**JAVA SWING BASED**

**STOCK MARKET PREDICTION SYSTEM**

**SQL CONNECTIVITY USING JDBC**

*A Report*

*Submitted in partial fulfillment of the Requirements for the COURSE*

**DATABASE MANAGEMENT SYSTEMS**

**By**

**ANURAG DVS <1602-21-737-064>**

**Under the guidance of Ms B. Leelavathy**

****

**Department of Information Technology Vasavi College of Engineering (Autonomous) (Affiliated to Osmania University) Ibrahimbagh, Hyderabad-31 2022-2023**

**BONAFIDE CERTIFICATE**

This is to certify that this project report titled

***‘STOCK MARKET PREDICTION’***

is a project work of **ANURAG DVS** bearing roll no. 1602-21-737-0xx who carried out this project under my supervision in the IV semester for

the academic year 2022- 2023

Signature Signature

External Examiner Internal Examiner

**ABSTRACT**

The Stock Market Prediction DBMS Project is a data-driven approach to predict stock prices based on historical data and market trends. This project aims to develop a database management system that utilizes machine learning algorithms to predict stock prices and generate insights for investors. The system collects real-time data from various sources and analyzes it using data mining techniques to provide accurate predictions. The project uses a robust database to store and manage the collected data, and a user-friendly interface for easy access and interpretation of predictions. The system will be helpful for investors to make informed decisions based on reliable predictions, and for researchers to analyze the behavior of the stock market.

**Requirement Analysis**

**LIST OF TABLES**

1. **Stocks Table**
2. **Historical Data Table**
3. **Prediction Data Table**
4. **User Account Table**
5. **Watchlist Table**

**LIST OF ATTRIBUTES FOR EACH TABLE**

1. **Stocks Table**

* **Stock Symbol**
* **Market Capitalization**
* **Price—earnings ratio(P/E)**
* **Earnings per share(EPS)**

1. **Historical Data Table**

* **Stock Symbol**
* **Date**
* **Opening Price**
* **Closing Price**

1. **Prediction Data Table**

* **Stock Symbol**
* **Date**
* **Predicted Price**

1. **User Account Table**

* **Username**
* **Password**
* **Phone Number**

1. **Watchlist Table**

* **User ID**
* **Stock Symbol**

**DOMAIN TYPES**

1. **VARCHAR2**
2. **FLOAT**
3. **DATE**
4. **INTEGER**
5. **TEXT**
6. **DECIMAL**

**DATA TYPES**

1. **Numeric data types**
2. **Character data types**
3. **Data and time data types**
4. **Text data type**

**AIM AND PRIORITY OF THE PROJECT**

To create a **Java GUI-based** desktop application that connects students looking for career choices with skills and Interest. It takes values like student name, username, Age, Skills, etc through forms which are then updated in the database using JDBC connectivity.

**ARCHITECTURE AND TECHNOLOGY**

**Software used:**

Java, Oracle 11g Database, Java SE version 14, Run SQL.

**Java SWING:**

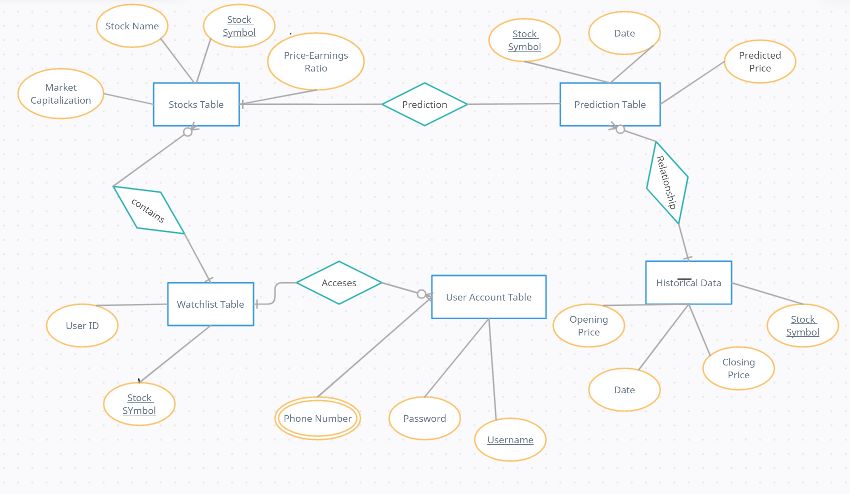
**Java SWING** is a GUI widget toolkit for Java. It is part of Oracle's JavaFoundation Classes (JFC) - an API for providing a graphical user interface (GUI) for Java programs.

Swing was developed to provide a more sophisticated set of GUI components than the earlier AWT. Swing provides a look and feel that emulates the look and feel of several platforms, and also supports a pluggable look and feel that allows applications to have a look and feel unrelated to the underlying platform. It has more powerful and flexible components than AWT. In addition to familiar components such as buttons, check boxes and labels, Swing provides several advanced components such as tabbed panel, scroll panes, trees, tables, and lists.

**SQL:**

Structure Query Language(SQL) is a database query language used for storing and managing data in **Relational** DBMS. SQL was the first commercial language introduced for E.F Codd's Relational model of database. Today almost all RDBMS (MySql, Oracle, Infomix, Sybase, MS Access) use **SQL** as the standard database query language. SQL is used to perform all types of data operations in RDBMS.

ENTITY RELATIONSHIP DIAGRAM



**DDL EXECUTIONS**

**SQL> create table stocks(**

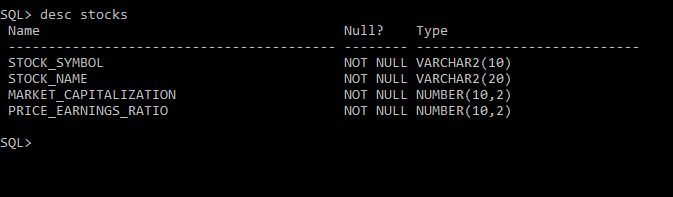
**2 stock\_symbol varchar2(10) primary key,**

**3 stock\_name varchar2(20) not null,**

**4 market\_capitalization decimal(10,2) not null,**

**5 price\_earnings\_ratio decimal(10,2) not null);**

**SQL> desc stocks**

****

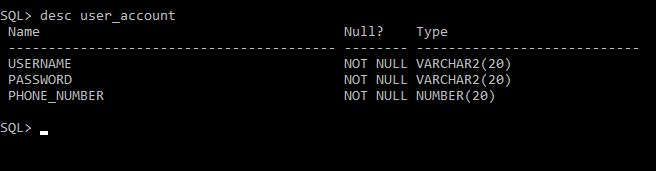
**SQL> create table user\_account(**

**2 username varchar2(20) primary key,**

**3 password varchar2(20) not null,**

**4 phone\_number number(20) not null);**

**SQL> desc user\_account**

****

**SQL> create table watchlist(**

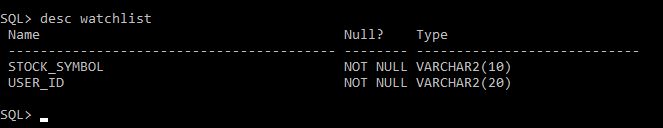
**2 stock\_symbol varchar2(10) not null,**

**3 user\_id varchar2(20) not null);**

**SQL> alter table watchlist**

**2 add constraint fk3\_stock\_symbol foreign key(stock\_symbol) references stocks;**

**SQL> desc watchlist**

****

**SQL> create table historical\_data(**

**2 stock\_symbol varchar2(10) not null,**

**3 day date not null,**

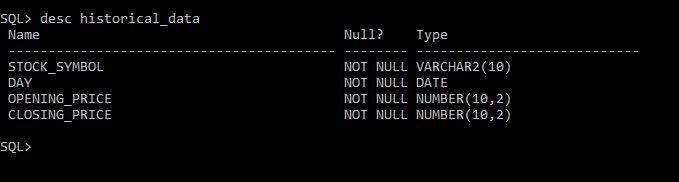
**4 opening\_price decimal(10,2) not null,**

**5 closing\_price decimal(10,2) not null);**

**SQL> alter table historical\_data**

**2 add constraint fk2\_stock\_symbol foreign key(stock\_symbol) references stocks;**

**SQL> desc historical\_data**

****

**SQL> create table Prediction(**

**2 stoct\_symbol varchar2(10) not null,**

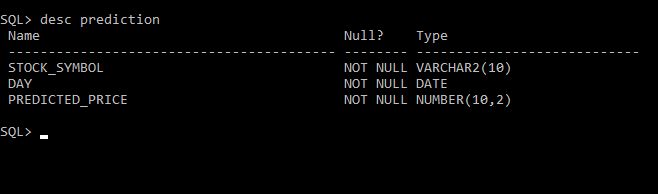
**3 day date not null,**

**4 predicted\_price decimal(10,2) not null);**

**SQL> alter table prediction**

**2 add constraint stock\_symbol foreign key(stock\_symbol) references stocks;**

**SQL> desc prediction**

****

**DML EXECUTIONS**

**SQL> insert into stocks values ('&stock\_symbol','&stock\_name',&market\_capitalization,&price\_earnings\_ratio);**

**Enter value for stock\_symbol: REL**

**Enter value for stock\_name: RELIANCE**

**Enter value for market\_capitalization: 163761.20**

**Enter value for price\_earnings\_ratio: 24.55**

**old 1: insert into stocks values ('&stock\_symbol','&stock\_name',&market\_capitalization,&price\_earnings\_ratio)**

**new 1: insert into stocks values ('REL','RELIANCE',163761.20,24.55)**

**1 row created.**

**SQL> insert into stocks values ('&stock\_symbol','&stock\_name',&market\_capitalization,&price\_earnings\_ratio);**

**Enter value for stock\_symbol: HDFC**

**Enter value for stock\_name: HDFC Ltd.**

**Enter value for market\_capitalization: 941637**

**Enter value for price\_earnings\_ratio: 20.47**

**old 1: insert into stocks values ('&stock\_symbol','&stock\_name',&market\_capitalization,&price\_earnings\_ratio)**

**new 1: insert into stocks values ('HDFC','HDFC Ltd.',941637,20.47)**

**1 row created.**

**SQL> insert into stocks values ('&stock\_symbol','&stock\_name',&market\_capitalization,&price\_earnings\_ratio);**

**Enter value for stock\_symbol: HUL**

**Enter value for stock\_name: HINDUSTAN UNILEVER**

**Enter value for market\_capitalization: 577365**

**Enter value for price\_earnings\_ratio: 58.77**

**old 1: insert into stocks values ('&stock\_symbol','&stock\_name',&market\_capitalization,&price\_earnings\_ratio)**

**new 1: insert into stocks values ('HUL','HINDUSTAN UNILEVER ',577365,58.77)**

**1 row created.**

**SQL> insert into stocks values ('&stock\_symbol','&stock\_name',&market\_capitalization,&price\_earnings\_ratio);**

**Enter value for stock\_symbol: LT**

**Enter value for stock\_name: LARSEN & TOUBRO**

**Enter value for market\_capitalization: 332312**

**Enter value for price\_earnings\_ratio: 32.89**

**old 1: insert into stocks values ('&stock\_symbol','&stock\_name',&market\_capitalization,&price\_earnings\_ratio)**

**new 1: insert into stocks values ('LT','LARSEN & TOUBRO',332312,32.89)**

**1 row created.**

**SQL> insert into stocks values ('&stock\_symbol','&stock\_name',&market\_capitalization,&price\_earnings\_ratio);**

**Enter value for stock\_symbol: INFY**

**Enter value for stock\_name: INFOSIS**

**Enter value for market\_capitalization: 519710**

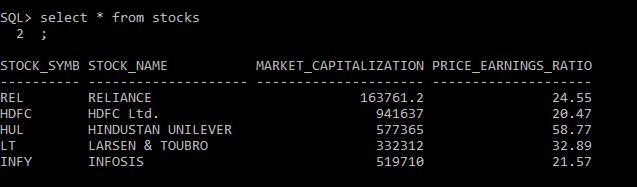
**Enter value for price\_earnings\_ratio: 21.57**

**old 1: insert into stocks values ('&stock\_symbol','&stock\_name',&market\_capitalization,&price\_earnings\_ratio)**

**new 1: insert into stocks values ('INFY','INFOSIS',519710,21.57)**

**1 row created.**

**SQL> SELECT \*FROM STOCKS;**

****

**SQL> insert into user\_account values ('&username','&password',&phone\_number);**

**Enter value for username: Anurag**

**Enter value for password: trader1**

**Enter value for phone\_number: 8919552106**

**old 1: insert into user\_account values ('&username','&password',&phone\_number)**

**new 1: insert into user\_account values ('Anurag','trader1',8919552106)**

**1 row created.**

**SQL> insert into user\_account values ('&username','&password',&phone\_number);**

**Enter value for username: prudhvi**

**Enter value for password: trader2**

**Enter value for phone\_number: 6309825346**

**old 1: insert into user\_account values ('&username','&password',&phone\_number)**

**new 1: insert into user\_account values ('prudhvi','trader2',6309825346)**

**1 row created.**

**SQL> insert into user\_account values ('&username','&password',&phone\_number);**

**Enter value for username: anish**

**Enter value for password: trader3**

**Enter value for phone\_number: 6303749658**

**old 1: insert into user\_account values ('&username','&password',&phone\_number)**

**new 1: insert into user\_account values ('anish','trader3',6303749658)**

**1 row created.**

**SQL> insert into user\_account values ('&username','&password',&phone\_number);**

**Enter value for username: shruthi**

**Enter value for password: trader4**

**Enter value for phone\_number: 9876598328**

**old 1: insert into user\_account values ('&username','&password',&phone\_number)**

**new 1: insert into user\_account values ('shruthi','trader4',9876598328)**

**1 row created.**

**SQL> insert into user\_account values ('&username','&password',&phone\_number);**

**Enter value for username: shriya**

**Enter value for password: trader5**

**Enter value for phone\_number: 9875668768**

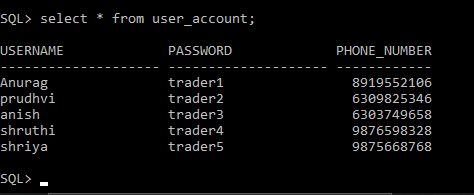
**old 1: insert into user\_account values ('&username','&password',&phone\_number)**

**new 1: insert into user\_account values ('shriya','trader5',9875668768)**

**1 row created.**

**SQL> select \* from user\_account**

**2 ;**

****

**SQL> insert into watchlist values ('&stock\_symbol','&user\_id');**

**Enter value for stock\_symbol: REL**

**Enter value for user\_id: Anurag**

**old 1: insert into watchlist values ('&stock\_symbol','&user\_id')**

**new 1: insert into watchlist values ('REL','Anurag')**

**1 row created.**

**SQL> insert into watchlist values ('&stock\_symbol','&user\_id');**

**Enter value for stock\_symbol: HDFC**

**Enter value for user\_id: Anurag**

**old 1: insert into watchlist values ('&stock\_symbol','&user\_id')**

**new 1: insert into watchlist values ('HDFC','Anurag')**

**1 row created.**

**SQL> insert into watchlist values ('&stock\_symbol','&user\_id');**

**Enter value for stock\_symbol: HUL**

**Enter value for user\_id: Anurag**

**old 1: insert into watchlist values ('&stock\_symbol','&user\_id')**

**new 1: insert into watchlist values ('HUL','Anurag')**

**1 row created.**

**SQL> insert into watchlist values ('&stock\_symbol','&user\_id');**

**Enter value for stock\_symbol: LT**

**Enter value for user\_id: Anish**

**old 1: insert into watchlist values ('&stock\_symbol','&user\_id')**

**new 1: insert into watchlist values ('LT','Anish')**

**1 row created.**

**SQL> insert into watchlist values ('&stock\_symbol','&user\_id');**

**Enter value for stock\_symbol: INFY**

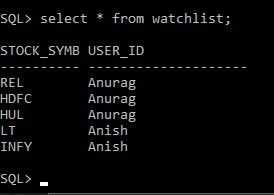
**Enter value for user\_id: Anish**

**old 1: insert into watchlist values ('&stock\_symbol','&user\_id')**

**new 1: insert into watchlist values ('INFY','Anish')**

**1 row created.**

**SQL> select \* from watchlist;**

****

**SQL> insert into historical\_data values ('&stock\_symbol','&day','&opening\_price','&closing\_price');**

**Enter value for stock\_symbol: REL**

**Enter value for day: 28-APR-2023**

**Enter value for opening\_price: 2381.75**

**Enter value for closing\_price: 2423.90**

**old 1: insert into historical\_data values ('&stock\_symbol','&day','&opening\_price','&closing\_price')**

**new 1: insert into historical\_data values ('REL','28-APR-2023','2381.75','2423.90')**

**1 row created.**

**SQL> insert into historical\_data values ('&stock\_symbol','&day','&opening\_price','&closing\_price');**

**Enter value for stock\_symbol: HDFC**

**Enter value for day: 28-APR-2023**

**Enter value for opening\_price: 1670**

**Enter value for closing\_price: 1691**

**old 1: insert into historical\_data values ('&stock\_symbol','&day','&opening\_price','&closing\_price')**

**new 1: insert into historical\_data values ('HDFC','28-APR-2023','1670','1691')**

**1 row created.**

**SQL> insert into historical\_data values ('&stock\_symbol','&day','&opening\_price','&closing\_price');**

**Enter value for stock\_symbol: HUL**

**Enter value for day: 28-APR-2023**

**Enter value for opening\_price: 2419**

**Enter value for closing\_price: 2467.80**

**old 1: insert into historical\_data values ('&stock\_symbol','&day','&opening\_price','&closing\_price')**

**new 1: insert into historical\_data values ('HUL','28-APR-2023','2419','2467.80')**

**1 row created.**

**SQL> insert into historical\_data values ('&stock\_symbol','&day','&opening\_price','&closing\_price');**

**Enter value for stock\_symbol: LT**

**Enter value for day: 28-APR-2023**

**Enter value for opening\_price: 2312.30**

**Enter value for closing\_price: 2369**

**old 1: insert into historical\_data values ('&stock\_symbol','&day','&opening\_price','&closing\_price')**

**new 1: insert into historical\_data values ('LT','28-APR-2023','2312.30','2369')**

**1 row created.**

**SQL> insert into historical\_data values ('&stock\_symbol','&day','&opening\_price','&closing\_price');**

**Enter value for stock\_symbol: INFY**

**Enter value for day: 28-APR-2023**

**Enter value for opening\_price: 1244.95**

**Enter value for closing\_price: 1259**

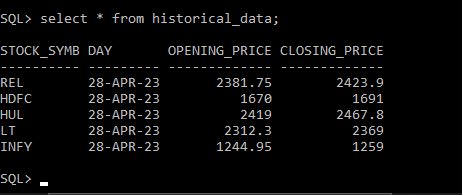
**old 1: insert into historical\_data values ('&stock\_symbol','&day','&opening\_price','&closing\_price')**

**new 1: insert into historical\_data values ('INFY','28-APR-2023','1244.95','1259')**

**1 row created.**

**SQL> SELECT \* FROM HISTORICAL\_DATA**

**2 ;**

****

**SQL> insert into prediction values ('&stock\_symbol','&day','&predicted\_price');**

**Enter value for stock\_symbol: REL**

**Enter value for day: 01-MAY-2023**

**Enter value for predicted\_price: 2450.50**

**old 1: insert into prediction values ('&stock\_symbol','&day','&predicted\_price')**

**new 1: insert into prediction values ('REL','01-MAY-2023','2450.50')**

**1 row created.**

**SQL> insert into prediction values ('&stock\_symbol','&day','&predicted\_price');**

**Enter value for stock\_symbol: HDFC**

**Enter value for day: 01-MAY-2023**

**Enter value for predicted\_price: 1670.60**

**old 1: insert into prediction values ('&stock\_symbol','&day','&predicted\_price')**

**new 1: insert into prediction values ('HDFC','01-MAY-2023','1670.60')**

**1 row created.**

**SQL> insert into prediction values ('&stock\_symbol','&day','&predicted\_price');**

**Enter value for stock\_symbol: HUL**

**Enter value for day: 01-MAY-2023**

**Enter value for predicted\_price: 2350**

**old 1: insert into prediction values ('&stock\_symbol','&day','&predicted\_price')**

**new 1: insert into prediction values ('HUL','01-MAY-2023','2350')**

**1 row created.**

**SQL> insert into prediction values ('&stock\_symbol','&day','&predicted\_price');**

**Enter value for stock\_symbol: LT**

**Enter value for day: 01-MAY-2023**

**Enter value for predicted\_price: 2380.40**

**old 1: insert into prediction values ('&stock\_symbol','&day','&predicted\_price')**

**new 1: insert into prediction values ('LT','01-MAY-2023','2380.40')**

**1 row created.**

**SQL> insert into prediction values ('&stock\_symbol','&day','&predicted\_price');**

**Enter value for stock\_symbol: INFY**

**Enter value for day: 01-MAY-2023**

**Enter value for predicted\_price: 1260.75**

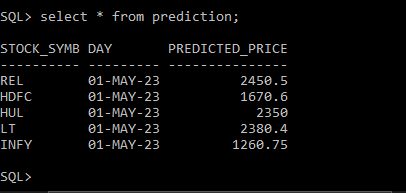
**old 1: insert into prediction values ('&stock\_symbol','&day','&predicted\_price')**

**new 1: insert into prediction values ('INFY','01-MAY-2023','1260.75')**

**1 row created.**

**SQL> SELECT \*FROM PREDICTION**

**2 ;**

****

**IMPLEMENTATION**

**JAVA-SQL Connectivity using JDBC:**

**Java Database Connectivity (JDBC)** is an application programming interface(API) for the programming language Java, which defines how a client may access a database. It is a Java-based data access technology used for Java database connectivity. It is part of the Java Standard Edition platform, from Oracle Corporation. It provides methods to query and update data in a database and is oriented towards relational databases.

The connection to the database can be performed using Java programming (JDBC API) as:

{

DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());

// Connect to Oracle Database

Connection con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE" ,"shruthi","kovvurss");

Statement statement = con.createStatement()

String query = "UPDATE SKILLS SET SS1=" +"'"+ jTextField3.getText() +"',SS2=" +"'"+ jTextField5.getText() +"', AOI ="+" '"+ jTextField2.getText() +"' WHERE SID =+" + jTextField4.getText();

ResultSet rs = statement.executeQuery(query);

JOptionPane.showMessageDialog(new JFrame(), "Upadated Successfully", "INFORMATION", JOptionPane.INFORMATION\_MESSAGE);

rs.close();

statement.close();

con.close(); }

**Front-end Programs (User Interfaces) Home Page:**

Main.java

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.Statement;

public class MainStock {

public static void main(String[] args) {

new PortalStockUI();

}

}

PortalStockUI.java

import java.awt.\*;

import javax.swing.\*;

import java.awt.Color;

import java.awt.FlowLayout;

import java.awt.Font;

import java.awt.GridLayout;

import java.awt.Label;

import java.awt.event.WindowAdapter;

import java.awt.event.WindowEvent;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JMenu;

import javax.swing.JMenuBar;

import javax.swing.JMenuItem;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.Statement;

public class PortalStockUI extends JFrame{

private static final long serialVersionUID = 1L;

private JMenuBar mnu;

private JMenu mnustocks;

private JMenu mnuUserAccount;

private JMenu mnuWatchlist;

private JMenu mnuprediction;

private JMenu mnuHistoricalData;

private JMenuItem insert1, update1, delete1, view1;

private JMenuItem insert2, update2, delete2, view2;

private JMenuItem insert3, update3, delete3, view3;

private JMenuItem insert4, update4, delete4, view4;

private JMenuItem insert5, update5, delete5, view5;

private JLabel labelName;

private static JPanel p0, p1;

public void initialize() {

mnu = new JMenuBar();

mnustocks = new JMenu("Stocks");

mnuUserAccount = new JMenu("User Account");

mnuWatchlist = new JMenu("Watchlist");

mnuprediction = new JMenu("Prediction");

mnuHistoricalData = new JMenu("Historical Data");

//Color lightblue= new Color(51, 204, 255);

labelName=new JLabel("Stock Management System");

labelName.setFont(new Font("Serif", Font.PLAIN, 40));

labelName.setForeground(Color.BLACK);

p1=new JPanel();

p0=new JPanel();

insert1 = new JMenuItem("Insert");

update1 = new JMenuItem("Update");

delete1 = new JMenuItem("Delete");

view1 = new JMenuItem("View");

insert2 = new JMenuItem("Insert");

update2 = new JMenuItem("Update");

delete2 = new JMenuItem("Delete");

view2 = new JMenuItem("View");

insert3 = new JMenuItem("Insert");

update3 = new JMenuItem("Update");

delete3 = new JMenuItem("Delete");

view3 = new JMenuItem("View");

insert4 = new JMenuItem("Insert");

update4 = new JMenuItem("Update");

delete4 = new JMenuItem("Delete");

view4 = new JMenuItem("View");

insert5 = new JMenuItem("Insert");

update5 = new JMenuItem("Update");

delete5 = new JMenuItem("Delete");

view5 = new JMenuItem("View");

}

void addComponentsToFrame() {

mnustocks.add(insert1);

mnustocks.add(delete1);

mnustocks.add(update1);

mnustocks.add(view1);

mnuUserAccount.add(insert2);

mnuUserAccount.add(delete2);

mnuUserAccount.add(update2);

mnuUserAccount.add(view2);

mnuWatchlist.add(insert4);

mnuWatchlist.add(delete4);

mnuWatchlist.add(update4);

mnuWatchlist.add(view4);

mnuprediction.add(insert3);

mnuprediction.add(delete3);

mnuprediction.add(update3);

mnuprediction.add(view3);

mnuHistoricalData.add(insert5);

mnuHistoricalData.add(delete5);

mnuHistoricalData.add(update5);

mnuHistoricalData.add(view5);

mnu.add(mnustocks);

mnu.add(mnuUserAccount);

mnu.add(mnuWatchlist);

mnu.add(mnuprediction);

mnu.add(mnuHistoricalData);

setJMenuBar(mnu);

p1.add(labelName);

p1.setAlignmentY(CENTER\_ALIGNMENT);

p1.setBounds(500, 500, 800, 100);

p0.add(p1);

p0.setBackground(Color.WHITE);

add(p0);

}

void closeWindow(){

try {

int a = JOptionPane.showConfirmDialog(this,"Are you sure want to Quit ?");

if(a == JOptionPane.YES\_OPTION){

System.exit(0);

}

else if (a == JOptionPane.NO\_OPTION) {

setDefaultCloseOperation(JFrame.DO\_NOTHING\_ON\_CLOSE);

}

else if (a == JOptionPane.CANCEL\_OPTION) {

setDefaultCloseOperation(JFrame.DO\_NOTHING\_ON\_CLOSE);

}

}

catch(Exception e) {

System.out.println(e);

}

}

void register() {

stocks s=new stocks(p0, PortalStockUI.this, insert1, update1, view1, delete1);

s.buildGUI();

UserAccount ua = new UserAccount(p0, PortalStockUI.this, insert2, update2, view2, delete2);

ua.buildGUI();

Watchlist w= new Watchlist(p0, PortalStockUI.this, insert3, update3, view3, delete3);

w.buildGUI();

prediction p= new prediction(p0, PortalStockUI.this, insert4, update4, view4, delete4);

p.buildGUI();

HistoricalData hd= new HistoricalData(p0, PortalStockUI.this, insert5, update5, view5, delete5);

hd.buildGUI();

addWindowListener(new WindowAdapter(){

public void windowClosing(WindowEvent we)

{

closeWindow();

}

});

}

public PortalStockUI() {

initialize();

addComponentsToFrame();

register();

pack();

setTitle("Stock Market Prediction Database");

setSize(800, 800);

setVisible(true);

}

}

Stocks.java

import java.awt.BorderLayout;

import java.awt.Choice;

import java.awt.Color;

import java.awt.FlowLayout;

import java.awt.Font;

import java.awt.GridLayout;

import java.awt.List;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.ItemEvent;

import java.awt.event.ItemListener;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JMenuItem;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JScrollPane;

import javax.swing.JTable;

import javax.swing.JTextField;

import javax.swing.table.DefaultTableModel;

public class stocks{

private JPanel pn, pn1, pn2, pn3;

private JFrame jf;

private JButton JB\_insert, JB\_modify, JB\_view, JB\_delete;

private JLabel JL\_stock\_symbol, JL\_stock\_name, JL\_market\_capitalization, JL\_price\_earnings\_ratio,JL\_date,JL\_status;

private JTextField JTF\_stock\_symbol, JTF\_stock\_name, JTF\_market\_capitalization, JTF\_price\_earnings\_ratio, JTF\_date,JTF\_status;

private JMenuItem insert2, update2, view2, delete2;

private List Stocks\_List;

private Choice stock\_name;

public stocks(JPanel pn, JFrame jf, JMenuItem insert2, JMenuItem update2, JMenuItem view2, JMenuItem

delete2){

try {

Class.forName("oracle.jdbc.driver.OracleDriver");

}

catch (Exception e) {

System.err.println("Unable to find and load driver");

System.exit(1);

}

this.jf = jf;

this.insert2 = insert2;

this.update2 = update2;

this.view2 = view2;

this.delete2 = delete2;

JL\_stock\_symbol = new JLabel("Stock Symbol : ");

JTF\_stock\_symbol = new JTextField(10);

JL\_stock\_name = new JLabel("Stock Name: ");

JTF\_stock\_name = new JTextField(10);

JL\_market\_capitalization = new JLabel("Market Capitalization: ");

JTF\_market\_capitalization = new JTextField(10);

JL\_price\_earnings\_ratio = new JLabel("price\_earnings\_ratio: ");

JTF\_price\_earnings\_ratio = new JTextField(10);

JL\_date = new JLabel("Date of request: ");

JTF\_date = new JTextField(10);

JL\_status = new JLabel("Status of request: ");

JTF\_status = new JTextField(10);

this.pn=pn;

}

private void displaySQLErrors(SQLException e) {

JOptionPane.showMessageDialog(pn1,"\nSQLException: " + e.getMessage() + "\n"+"SQLState: " +

e.getSQLState() + "\n"+"VendorError: " + e.getErrorCode() + "\n");

}

public void load\_details(){

try{

stock\_name= new Choice();

stock\_name.removeAll();

Connection con1 = DriverManager.getConnection("jdbc:oracle:thin:localhost:1521:xe","vasavi", "abcd");

Statement st1 = con1.createStatement();

ResultSet rs1 = st1.executeQuery("select \* from stocks");

while(rs1.next()) {

stock\_name.add(rs1.getString("stock\_name"));

}

}

catch(SQLException e) {

displaySQLErrors(e);

}

}

public void loaddetails(){

try{

Stocks\_List = new List();

Stocks\_List.removeAll();

Connection con2 = DriverManager.getConnection("jdbc:oracle:thin:localhost:1521:xe","vasavi", "abcd");

Statement st2 = con2.createStatement();

ResultSet rs2 = st2.executeQuery("select \* from stocks");

while(rs2.next()) {

Stocks\_List.add(rs2.getString("stock\_name"));

}

}

catch(SQLException e) {

displaySQLErrors(e);

}

}

public void buildGUI(){

insert2.addActionListener(new ActionListener(){

@Override

public void actionPerformed(ActionEvent aevt){

JB\_insert = new JButton("Submit");

JTF\_stock\_symbol.setText(null);

JTF\_stock\_name.setText(null);

JTF\_market\_capitalization.setText(null);

JTF\_price\_earnings\_ratio.setText(null);

JTF\_date.setText(null);

JTF\_status.setText(null);

loaddetails();

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

pn1 = new JPanel();

pn1.setLayout(new GridLayout(10,10));

pn1.add(JL\_stock\_symbol);

pn1.add(JTF\_stock\_symbol);

pn1.add(JL\_stock\_name);

pn1.add(JTF\_stock\_name);

pn1.add(JL\_market\_capitalization);

pn1.add(JTF\_market\_capitalization);

pn1.add(JL\_price\_earnings\_ratio);

pn1.add(JTF\_price\_earnings\_ratio);

pn1.add(JL\_date);

pn1.add(JTF\_date);

pn1.add(JL\_status);

pn1.add(JTF\_status);

pn3 = new JPanel(new FlowLayout());

pn3.add(JB\_insert);

pn1.setBounds(115, 80, 300, 250);

pn3.setBounds(200, 350, 75, 35);

pn2 = new JPanel(new FlowLayout());

Stocks\_List = new List(10);

loaddetails();

pn2.add(Stocks\_List);

pn2.setBounds(200, 350, 300, 180);

pn.add(pn1);

pn.add(pn3);

pn.add(pn2);

pn.setLayout(new BorderLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

JB\_insert.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

try{

Connection con = DriverManager.getConnection("jdbc:oracle:thin:localhost:1521:xe","vasavi", "abcd");

String query = "INSERT INTO stocks (stock\_symbol, stock\_name, market\_capitalization, price\_earnings\_ratio, date, status) values(?, ?, ?, ?, ?, ?)";

PreparedStatement stp = con.prepareStatement(query);

stp.setString(1, JTF\_stock\_symbol.getText());

stp.setString(2, JTF\_stock\_name.getText());

stp.setString(3, JTF\_market\_capitalization.getText());

stp.setString(4, JTF\_price\_earnings\_ratio.getText());

stp.setString(5, JTF\_date.getText());

stp.setString(6, JTF\_status.getText());

int i = stp.executeUpdate();

con.close();

if(i > 0){

JOptionPane.showMessageDialog(pn,"\nInserted successfully");

}

}

catch(SQLException e) {

displaySQLErrors(e);

}

}

});

}

});

update2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

JB\_modify = new JButton("Modify");

JTF\_stock\_symbol.setText(null);

JTF\_stock\_name.setText(null);

JTF\_market\_capitalization.setText(null);

JTF\_price\_earnings\_ratio.setText(null);

JTF\_date.setText(null);

JTF\_status.setText(null);

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

pn1 = new JPanel();

pn1.setLayout(new GridLayout(10, 10));

pn1.add(JL\_stock\_symbol);

pn1.add(JTF\_stock\_symbol);

pn1.add(JL\_stock\_name);

pn1.add(JTF\_stock\_name);

pn1.add(JL\_market\_capitalization);

pn1.add(JTF\_market\_capitalization);

pn1.add(JL\_price\_earnings\_ratio);

pn1.add(JTF\_price\_earnings\_ratio);

pn1.add(JL\_date);

pn1.add(JTF\_date);

pn1.add(JL\_status);

pn1.add(JTF\_status);

pn3 = new JPanel(new FlowLayout());

pn3.add(JB\_modify);

pn1.setBounds(115, 80, 300, 250);

pn3.setBounds(200, 350, 75, 35);

pn2 =new JPanel(new FlowLayout());

Stocks\_List = new List(10);

loaddetails();

pn2.add(Stocks\_List);

pn2.setBounds(200, 350, 300, 180);

pn.add(pn1);

pn.add(pn3);

pn.add(pn2);

pn.setLayout(new BorderLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

Stocks\_List.addItemListener(new ItemListener() {

public void itemStateChanged(ItemEvent ievt){

try {

Connection con3 = DriverManager.getConnection("jdbc:oracle:thin:localhost:1521:xe","vasavi", "abcd");

Statement st3 = con3.createStatement();

ResultSet rs3 = st3.executeQuery("select \* from stocks");

while (rs3.next()) {

if

(rs3.getString("stock\_name").equals(Stocks\_List.getSelectedItem()))

break;

}

if (!rs3.isAfterLast()) {

JTF\_stock\_symbol.setText(rs3.getString("stock\_symbol"));

JTF\_stock\_name.setText(rs3.getString("stock\_name"));

JTF\_market\_capitalization.setText(rs3.getString("market\_capitalization"));

JTF\_price\_earnings\_ratio.setText(rs3.getString("price\_earnings\_ratio"));

JTF\_date.setText(rs3.getString("date"));

JTF\_status.setText(rs3.getString("status"));

}

}

catch (SQLException selectException) {

displaySQLErrors(selectException);

}

}

});

JB\_modify.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

try{

int a = JOptionPane.showConfirmDialog(pn, "Are you sure want to update:");

if(a == JOptionPane.YES\_OPTION){

String query = "update stocks set stock\_symbol= ?,market\_capitalization=?,price\_earnings\_ratio=?, date = ?, status = ? where stock\_name= ?";

Connection con4 = DriverManager.getConnection("jdbc:oracle:thin:localhost:1521:xe","vasavi", "abcd");

PreparedStatement stp2 = con4.prepareStatement(query);

stp2.setString(1, JTF\_stock\_symbol.getText());

stp2.setString(2, JTF\_stock\_name.getText());

stp2.setString(3, JTF\_market\_capitalization.getText());

stp2.setString(4, JTF\_price\_earnings\_ratio.getText());

stp2.setString(5, JTF\_date.getText());

stp2.setString(6, JTF\_status.getText());

int i = stp2.executeUpdate();

if(i>0){

JOptionPane.showMessageDialog(pn,"\nUpdated rows succesfully");

}

loaddetails();

}

}

catch(SQLException e){

displaySQLErrors(e);

}

}

});

}

});

delete2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

JB\_delete = new JButton("Delete");

JTF\_stock\_symbol.setText(null);

JTF\_stock\_name.setText(null);

JTF\_market\_capitalization.setText(null);

JTF\_price\_earnings\_ratio.setText(null);

JTF\_date.setText(null);

JTF\_status.setText(null);

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

pn1 = new JPanel();

pn1.setLayout(new GridLayout(10, 10));

pn1.add(JL\_stock\_symbol);

pn1.add(JTF\_stock\_symbol);

pn1.add(JL\_stock\_name);

pn1.add(JTF\_stock\_name);

pn1.add(JL\_market\_capitalization);

pn1.add(JTF\_market\_capitalization);

pn1.add(JL\_price\_earnings\_ratio);

pn1.add(JTF\_price\_earnings\_ratio);

pn1.add(JL\_date);

pn1.add(JTF\_date);

pn1.add(JL\_status);

pn1.add(JTF\_status);

pn3 = new JPanel(new FlowLayout());

pn3.add(JB\_delete);

pn1.setBounds(115, 80, 300, 250);

pn3.setBounds(200, 350, 75, 35);

pn2 = new JPanel(new FlowLayout());

Stocks\_List = new List(10);

loaddetails();

pn2.add(Stocks\_List);

pn2.setBounds(200, 350, 300, 200);

pn.add(pn1);

pn.add(pn3);

pn.add(pn2);

pn.setLayout(new BorderLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

Stocks\_List.addItemListener(new ItemListener() {

public void itemStateChanged(ItemEvent ievt){

try {

Connection con5 = DriverManager.getConnection("jdbc:oracle:thin:localhost:1521:xe","vasavi", "abcd");

Statement st4 = con5.createStatement();

ResultSet rs4 = st4.executeQuery("select \* from stocks");

while (rs4.next()) {

if

(rs4.getString("stock\_name").equals(Stocks\_List.getSelectedItem()))

break;

}

if (!rs4.isAfterLast()) {

JTF\_stock\_symbol.setText(rs4.getString("stock\_symbol"));

JTF\_stock\_name.setText(rs4.getString("stock\_name"));

JTF\_market\_capitalization.setText(rs4.getString("market\_capitalization"));

JTF\_price\_earnings\_ratio.setText(rs4.getString("price\_earnings\_ratio"));

JTF\_date.setText(rs4.getString("date"));

JTF\_status.setText(rs4.getString("status"));

}

}

catch (SQLException selectException) {

displaySQLErrors(selectException);

}

}

});

JB\_delete.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

try {

int a = JOptionPane.showConfirmDialog(pn,"Are you sure want to Delete:");

if(a == JOptionPane.YES\_OPTION){

String query = "DELETE FROM stocks WHERE stock\_name = ?";

Connection con6 = DriverManager.getConnection("jdbc:oracle:thin:localhost:1521:xe","vasavi", "abcd");

PreparedStatement stp3 = con6.prepareStatement(query);

stp3.setString(1, JTF\_market\_capitalization.getText());

int i = stp3.executeUpdate();

if(i>0){

JOptionPane.showMessageDialog(pn,"\nDeleted rows succesfully");

}

loaddetails();

}

}

catch(SQLException e){

displaySQLErrors(e);

}

}

});

}

});

view2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

JLabel view = new JLabel("Stock Details View");

view.setForeground(Color.BLACK);

JB\_view = new JButton("View");

Font myFont = new Font("Serif",Font.BOLD,25);

view.setFont((myFont));

pn1 = new JPanel();

pn2 = new JPanel();

pn1.add(view);

pn2.add(JB\_view);

pn.add(pn1);

pn.add(pn2);

pn.setBounds(500, 800, 300, 300);

pn.setLayout(new FlowLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

JB\_view.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

JFrame jfr = new JFrame("stocks details");

JTable jt;

DefaultTableModel model = new DefaultTableModel();

jt = new JTable(model);

model.addColumn("stock\_symbol");

model.addColumn("stock\_name");

model.addColumn("market\_capitalization");

model.addColumn("price\_earnings\_ratio");

model.addColumn("date");

model.addColumn("status");

try {

Connection con7 = DriverManager.getConnection("jdbc:oracle:thin:localhost:1521:xe", "vasavi", "abcd");

Statement st4 = con7.createStatement();

ResultSet rs5 = st4.executeQuery("select \* from stocks");

while(rs5.next()){

model.addRow(new Object[]{rs5.getString("stock\_symbol"), rs5.getString("stock\_name"), rs5.getString("market\_capitalization"), rs5.getString("price\_earnings\_ratio"), rs5.getString("date"),rs5.getString("status")});

}

}

catch(SQLException e){

displaySQLErrors(e);

}

jt.setEnabled(false);

jt.setBounds(30, 40, 300, 300);

JScrollPane jsp = new JScrollPane(jt);

jfr.add(jsp);

jfr.setSize(800, 400);

jfr.setVisible(true);

}

});

}

});

}

}

User\_account.java

import java.awt.BorderLayout;

import java.awt.Choice;

import java.awt.Color;

import java.awt.FlowLayout;

import java.awt.Font;

import java.awt.GridLayout;

import java.awt.List;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.ItemEvent;

import java.awt.event.ItemListener;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JMenuItem;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JScrollPane;

import javax.swing.JTable;

import javax.swing.JTextField;

import javax.swing.table.DefaultTableModel;

public class UserAccount{

private JPanel pn, pn1, pn2, pn3;

private JFrame jf;

private JButton JB\_insert, JB\_modify, JB\_view, JB\_delete;

private JLabel JL\_username, JL\_password, JL\_phone\_number;

private JTextField JTF\_username, JTF\_password, JTF\_phone\_number;

private JMenuItem insert2, update2, view2, delete2;

private List Users\_List;

private Choice username;

public UserAccount(JPanel pn, JFrame jf, JMenuItem insert2, JMenuItem update2, JMenuItem view2, JMenuItem delete2){

try {

Class.forName("oracle.jdbc.driver.OracleDriver");

}

catch (Exception e) {

System.err.println("Unable to find and load driver");

System.exit(1);

}

this.jf = jf;

this.insert2 = insert2;

this.update2 = update2;

this.view2 = view2;

this.delete2 = delete2;

JL\_username = new JLabel("username : ");

JTF\_username = new JTextField(10);

JL\_password = new JLabel("password: ");

JTF\_password = new JTextField(10);

JL\_phone\_number = new JLabel("phone\_number: ");

JTF\_phone\_number = new JTextField(10);

this.pn=pn;

}

private void displaySQLErrors(SQLException e) {

JOptionPane.showMessageDialog(pn1,"\nSQLException: " + e.getMessage() + "\n"+"SQLState: " +

e.getSQLState() + "\n"+"VendorError: " + e.getErrorCode() + "\n");

}

public void load\_details(){

try{

username = new Choice();

username.removeAll();

Connection con1 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi","abcd");

Statement st1 = con1.createStatement();

ResultSet rs1 = st1.executeQuery("select \* from user\_account");

while(rs1.next()) {

username.add(rs1.getString("username"));

}

}

catch(SQLException e) {

displaySQLErrors(e);

}

}

public void loaddetails(){

try{

Users\_List = new List();

Users\_List.removeAll();

Connection con2 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st2 = con2.createStatement();

ResultSet rs2 = st2.executeQuery("select \* from user\_account");

while(rs2.next()) {

Users\_List.add(rs2.getString("username"));

}

}

catch(SQLException e) {

displaySQLErrors(e);

}

}

public void buildGUI(){

insert2.addActionListener(new ActionListener(){

@Override

public void actionPerformed(ActionEvent aevt){

JB\_insert = new JButton("Submit");

JTF\_username.setText(null);

JTF\_password.setText(null);

JTF\_phone\_number.setText(null);

loaddetails();

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

pn1 = new JPanel();

pn1.setLayout(new GridLayout(10, 10));

pn1.add(JL\_username);

pn1.add(JTF\_username);

pn1.add(JL\_password);

pn1.add(JTF\_password);

pn1.add(JL\_phone\_number);

pn1.add(JTF\_phone\_number);

pn3 = new JPanel(new FlowLayout());

pn3.add(JB\_insert);

pn1.setBounds(115, 80, 300, 250);

pn3.setBounds(200, 350, 75, 35);

pn2 = new JPanel(new FlowLayout());

Users\_List = new List(10);

loaddetails();

pn2.add(Users\_List);

pn2.setBounds(200, 350, 300, 180);

pn.add(pn1);

pn.add(pn3);

pn.add(pn2);

pn.setLayout(new BorderLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

JB\_insert.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

try{

Connection con =DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi","abcd");

String query = "INSERT INTO user\_account (username,password,phone\_number) values(?, ?, ?)";

PreparedStatement stp = con.prepareStatement(query);

stp.setString(1, JTF\_username.getText());

stp.setString(2, JTF\_password.getText());

stp.setString(3, JTF\_phone\_number.getText());

int i = stp.executeUpdate();

con.close();

if(i > 0){

JOptionPane.showMessageDialog(pn,"\nInserted successfully");

}

}

catch(SQLException e) {

displaySQLErrors(e);

}

}

});

}

});

update2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

JB\_modify = new JButton("Modify");

JTF\_username.setText(null);

JTF\_password.setText(null);

JTF\_phone\_number.setText(null);

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

pn1 = new JPanel();

pn1.setLayout(new GridLayout(10, 10));

pn1.add(JL\_username);

pn1.add(JTF\_username);

pn1.add(JL\_password);

pn1.add(JTF\_password);

pn1.add(JL\_phone\_number);

pn1.add(JTF\_phone\_number);

pn3 = new JPanel(new FlowLayout());

pn3.add(JB\_modify);

pn1.setBounds(115, 80, 300, 250);

pn3.setBounds(200, 350, 75, 35);

pn2 =new JPanel(new FlowLayout());

Users\_List = new List(10);

loaddetails();

pn2.add(Users\_List);

pn2.setBounds(200, 350, 300, 180);

pn.add(pn1);

pn.add(pn3);

pn.add(pn2);

pn.setLayout(new BorderLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

Users\_List.addItemListener(new ItemListener() {

public void itemStateChanged(ItemEvent ievt){

try {

Connection con3 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st3 = con3.createStatement();

ResultSet rs3 = st3.executeQuery("select \* from user\_account");

while (rs3.next()) {

if

(rs3.getString("username").equals(Users\_List.getSelectedItem()))

break;

}

if (!rs3.isAfterLast()) {

JTF\_username.setText(rs3.getString("username"));

JTF\_password.setText(rs3.getString("password"));

JTF\_phone\_number.setText(rs3.getString("phone\_number"));

}

}

catch (SQLException selectException) {

displaySQLErrors(selectException);

}

}

});

JB\_modify.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

try{

int a = JOptionPane.showConfirmDialog(pn, "Are you sure you want to update:");

if(a == JOptionPane.YES\_OPTION){

String query = "update user\_account set password = ?, phone\_number = ? where username=?";

Connection con4 =DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

PreparedStatement stp2 = con4.prepareStatement(query);

stp2.setString(1, JTF\_password.getText());

stp2.setString(2, JTF\_phone\_number.getText());

stp2.setString(3, JTF\_username.getText());

int i = stp2.executeUpdate();{

JOptionPane.showMessageDialog(pn,"\nUpdated rows succesfully");

}

loaddetails();

}

}

catch(SQLException e){

displaySQLErrors(e);

}

}

});

}

});

delete2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

JB\_delete = new JButton("Delete");

JTF\_username.setText(null);

JTF\_password.setText(null);

JTF\_phone\_number.setText(null);

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

pn1 = new JPanel();

pn1.setLayout(new GridLayout(10, 10));

pn1.add(JL\_username);

pn1.add(JTF\_username);

pn1.add(JL\_password);

pn1.add(JTF\_password);

pn1.add(JL\_phone\_number);

pn1.add(JTF\_phone\_number);

pn3 = new JPanel(new FlowLayout());

pn3.add(JB\_delete);

pn1.setBounds(115, 80, 300, 250);

pn3.setBounds(200, 350, 75, 35);

pn2 = new JPanel(new FlowLayout());

Users\_List = new List(10);

loaddetails();

pn2.add(Users\_List);

pn2.setBounds(200, 350, 300, 200);

pn.add(pn1);

pn.add(pn3);

pn.add(pn2);

pn.setLayout(new BorderLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

Users\_List.addItemListener(new ItemListener() {

public void itemStateChanged(ItemEvent ievt){

try {

Connection con5 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st4 = con5.createStatement();

ResultSet rs4 = st4.executeQuery("select \* from user\_account");

while (rs4.next()) {

if

(rs4.getString("username").equals(Users\_List.getSelectedItem()))

break;

}

if (!rs4.isAfterLast()) {

JTF\_username.setText(rs4.getString("username"));

JTF\_password.setText(rs4.getString("password"));

JTF\_phone\_number.setText(rs4.getString("phone\_number"));

}

}

catch (SQLException selectException) {

displaySQLErrors(selectException);

}

}

});

JB\_delete.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

try {

int a = JOptionPane.showConfirmDialog(pn, "Are you sure want to Delete:");

if(a == JOptionPane.YES\_OPTION){

String query = "DELETE FROM user\_account WHERE username = ? ";

Connection con6 = DriverManager.getConnection("jdbc:oracle:thin:@@localhost:1521:xe","vasavi", "abcd");

PreparedStatement stp3 = con6.prepareStatement(query);

stp3.setString(1, JTF\_username.getText());

int i = stp3.executeUpdate(query);

if(i>0){

JOptionPane.showMessageDialog(pn,"\nDeleted rows succesfully");

}

loaddetails();

}

}

catch(SQLException e){

displaySQLErrors(e);

}

}

});

}

});

view2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

JLabel view = new JLabel("User details View");

view.setForeground(Color.BLACK);

JB\_view = new JButton("View");

Font myFont = new Font("Serif", Font.BOLD,25);

view.setFont((myFont));

pn1 = new JPanel();

pn2 = new JPanel();

pn1.add(view);

pn2.add(JB\_view);

pn.add(pn1);

pn.add(pn2);

pn.setBounds(500, 800, 300, 300);

pn.setLayout(new FlowLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

JB\_view.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

JFrame jfr = new JFrame("User Details");

JTable jt;

DefaultTableModel model = new DefaultTableModel();

jt = new JTable(model);

model.addColumn("username");

model.addColumn("password");

model.addColumn("phone\_number");

try {

Connection con7 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st4 = con7.createStatement();

ResultSet rs5 = st4.executeQuery("select \* from user\_account");

while(rs5.next()){

model.addRow(new Object[]{rs5.getString("username"), rs5.getString("password"), rs5.getString("phone\_number")});

}

}

catch(SQLException e){

displaySQLErrors(e);

}

jt.setEnabled(false);

jt.setBounds(30, 40, 300, 300);

JScrollPane jsp = new JScrollPane(jt);

jfr.add(jsp);

jfr.setSize(800, 400);

jfr.setVisible(true);

}

});

}

});

}}

Watchlist.java

import java.awt.BorderLayout;

import java.awt.Choice;

import java.awt.Color;

import java.awt.FlowLayout;

import java.awt.Font;

import java.awt.GridLayout;

import java.awt.List;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.ItemEvent;

import java.awt.event.ItemListener;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JMenuItem;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JScrollPane;

import javax.swing.JTable;

import javax.swing.JTextField;

import javax.swing.table.DefaultTableModel;

public class Watchlist{

private JPanel pn, pn1, pn2, pn3;

private JFrame jf;

private JButton JB\_insert, JB\_modify, JB\_view, JB\_delete;

private JLabel JL\_stock\_symbol, JL\_user\_id;

private JTextField JTF\_stock\_symbol, JTF\_user\_id;

private JMenuItem insert2, update2, view2, delete2;

private List Watchlist\_List;

private Choice stock\_symbol;

public Watchlist(JPanel pn, JFrame jf, JMenuItem insert2, JMenuItem update2, JMenuItem view2, JMenuItem

delete2){

try {

Class.forName("oracle.jdbc.driver.OracleDriver");

}

catch (Exception e) {

System.err.println("Unable to find and load driver");

System.exit(1);

}

this.jf = jf;

this.insert2 = insert2;

this.update2 = update2;

this.view2 = view2;

this.delete2 = delete2;

JL\_stock\_symbol = new JLabel("stock\_symbol : ");

JTF\_stock\_symbol = new JTextField(10);

JL\_user\_id = new JLabel("User Id: ");

JTF\_user\_id = new JTextField(10);

this.pn=pn;

}

private void displaySQLErrors(SQLException e) {

JOptionPane.showMessageDialog(pn1,"\nSQLException: " + e.getMessage() + "\n"+"SQLState: " +

e.getSQLState() + "\n"+"VendorError: " + e.getErrorCode() + "\n");

}

public void load\_details(){

try{

stock\_symbol= new Choice();

stock\_symbol.removeAll();

Connection con1 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st1 = con1.createStatement();

ResultSet rs1 = st1.executeQuery("select \* from watchlist");

while(rs1.next()) {

stock\_symbol.add(rs1.getString("stock\_symbol"));

}

}

catch(SQLException e) {

displaySQLErrors(e);

}

}

public void loaddetails(){

try{

Watchlist\_List = new List();

Watchlist\_List.removeAll();

Connection con2 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st2 = con2.createStatement();

ResultSet rs2 = st2.executeQuery("select \* from watchlist");

while(rs2.next()) {

Watchlist\_List.add(rs2.getString("stock\_symbol"));

}

}

catch(SQLException e) {

displaySQLErrors(e);

}

}

public void buildGUI(){

insert2.addActionListener(new ActionListener(){

@Override

public void actionPerformed(ActionEvent aevt){

JB\_insert = new JButton("Submit");

JTF\_stock\_symbol.setText(null);

JTF\_user\_id.setText(null);

loaddetails();

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

pn1 = new JPanel();

pn1.setLayout(new GridLayout(10,10));

pn1.add(JL\_stock\_symbol);

pn1.add(JTF\_stock\_symbol);

pn1.add(JL\_user\_id);

pn1.add(JTF\_user\_id);

pn3 = new JPanel(new FlowLayout());

pn3.add(JB\_insert);

pn1.setBounds(115, 80, 300, 250);

pn3.setBounds(200, 350, 75, 35);

pn2 = new JPanel(new FlowLayout());

Watchlist\_List = new List(10);

loaddetails();

pn2.add(Watchlist\_List);

pn2.setBounds(200, 350, 300, 180);

pn.add(pn1);

pn.add(pn3);

pn.add(pn2);

pn.setLayout(new BorderLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

JB\_insert.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

try{

Connection con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

String query = "INSERT INTO watchlist (stock\_symbol, user\_id) values(?, ?)";

PreparedStatement stp = con.prepareStatement(query);

stp.setString(1, JTF\_stock\_symbol.getText());

stp.setString(2, JTF\_user\_id.getText());

int i = stp.executeUpdate();

con.close();

if(i > 0){

JOptionPane.showMessageDialog(pn,"\nInserted successfully");

}

}

catch(SQLException e) {

displaySQLErrors(e);

}

}

});

}

});

update2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

JB\_modify = new JButton("Modify");

JTF\_stock\_symbol.setText(null);

JTF\_user\_id.setText(null);

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

pn1 = new JPanel();

pn1.setLayout(new GridLayout(10, 10));

pn1.add(JL\_stock\_symbol);

pn1.add(JTF\_stock\_symbol);

pn1.add(JL\_user\_id);

pn1.add(JTF\_user\_id);

pn3 = new JPanel(new FlowLayout());

pn3.add(JB\_modify);

pn1.setBounds(115, 80, 300, 250);

pn3.setBounds(200, 350, 75, 35);

pn2 =new JPanel(new FlowLayout());

Watchlist\_List = new List(10);

loaddetails();

pn2.add(Watchlist\_List);

pn2.setBounds(200, 350, 300, 180);

pn.add(pn1);

pn.add(pn3);

pn.add(pn2);

pn.setLayout(new BorderLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

Watchlist\_List.addItemListener(new ItemListener() {

public void itemStateChanged(ItemEvent ievt){

try {

Connection con3 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st3 = con3.createStatement();

ResultSet rs3 = st3.executeQuery("select \* from watchlist");

while (rs3.next()) {

if

(rs3.getString("stock\_symbol").equals(Watchlist\_List.getSelectedItem()))

break;

}

if (!rs3.isAfterLast()) {

JTF\_stock\_symbol.setText(rs3.getString("stock\_symbol"));

JTF\_user\_id.setText(rs3.getString("user\_id"));

}

}

catch (SQLException selectException) {

displaySQLErrors(selectException);

}

}

});

JB\_modify.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

try{

int a = JOptionPane.showConfirmDialog(pn, "Are you sure want to update:");

if(a == JOptionPane.YES\_OPTION){

String query = "update watchlist set user\_id = ? where stock\_symbol = ?";

Connection con4 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

PreparedStatement stp2 = con4.prepareStatement(query);

stp2.setString(1, JTF\_stock\_symbol.getText());

stp2.setString(2, JTF\_user\_id.getText());

int i = stp2.executeUpdate();

if(i>0){

JOptionPane.showMessageDialog(pn,"\nUpdated rows succesfully");

}

loaddetails();

}

}

catch(SQLException e){

displaySQLErrors(e);

}

}

});

}

});

delete2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

JB\_delete = new JButton("Delete");

JTF\_stock\_symbol.setText(null);

JTF\_user\_id.setText(null);

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

pn1 = new JPanel();

pn1.setLayout(new GridLayout(10, 10));

pn1.add(JL\_stock\_symbol);

pn1.add(JTF\_stock\_symbol);

pn1.add(JL\_user\_id);

pn1.add(JTF\_user\_id);

pn3 = new JPanel(new FlowLayout());

pn3.add(JB\_delete);

pn1.setBounds(115, 80, 300, 250);

pn3.setBounds(200, 350, 75, 35);

pn2 = new JPanel(new FlowLayout());

Watchlist\_List = new List(10);

loaddetails();

pn2.add(Watchlist\_List);

pn2.setBounds(200, 350, 300, 200);

pn.add(pn1);

pn.add(pn3);

pn.add(pn2);

pn.setLayout(new BorderLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

Watchlist\_List.addItemListener(new ItemListener() {

public void itemStateChanged(ItemEvent ievt){

try {

Connection con5 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st4 = con5.createStatement();

ResultSet rs4 = st4.executeQuery("select \* from watchlist");

while (rs4.next()) {

if

(rs4.getString("stock\_symbol").equals(Watchlist\_List.getSelectedItem()))

break;

}

if (!rs4.isAfterLast()) {

JTF\_stock\_symbol.setText(rs4.getString("stock\_symbol"));

JTF\_user\_id.setText(rs4.getString("user\_id"));

}

}

catch (SQLException selectException) {

displaySQLErrors(selectException);

}

}

});

JB\_delete.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

try {

int a = JOptionPane.showConfirmDialog(pn,"Are you sure want to Delete:");

if(a == JOptionPane.YES\_OPTION){

String query = "DELETE FROM watchlist WHERE stock\_symbol = ?";

Connection con6 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

PreparedStatement stp3 = con6.prepareStatement(query);

stp3.setString(1, JTF\_stock\_symbol.getText());

int i = stp3.executeUpdate();

if(i>0){

JOptionPane.showMessageDialog(pn,"\nDeleted rows succesfully");

}

loaddetails();

}

}

catch(SQLException e){

displaySQLErrors(e);

}

}

});

}

});

view2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

JLabel view = new JLabel("Watchlist Details View");

view.setForeground(Color.BLACK);

JB\_view = new JButton("View");

Font myFont = new Font("Serif",Font.BOLD,25);

view.setFont((myFont));

pn1 = new JPanel();

pn2 = new JPanel();

pn1.add(view);

pn2.add(JB\_view);

pn.add(pn1);

pn.add(pn2);

pn.setBounds(500, 800, 300, 300);

pn.setLayout(new FlowLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

JB\_view.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

JFrame jfr = new JFrame("Watchlist details");

JTable jt;

DefaultTableModel model = new DefaultTableModel();

jt = new JTable(model);

model.addColumn("stock\_symbol");

model.addColumn("user\_id");

try {

Connection con7 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "vasavi", "abcd");

Statement st4 = con7.createStatement();

ResultSet rs5 = st4.executeQuery("select \* from watchlist");

while(rs5.next()){

model.addRow(new Object[]{rs5.getString("stock\_symbol"), rs5.getString("user\_id")});

}

}

catch(SQLException e){

displaySQLErrors(e);

}

jt.setEnabled(false);

jt.setBounds(30, 40, 300, 300);

JScrollPane jsp = new JScrollPane(jt);

jfr.add(jsp);

jfr.setSize(800, 400);

jfr.setVisible(true);

}

});

}

});

}

}

Historical\_data.java

import java.awt.BorderLayout;

import java.awt.Choice;

import java.awt.Color;

import java.awt.FlowLayout;

import java.awt.Font;

import java.awt.GridLayout;

import java.awt.List;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.ItemEvent;

import java.awt.event.ItemListener;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JMenuItem;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JScrollPane;

import javax.swing.JTable;

import javax.swing.JTextField;

import javax.swing.table.DefaultTableModel;

public class HistoricalData{

private JPanel pn, pn1, pn2, pn3;

private JFrame jf;

private JButton JB\_insert, JB\_modify, JB\_view, JB\_delete;

private JLabel JL\_stock\_symbol, JL\_day, JL\_opening\_price, JL\_closing\_price;

private JTextField JTF\_stock\_symbol, JTF\_day, JTF\_opening\_price, JTF\_closing\_price;

private JMenuItem insert2, update2, view2, delete2;

private List Historical\_Data\_List;

private Choice stock\_symbol;

public HistoricalData(JPanel pn, JFrame jf, JMenuItem insert2, JMenuItem update2, JMenuItem view2, JMenuItem

delete2){

try {

Class.forName("oracle.jdbc.driver.OracleDriver");

}

catch (Exception e) {

System.err.println("Unable to find and load driver");

System.exit(1);

}

this.jf = jf;

this.insert2 = insert2;

this.update2 = update2;

this.view2 = view2;

this.delete2 = delete2;

JL\_stock\_symbol = new JLabel("Stock Symbol : ");

JTF\_stock\_symbol = new JTextField(10);

JL\_day = new JLabel("Day: ");

JTF\_day = new JTextField(10);

JL\_opening\_price= new JLabel("Opening Price : ");

JTF\_opening\_price = new JTextField(10);

JL\_closing\_price = new JLabel("Closing Price: ");

JTF\_closing\_price= new JTextField(10);

this.pn=pn;

}

private void displaySQLErrors(SQLException e) {

JOptionPane.showMessageDialog(pn1,"\nSQLException: " + e.getMessage() + "\n"+"SQLState: " +

e.getSQLState() + "\n"+"VendorError: " + e.getErrorCode() + "\n");

}

public void load\_details(){

try{

stock\_symbol= new Choice();

stock\_symbol.removeAll();

Connection con1 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st1 = con1.createStatement();

ResultSet rs1 = st1.executeQuery("select \* from historical\_data");

while(rs1.next()) {

stock\_symbol.add(rs1.getString("Stock Id"));

}

}

catch(SQLException e) {

displaySQLErrors(e);

}

}

public void loaddetails(){

try{

Historical\_Data\_List = new List();

Historical\_Data\_List.removeAll();

Connection con2 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st2 = con2.createStatement();

ResultSet rs2 = st2.executeQuery("select \* from historical\_data");

while(rs2.next()) {

Historical\_Data\_List.add(rs2.getString("Stock Id"));

}

}

catch(SQLException e) {

displaySQLErrors(e);

}

}

public void buildGUI(){

insert2.addActionListener(new ActionListener(){

@Override

public void actionPerformed(ActionEvent aevt){

JB\_insert = new JButton("Submit");

JTF\_stock\_symbol.setText(null);

JTF\_day.setText(null);

JTF\_opening\_price.setText(null);

JTF\_closing\_price.setText(null);

loaddetails();

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

pn1 = new JPanel();

pn1.setLayout(new GridLayout(10,10));

pn1.add(JL\_stock\_symbol);

pn1.add(JTF\_stock\_symbol);

pn1.add(JL\_day);

pn1.add(JTF\_day);

pn1.add(JL\_opening\_price);

pn1.add(JTF\_opening\_price);

pn1.add(JL\_closing\_price);

pn1.add(JTF\_closing\_price);

pn3 = new JPanel(new FlowLayout());

pn3.add(JB\_insert);

pn1.setBounds(115, 80, 300, 250);

pn3.setBounds(200, 350, 75, 35);

pn2 = new JPanel(new FlowLayout());

Historical\_Data\_List = new List(10);

loaddetails();

pn2.add(Historical\_Data\_List);

pn2.setBounds(200, 350, 300, 180);

pn.add(pn1);

pn.add(pn3);

pn.add(pn2);

pn.setLayout(new BorderLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

JB\_insert.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

try{

Connection con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

String query = "INSERT INTO historical\_data (stock\_symbol, day, opening\_price,closing\_price) values(?, ?, ?, ?)";

PreparedStatement stp = con.prepareStatement(query);

stp.setString(1, JTF\_stock\_symbol.getText());

stp.setString(2, JTF\_day.getText());

stp.setString(3, JTF\_opening\_price.getText());

stp.setString(4, JTF\_closing\_price.getText());

int i = stp.executeUpdate();

con.close();

if(i > 0){

JOptionPane.showMessageDialog(pn,"\nInserted successfully");

}

}

catch(SQLException e) {

displaySQLErrors(e);

}

}

});

}

});

update2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

JB\_modify = new JButton("Modify");

JTF\_stock\_symbol.setText(null);

JTF\_day.setText(null);

JTF\_opening\_price.setText(null);

JTF\_closing\_price.setText(null);

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

pn1 = new JPanel();

pn1.setLayout(new GridLayout(10, 10));

pn1.add(JL\_stock\_symbol);

pn1.add(JTF\_stock\_symbol);

pn1.add(JL\_day);

pn1.add(JTF\_day);

pn1.add(JL\_opening\_price);

pn1.add(JTF\_opening\_price);

pn1.add(JL\_closing\_price);

pn1.add(JTF\_closing\_price);

pn3 = new JPanel(new FlowLayout());

pn3.add(JB\_modify);

pn1.setBounds(115, 80, 300, 250);

pn3.setBounds(200, 350, 75, 35);

pn2 =new JPanel(new FlowLayout());

Historical\_Data\_List = new List(10);

loaddetails();

pn2.add(Historical\_Data\_List);

pn2.setBounds(200, 350, 300, 180);

pn.add(pn1);

pn.add(pn3);

pn.add(pn2);

pn.setLayout(new BorderLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

Historical\_Data\_List.addItemListener(new ItemListener() {

public void itemStateChanged(ItemEvent ievt){

try {

Connection con3 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st3 = con3.createStatement();

ResultSet rs3 = st3.executeQuery("select \* from historical\_data");

while (rs3.next()) {

if

(rs3.getString("stock\_symbol").equals(Historical\_Data\_List.getSelectedItem()))

break;

}

if (!rs3.isAfterLast()) {

JTF\_stock\_symbol.setText(rs3.getString("stock\_symbol"));

JTF\_day.setText(rs3.getString("day"));

JTF\_opening\_price.setText(rs3.getString("opening\_price"));

JTF\_closing\_price.setText(rs3.getString("closing\_price"));

}

}

catch (SQLException selectException) {

displaySQLErrors(selectException);

}

}

});

JB\_modify.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

try{

int a = JOptionPane.showConfirmDialog(pn, "Are you sure want to update:");

if(a == JOptionPane.YES\_OPTION){

String query = "update historical\_data set day = ?, opening\_price = ?, closing\_price = ? where stock\_symbol= ?";

Connection con4 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

PreparedStatement stp2 = con4.prepareStatement(query);

stp2.setString(1, JTF\_stock\_symbol.getText());

stp2.setString(2, JTF\_day.getText());

stp2.setString(3, JTF\_opening\_price.getText());

stp2.setString(4, JTF\_closing\_price.getText());

int i = stp2.executeUpdate();

if(i>0){

JOptionPane.showMessageDialog(pn,"\nUpdated rows succesfully");

}

loaddetails();

}

}

catch(SQLException e){

displaySQLErrors(e);

}

}

});

}

});

delete2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

JB\_delete = new JButton("Delete");

JTF\_stock\_symbol.setText(null);

JTF\_day.setText(null);

JTF\_opening\_price.setText(null);

JTF\_closing\_price.setText(null);

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

pn1 = new JPanel();

pn1.setLayout(new GridLayout(10, 10));

pn1.add(JL\_stock\_symbol);

pn1.add(JTF\_stock\_symbol);

pn1.add(JL\_day);

pn1.add(JTF\_day);

pn1.add(JL\_opening\_price);

pn1.add(JTF\_opening\_price);

pn1.add(JL\_closing\_price);

pn1.add(JTF\_closing\_price);

pn3 = new JPanel(new FlowLayout());

pn3.add(JB\_delete);

pn1.setBounds(115, 80, 300, 250);

pn3.setBounds(200, 350, 75, 35);

pn2 = new JPanel(new FlowLayout());

Historical\_Data\_List = new List(10);

loaddetails();

pn2.add(Historical\_Data\_List);

pn2.setBounds(200, 350, 300, 200);

pn.add(pn1);

pn.add(pn3);

pn.add(pn2);

pn.setLayout(new BorderLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

Historical\_Data\_List.addItemListener(new ItemListener() {

public void itemStateChanged(ItemEvent ievt){

try {

Connection con5 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st4 = con5.createStatement();

ResultSet rs4 = st4.executeQuery("select \* from historical\_data");

while (rs4.next()) {

if

(rs4.getString("stock\_symbol").equals(Historical\_Data\_List.getSelectedItem()))

break;

}

if (!rs4.isAfterLast()) {

JTF\_stock\_symbol.setText(rs4.getString("stock\_symbol"));

JTF\_day.setText(rs4.getString("day"));

JTF\_opening\_price.setText(rs4.getString("opening\_price"));

JTF\_closing\_price.setText(rs4.getString("closing\_price"));

}

}

catch (SQLException selectException) {

displaySQLErrors(selectException);

}

}

});

JB\_delete.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

try {

int a = JOptionPane.showConfirmDialog(pn,"Are you sure want to Delete:");

if(a == JOptionPane.YES\_OPTION){

String query = "DELETE FROM historical\_data WHERE stock\_symbol = ?";

Connection con6 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

PreparedStatement stp3 = con6.prepareStatement(query);

stp3.setString(1, JTF\_stock\_symbol.getText());

int i = stp3.executeUpdate();

if(i>0){

JOptionPane.showMessageDialog(pn,"\nDeleted rows succesfully");

}

loaddetails();

}

}

catch(SQLException e){

displaySQLErrors(e);

}

}

});

}

});

view2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

JLabel view = new JLabel("Stock Details View");

view.setForeground(Color.BLACK);

JB\_view = new JButton("View");

Font myFont = new Font("Serif",Font.BOLD,25);

view.setFont((myFont));

pn1 = new JPanel();

pn2 = new JPanel();

pn1.add(view);

pn2.add(JB\_view);

pn.add(pn1);

pn.add(pn2);

pn.setBounds(500, 800, 300, 300);

pn.setLayout(new FlowLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

JB\_view.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

JFrame jfr = new JFrame("historical\_data details");

JTable jt;

DefaultTableModel model = new DefaultTableModel();

jt = new JTable(model);

model.addColumn("stock\_symbol");

model.addColumn("day");

model.addColumn("opening\_price");

model.addColumn("closing\_price");

try {

Connection con7 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "vasavi", "abcd");

Statement st4 = con7.createStatement();

ResultSet rs5 = st4.executeQuery("select \* from historical\_data");

while(rs5.next()){

model.addRow(new Object[]{rs5.getString("stock\_symbol"), rs5.getString("day"), rs5.getString("opening\_price"), rs5.getString("closing\_price")});

}

}

catch(SQLException e){

displaySQLErrors(e);

}

jt.setEnabled(false);

jt.setBounds(30, 40, 300, 300);

JScrollPane jsp = new JScrollPane(jt);

jfr.add(jsp);

jfr.setSize(800, 400);

jfr.setVisible(true);

}

});

}

});

}

}

Prediction.java

import java.awt.BorderLayout;

import java.awt.Choice;

import java.awt.Color;

import java.awt.FlowLayout;

import java.awt.Font;

import java.awt.GridLayout;

import java.awt.List;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.ItemEvent;

import java.awt.event.ItemListener;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JMenuItem;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JScrollPane;

import javax.swing.JTable;

import javax.swing.JTextField;

import javax.swing.table.DefaultTableModel;

public class prediction{

private JPanel pn, pn1, pn2, pn3;

private JFrame jf;

private JButton JB\_insert, JB\_modify, JB\_view, JB\_delete;

private JLabel JL\_stock\_symbol, JL\_day, JL\_predicted\_price;

private JTextField JTF\_stock\_symbol, JTF\_day, JTF\_predicted\_price;

private JMenuItem insert2, update2, view2, delete2;

private List prediction\_list;

private Choice stock\_symbol;

public prediction(JPanel pn, JFrame jf, JMenuItem insert2, JMenuItem update2, JMenuItem view2, JMenuItem

delete2){

try {

Class.forName("oracle.jdbc.driver.OracleDriver");

}

catch (Exception e) {

System.err.println("Unable to find and load driver");

System.exit(1);

}

this.jf = jf;

this.insert2 = insert2;

this.update2 = update2;

this.view2 = view2;

this.delete2 = delete2;

JL\_stock\_symbol = new JLabel("Stock Symbol : ");

JTF\_stock\_symbol = new JTextField(10);

JL\_day = new JLabel("Stock Day: ");

JTF\_day = new JTextField(10);

JL\_predicted\_price= new JLabel("Predicted Price : ");

JTF\_predicted\_price = new JTextField(10);

this.pn=pn;

}

private void displaySQLErrors(SQLException e) {

JOptionPane.showMessageDialog(pn1,"\nSQLException: " + e.getMessage() + "\n"+"SQLState: " +

e.getSQLState() + "\n"+"VendorError: " + e.getErrorCode() + "\n");

}

public void load\_details(){

try{

stock\_symbol= new Choice();

stock\_symbol.removeAll();

Connection con1 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st1 = con1.createStatement();

ResultSet rs1 = st1.executeQuery("select \* from prediction");

while(rs1.next()) {

stock\_symbol.add(rs1.getString("stock\_symbol"));

}

}

catch(SQLException e) {

displaySQLErrors(e);

}

}

public void loaddetails(){

try{

prediction\_list = new List();

prediction\_list.removeAll();

Connection con2 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st2 = con2.createStatement();

ResultSet rs2 = st2.executeQuery("select \* from prediction");

while(rs2.next()) {

prediction\_list.add(rs2.getString("stock\_symbol"));

}

}

catch(SQLException e) {

displaySQLErrors(e);

}

}

public void buildGUI(){

insert2.addActionListener(new ActionListener(){

@Override

public void actionPerformed(ActionEvent aevt){

JB\_insert = new JButton("Submit");

JTF\_stock\_symbol.setText(null);

JTF\_day.setText(null);

JTF\_predicted\_price.setText(null);

loaddetails();

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

pn1 = new JPanel();

pn1.setLayout(new GridLayout(10,10));

pn1.add(JL\_stock\_symbol);

pn1.add(JTF\_stock\_symbol);

pn1.add(JL\_day);

pn1.add(JTF\_day);

pn1.add(JL\_predicted\_price);

pn1.add(JTF\_predicted\_price);

pn3 = new JPanel(new FlowLayout());

pn3.add(JB\_insert);

pn1.setBounds(115, 80, 300, 250);

pn3.setBounds(200, 350, 75, 35);

pn2 = new JPanel(new FlowLayout());

prediction\_list = new List(10);

loaddetails();

pn2.add(prediction\_list);

pn2.setBounds(200, 350, 300, 180);

pn.add(pn1);

pn.add(pn3);

pn.add(pn2);

pn.setLayout(new BorderLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

JB\_insert.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

try{

Connection con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

String query = "INSERT INTO prediction(stock\_symbol, day, predicted\_price) values(?, ?, ?)";

PreparedStatement stp = con.prepareStatement(query);

stp.setString(1, JTF\_stock\_symbol.getText());

stp.setString(2, JTF\_day.getText());

stp.setString(3, JTF\_predicted\_price.getText());

int i = stp.executeUpdate();

con.close();

if(i > 0){

JOptionPane.showMessageDialog(pn,"\nInserted successfully");

}

}

catch(SQLException e) {

displaySQLErrors(e);

}

}

});

}

});

update2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

JB\_modify = new JButton("Modify");

JTF\_stock\_symbol.setText(null);

JTF\_day.setText(null);

JTF\_predicted\_price.setText(null);

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

pn1 = new JPanel();

pn1.setLayout(new GridLayout(10, 10));

pn1.add(JL\_stock\_symbol);

pn1.add(JTF\_stock\_symbol);

pn1.add(JL\_day);

pn1.add(JTF\_day);

pn1.add(JL\_predicted\_price);

pn1.add(JTF\_predicted\_price);

pn3 = new JPanel(new FlowLayout());

pn3.add(JB\_modify);

pn1.setBounds(115, 80, 300, 250);

pn3.setBounds(200, 350, 75, 35);

pn2 =new JPanel(new FlowLayout());

prediction\_list = new List(10);

loaddetails();

pn2.add(prediction\_list);

pn2.setBounds(200, 350, 300, 180);

pn.add(pn1);

pn.add(pn3);

pn.add(pn2);

pn.setLayout(new BorderLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

prediction\_list.addItemListener(new ItemListener() {

public void itemStateChanged(ItemEvent ievt){

try {

Connection con3 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st3 = con3.createStatement();

ResultSet rs3 = st3.executeQuery("select \* from prediction");

while (rs3.next()) {

if

(rs3.getString("stock\_symbol").equals(prediction\_list.getSelectedItem()))

break;

}

if (!rs3.isAfterLast()) {

JTF\_stock\_symbol.setText(rs3.getString("stock\_symbol"));

JTF\_day.setText(rs3.getString("day"));

JTF\_predicted\_price.setText(rs3.getString("predicted\_price"));

}

}

catch (SQLException selectException) {

displaySQLErrors(selectException);

}

}

});

JB\_modify.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

try{

int a = JOptionPane.showConfirmDialog(pn, "Are you sure want to update:");

if(a == JOptionPane.YES\_OPTION){

String query = "update Stock\_details set day = ?, predicted\_price = ? where stock\_symbol= ?";

Connection con4 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

PreparedStatement stp2 = con4.prepareStatement(query);

stp2.setString(1, JTF\_stock\_symbol.getText());

stp2.setString(2, JTF\_day.getText());

stp2.setString(3, JTF\_predicted\_price.getText());

int i = stp2.executeUpdate();

if(i>0){

JOptionPane.showMessageDialog(pn,"\nUpdated rows succesfully");

}

loaddetails();

}

}

catch(SQLException e){

displaySQLErrors(e);

}

}

});

}

});

delete2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

JB\_delete = new JButton("Delete");

JTF\_stock\_symbol.setText(null);

JTF\_day.setText(null);

JTF\_predicted\_price.setText(null);

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

pn1 = new JPanel();

pn1.setLayout(new GridLayout(10, 10));

pn1.add(JL\_stock\_symbol);

pn1.add(JTF\_stock\_symbol);

pn1.add(JL\_day);

pn1.add(JTF\_day);

pn1.add(JL\_predicted\_price);

pn1.add(JTF\_predicted\_price);

pn3 = new JPanel(new FlowLayout());

pn3.add(JB\_delete);

pn1.setBounds(115, 80, 300, 250);

pn3.setBounds(200, 350, 75, 35);

pn2 = new JPanel(new FlowLayout());

prediction\_list = new List(10);

loaddetails();

pn2.add(prediction\_list);

pn2.setBounds(200, 350, 300, 200);

pn.add(pn1);

pn.add(pn3);

pn.add(pn2);

pn.setLayout(new BorderLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

prediction\_list.addItemListener(new ItemListener() {

public void itemStateChanged(ItemEvent ievt){

try {

Connection con5 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

Statement st4 = con5.createStatement();

ResultSet rs4 = st4.executeQuery("select \* from prediction");

while (rs4.next()) {

if

(rs4.getString("stock\_symbol").equals(prediction\_list.getSelectedItem()))

break;

}

if (!rs4.isAfterLast()) {

JTF\_stock\_symbol.setText(rs4.getString("stock\_symbol"));

JTF\_day.setText(rs4.getString("day"));

JTF\_predicted\_price.setText(rs4.getString("predicted\_price"));

}

}

catch (SQLException selectException) {

displaySQLErrors(selectException);

}

}

});

JB\_delete.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

try {

int a = JOptionPane.showConfirmDialog(pn,"Are you sure want to Delete:");

if(a == JOptionPane.YES\_OPTION){

String query = "DELETE FROM prediction WHERE stock\_symbol = ?";

Connection con6 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","vasavi", "abcd");

PreparedStatement stp3 = con6.prepareStatement(query);

stp3.setString(1, JTF\_stock\_symbol.getText());

int i = stp3.executeUpdate();

if(i>0){

JOptionPane.showMessageDialog(pn,"\nDeleted rows succesfully");

}

loaddetails();

}

}

catch(SQLException e){

displaySQLErrors(e);

}

}

});

}

});

view2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

pn.removeAll();

jf.invalidate();

jf.validate();

jf.repaint();

JLabel view = new JLabel("Prediction Details View");

view.setForeground(Color.BLACK);

JB\_view = new JButton("View");

Font myFont = new Font("Serif",Font.BOLD,25);

view.setFont((myFont));

pn1 = new JPanel();

pn2 = new JPanel();

pn1.add(view);

pn2.add(JB\_view);

pn.add(pn1);

pn.add(pn2);

pn.setBounds(500, 800, 300, 300);

pn.setLayout(new FlowLayout());

jf.add(pn);

jf.setSize(800, 800);

jf.validate();

JB\_view.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent aevt){

JFrame jfr = new JFrame("prediction details");

JTable jt;

DefaultTableModel model = new DefaultTableModel();

jt = new JTable(model);

model.addColumn("stock\_symbol");

model.addColumn("day");

model.addColumn("predicted\_price");

try {

Connection con7 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "vasavi", "abcd");

Statement st4 = con7.createStatement();

ResultSet rs5 = st4.executeQuery("select \* from prediction");

while(rs5.next()){

model.addRow(new Object[]{rs5.getString("stock\_symbol"), rs5.getString("day"), rs5.getString("predicted\_price")});

}

}

catch(SQLException e){

displaySQLErrors(e);

}

jt.setEnabled(false);

jt.setBounds(30, 40, 300, 300);

JScrollPane jsp = new JScrollPane(jt);

jfr.add(jsp);

jfr.setSize(800, 400);

jfr.setVisible(true);

}

});

}

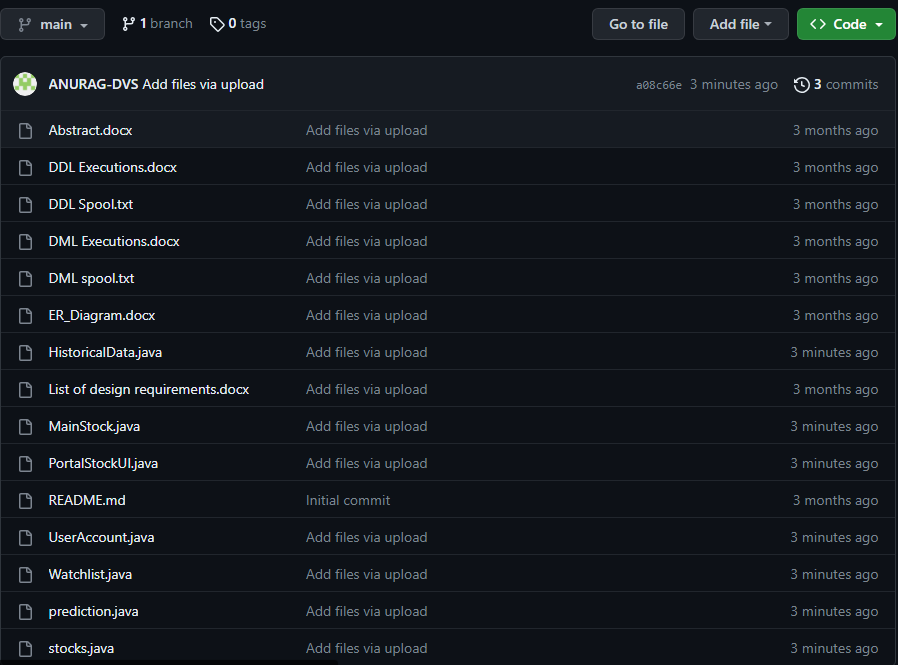
});

}

}

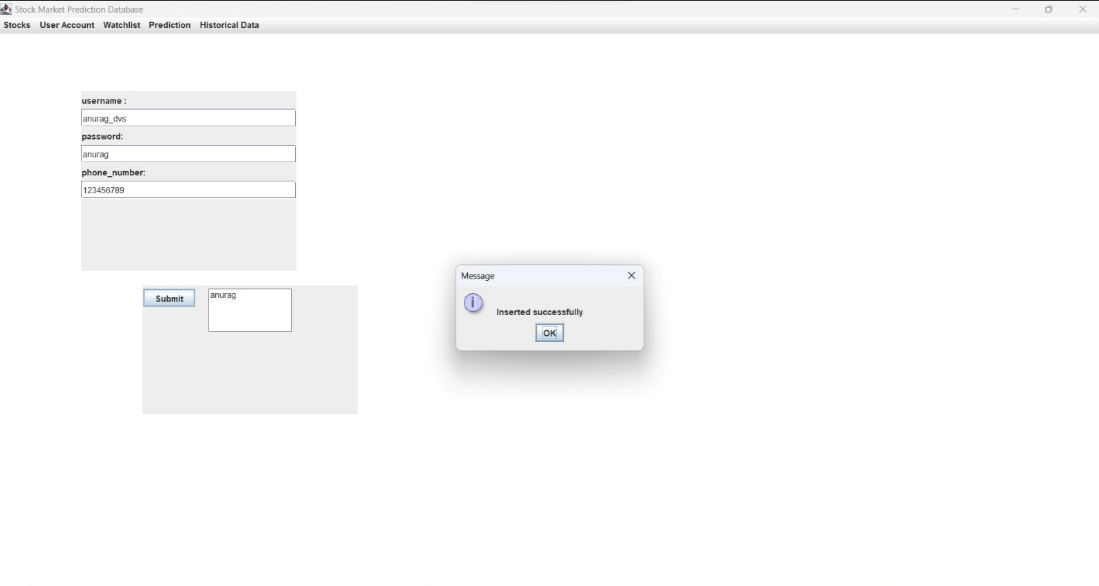
**GitHub Links and Folder Structure**

**Link: <** [**https://github.com/ANURAG-DVS/DBMS\_Project**](https://github.com/ANURAG-DVS/DBMS_Project)**>**

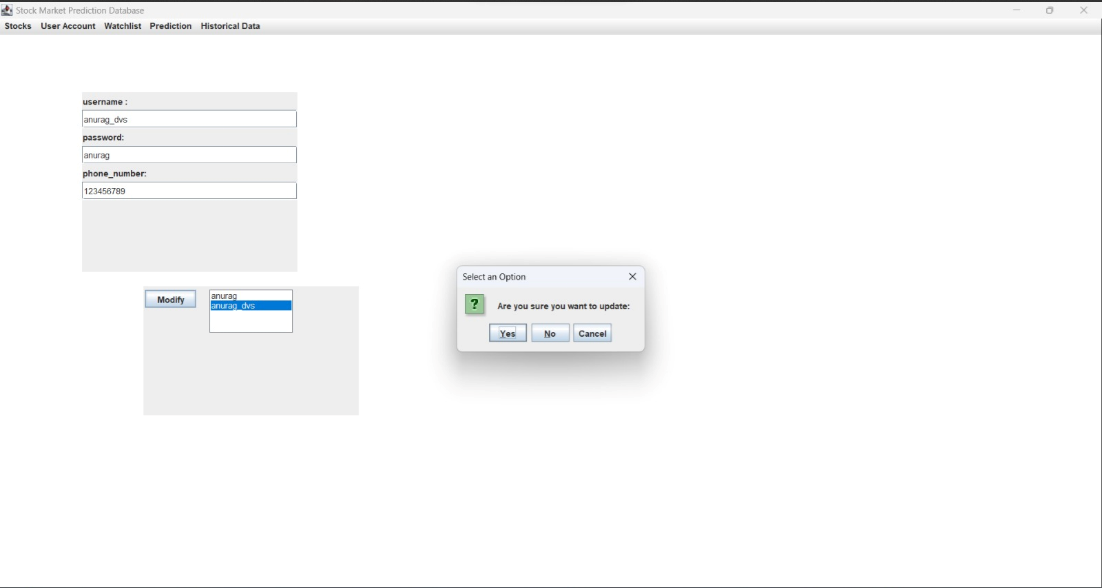
****

**TESTING**

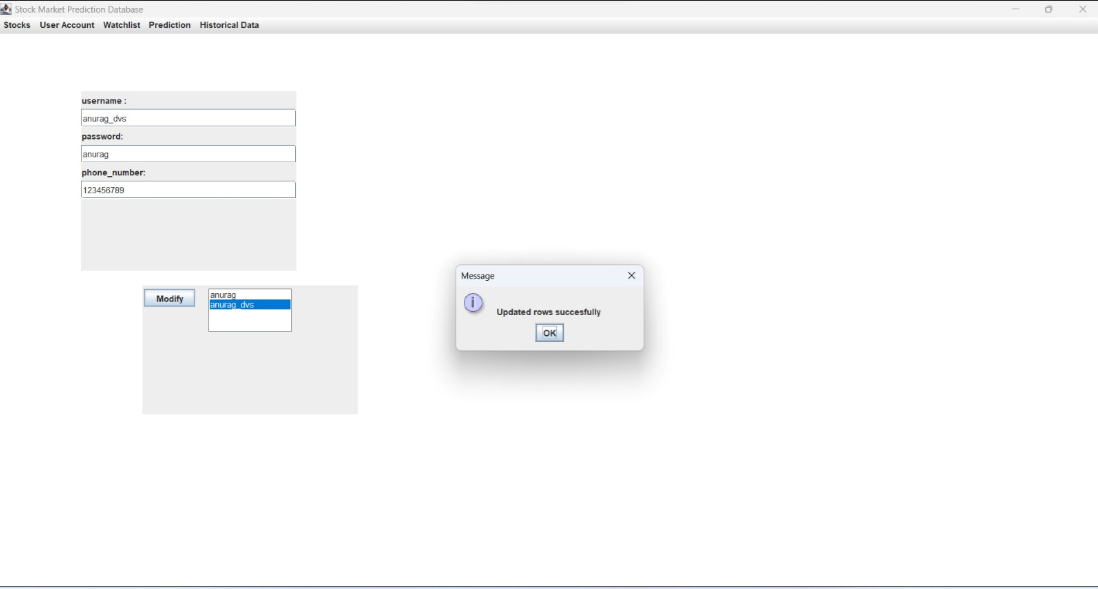
**1.**

****

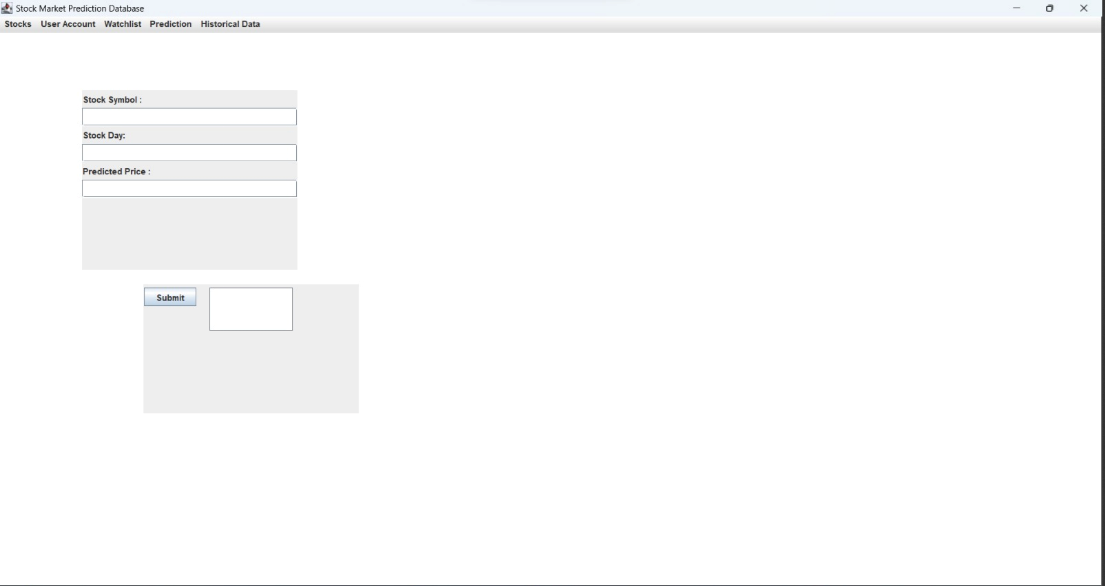
**2.**



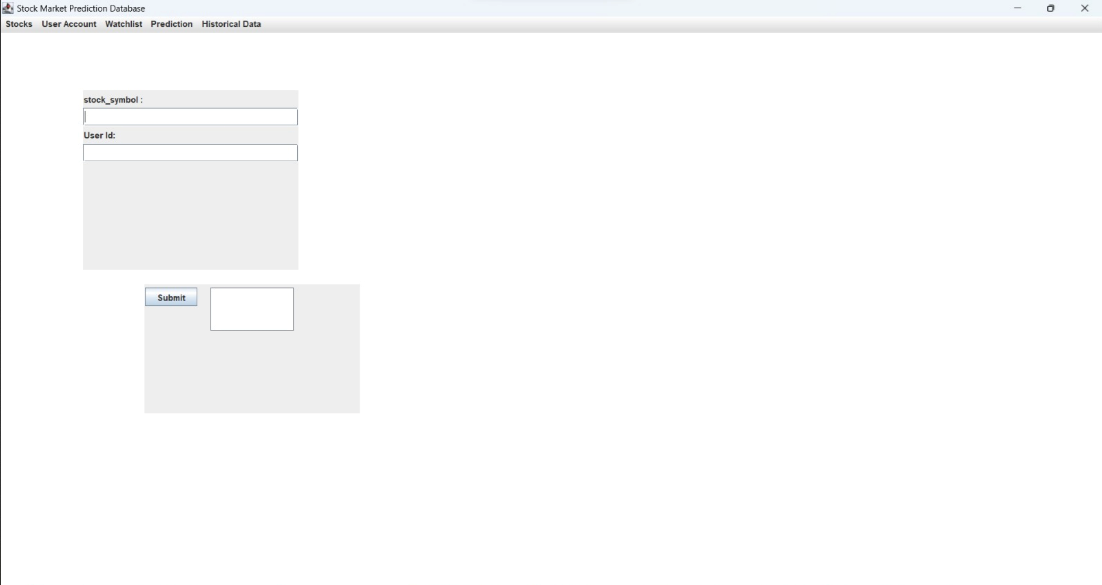
**3.**

****

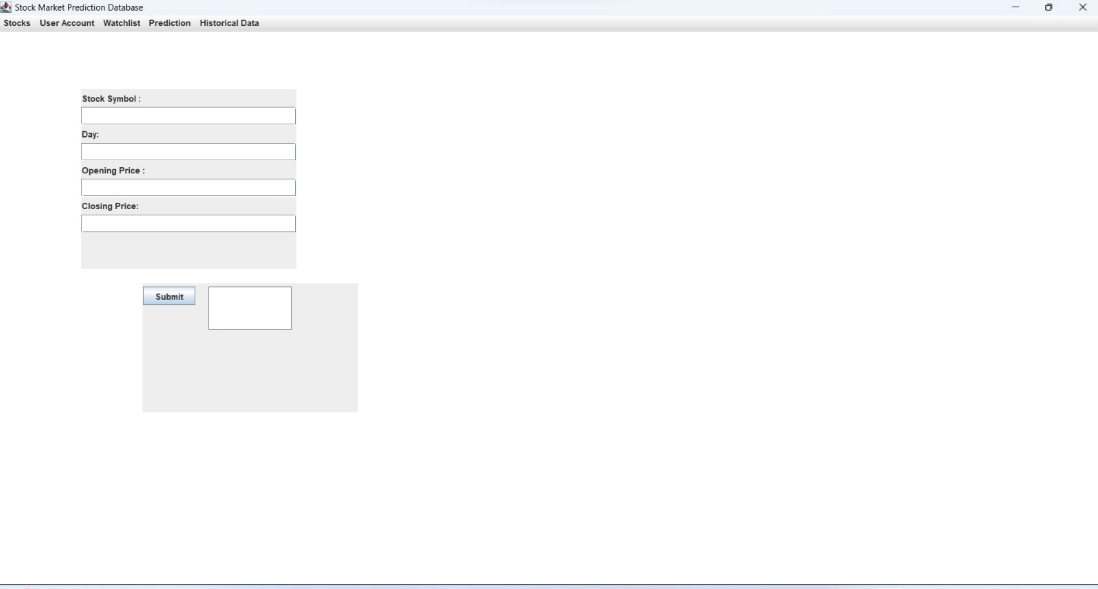
**4.**

****

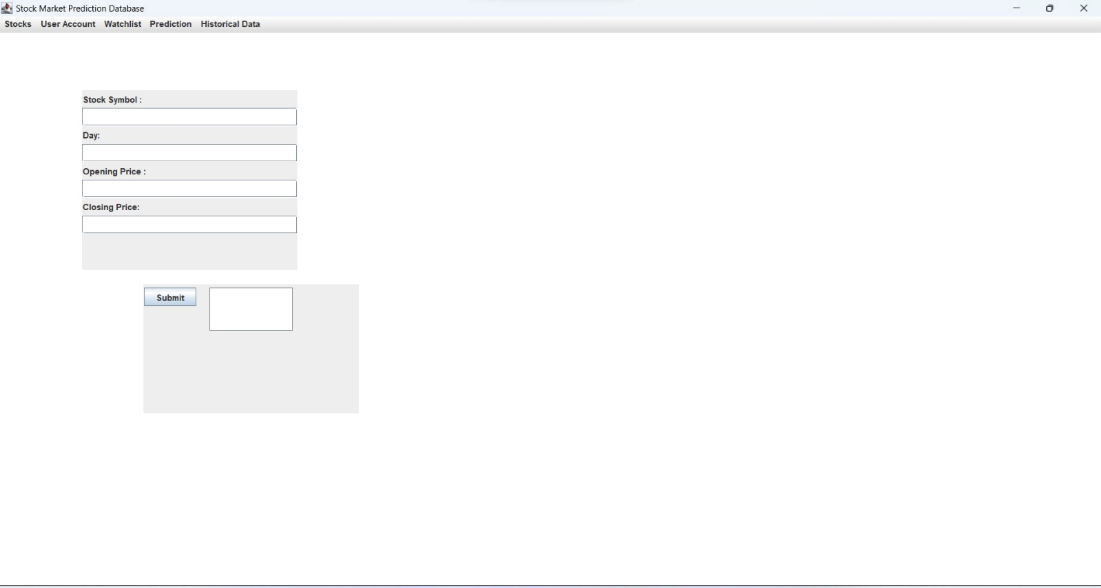
**5.**

****

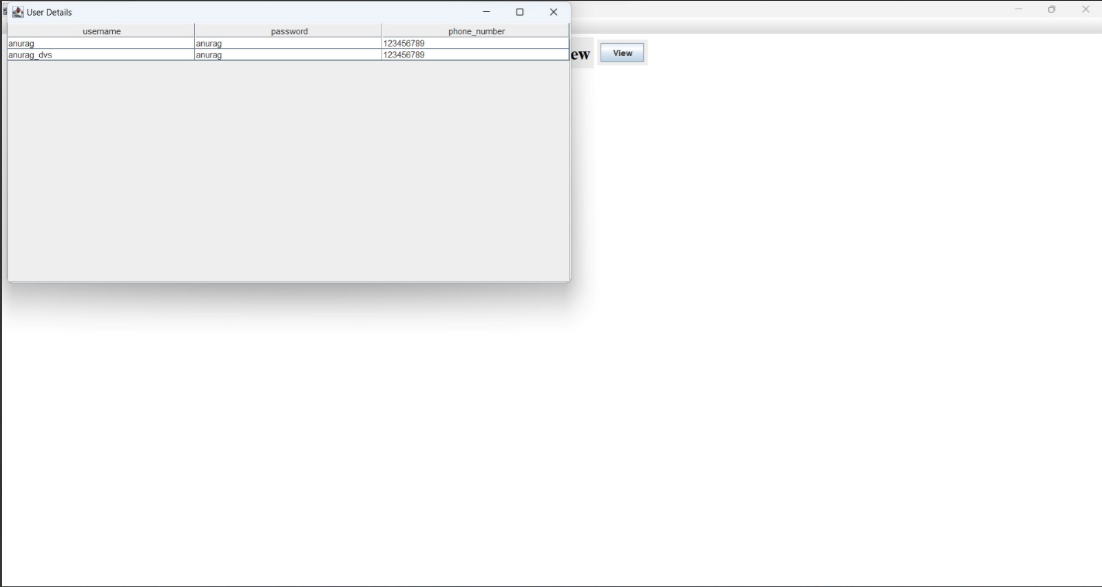
**6.**

****

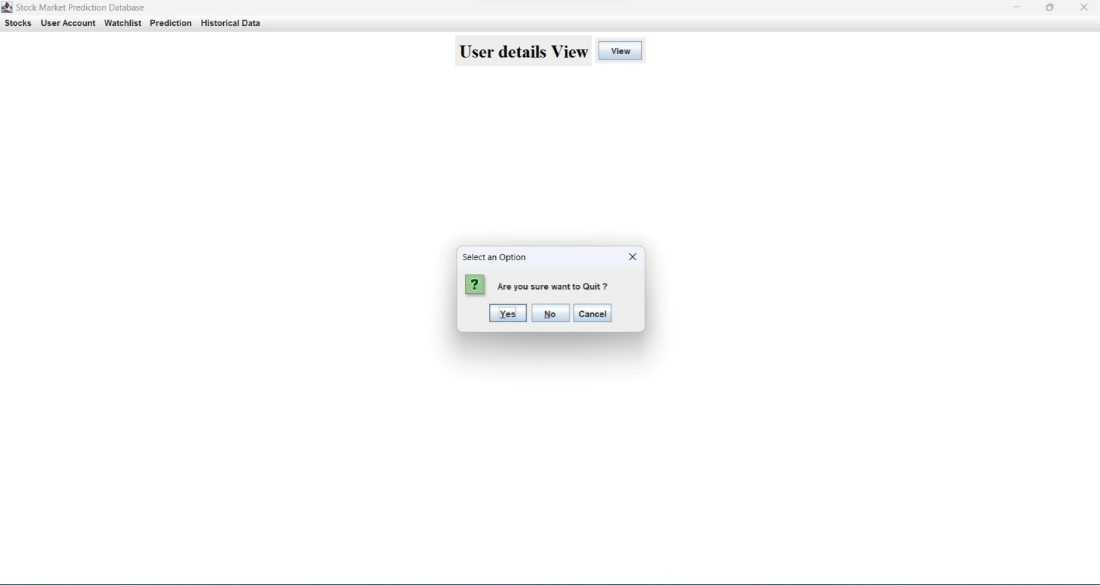
**7.**

****

**8.**



**9.**



**RESULTS**

I have successfully completed the mini-project ***“STOCK MARKET PREDICTION”*** .

**REFERENCES**

* https://docs.oracle.com/javase/7/docs/api/
* [https://www.javatpoint.com/java-swing](http://www.javatpoint.com/java-swing)
* https://stackoverflow.com/