# VIRTUAL ASSESSMENT MANAGEMENT SYSTEM

# - SQL CONNECTIVITY USING JDBC

*A*

*Report*

*Submitted in partial fulfillment of the Requirements for the award of the Degree of*

# BACHELOR OF ENGINEERING

IN

# INFORMATION TECHNOLOGY

By

K.Rushitha <1602-189-737-090>

Under the guidance of Ms B. Leelavathy



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

**BONAFIDE CERTIFICATE**

This is to certify that this project report titled ‘**VIRTUAL ASSESSMENT MANAGEMENT SYSTEM**’ is a project work of Ms. K.Rushitha bearing roll no. 1602-19-737-090 who carried out the project under my supervision in the IV semester for the academic year 2020- 2021.

Signature

Signature

Internal Examiner

**ABSTRACT:**

***Virtual Assessment Management System*** is a web application that establishes a network between the server and the students. Here admin can be an institute (or server itself) enter on the site the questions they want in the exam. These questions are displayed as a test to the eligible students. The answers entered by the students are then evaluated and their score is calculated and saved. This score then can be accessed by the institutes or known to students if they want to display them or determine the passes students or to evaluate their performance and also one such special feature is that it records the attendance of the students who attended the exam.

***Virtual Assessment*** provides the platform where its participates in, and it involves the users also in any tests conducted. Questions are posted by the site, and also users of the site. The site requires an institute to register before posting the questions.

The site has an administrator who keeps an eye on the overall functioning of the system. The site gets revenue by charging the institutes each time they want to conduct the exam. The system entitled “EXAM POINT” is application software, which aims at providing services to the institutes and providing them with an option of selecting the eligible students by themselves. It is developed by using J2EE technology and related database.

**INTRODUCTION**

# REQUIREMENTS

**Admin:**

* Admin Password : Apassword - VARCHAR2(20)
* Admin User Name : Aname - VARCHAR(20)

User

* UserName : username - VARCHAR(5)
* Password : password - VARCHAR(5)
* Registration ID : Reg Id - VARCHAR(5)

**User\_Details:**

* Student Name : Sname - VARCHAR2(20)
* Student Registeration number : Sregno - VARCHAR2(20)
* Student Login ID : Sid - VARCHAR(5)
* Student’s Login Password : Spassword - VARCHAR(10)
* Last Name of the User : Lname - VARCHAR(10)
* First Name Of The Usr : Fname - VARCHAR(10)

**Course\_Info:**

* Course ID : Cid - VARCHAR2(10)
* Course Name : Cname - VARCHAR2(10)

**Questions:**

* Question ID : Qid - VARCHAR2(10)
* Question Name : Qname - VARCHAR2(200)
* Answer of the question ID : Ans - VARCHAR2(100)
* Option 1 : Opt1 - VARCHAR2(5)
* Option 2 : Opt2 - VARCHAR2(5)
* Option 3 : Opt3 - VARCHAR2(5)
* Option 4 : Opt4 - VARCHAR2(5)

**Answers**:

* Course ID : cid - VARCHAR (10)
* Question Name : qname - VARCHAR2(100)
* Correct Option : crct\_opt - VARCHAR2(5)

**AIM AND PRIORITY OF THE PROJECT**

To create a Java GUI based information retrieval system of Google queries which takes the values like: user id, user name etc. from the person who posts the query/answers the query/rates the query, it also stores information about the query and its feedback. These values are to be updated in the database using JDBC connectivity .

**ARCHITECTURE AND TECHNOLOGY**

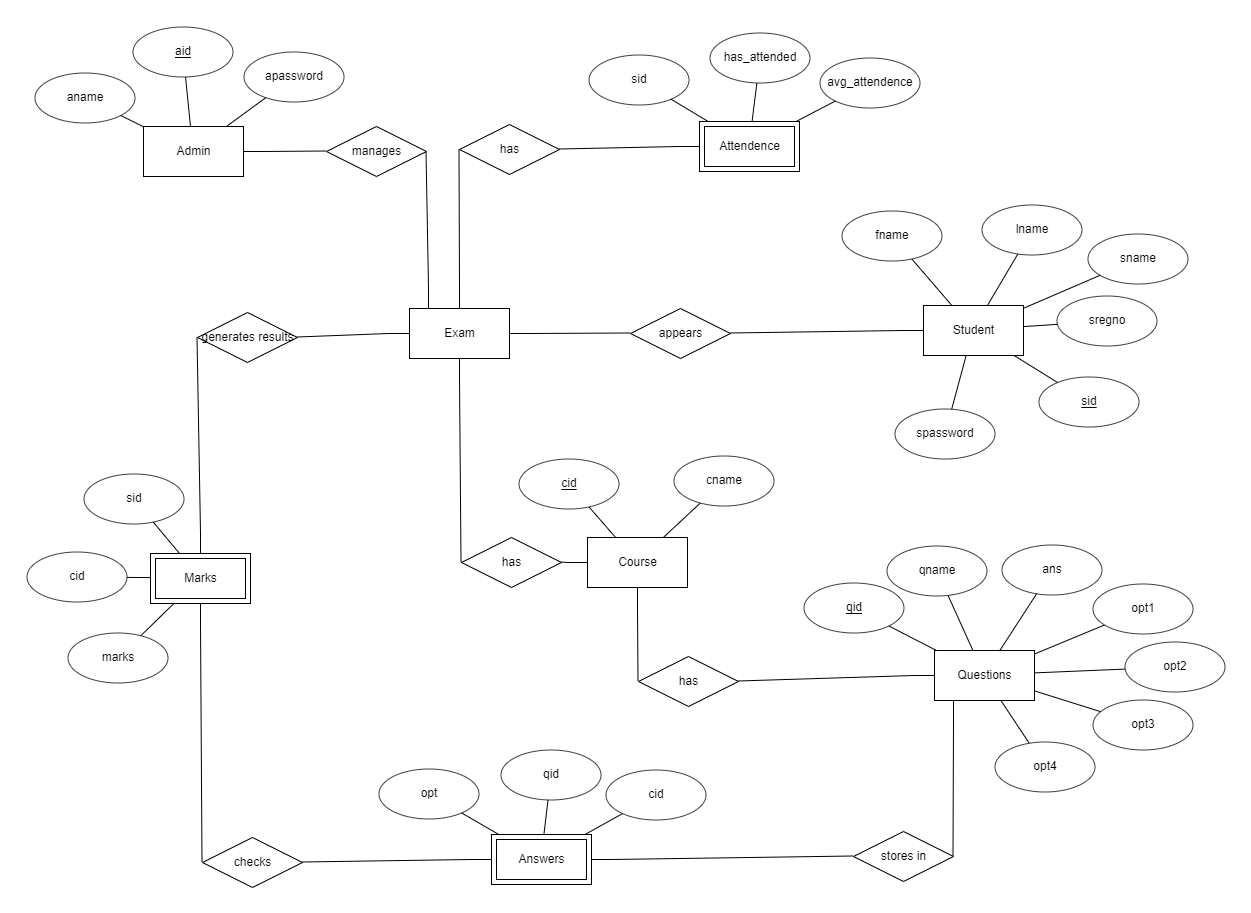
* **Software used:**
* Java Eclipse, Oracle 11g Database, Java SE version 13, SQL\*Plus.

* **Java AWT:**
* **Java AWT** (Abstract Window Toolkit) is an API to develop GUI or window-based applications in java.
* Java AWT components are platform-dependent i.e. components are displayed according to the view of operating system. AWT is heavyweight i.e. its components are using the resources of OS.
* The java.awt package provides classes for AWT API such as TextField, Label, TextArea, RadioButton, CheckBox, Choice, List etc.

* **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

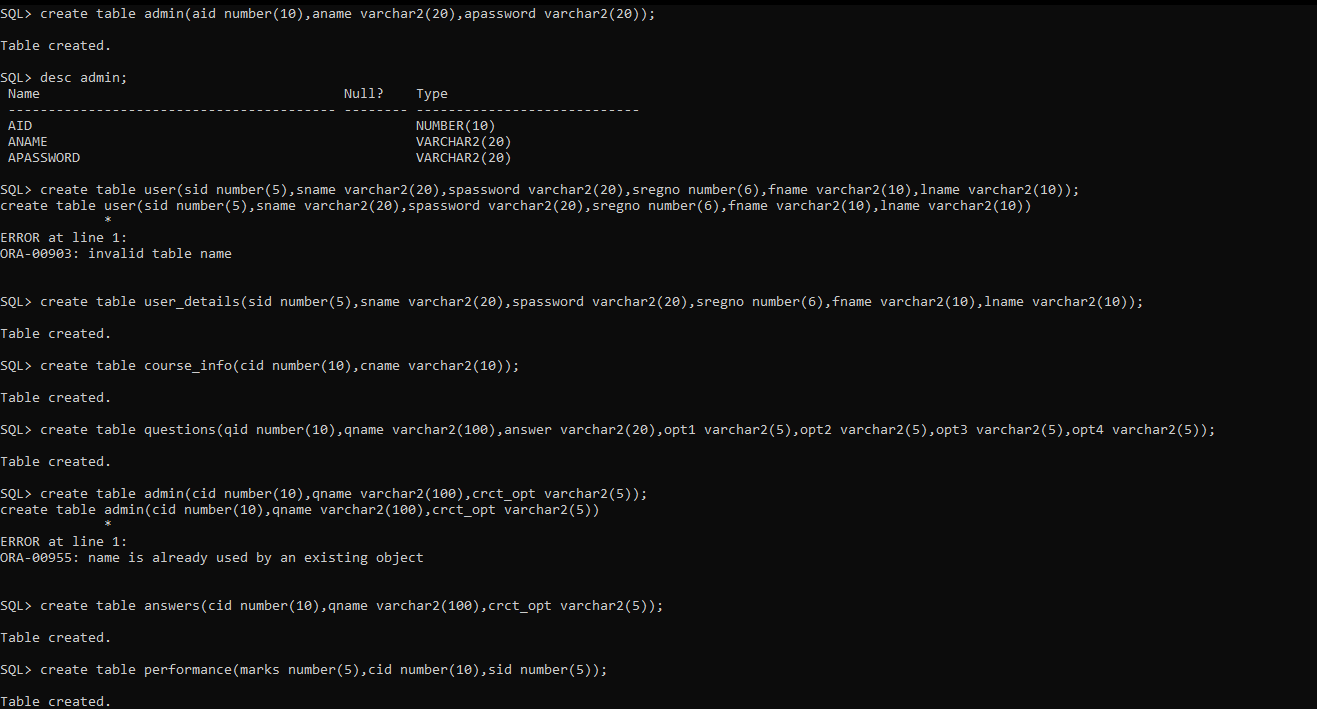
**DESIGN**

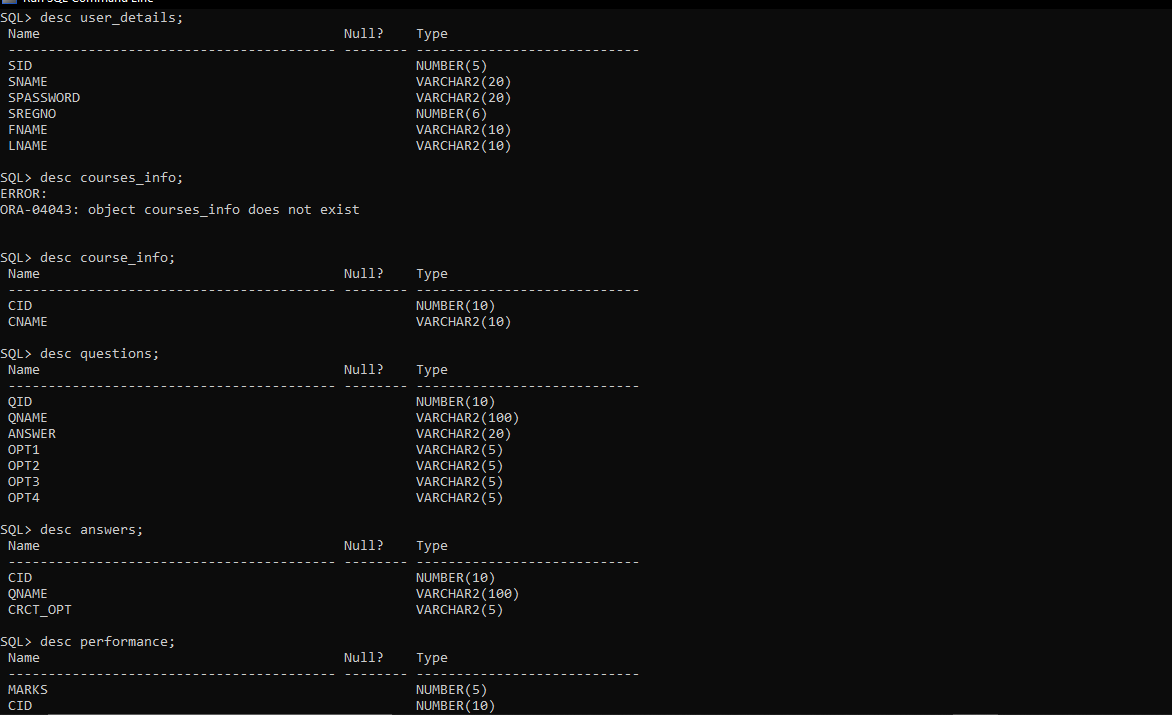
**Entity relationship diagram**

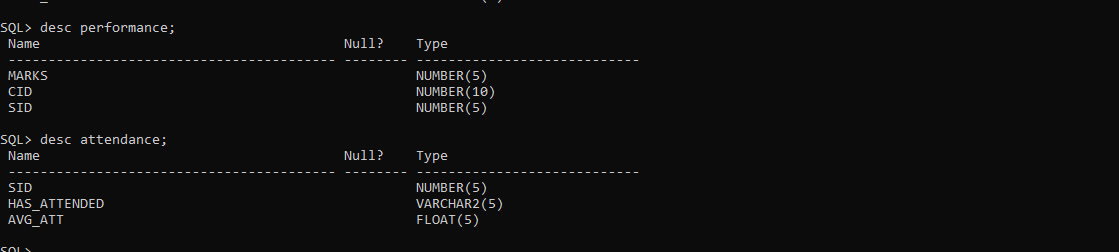


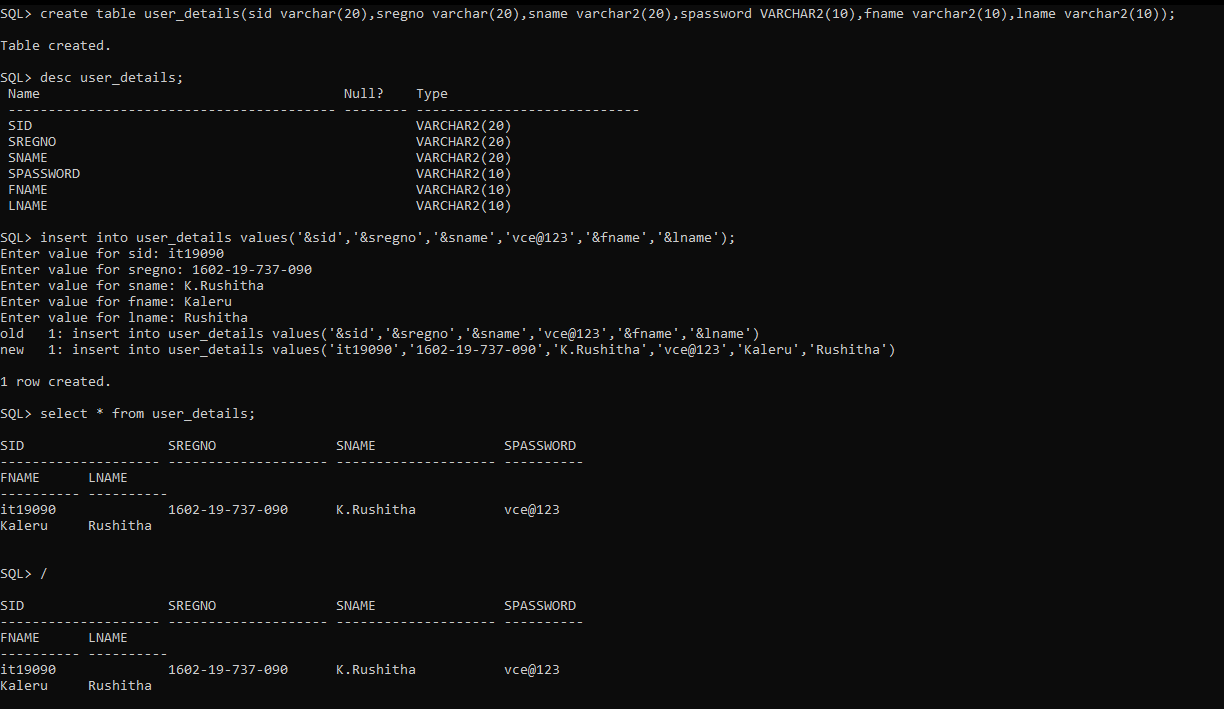
DML Operations

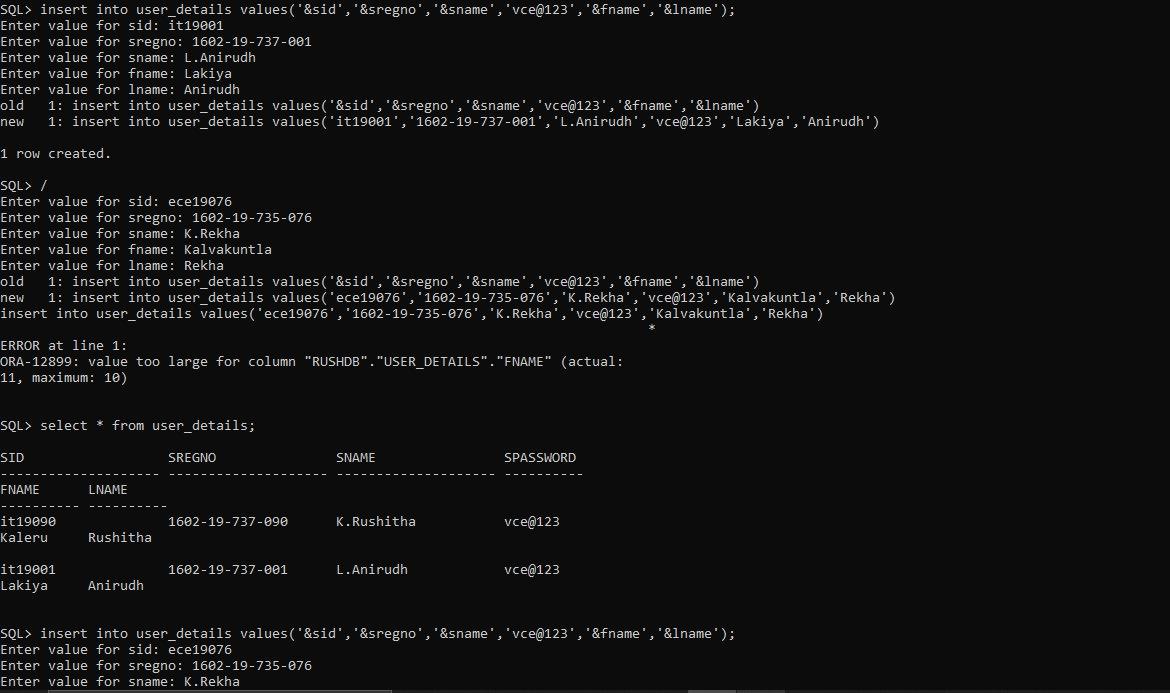
Tables ,Commands & Results

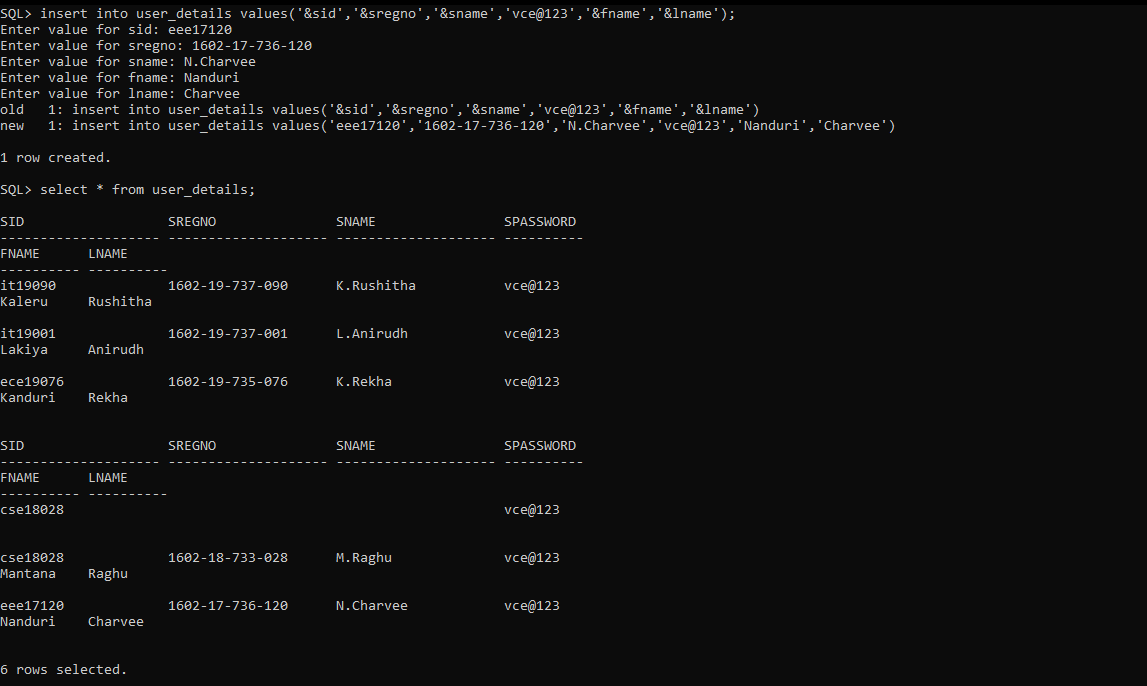


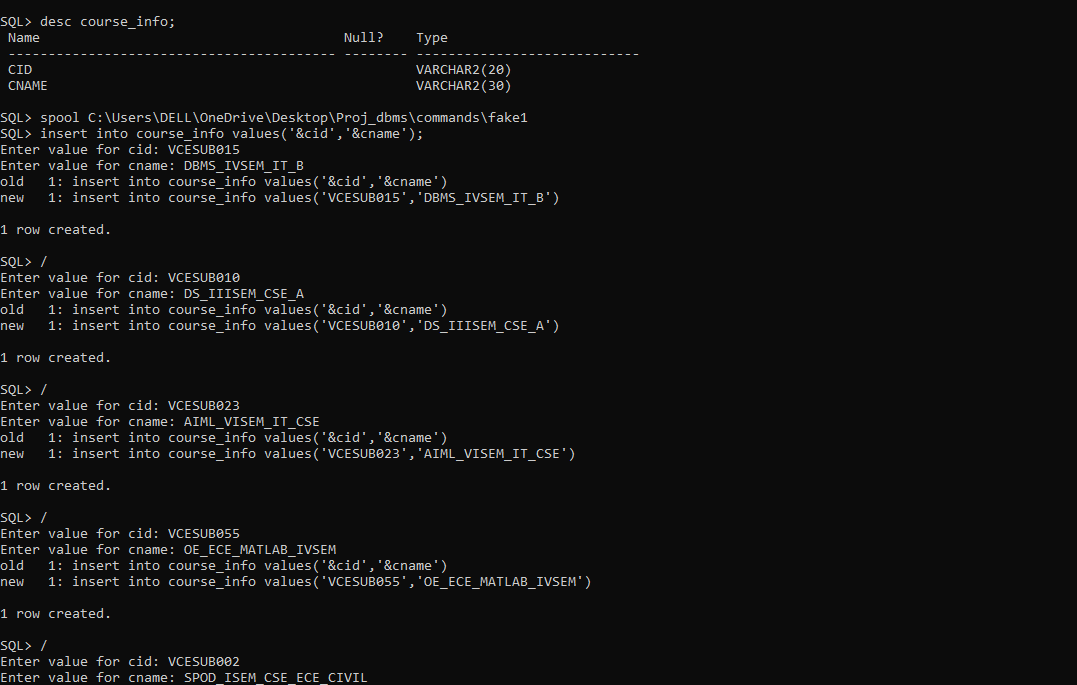












**IMPLEMENTATION**

**Front end programs and its connectivity**

**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:

Code:

import java.sql.\*;

import java.util.\*;

public class DBConnection {

private static DBConnection driver;

private Connection con;

private Statement stmt;

private DBConnection() {

try {

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

} catch (SQLException e) {

System.out.println("Message: " + e.getMessage());

}

// connect to Oracle

connect();

}

public static DBConnection getInstance() {

if (driver == null)

driver = new DBConnection();

return driver;

}

/\*

\* connects to Oracle database named using username and password

\*/

private void connect() {

String username = "project";

String password = "rushitha";

try {

con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", username, password);

//System.out.println("\nConnected to Oracle DataBase");

stmt = con.createStatement();

} catch (SQLException e) {

System.out.println("Message: " + e.getMessage() + "\nFailed to connect to Oracle DataBase");

}

}

/\*

\* execute any SQL statements alter the database (update, delete, insert)

\*/

public String executeAlter(String sqlstmt) {

try {

int res=stmt.executeUpdate(sqlstmt);

con.commit();

return "inserted "+Integer.toString(res)+" rows in table";

} catch (SQLException e) {

return e.getMessage();

}

}

/\*

\* execute query statements

\*/

public ResultSet executeQuery(String sqlstmt) {

try {

return stmt.executeQuery(sqlstmt);

} catch (SQLException e) {

System.out.println("Message: " + e.getMessage() + "\nUnable to execute: " + sqlstmt);

return null;

}

}

/\*

\* disconnect from Oracle database

\*/

public void disconnect() {

try {

con.close();

} catch (SQLException e) {

System.out.println("Message: " + e.getMessage());

System.out.println("\nFailed to disconnect from Oracle");

}

}}

Insert Queries:

import java.awt.BorderLayout;

import java.awt.EventQueue;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.JTable;

import javax.swing.border.EmptyBorder;

import javax.swing.table.DefaultTableModel;

import javax.swing.JMenuBar;

import javax.swing.JOptionPane;

import javax.swing.JMenu;

import javax.swing.JLabel;

import javax.swing.JTextField;

import javax.swing.JComboBox;

import javax.swing.DefaultComboBoxModel;

import java.awt.event.MouseAdapter;

import java.awt.event.MouseEvent;

import java.sql.ResultSet;

import java.sql.ResultSetMetaData;

import java.sql.SQLException;

import javax.swing.JButton;

import java.awt.Color;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

import java.awt.Font;

public class RegisterUser extends JFrame {

private JPanel contentPane;

private JTextField textField;

private JTextField textField\_1;

private JTextField textField\_2;

private JTextField textField\_3;

private JTextField textField\_4;

private JTextField textField\_5;

private JTextField textField\_6;

private JTextField textField\_8;

private JTextField textField\_9;

private JTextField textField\_10;

private DefaultTableModel dm, dm1, dm2;

private DBConnection jDbConnection;

private JTextField textField\_7;

/\*\*

\* Launch the application.

\*/

void loadTable(DefaultTableModel dm, JTable table, String queryString) {

dm.setColumnCount(0);

dm.setRowCount(0);

ResultSet resultSet = jDbConnection.executeQuery(queryString);

ResultSetMetaData rsmd;

try {

rsmd = resultSet.getMetaData();

//Coding to get columns-

int cols=rsmd.getColumnCount();

String c[]=new String[cols];

for(int i=0;i<cols;i++){

c[i]=rsmd.getColumnName(i+1);

dm.addColumn(c[i]);

}

//get data from rows

Object row[]=new Object[cols];

while(resultSet.next()){

for(int i=0;i<cols;i++){

row[i]=resultSet.getString(i+1);

}

dm.addRow(row);

}

table.setModel(dm);

} catch (SQLException e1) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

}

public static void main(String[] args) {

EventQueue.*invokeLater*(new Runnable() {

public void run() {

try {

RegisterUser frame = new RegisterUser();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the frame.

\*/

public RegisterUser() {

jDbConnection=DBConnection.*getInstance*();

dm = new DefaultTableModel();

dm1 = new DefaultTableModel();

dm2 = new DefaultTableModel();

setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

setBounds(100, 100, 633, 424);

JMenuBar menuBar = new JMenuBar();

setJMenuBar(menuBar);

JMenu mnNewMenu = new JMenu("Main Page");

mnNewMenu.addMouseListener(new MouseAdapter() {

*@Override*

public void mouseClicked(MouseEvent e) {

dispose();

new MainPage().setVisible(true);

}

});

menuBar.add(mnNewMenu);

contentPane = new JPanel();

contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

setContentPane(contentPane);

contentPane.setLayout(null);

JPanel panel = new JPanel();

panel.setBackground(new Color(255, 204, 204));

panel.setBounds(10, 0, 607, 352);

contentPane.add(panel);

panel.setLayout(null);

JLabel lblNewLabel = new JLabel("Registration ID");

lblNewLabel.setFont(new Font("Arial Black", Font.***ITALIC***, 11));

lblNewLabel.setBounds(10, 11, 105, 14);

panel.add(lblNewLabel);

JLabel lblNewLabel\_1 = new JLabel("First Name");

lblNewLabel\_1.setFont(new Font("Arial Black", Font.***ITALIC***, 11));

lblNewLabel\_1.setBounds(10, 122, 86, 14);

panel.add(lblNewLabel\_1);

JLabel lblNewLabel\_2 = new JLabel("Phone Number");

lblNewLabel\_2.setFont(new Font("Arial Black", Font.***ITALIC***, 11));

lblNewLabel\_2.setBounds(10, 222, 121, 14);

panel.add(lblNewLabel\_2);

JLabel lblNewLabel\_3 = new JLabel("Address");

lblNewLabel\_3.setFont(new Font("Arial Black", Font.***ITALIC***, 11));

lblNewLabel\_3.setBounds(23, 273, 73, 14);

panel.add(lblNewLabel\_3);

JLabel lblNewLabel\_4 = new JLabel("E-Mail");

lblNewLabel\_4.setFont(new Font("Arial Black", Font.***ITALIC***, 11));

lblNewLabel\_4.setBounds(345, 172, 46, 14);

panel.add(lblNewLabel\_4);

JLabel lblNewLabel\_5 = new JLabel("UserName");

lblNewLabel\_5.setFont(new Font("Arial Black", Font.***ITALIC***, 11));

lblNewLabel\_5.setBounds(10, 77, 86, 14);

panel.add(lblNewLabel\_5);

JLabel lblNewLabel\_6 = new JLabel("Last Name");

lblNewLabel\_6.setFont(new Font("Arial Black", Font.***ITALIC***, 11));

lblNewLabel\_6.setBounds(330, 122, 86, 14);

panel.add(lblNewLabel\_6);

JLabel lblNewLabel\_7 = new JLabel("Gender");

lblNewLabel\_7.setFont(new Font("Arial Black", Font.***ITALIC***, 11));

lblNewLabel\_7.setBounds(10, 172, 65, 14);

panel.add(lblNewLabel\_7);

JLabel lblNewLabel\_8 = new JLabel("DOB");

lblNewLabel\_8.setFont(new Font("Arial Black", Font.***ITALIC***, 11));

lblNewLabel\_8.setBounds(345, 222, 46, 14);

panel.add(lblNewLabel\_8);

JLabel lblNewLabel\_9 = new JLabel("Collage Address");

lblNewLabel\_9.setFont(new Font("Arial Black", Font.***ITALIC***, 11));

lblNewLabel\_9.setBounds(336, 273, 115, 14);

panel.add(lblNewLabel\_9);

JLabel lblNewLabel\_10 = new JLabel("Password");

lblNewLabel\_10.setFont(new Font("Arial Black", Font.***ITALIC***, 11));

lblNewLabel\_10.setBounds(330, 77, 86, 14);

panel.add(lblNewLabel\_10);

textField = new JTextField();

textField.setBounds(157, 8, 136, 20);

panel.add(textField);

textField.setColumns(10);

textField\_1 = new JTextField();

textField\_1.setBounds(157, 75, 136, 20);

panel.add(textField\_1);

textField\_1.setColumns(10);

textField\_2 = new JTextField();

textField\_2.setBounds(157, 120, 136, 20);

panel.add(textField\_2);

textField\_2.setColumns(10);

textField\_3 = new JTextField();

textField\_3.setBounds(157, 173, 46, 20);

panel.add(textField\_3);

textField\_3.setColumns(10);

textField\_4 = new JTextField();

textField\_4.setBounds(157, 219, 136, 20);

panel.add(textField\_4);

textField\_4.setColumns(10);

textField\_5 = new JTextField();

textField\_5.setBounds(157, 271, 136, 20);

panel.add(textField\_5);

textField\_5.setColumns(10);

textField\_6 = new JTextField();

textField\_6.setBounds(474, 75, 105, 20);

panel.add(textField\_6);

textField\_6.setColumns(10);

textField\_8 = new JTextField();

textField\_8.setBounds(474, 170, 105, 20);

panel.add(textField\_8);

textField\_8.setColumns(10);

textField\_9 = new JTextField();

textField\_9.setBounds(474, 219, 105, 20);

panel.add(textField\_9);

textField\_9.setColumns(10);

textField\_10 = new JTextField();

textField\_10.setBounds(474, 271, 105, 20);

panel.add(textField\_10);

textField\_10.setColumns(10);

JButton btnNewButton = new JButton("Clear All");

btnNewButton.setFont(new Font("Arial Black", Font.***BOLD***, 11));

btnNewButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

textField.setText("");

textField\_1.setText("");

textField\_2.setText("");

textField\_3.setText("");

textField\_4.setText("");

textField\_5.setText("");

textField\_6.setText("");

textField\_8.setText("");

textField\_9.setText("");

textField\_10.setText("");

}

});

btnNewButton.setBounds(42, 318, 105, 23);

panel.add(btnNewButton);

JButton btnNewButton\_2 = new JButton("Confirm");

btnNewButton\_2.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

String stringQuery="insert into userdetails values('";

stringQuery+=textField\_1.getText()+"','";

stringQuery+=textField\_6.getText()+"', '";

stringQuery+=textField\_2.getText()+"',' ";

stringQuery+=textField\_7.getText()+"',' ";

stringQuery+=textField\_3.getText()+"',' ";

stringQuery+=textField\_8.getText()+"', ";

stringQuery+=textField\_4.getText()+", '";

stringQuery+=textField\_9.getText()+"', ";

stringQuery+=textField.getText()+", '";

stringQuery+=textField\_5.getText()+"', '";

stringQuery+=textField\_10.getText()+"')";

//stringQuery+=")";

String res=jDbConnection.executeAlter(stringQuery);

if(res=="1")

{

JOptionPane.*showMessageDialog*(null,res);

}else

{

JOptionPane.*showMessageDialog*(null,res);

}

}

});

btnNewButton\_2.setFont(new Font("Arial Black", Font.***BOLD***, 11));

btnNewButton\_2.setBounds(475, 318, 104, 23);

panel.add(btnNewButton\_2);

textField\_7 = new JTextField();

textField\_7.setBounds(474, 120, 105, 20);

panel.add(textField\_7);

textField\_7.setColumns(10);

}

}

Main:

import java.awt.BorderLayout;

import java.awt.EventQueue;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.border.EmptyBorder;

import javax.swing.table.DefaultTableModel;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JTextField;

import javax.swing.JPasswordField;

import javax.swing.JButton;

import javax.swing.JSeparator;

import javax.swing.JTable;

import java.awt.Color;

import java.awt.Font;

import java.awt.event.ActionListener;

import java.sql.ResultSet;

import java.sql.ResultSetMetaData;

import java.sql.SQLException;

import java.awt.event.ActionEvent;

import javax.swing.ImageIcon;

public class Login extends JFrame {

private JPanel contentPane;

private JTextField txtUserName;

private DefaultTableModel dm, dm1, dm2;

private DBConnection jDbConnection;

public static boolean isLoggedin=false;

protected String username="admin";

protected String password="admin";

private JTextField txtPassword;

/\*\*

\* Launch the application.

\*/

void loadTable(DefaultTableModel dm, JTable table, String queryString) {

dm.setColumnCount(0);

dm.setRowCount(0);

ResultSet resultSet = jDbConnection.executeQuery(queryString);

ResultSetMetaData rsmd;

try {

rsmd = resultSet.getMetaData();

//Coding to get columns-

int cols=rsmd.getColumnCount();

String c[]=new String[cols];

for(int i=0;i<cols;i++){

c[i]=rsmd.getColumnName(i+1);

dm.addColumn(c[i]);

}

//get data from rows

Object row[]=new Object[cols];

while(resultSet.next()){

for(int i=0;i<cols;i++){

row[i]=resultSet.getString(i+1);

}

dm.addRow(row);

}

table.setModel(dm);

} catch (SQLException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

}

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

Login frame = new Login();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the frame.

\*/

public Login() {

jDbConnection=DBConnection.getInstance();

dm = new DefaultTableModel();

dm1 = new DefaultTableModel();

dm2 = new DefaultTableModel();

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setBounds(100, 100, 632, 424);

contentPane = new JPanel();

contentPane.setBackground(new Color(51, 204, 153));

contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

setContentPane(contentPane);

contentPane.setLayout(null);

JLabel lblNewLabel = new JLabel("ADMIN LOGIN");

lblNewLabel.setIcon(new ImageIcon("C:\\Users\\DELL\\eclipse-workspace\\VAMS\\Images\\icons8-microsoft-admin-30.png"));

lblNewLabel.setFont(new Font("Arial Black", Font.BOLD, 14));

lblNewLabel.setBounds(215, 3, 166, 46);

contentPane.add(lblNewLabel);

JButton btnLogin = new JButton("Login");

btnLogin.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

if(txtUserName.getText().equals(username))

{

if(txtPassword.getText().equals(password))

{

isLoggedin = true;

dispose();

new AdminMainPage().setVisible(true);

}else {

JOptionPane.showMessageDialog(null,"Invalid Password!");

}

}else {

JOptionPane.showMessageDialog(null,"Invalid User Name!");

}

}

});

btnLogin.setFont(new Font("Arial Black", Font.PLAIN, 13));

btnLogin.setBounds(72, 327, 89, 23);

contentPane.add(btnLogin);

JButton btnExit = new JButton("Exit");

btnExit.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

dispose();

new MainPage().setVisible(true);

}

});

btnExit.setFont(new Font("Arial Black", Font.PLAIN, 13));

btnExit.setBounds(453, 327, 89, 23);

contentPane.add(btnExit);

JSeparator separator = new JSeparator();

separator.setBackground(new Color(0, 100, 0));

separator.setBounds(46, 295, 516, 2);

contentPane.add(separator);

JSeparator separator\_1 = new JSeparator();

separator\_1.setBackground(new Color(0, 128, 0));

separator\_1.setBounds(46, 60, 516, 2);

contentPane.add(separator\_1);

JPanel panel = new JPanel();

panel.setBackground(new Color(204, 255, 204));

panel.setBounds(34, 73, 528, 211);

contentPane.add(panel);

panel.setLayout(null);

txtPassword = new JTextField();

txtPassword.setBackground(new Color(204, 255, 204));

txtPassword.setBounds(268, 130, 184, 20);

panel.add(txtPassword);

txtPassword.setColumns(10);

JLabel lblUserName = new JLabel("");

lblUserName.setBounds(101, 22, 89, 40);

panel.add(lblUserName);

lblUserName.setIcon(new ImageIcon("C:\\Users\\DELL\\eclipse-workspace\\VAMS\\Images\\icons8-user-50.png"));

lblUserName.setFont(new Font("Arial", Font.ITALIC, 13));

txtUserName = new JTextField();

txtUserName.setBackground(new Color(204, 255, 204));

txtUserName.setBounds(268, 42, 191, 20);

panel.add(txtUserName);

txtUserName.setColumns(10);

JLabel lblPassword = new JLabel("");

lblPassword.setBounds(101, 116, 112, 68);

panel.add(lblPassword);

lblPassword.setIcon(new ImageIcon("C:\\Users\\DELL\\eclipse-workspace\\VAMS\\Images\\icons8-password-50.png"));

lblPassword.setFont(new Font("Arial", Font.ITALIC, 13));

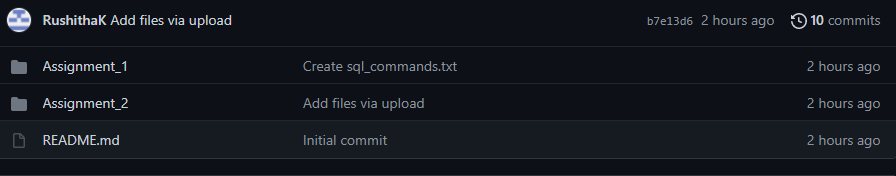
}

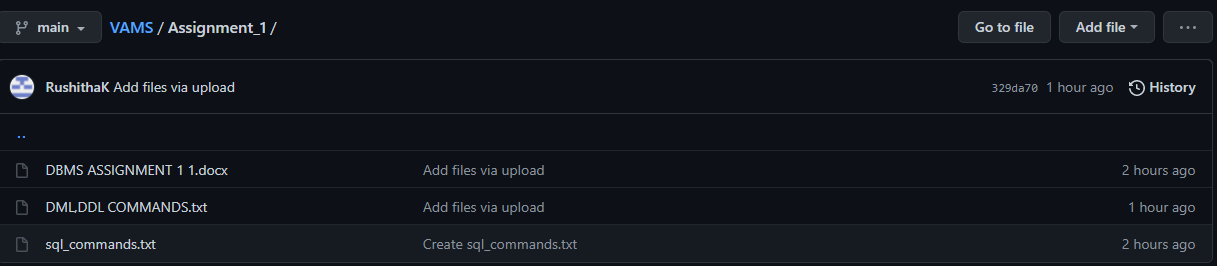
}

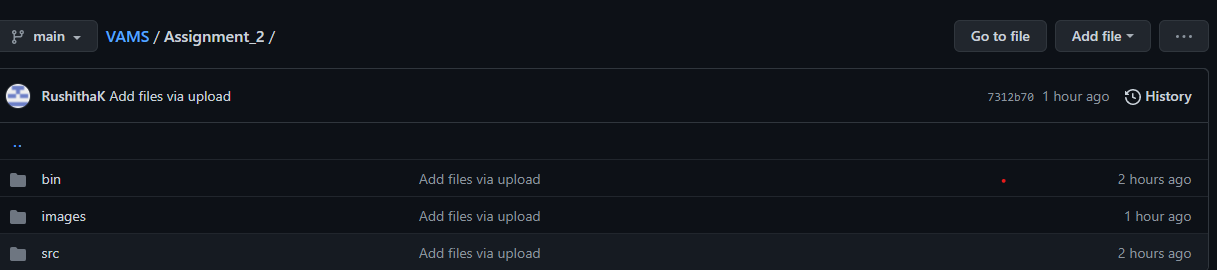
## Github links and folder structure:

Link: https://github.com/RushithaK/VAMS

Folder structure:



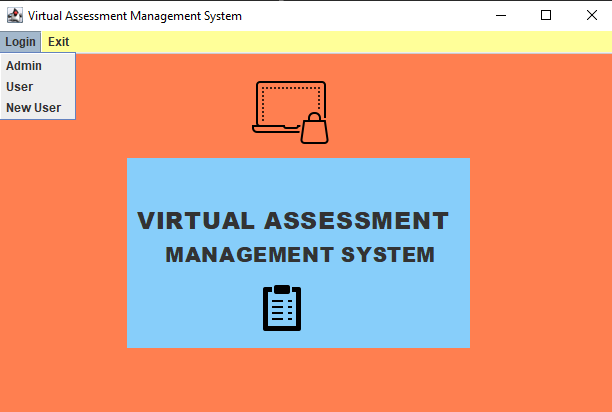




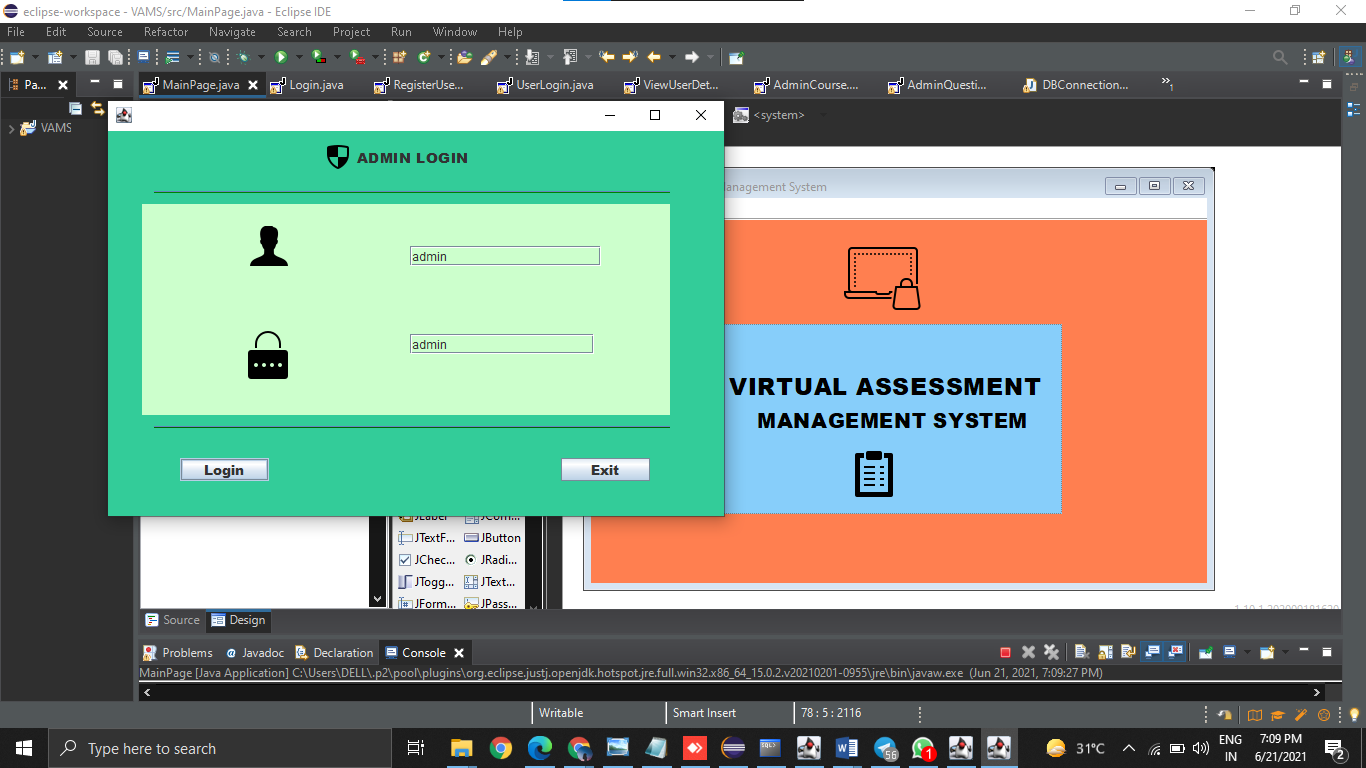
**TESTING**

Login Page

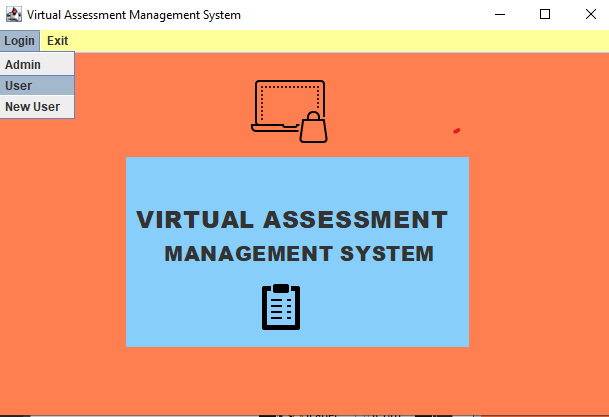
Login:

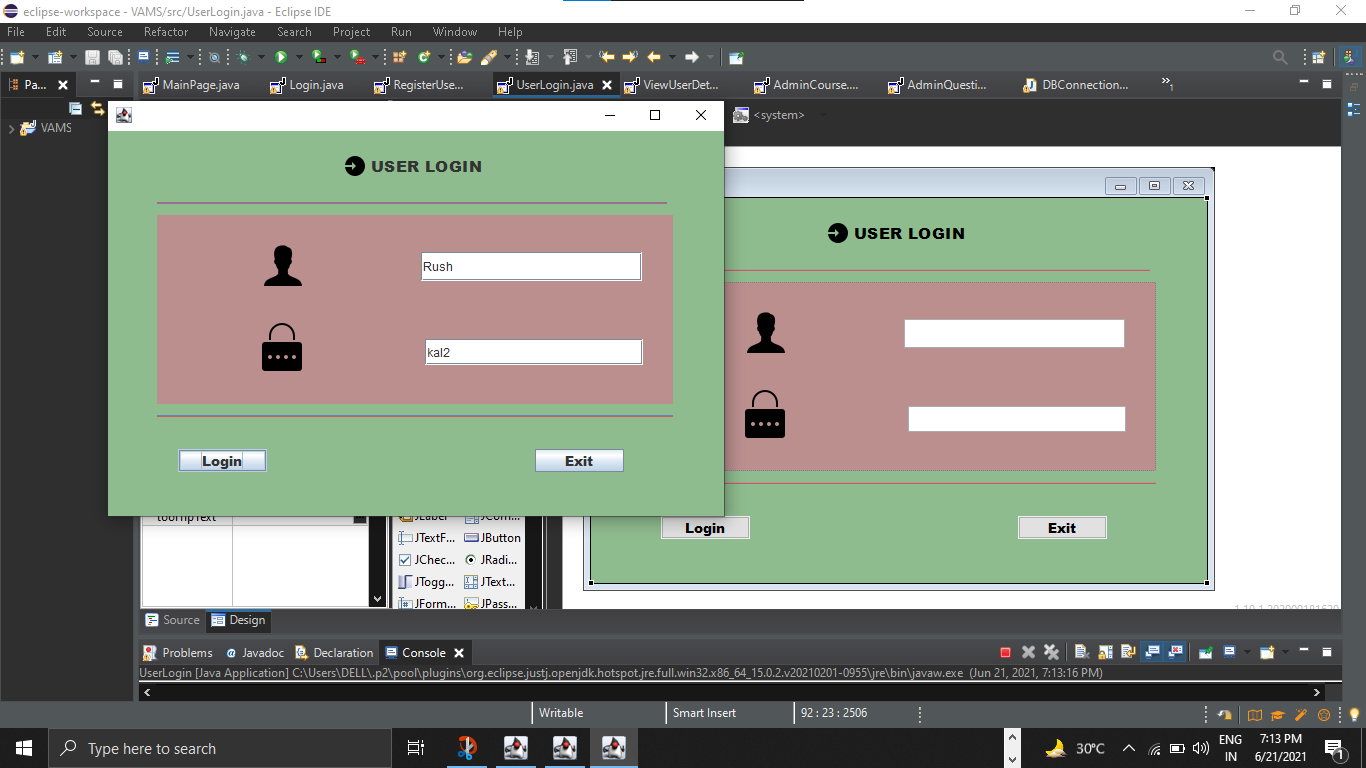


ADmin

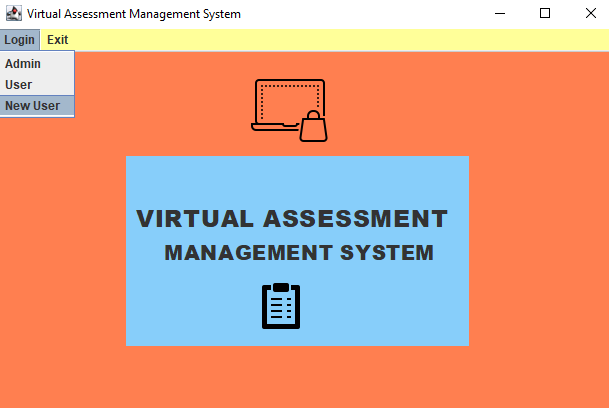


UserLogin:

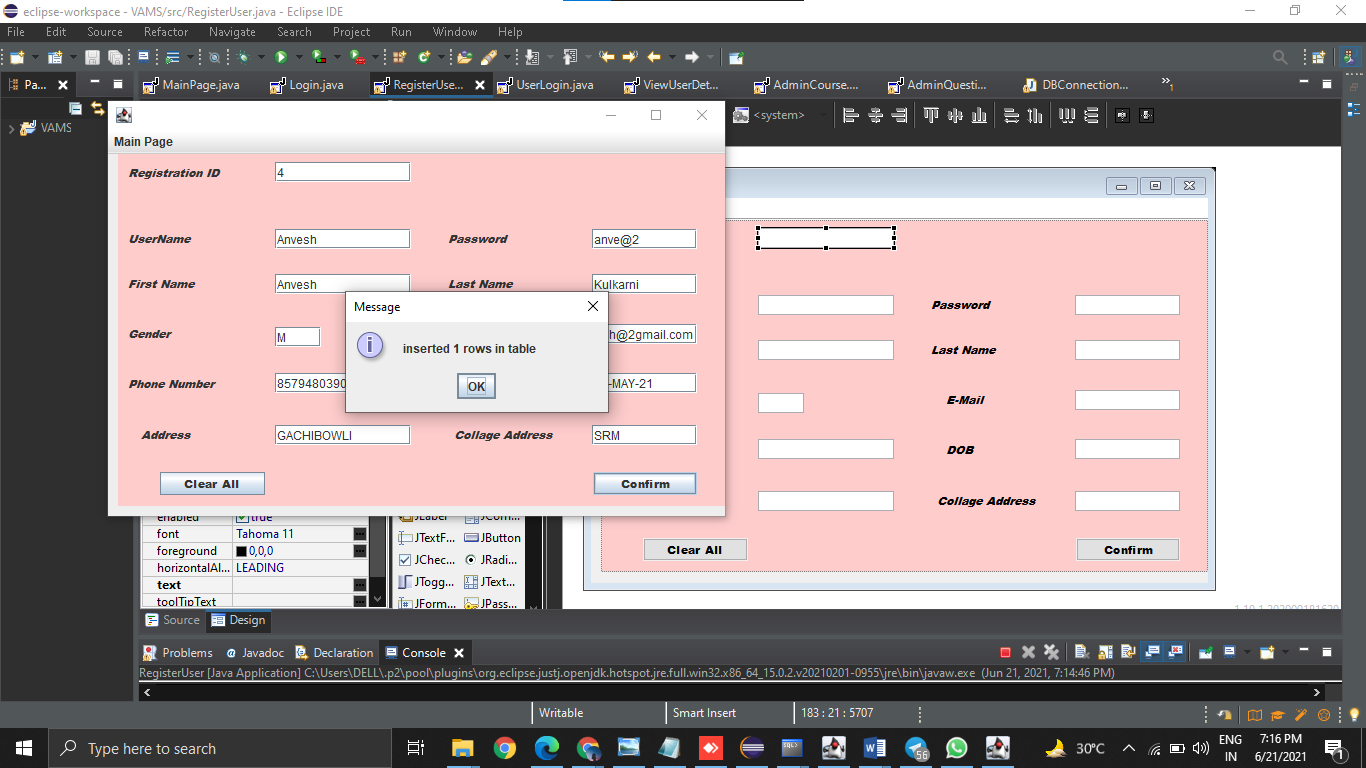




New User:

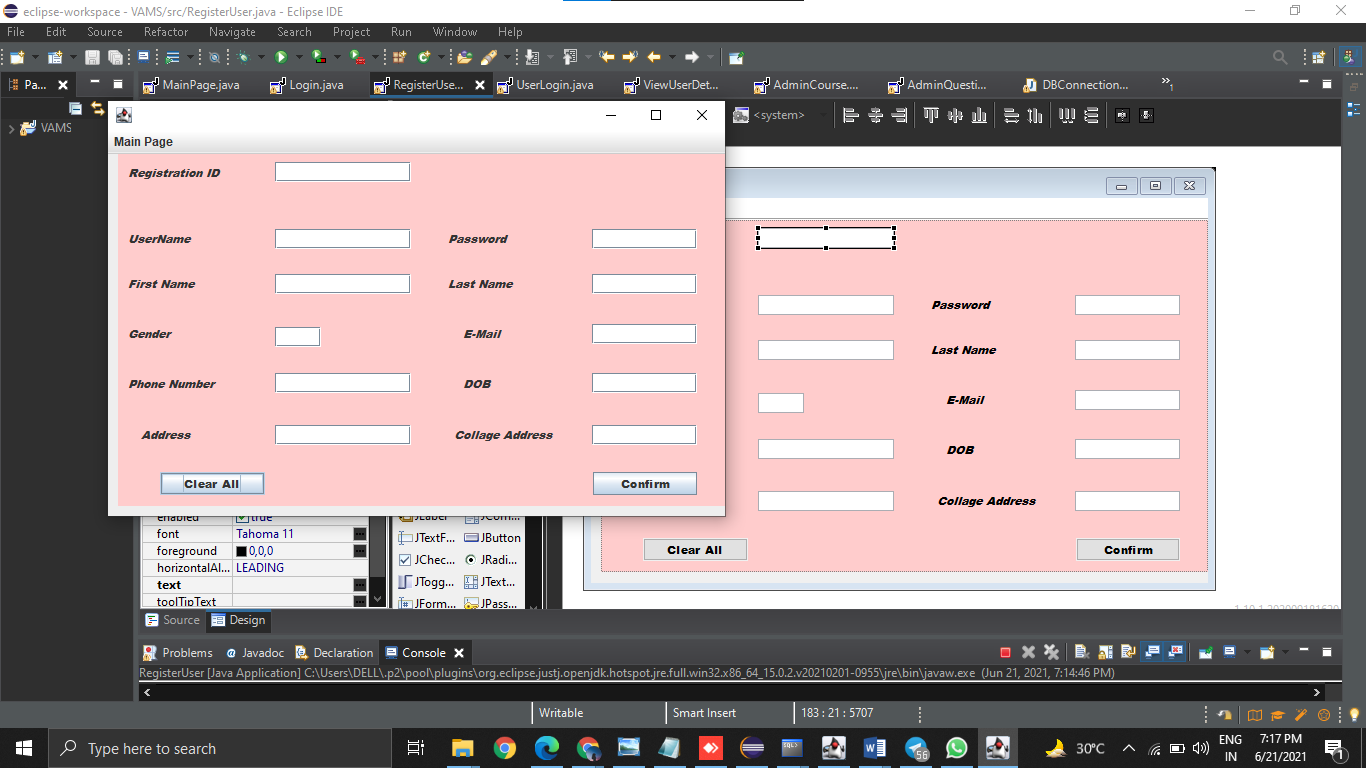


New User Insert:

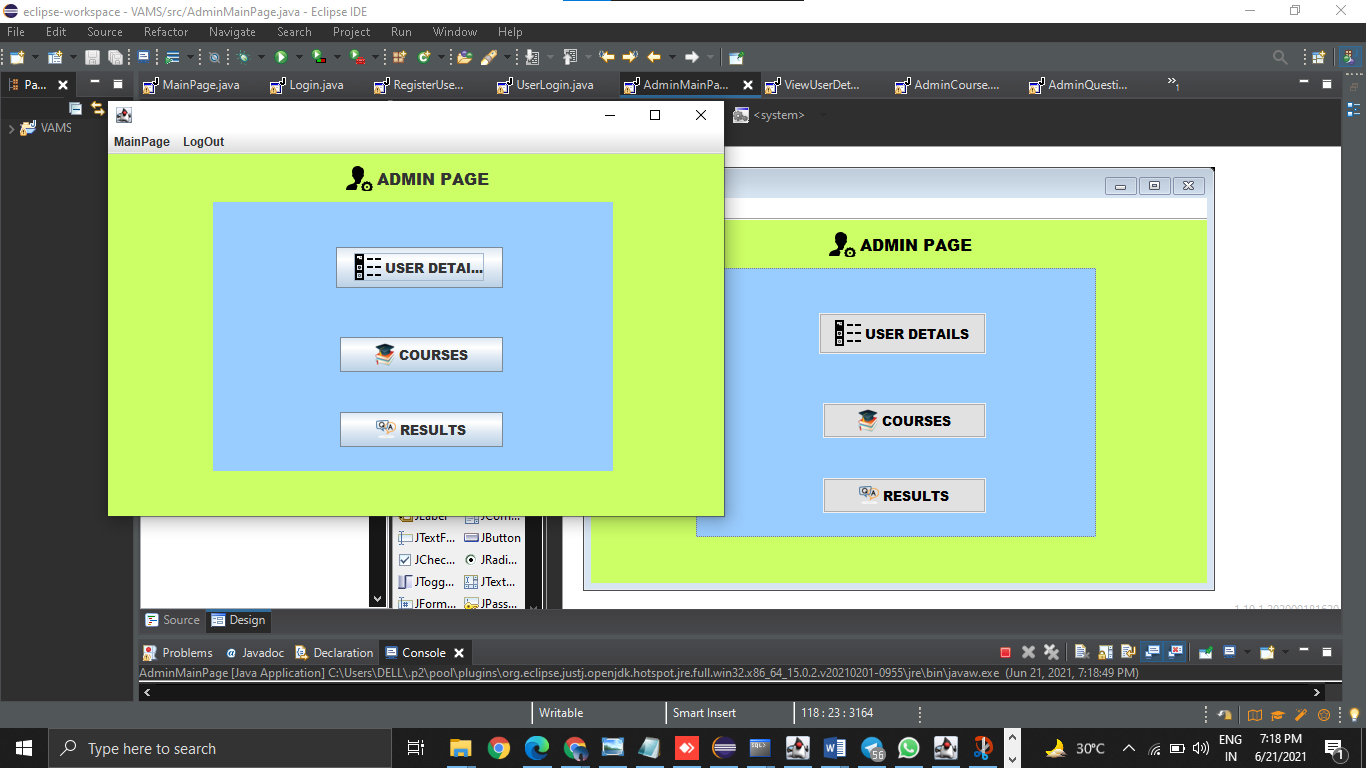


New User

Clear:

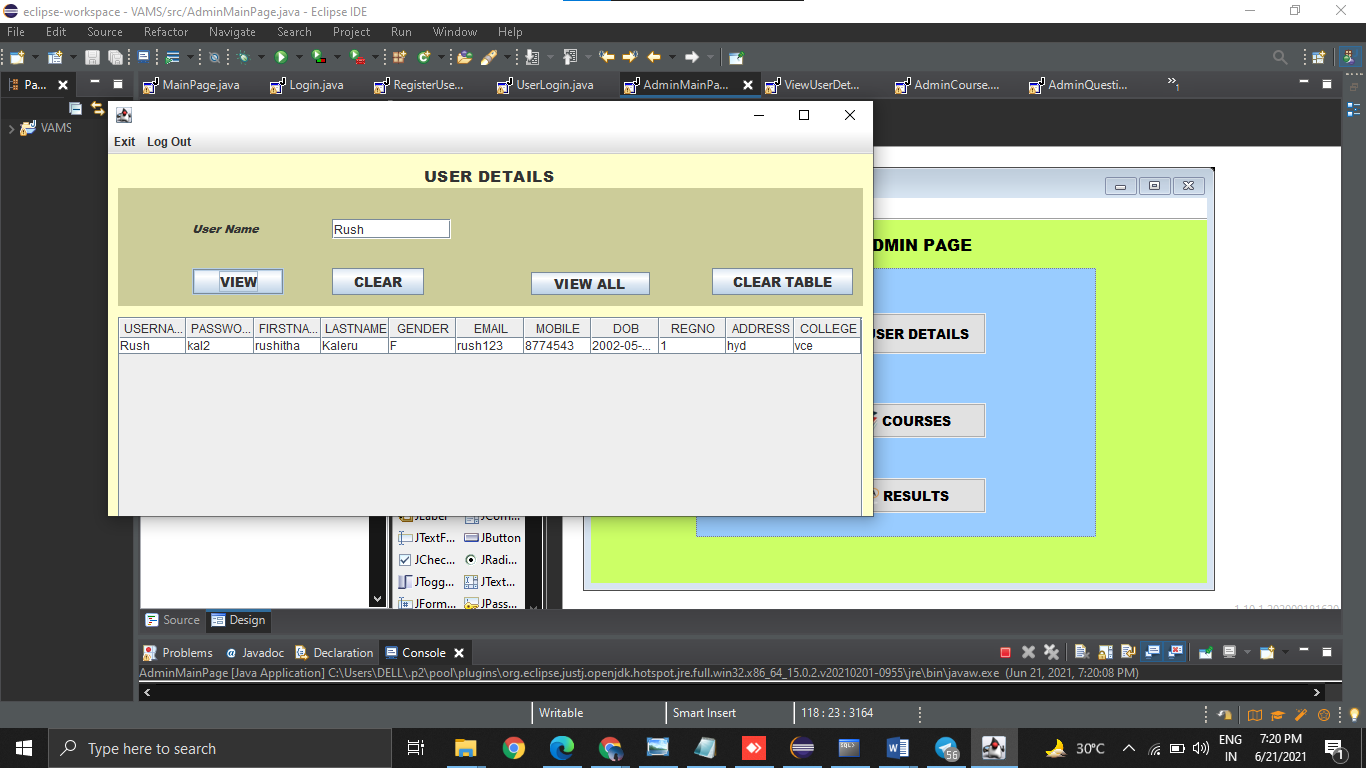


Admin Main Page:



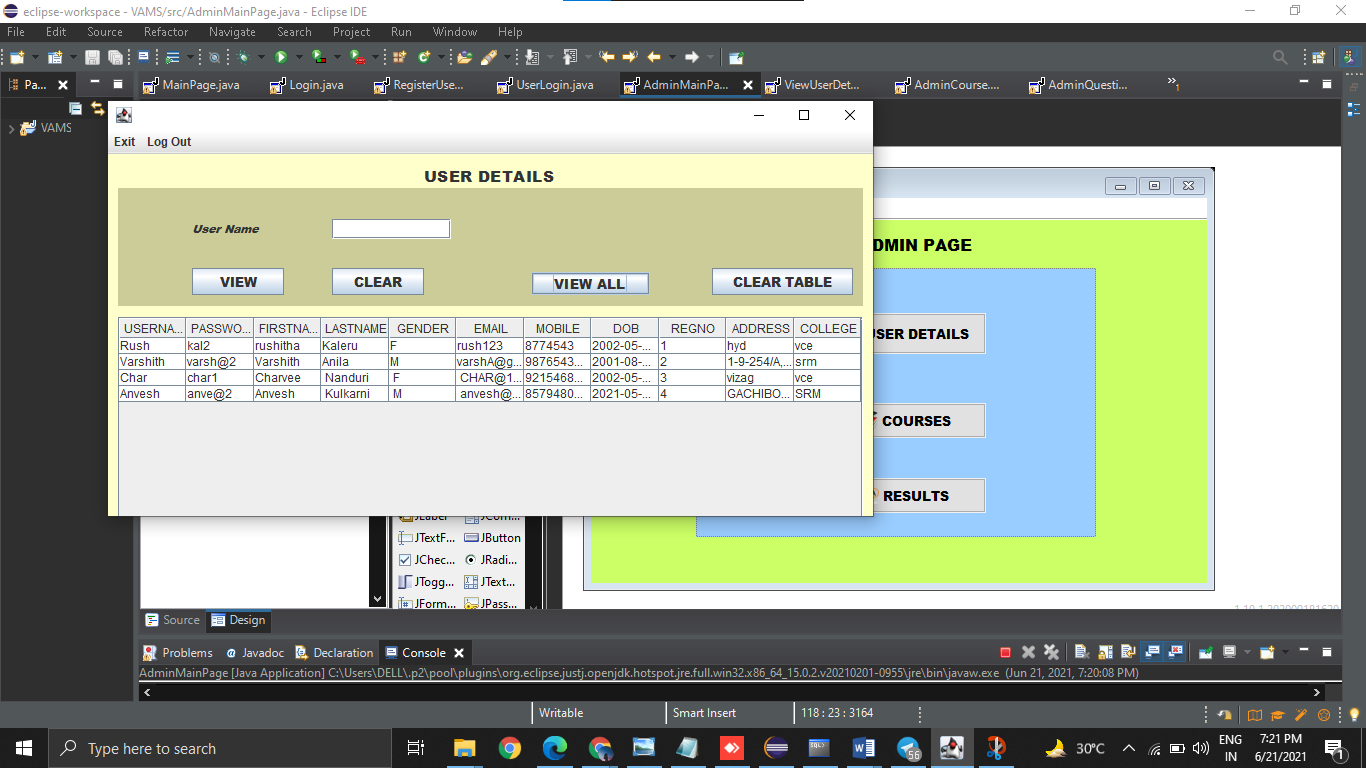
UserDetails:

View

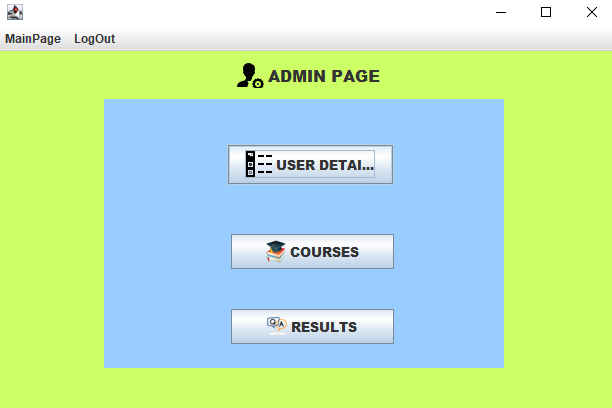


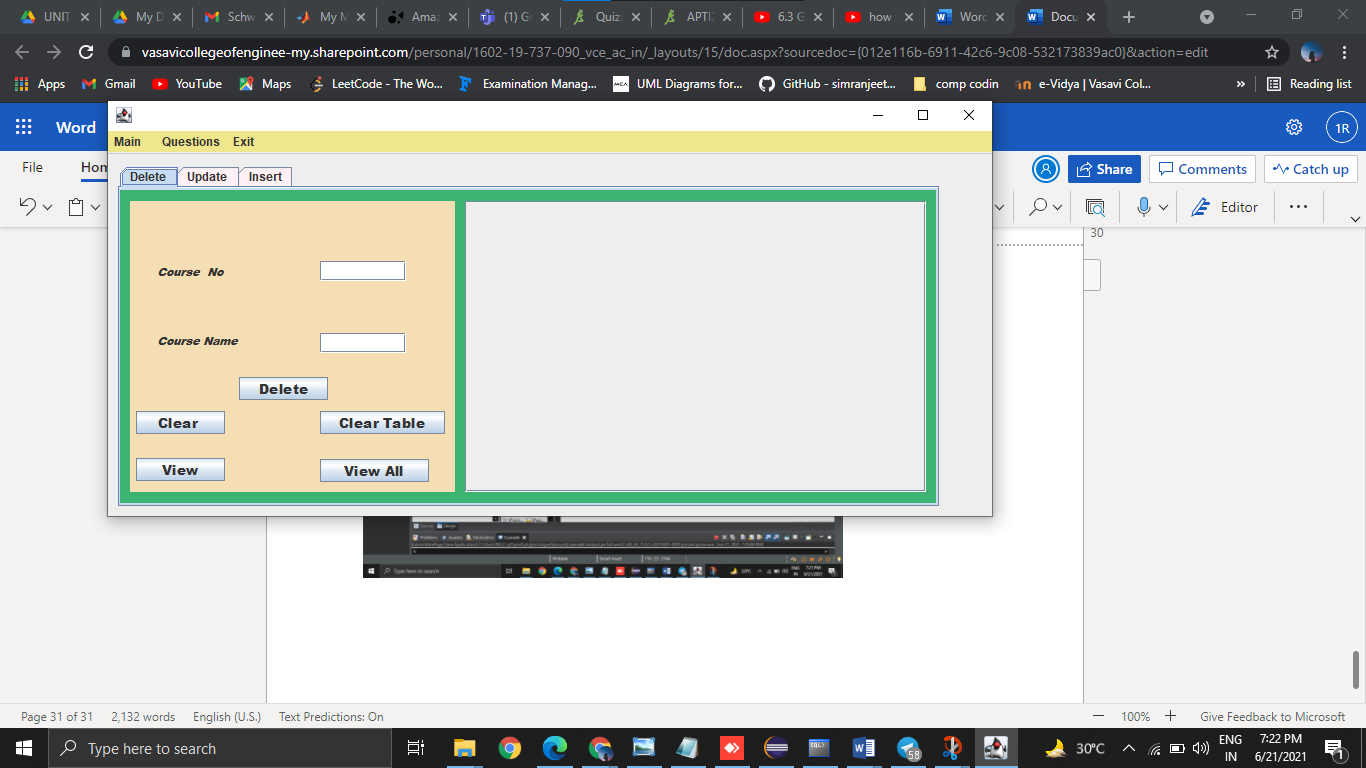
UserDetails

View All

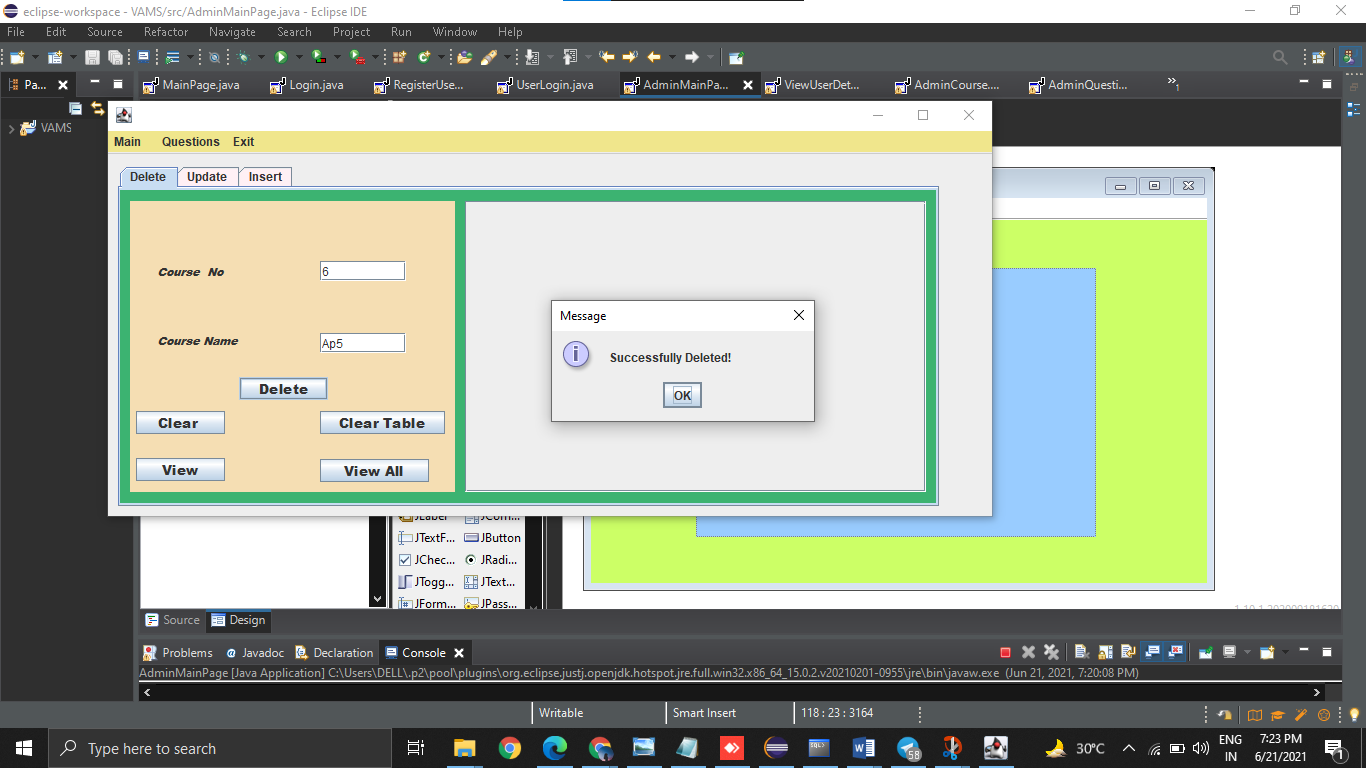


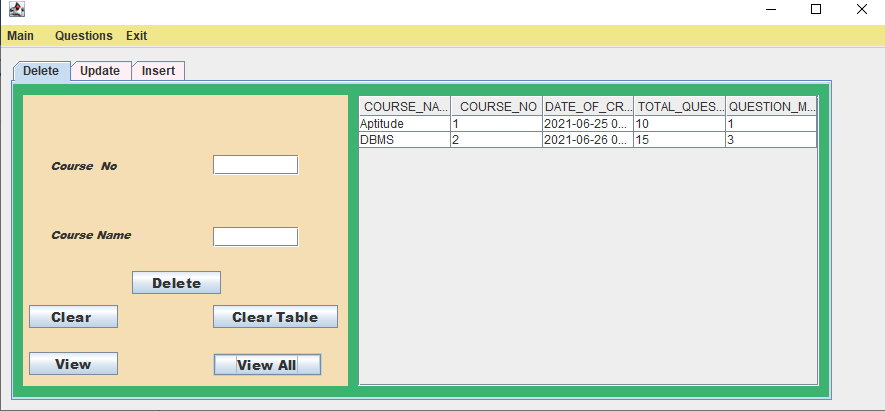
Courses:





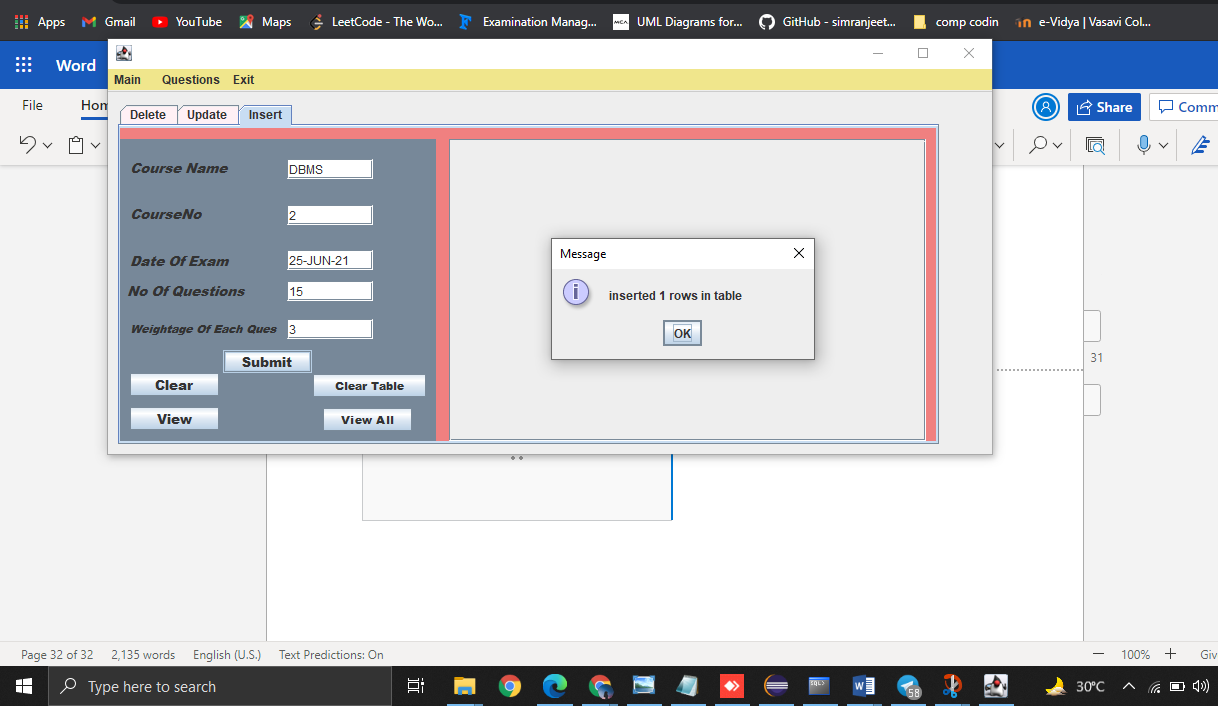
Courses Delete:



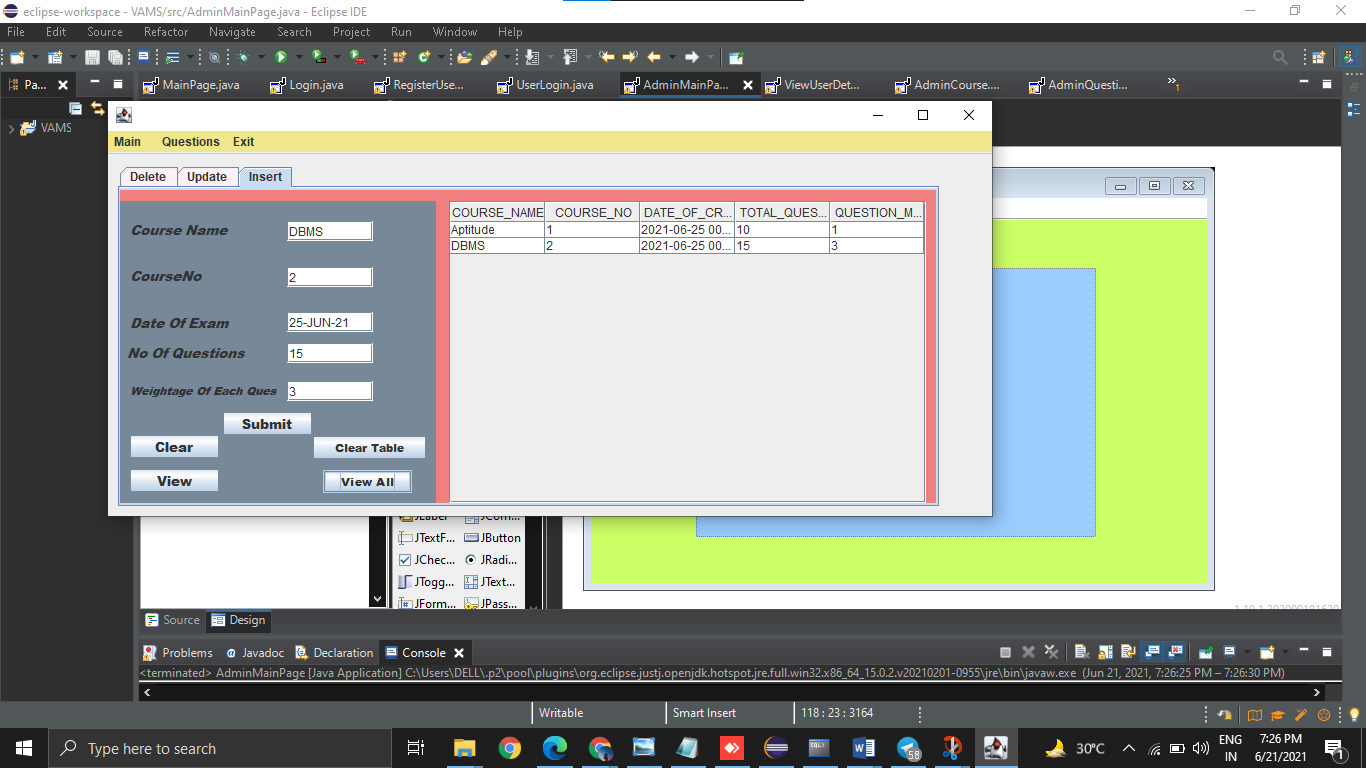


COURSE DETAILS:

INSERT

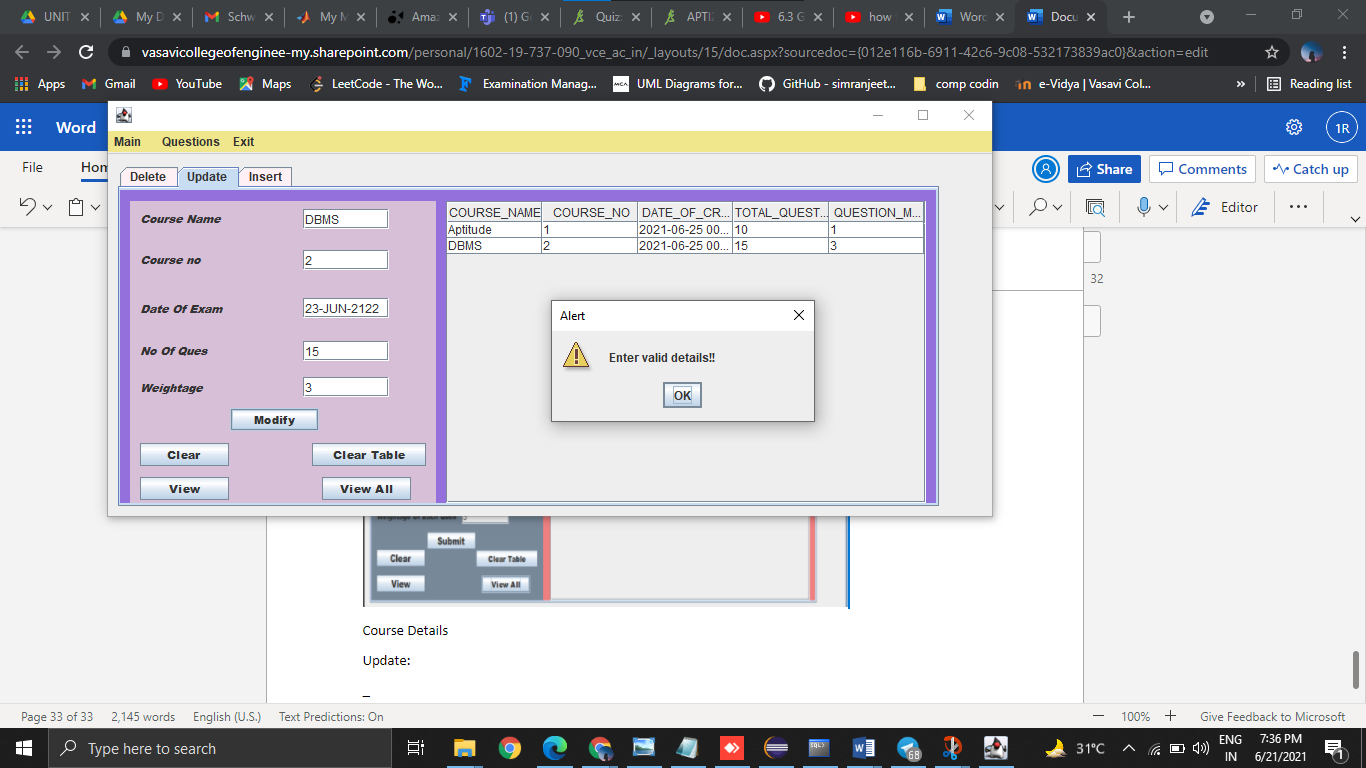


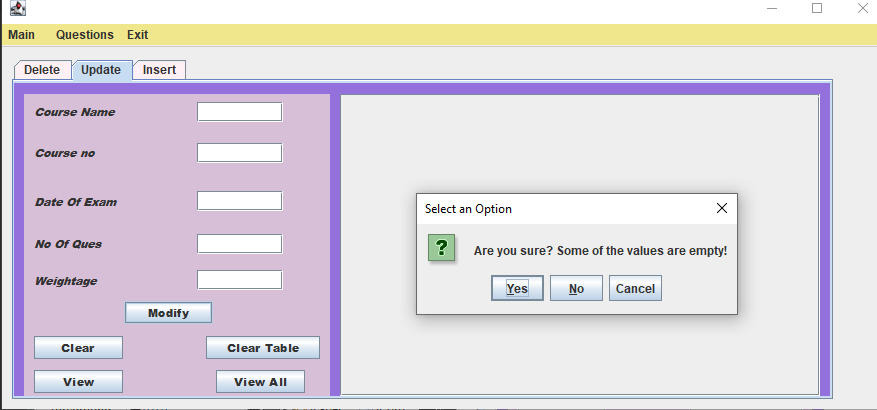
View And Clear



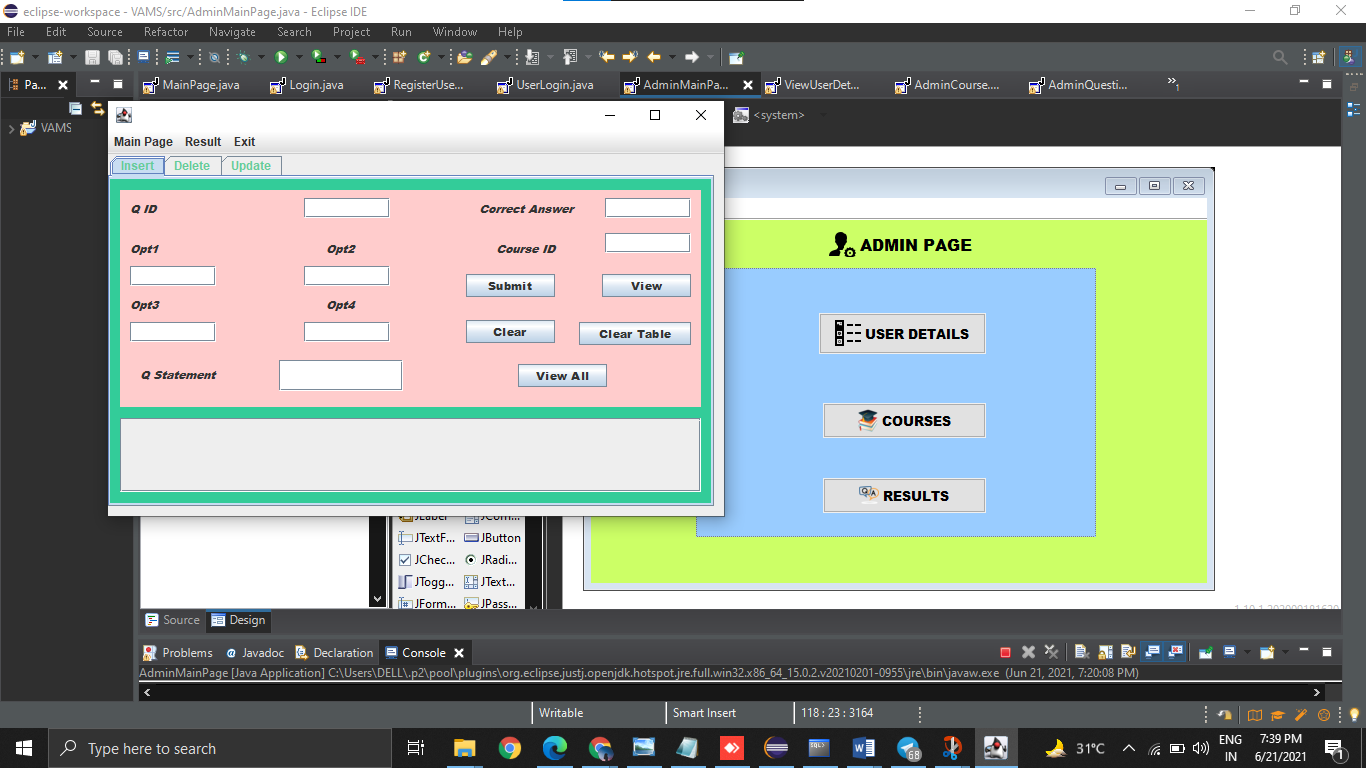
Course Details

Update:

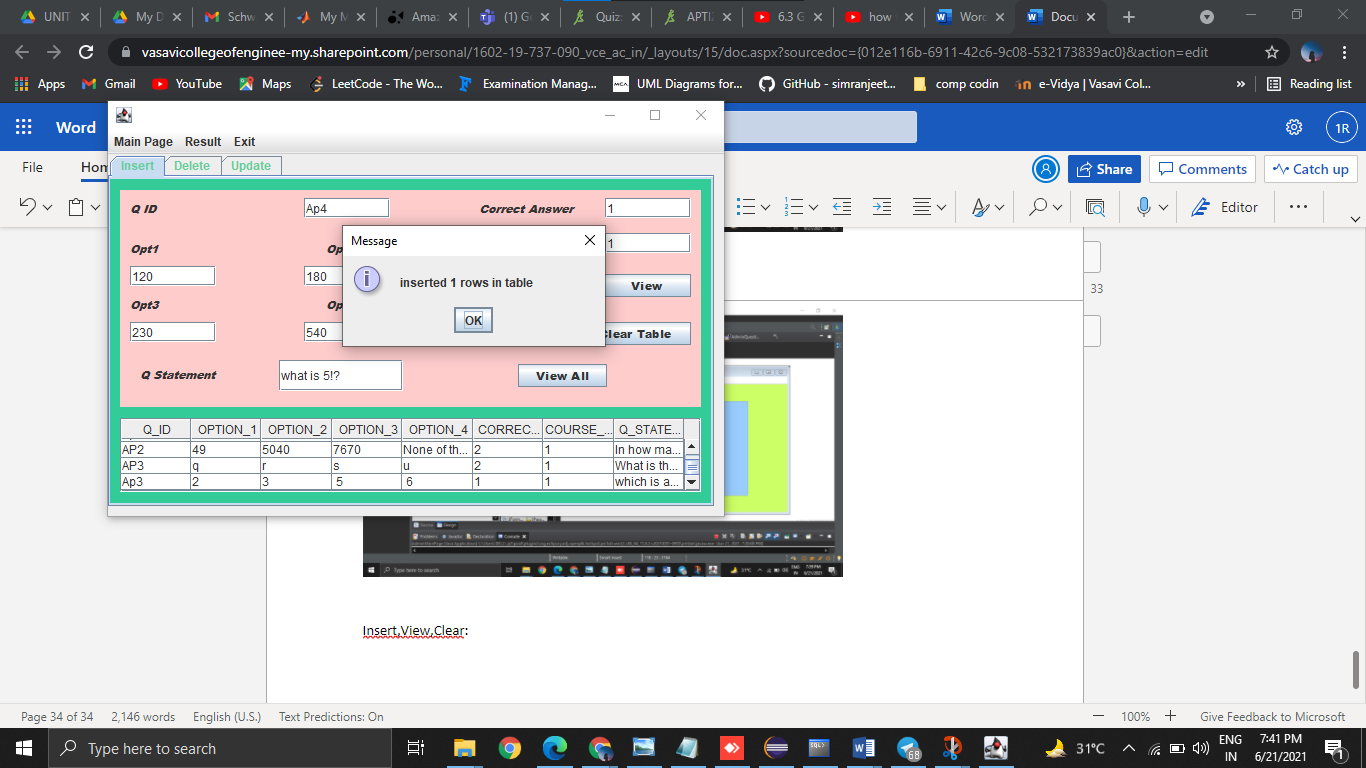


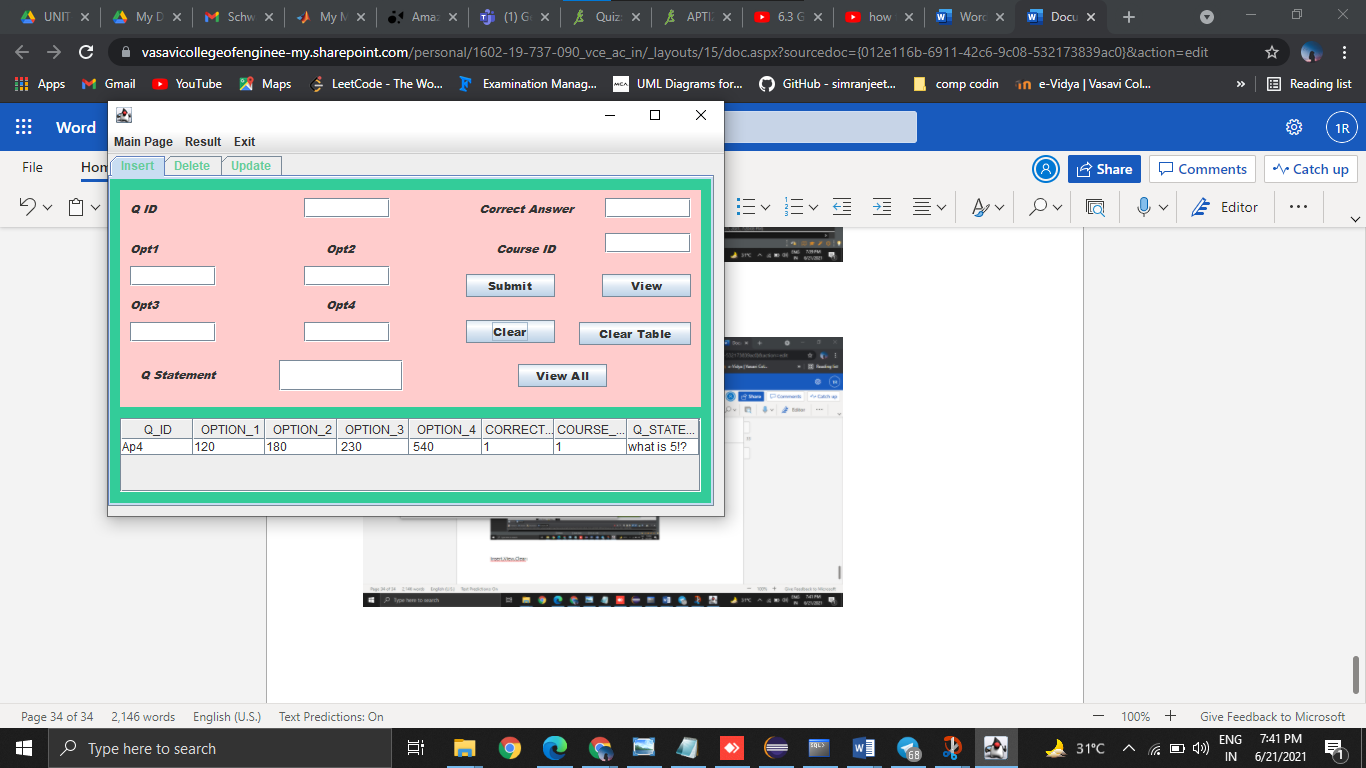


Questions:



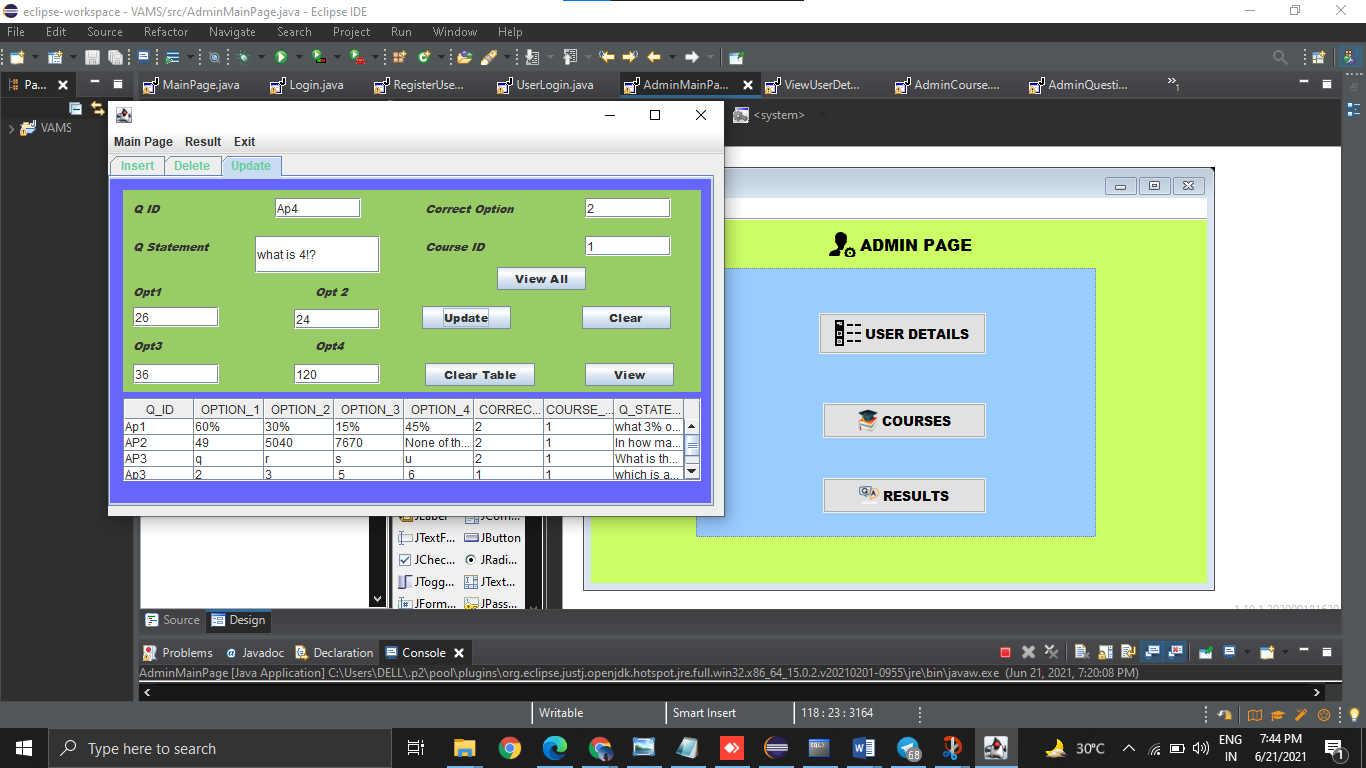
Insert,View,Clear:

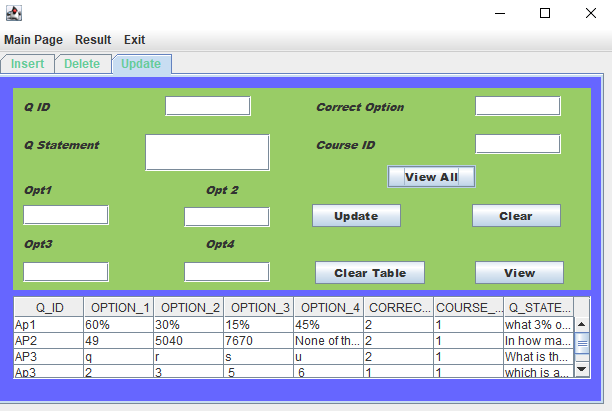


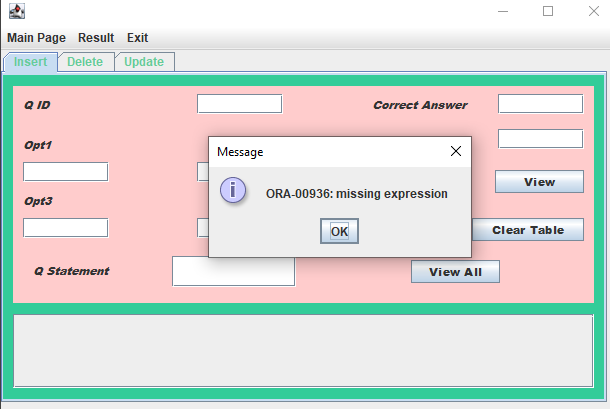


Questions

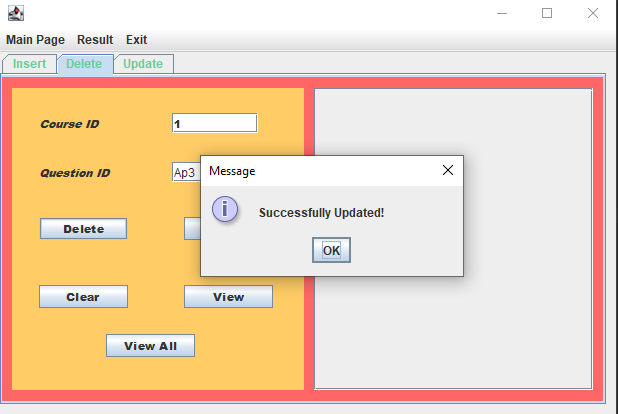
Update,View,Clear:

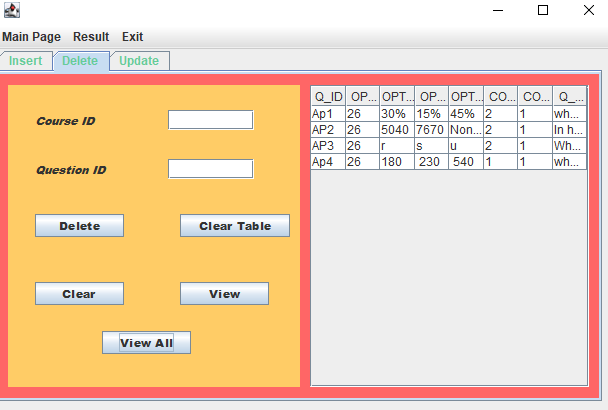






DELETE:





**DISCUSSION AND FUTURE WORK**

So far this project has helped us retrieve the information about the virtual exam system given . The updating deleting and inserting the courses and questions by the admin eventually is the crucial aspect here as it is the parameter that rates the query and its solution. It is a great project and had known lot of things while doing the project.

In future the most probable aspects could be involving attendance and web cam option using more queries with a high intellect merely to improve the application quality. The query also could be analyzed based on how difficult or easy its solution could be. The examination system could be made more complex rather than just being stuck to a normal application. Lastly, there could be a room for including advanced software and other technologies that could make the project more purposeful and better for future use.

**REFERENCES**

[https://www.oracle.com/in/database/technologies/112010-](https://www.oracle.com/in/database/technologies/112010-win64soft.html) [win64soft.html](https://www.oracle.com/in/database/technologies/112010-win64soft.html)

https://www.youtube.com/watch?v=HXV3zeQKqGY