Source Code

There is **no HTML**present in this code, as it focuses solely on server-side logic and database interactions within a Java application. The provided code is a Java class named **DB** that serves as a Data Access Object (DAO) for managing database operations related to a stock management system. It connects to a MySQL database using JDBC (Java Database Connectivity) and provides several methods for performing CRUD (Create, Read, Update, Delete) operations on products and cashiers. Key functionalities include adding, updating, deleting, and searching for products in the stock, as well as managing cashier information. The code handles database connections, executes SQL statements, and displays results or error messages using **JOptionPane** for user feedback.

The whole code is written in Java, and it includes the necessary imports for SQL connectivity, user interface dialog boxes, and Java utility classes.

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.text.DateFormat;

import java.text.SimpleDateFormat;

import java.util.ArrayList;

import java.util.Date;

import javax.swing.JOptionPane;

public class DB {

      public static Connection DBConnection()

      {

            Connection conn = null;

            try

            {

                  Class.forName("com.mysql.jdbc.Driver");

                  conn = DriverManager.getConnection("jdbc:mysql://localhost/caddey","root", "");

                  System.out.print("Database is connected !");

            }

            catch(Exception e)

            {

                  JOptionPane.showMessageDialog(null, "Do not connect to DB - Error:"+e);

            }

            return conn;

      }

      public static void addProductToDB(String id,String detail,String comp,int quan)

      {

            Connection conn=DBConnection();

            try {

                  Statement statement = conn.createStatement();

                  statement.executeUpdate("INSERT INTO stock VALUES ('"+id+"','"+detail+"','"+comp+"',"+quan+");");

                  JOptionPane.showMessageDialog(null, "Product added to database");

                  conn.close();

            } catch (SQLException e) {

                  // TODO Auto-generated catch block

                  JOptionPane.showMessageDialog(null, e.getMessage());

                  e.printStackTrace();

            }

      }

      public static void updateProductToDB(String id,String detail,String comp,int quan)

      {

            Connection conn=DBConnection();

            try {

                  Statement statement = conn.createStatement();

                  int status=statement.executeUpdate("UPDATE stock set Detail = '"+detail+"', Company = '"+comp+"', Quantity = "+quan+" WHERE ProductID = '"+id+"';");

                  if(status==1)

                  JOptionPane.showMessageDialog(null,  "Product updted");

                else

                  JOptionPane.showMessageDialog(null,  "ProductID not found!");

                  conn.close();

            } catch (SQLException e) {

                  // TODO Auto-generated catch block

                  JOptionPane.showMessageDialog(null, e.getMessage());

                  e.printStackTrace();

            }

      }

      public static void deleteProductToDB(String id)

      {

            Connection conn=DBConnection();

            try {

                  Statement statement = conn.createStatement();

                  int status=statement.executeUpdate("DELETE from stock WHERE ProductID = '"+id+"';");

                if(status==1)

                  JOptionPane.showMessageDialog(null,  "Product deleted");

                else

                  JOptionPane.showMessageDialog(null,  "ProductID not found!");

                  conn.close();

            } catch (SQLException e) {

                  // TODO Auto-generated catch block

                  JOptionPane.showMessageDialog(null, e.getMessage());

                  e.printStackTrace();

            }

      }

      public static void searchProduct(String id)

      {

            Connection conn=DBConnection();

            try {

                  Statement statement = conn.createStatement();

                  ResultSet rs = statement.executeQuery("Select \* from stock WHERE ProductID = '"+id+"';");

                  if (!rs.next())

                        JOptionPane.showMessageDialog(null,"No product found with this id!");

                  else

                        JOptionPane.showMessageDialog(null, "ProductID: "+id+"\nQuantity: "+rs.getString("Quantity"));

                  conn.close();

            } catch (SQLException e) {

                  // TODO Auto-generated catch block

                  JOptionPane.showMessageDialog(null, e.getMessage());

                  e.printStackTrace();

            }

      }

      public static void searchCashier(String email)

      {

            Connection conn=DBConnection();

            try {

                  Statement statement = conn.createStatement();

                  ResultSet rs = statement.executeQuery("Select \* from users WHERE Email = '"+email+"';");

                  if (!rs.next())

                        JOptionPane.showMessageDialog(null,"No cashier found with this email!");

                  else

                        JOptionPane.showMessageDialog(null, "Email: "+email+"\nPassword: "+rs.getString("Password"));

                  conn.close();

            } catch (SQLException e) {

                  // TODO Auto-generated catch block

                  JOptionPane.showMessageDialog(null, e.getMessage());

                  e.printStackTrace();

            }

      }

      public static boolean varifyLogin(String email,String pass)

      {

            boolean login=false;

            Connection conn=DBConnection();

            try {

                  Statement statement = conn.createStatement();

                  ResultSet rs = statement.executeQuery("Select \* from users WHERE Email = '"+email+"' and Password = '"+pass+"';");

                  if (!rs.next())

                        login=false;

                  else

                        login=true;

                  conn.close();

            } catch (SQLException e) {

                  // TODO Auto-generated catch block

                  JOptionPane.showMessageDialog(null, e.getMessage());

                  e.printStackTrace();

            }

            return login;

      }

      public static void addCashier(String user,String pass)

      {

            Connection conn=DBConnection();

            try {

                  Statement statement = conn.createStatement();

                  statement.executeUpdate("INSERT INTO users VALUES ('"+user+"','"+pass+"');");

                  JOptionPane.showMessageDialog(null, "Cashier added to database");

                  conn.close();

            } catch (SQLException e) {

                  // TODO Auto-generated catch block

                  JOptionPane.showMessageDialog(null, e.getMessage());

                  e.printStackTrace();

            }

      }

      public static void deleteCashier(String user,String pass)

      {

            Connection conn=DBConnection();

            try {

                  Statement statement = conn.createStatement();

                  int status=statement.executeUpdate("DELETE from users WHERE Email = '"+user+"' AND Password = '"+pass+"';");

                   if(status==1)

                        JOptionPane.showMessageDialog(null,  "Cashier deleted");

                      else

                        JOptionPane.showMessageDialog(null,  "Cashier not found!");

                  conn.close();

            } catch (SQLException e) {

                  // TODO Auto-generated catch block

                  JOptionPane.showMessageDialog(null, e.getMessage());

                  e.printStackTrace();

            }

      }

      public static String searchPDetail(String id,int q)

      {

            Connection conn=DBConnection();

            String rt="";

            try {

                  int quan;

                  Statement statement = conn.createStatement();

                  ResultSet rs = statement.executeQuery("Select \* from stock WHERE ProductID = '"+id+"';");

                  if (!rs.next())

                        rt="nill";

                  else{

                        quan=Integer.parseInt(rs.getString("Quantity"))-q;

                        if(quan<0)

                              rt="item is out of stock";

                        else

                        {

                              rt=rs.getString("Detail")+"%"+rs.getString("Company");

                              statement.executeUpdate("UPDATE stock set Quantity = "+quan+" WHERE ProductID = '"+id+"';");

                        }

                  }

                  conn.close();

            } catch (SQLException e) {

                  // TODO Auto-generated catch block

                  JOptionPane.showMessageDialog(null, e.getMessage());

                  e.printStackTrace();

            }

            return rt;

      }

      public static void addSaleToDB(Object data[],ArrayList<String> comp,String name)

      {

            Connection conn=DBConnection();

            DateFormat dateFormat = new SimpleDateFormat("yyyy/MM/dd");

            Date date = new Date();

            String d=dateFormat.format(date);

            try {

                  Statement statement = conn.createStatement();

                  for(int x=0;x<data.length;x=x+5)

                  {

                        statement.executeUpdate("INSERT INTO sale VALUES ('"+data[x]+"','"+comp.get(0)+"','"+d+"','"+data[x+3]+"',"+data[x+4]+",'"+name+"');");

                        comp.remove(0);

                  }

                  conn.close();

            } catch (SQLException e) {

                  // TODO Auto-generated catch block

                  JOptionPane.showMessageDialog(null, e.getMessage());

                  e.printStackTrace();

            }

      }

      public static ArrayList<String> getSale(String date,String comp)

      {

            String q;

            ArrayList<String> r=new ArrayList<String>();

            if(comp.equals("All"))

                  q="Select \* from sale WHERE Date = '"+date+"';";

            else

                  q="Select \* from sale WHERE Date = '"+date+"' AND Company = '"+comp+"';";

            Connection conn=DBConnection();

            try {

                  Statement statement = conn.createStatement();

                  ResultSet rs = statement.executeQuery(q);

                  while(rs.next())

                  {

                        r.add(rs.getString("Date"));

                        r.add(rs.getString("ProductID"));

                        r.add(rs.getString("Company"));

                        r.add(rs.getString("Payment"));

                  }

                  conn.close();

            } catch (SQLException e) {

                  // TODO Auto-generated catch block

                  JOptionPane.showMessageDialog(null, e.getMessage());

                  e.printStackTrace();

            }

            return r;

      }

      public static ArrayList<String> showStock(String comp)

      {

            String q;

            ArrayList<String> r=new ArrayList<String>();

            if(comp.equals("All"))

                  q="Select \* from stock;";

            else

                  q="Select \* from stock WHERE Company = '"+comp+"';";

            Connection conn=DBConnection();

            try {

                  Statement statement = conn.createStatement();

                  ResultSet rs = statement.executeQuery(q);

                  while(rs.next())

                  {

                        r.add(rs.getString("ProductID"));

                        r.add(rs.getString("Detail"));

                        r.add(rs.getString("Company"));

                        r.add(rs.getString("Quantity"));

                  }

                  conn.close();

            } catch (SQLException e) {

                  // TODO Auto-generated catch block

                  JOptionPane.showMessageDialog(null, e.getMessage());

                  e.printStackTrace();

            }

            return r;

      }

      public static String getPDetail(String id,int q)

      {

            Connection conn=DBConnection();

            String rt="";

            try {

                  int quan;

                  Statement statement = conn.createStatement();

                  ResultSet rs = statement.executeQuery("Select \* from stock WHERE ProductID = '"+id+"';");

                  if (!rs.next())

                        rt="nill";

                  else{

                        quan=Integer.parseInt(rs.getString("Quantity"))-q;

                        if(quan<0)

                              rt="item is out of stock";

                        else

                        {

                              rt=rs.getString("Detail")+"%"+rs.getString("Company");

                              statement.executeUpdate("UPDATE stock set Quantity = "+quan+" WHERE ProductID = '"+id+"';");

                        }

                  }

                  conn.close();

            } catch (SQLException e) {

                  // TODO Auto-generated catch block

                  JOptionPane.showMessageDialog(null, e.getMessage());

                  e.printStackTrace();

            }

            return rt;

      }

      public static ArrayList<String> searchP(String id)

      {

            Connection conn=DBConnection();

            ArrayList<String> data=new ArrayList<String>();

            try {

                  Statement statement = conn.createStatement();

                  ResultSet rs = statement.executeQuery("Select \* from stock WHERE ProductID = '"+id+"';");

                  if (rs.next())

                  {

                        data.add(rs.getString("Detail"));

                        data.add(rs.getString("Company"));

                        data.add(rs.getString("Quantity"));

                  }

                  conn.close();

            } catch (SQLException e) {

                  // TODO Auto-generated catch block

                  JOptionPane.showMessageDialog(null, e.getMessage());

                  e.printStackTrace();

            }

            return data;

      }

      public static void updateProduct(String id,int quan)

      {

            Connection conn=DBConnection();

            try {

                  Statement statement = conn.createStatement();

                  ResultSet rs = statement.executeQuery("Select \* from stock WHERE ProductID = '"+id+"';");

                  int q=0;

                  if(rs.next())

                  {

                        q=Integer.parseInt(rs.getString("Quantity"))+quan;

                        statement.executeUpdate("UPDATE stock set Quantity = "+q+" WHERE ProductID = '"+id+"';");

                  }

                  conn.close();

            } catch (SQLException e) {

                  // TODO Auto-generated catch block

                  JOptionPane.showMessageDialog(null, e.getMessage());

                  e.printStackTrace();

            }

      }

      public static void main(String args[])

      {

      }

}

**Deepak Oruganti**

VP IT Audit

Corporate Assurance & Advisory Services (CAAS)

Teams: 856-765-4809 | Mobile: 609-915-0045

[deepak.oruganti@fiserv.com](mailto:deepak.oruganti@fiserv.com)

**Fiserv**  
**Helping Small Businesses Get**[**Back2Business**](https://www.fiserv.com/en/lp/back2business.html?cid=link|fiservsignature|b2b||)  
[Fiserv](https://www.fiserv.com/?cid=link|fiservsignature|fiservlink||) | [Join Our Team](https://www.careers.fiserv.com/?utm_source=outlook&utm_medium=email&utm_campaign=signature) | [Twitter](https://twitter.com/fiserv) | [LinkedIn](https://www.linkedin.com/company/fiserv/) | [Facebook](https://www.facebook.com/Fiserv/)  
  
© 2021 Fiserv Inc. or its affiliates. Fiserv is a registered trademark of Fiserv Inc. [Privacy Notice](https://www.fiserv.com/en/about-fiserv/privacy-notice.html?cid=link|fiservsignature|privacypolicy||)  
© 2021 Fortune Media IP Limited. Used under license.

**From:** Deepak Oruganti <fy.25.fiserv.dlp.ia.deepak52025@gmail.com>  
**Sent:** Wednesday, June 4, 2025 6:16 PM  
**To:** Oruganti, Deepak (Berkeley Heights) <deepak.oruganti@fiserv.com>  
**Subject:** Fwd: For Review

|  |
| --- |
| ⚠ EXTERNAL MESSAGE – Think Before You Click |

For audit purposes.   
  
---------- Forwarded message ----------  
From: **Oruganti, Deepak (Berkeley Heights)** <[deepak.oruganti@fiserv.com](mailto:deepak.oruganti@fiserv.com)>  
Date: Wednesday, June 4, 2025  
Subject: For Review  
To: "[fy25.fiserv.dlp.ia.deepak52025@gmail.com](mailto:fy25.fiserv.dlp.ia.deepak52025@gmail.com)" <[fy25.fiserv.dlp.ia.deepak52025@gmail.com](mailto:fy25.fiserv.dlp.ia.deepak52025@gmail.com)>

The entire code sample is written in Java with **no HTML**components included. The focus is on creating a graphical user interface for an administrative management system.

The provided code is primarily Java and constitutes a Swing application for an Admin Panel. It includes various GUI components such as menu items for managing products, cashiers, stock, and sales, implementing an event-driven model where actions performed by the user trigger corresponding methods.

The structure includes imports for necessary Java packages such as **javax.swing** for building the graphical interface and **java.awt** for layout management. It defines the **AdminPanel** class, which extends **JFrame** and implements **ActionListener**, allowing it to respond to user interactions through menu selections. Overall, the code effectively organizes the functionality of the application through a combination of user interface components and event handling.

----

import java.awt.BorderLayout;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.util.ArrayList;

import javax.swing.JFrame;

import javax.swing.JMenuBar;

import javax.swing.JMenuItem;

import javax.swing.JPanel;

import javax.swing.JMenu;

import java.awt.Toolkit;

public class AdminPanel extends JFrame implements ActionListener {

      JMenuItem itmAddProduct;

      JMenu mnProduct;

      JMenuItem itmUpdateProduct;

      JMenuItem itmDeleteProduct;

      JMenu mnCashier ;

      JMenuItem itmDeleteCashier;

      JMenuItem itmAddCashier;

      JMenu mnStock ;

      JMenuItem itmShowStock;

      JMenu mnExport;

      ArrayList<JPanel> panels=new ArrayList<JPanel>();

      int cPanel=0;

      private JMenu mnSearch;

      private JMenuItem mntmSearchProduct;

      private JMenuItem mntmSearchCashier;

      private JMenu mnSale;

      private JMenuItem mntmPrintSale;

      /\*\*

       \* Create the frame.

       \*/

      public AdminPanel() {

            setIconImage(Toolkit.getDefaultToolkit().getImage("F:\\Working Directory\\fianl project with sql\\Bill\\logo.png"));

            setTitle("Admin Panel");

            setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

            setBounds(100, 100, 840, 619);

            JMenuBar menuBar = new JMenuBar();

            setJMenuBar(menuBar);

            mnProduct = new JMenu("Product");

            menuBar.add(mnProduct);

            itmAddProduct = new JMenuItem("Add Product");

            mnProduct.add(itmAddProduct);

            itmAddProduct.addActionListener(this);

            itmUpdateProduct = new JMenuItem("Update Product");

            mnProduct.add(itmUpdateProduct);

            itmUpdateProduct.addActionListener(this);

            itmDeleteProduct = new JMenuItem("Delete Product");

            mnProduct.add(itmDeleteProduct);

            itmDeleteProduct.addActionListener(this);

            mnCashier = new JMenu("Cashier");

            menuBar.add(mnCashier);

            itmAddCashier = new JMenuItem("Add Cashier");

            mnCashier.add(itmAddCashier);

            itmAddCashier.addActionListener(this);

            itmDeleteCashier = new JMenuItem("Delete Cashier");

            mnCashier.add(itmDeleteCashier);

            itmDeleteCashier.addActionListener(this);

            mnStock = new JMenu("Stock");

            menuBar.add(mnStock);

            itmShowStock = new JMenuItem("Show Stock");

            mnStock.add(itmShowStock);

            itmShowStock.addActionListener(this);

            mnSearch = new JMenu("Search");

            menuBar.add(mnSearch);

            mntmSearchProduct = new JMenuItem("Search Product");

            mnSearch.add(mntmSearchProduct);

            mntmSearchProduct.addActionListener(this);

            mntmSearchCashier = new JMenuItem("Search Cashier");

            mnSearch.add(mntmSearchCashier);

            mnSale = new JMenu("Sale");

            menuBar.add(mnSale);

            mntmPrintSale = new JMenuItem("Print Sale");

            mnSale.add(mntmPrintSale);

            mntmPrintSale.addActionListener(this);

            mnExport = new JMenu("Account");

            menuBar.add(mnExport);

            JMenuItem logout = new JMenuItem("Logout");

            mnExport.add(logout);

            logout.addActionListener(this);

            mntmSearchCashier.addActionListener(this);

            getContentPane().setLayout(new BorderLayout(0, 0));

            panels.add(new addProduct());

            panels.add(new updateProduct());

            panels.add(new deleteProduct());

            panels.add(new addCashier());

            panels.add(new deleteCashier());

            panels.add(new showStock());

            panels.add(new searchProduct());

            panels.add(new searchCashier());

            panels.add(new Sale());

            getContentPane().add(panels.get(0));

      }

      @Override

      public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            System.out.println("Selected: " + e.getActionCommand());

            if(e.getActionCommand().equals("Add Product"))

            {

                  System.out.println(panels.get(cPanel));

                  this.remove(panels.get(cPanel));

                  this.revalidate();

                  this.repaint();

                  getContentPane().add(panels.get(0));

                  this.setVisible(true);

                  cPanel=0;

                  this.setTitle("Add Product");

            }

            else if(e.getActionCommand().equals("Update Product"))

            {

                  this.remove(panels.get(cPanel));

                  this.revalidate();

                  this.repaint();

                  getContentPane().add(panels.get(1));

                  this.setVisible(true);

                  cPanel=1;

                  this.setTitle("Update Product");

            }

            else if(e.getActionCommand().equals("Delete Product"))

            {

                  this.remove(panels.get(cPanel));

                  this.revalidate();

                  this.repaint();

                  getContentPane().add(panels.get(2));

                  this.setVisible(true);

                  cPanel=2;

                  this.setTitle("Delete Product");

            }

            else if(e.getActionCommand().equals("Add Cashier"))

            {

                  this.remove(panels.get(cPanel));

                  this.revalidate();

                  this.repaint();

                  getContentPane().add(panels.get(3));

                  this.setVisible(true);

                  cPanel=3;

                  this.setTitle("Add Cashier");

            }

            else if(e.getActionCommand().equals("Delete Cashier"))

            {

                  this.remove(panels.get(cPanel));

                  this.revalidate();

                  this.repaint();

                  getContentPane().add(panels.get(4));

                  this.setVisible(true);

                  cPanel=4;

                  this.setTitle("Delete Cashier");

            }

            else if(e.getActionCommand().equals("Show Stock"))

            {

                  this.remove(panels.get(cPanel));

                  getContentPane().add(panels.get(5));

                  this.revalidate();

                  this.repaint();

                  this.setVisible(true);

                  cPanel=5;

                  this.setTitle("Show Stock");

            }

            else if(e.getActionCommand().equals("Search Product"))

            {

                  this.remove(panels.get(cPanel));

                  getContentPane().add(panels.get(6));

                  this.revalidate();

                  this.repaint();

                  this.setVisible(true);

                  cPanel=6;

                  this.setTitle("Search Product");

            }

            else if(e.getActionCommand().equals("Search Cashier"))

            {

                  this.remove(panels.get(cPanel));

                  getContentPane().add(panels.get(7));

                  this.revalidate();

                  this.repaint();

                  this.setVisible(true);

                  cPanel=7;

                  this.setTitle("Search Cashier");

            }

            else if(e.getActionCommand().equals("Print Sale"))

            {

                  this.remove(panels.get(cPanel));

                  getContentPane().add(panels.get(8));

                  this.revalidate();

                  this.repaint();

                  this.setVisible(true);

                  cPanel=8;

                  this.setTitle("Print Sale");

            }

            else if(e.getActionCommand().equals("Logout"))

            {

                  this.dispose();

            }

      }

}