**计算机工程学院**

**课**

**程**

**设**

**计**

**报**

**告**

|  |  |
| --- | --- |
| 课 程 名： | Java程序设计 |
| 设计题目： | 车辆管理系统 |
| 专业班级： |  |
| 姓 名： |  |
| 指导教师： |  |

2014年12月制

|  |  |  |
| --- | --- | --- |
| **设计**任务 | 设计一个车辆管理系统系统，实现汽车入库，汽车出库，汽车查询，汽车信息修改和管理员登录等功能。 | |
| 完成任务的思路、方法与算法  创建登录页面，  wps    Users表设计 | | |
| 程序框图  wps | | |
| 核心程序（对主要的语句要加注释）  Package controller;  CarDao.java   1. **package** controller; 3. **import** model.Cars; 5. **import** java.sql.Connection; 6. **import** java.sql.ResultSet; 7. **import** java.sql.SQLException; 8. **import** java.sql.Statement; 9. **import** java.util.ArrayList; 10. **import** java.util.List; 12. /\*\* 13. \* 汽车dao类 14. \*/ 15. **public** **class** CarDao { 16. /\*\* 17. \* 获取所有汽车信息 18. \* @param connection 19. \* @return 20. \* @throws SQLException 21. \*/ 22. **public** List<Cars> getAllCars(Connection connection) **throws** SQLException { 23. List<Cars> resCar = **new** ArrayList<Cars>(); 24. Cars currentCar = **null**; 25. String sql = "select \* from cars"; 26. Statement statement = connection.createStatement(); 27. ResultSet res = statement.executeQuery(sql); 28. **while**(res.next()){ 29. String plate = res.getString("plate"); 30. String model = res.getString("model"); 31. String owner = res.getString("owner"); 32. String tel = res.getString("tel"); 33. String color = res.getString("color"); 34. currentCar = **new** Cars(plate,model,owner,tel,color); 35. resCar.add(currentCar); 36. } 37. **return** resCar; 38. } 40. /\*\* 41. \* 插入汽车数据，汽车入库 42. \* @param connection 43. \* @param cars 44. \* @throws SQLException 45. \*/ 46. **public** **void** addCar(Connection connection, Cars cars) **throws** SQLException { 47. String sql = "insert into cars values (\"plate\",\"model\",\"owner\",\"tel\",\"color\")"; 48. sql = sql.replace("plate",cars.getPlate()); 49. sql = sql.replace("model",cars.getModel()); 50. sql = sql.replace("owner",cars.getOwner()); 51. sql = sql.replace("tel",cars.getTel()); 52. sql = sql.replace("color",cars.getColor()); 53. System.out.println(sql); 54. Statement statement = connection.createStatement(); 55. statement.executeUpdate(sql); 56. } 58. /\*\* 59. \* 删除汽车数据 60. \* @param connection 61. \* @param plate 62. \* @return 63. \* @throws SQLException 64. \*/ 65. **public** **boolean** subCar(Connection connection,String plate) **throws** SQLException { 66. String sql = "delete from cars where plate = \"xxx\";"; 67. sql = sql.replace("xxx",plate); 68. Statement statement = connection.createStatement(); 69. **int** row = statement.executeUpdate(sql); 70. System.out.println(sql); 71. **if**(row!=0){ 72. **return** **true**; 73. }**else** { 74. **return** **false**; 75. } 76. } 78. /\*\* 79. \* 查询汽车数据 80. \* @param connection 81. \* @param condition 82. \* @param type 83. \* @return 84. \* @throws SQLException 85. \*/ 86. **public** List<Cars> findCar(Connection connection, String condition, **int** type) **throws** SQLException { 87. String sql = **null**; 88. List<Cars> resCar = **new** ArrayList<>(); 89. Cars currentCar = **null**; 90. **switch** (type){ 91. **case** 1: 92. sql = "select \*from cars where plate like \"%condition%\""; 93. **break**; 94. **case** 2: 95. sql = "select \*from cars where model like \"%condition%\""; 96. **break**; 97. **case** 3: 98. sql = "select \*from cars where owner like \"%condition%\""; 99. **break**; 100. **case** 4: 101. sql = "select \*from cars where tel like \"%condition%\""; 102. **break**; 103. **case** 5: 104. sql = "select \*from cars where color like \"%condition%\""; 105. **break**; 106. **default**: 107. **return** resCar; 108. } 109. sql = sql.replace("condition",condition); 110. System.out.println(sql); 111. Statement statement = connection.createStatement(); 112. ResultSet res = statement.executeQuery(sql); 113. **while**(res.next()){ 114. String plate = res.getString("plate"); 115. String model = res.getString("model"); 116. String owner = res.getString("owner"); 117. String tel = res.getString("tel"); 118. String color = res.getString("color"); 119. currentCar = **new** Cars(plate,model,owner,tel,color); 120. resCar.add(currentCar); 121. } 122. **return** resCar; 123. } 125. /\*\* 126. \* 查询单条学生信息 用于修改 127. \* @param connection 128. \* @param p 129. \* @return 130. \* @throws SQLException 131. \*/ 132. **public** Cars findModifyTarget(Connection connection, String p) **throws** SQLException { 133. Cars TargetCar = **null**; 134. String sql = "select \*from cars where plate = \"???\";"; 135. sql = sql.replace("???",p); 136. System.out.println(sql); 137. Statement statement = connection.createStatement(); 138. ResultSet res = statement.executeQuery(sql); 139. **while**(res.next()){ 140. String plate = res.getString("plate"); 141. String model = res.getString("model"); 142. String owner = res.getString("owner"); 143. String tel = res.getString("tel"); 144. String color = res.getString("color"); 145. TargetCar = **new** Cars(plate,model,owner,tel,color); 146. } 147. **return** TargetCar; 148. } 150. /\*\* 151. \* 更新数据 152. \* @param connection 153. \* @param handledTarget 154. \* @return 155. \* @throws SQLException 156. \*/ 157. **public** **boolean** modifyTarget(Connection connection, Cars handledTarget) **throws** SQLException { 158. String sql = "update cars set model=\"@\",owner=\"#\",tel=\"$\",color=\"%\" where plate=\"^\";"; 159. sql = sql.replace("@",handledTarget.getModel()); 160. sql = sql.replace("#",handledTarget.getOwner()); 161. sql = sql.replace("$",handledTarget.getTel()); 162. sql = sql.replace("%",handledTarget.getColor()); 163. sql = sql.replace("^",handledTarget.getPlate()); 164. System.out.println(sql); 165. Statement statement = connection.createStatement(); 166. //执行更新返回影响行数 167. **int** res = statement.executeUpdate(sql); 168. **if**(res>0){ 169. **return** **true**; 170. }**else** { 171. **return** **false**; 172. } 173. } 175. **public** **static** **void** main(String[] args) **throws** SQLException, ClassNotFoundException { 176. //        Connection connection =new DButil().getCon(); 177. //        List<Cars> cars = findCar(connection,"车",2); 178. //        for(int i=0; i<cars.size(); i++){ 179. //            System.out.println(cars.get(i).getPlate()); 180. //        } 181. String sql = "update cars set model=\"@\",owner=\"#\",tel=\"$\",color=\"%\" where plate=\"^\";"; 182. sql = sql.replace("@","1"); 183. sql = sql.replace("#","####"); 184. sql = sql.replace("$","$$$$"); 185. System.out.println(sql); 186. } 187. }   UserDao.java   1. **package** controller; 3. **import** model.User; 5. **import** java.sql.Connection; 6. **import** java.sql.ResultSet; 7. **import** java.sql.SQLException; 8. **import** java.sql.Statement; 10. /\*\* 11. \* 用户DAO类 12. \*/ 13. **public** **class** UserDao { 14. /\*\* 15. \* 登录验证 16. \* @param connection 17. \* @param user 18. \* @return 19. \* @throws SQLException 20. \*/ 21. **public** User login(Connection connection, User user) **throws** SQLException { 22. User resUser = **null**; 23. String sql = "select \* from users where username = "+'"'+user.getUsername()+'"'+" and password="+user.getPassword(); 24. Statement statement = connection.createStatement(); 25. ResultSet res = statement.executeQuery(sql); 26. //查到结果就将结果存到resUser中返回 27. **if**(res.next()){ 28. resUser = **new** User(res.getString("username"),res.getString("password")); 29. resUser.setId(res.getInt("id")); 30. } 31. **return** resUser; 32. } 33. }   Package model;  Cars.java   1. **package** model; 3. /\*\*汽车信息实体 4. \* 5. \*/ 6. **public** **class** Cars { 7. **private** String plate; 8. **private** String model; 9. **private** String owner; 10. **private** String tel; 11. **private** String color; 13. **public** Cars(String plate, String model, String owner, String tel, String color) { 14. **this**.plate = plate; 15. **this**.model = model; 16. **this**.owner = owner; 17. **this**.tel = tel; 18. **this**.color = color; 19. } 21. **public** String getPlate() { 22. **return** plate; 23. } 25. **public** **void** setPlate(String plate) { 26. **this**.plate = plate; 27. } 29. **public** String getModel() { 30. **return** model; 31. } 33. **public** **void** setModel(String model) { 34. **this**.model = model; 35. } 37. **public** String getOwner() { 38. **return** owner; 39. } 41. **public** **void** setOwner(String owner) { 42. **this**.owner = owner; 43. } 45. **public** String getTel() { 46. **return** tel; 47. } 49. **public** **void** setTel(String tel) { 50. **this**.tel = tel; 51. } 53. **public** String getColor() { 54. **return** color; 55. } 57. **public** **void** setColor(String color) { 58. **this**.color = color; 59. } 60. }   Users.java   1. **package** model; 3. /\*\* 4. \* 用户实体 5. \*/ 6. **public** **class** User { 7. //用户编号 姓名 密码 8. **private** **int** id; 9. **private** String username; 10. **private** String password; 12. **public** User(String username, String password) { 13. **this**.username = username; 14. **this**.password = password; 15. } 17. **public** **int** getId() { 18. **return** id; 19. } 21. **public** String getUsername() { 22. **return** username; 23. } 25. **public** String getPassword() { 26. **return** password; 27. } 29. **public** **void** setId(**int** id) { 30. **this**.id = id; 31. } 33. **public** **void** setUsername(String username) { 34. **this**.username = username; 35. } 37. **public** **void** setPassword(String password) { 38. **this**.password = password; 39. } 40. }   Package utils;  DButil.java   1. **package** utils; 3. **import** java.sql.Connection; 4. **import** java.sql.DriverManager; 5. **import** java.sql.SQLException; 7. /\*\* 8. \* 此类用于jdbc连接数据库 9. \*/ 10. **public** **class** DButil { 11. **private** String DBurl = "jdbc:mysql://localhost:3309/车辆管理系统"; 12. **private** String DBusername = "root"; 13. **private** String DBpassword = "123456"; 14. **private** String jdbcName = "com.mysql.cj.jdbc.Driver"; 16. /\*\* 17. \* 获取数据库连接 18. \* @return 19. \* @throws ClassNotFoundException 20. \* @throws SQLException 21. \*/ 22. **public** Connection getCon() **throws** ClassNotFoundException, SQLException { 23. Class.forName(jdbcName); 24. Connection connection = DriverManager.getConnection(DBurl,DBusername,DBpassword); 25. **return** connection; 26. } 28. /\*\* 29. \* 关闭数据库连接 30. \*/ 31. **public** **void** closeConnection(Connection connection) **throws** SQLException { 32. **if**(connection!=**null**){ 33. connection.close(); 34. } 35. } 36. }   VarifyInput.java   1. **package** utils; 3. **public** **class** VarifyInput { 4. **public** **static** **boolean** isEmpty(String value){ 5. **if**(value == ""|| value.trim().equals("")){ 6. **return** **true**; 7. } 8. **return** **false**; 9. } 10. }   Package views.actions;  Modifycarlistener.java   1. **package** views.Actions; 3. **import** com.sun.tools.javac.Main; 4. **import** controller.CarDao; 5. **import** model.Cars; 6. **import** utils.DButil; 7. **import** utils.VarifyInput; 8. **import** views.MainTable; 9. **import** views.ModifyCar; 11. **import** javax.swing.\*; 12. **import** java.awt.event.ActionEvent; 13. **import** java.awt.event.ActionListener; 14. **import** java.sql.Connection; 15. **import** java.sql.SQLException; 17. /\*\* 18. \* 提交修改汽车信息的事件 19. \*/ 20. **public** **class** ModifyCarListener **implements** ActionListener { 21. ModifyCar modifyCar; 23. **public** **void** setModifyCar(ModifyCar modifyCar) { 24. **this**.modifyCar = modifyCar; 25. } 27. @Override 28. **public** **void** actionPerformed(ActionEvent actionEvent) { 29. DButil dButil = **new** DButil(); 30. **try** { 31. Connection connection = dButil.getCon(); 32. //获取输入框数据发起请求更改数据 33. String plate = modifyCar.plateInput.getText(); 34. String handledModel = modifyCar.modelInput.getText(); 35. String handledOwner = modifyCar.ownerInput.getText(); 36. String handledtel = modifyCar.telInput.getText(); 37. String handledColor = modifyCar.colorInput.getText(); 38. plate = plate.replace("请输入车牌号",""); 39. **if**(VarifyInput.isEmpty(plate)){ 40. JOptionPane.showMessageDialog(**null**,"未输入正确车牌号信息","警告",JOptionPane.WARNING\_MESSAGE); 41. **return**; 42. } 43. System.out.println(plate+"\n"+handledModel+"\n"+handledOwner+"\n"+handledtel+"\n"+handledColor); 44. Cars handledTarget = **new** Cars(plate,handledModel,handledOwner,handledtel,handledColor); 45. **boolean** SUCCESS =  **new** CarDao().modifyTarget(connection,handledTarget); 46. **if**(SUCCESS){ 47. JOptionPane.showMessageDialog(**null**,"修改成功"); 48. modifyCar.dispose(); 49. } 50. //修改视图 51. //获取操作表格的行号 52. **int** rowCount = 0; 53. **for**(**int** i = 0; i< MainTable.tableModel.getRowCount(); i++){ 54. **if**(MainTable.tableModel.getValueAt(i,0)==plate){ 55. **return**; 56. } 57. } 58. MainTable.tableModel.setValueAt(handledModel,rowCount,1); 59. MainTable.tableModel.setValueAt(handledOwner,rowCount,2); 60. MainTable.tableModel.setValueAt(handledtel,rowCount,3); 61. MainTable.tableModel.setValueAt(handledColor,rowCount,4); 62. } **catch** (ClassNotFoundException | SQLException e) { 63. **throw** **new** RuntimeException(e); 64. } 65. } 66. }   Removecarlistener.java   1. **package** views.Actions; 3. **import** controller.CarDao; 4. **import** utils.DButil; 5. **import** views.MainTable; 7. **import** javax.swing.\*; 8. **import** java.awt.event.ActionEvent; 9. **import** java.awt.event.ActionListener; 10. **import** java.sql.Connection; 11. **import** java.sql.SQLException; 13. **public** **class** RemoveCarListener **implements** ActionListener { 14. MainTable mainTable; 15. **public** **void** setMainTable(MainTable mainTable) { 16. **this**.mainTable = mainTable; 17. } 18. @Override 19. **public** **void** actionPerformed(ActionEvent actionEvent) { 20. //重新绘制窗口 21. **try** { 22. **try** { 23. Connection connection = **new** DButil().getCon(); 24. //根据行数删除数据 25. **int** count = mainTable.table.getSelectedRow(); 26. String plate =mainTable.table.getValueAt(count,0).toString(); 27. //表格模型删除对应列 28. mainTable.tableModel.removeRow(count); 29. **boolean** SUCCESS = **new** CarDao().subCar(connection,plate); 30. **if**(SUCCESS){ 31. JOptionPane.showMessageDialog(**null**,"删除成功"); 32. }**else** { 33. JOptionPane.showMessageDialog(**null**,"删除失败","出现异常",JOptionPane.ERROR\_MESSAGE); 34. } 35. }**catch** (ArrayIndexOutOfBoundsException e){ 36. //没有选中车辆则抛出异常 37. JOptionPane.showMessageDialog(**null**,"没有选中车辆！","警告",JOptionPane.WARNING\_MESSAGE); 38. } 39. } **catch** (ClassNotFoundException | SQLException e) { 40. JOptionPane.showMessageDialog(**null**,"没有选中车辆！","",JOptionPane.WARNING\_MESSAGE); 41. } 42. } 43. }   Searchcarlistener.java   1. **package** views.Actions; 3. **import** controller.CarDao; 4. **import** model.Cars; 5. **import** utils.DButil; 6. **import** utils.VarifyInput; 7. **import** views.QueryCar; 9. **import** javax.swing.\*; 10. **import** java.awt.event.ActionEvent; 11. **import** java.awt.event.ActionListener; 12. **import** java.sql.Connection; 13. **import** java.sql.SQLException; 14. **import** java.util.List; 16. /\*\* 17. \* 汽车查询 18. \* 根据不同条件查询数据 19. \* 数据显示到表格中 20. \*/ 21. **public** **class** SearchCarListener **implements** ActionListener { 22. QueryCar queryCar; 24. **public** **void** setQueryCar(QueryCar queryCar) { 25. **this**.queryCar = queryCar; 26. } 28. @Override 29. **public** **void** actionPerformed(ActionEvent actionEvent) { 30. //清除所有tablemodel中的数据 31. **if** (queryCar.tableModel.getRowCount() > 0) { 32. **for** (**int** i = queryCar.tableModel.getRowCount() - 1; i > -1; i--) { 33. queryCar.tableModel.removeRow(i); 34. } 35. } 36. DButil dButil = **new** DButil(); 37. **try** { 38. Connection connection = dButil.getCon(); 39. String condition = queryCar.conditions.getText(); 40. **int** type = queryCar.comboBox.getSelectedIndex(); 41. **if**(type==0 || VarifyInput.isEmpty(condition)){ 42. JOptionPane.showMessageDialog(**null**,"没有使用条件！","警告",JOptionPane.WARNING\_MESSAGE); 43. } 44. List<Cars> queryRes = **new** CarDao().findCar(connection,condition,type); 45. Object TableContent[][] = **new** Object[queryRes.size()][5]; 46. **for**(**int** i=0; i<queryRes.size();i++){ 47. TableContent[i][0] = queryRes.get(i).getPlate(); 48. TableContent[i][1] = queryRes.get(i).getModel(); 49. TableContent[i][2] = queryRes.get(i).getOwner(); 50. TableContent[i][3] = queryRes.get(i).getTel(); 51. TableContent[i][4] = queryRes.get(i).getColor(); 52. queryCar.tableModel.addRow(TableContent[i]); 53. } 54. dButil.closeConnection(connection); 55. } **catch** (ClassNotFoundException | SQLException e) { 56. **throw** **new** RuntimeException(e); 57. } 58. } 59. }   Submitcarlistener.java   1. **package** views.Actions; 3. **import** controller.CarDao; 4. **import** model.Cars; 5. **import** utils.DButil; 6. **import** utils.VarifyInput; 7. **import** views.AddCar; 8. **import** views.MainTable; 10. **import** javax.swing.\*; 11. **import** java.awt.event.ActionEvent; 12. **import** java.awt.event.ActionListener; 13. **import** java.sql.Connection; 14. **import** java.sql.SQLException; 16. /\*\* 17. \* 用于提交添加汽车信息 18. \*/ 19. **public** **class** SubmitAddCarListener **implements** ActionListener { 20. AddCar addCar; 22. **public** **void** setAddCar(AddCar addCar) { 23. **this**.addCar = addCar; 24. } 26. @Override 27. **public** **void** actionPerformed(ActionEvent actionEvent) { 28. DButil dButil = **new** DButil(); 29. **try** { 30. Connection connection = dButil.getCon(); 31. String plate = addCar.plateInput.getText(); 32. String model = addCar.modelInput.getText(); 33. String owner = addCar.ownerInput.getText(); 34. String tel = addCar.telInput.getText(); 35. String color = addCar.colorInput.getText(); 36. plate = plate.replace("请输入车牌号",""); 37. **if**( VarifyInput.isEmpty(plate)){ 38. JOptionPane.showMessageDialog(addCar,"车牌号不能为空！","出错了",JOptionPane.ERROR\_MESSAGE); 39. **return**; 40. } 41. Cars car = **new** Cars(plate,model,owner,tel,color); 42. **new** CarDao().addCar(connection,car); 43. Object[] addRow = {plate,model,owner,tel,color}; 44. MainTable.tableModel.addRow(addRow); 45. dButil.closeConnection(connection); 46. addCar.dispose(); 47. } **catch** (ClassNotFoundException | SQLException e) { 48. JOptionPane.showMessageDialog(addCar,"该汽车信息已存在，无法添加该信息","出错了",JOptionPane.ERROR\_MESSAGE); 49. } 50. } 51. }   Exitlister.java   1. **package** views.Actions; 3. **import** views.Login; 4. **import** views.MainTable; 6. **import** javax.swing.\*; 7. **import** java.awt.event.ActionEvent; 8. **import** java.awt.event.ActionListener; 10. **public** **class** ExitListener **implements** ActionListener { 11. MainTable mainTable; 12. **public** **void** setMainTable(MainTable mainTable) { 13. **this**.mainTable = mainTable; 14. } 15. @Override 16. **public** **void** actionPerformed(ActionEvent actionEvent) { 17. mainTable.dispose(); 18. Login login = **new** Login(); 19. login.usernameContext.setText(""); 20. login.passwordContext.setText(""); 21. } 22. }   Package views;  Addcar.java   1. **package** views; 3. **import** utils.VarifyInput; 4. **import** views.Actions.SubmitAddCarListener; 6. **import** javax.swing.\*; 7. **import** java.awt.\*; 8. **import** java.awt.event.FocusEvent; 9. **import** java.awt.event.FocusListener; 11. **public** **class** AddCar **extends** JFrame { 12. JFrame addCar; 13. **public** JTextField plateInput = **new** JTextField(20); 14. **public** JTextField modelInput = **new** JTextField(20); 15. **public** JTextField ownerInput = **new** JTextField(20); 16. **public** JTextField telInput = **new** JTextField(20); 17. **public** JTextField colorInput = **new** JTextField(20); 18. JPanel panel = **new** JPanel(); 19. JPanel mainpan = **new** JPanel(); 20. JButton addBtn = **new** JButton("入库"); 21. JButton cancelBtn = **new** JButton("取消"); 22. JPanel bottompan = **new** JPanel(); 23. JPanel top = **new** JPanel(); 24. JLabel title = **new** JLabel("汽车入库"); 25. SubmitAddCarListener submitAddCarListener; 27. **public** AddCar() **throws** HeadlessException { 28. addCar = **new** JFrame("添加汽车"); 29. setSize(500,400); 30. setLocationRelativeTo(**null**); 31. setResizable(**false**); 33. add(panel); 34. panel.setLayout(**new** BorderLayout()); 35. panel.add(top,BorderLayout.NORTH); 36. top.setLayout(**new** FlowLayout(FlowLayout.CENTER,30,20)); 37. top.add(title); 38. title.setFont(**new** Font("楷体",Font.BOLD,20)); 39. panel.add(mainpan,BorderLayout.CENTER); 40. mainpan.add(plateInput); 41. plateInput.setText("请输入车牌号"); 42. mainpan.add(ownerInput); 43. ownerInput.setText("请输入车主名"); 44. mainpan.add(modelInput); 45. modelInput.setText("请输入车型"); 46. mainpan.add(telInput); 47. telInput.setText("请输入车主电话"); 48. mainpan.add(colorInput); 49. colorInput.setText("请输入车辆颜色"); 50. mainpan.setLayout(**new** FlowLayout(FlowLayout.CENTER,60,30)); 51. panel.add(bottompan,BorderLayout.SOUTH); 52. bottompan.add(addBtn); 53. bottompan.add(cancelBtn); 54. setVisible(**true**); 56. plateInput.addFocusListener(**new** FocusListener() { 57. @Override 58. **public** **void** focusGained(FocusEvent focusEvent) { 59. **if**(plateInput.getText().equals("请输入车牌号")){ 60. plateInput.setText(""); 61. } 62. } 63. @Override 64. **public** **void** focusLost(FocusEvent focusEvent) { 65. **if**(VarifyInput.isEmpty(plateInput.getText())){ 66. plateInput.setText("请输入车牌号"); 67. } 68. } 69. }); 70. ownerInput.addFocusListener(**new** FocusListener() { 71. @Override 72. **public** **void** focusGained(FocusEvent focusEvent) { 73. **if**(ownerInput.getText().equals("请输入车主名")){ 74. ownerInput.setText(""); 75. } 76. } 77. @Override 78. **public** **void** focusLost(FocusEvent focusEvent) { 79. **if**(VarifyInput.isEmpty(ownerInput.getText())){ 80. ownerInput.setText("请输入车主名"); 81. } 82. } 83. }); 84. modelInput.addFocusListener(**new** FocusListener() { 85. @Override 86. **public** **void** focusGained(FocusEvent focusEvent) { 87. **if**(modelInput.getText().equals("请输入车型")){ 88. modelInput.setText(""); 89. } 90. } 91. @Override 92. **public** **void** focusLost(FocusEvent focusEvent) { 93. **if**(VarifyInput.isEmpty(modelInput.getText())){ 94. modelInput.setText("请输入车型"); 95. } 96. } 97. }); 98. telInput.addFocusListener(**new** FocusListener() { 99. @Override 100. **public** **void** focusGained(FocusEvent focusEvent) { 101. **if**(telInput.getText().equals("请输入车主电话")){ 102. telInput.setText(""); 103. } 104. } 105. @Override 106. **public** **void** focusLost(FocusEvent focusEvent) { 107. **if**(VarifyInput.isEmpty(telInput.getText())){ 108. telInput.setText("请输入车主电话"); 109. } 110. } 111. }); 112. colorInput.addFocusListener(**new** FocusListener() { 113. @Override 114. **public** **void** focusGained(FocusEvent focusEvent) { 115. **if**(colorInput.getText().equals("请输入车辆颜色")){ 116. colorInput.setText(""); 117. } 118. } 119. @Override 120. **public** **void** focusLost(FocusEvent focusEvent) { 121. **if**(VarifyInput.isEmpty(colorInput.getText())){ 122. colorInput.setText("请输入车辆颜色"); 123. } 124. } 125. }); 126. cancelBtn.addActionListener(actionEvent -> dispose()); 127. submitAddCarListener = **new** SubmitAddCarListener(); 128. submitAddCarListener.setAddCar(**this**); 129. addBtn.addActionListener(submitAddCarListener); 130. } 131. }   Login.java   1. **package** views; 3. **import** controller.UserDao; 4. **import** model.User; 5. **import** utils.DButil; 6. **import** utils.VarifyInput; 8. **import** javax.swing.\*; 9. **import** java.awt.\*; 10. **import** java.sql.Connection; 11. **import** java.sql.SQLException; 13. **public** **class** Login **extends** JFrame{ 14. //组件 15. JButton loginbutton = **new** JButton("登录"); 16. JButton restorebuton =**new** JButton("重置"); 17. **public** JTextField usernameContext = **new** JTextField(20); 18. **public** JPasswordField passwordContext = **new** JPasswordField(20); 19. JLabel labelTitle = **new** JLabel("车辆管理系统"); 20. JLabel labelUsername = **new** JLabel("username: "); 21. JLabel labelPassword = **new** JLabel("password: "); 23. //登录背景图片 24. ImageIcon bgcImg = **new** ImageIcon("asset/login.png"); 25. //布局容器 26. JPanel top = **new** JPanel(); 27. JPanel left = **new** JPanel(); 28. JPanel center = **new** JPanel(); 29. JPanel right = **new** JPanel(); 30. JPanel bottom = **new** JPanel(); 32. JPanel usernameInput = **new** JPanel(); 33. JPanel passwordInput = **new** JPanel(); 35. //用户登录验证类，数据库连接类 36. UserDao userDao = **new** UserDao(); 37. DButil dButil = **new** DButil();  40. **public** Login() **throws** HeadlessException { 41. setTitle("登录"); 42. setBounds(300,200,400,300); 43. setVisible(**true**); 44. setDefaultCloseOperation(EXIT\_ON\_CLOSE); 45. setLocationRelativeTo(**null**); 46. setResizable(**false**); 48. //创建并生成登录容器 49. JPanel loginPanel = **new** JPanel(); 50. loginPanel.setLayout(**new** BorderLayout()); 51. add(loginPanel); 53. //添加布局容器 54. loginPanel.add(top, BorderLayout.NORTH); 55. loginPanel.add(right,BorderLayout.WEST); 56. loginPanel.add(center,BorderLayout.CENTER); 57. loginPanel.add(left,BorderLayout.EAST); 58. loginPanel.add(bottom,BorderLayout.SOUTH); 60. // top容器 61. top.add(labelTitle); 62. labelTitle.setFont(**new** Font("楷体",Font.BOLD,20)); 63. top.setLayout(**new** FlowLayout(FlowLayout.CENTER,0,20)); 64. //right容器 66. //center容器 67. center.add(usernameInput); 68. center.add(passwordInput); 69. usernameInput.add(labelUsername); 70. passwordInput.add(labelPassword); 71. usernameInput.add(usernameContext); 72. passwordInput.add(passwordContext); 73. //right容器 75. //bottom容器 76. bottom.add(loginbutton); 77. bottom.add(restorebuton); 78. bottom.setLayout(**new** FlowLayout(FlowLayout.CENTER,10,40));  81. //重置功能 82. restorebuton.addActionListener(actionEvent -> { 83. usernameContext.setText(""); 84. passwordContext.setText(""); 85. JOptionPane.showMessageDialog(**null**,"已重置"); 86. });  89. //登录功能 90. loginbutton.addActionListener(actionEvent -> { 91. String username = usernameContext.getText(); 92. String password = **new** String(passwordContext.getPassword()); 93. **if**(VarifyInput.isEmpty(username)||VarifyInput.isEmpty(password)){ 94. JOptionPane.showMessageDialog(**null**,"用户名或密码为空!!","error",JOptionPane.ERROR\_MESSAGE); 95. **return**; 96. } 97. **try** { 98. //获取数据库连接 99. Connection conn = dButil.getCon(); 100. User user = **new** User(username,password); 101. User currentUser = userDao.login(conn,user); 102. **if**(currentUser==**null**){ 103. JOptionPane.showMessageDialog(**null**,"用户名或密码错误","警告",JOptionPane.WARNING\_MESSAGE); 104. **return**; 105. } 106. dispose(); 107. **new** MainTable(); 108. } **catch** (ClassNotFoundException | SQLException e) { 109. **throw** **new** RuntimeException(e); 110. } 111. }); 112. } 113. }   Maintable.java   1. **package** views; 3. **import** controller.CarDao; 4. **import** model.Cars; 5. **import** utils.DButil; 6. **import** views.Actions.ExitListener; 7. **import** views.Actions.RemoveCarListener; 9. **import** javax.swing.\*; 10. **import** javax.swing.table.DefaultTableModel; 11. **import** java.awt.\*; 12. **import** java.sql.Connection; 13. **import** java.sql.SQLException; 14. **import** java.util.List; 16. /\*\* 17. \* 显示主界面 18. \* 主界面组件： 19. \* 1. 按钮 汽车入库 显示新界面 20. \* 2. 按钮 汽车出库 直接请求删除该行 21. \* 3. 按钮 汽车查询 显示新界面 22. \* 4. 按钮 改写信息 显示新界面 23. \* 5. 按钮 刷新 重新请求数据repain表格 24. \* 6. 表格 显示数据库中所有数据 25. \* 26. \* 布局：BorderLayout 27. \* top 按钮 28. \* center 表格 29. \*/ 30. **public** **class** MainTable **extends** JFrame { 31. JButton addCar; //汽车入库 32. JButton subCar; //汽车出库 33. JButton queryCar; //查询汽车 34. JButton modifyCar; //修改汽车 35. JButton exit; //退出功能 36. Object[][] TableContent; 37. Object[]  TableColumn; 38. // 显示所有信息的表格 39. **public** JTable table; 40. //panel 41. JPanel top; 42. JPanel center; 43. RemoveCarListener removeCarListener; 44. ExitListener exitListener; 45. **public** **static** DefaultTableModel tableModel; //创建表格模型 47. **public** MainTable() **throws** HeadlessException, SQLException, ClassNotFoundException { 48. addCar = **new** JButton("汽车入库"); 49. subCar = **new** JButton("汽车出库"); 50. queryCar = **new** JButton("汽车查询"); 51. modifyCar = **new** JButton("修改信息"); 52. exit = **new** JButton("退出"); 53. top = **new** JPanel(); 54. center = **new** JPanel(); 56. setSize(1000,600); 57. setLocationRelativeTo(**null**); 58. setLayout(**new** BorderLayout()); 59. setBackground(Color.white); 60. setResizable(**false**); 61. add(top,BorderLayout.NORTH); 62. top.setPreferredSize(**new** Dimension(1000,60)); 63. top.setBackground(Color.gray); 64. top.add(addCar); 65. top.add(subCar); 66. top.add(queryCar); 67. top.add(modifyCar); 68. top.add(exit); 70. //删除汽车信息 71. removeCarListener = **new** RemoveCarListener(); 72. removeCarListener.setMainTable(**this**); 73. subCar.addActionListener(removeCarListener); 75. //刷新汽车信息 76. exitListener = **new** ExitListener(); 77. exitListener.setMainTable(**this**); 78. exit.addActionListener(exitListener); 80. //添加汽车信息 81. addCar.addActionListener(actionEvent -> **new** AddCar()); 83. //查询汽车信息 84. queryCar.addActionListener(actionEvent -> **new** QueryCar("查询汽车")); 86. //修改汽车信息 87. modifyCar.addActionListener(actionEvent -> **new** ModifyCar()); 88. //        addCar.addActionListener(); 89. //请求所有汽车数据 90. DButil dButil = **new** DButil(); 91. Connection connection = dButil.getCon(); 92. List<Cars> allCars =  **new** CarDao().getAllCars(connection); 93. dButil.closeConnection(connection); 94. TableContent = **new** Object[allCars.size()][5]; 95. TableColumn = **new** Object[]{"车牌", "车型", "车主", "电话", "颜色"}; 96. **for**(**int** i=0; i<allCars.size(); i++){ 97. TableContent[i][0] = allCars.get(i).getPlate(); 98. TableContent[i][1] = allCars.get(i).getModel(); 99. TableContent[i][2] = allCars.get(i).getOwner(); 100. TableContent[i][3] = allCars.get(i).getTel(); 101. TableContent[i][4] = allCars.get(i).getColor(); 102. } 103. add(center,BorderLayout.CENTER); 104. table = **new** JTable(TableContent,TableColumn); 105. table.setRowHeight(30); 106. //设置表格模型 107. tableModel = **new** DefaultTableModel(TableContent,TableColumn); 108. table.setModel(tableModel); 109. JScrollPane jsp = **new** JScrollPane(table); 110. center.add(jsp); 112. setVisible(**true**); 113. } 114. }   Modifycar.java   1. **package** views; 3. **import** controller.CarDao; 4. **import** model.Cars; 5. **import** utils.DButil; 6. **import** utils.VarifyInput; 7. **import** views.Actions.ModifyCarListener; 9. **import** javax.swing.\*; 10. **import** java.awt.\*; 11. **import** java.awt.event.ActionEvent; 12. **import** java.awt.event.ActionListener; 13. **import** java.awt.event.FocusEvent; 14. **import** java.awt.event.FocusListener; 15. **import** java.sql.Connection; 16. **import** java.sql.SQLException; 18. /\*\* 19. \* 修改汽车信息的界面 20. \*/ 21. **public** **class** ModifyCar **extends** JFrame { 22. JLabel title = **new** JLabel("修改汽车信息"); 23. JPanel top = **new** JPanel(); 24. JPanel plateBox = **new** JPanel(); 25. **public** JTextField plateInput = **new** JTextField(15); 26. JButton searchTarget = **new** JButton("输入车牌查找"); 27. JPanel modelBox = **new** JPanel(); 28. JLabel modelLabel = **new** JLabel("车型:"); 29. **public** JTextField modelInput = **new** JTextField(20); 30. JPanel ownerBox = **new** JPanel();; 31. JLabel ownerLabel = **new** JLabel("车主:"); 32. **public** JTextField ownerInput = **new** JTextField(20); 33. JPanel telBox = **new** JPanel();; 34. JLabel telLabel = **new** JLabel("电话:"); 35. **public** JTextField telInput = **new** JTextField(20); 36. JPanel colorBox = **new** JPanel();; 37. JLabel colorLabel = **new** JLabel("颜色:"); 38. **public** JTextField colorInput = **new** JTextField(20); 39. JPanel fun = **new** JPanel(); 40. JButton submitModify = **new** JButton("提交"); 41. JButton cancelModify = **new** JButton("取消"); 43. ModifyCarListener modifyCarListener; 45. **public** ModifyCar() **throws** HeadlessException { 46. setSize(300,300); 47. setLocationRelativeTo(**null**); 48. setLayout(**new** FlowLayout()); 49. setFont(**new** Font("微软雅黑",Font.PLAIN,20)); 50. setResizable(**false**); 51. top.add(title); 52. add(top); 53. add(plateBox); 54. plateBox.add(plateInput); 55. plateBox.add(searchTarget); 56. add(modelBox); 57. modelBox.add(modelLabel); 58. modelBox.add(modelInput); 59. add(ownerBox); 60. ownerBox.add(ownerLabel); 61. ownerBox.add(ownerInput); 62. add(telBox); 63. telBox.add(telLabel); 64. telBox.add(telInput); 65. add(colorBox); 66. colorBox.add(colorLabel); 67. colorBox.add(colorInput); 68. add(fun); 69. fun.add(submitModify); 70. fun.add(cancelModify); 72. plateInput.addFocusListener(**new** FocusListener() { 73. @Override 74. **public** **void** focusGained(FocusEvent focusEvent) { 75. **if**(plateInput.getText().equals("请输入车牌号")){ 76. plateInput.setText(""); 77. } 78. } 79. @Override 80. **public** **void** focusLost(FocusEvent focusEvent) { 81. **if**(VarifyInput.isEmpty(plateInput.getText())){ 82. plateInput.setText("请输入车牌号"); 83. } 84. } 85. }); 86. // 查询需要修改的信息 87. searchTarget.addActionListener(**new** ActionListener() { 88. @Override 89. **public** **void** actionPerformed(ActionEvent actionEvent) { 90. plateInput.setEditable(**false**);//点击查询后禁止更改输入框内容 91. DButil dButil = **new** DButil(); 92. **try** { 93. Connection connection = dButil.getCon(); 94. String plate = plateInput.getText(); 95. Cars targetCar = **new** CarDao().findModifyTarget(connection,plate); 96. **try**{ 97. modelInput.setText(targetCar.getModel()); 98. ownerInput.setText(targetCar.getOwner()); 99. telInput.setText(targetCar.getTel()); 100. colorInput.setText(targetCar.getColor()); 101. }**catch** (NullPointerException e){ 102. JOptionPane.showMessageDialog(**null**,"请输入正确的车牌信息","警告",JOptionPane.WARNING\_MESSAGE); 103. } 104. dButil.closeConnection(connection); 105. } **catch** (ClassNotFoundException | SQLException e) { 106. **throw** **new** RuntimeException(e); 107. } 108. } 109. }); 111. //提交修改 112. modifyCarListener = **new** ModifyCarListener(); 113. modifyCarListener.setModifyCar(**this**); 114. submitModify.addActionListener(modifyCarListener); 116. //放弃修改 117. cancelModify.addActionListener(actionEvent -> dispose()); 118. setVisible(**true**); 119. } 120. }   Querycar.java   1. **package** views; 3. **import** views.Actions.SearchCarListener; 5. **import** javax.swing.\*; 6. **import** javax.swing.table.DefaultTableModel; 8. **public** **class** QueryCar **extends** JFrame{ 9. JPanel panel = **new** JPanel();//面板 10. JLabel label = **new** JLabel("搜索条件："); 11. **public** JComboBox comboBox = **new** JComboBox();//创建下拉列表 12. **public** JTextField conditions = **new** JTextField(30); //条件字段 13. JButton searchBtn = **new** JButton("查询"); 14. JTable table = **new** JTable();   //显示查询结果 15. **public** DefaultTableModel tableModel; //查询结果表模型 16. JPanel tablePanel = **new** JPanel(); 18. **public** QueryCar(String title) { 19. setTitle(title); 20. setSize(500,300); 21. setLocationRelativeTo(**null**); 22. setResizable(**false**); 23. add(panel); 24. panel.add(label); 25. comboBox.addItem("--请选择查询条件--"); 26. comboBox.addItem("车牌"); 27. comboBox.addItem("车型"); 28. comboBox.addItem("车主"); 29. comboBox.addItem("电话"); 30. comboBox.addItem("颜色"); 31. panel.add(comboBox); 32. panel.add(conditions); 33. panel.add(searchBtn); 34. panel.add(tablePanel); 36. //点击查询将结果渲染到表格 37. SearchCarListener searchCarListener = **new** SearchCarListener(); 38. searchCarListener.setQueryCar(**this**); 39. searchBtn.addActionListener(searchCarListener); 41. tableModel = **new** DefaultTableModel(); 42. tableModel.addColumn("plate"); 43. tableModel.addColumn("model"); 44. tableModel.addColumn("owner"); 45. tableModel.addColumn("tel"); 46. tableModel.addColumn("color"); 47. table.setModel(tableModel); 48. JScrollPane jsp = **new** JScrollPane(table); 49. tablePanel.add(jsp); 50. setVisible(**true**); 51. } 52. }   /entry.java   1. **import** views.Login; 3. /\*\* 4. \* 程序执行入口 5. \*/ 6. **public** **class** Entry { 7. **public** **static** **void** main(String[] args) { 8. **new** Login(); 9. } 10. } | | |
| 解题结果及对结果的分析： | |
| 老师评语：    老师签名 年 月 日 | |

注：成绩分为优、良、中、及格、不及格五级