

Java程序设计

实验报告



题目 幸运观众手机号码抽取器

姓名 胡皓睿

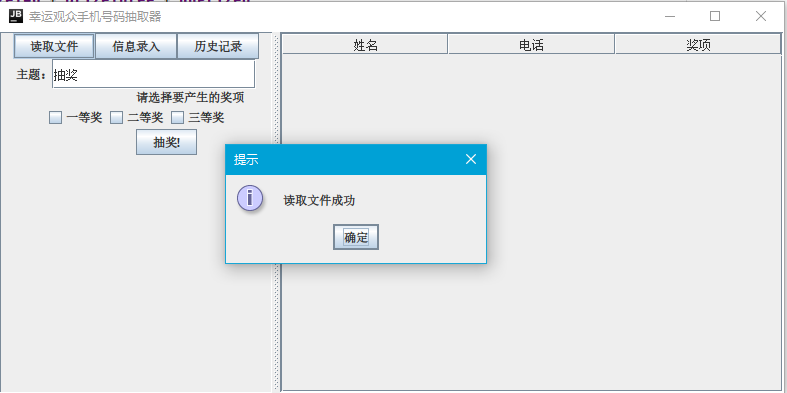
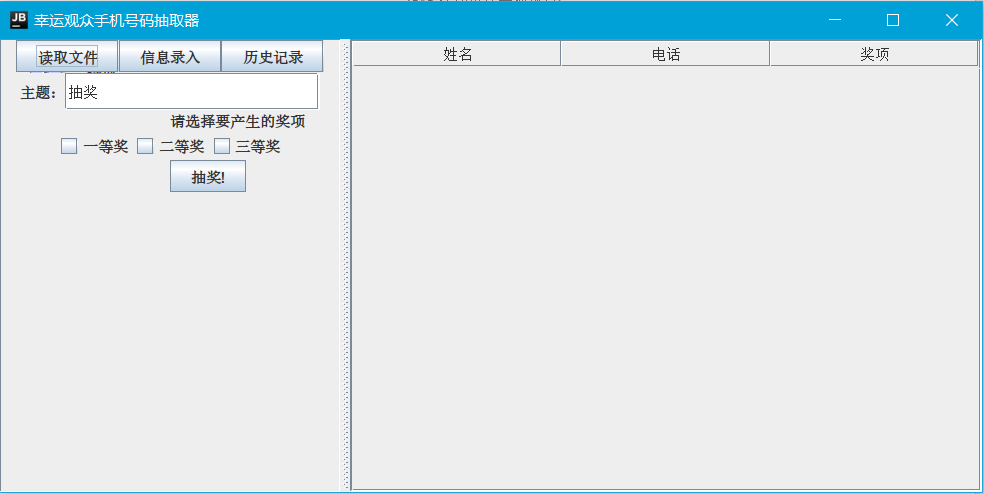
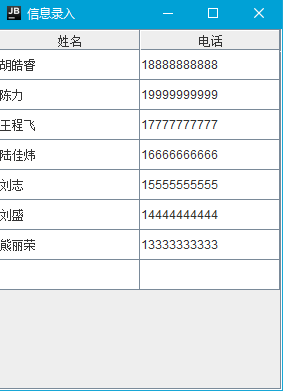
学号 201806061108

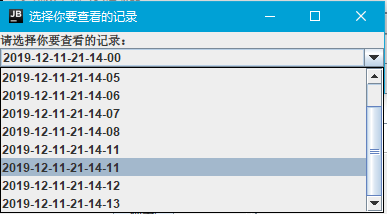
班级 计实1801

任课老师 胡皓睿

提交日期 2019年12月11日

1. **程序运行截图**

********

****

****

1. **程序包含的类**
   1. **FrameMain**

**程序的主界面，通过JButton可以进入其他界面**

* 1. **FrameAddPeople,FrameChooseHistory,FrameVievHistory**

**三个用户窗体，实现对应的功能**

* 1. **People**

**存储每个人的个人信息，提供了读取和设置信息的接口**

* 1. **Pool**

**奖池类，用于生成抽奖结果，提供了生成奖项的函数**

* 1. **Main**

**程序的入口**

* 1. **类之间的交互**

**程序由Main类进入，获取屏幕参数，调用FrameMain的构造函数，使主用户窗体在屏幕的正中间显示。FrameMain窗体中有三个JButton,通过ActionListener监听点击事件，分别实现对应的功能。在录入和使用人的信息时，使用People类的构造函数和getXXX方法，获取需要的数据。在抽奖时，使用Pool类的getPrize方法获取抽奖结果。**

1. **代码难点**
   1. **JLabel添加新项**

DefaultTableModel model = (DefaultTableModel) **display**.getModel

**添加新的项目时，需要使用JLabel的TableModel,但是直接通过getModel方法的model并不能使用，而是要通过强制类型转换变为DefaultTableModel，才能操作**

* 1. **交互逻辑的优化**

**在鼠标和键盘之间经常切换势必会造成软件使用效率的下降，所以我希望代码能够尽量减少这种情况。所以我使用了一个ComboBox来实现获奖人数的输入。**

1. **课程学习体会**

**Java是一门面向对象编程语言，不仅吸收了C++语言的各种优点，还摒弃了C++里难以理解的多继承、指针等概念，因此Java语言具有功能强大和简单易用两个特征。Java语言作为静态面向对象编程语言的代表，极好地实现了面向对象理论，允许程序员以优雅的思维方式进行复杂的编程。通过课程设计，我大致了解了Java的逻辑，体验到了面向对象的真谛，加深了我对代码学习的兴趣。通过对Swing的简单了解，我学会了使用Java实现简单的图形界面，为我以后使用其他方法实现图形界面打下了基础。**

1. **代码**
   1. **Main.java**

**import** java.awt.\*;  
  
**public class** Main {  
 **public static void** main(String[] args) {  
 Dimension screenSize = Toolkit.*getDefaultToolkit*().getScreenSize();  
 **int** x = (**int**) ((screenSize.getWidth() - 800) / 2);  
 **int** y = (**int**) ((screenSize.getHeight() - 400) / 2);  
 FrameMain frame = **new** FrameMain(x, y);  
 }  
}

* 1. **FrameMain.java**

**import** javax.swing.\*;  
**import** javax.swing.table.DefaultTableModel;  
**import** java.awt.\*;  
**import** java.awt.event.ActionEvent;  
**import** java.awt.event.ActionListener;  
**import** java.io.\*;  
**import** java.text.SimpleDateFormat;  
**import** java.util.\*;  
  
**public class** FrameMain **extends** JFrame **implements** ActionListener {  
 **private** JButton **ButtonReadFile**, **ButtonLottery**, **ButtonSetting**, **ButtonChooseHistory**;  
 **private** JCheckBox[] **Box\_AwardNum** = {**new** JCheckBox(**"一等奖"**), **new** JCheckBox(**"二等奖"**), **new** JCheckBox(**"三等奖"**)};  
 **private** JLabel[] **LabelAwards** = {**new** JLabel(**"一等奖个数："**), **new** JLabel(**"二等奖个数："**), **new** JLabel(**"三等奖个数："**)};  
 **private** JComboBox[] **ComboAwards** = {addNum(**new** JComboBox<>()), addNum(**new** JComboBox<>()), addNum(**new** JComboBox<>())};  
 **private** JTable **display** = **new** JTable(0, 3);  
 **private** JPanel[] **LabelComboAward** = {**new** JPanel(), **new** JPanel(), **new** JPanel()};  
 **private** JTextField **filedProject**;  
 ArrayList<People> **people** = **new** ArrayList<>();  
  
 **private** JComboBox addNum(JComboBox<Integer> jComboBox) {  
 **for** (**int** x = 1; x <= 255; x++) {  
 jComboBox.addItem(x);  
 }  
 **return** jComboBox;  
 }  
  
 **private void** readFile() **throws** FileNotFoundException {  
 **people**.clear();  
 String[] readline;  
 File file = **new** File(**"database.dat"**);  
 Scanner scanner = **new** Scanner(file);  
 **while** (scanner.hasNextLine()) {  
 readline = scanner.nextLine().split(**","**);  
 **people**.add(**new** People(readline[0], readline[1]));  
 }  
 scanner.close();  
 }  
  
 **private void** saveLog() **throws** IOException {  
 SimpleDateFormat formatter = **new** SimpleDateFormat(**"yyyy-MM-dd-HH-mm-ss"**);  
 Date date = **new** Date(System.*currentTimeMillis*());  
 String time = formatter.format(date);  
 File newFile = **new** File(time + **".dat"**);  
 File listFile = **new** File(**"filelist.dat"**);  
 **if** (!listFile.exists()) {  
 **if** (!listFile.createNewFile()) {  
 **throw new** IOException();  
 }  
 }  
 FileWriter fileWriter = **new** FileWriter(listFile.getName(), **true**);  
 fileWriter.write(time + **'\n'**);  
 fileWriter.close();  
 **if** (!newFile.exists()) {  
 **if** (!newFile.createNewFile()) {  
 **throw new** IOException();  
 }  
 }  
 fileWriter = **new** FileWriter(newFile.getName());  
 fileWriter.write(**filedProject**.getText() + **'\n'**);  
 **for** (People tmp : **people**) {  
 **if** (tmp.getAward() != 0) {  
 fileWriter.write(tmp.getName() + **','** + tmp.getTel() + **','** + tmp.getAward() + **'\n'**);  
 }  
 }  
 fileWriter.close();  
 }  
  
 **private int** getPrize(**int** n) {  
 **if** (**Box\_AwardNum**[n].isSelected()) {  
 **return** (**int**) **ComboAwards**[n].getSelectedItem();  
 } **else** {  
 **return** 0;  
 }  
 }  
  
 **private** String awardToString(**int** n) {  
 **switch** (n) {  
 **case** 1:  
 **return "一等奖"**;  
 **case** 2:  
 **return "二等奖"**;  
 **case** 3:  
 **return "三等奖"**;  
 **default**:  
 **return "未获奖"**;  
 }  
 }  
  
 **private** String telEncode(String tel) {  
 **if** (tel.length() < 2) {  
 **return "\*\*"**;  
 }  
 **return** tel.substring(0, tel.length() - 2) + **"\*\*"**;  
 }  
  
 **private void** lottery() {  
 ((DefaultTableModel) **display**.getModel()).getDataVector().clear();  
 Pool pool = **new** Pool(getPrize(0), getPrize(1), getPrize(2), **people**.size());  
 **for** (People tmp : **people**) {  
 tmp.setAward(pool.getPrize());  
 }  
 **for** (People tmp : **people**) {  
 **if** (tmp.getAward() == 1) {  
 DefaultTableModel model = (DefaultTableModel) **display**.getModel();*//获取defaulttablemodel* model.addRow(**new** String[]{tmp.getName(), telEncode(tmp.getTel()), awardToString(tmp.getAward())});  
 }  
 }  
 **for** (People tmp : **people**) {  
 **if** (tmp.getAward() == 2) {  
 DefaultTableModel model = (DefaultTableModel) **display**.getModel();*//获取defaulttablemodel* model.addRow(**new** String[]{tmp.getName(), telEncode(tmp.getTel()), awardToString(tmp.getAward())});  
 }  
 }  
 **for** (People tmp : **people**) {  
 **if** (tmp.getAward() == 3) {  
 DefaultTableModel model = (DefaultTableModel) **display**.getModel();*//获取defaulttablemodel* model.addRow(**new** String[]{tmp.getName(), telEncode(tmp.getTel()), awardToString(tmp.getAward())});  
 }  
 }  
 **try** {  
 saveLog();  
 } **catch** (IOException ignored) {  
 }  
 }  
  
 FrameMain(**int** x, **int** y) {  
 setTitle(**"幸运观众手机号码抽取器"**);  
 Container MainFrame = getContentPane();  
 setBounds(x, y, 800, 400);  
 JPanel settingArea = **new** JPanel();  
 settingArea.setLayout(**new** BoxLayout(settingArea, BoxLayout.***Y\_AXIS***));  
 JScrollPane displayArea = **new** JScrollPane(**display**);  
 MainFrame.add(displayArea);  
 **display**.setBorder(BorderFactory.*createEtchedBorder*());  
 **display**.getColumnModel().getColumn(0).setHeaderValue(**"姓名"**);  
 **display**.getColumnModel().getColumn(1).setHeaderValue(**"电话"**);  
 **display**.getColumnModel().getColumn(2).setHeaderValue(**"奖项"**);  
 **display**.setEnabled(**false**);  
 **display**.setRowHeight(30);  
 **display**.setCellSelectionEnabled(**false**);  
 JPanel award = **new** JPanel();  
 award.setLayout(**new** BoxLayout(award, BoxLayout.***X\_AXIS***));  
 JSplitPane splitPane = **new** JSplitPane(JSplitPane.***HORIZONTAL\_SPLIT***, settingArea, displayArea);  
 JLabel labelAward = **new** JLabel(**"请选择要产生的奖项"**);  
 **ButtonReadFile** = **new** JButton(**"读取文件"**);  
 **ButtonSetting** = **new** JButton(**"信息录入"**);  
 **ButtonLottery** = **new** JButton(**"抽奖!"**);  
 **ButtonChooseHistory** = **new** JButton(**"历史记录"**);  
 **ButtonReadFile**.addActionListener(**this**);  
 **ButtonSetting**.addActionListener(**this**);  
 **ButtonLottery**.addActionListener(**this**);  
 **ButtonChooseHistory**.addActionListener(**this**);  
 JPanel topArea = **new** JPanel();  
 topArea.setLayout(**new** BoxLayout(topArea, BoxLayout.***X\_AXIS***));  
 topArea.add(**ButtonReadFile**);  
 topArea.add(**ButtonSetting**);  
 topArea.add(**ButtonChooseHistory**);  
 JPanel nextArea = **new** JPanel();  
 nextArea.setMaximumSize(**new** Dimension(240, 30));  
 nextArea.setLayout(**new** BoxLayout(nextArea, BoxLayout.***X\_AXIS***));  
 JLabel labelProject = **new** JLabel(**"主题："**);  
 **filedProject** = **new** JTextField(**"抽奖"**);  
 nextArea.add(labelProject);  
 nextArea.add(**filedProject**);  
 settingArea.add(topArea);  
 settingArea.add(nextArea);  
 settingArea.add(labelAward);  
 **for** (JCheckBox tmp : **Box\_AwardNum**) {  
 tmp.addActionListener(**this**);  
 award.add(tmp);  
 }  
 settingArea.add(award);  
 JPanel awards = **new** JPanel();  
 awards.setLayout(**new** BoxLayout(awards, BoxLayout.***Y\_AXIS***));  
 **for** (**int** i = 0; i < 3; i++) {  
 **LabelComboAward**[i].setLayout(**new** BoxLayout(**LabelComboAward**[i], BoxLayout.***X\_AXIS***));  
 **LabelComboAward**[i].setMaximumSize(**new** Dimension(240, 0));  
 **LabelComboAward**[i].add(**LabelAwards**[i]);  
 **LabelAwards**[i].setVisible(**false**);  
 **LabelComboAward**[i].add(**ComboAwards**[i]);  
 **ComboAwards**[i].setVisible(**false**);  
 awards.add(**LabelComboAward**[i]);  
 }  
 settingArea.add(awards);  
 settingArea.add(**ButtonLottery**);  
 MainFrame.add(splitPane);  
 setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);  
 setVisible(**true**);  
 splitPane.setDividerLocation(0.35);  
 **try** {  
 readFile();  
 JOptionPane.*showMessageDialog*(**null**, **"读取文件成功"**, **"提示"**, JOptionPane.***INFORMATION\_MESSAGE***);  
 } **catch** (FileNotFoundException ignored) {  
 JOptionPane.*showMessageDialog*(**null**, **"请点击 信息录入 添加抽奖者"**, **"提示"**, JOptionPane.***ERROR\_MESSAGE***);  
 }  
 }  
  
 @Override  
 **public void** actionPerformed(ActionEvent actionEvent) {  
 **if** (actionEvent.getSource() == **ButtonReadFile**) {  
 **try** {  
 readFile();  
 JOptionPane.*showMessageDialog*(**null**, **"读取文件成功"**, **"提示"**, JOptionPane.***INFORMATION\_MESSAGE***);  
 } **catch** (FileNotFoundException e) {  
 JOptionPane.*showMessageDialog*(**null**, **"读取文件错误"**, **"提示"**, JOptionPane.***ERROR\_MESSAGE***);  
 }  
 }  
 **if** (actionEvent.getSource() == **ButtonLottery**) {  
 lottery();  
 JOptionPane.*showMessageDialog*(**null**, **"抽奖成功"**, **"提示"**, JOptionPane.***INFORMATION\_MESSAGE***);  
 }  
 **if** (actionEvent.getSource() == **ButtonSetting**) {  
 **new** FrameAddPeople(**people**, **this**, getX(), getY());  
 }  
 **if** (actionEvent.getSource() == **ButtonChooseHistory**) {  
 **new** FrameChooseHistory(getX(), getY());  
 }  
 **if** (actionEvent.getSource() == **Box\_AwardNum**[0] || actionEvent.getSource() == **Box\_AwardNum**[1] || actionEvent.getSource() == **Box\_AwardNum**[2]) {  
 **for** (**int** x = 0; x < 3; x++) {  
 **LabelAwards**[x].setVisible(**Box\_AwardNum**[x].isSelected());  
 **LabelComboAward**[x].setMaximumSize(**new** Dimension(240, **Box\_AwardNum**[x].isSelected() ? 30 : 0));  
 **ComboAwards**[x].setVisible(**Box\_AwardNum**[x].isSelected());  
 }  
 }  
 }  
}

* 1. People

**class** People {  
 **private** String **name**;  
 **private** String **tel**;  
 **private int award**;  
  
 People(String name, String tel) {  
 **this**.**name** = name;  
 **this**.**tel** = tel;  
 **this**.**award** = 0;  
 }  
  
 People(String name, String tel, **int** award) {  
 **this**.**name** = name;  
 **this**.**tel** = tel;  
 **this**.**award** = award;  
 }  
  
 String getName() {  
 **return name**;  
 }  
  
 String getTel() {  
 **return tel**;  
 }  
  
 **void** setAward(**int** award) {  
 **this**.**award** = award;  
 }  
  
 **int** getAward() {  
 **return award**;  
 }  
  
}

* 1. FrameAddPeople.java

**import** javax.swing.\*;  
**import** javax.swing.event.\*;  
**import** javax.swing.table.DefaultTableModel;  
**import** java.io.File;  
**import** java.io.FileWriter;  
**import** java.io.IOException;  
**import** java.util.ArrayList;  
  
**public class** FrameAddPeople **implements** TableModelListener {  
 **private** ArrayList<People> **people**;  
 **private** FrameMain **frameMain**;  
 **private** JTable **display** = **new** JTable(0, 2);  
 **private** JFrame **frame** = **new** JFrame(**"信息录入"**);  
  
 **private void** saveFile() {  
 File file = **new** File(**"database.dat"**);  
 **if** (!file.exists()) {  
 **try** {  
 file.createNewFile();  
 } **catch** (IOException ignored) {  
 }  
 }  
 **try** {  
 FileWriter fileWriter = **new** FileWriter(file.getName());  
 **for** (People tmp : **people**) {  
 fileWriter.write(tmp.getName() + **','** + tmp.getTel() + **'\n'**);  
 }  
 fileWriter.close();  
 } **catch** (IOException ignored) {  
 }  
 }  
  
 **private void** setArrayList() {  
 **people**.clear();  
 **int** rows = **display**.getRowCount();  
 **for** (**int** i = 0; i < rows; i++) {  
 **if** (!**display**.getValueAt(i, 0).equals(**""**) && !**display**.getValueAt(i, 1).equals(**""**)) {  
 **people**.add(**new** People((String) **display**.getValueAt(i, 0), (String) **display**.getValueAt(i, 1)));  
 }  
 }  
 **frameMain**.**people** = **people**;  
 }  
  
 **private void** initFrame(**int** x, **int** y) {  
 **int** RowCount = **display**.getRowCount();  
 **for** (**int** i = 0; i < RowCount; i++) {  
 **display**.remove(1);  
 }  
 DefaultTableModel model = (DefaultTableModel) **display**.getModel();*//获取defaulttablemodel* **for** (People tmp : **people**) {  
 model.addRow(**new** String[]{tmp.getName(), tmp.getTel()});  
 }  
 model.addRow(**new** String[]{**""**, **""**});  
 **display**.getModel().addTableModelListener(**this**);  
 JScrollPane displayArea = **new** JScrollPane(**display**);  
 **display**.getColumnModel().getColumn(0).setHeaderValue(**"姓名"**);  
 **display**.getColumnModel().getColumn(1).setHeaderValue(**"电话"**);  
 **display**.setRowHeight(30);  
 **frame**.setBounds(x + 20, y + 20, 300, 400);  
 **frame**.add(displayArea);  
 **frame**.setVisible(**true**);  
 }  
  
 FrameAddPeople(ArrayList<People> people, FrameMain frameMain, **int** x, **int** y) {  
 **this**.**frameMain** = frameMain;  
 **this**.**people** = people;  
 initFrame(x, y);  
 }  
  
 @Override  
 **public void** tableChanged(TableModelEvent tableModelEvent) {  
 setArrayList();  
 saveFile();  
 **if** (!(**display**.getValueAt(**display**.getRowCount() - 1, 0).equals(**""**) && **display**.getValueAt(**display**.getRowCount() - 1, 1).equals(**""**))) {  
 ((DefaultTableModel) **display**.getModel()).addRow(**new** String[]{**""**, **""**});  
 }  
 System.***out***.println(**"change"**);  
 }  
}

* 1. Pool.java

**class** Pool {  
 **private int prizeOne**, **prizeTwo**, **prizeThree**, **unPrized**;  
  
 Pool(**int** prizeOne, **int** prizeTwo, **int** prizeThree, **int** total) {  
 **this**.**prizeOne** = prizeOne;  
 **this**.**prizeTwo** = prizeTwo;  
 **this**.**prizeThree** = prizeThree;  
 **this**.**unPrized** = total - prizeOne - prizeTwo - prizeThree;  
 }  
  
 **int** getPrize() {  
 **int** num = **prizeOne** + **prizeTwo** + **prizeThree** + **unPrized**;  
 **int** tmp = (**int**) (Math.*random*() \* num + 1);  
 **if** (tmp <= **prizeOne**) {  
 **prizeOne**--;  
 **return** 1;  
 } **else if** (tmp <= **prizeOne** + **prizeTwo**) {  
 **prizeTwo**--;  
 **return** 2;  
 } **else if** (tmp <= **prizeOne** + **prizeTwo** + **prizeThree**) {  
 **prizeThree**--;  
 **return** 3;  
 } **else** {  
 **unPrized**--;  
 **return** 0;  
 }  
 }  
}

* 1. FrameChooseHistory.java

**import** javax.swing.\*;  
**import** java.awt.\*;  
**import** java.awt.event.ActionEvent;  
**import** java.awt.event.ActionListener;  
**import** java.io.File;  
**import** java.io.FileNotFoundException;  
**import** java.util.Scanner;  
**class** FrameChooseHistory **implements** ActionListener {  
 **private** JFrame **frame**;  
 **private** JComboBox<String> **historyList** = **new** JComboBox<>();  
  
 FrameChooseHistory(**int** x, **int** y) {  
 **frame** = **new** JFrame(**"选择你要查看的记录"**);  
 **frame**.setBounds(x + 20, y + 20, 400, 100);  
 **frame**.setVisible(**true**);  
 JLabel label = **new** JLabel(**"请选择你要查看的记录："**);  
 JButton button = **new** JButton(**"确定"**);  
 button.addActionListener(**this**);  
 **try** {  
 File file = **new** File(**"filelist.dat"**);  
 Scanner scanner = **new** Scanner(file);  
 **while** (scanner.hasNextLine()) {  
 String name = scanner.nextLine();  
 **historyList**.addItem(name);  
 }  
 scanner.close();  
 } **catch** (FileNotFoundException ignored) {  
 }  
 **frame**.add(**"North"**, label);  
 **frame**.add(**"Center"**, **historyList**);  
 **frame**.add(**"South"**, button);  
 }  
  
 @Override  
 **public void** actionPerformed(ActionEvent actionEvent) {  
 **new** FrameViewHistory((String) **historyList**.getSelectedItem(), **frame**.getX(), **frame**.getY());  
 }  
}

* 1. FrameViewHistory.java

**import** javax.swing.\*;  
**import** javax.swing.table.DefaultTableModel;  
**import** java.io.File;  
**import** java.io.FileNotFoundException;  
**import** java.util.ArrayList;  
**import** java.util.Scanner;  
  
**class** FrameViewHistory {  
 **private** String **history**;  
 **private** String **subject**;  
 **private** ArrayList<People> **people** = **new** ArrayList<>();  
  
 **private int** awardToInt(String award) {  
 **switch** (award) {  
 **case "1"**:  
 **return** 1;  
 **case "2"**:  
 **return** 2;  
 **case "3"**:  
 **return** 3;  
 **default**:  
 **return** 0;  
 }  
 }  
  
  
 **private void** readFile() {  
 **try** {  
 File file = **new** File(**history** + **".dat"**);  
 Scanner scanner = **new** Scanner(file);  
 **subject** = scanner.nextLine();  
 **while** (scanner.hasNextLine()) {  
 String[] readline = scanner.nextLine().split(**","**);  
 **people**.add(**new** People(readline[0], readline[1], awardToInt(readline[2])));  
 }  
 } **catch** (FileNotFoundException ignored) {  
 }  
 }  
  
 FrameViewHistory(String history, **int** x, **int** y) {  
 **this**.**history** = history;  
 readFile();  
 JFrame frame = **new** JFrame(history);  
 frame.setBounds(x + 20, y + 20, 500, 300);  
 frame.setVisible(**true**);  
 JLabel labelSubject = **new** JLabel(**"主题:"** + **subject**);  
 JTable display = **new** JTable(0, 3);  
 display.setEnabled(**false**);  
 JScrollPane displayArea = **new** JScrollPane(display);  
 display.setBorder(BorderFactory.*createEtchedBorder*());  
 display.getColumnModel().getColumn(0).setHeaderValue(**"姓名"**);  
 display.getColumnModel().getColumn(1).setHeaderValue(**"电话"**);  
 display.getColumnModel().getColumn(2).setHeaderValue(**"奖项"**);  
 DefaultTableModel model = (DefaultTableModel) display.getModel();  
 **for** (People tmp : **people**) {  
 model.addRow(**new** String[]{tmp.getName(), tmp.getTel(), awardToString(tmp.getAward())});  
 }  
 frame.add(**"North"**, labelSubject);  
 frame.add(displayArea);  
 }  
  
 **private** String awardToString(**int** award) {  
 **switch** (award) {  
 **case** 1:  
 **return "一等奖"**;  
 **case** 2:  
 **return "二等奖"**;  
 **case** 3:  
 **return "三等奖"**;  
 **default**:  
 **return "未获奖"**;  
 }  
 }  
}