;//用户总成员列表（修改后不再用了，可以删掉了）
12.Set<String> online;//在线人员名单
13.List<JLabel> userLabelList;//用户标签列表
14.Map<String,SingleChatFrame>singlewindow=new HashMap<>();//管理私聊窗
   口（私聊对象：私聊窗口）
15.Map<String,ArrayList<String>>messBuffer=new HashMap<>();//消息缓冲
   （私聊对象：发送信息缓冲）
16.List<PicInfo> myPicInfo = new LinkedList<>();//自己的表情信息
17.List<PicInfo> receivePicInfo = new LinkedList<>();//接收到的表情信息
18.//标志
```

```
19.boolean flag=true;//在线
20.//重要控件
21.Box menberListBox=createVerticalBox();//总成员列表面板
22.HistoryBoard hisbd=new HistoryBoard();//历史记录板
23.PicsJWindow picWindow=new PicsJWindow(this);//表情框
24.//聊天区
25.JScrollPane chatScroll = new JScrollPane();//聊天板的滚动底板
26.JTextPane chatBoard=new JTextPane();//聊天板
27.StyledDocument docChat = chatBoard.getStyledDocument();//聊天板的通用
   样式化文档
28.//输入区
29.JPanel inputArea = new JPanel(new BorderLayout());//输入区域
30.Box btnZone = Box.createHorizontalBox();//表情，历史记录，勿扰模式等开
   关控制区
31.JTextPane content = new JTextPane();//用户输入框
32.StyledDocument docContent = content.getStyledDocument();//输入框的通
   用样式化文档
```

客户端处理服务器返回 json 包

客户端持续监听服务器返回 json 包，如果是接收到他人登录/注册/下线的消息，需要 refreshFriendList 刷新在线人员面板（刷新实现在难点解决中展开）,如果是群发消息则将信息展示在聊天板中（在下面的发送表情功能中展开），如果是私聊消息则弹出私聊窗口开始私聊（在难点解决中展开），如果是强制下线则直接关闭 socket 并弹出提示提醒用户被强制下线。如果是查询历史记录则显示历史信息板（在下面历史信息中展开）。

登录注册功能

根据控件的 getText 获取输入框的用户名和密码信息，进行判空处理后调用 login 函数，根据信息封装 json 包并传送给服务器，然后等待服务器返回的结果包。上层处理接收 login 返回的 map，如果为空说明登录失败弹出提示，否则用服务器返回的 json 包去初始化聊天室窗口 ChattingRoomFrame。其中主要的是 login 函数，代码如下：

```
1. public static Map login(String name, String pwd) {
2.     try {
3.         JSONObject jsonObj = new JSONObject();
4.         jsonObj.put("command", COMMAND_LOGIN);
5.         jsonObj.put("user_name", name);
6.         jsonObj.put("user_pwd", pwd);
7.         // 以字符串传输json
8.         OutputStream out = socket.getOutputStream();
9.         PrintStream printStream = new PrintStream(out);
10.        printStream.println(jsonObj.toString());
```

```
        printStream.flush();
        // 接收返回的json 结果
        InputStream in = socket.getInputStream();
        Scanner sc = new Scanner(in);
        String msg="";
        if(sc.hasNext()){
            msg = sc.nextLine();
        }
        JSONObject receivedjsonObj = new JSONObject(msg);
        System.out.println(receivedjsonObj);
        if ((Integer) receivedjsonObj.get("result") == 1) {
            Map user = receivedjsonObj.toMap();
            return user;
        }
    } catch (IOException e) {
        e.printStackTrace();
    }
    return null;
}
```

历史信息功能

定义历史消息面板类 HistoryBoard，通过 addHistory 接口向面板中添加历史信息。
从 json 包中取出 history 字段信息转化为 List<String> historytext，并调用 addHistory 添加历史信息，代码略，展示效果如下：

发送表情功能

辅助类

ChatPic:
继承自 ImageIcon，封装图标，主要是加一个可修改的代号。新增属性图片代号 int im。

PicsJWindow:
继承自 JWindow（单纯的窗体没有关闭缩小等按键）制作自定义表情框。用 GridLayout(7, 15)制作 7*15 的网格存放 105 个表情，用 JLabel[] ico = new JLabel[105]的 JLabel 数组存放 105 个表情。
Init()初始化函数根据文件路径名找到表情存放的文件位置，读取 qqdefaultface 中的每个 gif 文件制成 JLabel 依次存入 ico 中，为每个表情添加鼠标事件移入时蓝边框移出时还原，点击后调用父窗口的 insertSendPic 将所选表情添加进输入框。
setVisible 用于设置窗口可见/隐藏

FontAndText:
字体和文本类，主要封装字体样式包括文本，字体类型，字号大小，字体颜色并将其封装成 SimpleAttributeSet attrSet 以方便 JTextPane 使用。

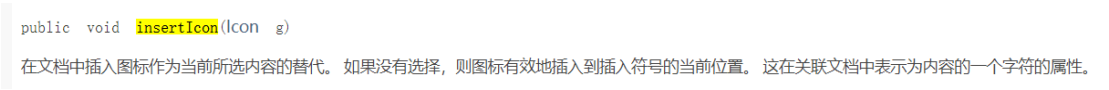
PicInfo：
图片信息。由于双方都有表情库，所以传输的时候不是发送图片，而是传文本串，表情在文本串中的位置，表情序号。然后接收方先插入文本串，然后根据序号找到表情，插入文本串中的对应位置。所以封装图片信息类存表情的位置 pos 和序号 val。

发送方发送表情时候的处理

发送区表情的显示

点击表情按键时调用 setVisible 设置表情框 PicsJWindow 可见，然后用户选择表情触发点击事件，表情框调用聊天板的 insertSendPic 在输入区中插入图片。调用 JTextPane 的 insertIcon 函数可以方便地在输入区中插入表情图片。

```
1. emoji.addActionListener(l-> {
2.     picWindow.setVisible(true);//弹出表情框
3. });
```




```
1. public void insertSendPic(ImageIcon imgIc) {
2.     content.insertIcon(imgIc); // 插入图片
3. }
```

发送的处理

发送的处理较为麻烦。用户按下发送按键后先判空处理，然后开始组织 json 包。

先调用 getFontAttrib 函数获取输入区的内容和各个控件的选择返回一个 FontAndText，内含发送文本，字体类型，字体大小，颜色等信息，将该 FontAndText 转成字符串加进 json 包的 font 字段

调用 buildPicInfo 获取表情信息。具体做法是通过输入区 content 的 StyledDocument 遍历每一位，判断是否为表情标签，如果是取出这张表情封装成 PicInfo 并加进 myPicInfo 数组中。且封装出表情信息串 info 返回，格式为“表情位置&表情序号+表情位置&表情序号+表情位置&表情序号……”如"8&3+12&11+"。将返回 info 装入 json 中的 picInfo 字段。

填写 command 和 text 内容后发送给服务器。

```
1. //返回文本样式信息
2. private FontAndText getFontAttrib() {
3.     FontAndText att = new FontAndText();
4.     att.setText(content.getText());//文本信息
```

```
    att.setName((String) fontName.getSelectedItem());
    att.setSize(Integer.parseInt((String) fontSize.getSelectedItem()
));
    String temp_color = (String) fontColor.getSelectedItem();
    if (temp_color.equals("黑色")) {
        att.setColor(new Color(0, 0, 0));
    } else if (temp_color.equals("红色")) {
        att.setColor(new Color(255, 0, 0));
    } else if (temp_color.equals("蓝色")) {
        att.setColor(new Color(0, 0, 255));
    } else if (temp_color.equals("黄色")) {
        att.setColor(new Color(255, 255, 0));
    } else if (temp_color.equals("绿色")) {
        att.setColor(new Color(0, 255, 0));
    }
    return att;
}
```

```
//重组表情信息，重组后的信息串  格式为   位置&代号+位置&代号+……
private String buildPicInfo() {
    StringBuilder sb = new StringBuilder("");
    //遍历jtextpane 找出所有的图片信息封装成指定格式
    for (int i = 0; i < this.content.getText().length(); i++) {
        if (docContent.getCharacterElement(i).getName().equals("icon
")) {
            Icon icon = StyleConstants.getIcon(content.getStyledDocu
ment().getCharacterElement(i).getAttributes());
            ChatPic cupic = (ChatPic) icon;
            PicInfo picInfo = new PicInfo(i, cupic.getIm() + "");
            myPicInfo.add(picInfo);
            sb.append(i).append("&").append(cupic.getIm()).append("+
");
        }
    }
    System.out.println(sb.toString());
    return sb.toString();
}
```

接收方接收信息时对表情的处理

接收线程接收到群发命令的 json 包，取出 name,text,picInfo,font 四个字段的数据，调用 addRecMsg 函数。

将 name 组合当前时间作为文本，加上 font 组成 FontAndText 类型的数据，调用 insert 函数插入聊天板中。（即每条信息前的用户 时间部分）

用 getReceiveFont 从 font 中拆解出信息，即将字符串 font 转换成 FontAndText 类型数据 attr 然后调用 insert 插入（即每条信息的正文部分）

如果检测到有表情信息，则调用 receivedPicInfo 将字符串 picInfo 拆解信息，将每个表情转换成 PicInfo 类型数据并存入 receivePicInfo 中。然后调用 insertPics 将 receivePicInfo 中的表情插入进文本中。注意 receivePicInfo 是缓存，每次用完要清空（即根据表情所在位置和序号在每条信息的正文部分中插入表情）

下面仅展示组织其他函数的 addRecMsg 函数:

```
1. public void addRecMsg(String uname, String message,String picInfo,St
   ring font) {
2.     String msg = uname + " " + sf.format(new Date());
3.     dateFont.setText(msg);//用户信息和时间
4.     insert(dateFont);
5.     pos1 = chatBoard.getCaretPosition();//记录插入的起始点
6.     if (!picInfo.equals("")) {/*存在表情信息*/
7.         FontAndText attr = getReceiveFont(font);
8.         insert(attr);
9.         receivedPicInfo(picInfo);
10.        insertPics();
11.    } else {
12.        FontAndText attr = getReceiveFont(font);
13.        insert(attr);
14.    }
15.}
```

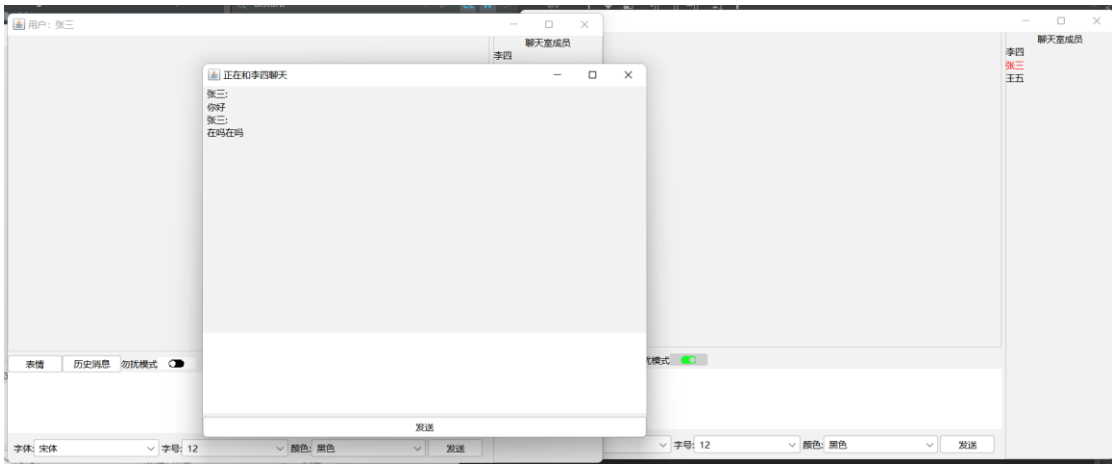
题目一的难点及解决

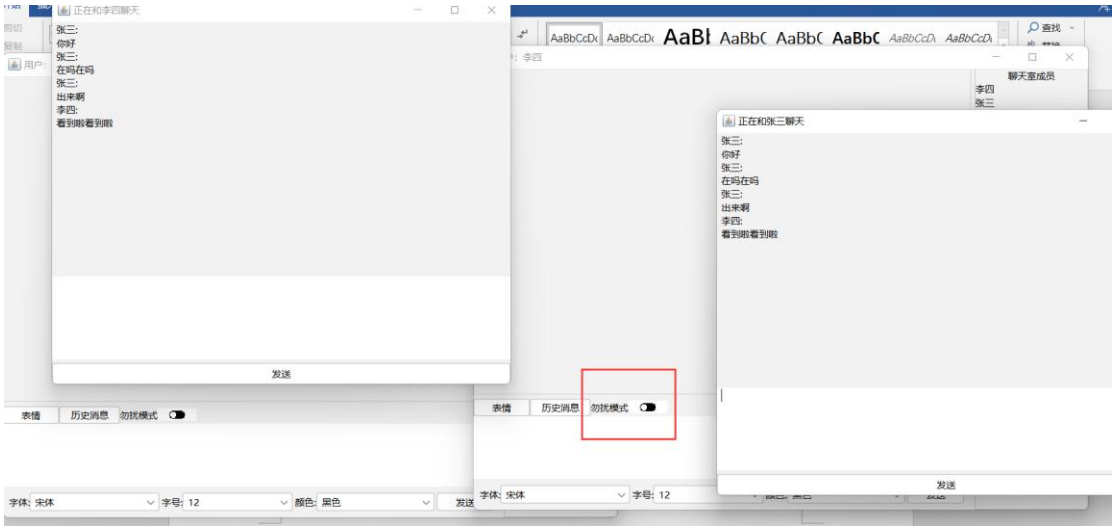
用户私聊信息的缓存（免打扰模式）

聊天室窗口右侧的在线成员列表中，双击标签即可弹出一个聊天对话框开始私人聊天，发送消息。考虑到如果每次有人发送私聊消息就弹出一个窗口，会对用户造成骚扰，于是增设了

一个免打扰功能，开启免打扰功能后将发送方标签标红并缓存消息。这里使用 Map<String,SingleChatFrame>singlewindow 来管理该用户的所有私聊窗口，用 Map<String,ArrayList<String>>messBuffer 来管理所有用户发来的消息缓存。每当接收到私聊消息时取出发送方 A，去 singlewindow 中找是否有 A，如果有则直接在该聊天窗口添加内容。如果没有，在 messBuffer 的 A 上增加一条消息缓存，然后检查用户是否开了免打扰。如果开了免打扰，则将对方标红，结束。否则取出 messBuffer 中 A 的消息缓存并弹出和 A 的聊天窗口。

```
1. else if(mes==COMMAND_SINGLE){
2.     //先检查 singlewindow 是否已经打开了窗口，如果打开了，控制该窗口添加消息
3.     String name=(String) receivedjsonObj.get("name");//name 是发送方的名字
4.     String text=(String) receivedjsonObj.get("text");//text 是对方发送的内容
5.     if(singlewindow.containsKey(name))
6.         singlewindow.get(name).addMes(name,text);
7.     //如果还没打开窗口，先缓存，并将用户标红
8.     else{
9.         if(!messBuffer.containsKey(name))
10.            messBuffer.put(name,new ArrayList<>());
11.        messBuffer.get(name).add(text);
12.        for (JLabel ulbl : userLabelList)
13.            if (name.equals(ulbl.getText()))
14.                ulbl.setForeground(Color.red);
15.        //如果关闭勿扰模式则直接弹出弹窗
16.        if(!disturb.isSelected())
17.            for(JLabel lbu:userLabelList)
18.                if(lbu.getText().equals(name)){
19.                    singlewindow.put(name,new SingleChatFrame(name));
20.                    lbu.setForeground(Color.black);
21.                }
22.    }
23.}
```





登录/注册/注销时在线成员列表的动态刷新

用 JPanel onlineUser 存储在线成员列表，最开始简单地控制 onlineUser 的 add 和 remove，发现更新没有效果，然后网上找了几个也会出现各种显示问题，比如直接白屏，最终找到了 panel.validate();panel.repaint(); panel.revalidate();试成功了，能够完成 Jpanel 的动态刷新。实现上登录/注册/注销都会提供要修改的用户名，在线用户名单存在 Set<String> online 中，直接对 online 进行修改，然后调用 genMenberList 函数，该函数每次都会遍历 online 重新绘制 onlineUser。

```
void genMenberList(){
    menberListBox.removeAll();
    menberListBox.repaint();
    menberListBox.invalidate();
    userLabelList = new ArrayList<JLabel>();
    for (String us:online) {
        JLabel lbu = new JLabel(us);
        lbu.addMouseListener(new MouseAdapter() {
            @Override
```

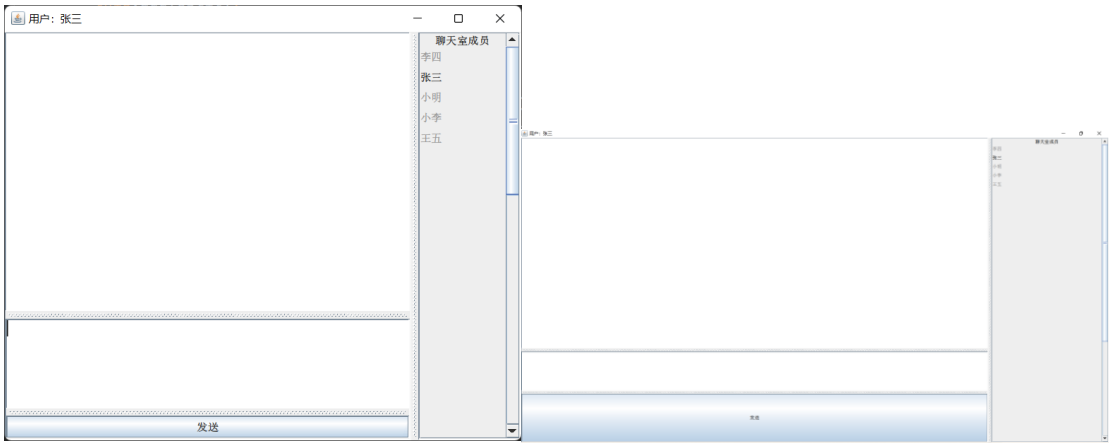
```
            public void mouseClicked(MouseEvent e) {
                // 用户图标双击鼠标时显示对话框
                if (e.getClickCount() == 2) {
                    singlewindow.put(us,new SingleChatFrame(us));
                    lbu.setForeground(Color.black);
                }
            }
        });
        userLabelList.add(lbu);
        menberListBox.add(lbu);
    }
    menberListBox.validate();
    menberListBox.updateUI();
}
```

```
1. //上下线刷新
2. public void refreshFriendList(String userName, String flag) {
3.     // 更新聊天室成员列表
4.     if (flag.equals("1")) {
5.         online.add(userName);
6.         genMenberList();
7.     }
8.     else{
9.         online.remove(userName);
10.        genMenberList();
11.    }
12.}
```

窗口放大缩小的动态变化

使用 JSplitPane 进行百分比分割时如果窗口放大缩小那么百分比分割的效果就会失效，经过搜索后添加监听事件监听窗口变化后部分解决。

```
13.this.addComponentListener(new ComponentAdapter(){
14.    public void componentResized(ComponentEvent e) {
15.        split.setDividerLocation(0.8);
16.        left.setDividerLocation(0.7);
17.        inputArea.setDividerLocation(0.8);
18.    }
19.});
```

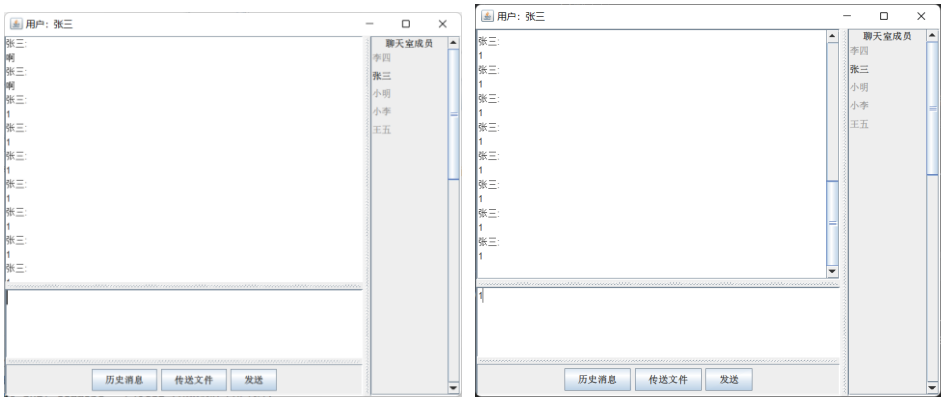


但放大后仍然有问题，后来经过尝试发现调换顺序后解决问题。

```
20.this.addComponentListener(new ComponentAdapter(){
21.    public void componentResized(ComponentEvent e) {
22.        inputArea.setDividerLocation(0.8);
23.        split.setDividerLocation(0.8);
24.        left.setDividerLocation(0.7);
25.    }
26.});
```

聊天窗口的显示

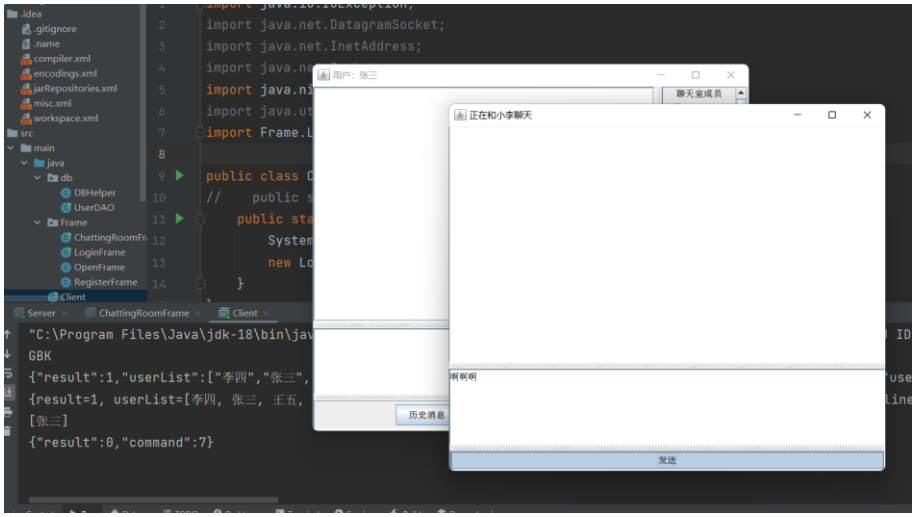
当输入多条信息时越界了聊天窗口中就看不到了



增设滚动条后得以解决。

私聊功能接收结果报文死机问题

为了共用一些变量我的私聊窗口和聊天区窗口放一个文件里了。私聊窗口中发送消息后需要接收结果报文，结果窗口死机了。

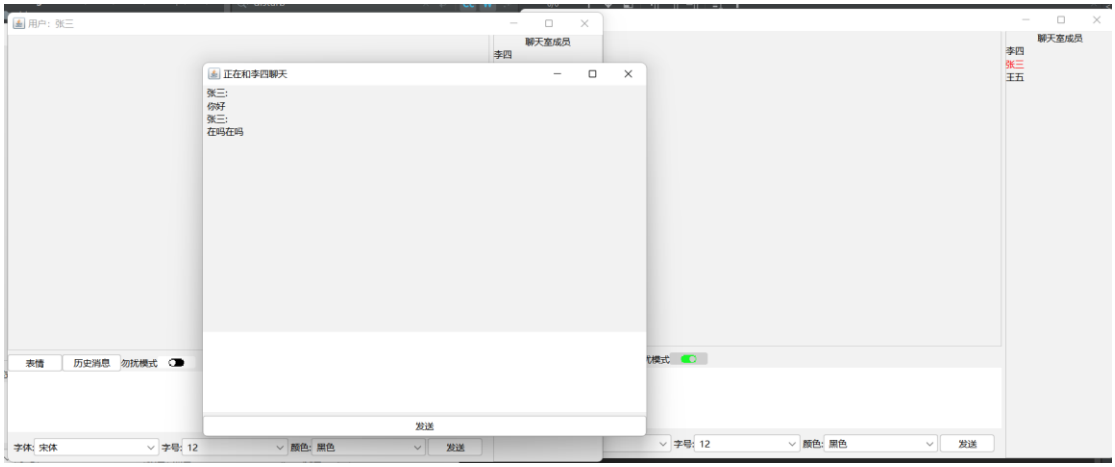


经排查发现由于私聊窗口和监听进程是并发的，所以私聊窗口等待的结果报文被聊天室窗口类中的独立的监听线程截获了，导致私聊窗口迟迟等不到回应报文死机了。这个想了挺久，最后是删去了私聊窗口等待结果报文的过程，结果报文由监听线程来接收，并调用私聊窗口的 response 接口来传递结果报文并更新私聊窗口。

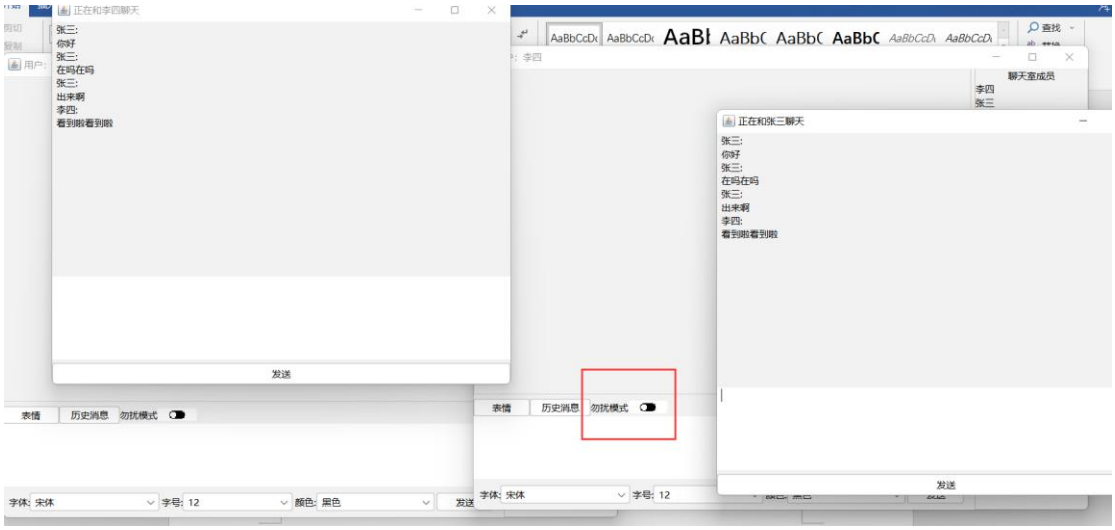
题目一的运行及测试结果

私聊功能

在免打扰模式下别人的私聊消息会进行缓存，并将发送方标红以提醒用户有新消息：



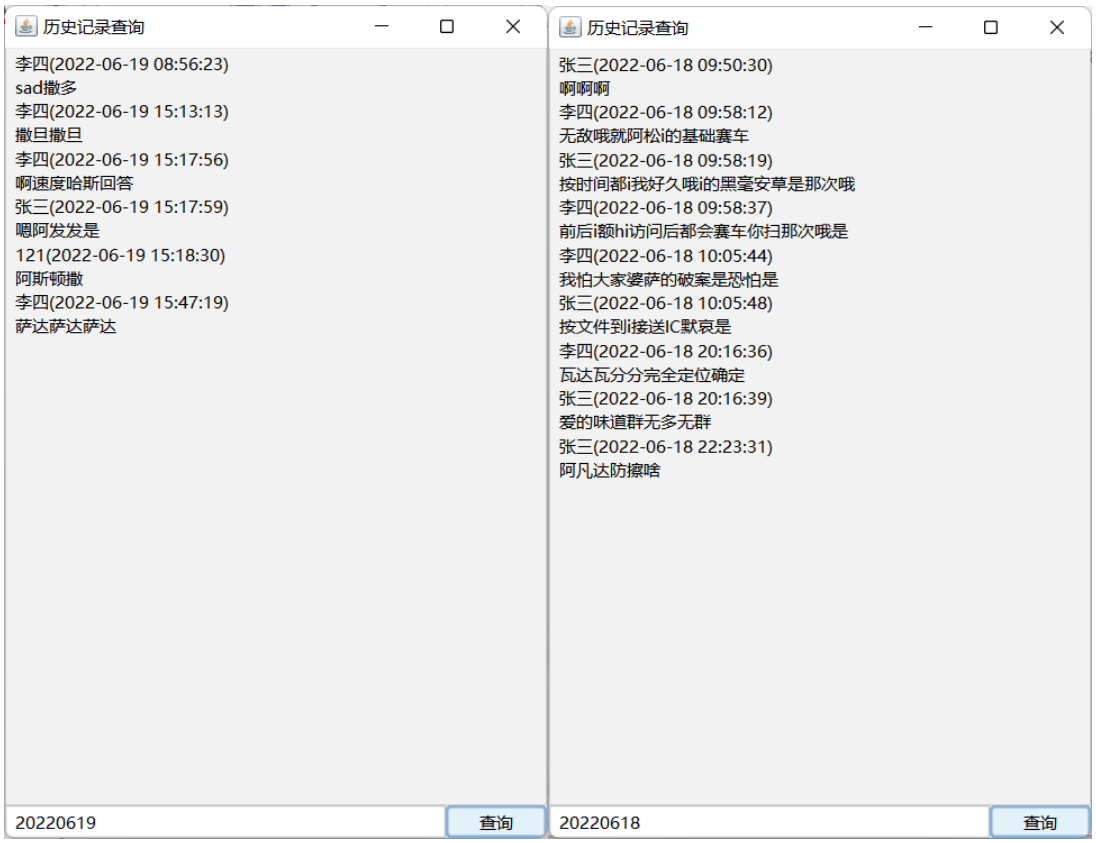
关闭免打扰后别人的私聊消息会直接弹出，如之前有消息会自动加载消息缓存：



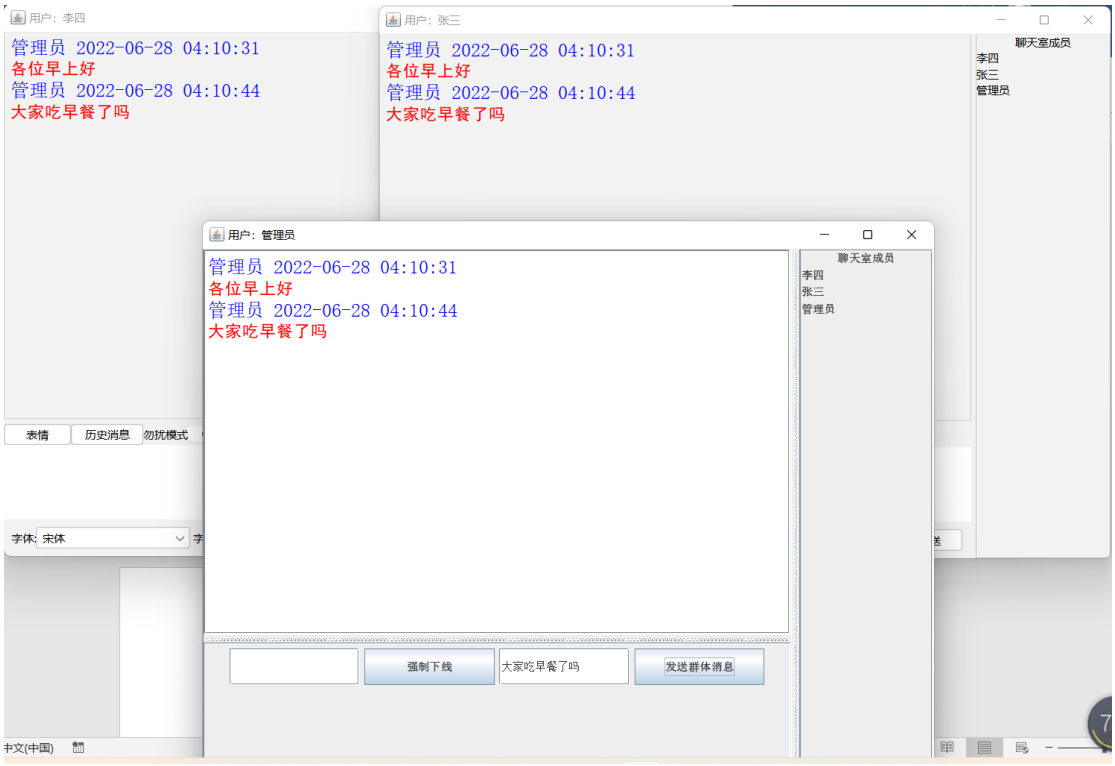
聊天室群聊功能

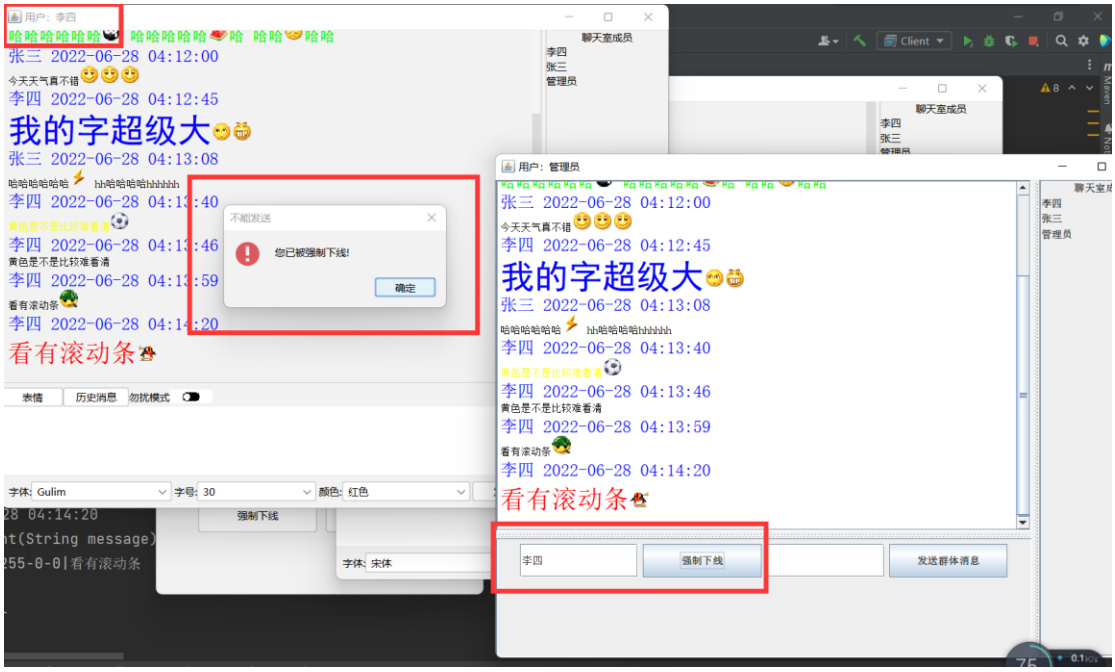
点击表情按键可以弹出表情框，发送区可自定义文字样式

历史消息查询


历史记录查询
李四(2022-06-19 08:56:23)
sad撒多
李四(2022-06-19 15:13:13)
撒旦撒旦
李四(2022-06-19 15:17:56)
啊速度哈斯回答
张三(2022-06-19 15:17:59)
嗯阿发发是
121(2022-06-19 15:18:30)
阿斯顿撒
李四(2022-06-19 15:47:19)
萨达萨达萨达
20220619
查询
历史记录查询
张三(2022-06-18 09:50:30)
啊啊啊
李四(2022-06-18 09:58:12)
无敌哦就阿松i的基础赛车
张三(2022-06-18 09:58:19)
按时间都i我好久哦i的黑毫安草是那次哦
李四(2022-06-18 09:58:37)
前后i额hi访问后都会赛车你扫那次哦是
李四(2022-06-18 10:05:44)
我怕大家婆萨的破案是恐怕是
张三(2022-06-18 10:05:48)
按文件到i接送IC默哀是
李四(2022-06-18 20:16:36)
瓦达瓦分分完全定位确定
张三(2022-06-18 20:16:39)
爱的味道群无多无群
张三(2022-06-18 22:23:31)
阿凡达防擦啥
20220618
查询

服务器端


用户：李四
管理员 2022-06-28 04:10:31
各位早上好
管理员 2022-06-28 04:10:44
大家吃早餐了吗
用户：张三
管理员 2022-06-28 04:10:31
各位早上好
管理员 2022-06-28 04:10:44
大家吃早餐了吗
聊天室成员
李四
张三
管理员
用户：管理员
管理员 2022-06-28 04:10:31
各位早上好
管理员 2022-06-28 04:10:44
大家吃早餐了吗
聊天室成员
李四
张三
管理员
表情
历史消息
勿扰模式
字体: 宋体
强制下线
大家吃早餐了吗
发送群体消息


用户：李四
张三 2022-06-28 04:12:00
今天天气真不错
李四 2022-06-28 04:12:45
我的字超级大
张三 2022-06-28 04:13:08
李四 2022-06-28 04:13:40
李四 2022-06-28 04:13:46
黄色是不是比较难看清
李四 2022-06-28 04:13:59
看有滚动条
李四 2022-06-28 04:14:20
看有滚动条
不能发送
您已被强制下线!
确定
聊天室成员
李四
张三
管理员
用户：管理员
张三 2022-06-28 04:12:00
今天天气真不错
李四 2022-06-28 04:12:45
我的字超级大
张三 2022-06-28 04:13:08
李四 2022-06-28 04:13:40
李四 2022-06-28 04:13:46
黄色是不是比较难看清
李四 2022-06-28 04:13:59
看有滚动条
李四 2022-06-28 04:14:20
看有滚动条
表情
历史消息
勿扰模式
字体: Gulim
字号: 30
颜色: 红色
强制下线
字体: 宋体
李四
强制下线
发送群体消息

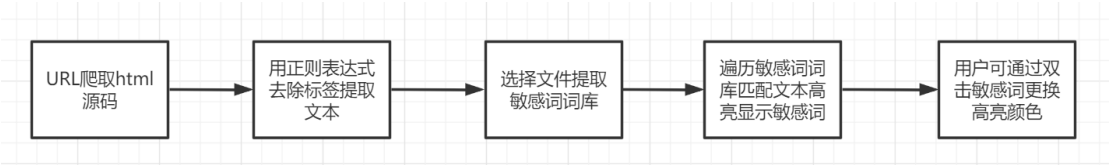
题目二要求

1.编写界面，输入一个网址，能够爬取该网址上所有的 HTML 源代码。
2.对网址中的文本进行提取。
3.建立敏感词库，用文本文件保存。
4.将该网址所对应的文本中的敏感词提取并高亮显示。
5.编写文本文件，可以存入多个网址；程序可爬取这些网址中的文本内容，将敏感词记录存入另一个文件，格式自定。
6.编写一个主界面,整合上述功能。

题目二的设计思路

题目比较简单就只设置一个文件，需要实现的功能依次为：
1.爬取 html 源码，主要是 URL 类和 URLConnection 类的使用
2.从 html 源码中提取文本，这里采用正则表达式去除标签
3.文本文件存储敏感词库,代码部分涉及文件读写,其中文件选择使用 swing 的 JFileChooser。
4.提取文本敏感词并高亮显示，主要是涉及 DefaultHighlighter 的使用
5.文本文件存多个网址并爬取主要页数涉及文件读写和选择
6.这里考虑到所有敏感词都使用一个颜色看不出区别，增加一个双击敏感词给该敏感词更换颜色的功能。

题目二的实现过程



爬取 html

这里主要是参考网上使用 URL 和 URLConnection 来爬取 html 文本的标准流程来写的，抽象类 URLConnection 是表示应用程序和 URL 之间的通信链接的所有类的超类。 该类的实例可以用于从 URL 引用的资源中读取和写入。获取输入流后读取每一行并添加进 text 中，最后返回 html 源代码 text。

```
1. public String getHtml(String website) {
2.     String str=null;
3.     String text="";  //保存网页的内容
4.     try {
5.         URL url=new URL(website); //建立对应的URL 对象
```

```
6.          URLConnection urlConne=url.openConnection(); //连接
7.          urlConne.connect();
8.          //获取输入流
9.          BufferedReader br=new BufferedReader(new InputStreamReader(u
   rlConne.getInputStream(),"UTF-8"));
10.         System.out.println("开始爬取");
11.         while(true) { //爬取到结束
12.             str=br.readLine();
13.             if(str==null) break;
14.             text+=(str+"\n");
15.         }
16.         br.close();  //关闭输入流
17.     }catch (Exception e) {
18.         // TODO: handle exception
19.         JOptionPane.showMessageDialog(null, website+"爬取源代码失败");
20.     }
21.     System.out.println("爬取结束");
22.     return text; //返回 html 代码文本
23.}
```

提取文本

提取文本主要是使用正则表达式匹配，注意到 script 是代码标签，style 是样式标签，这两种标签中间的内容是不需要的，所以先去除这两种标签以及它们中间的内容，然后去除剩下的标签（其他标签中间的内容要保留，这些就是需要提取的文本）。要注意正则表达式匹配的顺序不可以调换，如果提前去除标签就找不到 style 和 script 标签了，代码和样式也会保留，而这些是不需要的。

```
1. //设置正则表达式的匹配符
2. private String regExHtml="<[^>]+>";  //匹配标签
3. private String regExScript = "<script[^>]*?>[\\s\\S]*?<\\/script>";
    //匹配 script 标签
4. private String regExStyle = "<style[^>]*?>[\\s\\S]*?<\\/style>";  //
   匹配 style 标签
5. private String regExSpace="[\\s]{2,}"; //匹配连续空格或回车等
6. private String regExImg="&[\\S]*?;+"; //匹配网页上图案的乱码
7. //定义正则表达式
8. private Pattern pattern3=Pattern.compile(regExHtml, Pattern.CASE_INS
   ENSITIVE);
9. private Pattern pattern1=Pattern.compile(regExScript,Pattern.CASE_IN
   SENSITIVE);
10.private Pattern pattern2=Pattern.compile(regExStyle,Pattern.CASE_INS
   ENSITIVE);
```

```
11.private Pattern pattern4=Pattern.compile(regExSpace, Pattern.CASE_IN
   SENSITIVE);
12.private Pattern pattern5=Pattern.compile(regExImg,Pattern.CASE_INSEN
   SITIVE);
13.//对 html 进行正则匹配,提取出其中的文本
14.public String getText(String str) {
15.     Matcher matcher=pattern1.matcher(str);
16.     str=matcher.replaceAll("");  //匹配普通标签
17.     matcher=pattern2.matcher(str);
18.     str=matcher.replaceAll("");  //匹配 script 标签
19.     matcher=pattern3.matcher(str);
20.     str=matcher.replaceAll("");  //匹配 style 标签
21.     matcher=pattern4.matcher(str);
22.     str=matcher.replaceAll("\n"); //匹配连续回车或空格
23.     matcher=pattern5.matcher(str);
24.     str=matcher.replaceAll("");  //匹配网页图案出现的乱码
25.     return str;  //返回文本
26.}
```

界面设计和高亮显示

界面上一个输入区，包含一个输入框和一些按钮（开始爬取，爬取多个网址，删除一个结果页），一个显示区显示爬取的文本内容，一个侧栏显示敏感词。展示结果时多窗口会混乱，但爬过的结果由需要保留，为了方便起见，展示结果用 JTabbedPane 选项卡组件，将一次结果封装成一个 page 类，继承自 JPanel，每产生一次结果就建一个 page 并添加进选项卡中，设置删除按钮点击时获取当前页面并删除，即通过 tab.remove(tab.getSelectedIndex())删除。

Page 页接收爬取的文本并展示在 JTextArea 上，每个结果都可以选择自己的敏感词词库，所以每个 page 页都要有一个选择敏感词库的按键，通过 JFileChooser 选择文件读取敏感词存入 wordList 中。高亮显示时遍历 wordList 取出每一个敏感词，用 indexOf 在文本串中不断匹配该敏感词，用 hg.addHighlight(index, index+str.length(), painter)添加高亮。其中 addHighlight 的用法如下

addHighlight

Object addHighlight(int p0, int p1, Highlighter.HighlightPainter p) throws BadLocationException

在视图中添加突出显示。返回可用于引用突出显示的标记。

参数

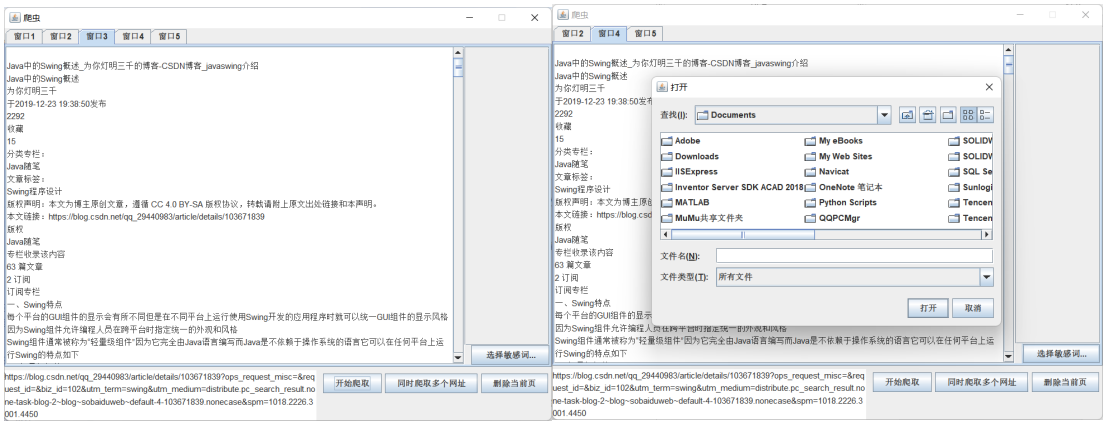
p0 - 范围的开头> = 0

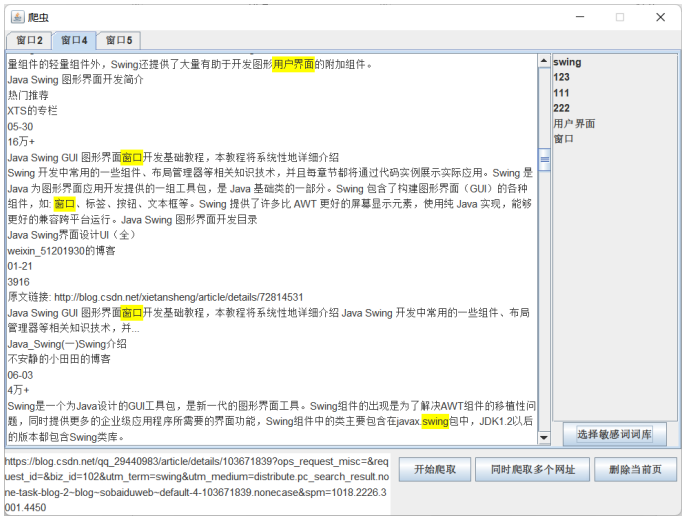
p1 - 范围的结尾> = p0

p - 用于实际突出显示的画家

```
1. public void showSensword() {
2.     Highlighter hg=txt.getHighlighter(); //设置文本框的高亮显示
3.     hg.removeAllHighlights(); //清除之前的高亮显示记录
```

```
4.      String text=txt.getText(); //得到文本框的文本
5.      DefaultHighlighter.DefaultHighlightPainter painter=new DefaultHighlighter.DefaultHighlightPainter(Color.YELLOW); //设置高亮显示颜色为黄色
6.      //对于每一个敏感词，用while循环遍历整个字符串找到匹配到的每一个位置，在每一个位置上增加一个高亮
7.      for(String str:wordList) { //匹配其中的每一个敏感词
8.          ArrayList<Object>tmp=new ArrayList<>();
9.          int index=0,cnt=0;
10.         while((index=text.indexOf(str,index))>=0) {
11.             try {
12.                 tmp.add(hg.addHighlight(index, index+str.length(), painter));//高亮显示匹配到的词语
13.                 index+=str.length(); //更新匹配条件继续匹配
14.                 cnt++;
15.             } catch (BadLocationException e) {
16.                 // TODO Auto-generated catch block
17.                 e.printStackTrace();
18.             }
19.         }
20.         hightLight.put(str,tmp);
21.         ColorState.put(str,0);//初始都为黄色
22.         System.out.println(cnt);
23.         count.put(str,(Integer) cnt);
24.     }
25.}
```





爬取多个网站

新建 SpiderAll 继承自 Thread 类，以便爬取文件中的多个网站的同时用户还能查询其他网站。调用前先用两个 JFileChooser 选择敏感词词库文件和存 URL 的文件，读取敏感词词库进入全局变量 wordList 中，然后和爬取一个网站的流程相同，调用 getHtml 获取 html 源代码，调用 getText 从源码中提取文本，遍历 wordList 取出每一个敏感词和文本进行匹配，每匹配到一个 wordNum 中该敏感词对应位+1。最终整合数据并写入文件即可。

```
26.public void run() {
27.    try {
28.        //读取网址库中的网址
29.        BufferedReader brr=new BufferedReader(new FileReader(file));
30.        //将匹配数据写入文本中
31.        PrintStream ps=new PrintStream(new File("data.txt"));
32.        ps.println("敏感词记录如下:");
33.        int size=wordList.size();
34.        while(true) {
35.            String website=brr.readLine();
36.            if(website==null) break;
37.            ps.println(website+"数据如下: ");
38.            String html=getHtml(website); //获取 html 代码
39.            String text=getText(html);  //匹配网页文本
40.            for(int i=0;i<size;i++) {  //在网页文本中进行匹配
41.                String word=wordList.get(i);
42.                int index=0,account=0,len=word.length();
43.                while((index=text.indexOf(word,index))>=0) {
44.                    account++;
45.                    int temp=wordNum.get(i); //更新数据
46.                    wordNum.set(i,++temp);
47.                    index+=len;  //更新匹配条件
48.                }
```

```
                ps.println(word+"  出现  "+account+"次"); //写入当前数据
            }
            ps.println();
        }
        brr.close(); //关闭文件流
        System.out.println("爬取完毕");
        ps.println("总数据如下:     "); //写入总数据
        for(int i=0;i<size;i++) {
            ps.println(wordList.get(i)+"  出现    "+wordNum.get(i)+"次");
        }
        ps.close();  //关闭文件流
        JOptionPane.showMessageDialog(null, "爬取完毕！请打开文件查看!");
    }catch (Exception e) {
        // TODO: handle exception
        JOptionPane.showMessageDialog(null, "爬取失败");
    }
}
```

题目二的难点及解决

给敏感词的高亮换色

考虑到所有敏感词都使用一个颜色看不出区别，为侧栏的每个敏感词标签绑定一个双击事件，每次双击时就给该敏感词对应的高亮换一种颜色。

每个 page 页面中用 HashMap<String,ArrayList<Object>>hightLight 存储每个敏感词的高亮对象数组。高亮标出敏感词时用的是 tmp.add(hg.addHighlight(index, index+str.length(), painter)); 然后 hightLight.put(str,tmp);这样就把所有敏感词对应的高亮对象存储进 hightLight 中了。HashMap<String,Integer>ColorState 记录每个敏感词对应的颜色，用 Int 来代表颜色，总共 4 种，即每次换颜色就是+1 模 4 再去找对应的颜色。绑定双击事件如下:

```
jbl.addMouseListener(new MouseAdapter() {
    @Override
    public void mouseClicked(MouseEvent e) {
        // 双击敏感词换色。根据点击标签的文字找到该关键字对应的那几个高亮对象并一一删除
        // 在文本中重新对该关键字上色，用不同颜色。
        if (e.getClickCount() == 2) {
```

```
7.              String s=jbl.getText();
8.              System.out.println(s);
9.              Highlighter hg=txt.getHighlighter();//获取文本总的高亮对象
10.             ArrayList<Object>arr=hightLight.get(s);
11.             for(Object obj:arr){
12.                 hg.removeHighlight(obj);
13.             }
14.             //获得下一个颜色重新绘制
15.             ColorState.put(s,(ColorState.get(s)+1)%ColorTrans.size()
    );
16.             DefaultHighlighter.DefaultHighlightPainter painter=new
17.                 DefaultHighlighter.DefaultHighlightPainter(ColorTran
    s.get(ColorState.get(s))); //设置高亮显示颜色为黄色
18.             //重新绘制高亮
19.             ArrayList<Object>tmp=new ArrayList<>();
20.             int index=0;
21.             String text=txt.getText(); //得到文本框的文本
22.             while((index=text.indexOf(str,index))>=0) {
23.                 try {
24.                     tmp.add(hg.addHighlight(index, index+str.length(
    ), painter));//高亮显示匹配到的词语
25.                     index+=str.length(); //更新匹配条件继续匹配
26.                 } catch (BadLocationException i) {
27.                     // TODO Auto-generated catch block
28.                     i.printStackTrace();
29.                 }
30.             }
31.             hightLight.put(str,tmp);
32.         }
33.     }
34.});
```

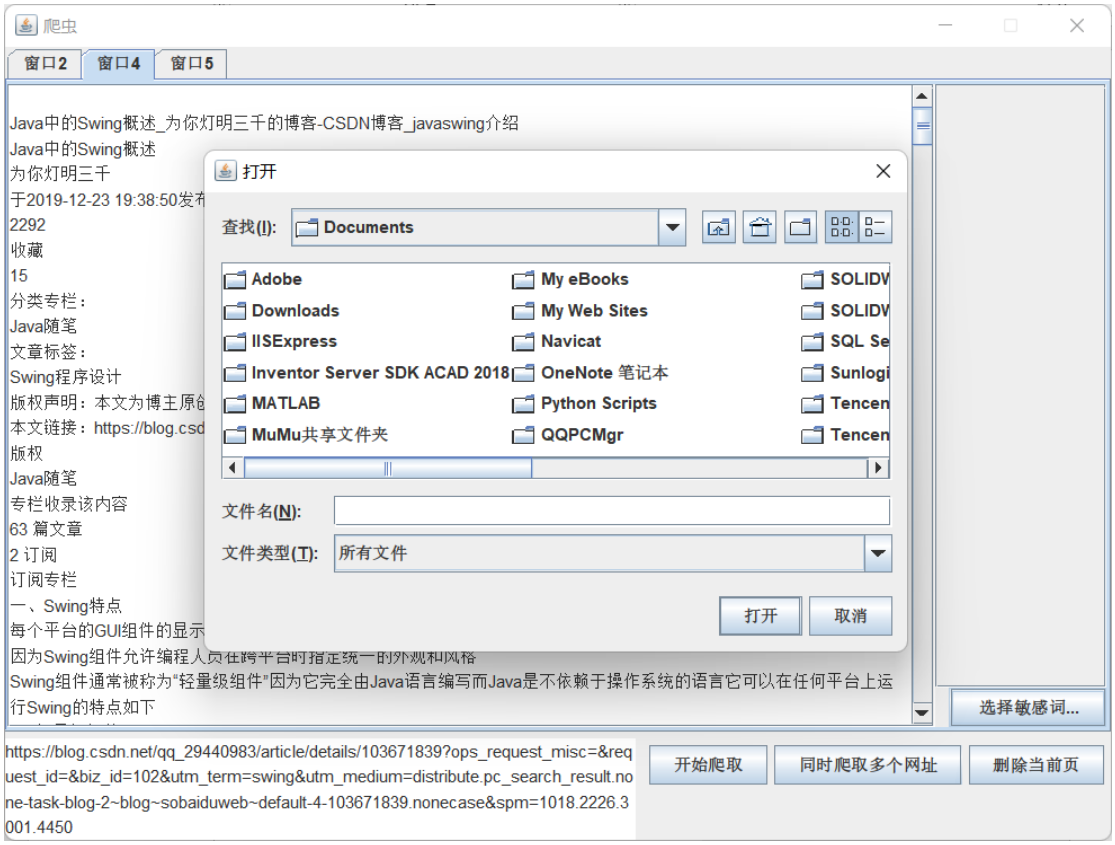

爬虫
窗口2
窗口4
窗口5
量组件的轻量组件外，Swing还提供了大量有助于开发图形用户界面的附加组件。
Java Swing 图形界面开发简介
热门推荐
XTS的专栏
05-30
16万+
Java Swing GUI 图形界面窗口开发基础教程，本教程将系统性地详细介绍
Swing 开发中常用的一些组件、布局管理器等相关知识技术，并且每章节都将通过代码实例展示实际应用。Swing 是 Java 为图形界面应用开发提供的一组工具包，是 Java 基础类的一部分。Swing 包含了构建图形界面（GUI）的各种组件，如: 窗口、标签、按钮、文本框等。Swing 提供了许多比 AWT 更好的屏幕显示元素，使用纯 Java 实现，能够更好的兼容跨平台运行。Java Swing 图形界面开发目录
Java Swing界面设计UI（全）
weixin_51201930的博客
01-21
3916
原文链接: http://blog.csdn.net/xietansheng/article/details/72814531
Java Swing GUI 图形界面窗口开发基础教程，本教程将系统性地详细介绍 Java Swing 开发中常用的一些组件、布局管理器等相关知识技术，并...
Java_Swing(一)Swing介绍
不安静的小田田的博客
06-03
4万+
Swing是一个为Java设计的GUI工具包，是新一代的图形界面工具。Swing组件的出现是为了解决AWT组件的移植性问题，同时提供更多的企业级应用程序所需要的界面功能，Swing组件中的类主要包含在javax.swing包中，JDK1.2以后的版本都包含Swing类库。
swing
123
111
222
用户界面
窗口
选择敏感词词库
https://blog.csdn.net/qq_29440983/article/details/103671839?ops_request_misc=&request_id=&biz_id=102&utm_term=swing&utm_medium=distribute.pc_search_result.none-task-blog-2~blog~sobaiduweb~default-4-103671839.nonecase&spm=1018.2226.3001.4450
开始爬取
同时爬取多个网址
删除当前页

7 双击换色效果

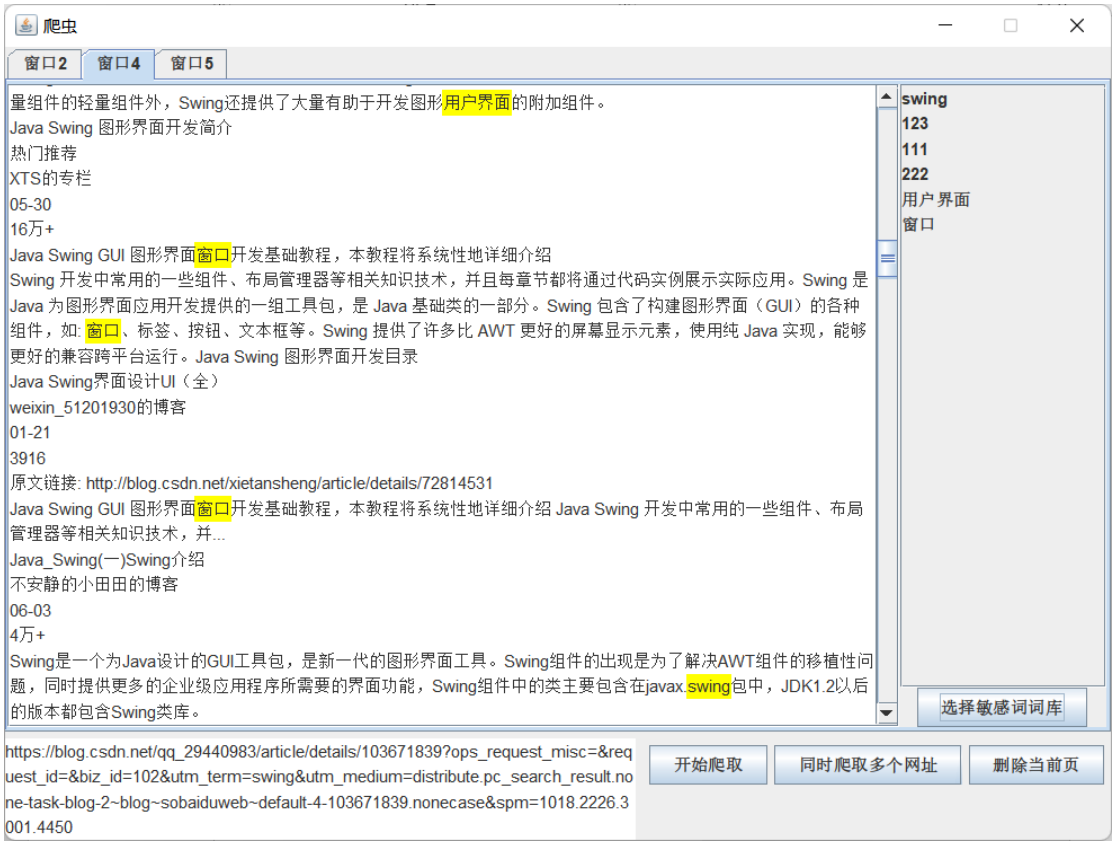
题目二的运行及测试结果

单个网页爬取效果图


爬虫
窗口2
窗口4
窗口5
Java中的Swing概述_为你灯明三千的博客-CSDN博客_javaswing介绍
Java中的Swing概述
为你灯明三千
于2019-12-23 19:38:50发布
2292
收藏
15
分类专栏：
Java随笔
文章标签：
Swing程序设计
版权声明：本文为博主原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明。
本文链接：https://blog.csdn.net/qq_29440983/article/details/103671839
版权
Java随笔
专栏收录该内容
63 篇文章
2 订阅
订阅专栏
一、Swing特点
每个平台的GUI组件的显示会有所不同但是在不同平台上运行使用Swing开发的应用程序时就可以统一GUI组件的显示风格
因为Swing组件允许编程人员在跨平台时指定统一的外观和风格
Swing组件通常被称为"轻量级组件"因为它完全由Java语言编写而Java是不依赖于操作系统的语言它可以在任何平台上运
行Swing的特点如下
选择敏感词...
https://blog.csdn.net/qq_29440983/article/details/103671839?ops_request_misc=&req
uest_id=&biz_id=102&utm_term=swing&utm_medium=distribute.pc_search_result.no
ne-task-blog-2~blog~sobaiduweb~default-4-103671839.nonecase&spm=1018.2226.3
001.4450
开始爬取
同时爬取多个网址
删除当前页


爬虫
窗口2
窗口4
窗口5
Java中的Swing概述_为你灯明三千的博客-CSDN博客_javaswing介绍
Java中的Swing概述
为你灯明三千
2292
收藏
15
分类专栏：
Java随笔
文章标签：
Swing程序设计
Java随笔
专栏收录该内容
63 篇文章
2 订阅
订阅专栏
一、Swing特点
Swing组件通常被称为"轻量级组件"因为它完全由Java语言编写而Java是不依赖于操作系统的语言它可以在任何平台上运
行Swing的特点如下
打开
查找(I):
Documents
Adobe
Downloads
IISExpress
Inventor Server SDK ACAD 2018
MATLAB
MuMu共享文件夹
My eBooks
My Web Sites
Navicat
OneNote 笔记本
Python Scripts
QQPCMgr
文件名(N):
文件类型(T):
所有文件
打开
取消
选择敏感词...
https://blog.csdn.net/qq_29440983/article/details/103671839?ops_request_misc=&req
uest_id=&biz_id=102&utm_term=swing&utm_medium=distribute.pc_search_result.no
ne-task-blog-2~blog~sobaiduweb~default-4-103671839.nonecase&spm=1018.2226.3
001.4450
开始爬取
同时爬取多个网址
删除当前页

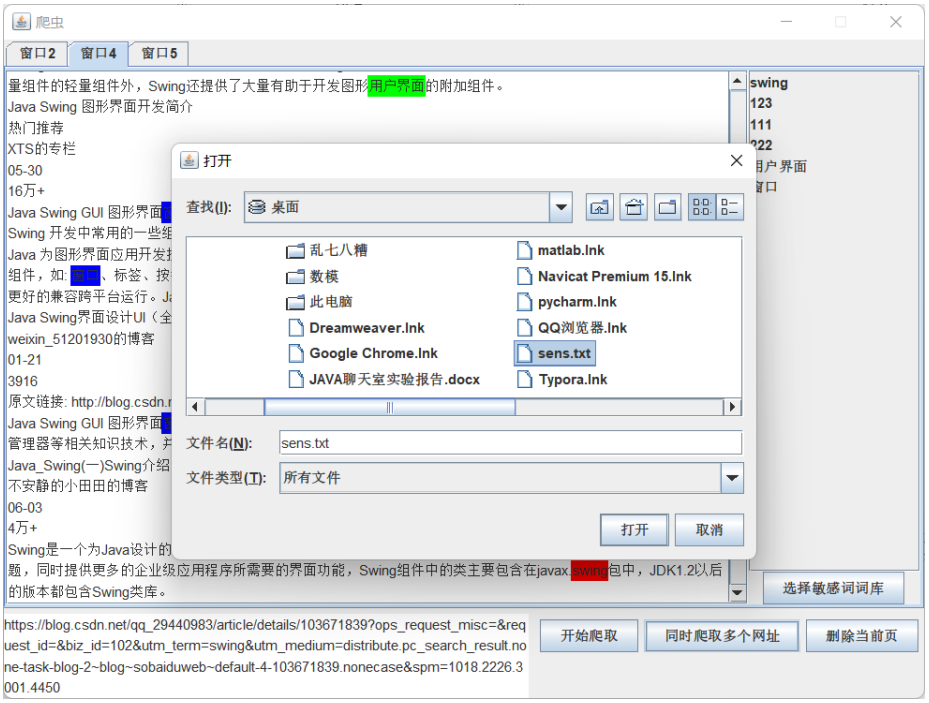
8 选择敏感词


爬虫
窗口2
窗口4
窗口5
量组件的轻量组件外，Swing还提供了大量有助于开发图形用户界面的附加组件。
Java Swing 图形界面开发简介
热门推荐
XTS的专栏
05-30
16万+
Java Swing GUI 图形界面窗口开发基础教程，本教程将系统性地详细介绍
Swing 开发中常用的一些组件、布局管理器等相关知识技术，并且每章节都将通过代码实例展示实际应用。Swing 是Java 为图形界面应用开发提供的一组工具包，是 Java 基础类的一部分。Swing 包含了构建图形界面（GUI）的各种组件，如: 窗口、标签、按钮、文本框等。Swing 提供了许多比 AWT 更好的屏幕显示元素，使用纯 Java 实现，能够更好的兼容跨平台运行。Java Swing 图形界面开发目录
Java Swing界面设计UI（全）
weixin_51201930的博客
01-21
3916
原文链接: http://blog.csdn.net/xietansheng/article/details/72814531
Java Swing GUI 图形界面窗口开发基础教程，本教程将系统性地详细介绍 Java Swing 开发中常用的一些组件、布局管理器等相关知识技术，并...
Java_Swing(一)Swing介绍
不安静的小田田的博客
06-03
4万+
Swing是一个为Java设计的GUI工具包，是新一代的图形界面工具。Swing组件的出现是为了解决AWT组件的移植性问题，同时提供更多的企业级应用程序所需要的界面功能，Swing组件中的类主要包含在javax.swing包中，JDK1.2以后的版本都包含Swing类库。
swing
123
111
222
用户界面
窗口
选择敏感词词库
https://blog.csdn.net/qq_29440983/article/details/103671839?ops_request_misc=&request_id=&biz_id=102&utm_term=swing&utm_medium=distribute.pc_search_result.none-task-blog-2~blog~sobaiduweb~default-4-103671839.nonecase&spm=1018.2226.3001.4450
开始爬取
同时爬取多个网址
删除当前页

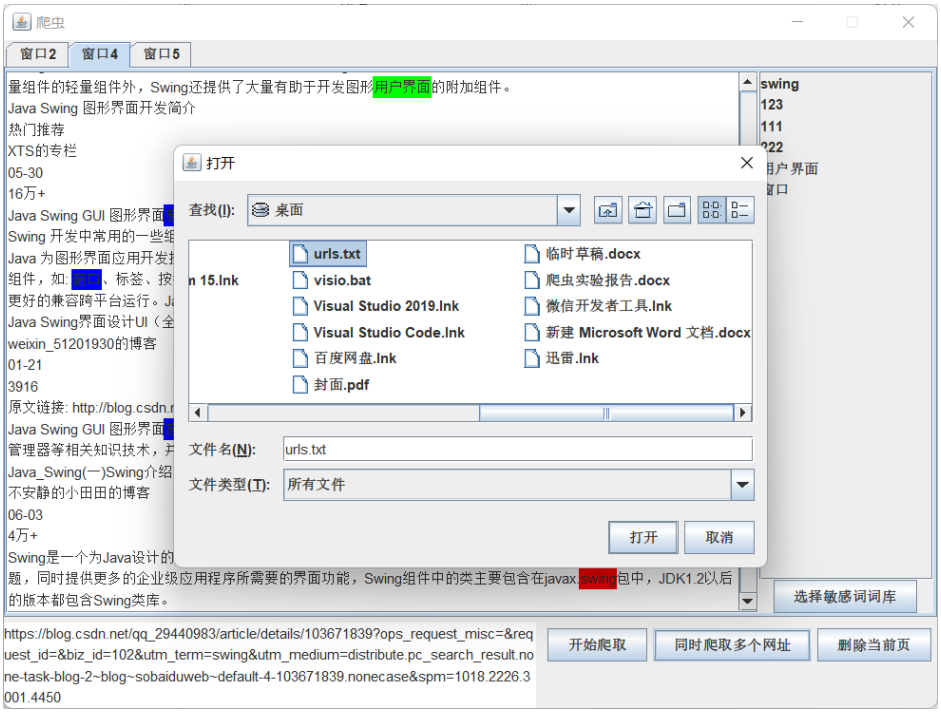

爬虫
窗口2
窗口4
窗口5
量组件的轻量组件外，Swing还提供了大量有助于开发图形用户界面的附加组件。
Java Swing 图形界面开发简介
热门推荐
XTS的专栏
05-30
16万+
Java Swing GUI 图形界面窗口开发基础教程，本教程将系统性地详细介绍
Swing 开发中常用的一些组件、布局管理器等相关知识技术，并且每章节都将通过代码实例展示实际应用。Swing 是Java 为图形界面应用开发提供的一组工具包，是 Java 基础类的一部分。Swing 包含了构建图形界面（GUI）的各种组件，如: 窗口、标签、按钮、文本框等。Swing 提供了许多比 AWT 更好的屏幕显示元素，使用纯 Java 实现，能够更好的兼容跨平台运行。Java Swing 图形界面开发目录
Java Swing界面设计UI（全）
weixin_51201930的博客
01-21
3916
原文链接: http://blog.csdn.net/xietansheng/article/details/72814531
Java Swing GUI 图形界面窗口开发基础教程，本教程将系统性地详细介绍 Java Swing 开发中常用的一些组件、布局管理器等相关知识技术，并...
Java_Swing(一)Swing介绍
不安静的小田田的博客
06-03
4万+
Swing是一个为Java设计的GUI工具包，是新一代的图形界面工具。Swing组件的出现是为了解决AWT组件的移植性问题，同时提供更多的企业级应用程序所需要的界面功能，Swing组件中的类主要包含在javax.swing包中，JDK1.2以后的版本都包含Swing类库。
swing
123
111
222
用户界面
窗口
选择敏感词词库
https://blog.csdn.net/qq_29440983/article/details/103671839?ops_request_misc=&request_id=&biz_id=102&utm_term=swing&utm_medium=distribute.pc_search_result.none-task-blog-2~blog~sobaiduweb~default-4-103671839.nonecase&spm=1018.2226.3001.4450
开始爬取
同时爬取多个网址
删除当前页

9 双击更换颜色

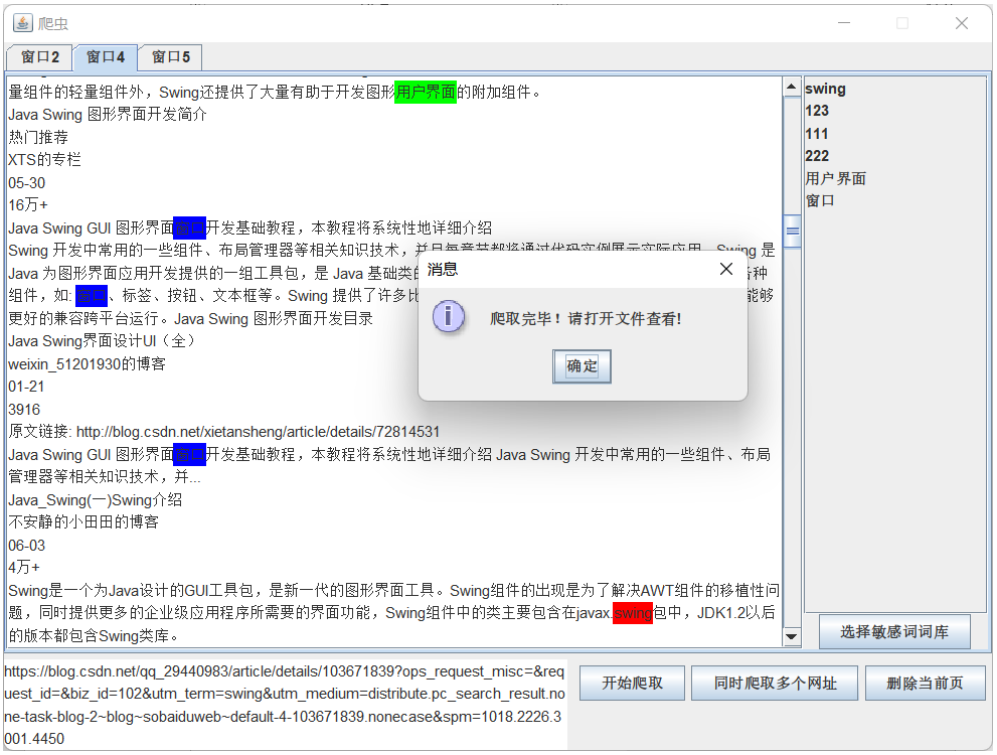
爬取多个网站

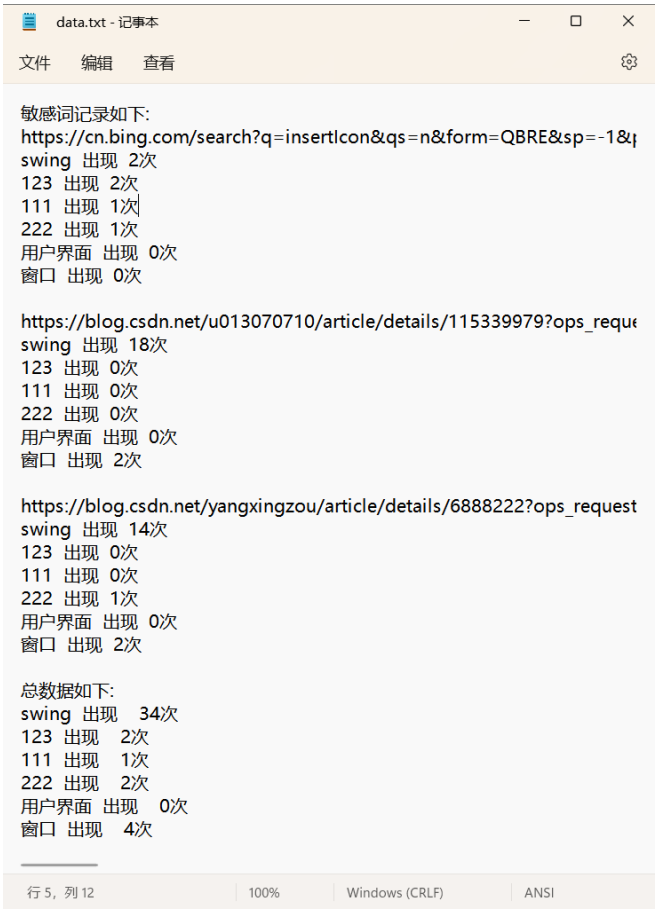


10 选择敏感词库



11 选择 URL 文件





12 爬取结果写入文本文件

总体心得体会

由于上学期没有学 JAVA，这次是边做边学的，这算是我写的第一个 java 程序了。感觉是从零开始学一路跌跌撞撞做了很久，很多时候辛辛苦苦写了一大片然后发现有很多更好的做法然后又改了一大片。总得来说收获非常大，从 JAVA 的基本语法和基本容器的使用到用 JAVA 连接数据库，JSON 传输，可视化编程，多线程编程，时间日期操作，网络通信 SOCKET 编程等很多方面的知识都有涉及到，学到了非常多。爬虫项目也是熟悉了 URL 类和 URLconnections 类和正则表达式的使用，以及文件流读取，字符串匹配等相关知识。同时也加强了图形化界面编程的训练，通过继承自定义控件，选项卡组件的使用等也更熟悉了。这是我这学期付出最多时间和精力的的一个项目了，看着程序从功能简陋界面粗糙结构混杂一点点变得功能齐全界面看得过眼程序健壮性也还可以的的程度，还是充满成就感的。