**JAVA课程设计**

**个人中心需求开发文档**

**目录**

[一、 个人中心的需求分析 1](#_Toc6066)

[（一） QQ空间的用户主要具备以下几个特点: 1](#_Toc856)

[（二）用户的需求有以下特点 1](#_Toc24001)

[二、项目概述 2](#_Toc106)

[三、 开发架构  2](#_Toc9463)

[（一）系统框架 2](#_Toc27433)

[（二）开发环境 3](#_Toc12470)

[四、功能框架图 4](#_Toc15505)

[系统功能的结构图 4](#_Toc21673)

[五、各功能模块的示意图和描述 5](#_Toc9687)

[（一）登录 5](#_Toc16536)

[（二） 注册 5](#_Toc30053)

[（三） 个人信息完善 7](#_Toc30723)

[（四） 个人中心 7](#_Toc15497)

[六、数据库设计 9](#_Toc15238)

[七、 设计及代码 10](#_Toc979)

[（一） 登录界面 10](#_Toc6730)

[（二）注册代码 13](#_Toc2877)

[（三）个人主页 19](#_Toc21520)

[（四）说说发表 28](#_Toc15488)

[（五）通讯录代码 31](#_Toc14499)

[（六）计算器代码 33](#_Toc26964)

[（七）查看说说 40](#_Toc22439)

[（八）个人信息完善 43](#_Toc31848)

[（九）背景音乐代码 49](#_Toc11303)

[（十）聊天室代码 51](#_Toc15185)

[（十一）数据库连接代码 57](#_Toc2385)

[（十二）图片浏览 59](#_Toc26822)

[（十三）图片设置代码 63](#_Toc31289)

1. **个人中心的需求分析**

腾讯QQ的出现极大限度的满足了人们与陌生人交友的需求，并且提供了极大的便利，所以在短时间内腾讯QQ就积累了一大批用户。当时接触网络的用户群体与今天相比更集中，主要都集中在学生群体和白领群体，基本以年轻人为主。当时正值个人博客热，很多人可以在网络上通过写个人博客的方式展示自己的才华，获得了大量的关注。在学习了JAVA的相关知识后，我们小组决定以QQ为参照，设计一个相对简单的个人中心。

但是不是所有人都有写作方面的才华，对于更多数人来说，在网上分享自己生活中的点滴，获得更多的关注并享受分享的乐趣就成了一种奢望。而QQ空间的出现，通过直接在QQ客户端上开放接口引入用户，给广大的普通网民提供了便利。而且走的是个性化空间+博客的路线，可以让每个人在满足分享自己生活中点滴的需求的同时，还加入了个性化的元素。当时推出的空间花藤、空间背景、播放器皮肤和个性化排版等功能，极大的满足了当时上网主流用户的需求。短时间内就积累了大量的用户，后来又不断的在此基础上开发个人相册、心情分享、QQ城市达人等功能，逐渐具备了早期SNS网站的雏形。

1. QQ空间的用户主要具备以下几个特点:
2. 追求个性，乐于分享和展示自我 

（2）有一定的经济来源，上网娱乐是他们的主要娱乐手段

（3）有稳定的上网时间 

（4） 互联网重度用户

（二）用户的需求有以下特点

（1）通过个人中心分享自己的心情以及即时状态，并和朋友互动、展示自己。

产品功能1： 日志、相册、说说、分享、个性装扮、个性头像、黄钻,通过空间看其他人的心情以及即时状态，并和朋友互动（满足窥探欲望、八卦心理、与朋友之间保持联系，乐于看到别人的分享）

产品功能2： 日志、相册、说说的转发和评论功能，而且该功能设定为只要发布用户是你的好友，你就可以看到他该条分享下来自所有用户的评论

产品功能3： 腾讯开放平台上自主研发和第三方提供的海量游戏及应用，并且通过QQ空间平台可以随时更新自己的游戏状态（默认开启）和与好友互动游戏

（2）上网玩游戏，并和朋友互动（游戏作为主要的娱乐项目，投入不多，获得特殊的社交满足感（游戏社交），投入与收获性价比高）。

**二、项目概述**

通过使用Java，设计一个个人中心，该个人中心包括以下几个功能：登录，注册，完善个人信息，个人用户的主界面（登录后），在主界面中功能包含：设置（背景音乐设置、背景图片的浏览、背景图片的设置）、通讯录（查看通讯录的好友）、聊天室（客户端、服务器端的上线）、个人工具（计算器、新建文件）、说说的管理（发表说说、查看说说）、修改个人信息（对昵称等基本信息的修改保存）等功能。

**三、 开发架构**

（一）系统框架

三层架构:

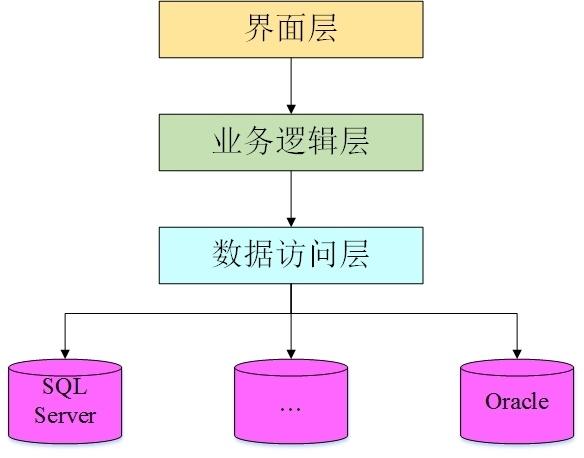
三层架构(3-tier architecture) 通常意义上的三层架构就是将整个业务应用划分为：表现（UI）、[业务逻辑层](http://baike.sogou.com/lemma/ShowInnerLink.htm?lemmaId=7792526&ss_c=ssc.citiao.link" \t "http://baike.sogou.com/_blank)（BLL）、数据访问层（DAL）。区分层次的目的即为了“高内聚，低耦合”的思想。

(1) 表现层（UI）：通俗讲就是展现给用户的界面，即用户在使用一个系统的时候他的所见所得。

(2) 业务逻辑层（BLL）：针对具体问题的操作，也可以说是对数据层的操作，对数据业务逻辑处理。

(3) 数据访问层（DAL）：该层所做事务直接操作数据库，针对数据的增添、删除、修改、查找等。

三层架构图如下：



（二）开发环境

Eclipse：

Eclipse是著名的跨平台的自由[集成开发环境](http://baike.sogou.com/lemma/ShowInnerLink.htm?lemmaId=156375&ss_c=ssc.citiao.link" \t "http://baike.sogou.com/_blank)（IDE）。最初主要用来Java语言开发，但是目前亦有人通过外挂程式使其作为其他[计算机语言](http://baike.sogou.com/lemma/ShowInnerLink.htm?lemmaId=664318&ss_c=ssc.citiao.link" \t "http://baike.sogou.com/_blank)比如C++和Python的开发工具。Eclipse本身只是一个框架平台，但是众多外挂程式的支持使得Eclipse拥有其他功能相对固定的IDE软体很难具有的灵活性。许多软体开发商以Eclipse为框架开发自己的IDE。Eclipse最初由[IBM公司](http://baike.sogou.com/lemma/ShowInnerLink.htm?lemmaId=498245&ss_c=ssc.citiao.link" \t "http://baike.sogou.com/_blank)开发，2001年11月贡献给[开源社区](http://baike.sogou.com/lemma/ShowInnerLink.htm?lemmaId=7825672&ss_c=ssc.citiao.link" \t "http://baike.sogou.com/_blank)，现在它由非营利软体供应商联盟Eclipse基金会（Eclipse [Foundation](http://baike.sogou.com/lemma/ShowInnerLink.htm?lemmaId=73807987&ss_c=ssc.citiao.link" \t "http://baike.sogou.com/_blank)）管理。

Microsoft SQL Server 2008 R2

结构化查询语言（英文简称：SQL）是一种特殊目的的编程语言，是一种数据库查询和[程序设计语言](http://baike.sogou.com/lemma/ShowInnerLink.htm?lemmaId=609078&ss_c=ssc.citiao.link" \t "http://baike.sogou.com/_blank)，用于存取数据以及查询、更新和管理关系[数据库系统](http://baike.sogou.com/lemma/ShowInnerLink.htm?lemmaId=277106&ss_c=ssc.citiao.link" \t "http://baike.sogou.com/_blank)；同时也是[数据库脚本文件](http://baike.sogou.com/lemma/ShowInnerLink.htm?lemmaId=71639443&ss_c=ssc.citiao.link" \t "http://baike.sogou.com/_blank)的[扩展名](http://baike.sogou.com/lemma/ShowInnerLink.htm?lemmaId=190718&ss_c=ssc.citiao.link" \t "http://baike.sogou.com/_blank)。

结构化查询语言是高级的非过程化编程语言，允许用户在高层数据结构上工作。它不要求用户指定对数据的存放方法，也不需要用户了解具体的数据存放方式，所以具有完全不同底层结构的不同 数据库系统,，可以使用相同的结构化查询语言作为数据输入与管理的接口。结构化查询语言语句可以嵌套，这使它具有极大的灵活性和强大的功能。

**四、功能框架图**

系统功能的结构图

QQ空间

登录界面

登录后的界面

登录

注册

通讯录

个

人

工具

设

置

聊天室

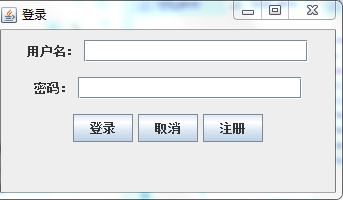
说说

完善个人资料

以上是系统的结构，用户进入登录界面后，先进行登录，然后注册，之后进入完善个人资料的界面。登录注册完成后进入个人主页，有设置、个人工具、说说，设置中有背景图片的设置，个人工具中有计算器和创建文件和发表说说。

**五、各功能模块的示意图和描述**

（一）登录



用户名

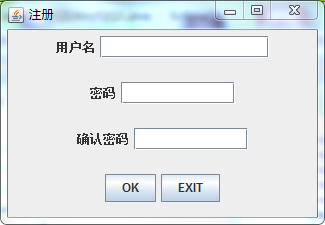
密码

登录

注册

如图登录界面包含：用户名、密码设置、登录、取消登录以及注册新用户。进入此界面会提示可先行注册，若已存在用户，可以进行登录输入用户名和密码以及确认密码之后进入完善个人信息完善的界面。

1. 注册



注册

密码是否符合要求

确认密码

用户名

密码

是

登录

否

重新登录

提示密码长度要4个以上，且密码必须包括数字及字母

界面包括用户名、密码及确认密码，输入用户名和密码进行注册，其中密码要求4个字符以上，且密码必须包含数字和字母，在填写完成后，点击OK注册，若放弃注册可点击exit,进行取消。

1. 个人信息完善



年龄段

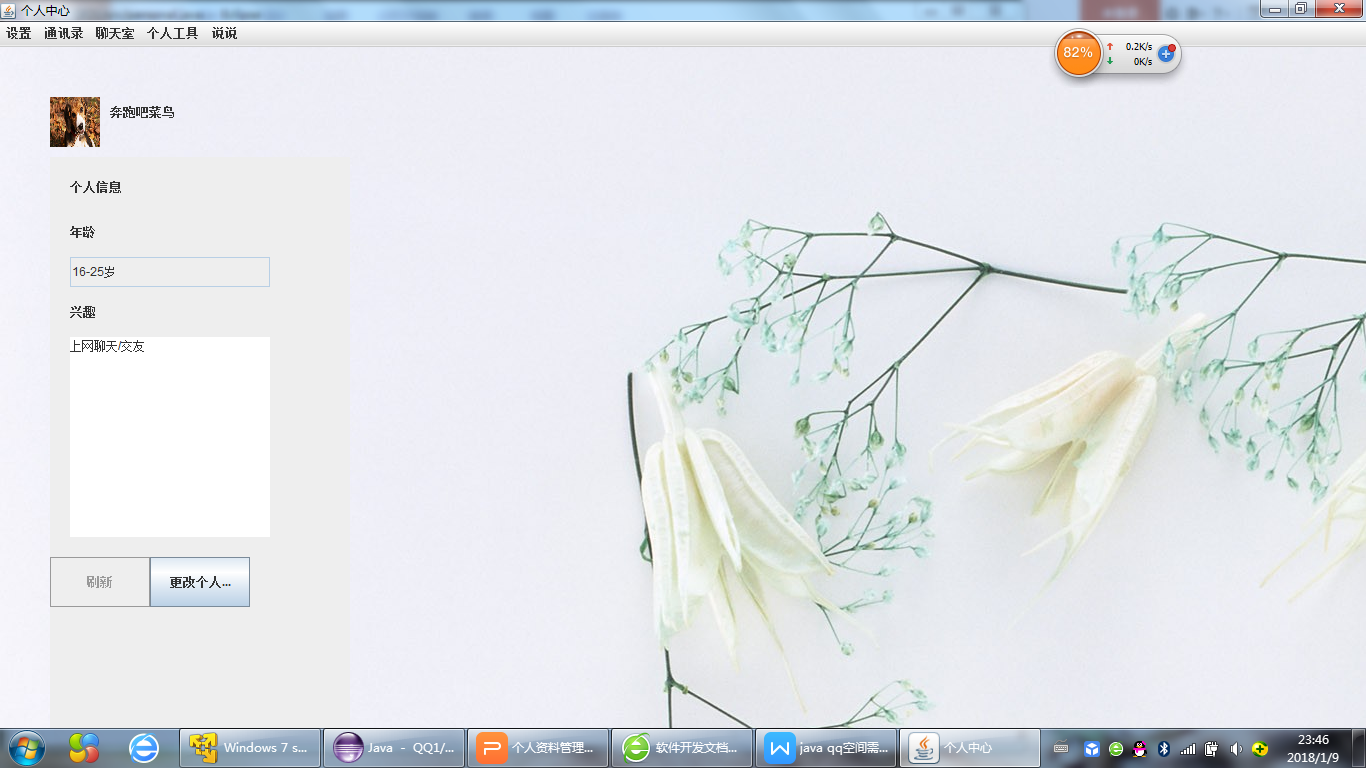
完善个人信息

登录

兴趣爱好

信息的完善包括年龄段的选择、兴趣爱好，在填写完后，点击提交将会，出现需对信息进行核对，若无问题，而提交成功。登录成功后会跳转到个人信息完善的界面，包括年龄段和兴趣爱好的信息完善。点击“退出”,后会进入个人主页。

1. 个人中心



登录

发表说说

个人信息完善

设置

播放音乐背景

设置图片背景

个人工具

查看说说

计算器

创建文件

个人信息完善后会跳转到个人中心页面，包括设置、通讯录、聊天室、个人工具、发表说说、个人信息的更改。其中设置有背景墙设置和预览以及播放背景音乐，个人工具包括计算器和创建文件，说说的管理有说说的发表和查看。

六、数据库设计

Microsoft SQL Server 2008 R2数据库表:

表1 tb\_denglu登录表

|  |  |  |  |
| --- | --- | --- | --- |
| 数据名称 | 数据表示 | 数据类型 | 数据长度 |
| 用户名 | Name | char | 15 |
| 密码 | Password | char | 10 |

表2 tb\_friends好友列表

|  |  |  |  |
| --- | --- | --- | --- |
| 数据名称 | 数据表示 | 数据类型 | 数据长度 |
| 好友用户名 | No | char | 10 |
| 好友昵称 | Name | varchar | 30 |
| 好友头像 | content | image | null |

表3 tb\_personal 个人信息完善表

|  |  |  |  |
| --- | --- | --- | --- |
| 数据名称 | 数据表示 | 数据类型 | 数据长度 |
| 用户名 | No | char | 10 |
| 昵称 | Name | varchar | 30 |
| 年龄段 | Age | varchar | 25 |
| 兴趣爱好 | Xinqu | varchar | 50 |
| 其他 | others | varchar | 100 |

表4 tb\_shuoshuo说说内容表

|  |  |  |  |
| --- | --- | --- | --- |
| 数据名称 | 数据表示 | 数据类型 | 数据长度 |
| 用户名 | No | char | 15 |
| 说说内容 | shuoshuo | varchar | max |
| 发表时间 | shijian | varchar | 30 |

1. 设计及代码
2. 登录界面

**import** java.awt.FlowLayout;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** java.sql.SQLException;

**import** javax.swing.JButton;

**import** javax.swing.JFrame;

**import** javax.swing.JLabel;

**import** javax.swing.JOptionPane;

**import** javax.swing.JPanel;

**import** javax.swing.JPasswordField;

**import** javax.swing.JTextField;

**public** **class** denglu **extends** JFrame **implements** ActionListener

{

/\*\*

\* **@param** args

\*/

JLabel name=**new** JLabel("用户名：");

JLabel passwrd=**new** JLabel("密码：");

JTextField nametext=**new** JTextField(20);

JPasswordField passwrdtext=**new** JPasswordField(20);

JButton but1=**new** JButton("登录");

JButton but2=**new** JButton("取消");

JButton but3=**new** JButton("注册");

JPanel jp1=**new** JPanel();

JPanel jp2=**new** JPanel();

JPanel jp3=**new** JPanel();

JPanel jp=**new** JPanel();

**public** denglu()

{

jp.setLayout(**new** FlowLayout());

name.setBounds(20,10,40,40);

jp1.add(name);

jp1.add(nametext);

jp.add(jp1);

jp2.add(passwrd);

jp2.add(passwrdtext);

jp.add(jp2);

but1.addActionListener(**this**);

but2.addActionListener(**this**);

but3.addActionListener(**this**);

jp3.add(but1);

jp3.add(but2);

jp3.add(but3);

jp.add(jp3);

**this**.add(jp);

**this**.setVisible(**true**);

**this**.setTitle("登录");

**this**.setBounds(100,100,350,200);

**this**.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);

}

**public** **void** actionPerformed(ActionEvent e)

{

String name;

String passwrd;

name=**new** String(nametext.getText().trim());

passwrd=**new** String(passwrdtext.getPassword());

**if**(e.getSource()==but1)

{

sql s=**new** sql();

**boolean** b=**false**;

String sql="select \* from tb\_denglu where Name=? and Password=?;";

**try** {

b=s.look(sql,name,passwrd);

} **catch** (SQLException e1) {

// **TODO** 自动生成的 catch 块

e1.printStackTrace();

}

s.closeconn();

**if**(b)

{

**int** n=JOptionPane.*showConfirmDialog*(**this**, "登录成功！是否先去完善个人信息？", "提示", JOptionPane.*YES\_NO\_OPTION*);

**if**(n==JOptionPane.*YES\_OPTION*)

{

**new** personal(name);

**this**.setVisible(**false**);

}

**else**

{

**this**.setVisible(**false**);

zhuye zhuye=**new** zhuye(name);

}

}

**else**

{

JOptionPane.*showMessageDialog*(**null**, "该用户不存在,请重新登录");

}

}

**if**(e.getSource()==but2)

{

passwrdtext.setText("");

System.*exit*(0);

}

**if**(e.getSource()==but3)

{

**this**.setVisible(**false**);

**new** zhuce();

}

}

}

（二）注册代码

import java.awt.Container;

import java.awt.GridLayout;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.sql.SQLException;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JPasswordField;

import javax.swing.JTextField;

public class zhuce extends JFrame implements ActionListener {

JTextField nameTextField;

JPasswordField passwordTextField;

JPasswordField conpassword;

JButton okButton;

JButton exitButton;

public zhuce()

{

super("注册");

JLabel name=new JLabel("用户名");

JLabel password=new JLabel("密码");

JLabel password2=new JLabel("确认密码");

okButton=new JButton("OK");

exitButton=new JButton("EXIT");

okButton.addActionListener(this);

exitButton.addActionListener(this);

nameTextField=new JTextField(15);

passwordTextField=new JPasswordField(10);

conpassword=new JPasswordField(10);

JPanel NamePanel=new JPanel();

NamePanel.add(name);

NamePanel.add(nameTextField);

JPanel passwordPanel1=new JPanel();

passwordPanel1.add(password);

passwordPanel1.add(passwordTextField);

JPanel passwordPanel2=new JPanel();

passwordPanel2.add(password2);

passwordPanel2.add(conpassword);

JPanel buttonPanel=new JPanel();

buttonPanel.add(okButton);

buttonPanel.add(exitButton);

Container container=this.getContentPane();

container.setLayout(new GridLayout(4,1));

container.add(NamePanel);

container.add(passwordPanel1);

container.add(passwordPanel2);

container.add(buttonPanel);

setSize(325,225);

setVisible(true);

}

public void actionPerformed(ActionEvent e)

{

if(e.getSource()==okButton)

{

validateDate();

}

if(e.getSource()==exitButton)

{

this.setVisible(false);

new denglu();

}

}

private void validateDate()

{

if(nameTextField.getText().equals("") || passwordTextField.getPassword().equals("") || conpassword.getPassword().equals(""))

{

JOptionPane.showMessageDialog(this,"请输入信息");

}

else if(!nameTextField.getText().matches("[a-zA-Z][a-zA-Z0-9]\*"))

{

JOptionPane.showMessageDialog(this, "用户名必须以字母开头");

}

else if(nameTextField.getText().length()>15)

JOptionPane.showMessageDialog(this,"用户名不可超过15个");

else if(passwordTextField.getPassword().length<4)

JOptionPane.showMessageDialog(this,"密码长度过短（4个以上）");

else if(!validatePassword())

JOptionPane.showMessageDialog(this,"密码必须包含数字及字母");

else if(!compareTwoPassword())

JOptionPane.showMessageDialog(this, "前后密码输入不一致");

else

{

sql s=new sql();

boolean b=false;

String asql=new String("insert tb\_denglu(Name,Password) values(?,?);");

String ssql=new String("select \* from tb\_denglu where Name=? and Password=?;");

String name=new String(nameTextField.getText());

String passwrd=new String(passwordTextField.getPassword());

try {

b=s.look(ssql,name,passwrd);

} catch (SQLException e) {

// TODO 自动生成的 catch 块

e.printStackTrace();

}

if(b)

{

JOptionPane.showMessageDialog(null, "该用户已存在，请重新注册");

}

else

{

try {

b=s.add(asql,name,passwrd);

} catch (SQLException e) {

// TODO 自动生成的 catch 块

e.printStackTrace();

}

if(b)

{

int n=JOptionPane.showConfirmDialog(this,"注册成功,是否去登录？","提示",JOptionPane.YES\_NO\_OPTION);

if(n==JOptionPane.YES\_OPTION)

{

s.closeconn();

new denglu();

this.setVisible(false);

}

}

}

}

}

private boolean validatePassword()

{

boolean b=false;

String password=new String(passwordTextField.getPassword());

b=password.matches("[a-zA-Z]+[0-9]+");

return b;

}

private boolean compareTwoPassword()

{

boolean b=false;

String password1=new String(passwordTextField.getPassword());

String password2=new String(conpassword.getPassword());

if(password1.equals(password2))

b=true;

return b;

}

}

（三）个人主页

import java.awt.\*;

import java.awt.event.\*;

import java.io.\*;

import java.sql.SQLException;

import java.util.\*;

import javax.swing.\*;

import lookphoto.\*;

import chatRoom.\*;

import makephoto.\*;

public class QQ1{

public static void main(String[] args) {

// TODO 自动生成的方法存根

new denglu();

}

}

class zhuye extends JFrame implements ActionListener

{

private JMenuBar mb=new JMenuBar();

private JMenu mm=new JMenu("设置");

private JMenu mm2=new JMenu("个人工具");

private JMenu mm3=new JMenu("说说");

private JMenu mtongxunlu=new JMenu("通讯录");

private JMenu mchat=new JMenu("聊天室");

private JMenuItem fabiao=new JMenuItem("发表说说");

private JMenuItem kanshuoshuo=new JMenuItem("查看说说");

private JMenuItem mi1=new JMenuItem("背景音乐");

private JMenuItem mi2=new JMenuItem("浏览背景图片");

JMenuItem mi5=new JMenuItem("确定更换背景");

JMenuItem mi3=new JMenuItem("计算器");

JMenuItem mi4=new JMenuItem("新建文件");

JMenuItem chatserver=new JMenuItem("服务器上线");

JMenuItem chatclient=new JMenuItem("客户端上线");

JMenuItem kantongxunlu=new JMenuItem("查看通讯录");

JLabel touxiang=new JLabel();

JLabel name=new JLabel();

ImageIcon image;

Image img;

JLabel imgLabel = new JLabel();

public String filename;

JPanel jp=new JPanel();

JButton changebut=new JButton("更改个人信息");

JButton shuaxin=new JButton("刷新");

JLabel age,xingqu,title;

JTextField tage=new JTextField(20);

JTextArea txingqu=new JTextArea(4,4);

String no;

Vector information=new Vector(3,3);

Container cp=this.getContentPane();

public zhuye(String No)

{

super("个人中心");

this.no=No;

name.setBounds(110, 50,500,30);

title=new JLabel("个人信息");

title.setBounds(20, 20, 200, 20);

age=new JLabel("年龄");

age.setBounds(20, 60, 100, 30);

tage.setBounds(20, 100, 200, 30);

tage.setEditable(false);

xingqu=new JLabel("兴趣");

xingqu.setBounds(20,140,60,30);

txingqu.setBounds(20, 180, 200, 200);

txingqu.setEditable(false);

sql s=new sql();

try {

information=s.selectpersonal("select \* from tb\_personal where No=?",no);

} catch (SQLException e1) {

// TODO 自动生成的 catch 块

e1.printStackTrace();

}

name.setText((String) information.get(0));

tage.setText((String)information.get(1));

txingqu.setText((String)information.get(2));

s.closeconn();

changebut.setBounds(100,400,100,50);

shuaxin.setBounds(0, 400, 100, 50);

shuaxin.setEnabled(false);

shuaxin.addActionListener(this);

changebut.addActionListener(this);

makephone("src/image/头像.jpg",0);

touxiang.setBounds(50, 50, 50, 50);

//touxiang.addMouseListener(new MouseAdapter(){

//public void MouseClicked(MouseEvent e){

// int n=JOptionPane.showConfirmDialog(null, "确定更换头像","提示", JOptionPane.YES\_NO\_OPTION);

//if(n==JOptionPane.YES\_OPTION)

// {

//new shangchuan();

// }

// }

//});

mm.add(mi1);

mm.add(mi2);

mm.add(mi5);

mb.add(mm);

mtongxunlu.add(kantongxunlu);

mb.add(mtongxunlu);

mchat.add(chatserver);

mchat.add(chatclient);

mb.add(mchat);

mm2.add(mi3);

mm2.add(mi4);

mb.add(mm2);

mm3.add(fabiao);

mm3.add(kanshuoshuo);

mb.add(mm3);

mi1.addActionListener(this);

mi2.addActionListener(this);

mi3.addActionListener(this);

mi4.addActionListener(this);

chatserver.addActionListener(this);

chatclient.addActionListener(this);

kantongxunlu.addActionListener(this);

fabiao.addActionListener(this);

kanshuoshuo.addActionListener(this);

makephone("src/image/49.jpg",1);

//beijing("src/image/49.jpg");

cp.setLayout(new BorderLayout());

((JPanel)cp).setOpaque(false); //注意这里，将内容面板设为透明。这样LayeredPane面板中的背景才能显示出来

jp.setLayout(null);

jp.setBounds(50,110,300,800);

jp.add(title);

jp.add(age);

jp.add(tage);

jp.add(xingqu);

jp.add(txingqu);

jp.add(changebut);

jp.add(shuaxin);

cp.add(jp);

cp.add(touxiang);

cp.add(name);//主页设计

this.setLayout(null);

this.setExtendedState( Frame.MAXIMIZED\_BOTH );

this.setJMenuBar(mb);

this.setVisible(true);

this.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

}

public void makephone(String filename,int i)

{

int width=50,height=50;

this.image=new ImageIcon(filename);

img=image.getImage();

if(i==0)

{

img = img.getScaledInstance(width, height, Image.SCALE\_DEFAULT);

image.setImage(img);

this.touxiang.setIcon(image);

}

else

{

image.setImage(img);

JLabel imgLabel = new JLabel(image);//将背景图放在标签里

this.getLayeredPane().add(imgLabel, new Integer(Integer.MIN\_VALUE));//注意这里是关键，将背景标签添加到jfram的LayeredPane面板里

imgLabel.setBounds(0,0,image.getIconWidth(), image.getIconHeight());//设置背景标签的位置

}

}

//public void beijing(String path)

//{

//this.path=path;

//ImageIcon img = new ImageIcon(path);//这是背景图片

//JLabel imgLabel = new JLabel(img);//将背景图放在标签里

//this.getLayeredPane().add(imgLabel, new Integer(Integer.MIN\_VALUE));//注意这里是关键，将背景标签添加到jfram的LayeredPane面板里

//imgLabel.setBounds(0,0,img.getIconWidth(), img.getIconHeight());//设置背景标签的位置

//}

public void setname(String filename)

{

this.name.setText(filename);

}

public void actionPerformed(ActionEvent e)

{

String filename=null;

if(e.getSource()==mi4)

new createfile();

if(e.getSource()==mi1)

{

new yinyue();

}

if(e.getSource()==mi2)

{

shangchuan check=new shangchuan();

}

if(e.getSource()==mi5)

{

shangchuan check1=new shangchuan(1);

filename=check1.getfilename();

if(filename!="")

{

makephoto m=new makephoto();

m.makebeijign(this, cp, filename);

}

else

{

JOptionPane.showMessageDialog(null, "更换失败");

}

}

if(e.getSource()==mi3)

new jisuanqi();

if(e.getSource()==fabiao)

new fabiao(no);

if(e.getSource()==kanshuoshuo)

{

new kanshuoshuo(no);

}

if(e.getSource()==kantongxunlu)

{

new friends();

}

if(e.getSource()==chatserver)

{

chatServer server=new chatServer();

}

if(e.getSource()==chatclient)

{

chatClient client=new chatClient();

}

if(e.getSource()==changebut)

{

personal p=new personal(no);

p.setText();

shuaxin.setEnabled(true);

}

if(e.getSource()==shuaxin)

{

sql s=new sql();

Vector inform=new Vector(3,3);

try {

inform=s.selectpersonal("select \* from tb\_personal where No=?",no);

} catch (SQLException e1) {

// TODO 自动生成的 catch 块

e1.printStackTrace();

}

name.setText((String) inform.get(0));

tage.setText((String)inform.get(1));

txingqu.setText((String)inform.get(2));

s.closeconn();

}

}

}

class ImagePanel extends JPanel

{

JLabel jl=new JLabel();

ImageIcon image;

Image img;

int width;

int height;

public void loadImage(String filename)

{

width=600;

height=600;

jl.setSize(width,height);

this.image=new ImageIcon(filename);

img=image.getImage();

img = img.getScaledInstance(width, height, Image.SCALE\_DEFAULT);

image.setImage(img);

this.jl.setIcon(image);

this.add(jl);

this.setVisible(true);

}

}

（四）说说发表

import java.awt.Color;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.sql.\*;

import java.text.\*;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JTextArea;

import java.util.Date;

public class fabiao extends JFrame implements ActionListener{

JPanel jfabiao=new JPanel();

JTextArea content;

JButton queding;

JLabel title=new JLabel("请输入内容：");

String no;

String time;

public fabiao(String no)

{

super("说说发表");

Date date=new Date();

DateFormat format=new SimpleDateFormat("yyyy-MM-dd");

time=format.format(date);

this.no=no;

title.setForeground(Color.white);

title.setBounds(20, 20, 200, 30);

content=new JTextArea(4,3);

content.setBackground(Color.white);

content.setBounds(20, 60, 600, 300);

queding=new JButton("发表");

queding.setBounds(20,400, 80,30);

queding.setBackground(Color.blue);

queding.addActionListener(this);

jfabiao.setLayout(null);

jfabiao.add(title);

jfabiao.add(content);

jfabiao.add(queding);

jfabiao.setBackground(Color.black );

jfabiao.setBounds(50,110,600,400);

this.add(jfabiao);

this.setBounds(400,80,650,500);

this.setVisible(true);

}

public void actionPerformed(ActionEvent e)

{

if(e.getSource()==queding)

{

sql s=new sql();

boolean b=false;

String sql=new String("insert tb\_shuoshuo(No,shuoshuo,shijian) values(?,?,?);");

try {

b=s.addshuoshuo(sql,no,content.getText(),time);

} catch (SQLException e1) {

// TODO 自动生成的 catch 块

e1.printStackTrace();

}

if(b)

{

JOptionPane.showMessageDialog(null, "发表成功");

content.setText(null);

}

}

}

}

（五）通讯录代码

import java.awt.BorderLayout;

import java.awt.Container;

import java.awt.Font;

import java.awt.Image;

import java.sql.SQLException;

import java.util.\*;

import javax.swing.\*;

public class friends extends JFrame {

JLabel[] jl,jnicheng;

JLabel title=new JLabel("好友列表");

sql s=new sql();

Container con;

public friends()

{

super("通讯录");

con=this.getContentPane();

Vector nicheng=new Vector(5,5);

try {

nicheng = s.getnicheng("select \* from tb\_friends;");

} catch (SQLException e) {

// TODO 自动生成的 catch 块

e.printStackTrace();

}

int n=nicheng.size();

title.setFont(new Font("宋体", Font.PLAIN, 30));

jl=new JLabel[n];

jnicheng=new JLabel[n];

for(int i=0;i<nicheng.size();i++)

{

try {

jl[i]=s.gettouxiang("select \* from tb\_friends where Name=?;",(String)nicheng.get(i));

} catch (ClassNotFoundException e) {

// TODO 自动生成的 catch 块

e.printStackTrace();

}

String name=(String)nicheng.get(i);

jnicheng[i]=new JLabel(name);

jnicheng[i].setFont(new Font("宋体", Font.PLAIN, 30));

}

for(int i=0;i<n;i++)

{

jl[i].setBounds(30, 50+i\*100, 50, 50);

this.add(jl[i]);

jnicheng[i].setBounds(100, 50+i\*100,300,50);

this.add(jnicheng[i]);

}

con.add(title,BorderLayout.NORTH);

this.setVisible(true);

this.setBounds(400, 20, 400, 700);

}

}

（六）计算器代码

import java.awt.Button;

import java.awt.Color;

import java.awt.Event;

import java.awt.FlowLayout;

import java.awt.GridLayout;

import java.awt.Panel;

import java.awt.TextField;

import javax.swing.JFrame;

public class jisuanqi extends JFrame{

TextField answerText;

// 显示输人和结果的文本域

Button pointButton,equalButton,plusButton;

Button minusButton,clearButton,multiButton,divisionButton;

//运算符按钮

Button[] b=new Button[10] ;//数字按钮

String currentOp,preOp;//当前操作和上一步操作

String foreText,backText;//当前输人和上一次输人

boolean isFloat= false;//运算类型标志

public jisuanqi ()

{ //实例化面板

Panel panel1=new Panel () ;//实例化面板

Panel panel2=new Panel () ;

Panel panel3=new Panel () ;

currentOp= new String("") ;//实例化各组件

preOp = new String("");

foreText = new String ("") ;

backText= new String("") ;

answerText = new TextField(20) ;

this.setBackground (Color.lightGray) ; //设置Applet 背景色

this.setForeground (Color.blue) ; //设置Applet 前景色

for(int i=9;i>=0;i--)

{

b[i]=new Button(Integer.toString(i) );

//实例化数字按钮

panel2.add(b[i]) ;//增加按钮到面板

}

pointButton = new Button(".") ;//实例化按钮

equalButton= new Button ("=") ;

equalButton.setForeground (Color.red) ;//设置按钮前景色

clearButton = new Button("清除") ;

clearButton.setForeground(Color.red) ;

divisionButton= new Button ("/") ;

divisionButton.setForeground(Color.red) ;

multiButton = new Button("\*") ;

multiButton.setForeground(Color.red) ;

minusButton = new Button ("-") ;

minusButton.setForeground (Color.red) ;

plusButton= new Button("+") ;

plusButton.setForeground(Color.red) ;

setLayout (new FlowLayout () ); //设置布局管理器

panel1.setLayout (new FlowLayout () );

panel2.setLayout (new GridLayout (4,3) );

panel3.setLayout (new GridLayout (4,1) );

panel1.add (answerText) ;

//增加组件到面板

panel1.add (clearButton) ;

panel2.add (pointButton) ;

panel2.add (equalButton) ;

panel3.add (plusButton) ;

panel3.add (minusButton) ;

panel3.add (multiButton) ;

panel3.add (divisionButton) ;

add (panel1) ;//增加组件到Applet

add (panel2);

add(panel3) ;

this.add(panel1);

this.add(panel2);

this.add(panel3);

this.setVisible(true);

this.setBounds(400,250,300,300);

}

public boolean action (Event e,Object o) {

//事件处理

String s=new String("") ;

for (int i=0;i<10;i++) {

if (e.target==b[i]||e.target==pointButton)

{ // 按钮事件来自于数字按钮和点按钮

if (e.target != pointButton)

{

s= (String)o;

doForeText(s);//处理输入;

}

if((e.target==pointButton)&&(!isFloat))

{ //浮点数输人

isFloat= true;//设置运算标志

s = (String) o;

if (foreText.equals (""))

{

foreText += "0.";

}

else

{

doForeText(s);

}

}

}

}

if ((e.target==multiButton)||(e.target==divisionButton)

||(e.target==plusButton)||(e.target==minusButton))

{//处理运算

if (foreText !="")

{

currentOp=((String)o);

doOperator();

}

else

{

preOp=((String)o);

}

}

if (e.target==equalButton)

{ //等于按钮事件处理

doOperator();//处理运算

}

if(e.target==clearButton)

doClear();

return true;

}

public void doOperator ()

{

double dFore,dBack;

Double d;

if(preOp.equals(""))

{

backText = foreText;

foreText = "";

answerText.setText (backText) ;//显示文本

}

else

{

dFore = (new Double (foreText)).doubleValue () ;//得到第一输入

dBack = (new Double (backText) ).doubleValue () ;//得到第二输入

foreText = "";

backText = answerText.getText () ;

if (preOp.equals ("+") )

{ //加运算处理

d= new Double ((dBack + dFore) );//得到显示结果

answerText.setText (d.toString () );//显示运算结果

backText = d.toString () ;

}

if(preOp.equals ("-") )

{

d = new Double ((dBack - dFore) ); //得到显示结果

answerText.setText (d.toString () ); //显示运算结果

backText = d.toString () ;

}

if(preOp.equals ("\*") )

{

d= new Double ((dBack \* dFore) );//得到显示结果

answerText.setText (d.toString () );//显示运算结果

backText = d.toString () ;

}

if(preOp.equals("/"))

{

if (dFore==0)

{

answerText.setText("除数不能为0");//显示出错信息

return;

}

d= new Double ((dBack / dFore) );//得到显示结果

answerText.setText (d.toString () );//显示运算结果

backText = d.toString () ;

}

}

preOp=currentOp;

}

public void doForeText(String s)

{

foreText += s;

answerText.setText (foreText) ;//显示输 人

}

public void doBackText (String s)

{

backText= foreText;

foreText = "" ;

answerText.setText (foreText) ;//显示输入

}

public void doClear () {//清除输入

currentOp= "";

preOp="";

foreText="";

backText= "";

isFloat= false;

answerText.setText ("") ;

}

}

（七）查看说说

import java.awt.\*;

import java.awt.event.\*;

import java.sql.SQLException;

import java.util.\*;

import javax.swing.\*;

import makephoto.\*;

public class kanshuoshuo extends JFrame implements ActionListener{

JTextArea shuoshuo;

JScrollPane js;

JButton jb[]={new JButton("刷新"),new JButton("退出")};

JPanel jp=new JPanel();

String no;

Vector vshuo;

public kanshuoshuo(String no)

{

super("说说内容");

this.no=no;

Container con=this.getContentPane();

shuoshuo=new shuobj("src/image/绿叶.jpg");

js=new JScrollPane(shuoshuo);

shuoshuo.setLineWrap(true);

shuoshuo.setFont(new Font("宋体", Font.PLAIN, 30));

shuoshuo.setEditable(false);

jp.setLayout(new FlowLayout());

for(int i=0;i<2;i++)

{

jb[i].addActionListener(this);

jp.add(jb[i]);

}

showshuo();

con.add(js,BorderLayout.CENTER);

con.add(jp,BorderLayout.SOUTH);

this.setBounds(200,100, 800, 600);

this.setVisible(true);

this.add(con);

}

public void actionPerformed(ActionEvent e)

{

if(e.getSource()==jb[0])

{

shuoshuo.setText(null);

showshuo();

}

if(e.getSource()==jb[1])

{

this.setVisible(false);

}

}

public void showshuo()

{

sql s=new sql();

try {

vshuo=s.selectshuoshuo("select \* from tb\_shuoshuo ss join tb\_personal p on ss.No=p.No;");

} catch (SQLException e) {

// TODO 自动生成的 catch 块

e.printStackTrace();

}

for(int i=0;i<vshuo.size();i++)

{

cshuoshuo cshuo=(cshuoshuo)vshuo.get(i);

shuoshuo.append(cshuo.getname()+"\n");

shuoshuo.append(cshuo.gettime()+"\n");

shuoshuo.append(cshuo.getcontent()+"\n");

shuoshuo.append("\n");

}

s.closeconn();

}

}

（八）个人信息完善

import java.awt.\*;

import java.awt.event.\*;

import java.sql.SQLException;

import makephoto.\*;

import javax.swing.\*;

public class personal extends JFrame implements ItemListener{

String age,xingqu;

JFrame jf=new JFrame();

JLabel label[]={new JLabel("昵称"),new JLabel("年龄段："),new JLabel("兴趣爱好 ："),new JLabel("其他：")};

JRadioButton jrage[]={new JRadioButton("5-15岁"),new JRadioButton("16-25岁"),new JRadioButton("26-35岁"),new JRadioButton("36-45岁"),new JRadioButton("46-55岁")};

JCheckBox jcxingqu[]={new JCheckBox("上网聊天/交友"),new JCheckBox("体育/户外健身"),new JCheckBox("汽车/购物"),new JCheckBox("旅游/度假"),new JCheckBox("时尚服装/化妆品")};

JButton jb[]={new JButton("提交"),new JButton("清空"),new JButton("退出")};

JTextArea others;

JTextField name=new JTextField(200);

ButtonGroup bg=new ButtonGroup();

JPanel jp,jpanel;

String no;

sql s=new sql();

boolean b=false;

Container con=jf.getContentPane();

public personal(String No)

{

this.no=No;

jpanel=new JPanel();

jp=new JPanel();

jp.setLayout(null);

name.setBounds(30, 50,200,30);

jp.add(name);

for(int i=0;i<4;i++)

{

label[i].setBounds(30,20+i\*80,80,30);

jp.add(label[i]);

}

for(int i=0;i<5;i++)

{

jrage[i].setBounds(40+i\*100,150,80,30);

jrage[i].addItemListener(this);

bg.add(jrage[i]);

jcxingqu[i].setBounds(40+i\*120,220,120,30);

jcxingqu[i].addItemListener(this);

jp.add(jrage[i]);

jp.add(jcxingqu[i]);

}

for(int i=0;i<3;i++)

{

jb[i].setBounds(450+i\*100,550,70,30);

jp.add(jb[i]);

}

others=new JTextArea(4,4);

others.setBounds(30, 300, 500,200);

jp.add(others);

jb[0].addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

int n=JOptionPane.showConfirmDialog(null,"个人信息："+" 昵称："+name.getText()+" 年龄为："+age+" 兴趣为："+xingqu+" 是否提交？","提示",JOptionPane.YES\_NO\_OPTION);

if(n==JOptionPane.YES\_OPTION)

{

boolean b=false;

String sql=new String();

if(jb[0].getText()=="提交")

{

sql="insert tb\_personal(Name,Age,Xinqu,others,No) values(?,?,?,?,?);";

try {

b=s.addpersonal(sql,no, name.getText(),age,xingqu,others.getText());

} catch (SQLException e) {

// TODO 自动生成的 catch 块

e.printStackTrace();

}

jb[0].setText("更改");

jb[2].setText("去主页");

if(b)

{

JOptionPane.showMessageDialog(jpanel, "提交成功","提示",JOptionPane.WARNING\_MESSAGE);

}

}

else if(jb[0].getText()=="更改")

{

sql="update tb\_personal set Name=?,Age=?,Xinqu=?,others=? where No=?;";

try {

b=s.addpersonal(sql, no, name.getText(), age, xingqu, others.getText());

} catch (SQLException e) {

// TODO 自动生成的 catch 块

e.printStackTrace();

}

if(b)

{

JOptionPane.showMessageDialog(jpanel, "更改成功","提示",JOptionPane.WARNING\_MESSAGE);

}

}

}

s.closeconn();

}

});

jb[1].addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

bg.clearSelection();

for(int i=0;i<5;i++){

jcxingqu[i].setSelected(false);

}

}

});

jb[2].addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent event)

{

if(jb[2].getText()=="去主页")

{

jf.setVisible(false);

new zhuye(no);

}

else if(jb[2].getText()=="退出")

{

jf.setVisible(false);

}

}

});

con.add(jp);

jf.setBounds(200,50,750,700);

jf.setVisible(true);

jf.setTitle("个人信息填写");

}

public void itemStateChanged(ItemEvent e)

{

age=new String();

xingqu=new String();

for(int i=0;i<5;i++)

{

if(jrage[i].isSelected())

{

age+=jrage[i].getText();

}

if(jcxingqu[i].isSelected())

{

xingqu+=jcxingqu[i].getText();

}

}

}

public void setText()

{

jb[0].setText("更改");

}

}

（九）背景音乐代码

import java.applet.Applet;

import java.applet.AudioClip;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.ItemEvent;

import java.awt.event.ItemListener;

import java.io.File;

import java.net.\*;

import javax.swing.\*;

public class shengyin extends Applet implements ActionListener,ItemListener {

AudioClip ac;

Button button\_play,button\_loop,button\_stop;

String mname="那个男孩";

JComboBox cbmusic=new JComboBox();

Label title=new Label("请选择音乐");

JButton ok=new JButton("确定");

JPanel jp=new JPanel();

public void init () {

add(title);

title.setBounds(100, 100, 100, 30);

cbmusic.addItem("那个男孩");

cbmusic.addItem("像风一样");

cbmusic.setVisible(true);

cbmusic.setSize(300,30);

cbmusic.setBounds(60, 40, 250, 30);

cbmusic.addItemListener(this);

add(cbmusic);

add(ok);

ok.setBounds(320, 40, 70, 30);

ok.addActionListener(this);

Label label= new Label ("音乐载入成功!点击按钮开始播放!") ;

jp.add(label) ;

button\_play= new Button ("开始播放") ;// 添加按钮控制

button\_loop= new Button ("循环播放") ;

button\_stop= new Button ("停止播放") ;

button\_play.addActionListener (this) ;// 按钮添加事件监听器

button\_stop.addActionListener (this) ;

button\_loop.addActionListener (this) ;

jp.add (button\_play) ;

jp.add (button\_loop) ;

jp.add (button\_stop) ;

jp.setSize(100,100);

jp.setBounds(60, 80, 250, 300);

jp.setVisible(false);

add(jp);

setLayout(null);

setSize(400,300);

}

public void actionPerformed (ActionEvent e){ // 实现监听器方法

if (e.getSource () == button\_play) {

ac.play();// 开始播放

}else if (e.getSource () == button\_loop) {

ac.loop();// 循环播放

}

if (e.getSource ()== button\_stop) {

ac.stop();// 停止播放

}

if(e.getSource()==ok)

{

ac=this.getAudioClip(this.getCodeBase(), "music/"+this.mname+".wav");

jp.setVisible(true);

}

}

public void itemStateChanged(ItemEvent e)

{

if(e.getStateChange()==ItemEvent.SELECTED)

{

this.mname=(String)cbmusic.getSelectedItem();

}

}

}

（十）聊天室代码

服务器端代码：

import java.awt.\*;

import java.awt.event.\*;

import java.io.\*;

import java.net.\*;

import javax.swing.\*;

public class chatServer implements ActionListener,Runnable {

JTextArea showArea;

JTextField msgText;

JFrame mainJframe ;

JButton sentBtn;

Container con;

JScrollPane JSPane;

JPanel pane;

ServerSocket serverSocket;

Socket connectToclient;

DataInputStream inFromClient;

DataOutputStream outToclient;

Thread thread;

public chatServer () { // 设置界面

mainJframe=new JFrame ("聊天一一服务器端") ;

con=mainJframe.getContentPane () ;

showArea=new JTextArea () ;//创建文本域

showArea.setEditable (false) ;

showArea.setLineWrap (true) ;

JSPane=new JScrollPane (showArea) ;// 创建滚动面板

msgText=new JTextField () ;// 创建文本框

msgText.setColumns (30) ;

msgText.addActionListener (this) ;

sentBtn=new JButton ("发送") ;// 创建发送按钮

sentBtn.addActionListener(this) ;

pane=new JPanel () ;

pane.setLayout (new FlowLayout ()) ;// 进行组件布局管理

pane.add (msgText) ;

pane.add (sentBtn) ;

con.add (JSPane,BorderLayout.CENTER) ;

con.add (pane,BorderLayout.SOUTH) ;

mainJframe.setSize(500,400) ;

mainJframe.setVisible (true) ;

mainJframe.setDefaultCloseOperation (JFrame.EXIT\_ON\_CLOSE) ;

try{

serverSocket=new ServerSocket (5500) ;// 创建服务套接字

showArea.append("正在等待对话请求\n") ;

connectToclient=serverSocket.accept () ;// 侦听客户端的连接

inFromClient=new DataInputStream (connectToclient.getInputStream ()) ;

outToclient=new DataOutputStream (connectToclient.getOutputStream() );

thread=new Thread (this) ; // 启动线程在后台来接收对方的消息

thread.setPriority (Thread.MIN\_PRIORITY) ;

thread.start () ;

}catch (IOException e) {

showArea.append ("对不起，不能创建服务器\n") ;

msgText.setEditable (false) ;

sentBtn.setEnabled(false) ;

}

}

public void actionPerformed (ActionEvent e) {// 响应按钮事件，发送消息给对方

String s=msgText.getText () ;

if (s.length()>0) {

try{

outToclient.writeUTF(s) ;

outToclient.flush() ;

showArea.append ("我说: "+msgText.getText () +"\n") ;

msgText.setText (null) ;

}

catch (IOException el) {

showArea.append ("你的消息: “"+msgText.getText () +"” 未能发送出去\n" );

}

}

}

public void run() { // 本线程负责将客户机传来的信息显示在对话区域

try{

while(true) {

showArea.append ("对方说: "+inFromClient.readUTF () +"\n") ;

Thread.sleep(1000) ;

}

}catch (IOException el) {

}catch (InterruptedException e){ }

}

}

客户端代码：

package chatRoom;

import java.awt.\*;

import java.awt.event.\*;

import java.io.\*;

import java.net.\*;

import javax.swing.\*;

public class chatClient implements ActionListener,Runnable{

JTextArea showArea;

JTextField msgText;

JFrame mainJframe ;

JButton sentBtn;

JScrollPane JSPane;

JPanel pane;

Container con;

Thread thread=null;

Socket connectToServer;

DataInputStream inFromServer;

DataOutputStream outToServer;

public chatClient () {

mainJframe=new JFrame ("聊天一客户端") ;

con=mainJframe.getContentPane () ;

showArea=new JTextArea () ;

showArea.setEditable (false) ;

showArea.setLineWrap (true) ;

JSPane=new JScrollPane (showArea) ;// 创建滚动面板

msgText=new JTextField () ;// 创建文本框

msgText.setColumns (30) ;

msgText.addActionListener (this) ;

sentBtn=new JButton ("发送") ;// 创建发送按钮

sentBtn.addActionListener(this) ;

pane=new JPanel () ;

pane.setLayout (new FlowLayout ()) ;// 进行组件布局管理

pane.add (msgText) ;

pane.add (sentBtn) ;

con.add (JSPane,BorderLayout.CENTER) ;

con.add (pane,BorderLayout.SOUTH) ;

mainJframe.setSize (500,400) ;

mainJframe.setVisible (true) ;

try{ // 创建套接字连接到服务器

connectToServer=new Socket ("localhost",5500) ;

inFromServer=new DataInputStream (connectToServer.getInputStream() );

outToServer=new DataOutputStream (connectToServer.getOutputStream() );

showArea.append ("连接成功，请说话\n") ;

thread=new Thread (this) ;// 创建线程在后台处理对方的消息

thread.setPriority (Thread.MIN\_PRIORITY) ;

thread.start () ;

}catch (IOException e) {

showArea.append ("对不起，没能连接到服务器\n" );

msgText.setEditable (false) ;

sentBtn.setEnabled (false) ;

}

}

public void actionPerformed (ActionEvent e) { //响应按钮事件，发送消息给对方

String s=msgText.getText () ;

if (s.length()>0) {

try{

outToServer.writeUTF (s) ;

outToServer.flush () ;

showArea.append ("我说: "+msgText.getText () +"\n") ;

msgText.setText (null) ;

}catch (IOException el) {

showArea.append ("你的消息: “"+msgText.getText () +"”未能发送出去\n") ;

}

}

}

public void run () { // 本线程负责将服务器传来的信息显示在对话区域

try{

while(true) {

showArea.append ("对方说: "+inFromServer.readUTF() +"\n") ;

Thread.sleep (1000) ;

}

}catch (IOException el) {

}catch (InterruptedException e){ }

}

}

（十一）数据库连接代码

package com.lgq.utils;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

public class DBUtil {

static{

try {

Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");

} catch (ClassNotFoundException e) {

e.printStackTrace();

}

}

public static Connection getConnetion() throws SQLException, ClassNotFoundException{

return DriverManager.getConnection("jdbc:sqlserver://192.168.140.129;databaseName=EduBase1;","java","123");

}

public static void release(Connection con,PreparedStatement stmt,ResultSet rs){

if(rs!=null){

try {

rs.close();

} catch (SQLException e) {

e.printStackTrace();

}

}

if(stmt!=null){

try {

stmt.close();

} catch (SQLException e) {

e.printStackTrace();

}

}

if(con!=null){

try {

con.close();

} catch (SQLException e) {

e.printStackTrace();

}

}

}

}

（十二）图片浏览

package lookphoto;

import java.awt.BorderLayout;

import java.awt.Container;

import java.awt.Frame;

import java.awt.Image;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.File;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFileChooser;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

public class shangchuan extends JFrame implements ActionListener

{

static String fileName;

JFrame jf=new JFrame("图片预览");

JButton buttonFile=new JButton("预览");

ImagePanel imagePane=new ImagePanel();

JButton buttonok=new JButton("确定");

JLabel jl=new JLabel();

public shangchuan()

{

buttonok.addActionListener(this);

buttonFile.addActionListener(this);

JPanel jpanel=new JPanel();

jpanel.add(buttonok);

Container content=jf.getContentPane();

content.add(buttonFile,BorderLayout.NORTH);

content.add(buttonok,BorderLayout.SOUTH);

content.add(imagePane,BorderLayout.CENTER);

jf.setBounds(150, 150, 800, 700);

jf.setVisible(true);

}

public shangchuan(int i)

{

}

public void actionPerformed(ActionEvent e)

{

String filename=null;

if(e.getSource()==buttonFile)

{

JFileChooser choose=new JFileChooser();

choose.setFileSelectionMode(JFileChooser.FILES\_ONLY);

choose.setCurrentDirectory(new File("."));

choose.setFileFilter(new javax.swing.filechooser.FileFilter()

{

public boolean accept(File file){

String name=file.getName().toLowerCase();

return name.endsWith(".gif") || name.endsWith(".jpg") || name.endsWith(".jpeg") || file.isDirectory();

}

public String getDescription()

{

return "图像文件";

}

});

int result =choose.showOpenDialog(this);

if(result==JFileChooser.APPROVE\_OPTION)

{

fileName=choose.getSelectedFile().getAbsolutePath();

imagePane.loadImage(fileName);

}

}

if(e.getSource()==buttonok)

{

if(fileName!="")

{

this.fileName=filename;

this.setVisible(false);

}

}

}

public String getfilename()

{

return this.fileName;

}

}

class ImagePanel extends JPanel

{

JLabel jl=new JLabel();

ImageIcon image;

Image img;

int width;

int height;

public void loadImage(String filename)

{

width=600;

height=600;

jl.setSize(width,height);

this.image=new ImageIcon(filename);

img=image.getImage();

img = img.getScaledInstance(width, height, Image.SCALE\_DEFAULT);

image.setImage(img);

this.jl.setIcon(image);

this.add(jl);

this.setVisible(true);

}

}

（十三）图片设置代码

背景图片设置：

package makephoto;

import java.awt.\*;

import javax.swing.\*;

public class makephoto extends JFrame

{

ImageIcon image;

Image img;

public void makephone(JLabel jl,String filename)

{

int width=50,height=50;

this.image=new ImageIcon(filename);

img=image.getImage();

img = img.getScaledInstance(width, height, Image.SCALE\_DEFAULT);

image.setImage(img);

jl.setIcon(image);

}

public void makebeijign(JFrame jf,Container con,String filename)

{

image=new ImageIcon(filename);

img=image.getImage();

image.setImage(img);

JLabel imgLabel = new JLabel(image);//将背景图放在标签里

jf.getLayeredPane().add(imgLabel, new Integer(Integer.MIN\_VALUE));//注意这里是关键，将背景标签添加到jfram的LayeredPane面板里

imgLabel.setBounds(0,0,image.getIconWidth(), image.getIconHeight());//设置背景标签的位置

((JPanel)con).setOpaque(false);

}

}

说说背景图设置：

package makephoto;

import java.awt.Graphics;

import java.awt.Image;

import javax.swing.ImageIcon;

import javax.swing.JTextArea;

public class shuobj extends JTextArea

{

Image img;

public shuobj(String filename)

{

ImageIcon imageIcon=new ImageIcon(filename);

img=imageIcon.getImage();

setOpaque(false);

}

public void paint(Graphics g)

{

g.drawImage(img,0,0,this);

super.paint(g);

}

}