

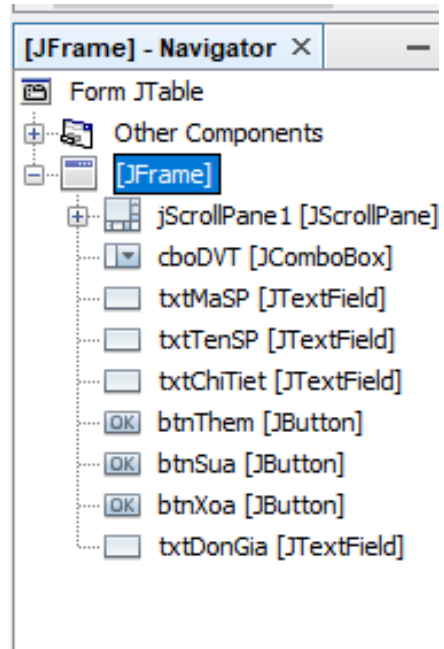
Connect Database

Gv HungNQ

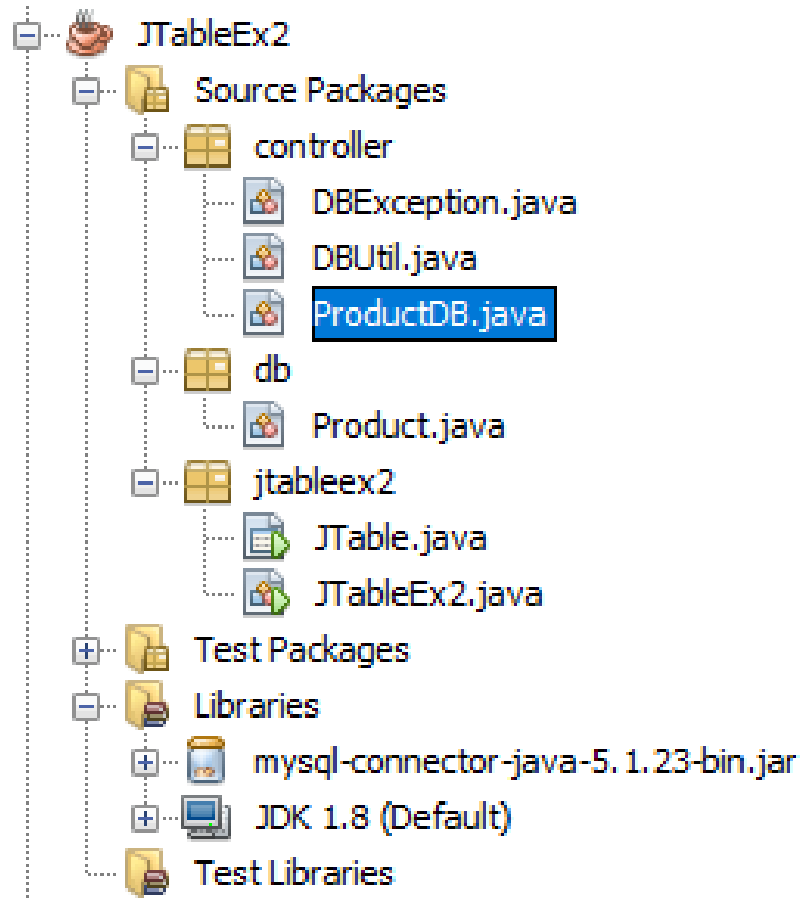
Create app with Interface

[illegible]

Navigator



Structure of App



Process with data on Form

- For Combobox:

```
String[] combo = {"Hop", "Kg", "Qua", "Chiec"};
```

```
for(String item: combo)
{
    cboDVT.addItem(item);
}
```

For Jtable (two array)

```
DefaultTableModel myModel;  
//header cua JTable  
String[] headers = new String[]{"Ma SP", "Ten SP", "Chi tiet", "Don gia"};  
String[][] datas = new String[][]{  
    {"SP01", "Xa phong", "120", "Hop"},  
    {"SP02", "Gao", "1300", "Kg"},  
    {"SP3", "Xoai", "10", "Qua"},  
    {"SP4", "But", "35", "Chiec"}  
};
```

For Jtable (two array)

```
String[] combo = {"Hop", "Kg", "Qua", "Chiec"};
public JTable() throws DBException {
    initComponents();
    myModel = new DefaultTableModel(datas, headers);
    // get Header the table
```

Bind data from JTable to JTextField

```
private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {  
    //lay ve dong  
    int row = jTable1.getSelectedRow();  
    //lay ve o  
    txtMaSP.setText(jTable1.getValueAt(row, 0).toString());  
    txtTenSP.setText(jTable1.getValueAt(row, 1).toString());  
    txtChiTiet.setText(jTable1.getValueAt(row, 2).toString());  
    txtDonGia.setText(jTable1.getValueAt(row, 3).toString());  
    //cboDVT.setSelectedItem(jTable1.getValueAt(row, 3));  
}
```


Add data to JTable

```
private void btnThemActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    Vector dataRow = new Vector();  
    dataRow.add(txtMaSP.getText().trim());  
    dataRow.add(txtTenSP.getText().trim());  
    dataRow.add(txtChiTiet.getText().trim());  
    dataRow.add(txtDonGia.getText().trim());  
    //dataRow.add(cboDVT.getSelectedItem());  
    //them vao JTable  
    myModel.addRow(dataRow);  
}
```

Update data on Jtable

```
private void btnSuaActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
  
    int row = jTable1.getSelectedRow();  
    if(row<0)  
    {  
        JOptionPane.showMessageDialog(null, "Chua chon dong");  
    }  
    //cap nhat tren JTable  
    myModel.setValueAt(txtTenSP.getText().trim(), row, 1);  
    myModel.setValueAt(txtChiTiet.getText().trim(), row, 2);  
    myModel.setValueAt(txtDonGia.getText().trim(), row, 3);  
    //myModel.setValueAt(cboDVT.getSelectedItem(), row, 3);  
    ...  
}
```

Delete data from JTable

```
private void btnXoaActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    int row = jTable1.getSelectedRow();  
    if(row<0)  
    {  
        JOptionPane.showMessageDialog(null, "Chua chon dong");  
    }  
  
    //xoa khoi JTable  
    myModel.removeRow(row);  
}
```

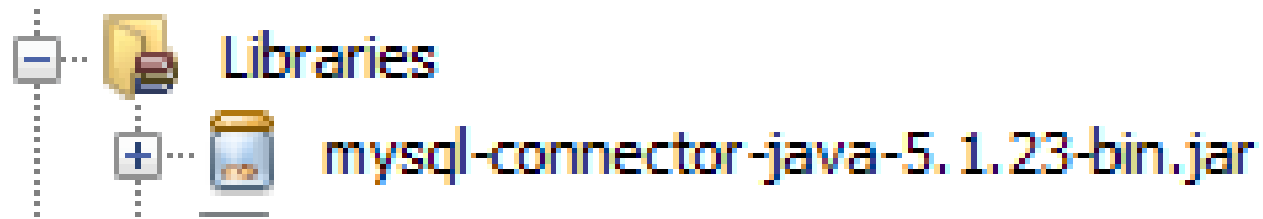
Working with Databasle

Gv HungNQ

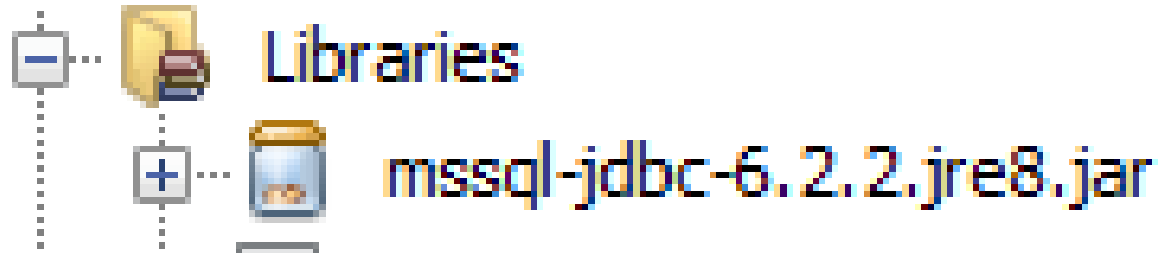
MySQL

-

```
// set the db url, username, and password  
String url = "jdbc:mysql://localhost:3306/mma";  
String username = "root";  
String password = "";
```



Sql Server

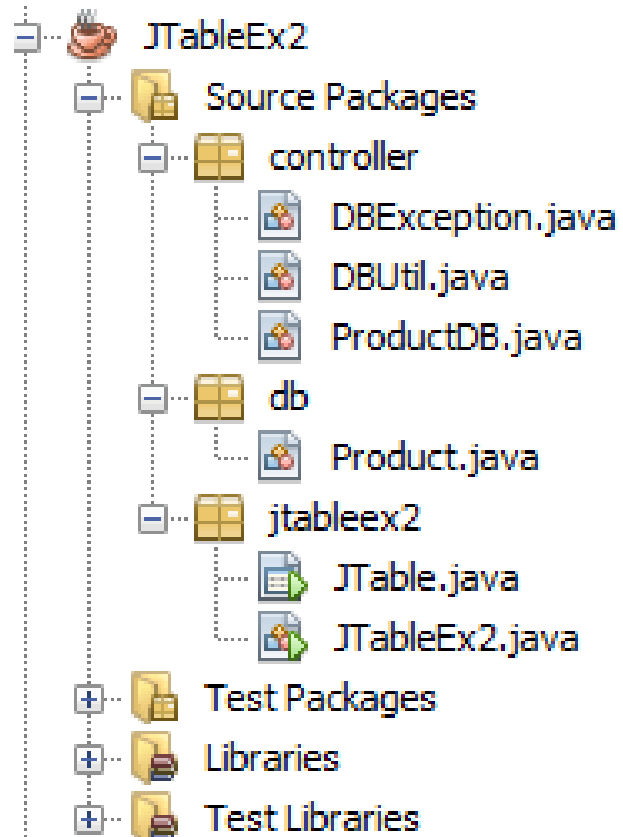


```
String url = "jdbc:sqlserver://MRHNQ\\SQLEXPRESS:1433;databaseName=Polyprow";
```

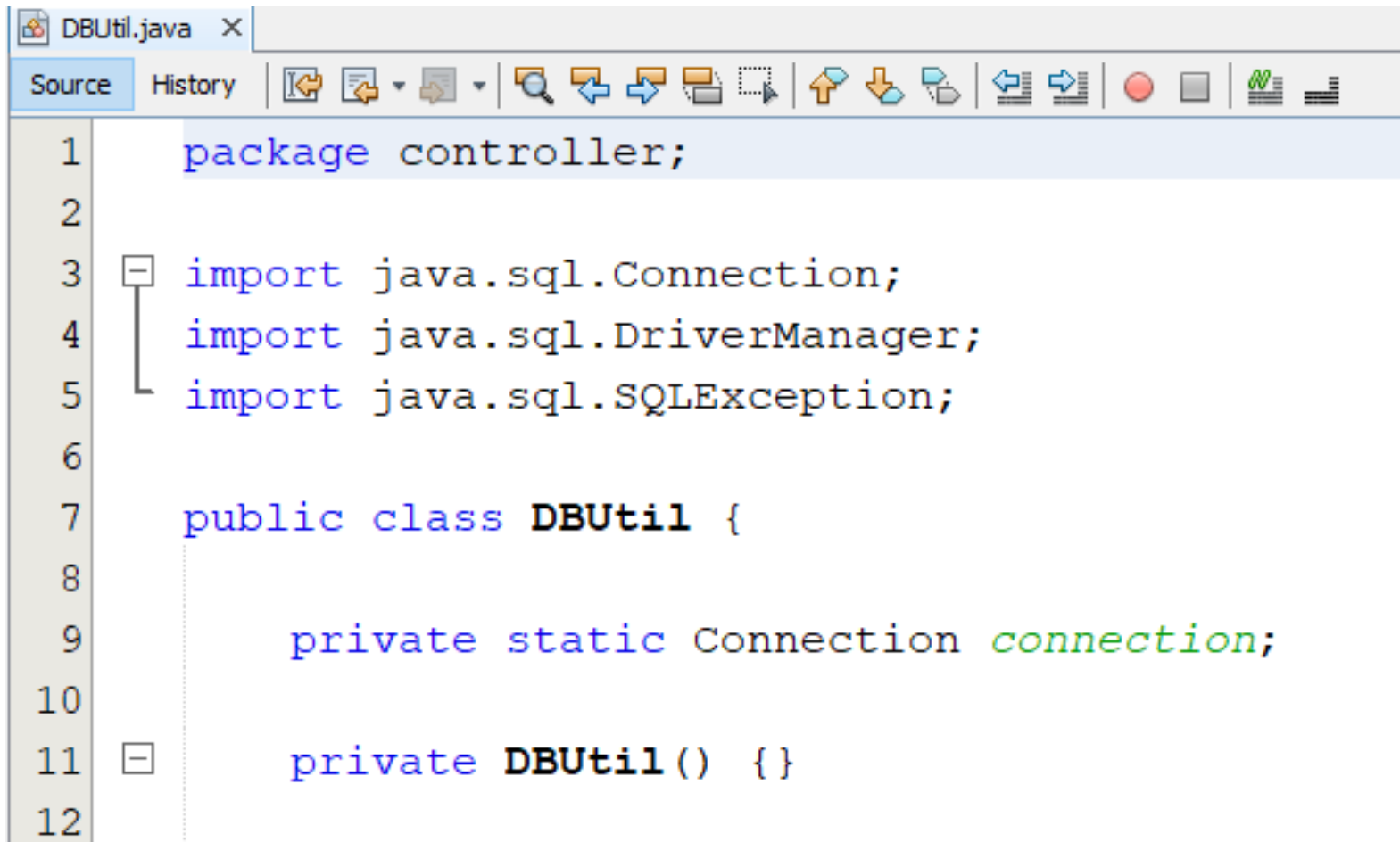
```
String username = "sa";
```

```
String password = "sa";
```

Structure of App

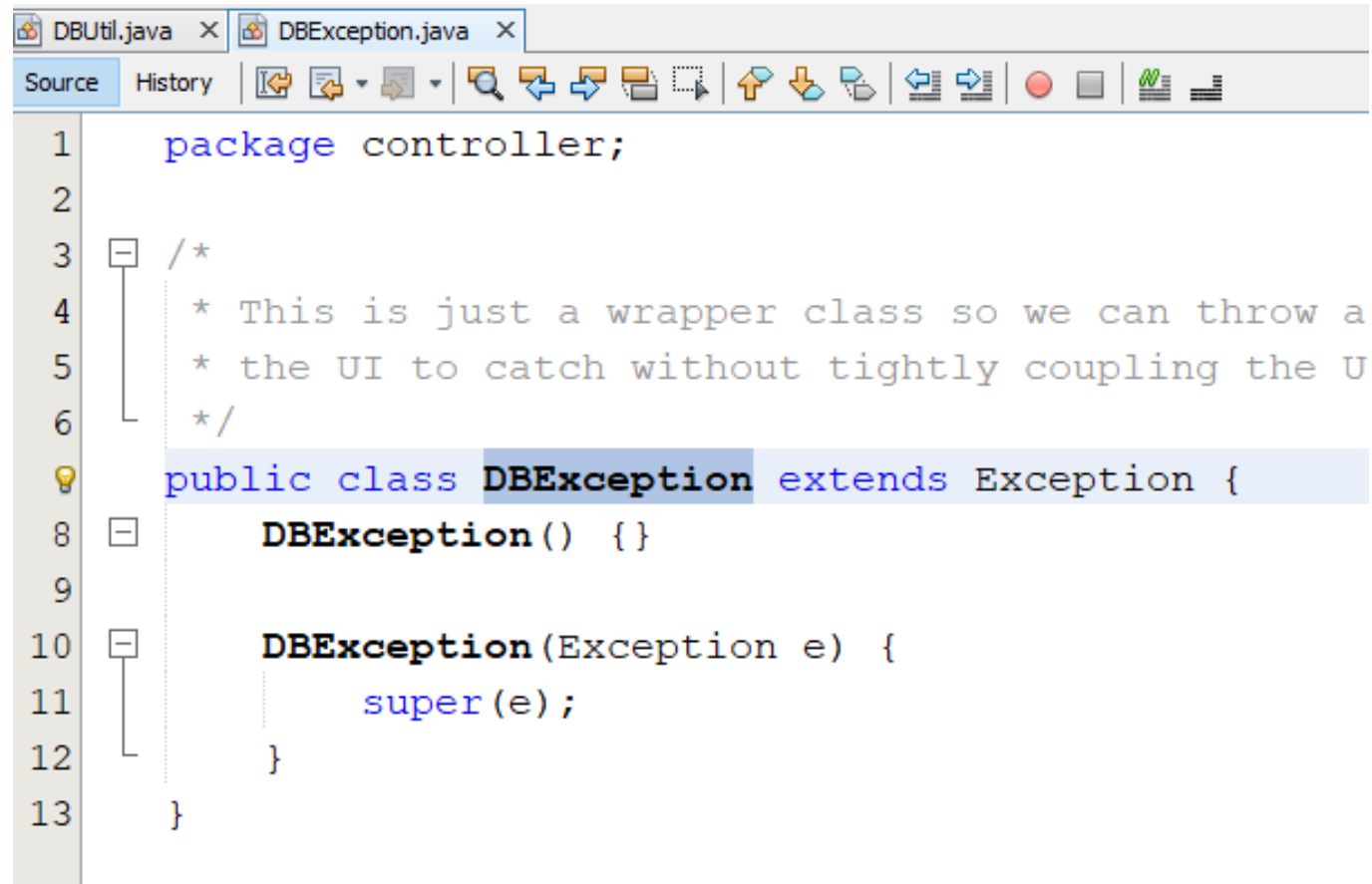


DBUtil.java (connect and disconnect)

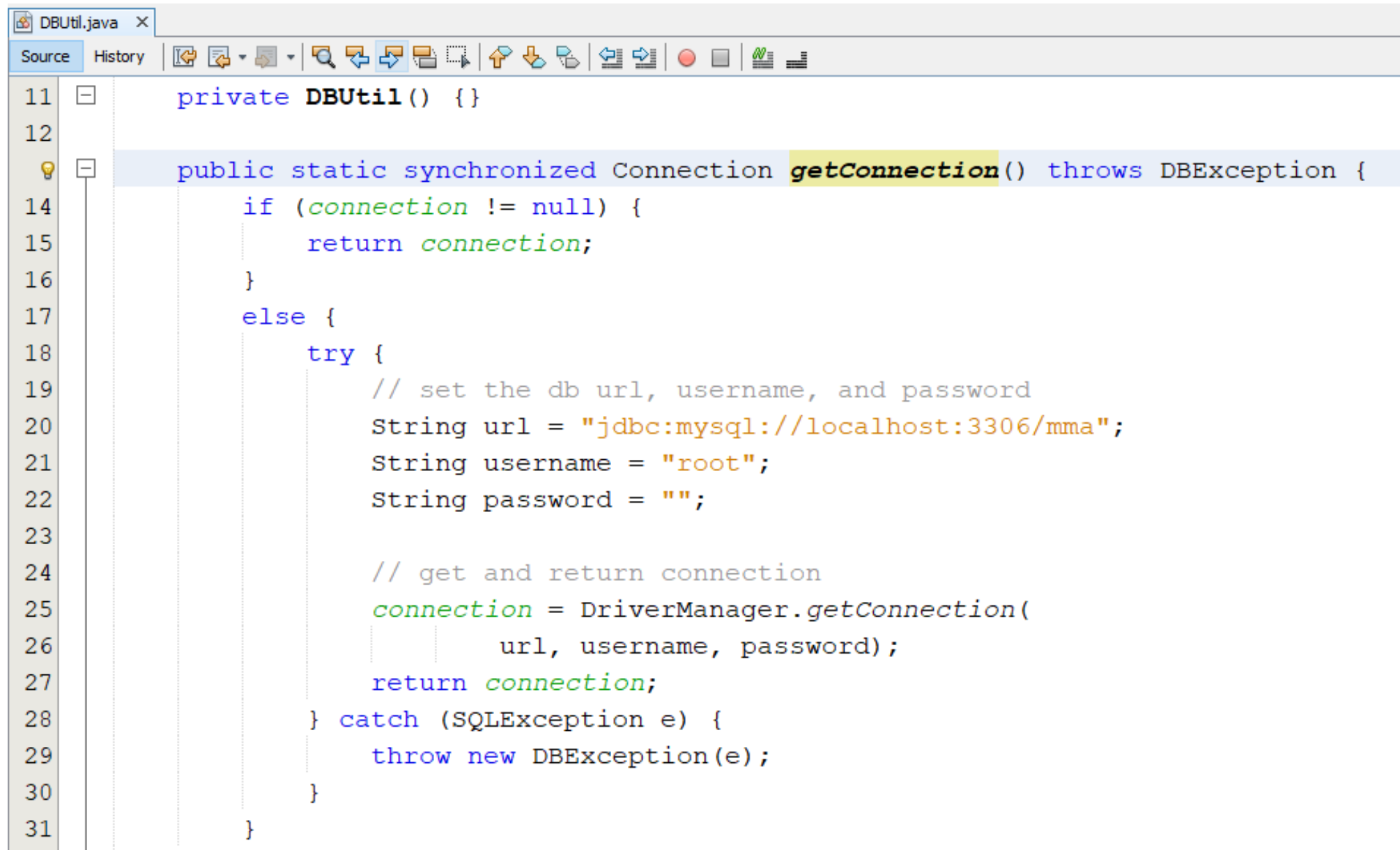


```
DBUtil.java x
Source History
1 package controller;
2
3 import java.sql.Connection;
4 import java.sql.DriverManager;
5 import java.sql.SQLException;
6
7 public class DBUtil {
8
9     private static Connection connection;
10
11     private DBUtil() {}
12
```



DBException.java



```
1  package controller;
2
3  /*
4   * This is just a wrapper class so we can throw a
5   * the UI to catch without tightly coupling the U
6   */
7  public class DBException extends Exception {
8      DBException() {}
9
10     DBException(Exception e) {
11         super(e);
12     }
13 }
```

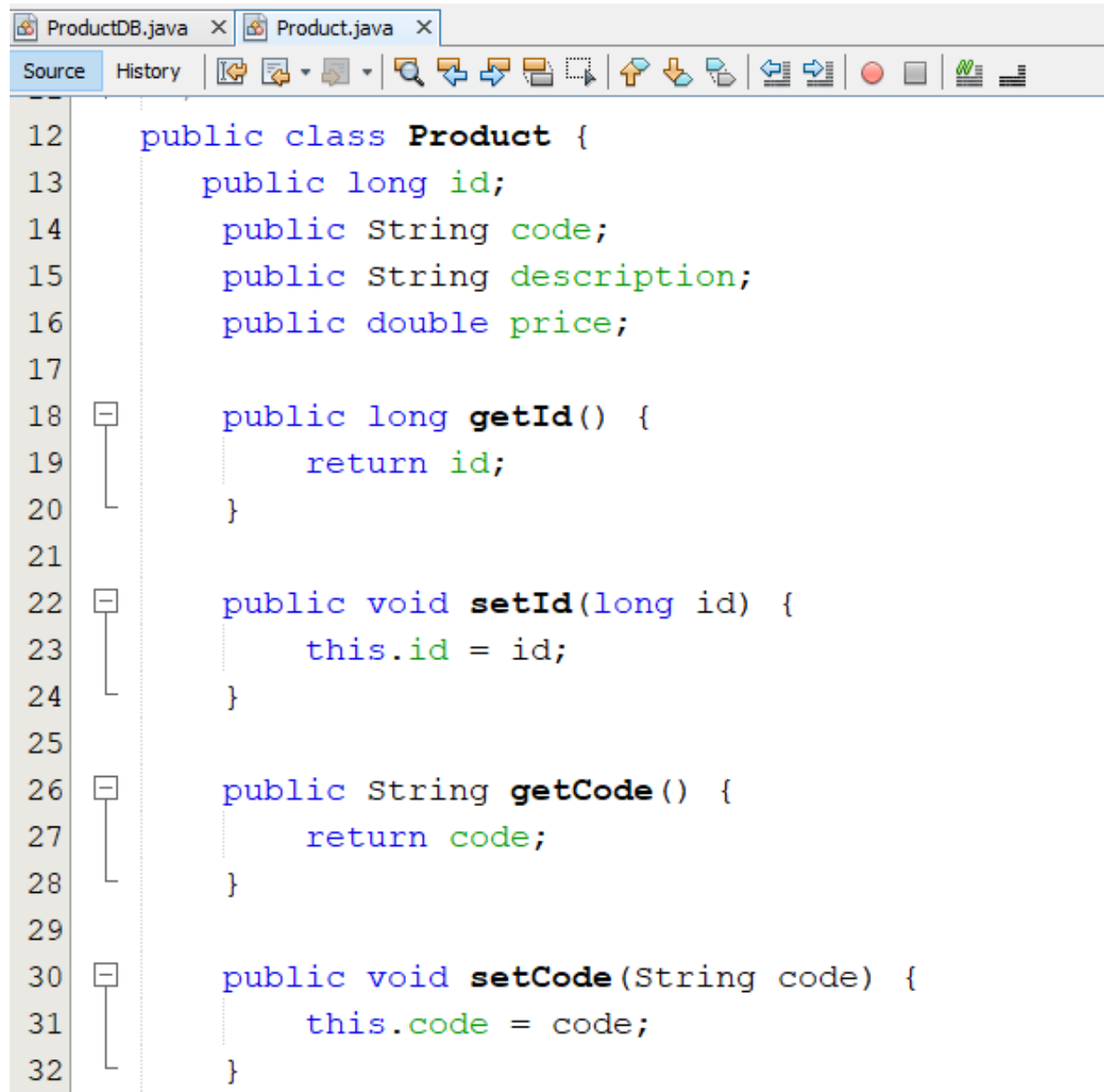


```
11 private DBUtil() {}
12
13 public static synchronized Connection getConnection() throws DBException {
14     if (connection != null) {
15         return connection;
16     }
17     else {
18         try {
19             // set the db url, username, and password
20             String url = "jdbc:mysql://localhost:3306/mma";
21             String username = "root";
22             String password = "";
23
24             // get and return connection
25             connection = DriverManager.getConnection(
26                 url, username, password);
27             return connection;
28         } catch (SQLException e) {
29             throw new DBException(e);
30         }
31     }
}
```



```
public static synchronized void closeConnection() throws DBException {  
    if (connection != null) {  
        try {  
            connection.close();  
        } catch (SQLException e) {  
            throw new DBException(e);  
        } finally {  
            connection = null;  
        }  
    }  
}
```

Product.java



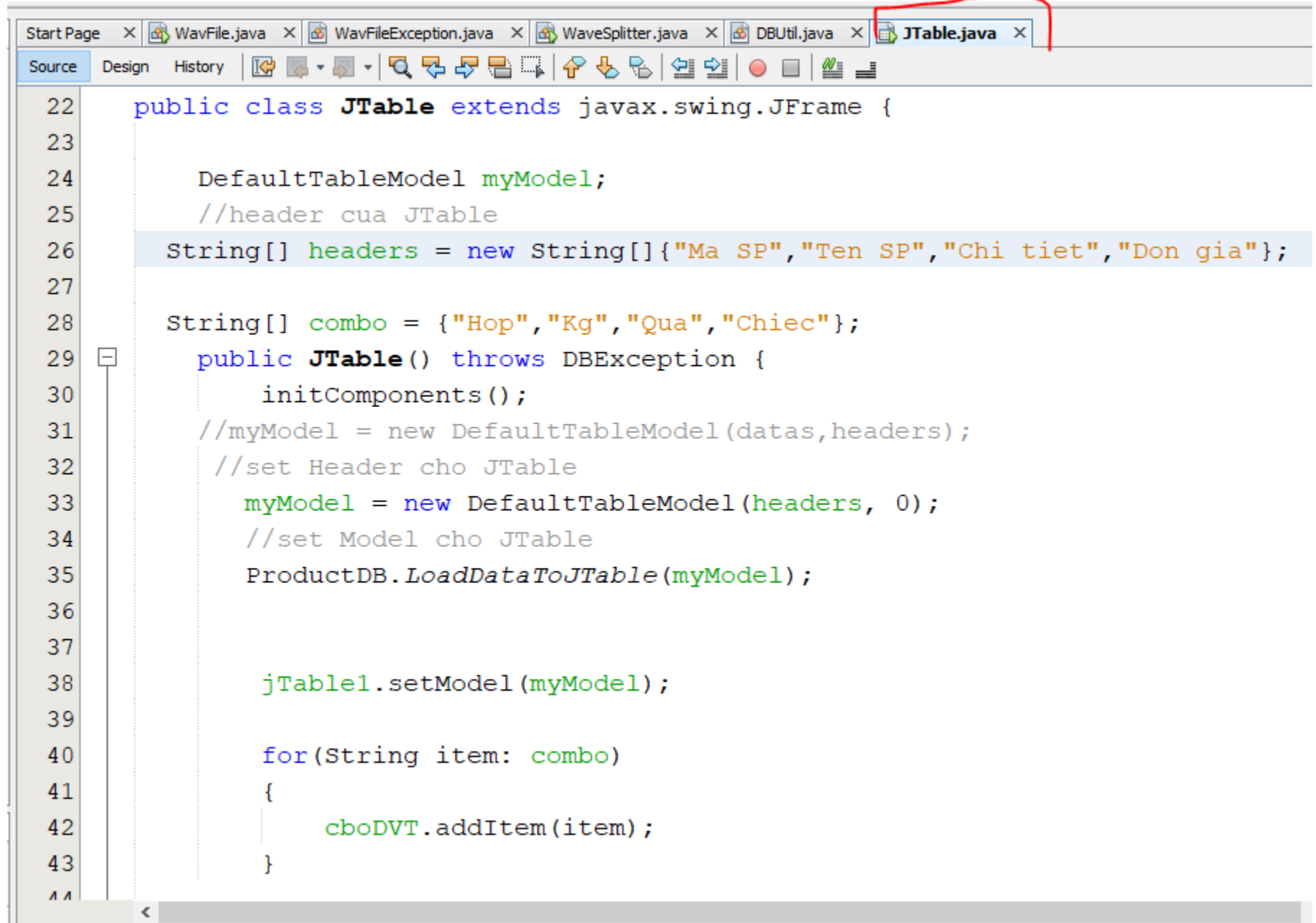
```
12 public class Product {
13     public long id;
14     public String code;
15     public String description;
16     public double price;
17
18     public long getId() {
19         return id;
20     }
21
22     public void setId(long id) {
23         this.id = id;
24     }
25
26     public String getCode() {
27         return code;
28     }
29
30     public void setCode(String code) {
31         this.code = code;
32     }
```

ProductDB

```
64 //ham lay du lieu tu CSDL
65 public static void LoadDataToJTable(DefaultTableModel myModel) throws DBException{
66 //khai bao chuoi truy van
67 String sql = "SELECT * FROM product";
68 //mo ket noi CSDL
69 Connection connection = DBUtil.getConnection();
70 //chuan bi doc du lieu
71 try (PreparedStatement ps = connection.prepareStatement(sql);
72      ResultSet rs = ps.executeQuery()) {
73     while (rs.next()) {
74         Vector row=new Vector();
75         int productID = rs.getInt("ProductID");
76         String code = rs.getString("Code");
77         String name = rs.getString("Description");
78         double price = rs.getDouble("ListPrice");
79         //gan vao doi tuong Product
80         Product p = new Product();
81         p.setId(productID);
82         p.setCode(code);
83         p.setDescription(name);
84         p.setPrice(price);
```

```
84         p.setPrice(price);
85         //dua vao vector
86         row.add(p.id );
87         row.add(p.code);
88         row.add(p.description);
89         row.add(p.price);
90         //dua bao model
91         myModel.addRow(row);
92
93     }
94
95     } catch (SQLException e) {
96         throw new DBException(e);
97     }
98 }
```

JTable



```
22 public class JTable extends javax.swing.JFrame {
23
24     DefaultTableModel myModel;
25     //header cua JTable
26     String[] headers = new String[]{"Ma SP", "Ten SP", "Chi tiet", "Don gia"};
27
28     String[] combo = {"Hop", "Kg", "Qua", "Chiec"};
29     public JTable() throws DBException {
30         initComponents();
31         //myModel = new DefaultTableModel(datas, headers);
32         //set Header cho JTable
33         myModel = new DefaultTableModel(headers, 0);
34         //set Model cho JTable
35         ProductDB.LoadDataToJTable(myModel);
36
37
38         jTable1.setModel(myModel);
39
40         for(String item: combo)
41         {
42             cboDVT.addItem(item);
43         }
44 }
```