**计算机科学与技术学院《Java语言课程实训》过程记录**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 题目 | 易管理 客户资料管理系统 | | | | 序号 | 96 |
| 学号 | 18110506084 | 姓名 | 李威 | 班级 | 软件1803 | |
| **2019年 12 月 30 日 （第1天）**  **一、所涉及技术点及收获:**  1、安装eclipse，环境配置  配置eclipse.ini，打开eclipse安装目录，记事本打开eclipse.ini，修改最大最小运行内存：  -Xms512m  -Xmx4096m  2、windowbuilder  解压 windowbuilder 的 repository.zip ，只保留 features 和 plugins 两个文件夹，移动到eclipse安装目录下的dropins 文件夹中。  3、 lombok  将lombok下文件夹 安装包 下的 lombok.jar 移动到eclipse安装目录根目录，在eclipse.ini中插入这段语句  -javaagent:lombok.jar  -Xbootclasspath/a:lombok.jar   1. SQLyog\_1251   打开 key.txt 复制激活码8d8120df-a5c3-4989-8f47-5afc79c56e7c  打开SQLyog.exe，选择简体中文，随便输入名称，和复制的激活码，就可以使用了。  不要检查更新。  5、usbwebservercncn.exe(password-usbw)  直接运行数据库软件，记录数据库用户名，密码，端口号。我的是 root, usbw, 3307  保持后台运行  6、项目及数据库  先导入数据库，tcbd1011.sql，tcbd1011数据库就导入成功了。  打开SQLyog\_1251，新建连接，数据库tcbd1011，填入用户名，密码，端口号。  保持后台运行SQLyog。  7、PowerDesigner165  解压PowerDesigner165， 运行PowerDesigner165\_Evaluation.1428562995.exe，点next，语言选择PRC，同意条款，点next，选择安装位置，点next，取消勾选Eclipse 3 Plug-in 和 SQL Antwhere 12 Drivers，点next， 全部勾选，一路next直接安装，等待安装完成。  解压PowerDesigner165\_破解文件，把pdflm16.dll放在安装目录根目录，覆盖原文件，试用期不会过期。  **二、编程遇到的问题及解决方案**  1、修改eclipse.ini后eclipse无法运行  解决方案：配置信息出错，检查语法是否错误，修改后重新运行。  注意代码顺序  -Xbootclasspath/a:lombok.jar  -javaagent:lombok.jar  就会导致这中错误。   1. 没有usbwebservercncn.exe   解决方案：安装usbwebservercncn.exe，或者安装同类可运行MySQL的集成软件或直接运行MySQL服务。   1. tcbd1011.sql导入失败   解决方案：成功启动MySQL后，在MySQL管理界面或SQLyog操作界面进行基本配置后，Import导入文件tcbd1011.sql，导入路径选择tcbd1011.sql文件路径，成功后生成tcbd1011数据库及tcbd1011数据库下多个表。  **2019年 12 月 31 日 （第2天）**  **一、所涉及技术点及收获:**  1、导入edu项目  Package Explorer窗口右键选择 Import，导入项目，选择General 中的 Existing Projects into Workspace，点击next，选择Select root directoty ，选择 项目及数据库中edu文件夹，勾选Copy projects into workspace，点击Finish，就可以了。   1. 运行样例程序   Package Explorer窗口右上方找到下三角号按钮，点击选择Package Presentation，选择Hierarchical，垂直视图。  找到edu项目，打开src目录，打开c3p0-config.xml，选择查看窗口下方“Design”“Source”，选择Source，修改代码：  其中修改"jdbcUrl"，"user"，"password"，保存。  <?xml version="1.0" encoding="UTF-8"?> <c3p0-config> <named-config name="tcbd1011"> <property name="driverClass">com.mysql.jdbc.Driver</property> <property name="jdbcUrl">jdbc:mysql://localhost:3307/tcbd1011?useSSL=false&amp;characterEncoding=utf-8&amp;useUnicode=true</property> <property name="user">root</property> <property name="password">usbw</property> <property name="maxPoolSize">20</property> <property name="minPoolSize">3</property> <property name="maxStatements">30</property> <property name="maxIdleTime">150</property> </named-config> </c3p0-config>  依次选择src->user->swing->LoginFrame，双击打开，选择查看窗口下方“Design”“Source”，选择Source，找到main方法，右键Run As，1 Java Application，成功运行。  3、新建PowerDesigner  选择File -> New Model， 依次选择Model types，Physical Data Modal，Physical Diagram，然后DBMS选择MySQL 5.0，点击OK。  **二、编程遇到的问题及解决方案**  1、导入edu项目后大量报错不能运行  解决方案：导入方式不对。  Package Explorer窗口右键选择 Import，导入项目，选择General 中的 Existing Projects into Workspace，点击next，选择Select root directoty ，选择 项目及数据库中edu文件夹，勾选Copy projects into workspace，点击Finish，就可以了。  2、导入edu项目后不能登录，点击登录按钮没反应  解决方案：数据库连接问题。  找到edu项目，打开src目录，打开c3p0-config.xml，选择查看窗口下方“Design”“Source”，选择Source，修改代码：其中修改"jdbcUrl"，"user"，"password"，保存。  3、新建PowerDesigner无法进行数据库设计  解决方案：选择File -> New Model， 依次选择Model types，Physical Data Modal，Physical Diagram，然后DBMS选择MySQL 5.0，点击OK。  4、PowerDesigner设计项目无法导出  解决方案：先保存后命名再导出  **2020年 1 月 2 日 （第3天）**  **一、所涉及技术点及收获:**  1、个性化eclipse设置字符编码UTF-8  点击eclipse选项 Window -> Preferences ->General，找到Workspace，里面找到Text file encoding，选择Other: UTF-8。  2、调整字号  点击eclipse选项 Window -> Preferences ->General -> Appearance -> Colors and Fonts，里面找到Basic -> 最后面的Text Font ，选择Edit，即可调整。  3、调整主题  点击eclipse选项 Window -> Preferences ->General -> Appearance，里面找到Theme，即可调整。  可以选择炫酷的Dark，暗黑模式。  4、调整布局  鼠标滑倒eclipse右边，包含Java EE、Open Perspective等三个选项的组建上，拖拉到eclipse左边边框上。  点击eclipse选项 Window -> Appearance -> Hide Toolbar，隐藏工具栏。  5、快捷键：  • ctrl + 3 快速新建project/class/package  • ctrl + 1 快速解决错误，创建方法  • alt + / 快速创建main方法，添加提示解决错误  • ctrl + d 删除本行  • ctrl + shift + o 快速导入package包  • ctrl + t 看到某一个类的继承关系  6、创建项目的重点  报名规范：网址倒过来写https://www.sdut.edu.cn/  包名：cn.edu.sdut.每一张表（功能）  企业级开发：SSM框架缩写  • S: Spring  • S: SpringMVC  • M: JDBC > Dbutils > mybatis  cn.edu.sdut.comm（核心包）  1.和数据库相关部分  • c3p0-config.xml 连接数据库信息  • JDBCUtils 类  • BeanFactory 类  • CommonUtils 桥梁类  • JDBC封装  • BaseDaoImpl 泛型  List<E> 泛型，做数据检查的，数据类型必须是类的类型  java默认提供的4种最基本的类型  KTVE  • T: 代表java中 的任意类型 Object Integer Double  • E: 简单的理解就是数据库中的某一张表  • KV: key -value 一对一出现 MMap<K, V>  **二、编程遇到的问题及解决方案**  1、导入项目中大量乱码，执行程序后也大量乱码  解决方案：没有设置字符编码 。点击eclipse选项 Window -> Preferences ->General，找到Workspace，里面找到Text file encoding，选择Other: UTF-8。  2、使用ctrl+3快捷键，输入框太小。  解决方案：鼠标滑倒eclipse右边，包含Java EE、Open Perspective等三个选项的组建上，拖拉到eclipse左边边框上。  点击eclipse选项 Window -> Appearance -> Hide Toolbar，隐藏工具栏。  **2020年 1 月 3 日 （第4天）**  **一、所涉及技术点及收获:**  1、项目需要的jar包  1.数据库相关  mysql-connector-java-5.1.41.jar  c3p0-0.9.5.2.jar  mchange-commons-java-0.2.11.jar  2.封装JDBC的jar包  commons-dbutils-1.6.jar  3.优化Java封装特性的jar包  lombok-1.18.0.jar  2、导入需要的jar包  ctrl + 3 输入 new java project 创建新java项目，base  ctrl + 3 输入 new folder 创建新文件夹 lib，在工程base目录下  ctrl + c , ctrl + v 复制样例java项目lib文件夹下上面5个jar包到base->lib目录下，  选中5个jar包，右键，Build Path -> Add Build Path，编译即可。  3、创建核心包（package）结构 comm  参考样例项目包结构。  ctrl + 3 输入 new package创建新的java package: cn.edu.sdut  依次创建新的java package:  • cn.edu.sdut.comm  • cn.edu.sdut.comm.base.dao  • cn.edu.sdut.comm.datasource.util  • cn.edu.sdut.comm.properties  • cn.edu.sdut.comm.properties.factory  • cn.edu.sdut.comm.properties.util  然后复制c3p0-config.xml和applicationContext.properties到src目录下，清空applicationContext.properties文件内容。  4、将样例项目中的BaseDaoImpl.java，JDBCUtils.java，BeanFactory.java，PropertiesUtils.java，CommonUtils.java，复制到base工程相应包结构中。如有报错，先删除原有import语句，可以使用ctrl+d快捷键。  import cn.kgc.tangcco.comm.datasource.util.JDBCUtils;  再引入当前package包结构对应路径，可以使用ctrl+shift+o快捷键。  import cn.edu.sdut.comm.datasource.util.JDBCUtils;  **二、编程遇到的问题及解决方案**  1、导入需要的jar包出错  解决方案：选中5个jar包，右键，Build Path -> Add Build Path，编译即可。  2、Package Explorer 窗口怎么打开？  解决方案：点击eclipse选项 Window -> Show Views -> Others，搜索package，选择Java>Package Explorer，Open即可。   1. 将样例项目中的BaseDaoImpl.java，JDBCUtils.java，BeanFactory.java，PropertiesUtils.java，CommonUtils.java，复制到base工程相应包结构，有报错   解决方案：a，PropertiesUtils.java，CommonUtils.java，复制到base工程相应包结构中。如有报错，先删除原有import语句，可以使用ctrl+d快捷键。  import cn.kgc.tangcco.comm.datasource.util.JDBCUtils;  再引入当前package包结构对应路径，可以使用ctrl+shift+o快捷键。  import cn.edu.sdut.comm.datasource.util.JDBCUtils;  **2020年 1 月 4 日 （第 5 天）**  **一、所涉及技术点及收获:**  1、测试数据库连接  新建测试包cn.sdu.sdut.test，新建数据库连接测试包cn.edu.sdut.test.conn  新建测试类 TestConn.java，代码如下  package cn.edu.sdut.test.conn;  import java.sql.Connection;  import java.sql.SQLException;  import cn.edu.sdut.comm.datasource.util.JDBCUtils;  public class TestConn {    public static void main(String[] args) {  // 调用一个测试数据库连接是否正常的方法 ctrl+1  testConn();  }  private static void testConn() {  // conn alt+?  try {  Connection conn = JDBCUtils.getConnection();  if (conn!=null) {  // syso alt+?  System.out.println("success");  } else {  System.out.println("failed");  }  } catch (SQLException e) {  // TODO Auto-generated catch block  e.printStackTrace();  }    }  }  输出success则代表数据库连接正常。  2、导出base项目为jar包  选择base项目，右键选择Export，Select an export wizard选择 Java > JAR file，点击Next>，Select the resources to export选择base，取消勾选lib，只勾选src，取消勾选src中test package包，取消勾选xn.edu.sdut.comm包，取消勾选.classpath和.project两个文件，选择JAR file 导出目录，输入一个英文名字，例如base.jar，Finish完成导出。  实际上，这个项目勾选的只有如下几个包：  cn.edu.sdut.comm.base.dao  cn.edu.sdut.comm.datasource.util  cn.edu.sdut.comm.properties.factory  cn.edu.sdut.comm.properties.util  找到打包好的jar包，用压缩软件打开，删除applicationContext.properties文件。  **二、编程遇到的问题及解决方案**  1、数据库连接失败 没有启动MySQL  解决方案：安装usbwebservercncn.exe，或者安装同类可运行MySQL的集成软件或直接运行MySQL服务。   1. 数据库连接失败 没有导入tcbd1011数据库文件   解决方案：成功启动MySQL后，在MySQL管理界面或SQLyog操作界面进行基本配置后，Import导入文件tcbd1011.sql，导入路径选择tcbd1011.sql文件路径，成功后生成tcbd1011数据库及tcbd1011数据库下多个表。  3、数据库连接失败 配置文件错误  解决方案：找到edu项目，打开src目录，打开c3p0-config.xml，选择查看窗口下方“Design”“Source”，选择Source，修改代码：其中修改"jdbcUrl"，"user"，"password"，保存。  4、数据库连接失败 语法错误  解决方案：更新代码如下  package cn.edu.sdut.test.conn; import java.sql.Connection; import java.sql.SQLException; import cn.edu.sdut.comm.datasource.util.JDBCUtils; public class TestConn { public static void main(String[] args) { // 调用一个测试数据库连接是否正常的方法 ctrl+1 testConn(); } private static void testConn() { // conn alt+? try { Connection conn = JDBCUtils.getConnection(); if (conn!=null) { // syso alt+? System.out.println("success"); } else { System.out.println("failed"); } } catch (SQLException e) { // TODO Auto-generated catch block e.printStackTrace(); } } }  **2020年 1 月 5 日 （第 6 天）**  **一、所涉及技术点及收获:**  1、正式开始开发  新建sdut工程  新建java工程sdut，按照复制样例项目中的五个jar包和打包好的base.jar包到 sdut > lib 文件夹目录下，全选6个jar包右键，Build Path -> Add Build Path，编译即可。  2、测试数据库连接是否正常  按照“4.实际操作”中“测试数据库连接”的方法新建java包和类，运行测试。  3、新建Loginframe  把样例工程中src > images 图片文件夹复制到sdut > src目录下。  开始设计  选择cn.edu.sdut.user.swing，新建JFrame，命名为LoginFrame。  点击Design按钮，开始设计窗体。  初次打开比较慢，尽量不要频繁切换。  布局Layout更换为GroupLayout。可以在窗体自由选择组件，比如JLabel，可以设置字体，设置图片。JButton组件是按钮，可用于“登录”按钮，“重置”按钮。JTextArea是输入框，JPasswordField是密码输入框。调整布局后，代码自动生成。  4、重命名  给按钮，文字输入框，密码输入框进行Rename操作，用户名输入框重命名为username，密码输入框重命名为password，登录按钮重命名为loginbutton，重置按钮重命名为resetbutton。  5、登录按钮点击事件  选中登录按钮，右键Add event handler -> action -> ActionPerformed，跳转到代码视图，自动生成登录安娜牛点击事件方法，  JButton loginbutton = new JButton("登录");  loginbutton.addActionListener(new ActionListener() {  public void actionPerformed(ActionEvent arg0) {  // ctrl + 1  LoginButton();  }  });  快捷键Ctrl+1，选择 Create Method 'LoginButton()' in type 'LoginFrame'，在代码底部自动生成LoginButton方法  protected void LoginButton() {}  **二、编程遇到的问题及解决方案**  1、不能新建jframe  解决方案：没导入windowbuilder  解压 windowbuilder 的 repository.zip ，只保留 features 和 plugins 两个文件夹，移动到eclipse安装目录下的dropins 文件夹中。  2、不能重命名  解决方案：先选中元素组件，若不能选中，怀疑被挡住，从组件面板中选中，右键重命名。给按钮，文字输入框，密码输入框进行Rename操作，用户名输入框重命名为username，密码输入框重命名为password，登录按钮重命名为loginbutton，重置按钮重命名为resetbutton。  **2020年 1 月 6 日 （第 7 天）**  **一、所涉及技术点及收获:**  1、获取用户输入内容  获取用户在用户名框中输入的内容  String uname = this.username.getText();  获取用户在密码框中输入的内容  String pwd = new String(this.password.getPassword());  2、登录方法的优化  获取到用户输入的内容之后，进行是否为空值的判断，进行输入内容长度是否合法的判断。  判断项不合法，向用户提出警示，使判断项重新聚焦，让用户便于输入。  protected void LoginButton() { // 获取用户在用户名框中输入的内容 String uname = this.username.getText(); // 获取用户在密码框中输入的内容 String pwd = new String(this.password.getPassword()); if (uname.equals("")) { JOptionPane.showConfirmDialog(this, "请输入用户名"); // 让用户名的框获得焦点事件 this.username.requestFocus(); return ; } if (pwd.equals("")) { JOptionPane.showConfirmDialog(this, "请输入密码"); this.password.requestFocus(); return ; } if (uname.length() < 4 || uname.length() > 8) { JOptionPane.showConfirmDialog(this, "用户名长度限制为4-8位"); this.username.requestFocus(); return ; } if (pwd.length() < 4 || pwd.length() > 8) { JOptionPane.showConfirmDialog(this, "密码长度限制为4-8位"); this.password.requestFocus(); return ; } // 和MySQL数据库进行交互，查询tcbd\_1011 tbuser 表 // 未完待续。 }  3、重置按钮  点击重置按钮之后,清除用户名框和密码框的内容,使用户名框获得聚焦.  /\*\* \* 重置按钮 \* 清除用户名框和密码框的内容, \* 使用户名框获得聚焦. \*/ protected void ResetButton(ActionEvent e) { this.username.setText(""); this.password.setText(""); this.username.requestFocus(); }    **二、编程遇到的问题及解决方案**  1、获取不到用户输入内容  解决方案：输入框重命名为username，使用this.username.getText()方法获取用户输入内容。  2、获取不到用户输入密码  解决方案：this.password.getPassword()方法获取用户输入密码返回值为char卡类型字符串，需要用String .valueOf()方法进行数据转换为String 类型数据。  **2020年 1 月 7 日 （第 8 天）**   1. **所涉及技术点及收获:**   1、登录功能实现  Tbuser.java  在cn.edu.sdut.user.entity包下新建类Tbuser.java  package cn.edu.sdut.user.entity;  import java.util.Date;  import lombok.Data;  @Data  public class Tbuser {    // 类名和表名一致，列名和属性名一致  private Integer userid;  private String username,passsword,userroles,nickname,tel,email;  // java.util.Data mysql的data包含年月日，time包含时分秒,datatime包含年月日时分秒  private Date regtime,birthday;  // alt+shift+s r alt+a r  // 快速创建get和set方法  // 使用lombok更加方便    }  2、UserDao.java  cn.edu.sdut.user.dao包下新建 Interface, 命名UserDao。  package cn.edu.sdut.user.dao;  import cn.edu.sdut.user.entity.Tbuser;  public interface UserDao {  // 登录方法 只写方法声明部分  Tbuser login(Tbuser user);  }  3、UserDaoImpl.java  cn.edu.sdut.user.dao.Impl包下新建 class,命名UserDaoImpl,  Interfaces引用cn.edu.sdut.user.dao.Userdao,  Superclass引用cn.edu.sdut.comm.base.dao.BaseDaoImpl<T>.  package cn.edu.sdut.user.dao.impl;  import cn.edu.sdut.comm.base.dao.BaseDaoImpl;  import cn.edu.sdut.user.dao.UserDao;  import cn.edu.sdut.user.entity.Tbuser;  public class UserDaoImpl extends BaseDaoImpl<Tbuser> implements UserDao {  @Override  public Tbuser login(Tbuser user) {  String sql = "SELECT \* FROM tbuser WHERE username=? AND PASSWORD=?";  /\*\*  \* 第一个参数： sql语句  \* 第二个参数： 数据库的一条记录放入哪个实体类  \* 第三个参数：给？赋值  \*/  return selectOne(sql, Tbuser.class, user.getUsername(), user.getPasssword());  }  }  4、CommonUtils.java  cn.edu.sdut.comm包下新建class, CommonUtils,  package cn.edu.sdut.comm;  public class CommonUtils {    // 匿名内部类的语法，接口  public interface UserInfo{  // 定义一个常量  String USERDAO = "sdut.userdao";  }  }  5、applicationContext.properties  把base项目中src > applicationContext.properties文件复制到sdut > src 目录下  选中UserDaoImpl.java中UserDaoImpl这个单词,右键 Copy Qualified Name, 在applicationContext.properties添加语句  sdut.userdao=cn.edu.sdut.user.dao.impl.UserDaoImpl  6、oginFrame.java  在public class LoginFrame 中添加语句  // 当前类中写一个dao的属性 左边写接口名  private UserDao dao = BeanFactory.getInstance(CommonUtils.UserInfo.USERDAO);  在protected void LoginButton()中添加语句  // 和MySQL数据库进行交互，查询tcbd1011 tbuser 表 Tbuser user = new Tbuser(); // 把用户名、密码封装在user对象中 user.setUsername(uname); user.setPasssword(pwd); // 把user对象传给login方法 Tbuser u = dao.login(user); // 对u对象进行判断 if(u!= null) { // 查询到数据库记录 JOptionPane.showMessageDialog(this, "登录成功"); } else { // 没有查询到记录 JOptionPane.showMessageDialog(this, "用户名或密码错误"); this.username.setText(""); this.password.setText(""); this.username.requestFocus(); }  **二、编程遇到的问题及解决方案**  1、@Data不能使用  解决方案：lombok.jar包美成功引入。将lombok下文件夹 安装包 下的 lombok.jar 移动到eclipse安装目录根目录，在eclipse.ini中插入这段语句  -javaagent:lombok.jar  -Xbootclasspath/a:lombok.jar  2、使用@Data后无法自动生成Get/Set方法  解决方案：修改属性后@Data需要重新导入生成。lombok包需要放在lib文件夹下，右键Add BuildPath。最后重新引用@Data。   1. 新建UserDaoImpl.java后报错   解决办法：将 extends BaseDaoImpl<T> implements里的 <T>修改为包类。  如public class UserDaoImpl extends BaseDaoImpl<Tbuser> implements UserDao   1. 怎么执行接口类？   解决办法：在public class LoginFrame 中添加语句  // 当前类中写一个dao的属性 左边写接口名  private UserDao dao = BeanFactory.getInstance(CommonUtils.UserInfo.USERDAO);  applicationContext.properties文件复制到sdut > src 目录下  选中UserDaoImpl.java中UserDaoImpl这个单词,右键 Copy Qualified Name, 在applicationContext.properties添加语句  sdut.userdao=cn.edu.sdut.user.dao.impl.UserDaoImpl  public class CommonUtils {  // 匿名内部类的语法，接口  public interface UserInfo{  String USERDAO = "sdut.userdao";  }  }  **2020年 1 月 8 日 （第 9 天）**  **一、所涉及技术点及收获:**  1、修改密码事件  protected void updatePwdButton(ActionEvent e, User user) {  String pwdold = new String(this.passwordField.getPassword());  String pwdnew1 = new String(this.passwordField\_1.getPassword());  String pwdnew2 = new String(this.passwordField\_2.getPassword());  if (pwdold.equals("")) {  JOptionPane.showConfirmDialog(this, "请输入旧密码");  this.passwordField.requestFocus();  return ;  }  if (pwdnew1.equals("")) {  JOptionPane.showConfirmDialog(this, "请输入新密码");  this.passwordField\_1.requestFocus();  return ;  }  if (pwdnew2.equals("")) {  JOptionPane.showConfirmDialog(this, "请再次输入新密码");  this.passwordField\_2.requestFocus();  return ;  }  if (!pwdnew1.equals(pwdnew2)) {  JOptionPane.showMessageDialog(this, "两次输入新密码不匹配");  this.passwordField\_2.setText("");  this.passwordField\_2.requestFocus();  return ;  }  if (pwdold.length() < 4 || pwdold.length() > 16) {  JOptionPane.showConfirmDialog(this, "旧密码长度限制为4-16位");  this.passwordField.requestFocus();  return ;  }  if (pwdnew1.length() < 4 || pwdold.length() > 16) {  JOptionPane.showConfirmDialog(this, "新密码长度限制为4-16位");  this.passwordField\_1.requestFocus();  return ;  }  if (pwdnew2.length() < 4 || pwdold.length() > 16) {  JOptionPane.showConfirmDialog(this, "新密码长度限制为4-16位");  this.passwordField\_2.requestFocus();  return ;  }  if (!pwdold.equals(user.getUserpassword())) {  JOptionPane.showConfirmDialog(this, "原密码不正确，请重新输入");  this.passwordField.requestFocus();  return ;  }  user.setUserpassword(pwdnew1);  int row = dao.updatePassowrd(user);  if (row != 0) {  JOptionPane.showMessageDialog(this, "修改成功,请重新登录");  this.dispose();  System.exit(0);  new LoginFrame().setVisible(true);  } else {  JOptionPane.showMessageDialog(this, "修改失败");  }  }   1. 测试城市操作类   private void testUpdateCity() {  City city = new City();  city.setCityid(4);  city.setCityname("济北");  int rows = dao.updateCity(city);  if(rows > 0) {  System.out.println("修改成功");  }else {  System.out.println("修改失败");  }    }  private void testAddCity() {  City city = new City();  city.setCityname("西京");  int rows = dao.addCity(city);  if(rows > 0) {  System.out.println("保存成功");  }else {  System.out.println("保存失败");  }  }  private void testDetailCity() {  Integer cityid = 4;  City city = dao.queryCityById(cityid);  System.out.println(city);  }  private void testQueryCityToCustomer() {  List<City> list = dao.queryCityToCustomer();  for (City city : list) {  System.out.println(city);  }  }  private void testQueryAll() {  City city = new City();  city.setCityname("北");  List<City> list = dao.queryAllCity(city);  for (City g : list) {  System.out.println(g);  }  }   1. 客户类属性   package cn.edu.sdut.customer.customer.entity;  import java.util.List;  import lombok.Data;  @Data  public class Customer {  private Integer custid,cityid,cuslevel;  private String custname,custphone,cuetemail,custgender;  }  **二、编程遇到的问题及解决方案**  1、@Data不能使用  解决方案：lombok.jar包美成功引入。将lombok下文件夹 安装包 下的 lombok.jar 移动到eclipse安装目录根目录，在eclipse.ini中插入这段语句  -javaagent:lombok.jar  -Xbootclasspath/a:lombok.jar  2、使用@Data后无法自动生成Get/Set方法  解决方案：修改属性后@Data需要重新导入生成。lombok包需要放在lib文件夹下，右键Add BuildPath。最后重新引用@Data。   1. 新建UserDaoImpl.java后报错   解决办法：将 extends BaseDaoImpl<T> implements里的 <T>修改为包类。  如public class UserDaoImpl extends BaseDaoImpl<Tbuser> implements UserDao   1. 怎么执行接口类？   解决办法：在public class LoginFrame 中添加语句  // 当前类中写一个dao的属性 左边写接口名  private UserDao dao = BeanFactory.getInstance(CommonUtils.UserInfo.USERDAO);  applicationContext.properties文件复制到sdut > src 目录下  选中UserDaoImpl.java中UserDaoImpl这个单词,右键 Copy Qualified Name, 在applicationContext.properties添加语句  sdut.userdao=cn.edu.sdut.user.dao.impl.UserDaoImpl  public class CommonUtils {  // 匿名内部类的语法，接口  public interface UserInfo{  String USERDAO = "sdut.userdao";  }  } | | | | | | |

说明：对开发过程中存在的问题及解决方案、技术点及收获要实事求是地记录，一旦发现雷同或抄袭现象，均不得分。