

单元 10.2 个性化通讯录（二）

单元教学设计

学习阶段		二、实用程序开发训练		学时	
项目 10		个性化通讯录		学时	12
单元 10.2		个性化通讯录（一）		学时	2
教学目标	能力目标	知识目标		素质目标	
	能够建立 JDBC 数据库连接，能够将数据存入数据库，能够将数据从数据库中读取出来	1. 理解集合集合的概念 2. 掌握集合基本操作	课前准备	严谨务实的工作态度； 良好的心理素质； 用于克服困难的能力； 创新与追求完美的精神。	
		3. 灵活掌握对数据库的读写操作 4. 掌握桌面软件的开发流程 5. 学会使用鼠标监听器	课上学习练习		
课前准备	视频	预习：1. 《Java 中的集合框架概述（09:50）》 2. 《Collection 接口 & List 接口简介（02:28）》 复习：《什么是 Java 中的封装（05:53）》			
	课前准备：. 做好数据库的准备：做好几个界面。				
	个性化通讯录-新建联系人 请输入联系人信息： 姓名： 单位： 标识： 手机： 邮箱： 固定电话： 工作组： 地址： 备注： 保存联系人 重新填写 个性化通讯录-显示联系人 通讯录 姓名 部门 手机 小组 个性化通讯录-显示联系人 显示联系人详细信息： 姓名： 显示姓名内容 单位： 显示单位内容 标识： 显示标识内容 手机： 显示手机内容 邮箱： 显示邮箱内容 固定电话： 显示固定电话内容 工作组： 显示工作组内容 地址： 显示地址内容 备注： 显示备注内容				
项目引入	演示本节课项目目标，本节课要实现如下功能： （1）新建联系人 （2）显示联系人列表功能。 （3）显示联系人详细信息				
项目进度安排	任务	功能要求	训练技能点	完成方式	
	1	与数据库建立连接，读取数据库的数据，并将数据在控制台输出。	1. JDBC 连接数据库 2. 数据库读写	教师：演示讲解 学生：编程实现	
	2	用图形界面实现如下功能： （1）新建联系人 （2）显示联系人列表功能。 （3）显示联系人详细信息	1. 封装 2. 集合操作 3. 基本组件：单选按钮、多选按钮、下拉列表	教师：演示讲解 学生：编程实现	
	3	用图形界面实现如下功能：	1. JFileChooser	教师：演示讲解	

		(1) 修改联系人 (2) 检索联系人	2. 窗口间信息的传递 3. CardLayout	学生：编程实现
十分测试	检查集合操作： 用输入框输入任意多个字符串，以#结束，然后将所有的字符串输出。			
能力训练任务	任务一	设计 Friend 类，封装联系人信息。		
	任务二	新建联系人 课程思政 知识点 SQL 注入问题及解决举例 思政点 用户登录时，验证用户名和密码，SQL 语句为： select * from user where name= "?" and password=?; 使用 Statement 接口执行该 SQL 语句，用户输入：ABC(4242431 or 1=1)，就会形成： select * from user where name= "ABC" and password=4242431 or 1=1; 上述 SQL 语句条件恒成立，为不法分子非法获取数据库信息提供了可乘之机，开发的系统安全性不够。使用 PreparedStatement 来执行 SQL 语句可以解决如上问题。另外，还可以将代码简化，可读性好，避免出错。 思政融入 对学生进行遵守宪法法律法规和安全教育。自己编写的代码安全性和上网安全性。引导学生关注网络安全、合法获取信息，免受不法分子伤害，谨防套路贷、电信诈骗等。 带领学生学习《刑法》第 285-288 条款的内容，以及违反之后被处罚的案例。教育学生遵纪守法，不做对社会有害事情。 关注网络安排，免受不法分子伤害。不要网络上借款；不要点击无关的非法链接；远离传销！		
	任务三	显示所有联系人		
	任务四	显示某一个联系人		
	课后作业	完善程序		
课后学习资源		在线学习平台		
课程思政		思政资料： 1.大数据到底和我们的生活联系有多紧密？听听科学院院士怎么说 2.生活大数据：超一半网民，有各种睡眠问题，睡眠平均时间 7 小时生活中的大数据——居住篇 3.生活大数据：中国发展人工智能，拥有三大优点，首先就是大数据生活大数据： 4.人工智能时代到来，以下行业更吃香，有你喜欢的吗 5.生活大数据：模拟人类思维过程，产生了人工智能，逐步渗透生活中		
课后		思政论坛： 1.生活中的大数据：大数据在我们生活中的应用场景，和对我们生活的影响。 2.发生在我们身边的大数据都是怎样采集的？		

单元教学进度设计

Step1: 项目导入 (20) 分钟

课前检测【10 分钟】

教学内容	教师活动	学生活动
用输入框输入任意多个字符串，以#结束，然后将所有的字符串输出。 <pre>public class TestArrayList { public void test(){ List<String> strs=new ArrayList<String> (); String s=""; while(!s.equals("#")){ s=JOptionPane.showInputDialog("请输入要保存的字符串，以#结 束: "); strs.add(s); } for(String ss:strs){ System.out.println(ss); } } public static void main(String[] args){ TestArrayList t=new TestArrayList(); t.test(); } }</pre>	辅导	编程
对集合的输出有如下几种常用方式： 方式一： <pre>for(int i=0;i<strs.size();i++){ String ss=strs.get(i); System.out.println(ss); }</pre> 方式二： <pre>for(String ss:strs){ System.out.println(ss); }</pre> 方式三： <pre>Iterator it=strs.iterator(); while(it.hasNext()){</pre>	讲解总结	听讲记录

<pre>String ss=(String)it.next(); System.out.println(ss); }</pre>		
1. 定义集合的时候，用到泛型，指明这个集合中，只允许放入 String 字符串。 2. 如果使用泛型，则取出的时候不需要进行类型转换。 3. 如果不适用泛型，则取出时候须要进行类型转换。	检查	互查

课题导入

新课导入	演示本节课的项目目标，要做的如下几个功能： 1.新建联系人信息 2.显示所有联系人信息	提出问题	讨论思考	
------	---	------	------	--

Step2: 任务实施

任务 1: (25 分钟)

教 学 环节	教学内容					教师活动	学生活动	
任务引入	请同学们根据 Friend 表，设计 Friend 类，封装联系人信息。					辅导	编程	
	表 Friend							
	编号	字段名	字段类型	约束	备注			
	1.	id	int	primary key auto_increment	主键			
	2.	name	varchar(20)	not null	姓名			
	3.	dept	varchar(50)		单位			
	4.	mark	varchar(20)		标识			
	5.	phone	varchar(20)		手机			
	6.	email	varchar(30)		邮箱			
	7.	tel	varchar(20)		固定电话			
	8.	workGroup	varchar(20)		工作组			
	9.	address	varchar(50)		地址			
10.	note	text		备注				
任务实施	<pre>public class Friend { private int id; private String name; private String dept; private String mark; private String phone; private String email; private String tel; private String workGroup; private String address; private String note; public int getId() { return id; } public void setId(int id) { this.id = id; } }</pre>				我们一般情况下，叫这种类叫实体类，与所对应数据库表项对应： 1.表明=类名 2.属性名=字段名 3.属性类型对应字段类型 （1）Varchar→String （2）int→int （3）text→String		辅导	编程

任务 2: (45 分钟)

任务引入	结合课前做好的 NewFriend 窗口，新建联络人。	引入	思考
------	-----------------------------	----	----

任务部署	1.给保存按钮添加事件监听器 2.点击按钮后，作如下事情： （1）将文本框的输入，封装到一个 Friend 类中。 （2）将对象插入到数据库中。 （3）操作成功后，显示操作成功的信息。	讲解总结	思考
	1.给保存按钮添加事件监听器，将输入框中所有的信息封装到一个 Friend 类中。 Friend f=new Friend(); f.setName(tfs[0].getText()); f.setDept(tfs[1].getText()); f.setMark(tfs[2].getText()); f.setPhone(tfs[3].getText()); f.setEmail(tfs[4].getText()); f.setTel(tfs[5].getText()); f.setWorkGroup(tfs[6].getText()); f.setAddress(tfs[7].getText()); f.setNote(tfs[8].getText());	辅导	编程
任务执行	2.写一个方法，将一个 Friend 对象插入到数据库中。 （1）明确表和类的关系，一条记录和一个对象的关系。 （2）几个类的作用： Friend 类：对 Friend 表中的数据进行封装 FriendManager 类，对 Friend 表进行操作。 **Frame 类，向用户展示要显示的内容或操作。 第一步： FriendManager 类中，有方法 InsertFriend，实现将数据插入到数据库中去。 <pre> public int insertFriend(Friend f){ int i=0; try{ Connection conn=this.getConn(); String sql="insert into friend " + "(name,dept,mark,phone,email,tel,workGroup,address,note) " + "values(?,?,?,?,?,?,?,?)"; PreparedStatement ps=conn.prepareStatement(sql); ps.setString(1, f.getName()); ps.setString(2, f.getDept()); ps.setString(3, f.getMark()); ps.setString(4, f.getPhone()); ps.setString(5, f.getEmail()); ps.setString(6, f.getTel()); ps.setString(7, f.getWorkGroup()); ps.setString(8, f.getAddress()); ps.setString(9, f.getNote()); i=ps.executeUpdate(); this.closeConn(conn); }catch(Exception ex){ ex.printStackTrace(); } return i; } </pre> sql语句用通配符来代替变量。 专门用来运载带? sql语句的特殊小车 为每个?赋值，类型要与应有的变量类型一致。 第二步： 调用 FriendManager 类中的 insertFriend(Friend f)方法，将封装好的 Friend 对象添加到数据库中，如果添加成功，则展示添加成功信息。	演示讲解	学习修改

	<pre> FriendManager fm=new FriendManager(); int i=fm.insertFriend(f); if(i>0){ JOptionPane.showMessageDialog(null, "联系人添加成功!"); } </pre> 		
课程思政	知识点 SQL 注入问题及解决举例 思政点 用户登录时，验证用户名和密码，SQL 语句为： select * from user where name= "?" and password=?; 使用 Statement 接口执行该 SQL 语句，用户输入：ABC(4242431 or 1=1)，就会形成： select * from user where name= "ABC" and password=4242431 or 1=1; 上述 SQL 语句条件恒成立，为不法分子非法获取数据库信息提供了可乘之机，开发的系统安全性不够。使用 PreparedStatement 来执行 SQL 语句可以解决如上问题。另外，还可以将代码简化，可读性好，避免出错。 思政融入 对学生进行遵守宪法法律法规和安全教育。自己编写的代码安全性和上网安全性。引导学生关注网络安全、合法获取信息，免受不法分子伤害，谨防套路贷、电信诈骗等。 带领学生学习《刑法》第 285-288 条款的内容，以及违反之后被处罚的案例。教育学生遵纪守法，不做对社会有害事情。 关注网络安排，免受不法分子伤害。不要网络上借款；不要点击无关的非法链接；远离传销！	主持引导	参与讨论

任务 3：（45 分钟）

教学环节	教学内容	教师活动	学生活动
任务实施	具体步骤： 1.在 FriendManager 类中，定义一个方法，用来查询数据库，取出所有信息，每一条记录都封装在一个 Friend 类的对象中，然后将所有的结果都放入一个集合中。	演示讲解	学习记录

```

public ArrayList getAllFriends(){
    ArrayList fs=new ArrayList();
    try{
        Connection conn=this.getConn();
        String sql="select * from friends";
        Statement stmt=conn.createStatement();
        ResultSet rs=stmt.executeQuery(sql);
        while(rs.next()){
            Friend f=new Friend();
            f.setAddress(rs.getString("address"));
            f.setDept(rs.getString("dept"));
            f.setEmail(rs.getString("email"));
            f.setId(rs.getInt("id"));
            f.setMark(rs.getString("mark"));
            f.setName(rs.getString("name"));
            f.setNote(rs.getString("note"));
            f.setPhone(rs.getString("phone"));
            f.setTel(rs.getString("tel"));
            f.setWorkGroup(rs.getString("workGroup"));
            fs.add(f);
        }
        this.closeConn(conn);
    }
}

```

1.建立连接
2.sql语句
3.来回运输的小车
4.查询结果放入结果集
5.依次封装后放入集合
6.断开连接

2.在 FriendListFrame 类中调用 FriendManager 类中的 getAllFriend () 方法，取出所有联系人信息集合，并显示所有联系人。

```

public void showFriends(){
    FriendManager fm=new FriendManager();
    fs=fm.getAllFriends();
    for(Friend f:fs){
        String[] f_array=new String[4];
        f_array[0]=f.getName();
        f_array[1]=f.getDept();
        f_array[2]=f.getPhone();
        f_array[3]=f.getWorkGroup();
        data.addRow(f_array);
    }
}

```

在JTable中添加行

个性化通讯录-显示联系人

通讯录			
姓名	部门	手机	小组
李庆	15网3	17862341659	1组
刘磊	15网3	17862342386	1组
陈丹妮	15网3	17806214223	2组
董银娟	15网3	17862792656	2组
金翰林	15网络3班	13565984468	二组

1.在构造方法中调用。可以一打开页面就显示所有联系人。

任务 4: (45 分钟)

教学环节	教学内容	教师活动	学生活动
任务导入	显示某联系人的详细信息，可以用来查看联系人信息。	演示讲解	学习记录
任务部署	1.在表格表格上添加鼠标监听器，用来监听鼠标事件。 2.鼠标点击某一行时，将该行的 Friend 对象传递给显示界面 FriendShowFrame 3.在显示界面 FriendShowFrame 中显示信息。		
任务实施	第一步: 在表格表格上添加鼠标监听器，用来监听鼠标事件。 (1) 定义鼠标监听器。鼠标监听器接口 <code>MouseListener</code> 有 5 个方法要实现，我们须要用到的是 <code>clicked</code> 方法。 <pre> public class MyMouseListener implements MouseListener{ @Override public void mouseClicked(MouseEvent e) { int row=table.getSelectedRow(); Friend f_show=fs.get(row); new FriendShowFrame(f_show); } @Override public void mouseEntered(MouseEvent e) { } @Override public void mouseExited(MouseEvent e) { } @Override public void mousePressed(MouseEvent e) { } @Override public void mouseReleased(MouseEvent e) { } } </pre> 双击后执行 鼠标经过执行 鼠标离开执行 鼠标按下执行 鼠标放开执行 (2) 将鼠标监听器注册到表格上。 <pre> table.addMouseListener(new MyMouseListener()); </pre> 第二步: 鼠标点击某一行时,将该行的 Friend 对象传递给显示界面 FriendShowFrame <pre> int row=table.getSelectedRow(); Friend f_show=fs.get(row); </pre> 获得行号，也可以获得列号。 获得行号后，行号正好是改 Friend 对象在 ArrayList 中的编号。 第三步: 在显示界面 FriendShowFrame 中显示信息。	演示讲解	学习记录

	<pre>public void showFriend(){ ls2[0].setText(f.getName()); ls2[1].setText(f.getDept()); ls2[2].setText(f.getMark()); ls2[3].setText(f.getPhone()); ls2[4].setText(f.getEmail()); ls2[5].setText(f.getTel()); ls2[6].setText(f.getWorkGroup()); ls2[7].setText(f.getAddress()); ls2[8].setText(f.getNote()); }</pre>		
错误情境	Exception in thread "AWT-EventQueue-0" java.lang.ArrayIndexOutOfBoundsException: -1 at java.util.ArrayList.get(Unknown Source) at unit2.job3.FriendListFrame\$MyMouseListener.mouseClicked(FriendListFrame.java:70) at java.awt.AWTEventMulticaster.mouseClicked(Unknown Source) at java.awt.Component.processMouseEvent(Unknown Source) at javax.swing.JComponent.processMouseEvent(Unknown Source) at java.awt.Component.processEvent(Unknown Source) at java.awt.Container.processEvent(Unknown Source) at java.awt.Component.dispatchEventImpl(Unknown Source) at java.awt.Container.dispatchEventImpl(Unknown Source) <code>table.setEnabled(false);</code> 错误原因：将 JTable 设置为 表格已经是不可编辑了，点击行就毫无意义了。		

Step3: 总结与课后安排

教学环节	教学内容	教师活动	学生活动
教学小结	1.集合和集合操作 （1）什么是集合 （2）集合的基本操作 2.鼠标监听 （1）以内部类实现鼠标监听器 （2）鼠标监听器的注册和使用	总结	听讲记录
布置作业	课后任务 同步课外项目：图书管理系统。 本节复习：课堂录像、项目说明文档 作业要求：程序上传至云盘，放在 *组-姓名 文件夹下。		
课后学习资源	课程录像 常用 SQL 语句		
课程思政	思政资料： 1.大数据到底和我们的生活联系有多紧密？听听科学院院士怎么说 2.生活大数据：超一半网民，有各种睡眠问题，睡眠平均时间 7 小时生活中的大数据——居住篇 3.生活大数据：中国发展人工智能，拥有三大优点，首先就是大数据生活大数据： 4.人工智能时代到来，以下行业更吃香，有你喜欢的吗 5.生活大数据：模拟人类思维过程，产生了人工智能，逐步渗透生活中		
课后	思政论坛： 1.生活中的大数据：大数据在我们生活中的应用场景，和对我们生活的影响。 2.发生在我们身边的大数据都是怎样采集的？		