File - DatabaseTools.java

```
 1 import java.io.IOException;
 2 import java.io.InputStream;
 3 import java.sql.*;
 4 import java.util.ArrayList;
 5 import java.util.Properties;
 6
 7 public class DatabaseTools {
 8     public static Connection getConnected() {
 9         Connection conn = null;
10
11         try {
12             //Load MySQL database driver
13             Class.forName("com.mysql.jdbc.Driver");
14
15             // Connection string
16             String dbURL = getConnectionString();
17
18             // Credentials
19             String username = "root";
20             String password = "mysql";
21
22             // Set connection to database to
   Connection object
23             conn = DriverManager.getConnection(
   dbURL, username, password);
24         } catch (ClassNotFoundException e) {
25             e.printStackTrace();
26         } catch (SQLException e) {
27             e.printStackTrace();
28         }
29
30         return conn;
31     }
```

File - DatabaseTools.java

```
32
33      public static void insert(User user) {
34          // Connection to database;
35          Connection myConn = DatabaseTools.
    getConnected();
36
37          PreparedStatement ps = null;
38
39          // Insert query
40          String queryInsert = "INSERT INTO userdata
     (firstname, lastname, email) VALUES (?, ?, ?);";
41
42          try {
43              // Set PreparedStatement object to
    instance with query
44              ps = myConn.prepareStatement(
    queryInsert);
45
46              // Set values for parameters
47              ps.setString(1, user.getFirstName());
48              ps.setString(2, user.getLastName());
49              ps.setString(3, user.getEmail());
50
51              // Execute insert on database
52              ps.execute();
53          } catch (SQLException e) {
54              e.printStackTrace();
55          } finally {
56              DatabaseTools.closePreparedStatement(ps
    );
57              DatabaseTools.closeConnection(myConn);
58          }
59      }
```

File - DatabaseTools.java

```
60
61      public static ArrayList<User> selectAllUsers()
    {
62          // Connection to database
63          Connection conn = DatabaseTools.
    getConnected();
64
65          PreparedStatement ps = null;
66          ResultSet rs = null;
67          ArrayList<User> myUserList = new ArrayList<
    >();
68
69          // Select statement
70          String selectQuery = "SELECT userid,
    firstname, lastname, email FROM userdata;";
71
72          try {
73              // Create prepared statement object
74              ps = conn.prepareStatement(selectQuery)
    ;
75
76              // Execute query and return result set
77              rs = ps.executeQuery();
78
79              // Loop through all rows in result set
80              while (rs.next()) {
81                  // Create user object to add to
    list.
82                  User user = new User();
83
84                  // Add values for row of result set
     to properties of User object
85                  user.setId(rs.getInt("userid"));
```

File - DatabaseTools.java

```
 86                  user.setFirstName(rs.getString("
    firstname"));
 87                  user.setLastName(rs.getString("
    lastname"));
 88                  user.setEmail(rs.getString("email"
    ));
 89
 90                  // Add user to list
 91                  myUserList.add(user);
 92              }
 93          } catch (SQLException e) {
 94              e.printStackTrace();
 95          } finally {
 96              DatabaseTools.closePreparedStatement(
    ps);
 97              DatabaseTools.closeResultSet(rs);
 98              DatabaseTools.closeConnection(conn);
 99          }
100
101          return myUserList;
102      }
103
104      public static void update(User user) {
105          // Connection to database;
106          Connection myConn = DatabaseTools.
    getConnected();
107
108          PreparedStatement ps = null;
109
110          // Update query
111          String queryUpdate = "UPDATE userdata SET
    firstname = ?, lastname = ?, email = ? WHERE
    firstname = ? AND lastname = ?;";
```

File - DatabaseTools.java

```
112
113         try {
114             // Set PreparedStatement object to
    instance with query
115             ps = myConn.prepareStatement(
    queryUpdate);
116
117             // Set values for parameters
118             ps.setString(1, user.getFirstName());
119             ps.setString(2, user.getLastName());
120             ps.setString(3, user.getEmail());
121             ps.setString(4, user.getFirstName());
122             ps.setString(5, user.getLastName());
123
124             // Execute update on database
125             ps.executeUpdate();
126         } catch (SQLException e) {
127             e.printStackTrace();
128         } finally {
129             DatabaseTools.closePreparedStatement(
    ps);
130             DatabaseTools.closeConnection(myConn);
131         }
132     }
133
134     public static void delete(String firstName,
    String lastName) {
135         // Connection to database;
136         Connection myConn = DatabaseTools.
    getConnected();
137
138         PreparedStatement ps = null;
139
```

File - DatabaseTools.java

```
140         // Update query
141         String queryUpdate = "DELETE FROM userdata
     WHERE firstname = ? AND lastname = ?;";
142 
143         try {
144             // Set PreparedStatement object to
    instance with query
145             ps = myConn.prepareStatement(
    queryUpdate);
146 
147             // Set values for parameters
148             ps.setString(1, firstName);
149             ps.setString(2, lastName);
150 
151             // Execute update on database
152             ps.executeUpdate();
153         } catch (SQLException e) {
154             e.printStackTrace();
155         } finally {
156             DatabaseTools.closePreparedStatement(
    ps);
157             DatabaseTools.closeConnection(myConn);
158         }
159     }
160 
161     public static void closePreparedStatement(
    Statement ps) {
162         try {
163             if (ps != null) {
164                 ps.close();
165             }
166         } catch (SQLException e) {
167             e.printStackTrace();
```

File - DatabaseTools.java

```
168         }
169     }
170 
171     public static void closeConnection(Connection
    conn) {
172         try {
173             if (conn != null) {
174                 conn.close();
175             }
176         } catch (SQLException e) {
177             e.printStackTrace();
178         }
179     }
180 
181     public static void closeResultSet(ResultSet rs
    ) {
182         try {
183             if (rs != null) {
184                 rs.close();
185             }
186         } catch (SQLException e) {
187             e.printStackTrace();
188         }
189     }
190 
191     public static String getConnectionString() {
192         Properties properties = new Properties();
193         InputStream input = Thread.currentThread()
    .getContextClassLoader().getResourceAsStream("
    settings.properties");
194         try {
195             properties.load(input);
196         } catch (IOException e) {
```

File - DatabaseTools.java

```
197             e.printStackTrace();
198         }
199 
200         return properties.getProperty("dbURL").
    toString();
201     }
202 }
203 
```