**JAVA SWING BASED- STUDENTS DORMITORY- SQL CONNECTIVITY USING JDBC**

A

Report

Submitted in partial fulfilment of the

Requirements for the award of the Degree of

**BACHELOR OF ENGINEERING**

**IN**

**INFORMATION TECHNOLOGY**

By

SUSHMA MANE <1602-19-737-117>

Under the Guidance of

B. Leelavathy



Department of Information Technology

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

Ibrahimbagh, Hyderabad-31

2020-2021

**BONAFIDE CERTIFICATE**

This to Certify that the project report titled

“STUDENTS DORMITORY”

project work of SUHSMA MANE bearing

Roll.no:1602-19-737-117 who carried out this

project under my supervision in the IV semester for

the academic year 2020-2021.

Signature Signature External examiner Internal examiner

**ABSTRACT**

The students dormitory is the place to students from far away distance to a new place spend their daily life.This database provides the details of students and their information and other factors.It allows the database administrator to allocate a seat or room of their choice in the dormitory.This is an automated system where students can know all the services provided by dormitory through online.The database manages the details of old and new students,seats allocation,rooms ,payments using the web portal. When we enter the data it is stored in database and we can view when it’s needed and delete the record .

ROLL NO: 1602-19-737-117

NAME: MANE SUSHMA

DBMS MINI PROJECT

TILTLE: STUDENTS DORMITORY

**DESIGN REQUIREMENTS:-**

**List of Tables:**

**\***DataBase administrator

**\***Dormitory

\*Students

\*Fees

\*Room

\*Staff

\*Guests

**List of attributes with their domain types:-**

**\***DB Administrator :

Administratorid : admin\_id –number(10);

Asministrator name : admin\_name : varchar2(20);

Mobile number : admin\_mob-number(20);

\*Dormitory :

Dormitory number : dorm\_id - number(10);

No.of.students : ns - number(10);

No.of.rooms : nr - number(10);

Address : dor\_add - varchar2(30);

\*Students :

Student identification number : stu\_id -number(10);

Studentname : stu\_name - varchar2(20);

Gaurdian name : gau\_name – varchar2(30);

Mobile number : stu\_mob – number(20);

Government Proof number : gov\_id – number(20);

\*Fees :

Year and month of fee to pay : fee – varchar2(20);

Fee condition : sfee – varchar2(20);

\*Room :

Room number : room\_id –number(10);

No.of beds in room : nbr-number(10);

No.of Students in each room : nsr-number(10);

Room condition : con-varchar2(20);

\*Staff :

Staff identification number : stf\_id-number(10);

Staff name : stf\_name-varchar2(20);

Mobile number : stf\_mob-numbeer(20);

Address : stf\_add-varchar2(20);

Salary : stf\_sal-number(10);

\*Guests :

Date of coming : dat-varchar2(20);

Visitors name : gname-varchar2(25);

Studentid : stu\_id-number(10);

Entered time : time\_in-varchar2(20);

Exit time : time\_out-varchar2(20);

DBMS MINI PROJECT

TILTLE: E MEDICINE SALES SYSTEM

**THROUGH THIS PROJECT:** This project helps to store the data of admin and students in every room and track their details. It manages the student information, room information, room allocation details, fee details and staff details of the hostel. It keeps track of the number of students in the room and availability of the room. It helps organization from the manual work from which it is very difficult to find the record of the students.

**AIM:**

To create a Java GUI based registration form which takes the values like: student ID, student name, admin name, phone number, address etc from the user. These values are to be updated in the database using JDBC connectivity.

**ARCHITECTURE AND TECHNOLOGY USED:**

**SOFTWARE USED:**

Java Eclipse, Oracle 11g Database, Java SE version 8, SQL Plus.

**Java SWING**:

SWING is a GUI widget toolkit for Java. It is part of Oracle's Java Foundation 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.

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:-

public void connectToDB()

{

try {

Connection con=DriverManager.getConnection(

"jdbc:oracle:thin:@localhost:1521:xe","sushma","vasavi");

statement=con.createStatement();

statement.executeUpdate("commit");

}

catch (SQLException connectException)

{

System.out.println(connectException.getMessage());

System.out.println(connectException.getSQLState());

System.out.println(connectException.getErrorCode());

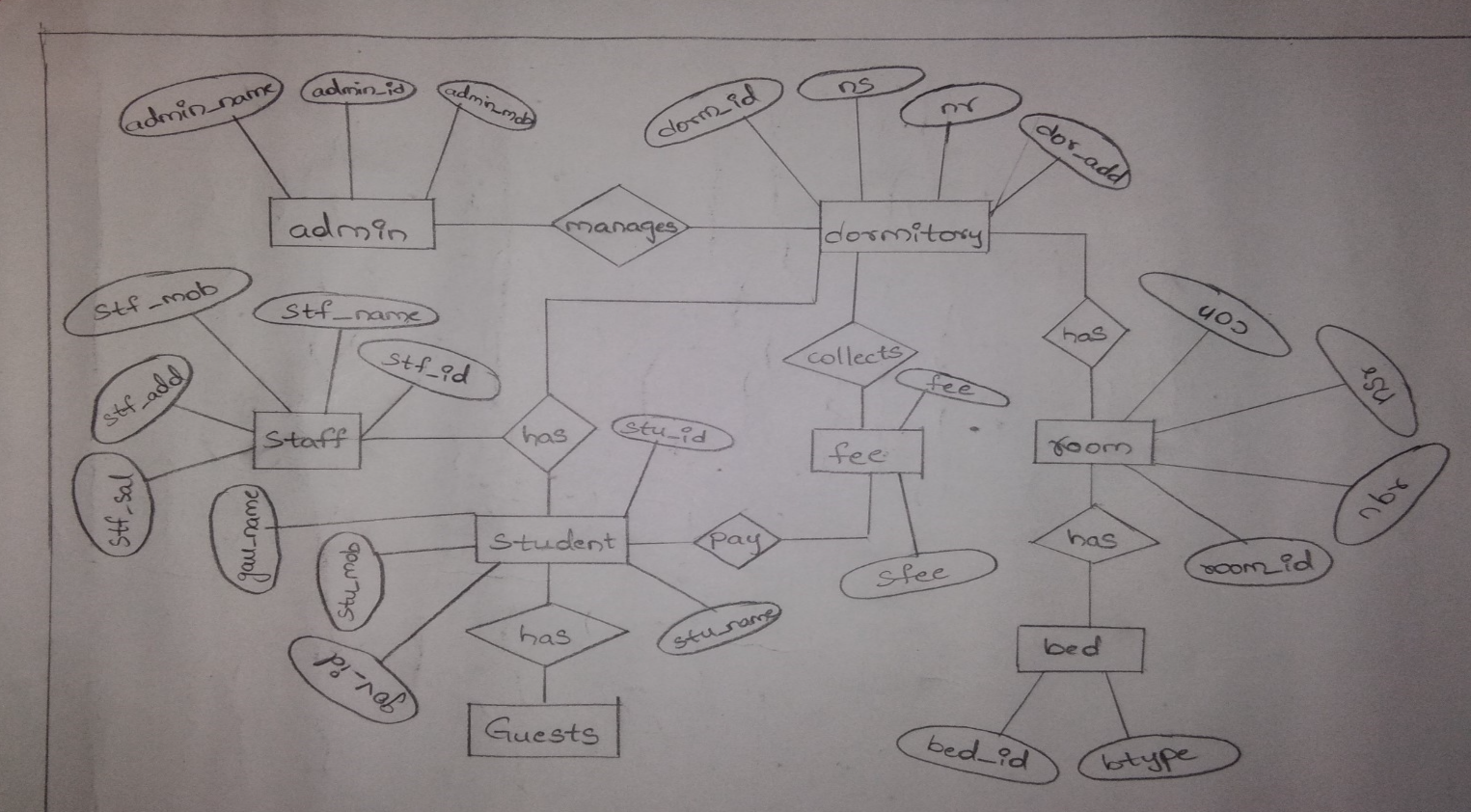
System.exit(1);

}

Thus, the connection from Java to Oracle database is performed and therefore, can be used for updating tables in the database directly.

**DESIGN:**

**ER DIAGRAM** :



ROLL NO: 1602-19-737-117

NAME: MANE SUSHMA

**DATA DESIGN:**

**Mapping Cardinalities and Participation Constraints:**

**\***Here the database administrator manages a dormitory .So the administrator is connected to dormitory by relation “manages”.So this is ont to one participation.

\*A single dormitory has many students and staffs.So these are one to many participation.

\*And there will be many rooms in dormitory for each floor.So it is one to many participation.

\*A single room can have many beds and tables .It is one to many participation.

\*For a single student there can be many visitors.So it is one to many relation.

**DDL COMMANDS:**

Table Created in SQL for above mentioned purpose is as:

**Query:**

**//following are ddl commands**

Sssd

SQL> create table admin(

2 admin\_id number(10) not null,

3 admin\_name varchar2(20) not null,

4 admin\_mob number(10));

Table created.

SQL> create table dormitory(

2 dorm\_id number(10),

3 ns number(10) not null,

4 nr number(10) not null,

5 dorm\_add varchar2(30) not null,

6 primary key(dorm\_id));

Table created.

SQL> create table room(

2 room\_id number(10),

3 nbr number(10),

4 nsr number(10),

5 con varchar2(20),

6 primary key(room\_id));

Table created.

SQL> create table stud(

2 stu\_id number(10),

3 stu\_name varchar2(20),

4 gau\_name varchar2(20),

5 stu\_mob number(20),

6 gov\_id number(20) not null,

7 room\_id number(10),

8 primary key(stu\_id),

9 FOREIGN KEY(room\_id) REFERENCES room(room\_id));

Table created.

SQL> create table staff(

2 stf\_id number(10),

3 stf\_name varchar2(20) not null,

4 stf\_mob number(20),

5 stf\_add varchar2(20),

6 sal number(10));

Table created.

SQL> create table guest(

2 dat varchar2(20) not null,

3 gname varchar2(25),

4 stu\_id number(10),

5 time\_in varchar2(20) not null,

6 time\_out varchar2(20) not null,

7 foreign key(stu\_id) references stud(stu\_id));

Table created.

SQL> create table fee(

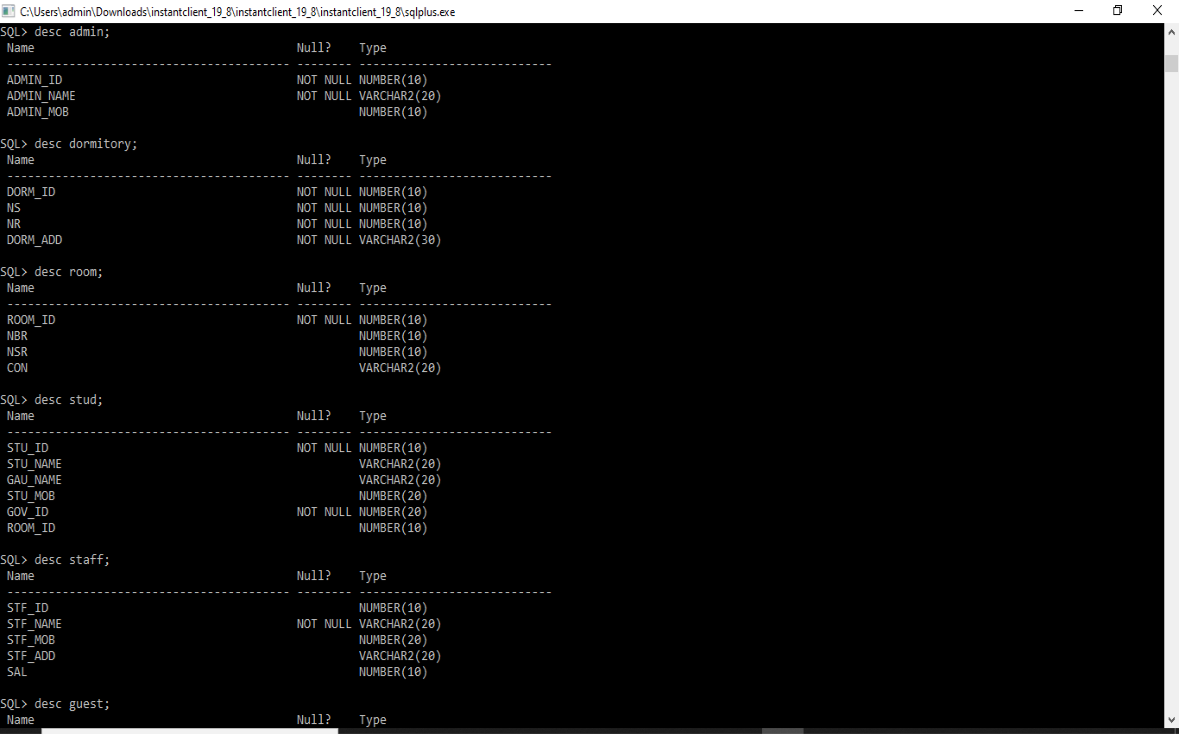
2 feem varchar2(20),

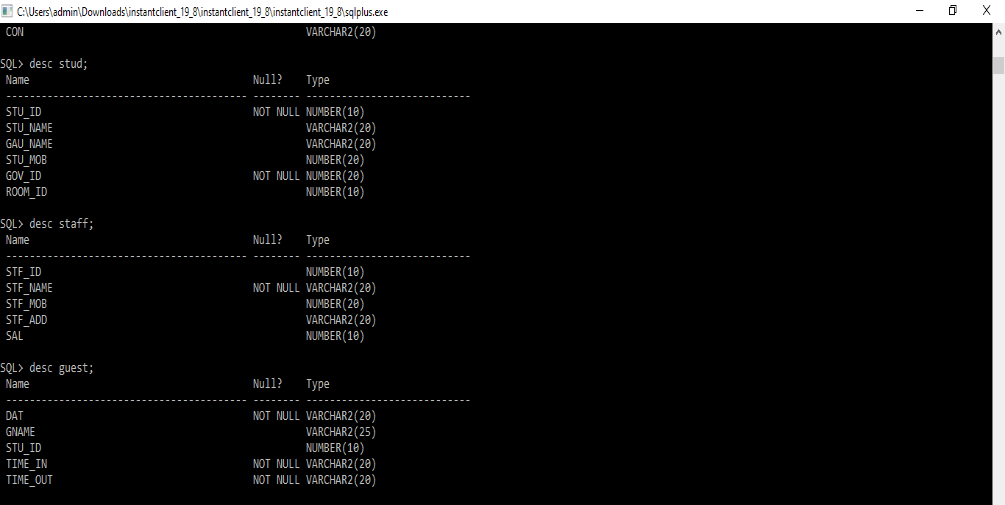
3 sfee varchar2(20),

4 stu\_id number(10),

5 foreign key(stu\_id) references stud(stu\_id));

Table created.





**Implementation:**

**Program:**

**User Interface:**

import javax.swing.\*;

import java.awt.event.\*;

import java.sql.\*;

import java.awt.Font;

import java.awt.Color;

public class STUDORM implements ActionListener {

JFrame f= new JFrame("MAIN PAGE");

JLabel l1,l2, l3, l4, l5, l6,l;

JTextField tf1,tf2,tf3,tf4,tf5,tf6;

JButton bu;

JLabel ll1,ll2, ll3, ll4, ll5, ll6,ll;

JTextField tff1,tff2,tff3,tff4,tff5,tff6;

JLabel li1,li2, li3, li4, li5, li6,li;

JTextField tif1,tif2,tif3,tif4,tif5,tif6;

JButton bu1,bu2;

JLabel L1,L2, L3, L4, L5, L6,L;

JTextField TF1,TF2,TF3,TF4,TF5,TF6;

JButton BU;

JLabel LL1,LL2, LL3, LL4, LL5, LL6,LL;

JTextField TFF1,TFF2,TFF3,TFF4,TFF5,TFF6;

JLabel LI1,LI2, LI3, LI4, LI5, LI6,LI;

JTextField TIF1,TIF2,TIF3,TIF4,TIF5,TIF6;

JButton BU1,BU2;

Connection con;Statement stmt;

JFrame p=new JFrame("admin insert");

JFrame q=new JFrame("admin modify");

JFrame r=new JFrame("admin delete");

JFrame s=new JFrame("dormitory insert");

JFrame t=new JFrame("dormitory modify");

JFrame p1=new JFrame("dormitory delete");

JFrame q1=new JFrame("insert room");

JFrame r1=new JFrame("modify room");

JFrame s1=new JFrame("delete room");

JFrame p2=new JFrame("insert student");

JFrame q2=new JFrame("modify student");

JFrame r2=new JFrame("delete student");

JFrame p3=new JFrame("insert guest");

JFrame q3=new JFrame("modify guest");

JFrame r3=new JFrame("delete guest");

JFrame p4=new JFrame("insert staff");

JFrame q4=new JFrame("modify staff");

JFrame r4=new JFrame("delete staff");

JFrame p5=new JFrame("insert fee");

JFrame q5=new JFrame("modify fee");

JFrame r5=new JFrame("delete fee");

JMenuBar mb=new JMenuBar();

JMenu menu=new JMenu("ADMIN");

JMenu menu1=new JMenu("DORMITORY");

JMenu menu2=new JMenu("ROOM");

JMenuItem i1=new JMenuItem("insert admin");

JMenuItem i2=new JMenuItem("modify admin");

JMenuItem i3=new JMenuItem("delete admin");

JMenuItem i4=new JMenuItem("insert dorm");

JMenuItem i5=new JMenuItem("modify dorm");

JMenuItem i6=new JMenuItem("delete dorm");

JMenuItem i7=new JMenuItem("insert room");

JMenuItem i8=new JMenuItem("modify room");

JMenuItem i9=new JMenuItem("delete room");

JMenu menu3=new JMenu("STUDENT");

JMenuItem a=new JMenuItem("insert student");

JMenuItem b=new JMenuItem("modify student");

JMenuItem c=new JMenuItem("delete student");

JMenu menu4=new JMenu("GUEST");

JMenuItem a1=new JMenuItem("insert guest");

JMenuItem b1=new JMenuItem("modify guest");

JMenuItem c1=new JMenuItem("delete guest");

JMenu menu5=new JMenu("STAFF");

JMenuItem a2=new JMenuItem("insert staff");

JMenuItem b2=new JMenuItem("modify staff");

JMenuItem c2=new JMenuItem("delete staff");

JMenu menu6=new JMenu("FEE");

JMenuItem a3=new JMenuItem("insert fee");

JMenuItem b3=new JMenuItem("modify fee");

JMenuItem c3=new JMenuItem("delete fee");

STUDORM(){

l1=new JLabel(" WELCOME TO STUDENTS DORMITORY ");

l1.setFont(new Font("Times New Roman", Font.BOLD, 30));

l1.setBounds(20,100,664,76);

menu.setForeground(Color.BLACK);

menu.add(i1);

menu.add(i2);

menu.add(i3);

mb.add(menu);

menu1.setForeground(Color.BLACK);

menu1.add(i4);

menu1.add(i5);

menu1.add(i6);

mb.add(menu1);

menu2.setForeground(Color.BLACK);

menu2.add(i7);

menu2.add(i8);

menu2.add(i9);

mb.add(menu2);

menu3.setForeground(Color.BLACK);

menu3.add(a);

menu3.add(b);

menu3.add(c);

mb.add(menu3);

menu4.setForeground(Color.BLACK);

menu4.add(a1);

menu4.add(b1);

menu4.add(c1);

mb.add(menu4);

menu5.setForeground(Color.BLACK);

menu5.add(a2);

menu5.add(b2);

menu5.add(c2);

mb.add(menu5);

menu6.setForeground(Color.BLACK);

menu6.add(a3);

menu6.add(b3);

menu6.add(c3);

mb.add(menu6);

f.getContentPane().setForeground(Color.GRAY);

f.getContentPane().setBackground(Color.CYAN);

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

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

q.getContentPane().setBackground(Color.PINK);

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

r.getContentPane().setBackground(Color.PINK);

f.getContentPane().add(l1);

f.setJMenuBar(mb);

f.setSize(793,486);

f.getContentPane().setLayout(null);

f.setVisible(true);

p.setSize(700,555);

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

p.getContentPane().setBackground(Color.PINK);

q.setSize(700,700);

r.setSize(700,700);

s.setSize(800,800);

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

s.getContentPane().setBackground(Color.YELLOW);

t.setSize(800,800);

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

t.getContentPane().setBackground(Color.YELLOW);

p1.setSize(800,800);

p1.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

p1.getContentPane().setBackground(Color.YELLOW);

q1.setSize(400,400);

r1.setSize(400,400);

s1.setSize(400,400);

p2.setSize(400,400);

q2.setSize(400,400);

r2.setSize(400,400);

p3.setSize(400,400);

q3.setSize(400,400);

r3.setSize(400,400);

p4.setSize(400,400);

q4.setSize(400,400);

r4.setSize(400,400);

p5.setSize(400,400);

q5.setSize(400,400);

r5.setSize(400,400);

bu=new JButton("INSERT\_ADMIN");

bu.setFont(new Font("Tahoma", Font.PLAIN, 14));

bu.setBackground(Color.WHITE);

l=new JLabel(" ADMIN INSERTION");

l.setFont(new Font("Tahoma", Font.PLAIN, 18));

l.setVerticalAlignment(SwingConstants.TOP);

l.setBounds(100, 30, 400, 30);

l2 = new JLabel("ADMIN\_Id:");

l2.setFont(new Font("Tahoma", Font.PLAIN, 14));

l3 = new JLabel("ADMIN\_Name:");

l3.setFont(new Font("Tahoma", Font.PLAIN, 14));

l4 = new JLabel("ADMIN\_MOB:");

l4.setFont(new Font("Tahoma", Font.PLAIN, 14));

tf1 = new JTextField();

tf2 = new JTextField();

tf3 = new JTextField();

l2.setBounds(80, 70, 200, 30);

l3.setBounds(80, 110, 200, 30);

l4.setBounds(80, 150, 200, 30);

tf1.setBounds(300, 70, 200, 30);

tf2.setBounds(300, 110, 200, 30);

tf3.setBounds(300, 150, 200, 30);

//tf4.setBounds(300, 190, 200, 30);

//tf5.setBounds(300, 230, 200, 30);

bu.setBounds(170, 350, 150, 30);

p.getContentPane().add(tf1);p.getContentPane().add(tf2);p.getContentPane().add(tf3);

//p.getContentPane().add(tf4);p.getContentPane().add(tf5);

p.getContentPane().add(bu);p.getContentPane().add(l2);p.getContentPane().add(l3);p.getContentPane().add(l4);

//p.getContentPane().add(l5);p.getContentPane().add(l6);

p.getContentPane().add(l);

bu.addActionListener(this);

bu1=new JButton("MODIFY\_ADMIN");

bu1.setFont(new Font("Tahoma", Font.PLAIN, 14));

bu1.setBackground(Color.WHITE);

ll=new JLabel(" ADMIN MODIFICATION");

ll.setFont(new Font("Tahoma", Font.PLAIN, 16));

ll.setVerticalAlignment(SwingConstants.TOP);

ll.setBounds(100, 30, 400, 30);

ll2 = new JLabel("ADMIN\_Id:");

ll2.setFont(new Font("Tahoma", Font.PLAIN, 14));

ll3 = new JLabel("ADMIN\_Name:");

ll3.setFont(new Font("Tahoma", Font.PLAIN, 14));

ll4 = new JLabel("ADMIN\_mob:");

ll4.setFont(new Font("Tahoma", Font.PLAIN, 14));

/\* ll5 = new JLabel("Password:");

ll5.setFont(new Font("Tahoma", Font.PLAIN, 14));

ll6 = new JLabel("Address:");

ll6.setFont(new Font("Tahoma", Font.PLAIN, 14)); \*/

tff1 = new JTextField();

tff2 = new JTextField();

tff3 = new JTextField();

// tff4 = new JTextField();

//tff5 = new JTextField();

ll2.setBounds(80, 70, 200, 30);

ll3.setBounds(80, 110, 200, 30);

ll4.setBounds(80, 150, 200, 30);

// ll5.setBounds(80, 190, 200, 30);

//ll6.setBounds(80, 230, 200, 30);

tff1.setBounds(300, 70, 200, 30);

tff2.setBounds(300, 110, 200, 30);

tff3.setBounds(300, 150, 200, 30);

//tff4.setBounds(300, 190, 200, 30);

//tff5.setBounds(300, 230, 200, 30);

bu1.setBounds(170, 350, 150, 30);

q.getContentPane().add(tff1);q.getContentPane().add(tff2);q.getContentPane().add(tff3);

//q.getContentPane().add(tff4);q.getContentPane().add(tff5);

q.getContentPane().add(bu1);q.getContentPane().add(ll2);q.getContentPane().add(ll3);q.getContentPane().add(ll4);

//q.getContentPane().add(ll5);q.getContentPane().add(ll6);

q.getContentPane().add(ll);

bu1.addActionListener(this);

bu2=new JButton("DELETE\_ADMIN");

bu2.setFont(new Font("Tahoma", Font.PLAIN, 14));

bu2.setBackground(Color.WHITE);

li=new JLabel(" ADMIN DELETION");

li.setFont(new Font("Tahoma", Font.PLAIN, 18));

li.setVerticalAlignment(SwingConstants.TOP);

li.setBounds(100, 30, 400, 30);

li2 = new JLabel("ADMIN\_Id:");

li2.setFont(new Font("Tahoma", Font.PLAIN, 14));

li3 = new JLabel("ADMIN\_Name:");

li3.setFont(new Font("Tahoma", Font.PLAIN, 14));

li4 = new JLabel("ADMIN\_mob:");

li4.setFont(new Font("Tahoma", Font.PLAIN, 14));

/\*li5 = new JLabel("Password:");

li5.setFont(new Font("Tahoma", Font.PLAIN, 14));

li6 = new JLabel("Address:");

li6.setFont(new Font("Tahoma", Font.PLAIN, 14)); \*/

tif1 = new JTextField();

tif2 = new JTextField();

tif3 = new JTextField();

//tif4 = new JTextField();

//tif5 = new JTextField();

li2.setBounds(80, 70, 200, 30);

li3.setBounds(80, 110, 200, 30);

li4.setBounds(80, 150, 200, 30);

//li5.setBounds(80, 190, 200, 30);

//li6.setBounds(80, 230, 200, 30);

tif1.setBounds(300, 70, 200, 30);

tif2.setBounds(300, 110, 200, 30);

tif3.setBounds(300, 150, 200, 30);

// tif4.setBounds(300, 190, 200, 30);

//tif5.setBounds(300, 230, 200, 30);

bu2.setBounds(170, 350, 150, 30);

r.getContentPane().add(tif1);r.getContentPane().add(tif2);r.getContentPane().add(tif3);

//r.getContentPane().add(tif4);r.getContentPane().add(tif5);

r.getContentPane().add(bu2);r.getContentPane().add(li2);r.getContentPane().add(li3);r.getContentPane().add(li4);

//r.getContentPane().add(li5);r.getContentPane().add(li6);

r.getContentPane().add(li);

bu2.addActionListener(this);

BU=new JButton("INSERT\_DORM");

BU.setFont(new Font("Tahoma", Font.PLAIN, 14));

BU.setBackground(Color.WHITE);

L=new JLabel(" DORMITORY INSERTION");

L.setFont(new Font("Tahoma", Font.PLAIN, 18));

L.setVerticalAlignment(SwingConstants.TOP);

L.setBounds(100, 30, 400, 30);

L2 = new JLabel("DORM\_Id:");

L2.setFont(new Font("Tahoma", Font.PLAIN, 14));

L3 = new JLabel("NO.OF STU");

L3.setFont(new Font("Tahoma", Font.PLAIN, 14));

L4 = new JLabel("NO.OF ROOMS:");

L4.setFont(new Font("Tahoma", Font.PLAIN, 14));

L5 = new JLabel("DORM\_ADDR:");

L5.setFont(new Font("Tahoma", Font.PLAIN, 14));

TF1 = new JTextField();

TF2 = new JTextField();

TF3 = new JTextField();

TF4 = new JTextField();

//tf5 = new JTextField();

L2.setBounds(80, 70, 200, 30);

L3.setBounds(80, 110, 200, 30);

L4.setBounds(80, 150, 200, 30);

L5.setBounds(80, 190, 200, 30);

//l6.setBounds(80, 230, 200, 30);

TF1.setBounds(300, 70, 200, 30);

TF2.setBounds(300, 110, 200, 30);

TF3.setBounds(300, 150, 200, 30);

TF4.setBounds(300, 190, 200, 30);

//tf5.setBounds(300, 230, 200, 30);

BU.setBounds(170, 350, 150, 30);

s.getContentPane().add(TF1);s.getContentPane().add(TF2);s.getContentPane().add(TF3);

s.getContentPane().add(TF4);

//p.getContentPane().add(tf5);

s.getContentPane().add(BU);s.getContentPane().add(L2);s.getContentPane().add(L3);s.getContentPane().add(L4);

s.getContentPane().add(L5);

//p.getContentPane().add(l6);

s.getContentPane().add(L);

BU.addActionListener(this);

BU1=new JButton("MODIFY\_DORM");

BU1.setFont(new Font("Tahoma", Font.PLAIN, 14));

BU1.setBackground(Color.WHITE);

LL=new JLabel(" DORM MODIFICATION");

LL.setFont(new Font("Tahoma", Font.PLAIN, 16));

LL.setVerticalAlignment(SwingConstants.TOP);

LL.setBounds(100, 30, 400, 30);

LL2 = new JLabel("DORM\_Id:");

LL2.setFont(new Font("Tahoma", Font.PLAIN, 14));

LL3 = new JLabel("NO.OF STU:");

LL3.setFont(new Font("Tahoma", Font.PLAIN, 14));

LL4 = new JLabel("NO.OF ROOMS:");

LL4.setFont(new Font("Tahoma", Font.PLAIN, 14));

LL5 = new JLabel("DORM\_ADDR");

LL5.setFont(new Font("Tahoma", Font.PLAIN, 14));

//ll6 = new JLabel("Address:");

//ll6.setFont(new Font("Tahoma", Font.PLAIN, 14)); \*/

TFF1 = new JTextField();

TFF2 = new JTextField();

TFF3 = new JTextField();

TFF4 = new JTextField();

//tff5 = new JTextField();

LL2.setBounds(80, 70, 200, 30);

LL3.setBounds(80, 110, 200, 30);

LL4.setBounds(80, 150, 200, 30);

LL5.setBounds(80, 190, 200, 30);

//ll6.setBounds(80, 230, 200, 30);

TFF1.setBounds(300, 70, 200, 30);

TFF2.setBounds(300, 110, 200, 30);

TFF3.setBounds(300, 150, 200, 30);

TFF4.setBounds(300, 190, 200, 30);

//tff5.setBounds(300, 230, 200, 30);

BU1.setBounds(170, 350, 150, 30);

t.getContentPane().add(TFF1);t.getContentPane().add(TFF2);t.getContentPane().add(TFF3);

t.getContentPane().add(TFF4);

//q.getContentPane().add(tff5);

t.getContentPane().add(BU1);t.getContentPane().add(LL2);t.getContentPane().add(LL3);t.getContentPane().add(LL4);

t.getContentPane().add(LL5);

//q.getContentPane().add(ll6);

t.getContentPane().add(LL);

BU1.addActionListener(this);

BU2=new JButton("DELETE\_dorm");

BU2.setFont(new Font("Tahoma", Font.PLAIN, 14));

BU2.setBackground(Color.WHITE);

LI=new JLabel(" DORMITORY DELETION");

LI.setFont(new Font("Tahoma", Font.PLAIN, 18));

LI.setVerticalAlignment(SwingConstants.TOP);

LI.setBounds(100, 30, 400, 30);

LI2 = new JLabel("DORM\_Id:");

LI2.setFont(new Font("Tahoma", Font.PLAIN, 14));

LI3 = new JLabel("NO.OF STU:");

LI3.setFont(new Font("Tahoma", Font.PLAIN, 14));

LI4 = new JLabel("NO.OF ROOMS:");

LI4.setFont(new Font("Tahoma", Font.PLAIN, 14));

LI5 = new JLabel("DORM\_ADDR:");

LI5.setFont(new Font("Tahoma", Font.PLAIN, 14));

//li6 = new JLabel("Address:");

//li6.setFont(new Font("Tahoma", Font.PLAIN, 14)); \*/

TIF1 = new JTextField();

TIF2 = new JTextField();

TIF3 = new JTextField();

TIF4 = new JTextField();

//tif5 = new JTextField();

LI2.setBounds(80, 70, 200, 30);

LI3.setBounds(80, 110, 200, 30);

LI4.setBounds(80, 150, 200, 30);

LI5.setBounds(80, 190, 200, 30);

//li6.setBounds(80, 230, 200, 30);

TIF1.setBounds(300, 70, 200, 30);

TIF2.setBounds(300, 110, 200, 30);

TIF3.setBounds(300, 150, 200, 30);

TIF4.setBounds(300, 190, 200, 30);

//tif5.setBounds(300, 230, 200, 30);

BU2.setBounds(170, 350, 150, 30);

p1.getContentPane().add(TIF1);p1.getContentPane().add(TIF2);p1.getContentPane().add(TIF3);

p1.getContentPane().add(TIF4);

//r.getContentPane().add(tif5);

p1.getContentPane().add(BU2);p1.getContentPane().add(LI2);p1.getContentPane().add(LI3);p1.getContentPane().add(LI4);

p1.getContentPane().add(LI5);

//r.getContentPane().add(li6);

p1.getContentPane().add(LI);

i1.addActionListener(this);

i2.addActionListener(this);

i3.addActionListener(this);

i4.addActionListener(this);

i5.addActionListener(this);

i6.addActionListener(this);

i7.addActionListener(this);

i8.addActionListener(this);

i9.addActionListener(this);

a.addActionListener(this);

b.addActionListener(this);

c.addActionListener(this);

a1.addActionListener(this);

b1.addActionListener(this);

c1.addActionListener(this);

a2.addActionListener(this);

b2.addActionListener(this);

c2.addActionListener(this);

a3.addActionListener(this);

b3.addActionListener(this);

c3.addActionListener(this);

}

public void actionPerformed(ActionEvent e) {

if(e.getSource()==i1) {

p.setVisible(true);

bu.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

{

int x = 0;

String s1 = tf1.getText();

String s2 = tf2.getText();

String s3 = tf3.getText();

//String s4 = tf4.getText();

//String s5 = tf5.getText();

if (true)

{

try

{

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

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

PreparedStatement ps = con.prepareStatement("insert into admin values(?,?,?)");

ps.setString(1, s1);

ps.setString(2, s2);

ps.setString(3, s3);

//ps.setString(4, s4);

//ps.setString(5, s5);

ResultSet rs = ps.executeQuery();

x++;

if (x > 0)

{

JOptionPane.showMessageDialog(bu, "Data Inserted Successfully");

}

}

catch (Exception ex)

{

System.out.println(ex);

}

}

else

{

JOptionPane.showMessageDialog(bu, "not possible");

}

}

}

});

}

if(e.getSource()==i2){

q.setVisible(true);

bu1.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

{

int x = 0;

String s1 = tff1.getText();

String s2 = tff2.getText();

String s3 = tff3.getText();

// String s4 = tff4.getText();

//String s5 = tff5.getText();

//String s6 = tff6.getText();

if (true)

{

try

{

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

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

String query="update admin set admin\_name=? ,admin\_mob=? "+"where admin\_id=?";

PreparedStatement ps = con.prepareStatement(query);

ps.setString(1, s1);

ps.setString(2, s2);

ps.setString(3, s3);

//ps.setString(4, s5);

//ps.setString(5, s1);

ResultSet rs = ps.executeQuery();

x++;

if (x > 0)

{

JOptionPane.showMessageDialog(bu1, "Data Updated Successfully");

}

}

catch (Exception ex)

{

System.out.println(ex);

}

}

else

{

JOptionPane.showMessageDialog(bu1, "not possible");

}

}

}

});

}

if(e.getSource()==i3) {

r.setVisible(true);

String s1 = tif1.getText();

bu2.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

{

int x = 0;

//String s1 = tif1.getText();

if (true)

{

try

{

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

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

String query="delete from admin where admin\_id=?";

PreparedStatement ps = con.prepareStatement(query);

ps.setString(1, s1);

ResultSet rs = ps.executeQuery();

x++;

if (x > 0)

{

JOptionPane.showMessageDialog(bu2, "Data Deleted Successfully");

}

//conn.close();

}

catch (Exception ex)

{

System.out.println(ex);

}

}

else

{

JOptionPane.showMessageDialog(bu2, "not possible");

}

}

}

});

}

//public void actionPerformed(ActionEvent e) {

if(e.getSource()==i4) {

s.setVisible(true);

BU.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

{

int x = 0;

String s11 = TF1.getText();

String s22 = TF2.getText();

String s33 = TF3.getText();

String s44 = TF4.getText();

//String s5 = tf5.getText();

if (true)

{

try

{

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

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

PreparedStatement ps = con.prepareStatement("insert into dormitory values(?,?,?,?)");

ps.setString(1, s11);

ps.setString(2, s22);

ps.setString(3, s33);

ps.setString(4, s44);

//ps.setString(5, s5);

ResultSet rs = ps.executeQuery();

x++;

if (x > 0)

{

JOptionPane.showMessageDialog(BU, "Data Inserted Successfully");

}

}

catch (Exception ex1)

{

System.out.println(ex1);

}

}

else

{

JOptionPane.showMessageDialog(BU, "not possible");

}

}

}

});

}

if(e.getSource()==i5) {

t.setVisible(true);

BU1.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

{

int x = 0;

String s111 = TFF1.getText();

String s222 = TFF2.getText();

String s333 = TFF3.getText();

String s444 = TFF4.getText();

//String s5 = tff5.getText();

//String s6 = tff6.getText();

if (true)

{

try

{

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

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

String query="update dormitory set ns=? , nr =?, dorm\_add=? "+"where DORM\_ID=?";

PreparedStatement ps = con.prepareStatement(query);

ps.setString(1, s111);

ps.setString(2, s222);

ps.setString(3, s333);

ps.setString(4, s444);

//ps.setString(5, s1);

ResultSet rs = ps.executeQuery();

x++;

if (x > 0)

{

JOptionPane.showMessageDialog(BU1, "Data Updated Successfully");

}

}

catch (Exception ex2)

{

System.out.println(ex2);

}

}

else

{

JOptionPane.showMessageDialog(BU1, "not possible");

}

}

}

});

}

if(e.getSource()==i6) {

p1.setVisible(true);

String s1 = TIF1.getText();

BU2.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent arg0) {

{

int x = 0;

if (true)

{

try

{

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

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

String query="delete from dormitory where dorm\_id=?";

PreparedStatement ps = con.prepareStatement(query);

ps.setString(1, s1);

ResultSet rs = ps.executeQuery();

x++;

if (x > 0)

{

JOptionPane.showMessageDialog(BU2, "Data Deleted Successfully");

}

}

catch (Exception ex3)

{

System.out.println(ex3);

}

}

else

{

JOptionPane.showMessageDialog(BU2, "not possible");

}

}

}

});

}

if(e.getSource()==i7)

q1.setVisible(true);

if(e.getSource()==i8)

r1.setVisible(true);

if(e.getSource()==i9)

s1.setVisible(true);

if(e.getSource()==a)

p2.setVisible(true);

if(e.getSource()==b)

q2.setVisible(true);

if(e.getSource()==c)

r2.setVisible(true);

if(e.getSource()==a1)

p3.setVisible(true);

if(e.getSource()==b1)

q3.setVisible(true);

if(e.getSource()==c1)

r3.setVisible(true);

if(e.getSource()==a2)

p4.setVisible(true);

if(e.getSource()==b2)

q4.setVisible(true);

if(e.getSource()==c2)

r4.setVisible(true);

}

public static void main(String[] args) {

new STUDORM();

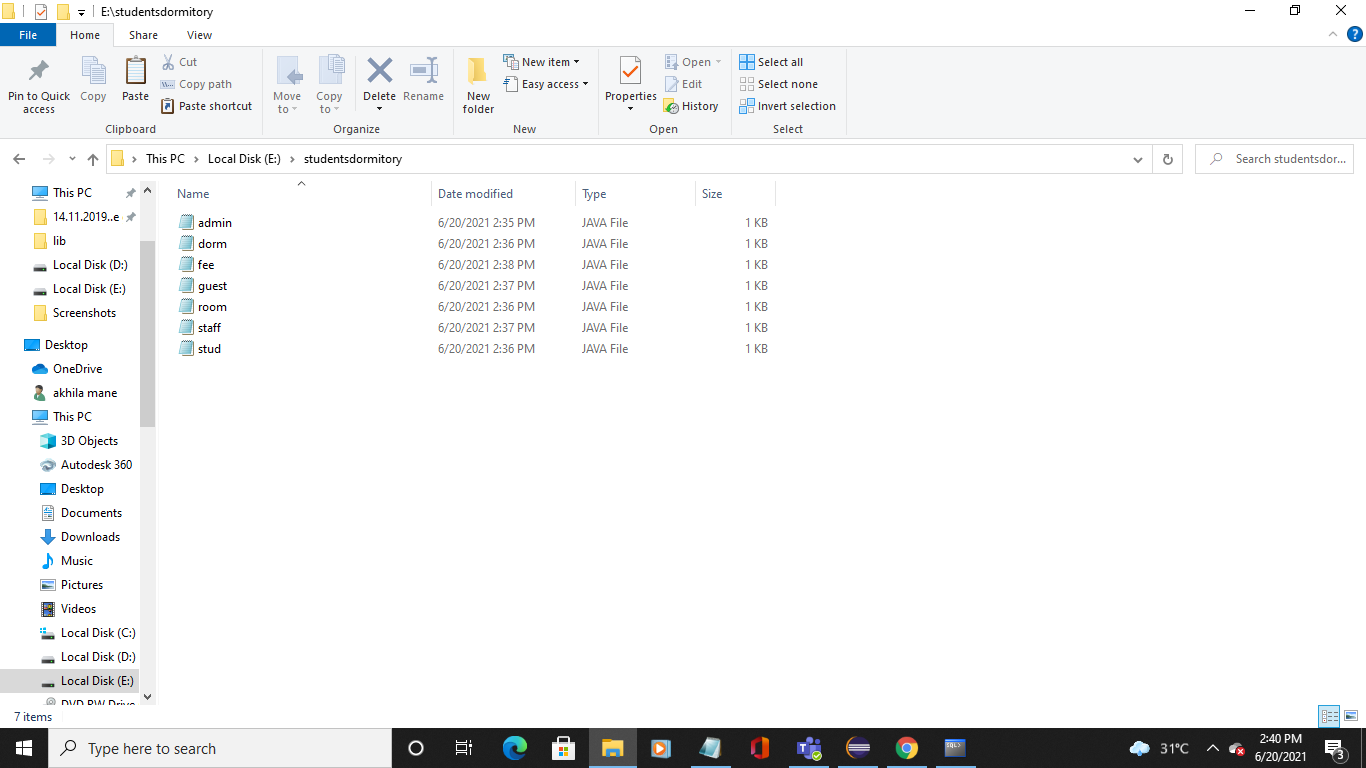
}

}

**GITHUB LINK:**

https://github.com/Sushmamane/DBMS\_StudentsDormitory.git

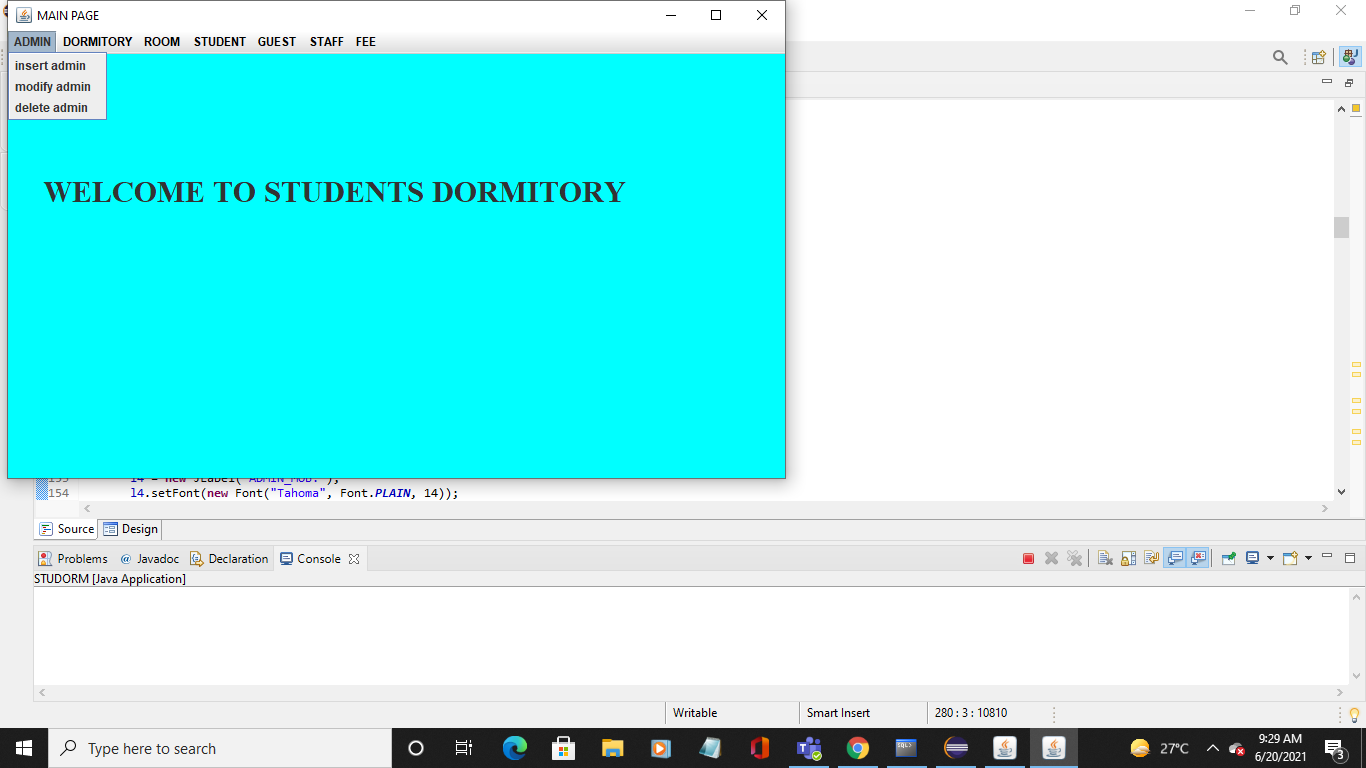
**FOLDER STRUCTURE:**

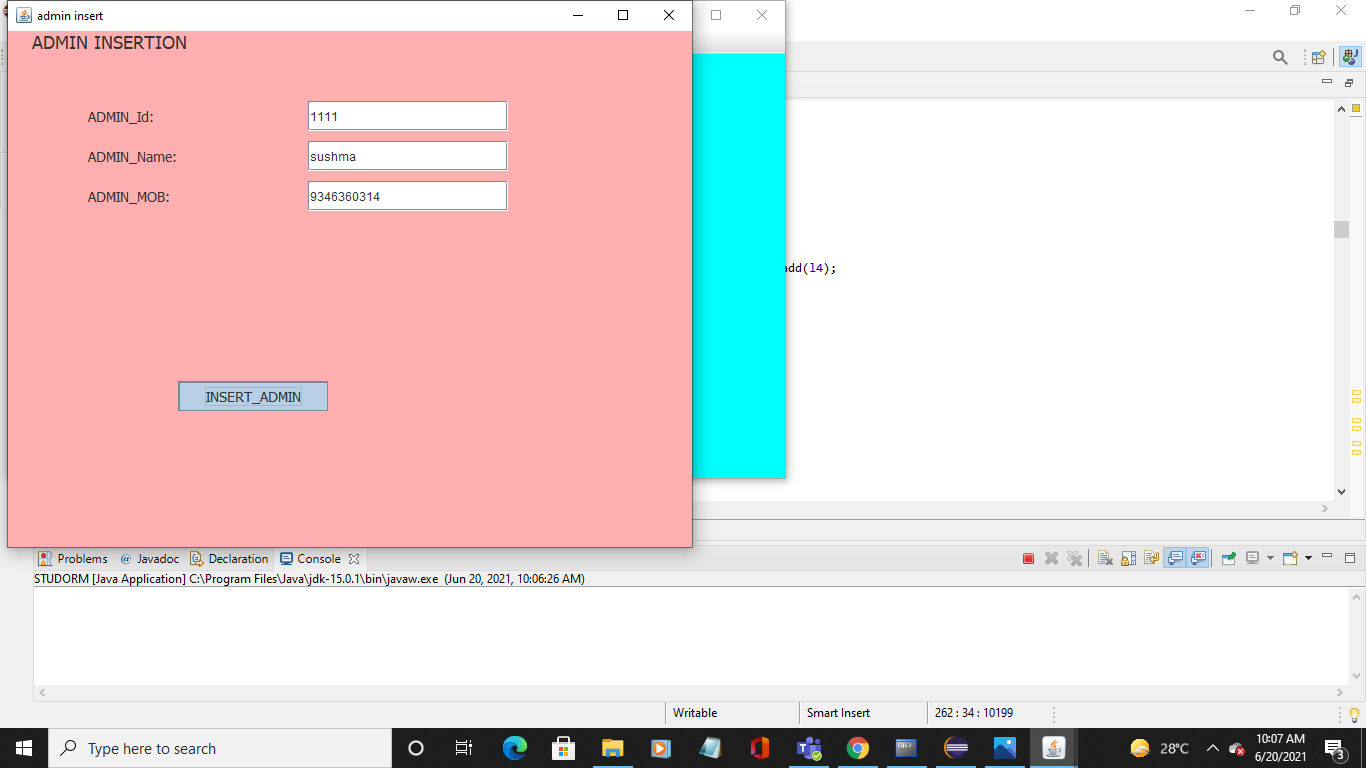


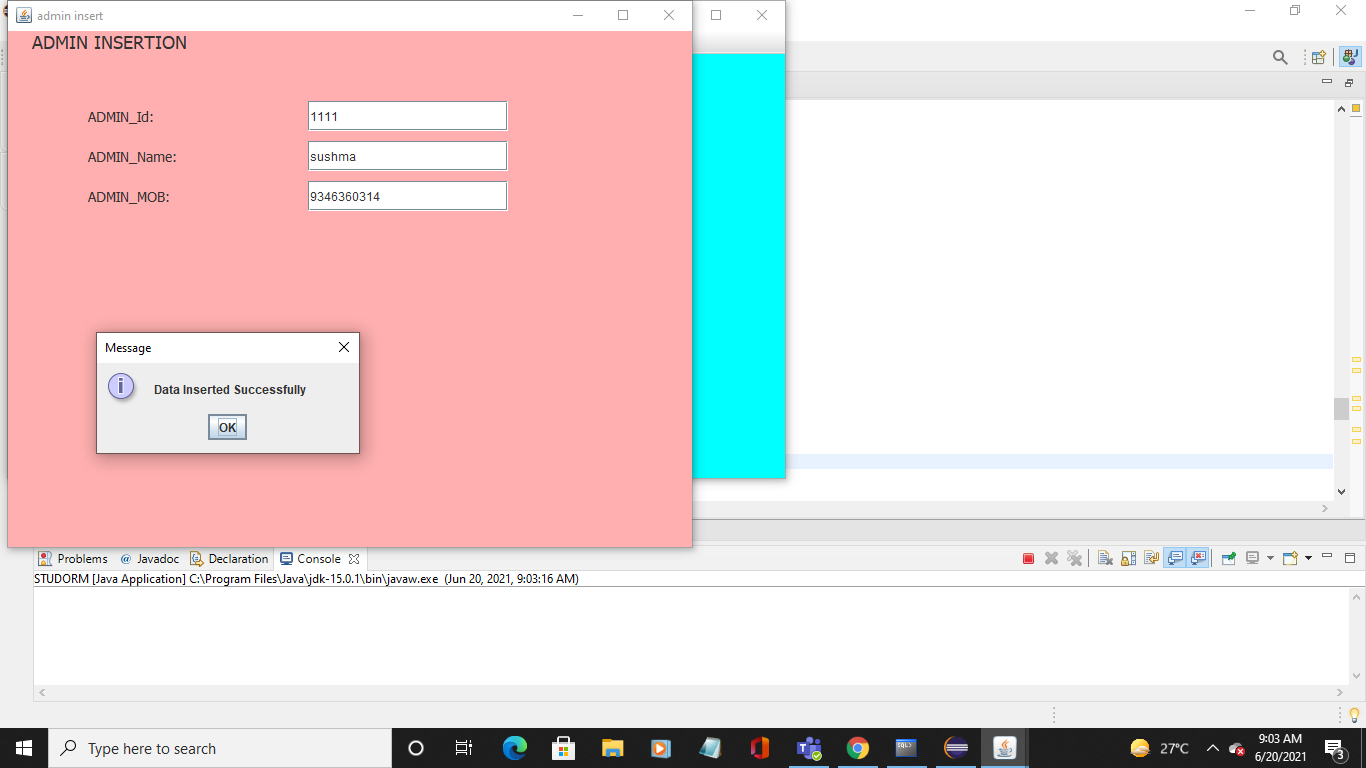
**Testing:**

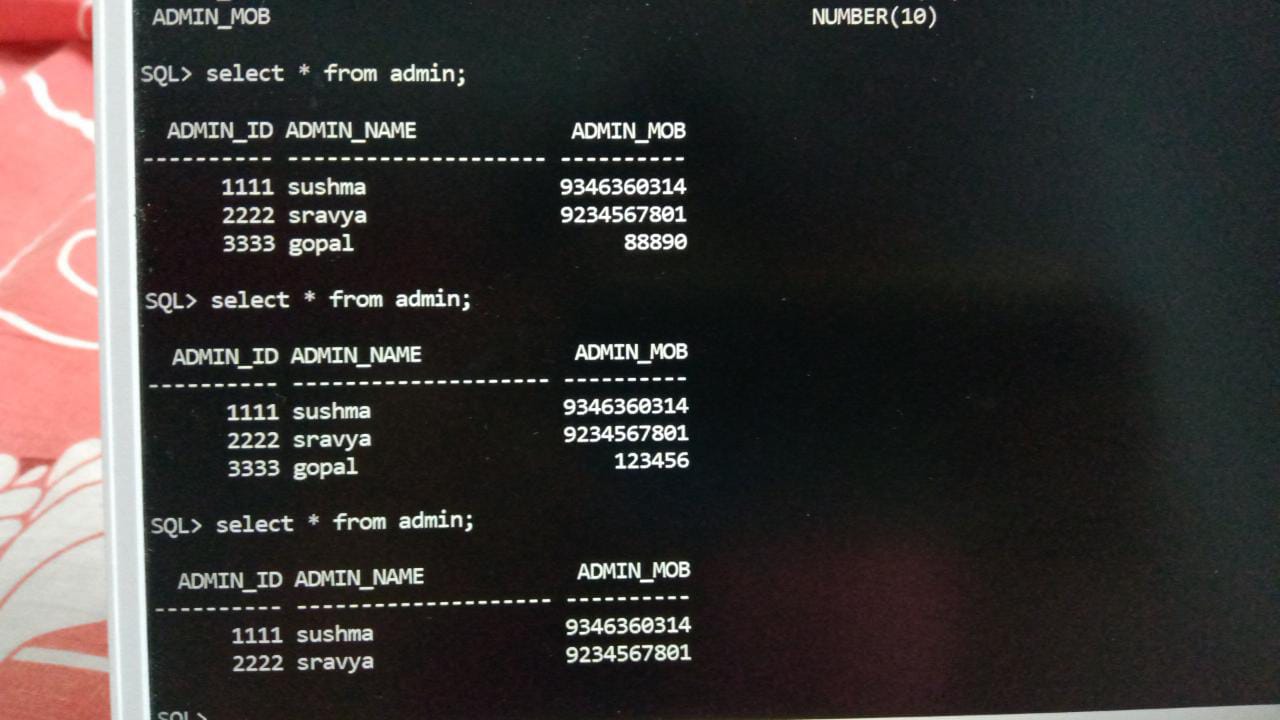
**OUTPUT SCREENSHOTS:**

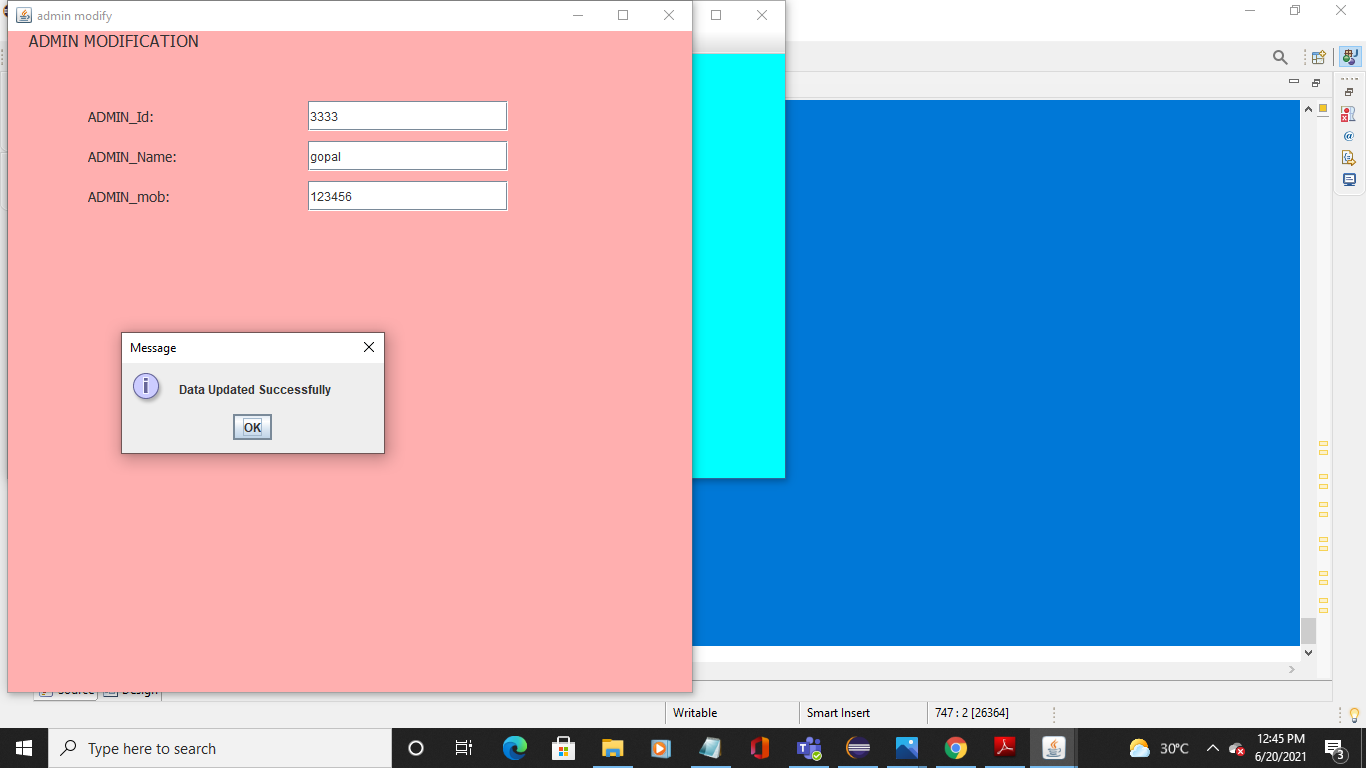
**Java GUI Screenshot:**

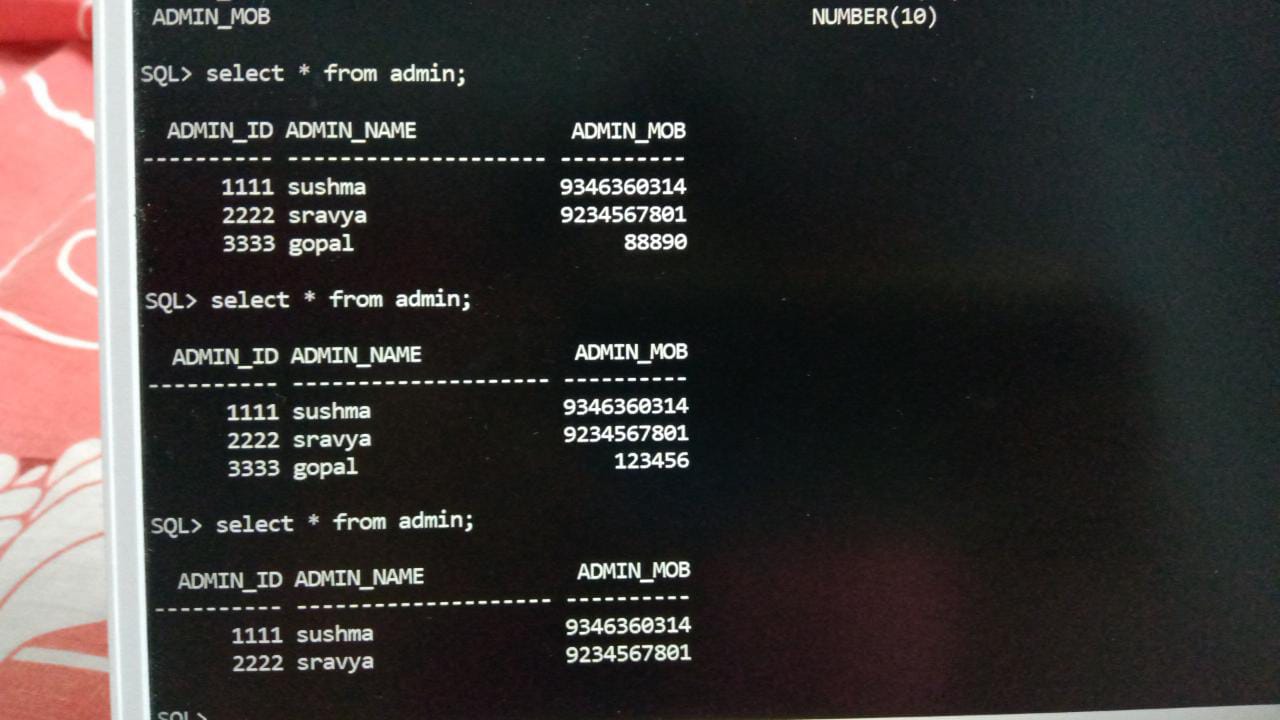


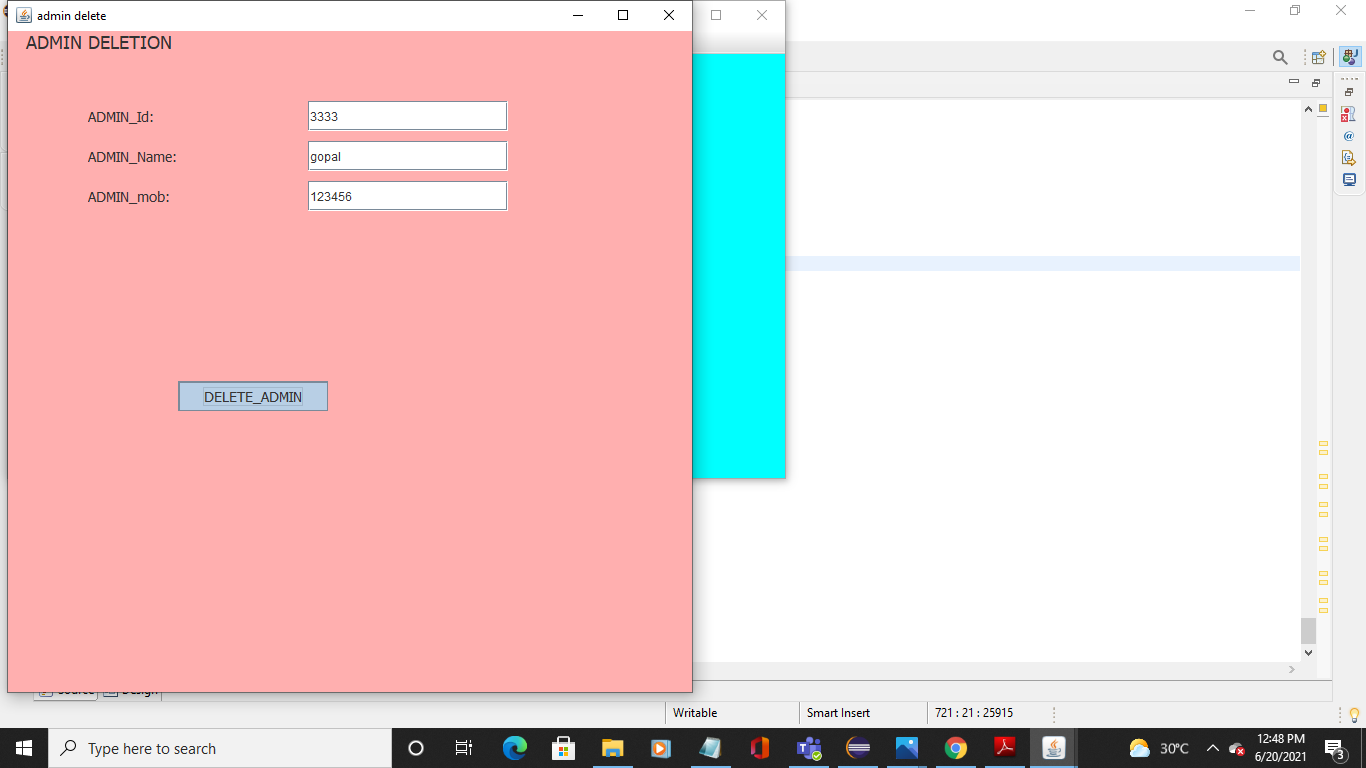


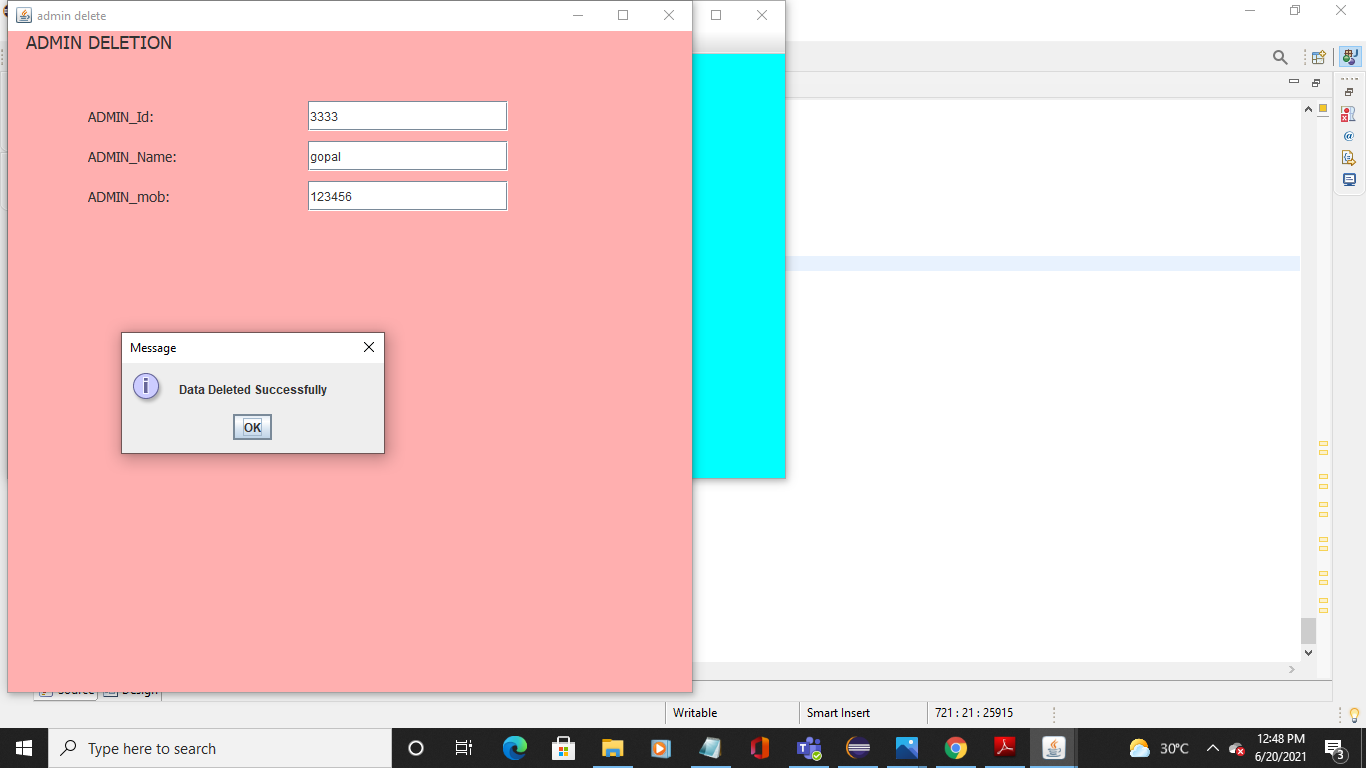


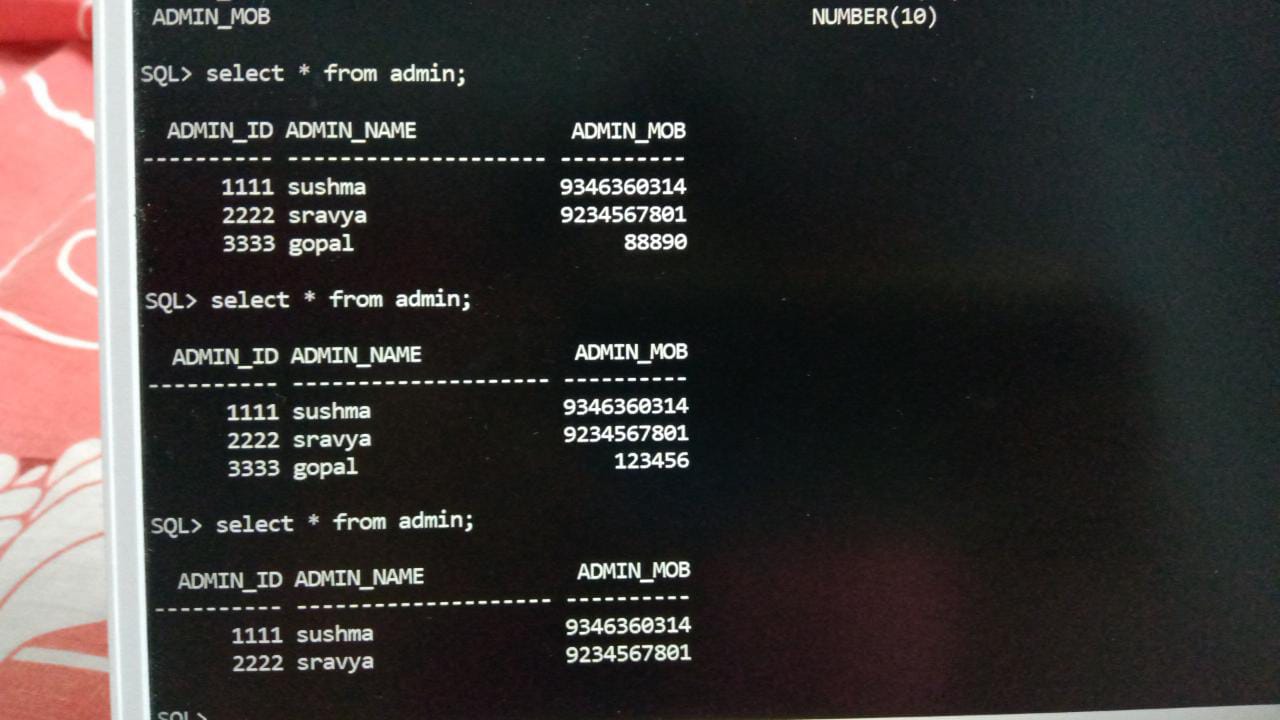


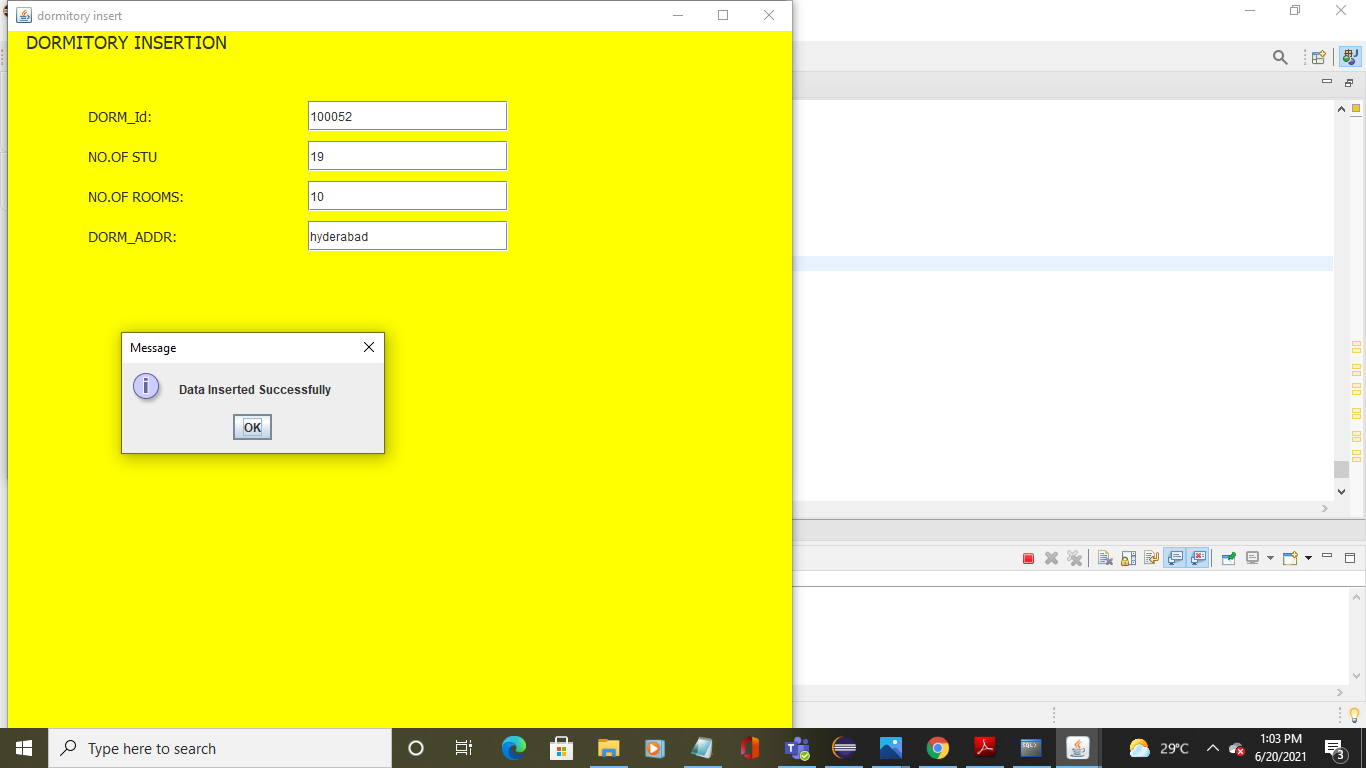


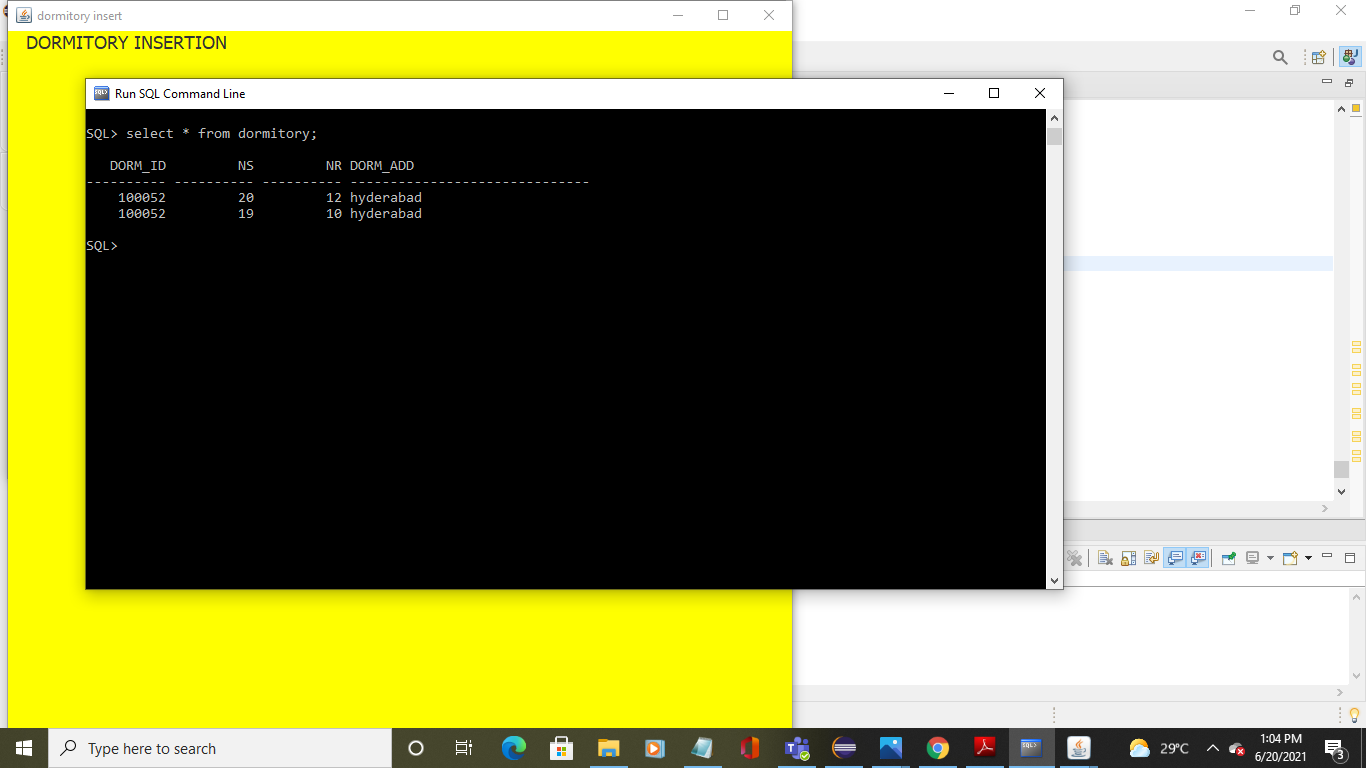


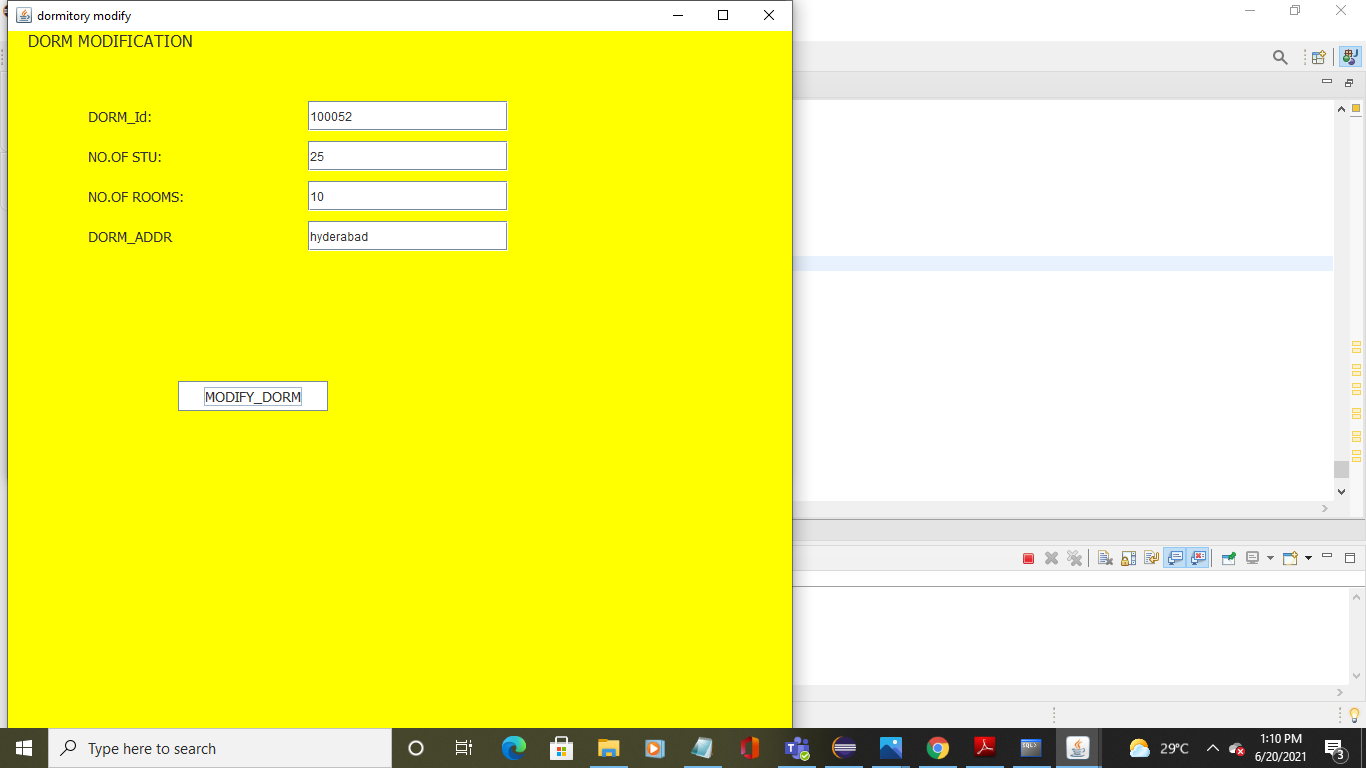


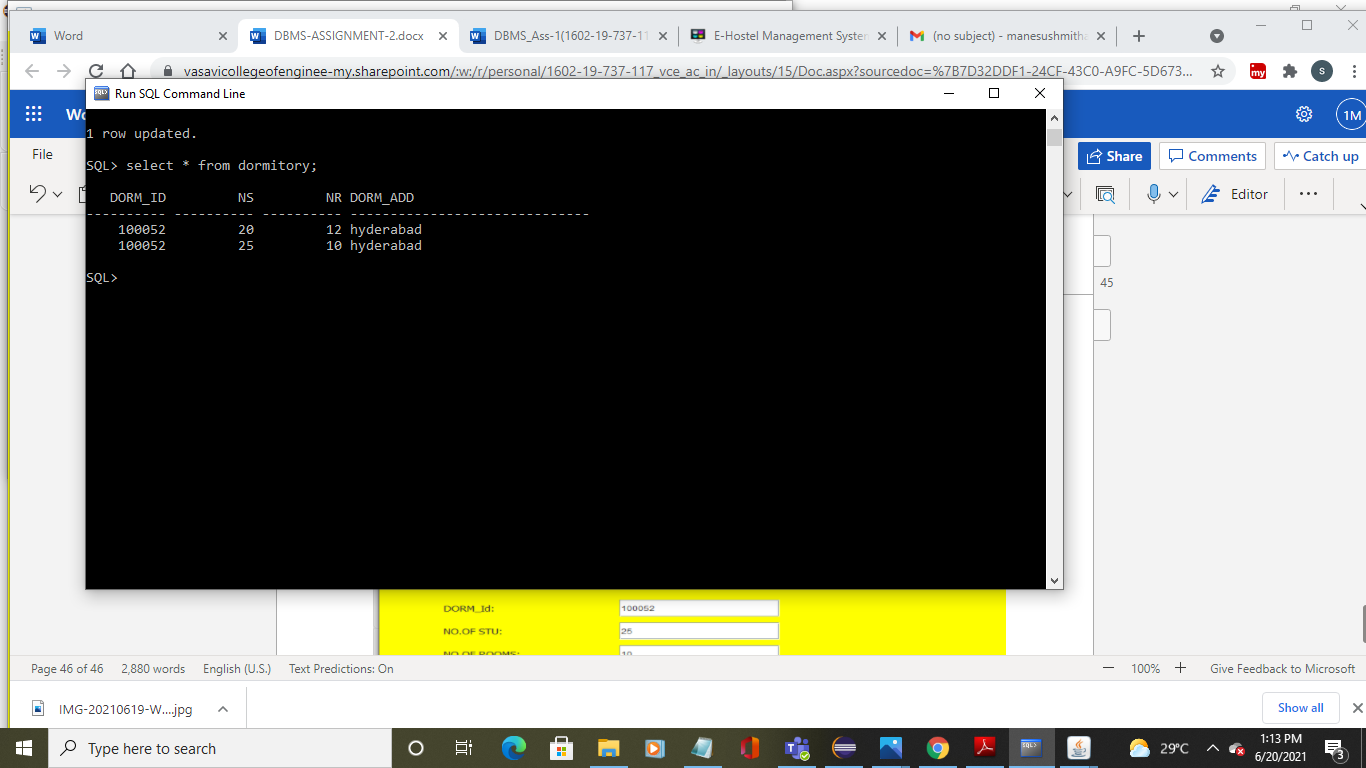


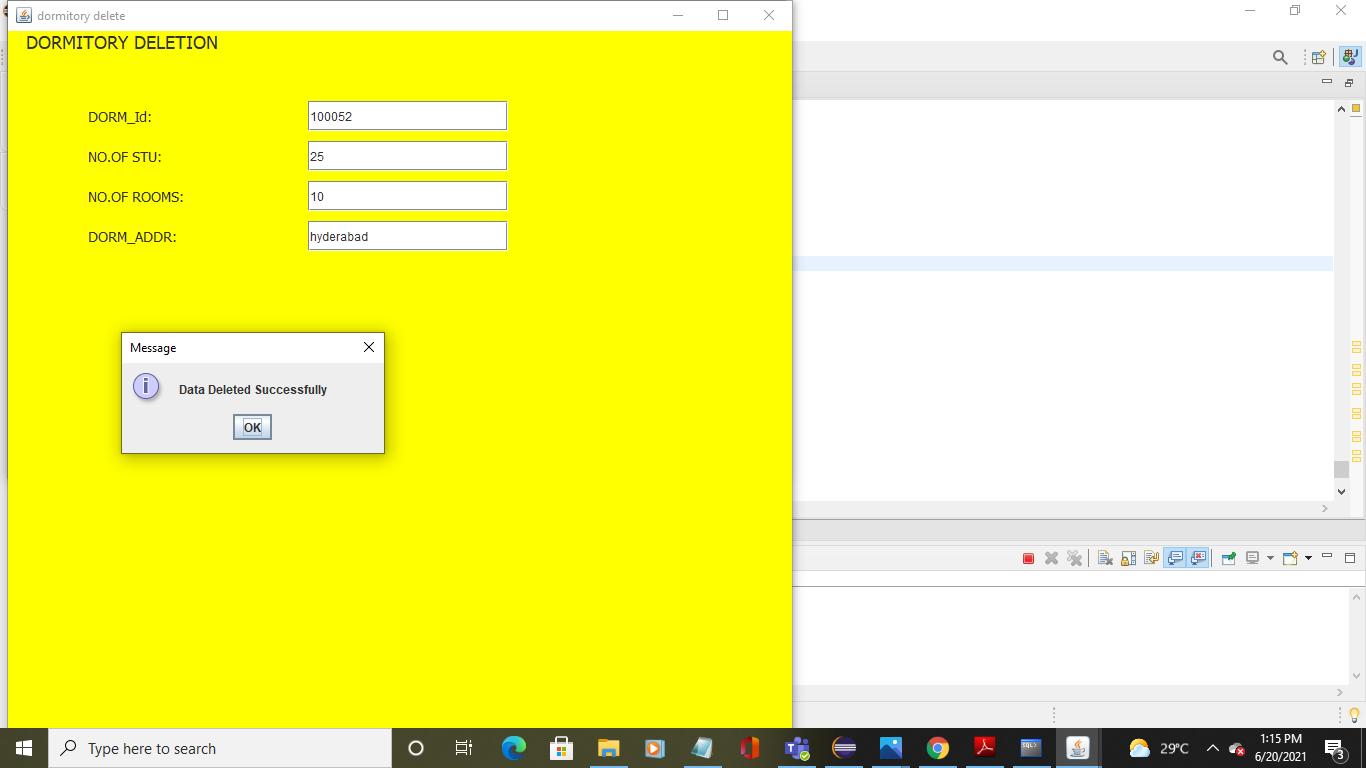


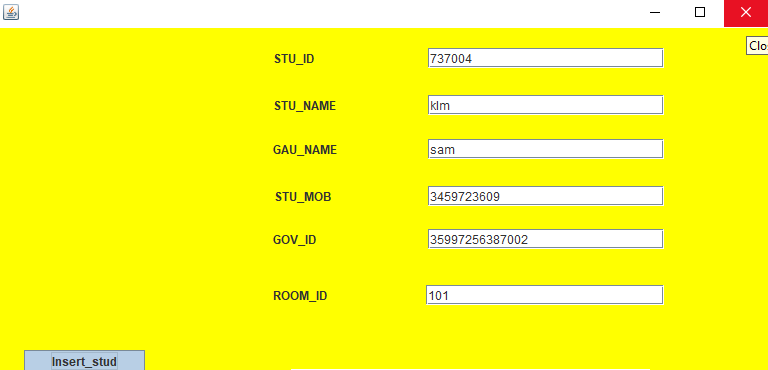


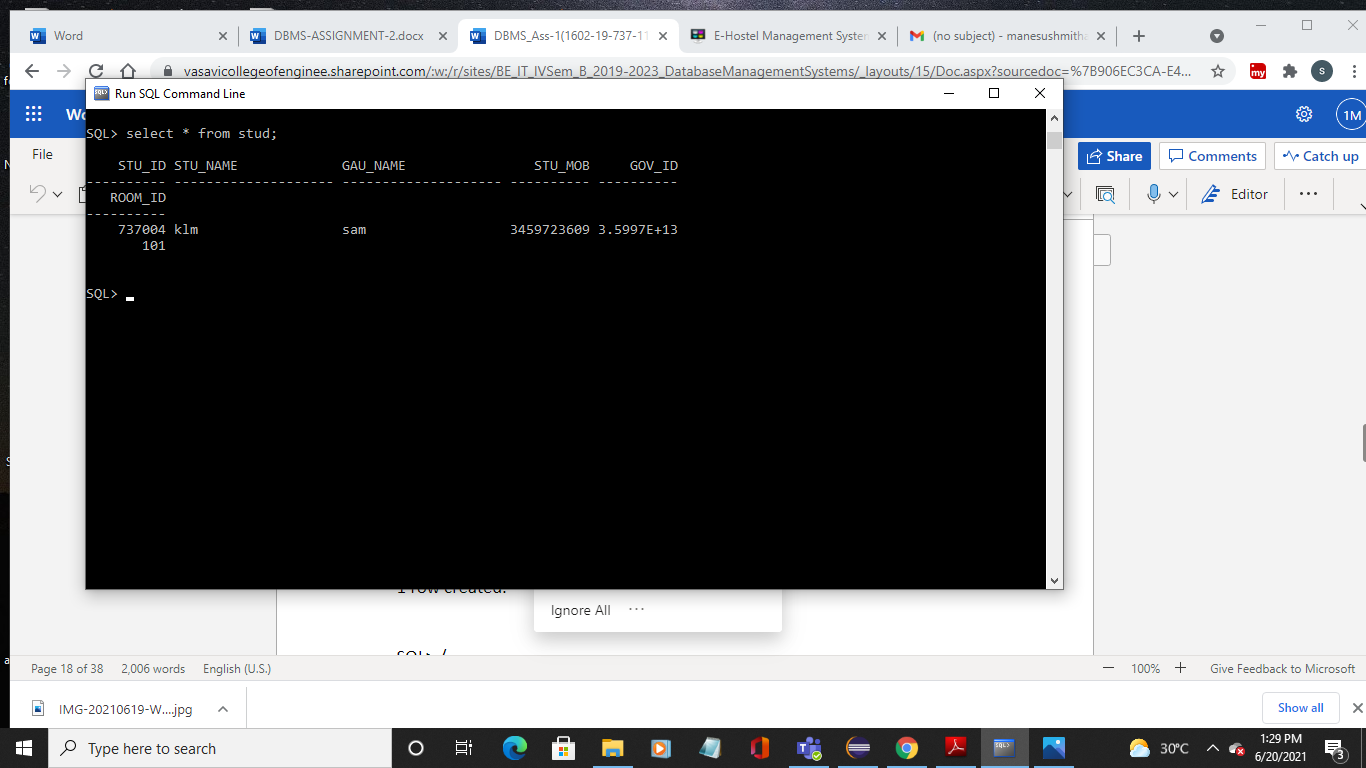












**RESULT:**

I had successfully completed MINI PROJECT on “STUDENTS DORMITORY”.

**Discussion and future Work:**

It is easy to extend the system that we have proposed. A person could see any of the issued, unissued or all the rooms according to his/her will. In future we can implement some features for “STUDENTS DORMITORY” project. In this system its possible to categorize room rent for middle class students and poor students. Some poor students are given a particular concession for the entire year.Even wages of staff can be calculated every month.

**CONCLUSION:**

Thus, a Java SWING based STUDENTS DORMITORY is created which is connected to the Oracle 11g database. Therefore, all the entries and details are directly updated on their respective tables created in the database.

**REFERENCES:**

▪ https://docs.oracle.com/javase/7/docs/api/

▪ https://www.javatpoint.com/dbms-tutorial

▪ https://youtu.be/obmNZNP8euU