****

**Java期末设计报告**

**姓名：曾朴**

**班级：计算机与信息工程学院2019级软件1班**

**学号：1930101**

**目录**

1. 程序的设计目标应实现的功能----------------------2

1.设计目的------------------------------------------3

2.应实现的功能--------------------------------------3

1. 每个类的作用和它们的联系------------------------3

1.各类的作用----------------------------------------3

2.各类的联系----------------------------------------3

1. 核心类的设计思路和关键代码----------------------

1.核心类设计思路------------------------------------5

2.关键代码------------------------------------------5

1. 测试数据及测试输出结果--------------------------
2. 项目设计的收获----------------------------------5

1. 涉及到的知识点-----------------------------------5

2. 遇到的问题以及产生的思考-------------------------5

3. 程序调试能力的思考-------------------------------5

**一．程序的设计目标应实现的功能**

**1.设计目的**

本《个人通讯录管理程序》具有对联系人信息进行管理的功能。能够对

联系人的姓名，职业，手机号码，电子邮箱，进行保存，还可以对联系人进行分组管理，简单方便，使用容易。

**2.应实现的功能**

本《个人通讯录管理系统》的主页面有多级菜单，能显示出组操作，联系人操作，子菜单分别能显示对应的功能。

在通讯录管理系统中，可以对组进行添加，删除，按组名查询和修改等操作，联系人也可以进行增加，删除，修改，和查询操作，查询可以选择按姓名，性别，手机号，电子邮箱进行查询。

二．每个类的作用和它们的联系

**1.各类的作用**

本《通讯录管理系统》一共有6个类，分别是PersonalInformation(个人信息类)，GroupInformation（群组信息类），AddressAdmin(通讯录管理类)，GroupAdmin（群组管理类），Main（主程序），Menu（菜单类）。

个人信息类主要是存放联系人各类属性的类，包含姓名，性别，职业，手机号码，电子邮箱等属性。分别有Getter和Setter方法。方便赋值和取值，也带有一个有参的构造器，方便对实例化对象进行初始化。

群组信息类包含了群组名称属性和一个ArrayList对象，分别带有Getter和Setter方法。方便赋值和取值，也带有一个有参的构造器，方便对对象进行初始化，这样设计实例化的对象群组就带有群组名字属性。

通讯录管理类，包含对联系人的增加，删除，修改等方法。其中，查询联系人可以按联系人各类属性进行查询。

群组管理类，包含对群组的增加，查找，删除，修改等方法，话可以输出所有组。

菜单类，主要存放各类菜单，方便其他类调用。

Main类，主要存放一个大集合ArrayList，和一些人机交互的信息，实现用户操作命令式菜单，调用相应的后台方法。

**2.各类的联系**

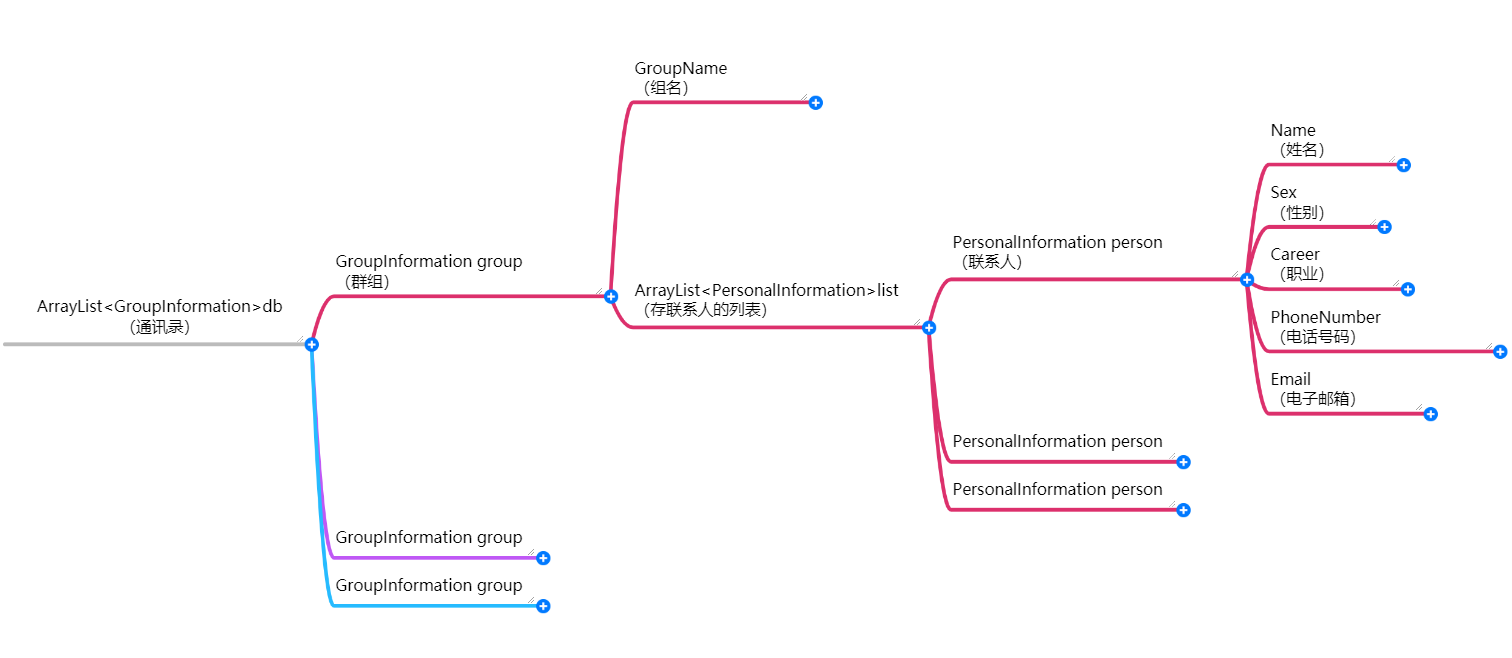
两个信息类（个人信息类）（群组信息类），用来产生联系人对象和群组对象。

通讯录管理类主要是对联系人对象管理，联系人类在通讯录管理类实例化后可以通过添加联系人方法添加到底层ArrayList中，通讯录管理类的其他方法可根据用户要求定位到该联系人对象进行管理。

群组管理类主要对群组对象进行管理类，群组类（包含组名和群组内联系人列表）在群组管理类实例化后可以通过添加群组方法添加到主ArrayList中，群组管理类的其他方法可根据用户要求定位到该群组对象进行管理。

菜单类存放各类菜单，主要给main方法进行调用。

Main类包含一个主ArrayList对象db，它存放这本系统的所有信息，通讯录管理类和群组管理类在Main实例化后，Main方法读取一些用户输入的信息可以分别调用各类管理方法进行管理该系统。

 系统关系图如下：

**三．核心类的设计思路和关键代码**

1.核心类设计思路

通讯录管理类主要是对联系人对象管理，联系人类在通讯录管理类实例化后可以通过添加联系人方法添加到底层ArrayList中，通讯录管理类的其他方法可根据用户要求定位到该联系人对象进行管理。

群组管理类主要对群组对象进行管理类，群组类（包含组名和群组内联系人列表）在群组管理类实例化后可以通过添加群组方法添加到主ArrayList中，群组管理类的其他方法可根据用户要求定位到该群组对象进行管理。

2.关键代码

个人信息类

**package** addressbook;

**public** **class** PersonalInformation {

**private** String Name;

**private** String Sex;

**private** String Career;

**private** String PhoneNumber;

**private** String Email;

**public** PersonalInformation(String name, String sex, String career, String phoneNumber, String email) {

**super**();

Name = name;

Sex= sex;

Career = career;

PhoneNumber = phoneNumber;

Email = email;

}

**public** String getName() {

**return** Name;

}

**public** **void** setName(String name) {

Name = name;

}

**public** String getSex() {

**return** Sex;

}

**public** **void** setSex(String sex) {

**this**.Sex = sex;

}

**public** String getCareer() {

**return** Career;

}

**public** **void** setCareer(String career) {

Career = career;

}

**public** String getPhoneNumber() {

**return** PhoneNumber;

}

**public** **void** setPhoneNumber(String phoneNumber) {

PhoneNumber = phoneNumber;

}

**public** String getEmail() {

**return** Email;

}

**public** **void** setEmail(String email) {

Email = email;

}

@Override

**public** String toString() {

**return** "姓名："+Name+"\t"+"性别："+Sex+" "+"职业："+Career+"\t"+"手机号码："+PhoneNumber+"\t"+"电子邮箱："+Email+"\t";

}

}

群组信息类

**package** addressbook;

**import** java.util.ArrayList;

**public** **class** GroupInformation {

**private** String GroupName;

**private** ArrayList<PersonalInformation>list=**new** ArrayList<PersonalInformation>();

**public** GroupInformation( String groupName,ArrayList<PersonalInformation> list) {

**this**.GroupName = groupName;

**this**.list = list;

}

**public** String getGroupName() {

**return** GroupName;

}

**public** **void** setGroupName(String groupName) {

GroupName = groupName;

}

**public** ArrayList<PersonalInformation> getList() {

**return** list;

}

**public** **void** setList(ArrayList<PersonalInformation> list) {

**this**.list = list;

}

**public** **void** ListString() {

System.***out***.println("组名："+GroupName+"\n"+"成员");

**for** (PersonalInformation s : **this**.list){

System.***out***.println(s.toString());

};

}

}

群组管理类——添加群组方法

**public** **void** Addgroup(String groupname,ArrayList<GroupInformation> db) {//增加组

GroupInformation group=**new** GroupInformation(groupname, **new** ArrayList<PersonalInformation>());

db.add(group);

System.***out***.println("添加成功");

通讯录管理类——添加联系人方法

**public** **void** Add(String name,String sex,String career,String phonenumber,String email,String groupname,ArrayList<GroupInformation> db) {

**boolean** isFound=**false**;

**for** (**int** i = 0; i < db.size(); i++) {

**if** (db.get(i)!=**null**) {

GroupInformation group=db.get(i);

**if** (group.getGroupName().equals(groupname)) {

isFound=**true**;

db.get(i).getList().add(**new** PersonalInformation(name, sex,career, phonenumber, email));

System.***out***.println("添加成功");

**break**;

}

}

}

**if** (!isFound) {

System.***out***.println("查无此组");

}

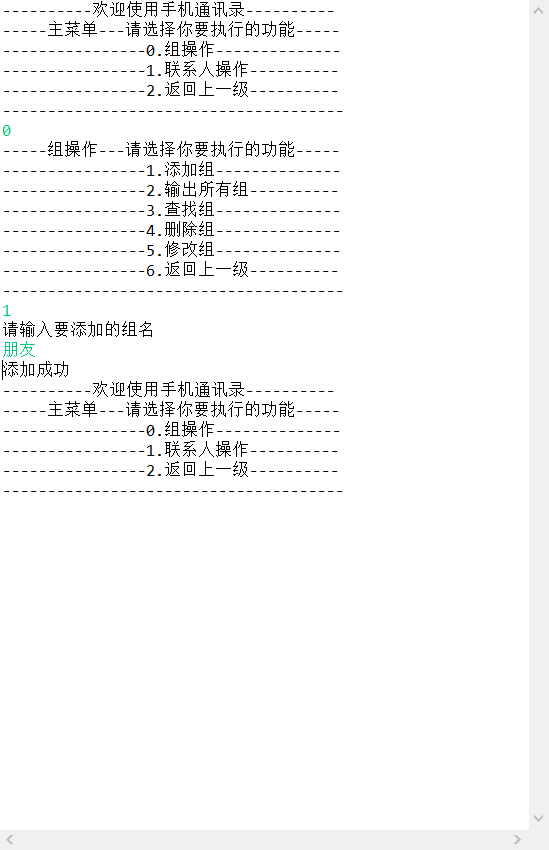
**return**;

}

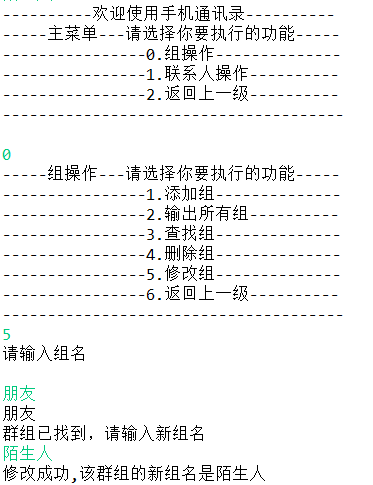
**四.测试数据及测试输出结果**

（1）添加一个群组，朋友，对其进行修改查找删除操作。

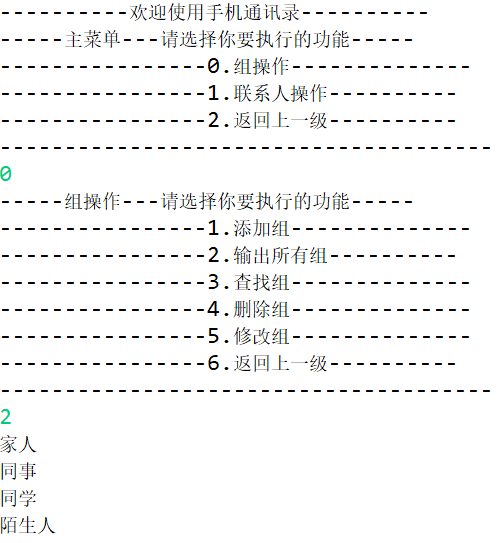
添加群组



修改群组



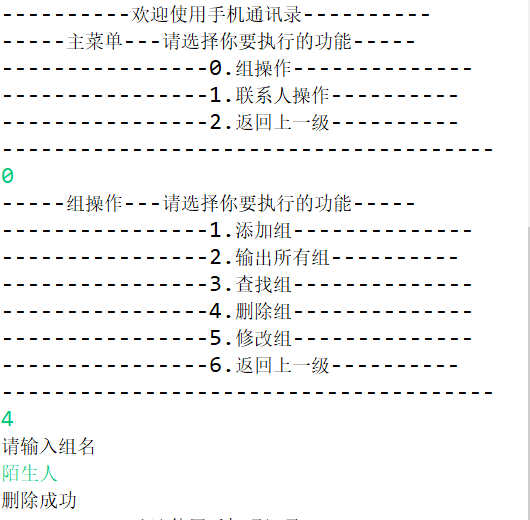
输出所有组



查找组



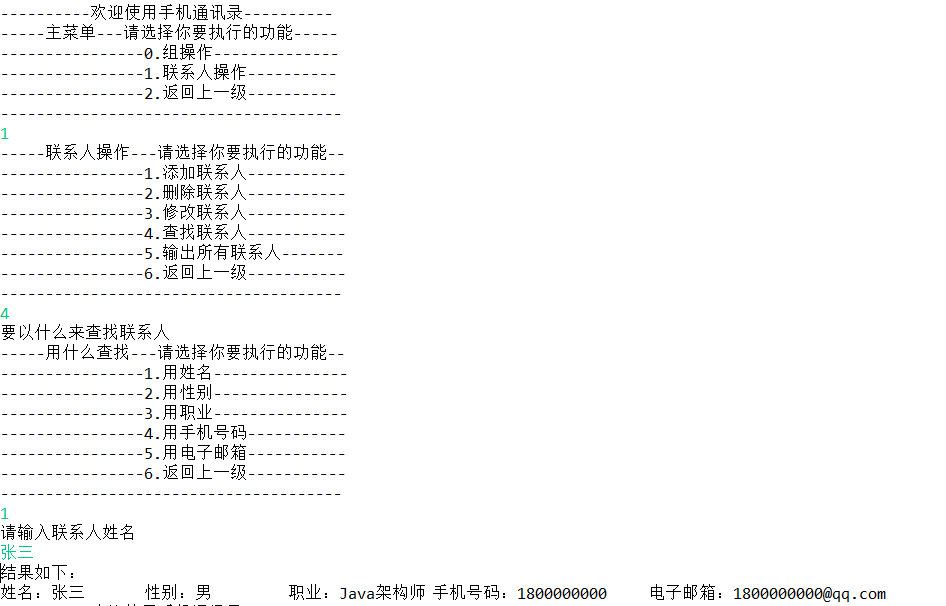
删除组



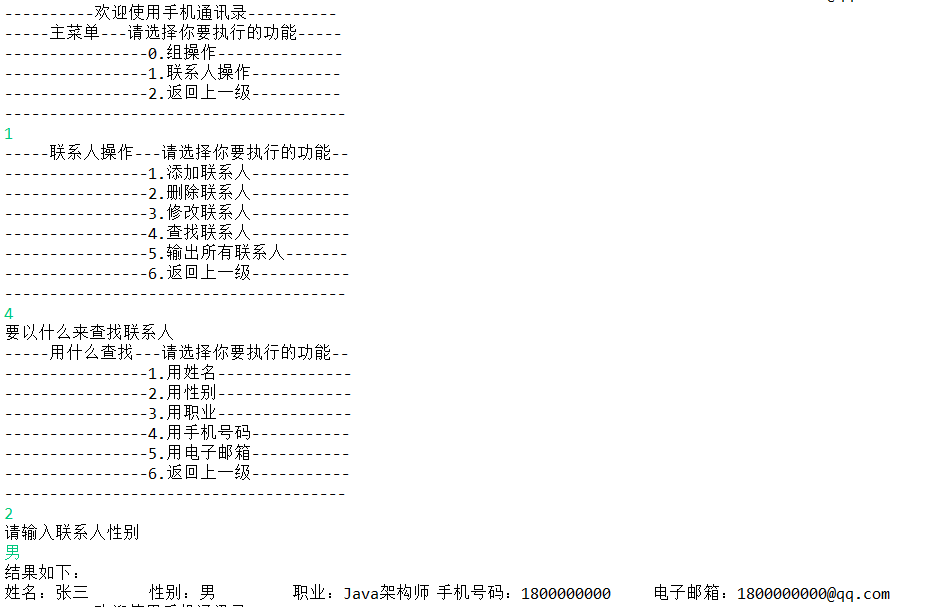
（2）添加一个联系人到新添加的群组朋友中，张三，对其进行查找修改删除操作。

查找联系人

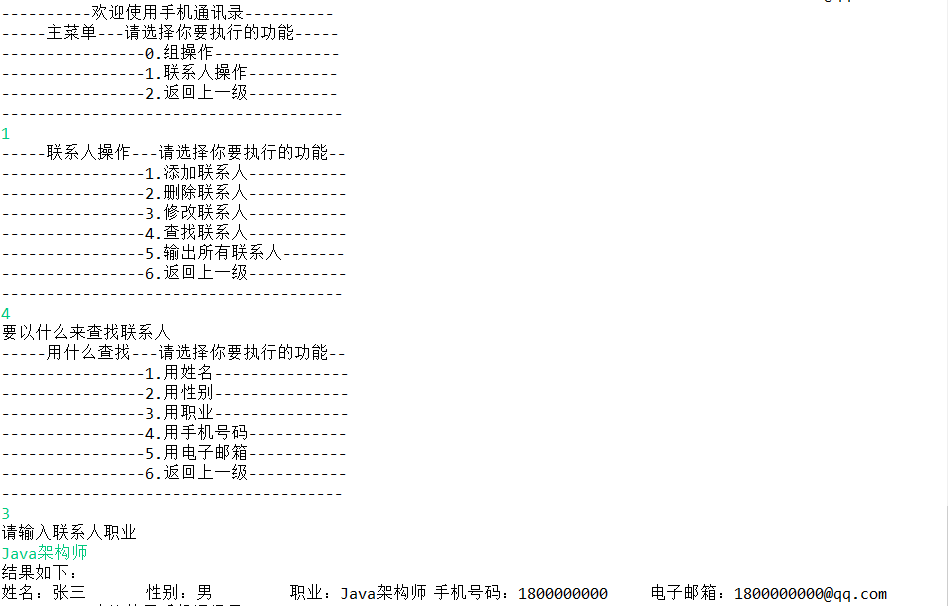
**用姓名**



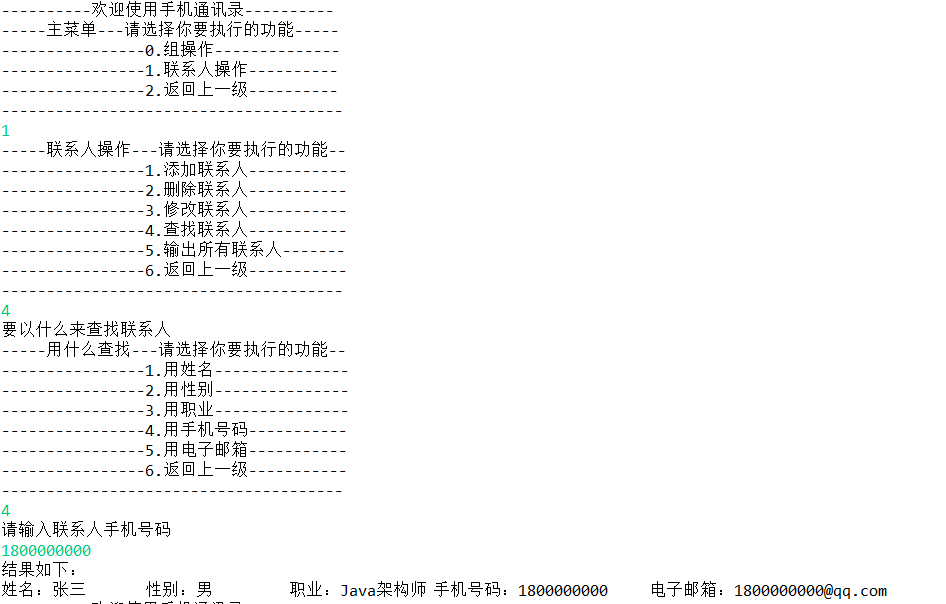
**用性别**



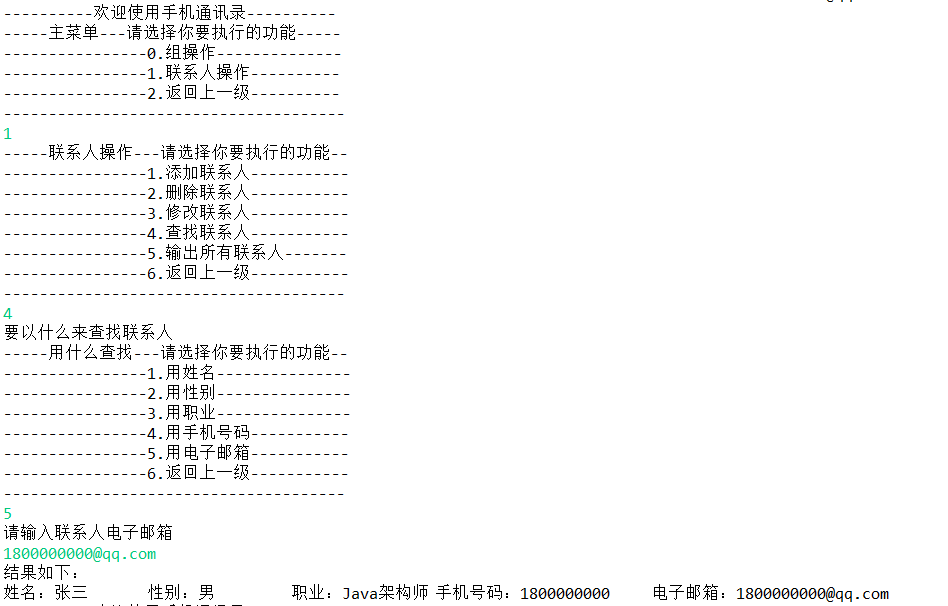
**用职业**



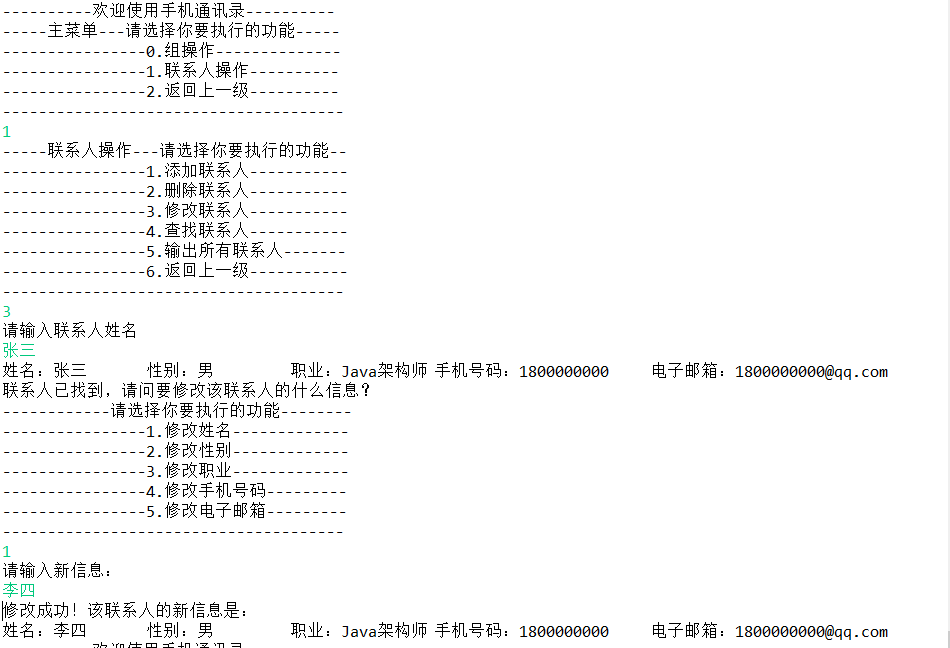
**用手机号码**



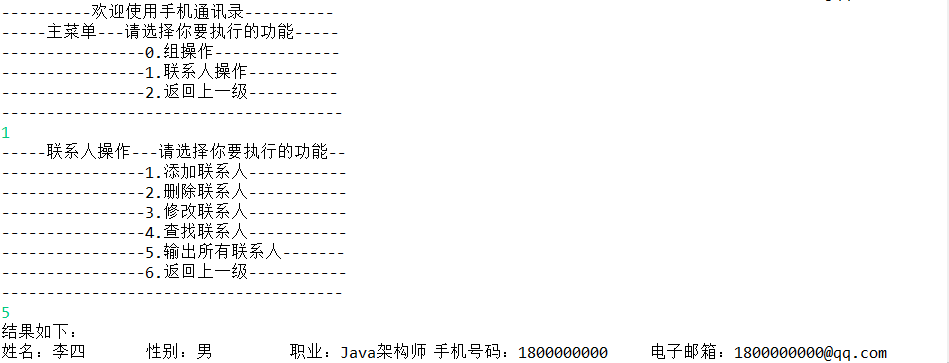
**用电子邮箱**



修改联系人



输出所有人



删除联系人



**五.项目设计的收获**

1. 涉及到的知识点

主要知识点是用到ArrayList集合，集合内部装的是一组person对象。

ArrayList是List接口的一个实现类在ArrayList内部封装了一个长度可变的数组对象，当存入的元素超过数组长度是，ArrayList会在内存中分配一个更大的数组来存储这些元素。

2. 遇到的问题以及产生的思考

通讯录管理系统开始编写时第一次接触ArrayList，也考虑过运用普通数组来编写。但课堂学习的数组通常是装一些基本数据类型的值，但每个联系人的信息是包括各种属性的，按这样的思路无法将这些信息有序的装入数组。后期，开始接触ArrayList可变数组对象，想到可以分别一个学生信息类，群组信息类，并将它们创建对象，有层次地装进ArrayList内。

3. 程序调试能力的思考

一般采用三个步骤调试程序。

先将测试各种功能能否正常使用，如果程序中输出异常字符。先观察各类的基本结构有无基本成分缺失，或者将自行模拟运行程序，思考方法逻辑是否正确，若问题无法解决，设置断点，用eclipse的debug功能查看程序运行路线。