JAVA AWT BASED- VCE NETWORK HARDWARE BASE - 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

P.ANOOP <1602-18-737-065>

Department of Information Technology

Vasavi College of Engineering (Autonomous)

Ibrahimbagh, Hyderabad-31

2020

BONAFIDE CERTIFICATE

This to Certify that the project report titled "VCE NETWORK HARDWARE BASE" project work of PANDIRI ANOOP bearing Roll.no: 1602-18-737-065 who carried out this project under my supervision in the IV semester for the academic year 2019-2020.

Signature

external examiner

Signature

internal examiner

ABSTRACT

VCE Network Hardware Base is a database that consists information about various Networking Hardware like Router, Switch, Hub, Bridge. The database contains both data (wired and wireless) in the College Campus. Network Hardware Base is important both in ensuring the correct operation of network devices and in maintaining the services that run on them. It describes how the network is being connected in our college across the various blocks. When you enter the data it is stored in the data base and is displayed as of when it is needed..

INTRODUCTION

➢ REQUIREMENTS FOR VCE NETWORK HARDWARE BASE :

List of tables :

- Block
- Lab
- Computers
- Lan
- Wan
- Internet

List of attributes with their domain types:

ENTITY	ATTRIBUTES	DOMAIN
BLOCK	1. bname 2. branch 3. hod	Varchar2(20) Varchar2(20) Varchar2(20)
LAB	1. Labname 2. Floor	Varchar2(20) Number(10)
Computers	1. cid 2. Manufacturer 3. MAC_ADDRESS	Number(10) Varchar2(20) Varchar2(20)
LAN	1. l_device_id 2. l_device_name 3. l_speed 4. l_ip_address	Number(10) Varchar2(20) Varchar2(10) Varchar2(20)
WAN	1. w_device_id 2. w_device_name 3. w_speed 4. w_ip_address	Number(10) Varchar2(20) Varchar2(10) Varchar2(20)
Internet	1. ISP 2. Website	Varchar2(20) Varchar2(20)

MAPPING CARDINALITIES :

- WAN is interconnection of LANs therefore one to many cardinalities between WAN and LAN.

- Computers are mandatory in LAB therefore many to one mapping cardinalities between a Computer and LAB.

- There is no rule that a Block should have Labs therefore one to many mapping cardinalities between Block and Lab.

➢ SPECIFIC GOAL OF THE PROJECT:

The main goal to be achieved through this project was to provide a facility to the College to display the details of various Network Hardware used to provide connection to Labs in various Blocks.

This Application is used to Store data of various Network Hardware like Switches, Routers, HUBs and Computers.

➢ Architecture and technology used:

SQL Plus is the most basic Oracle Database utility with a basic command-line interface, commonly used by users, administrators and programmers.

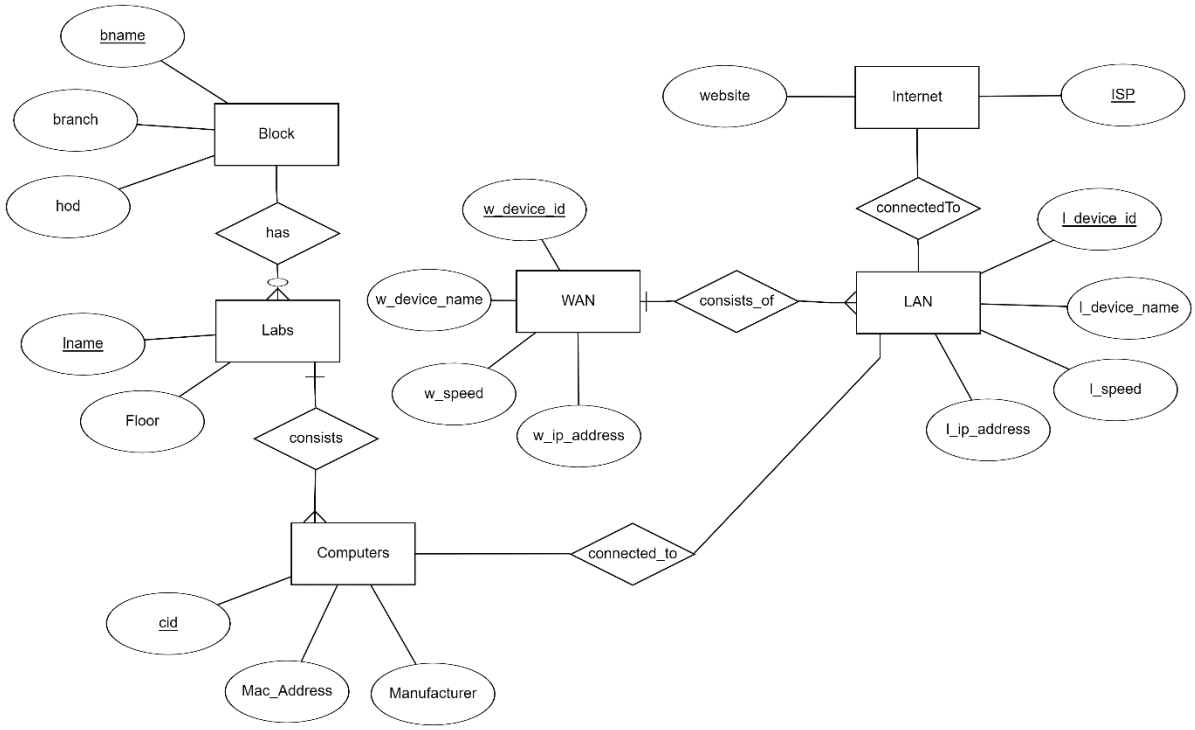
The interface of SQL Plus is used for creating the database. DDL and DML commands are implemented for operations being executed. The details of various Blocks, Labs, Computers, Network devices are stored in the form of tables in the database.

Eclipse is an Integrated development environment (IDE) used in computer programming. It contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in java and its primary use is for developing Java applications, but it may also be used to develop applications in other programming languages via plug-ins, including Erlang, JavaScripts etc.

The front end application code is written in "**Java**" using Eclipse. The portal for front end application is designed through Eclipse, runs and has the capacity to connect with the database which has data inserted using SQL.

➢ DESIGN:

i)ER DIAGRAM:



P.ANOOP
1602-18-737-065

DDL COMMANDS :

```
SQL*Plus: Release 11.2.0.1.0 Production on Sun Feb 23 20:16:02 2020

Copyright (c) 1982, 2010, Oracle.  All rights reserved.

Enter user-name: it18737065
Enter password:

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> create table INTERNET(ISP_name varchar2(20) PRIMARY KEY,website varchar2(20));

Table created.

SQL> create table LAN(l_device_id Number(10) PRIMARY KEY,l_device_name varchar2(20),l_speed varchar2(10),l_ip_address varchar2(20));

Table created.

SQL> create table WAN(w_device_id Number(10) PRIMARY KEY,w_device_name varchar2(20),w_speed varchar2(10),w_ip_address varchar2(20));

Table created.

SQL> create table COMPUTERS(cid Number(10) PRIMARY KEY,Manufacturer varchar2(20),MAC_ADDRESS Varchar2(20));

Table created.

SQL> create table LABS(LabName  varchar2(20) PRIMARY KEY,Floor Number(10));

Table created.

SQL> create table BLOCK(bname varchar2(20) PRIMARY KEY,branch varchar2(20),hod varchar2(20));

Table created.


SQL> alter table INTERNET rename column ISP_Name to ISP;

Table altered.


SQL> create table connectedTo(ISP varchar2(20),l_device_id Number(10),FOREIGN KEY(ISP) references INTERNET,FOREIGN KEY(l_device_id) references LAN);

Table created.

SQL> create table consists_of(l_device_id Number(10),w_device_id Number(10),FOREIGN KEY(w_device_id) references WAN,FOREIGN KEY(l_device_id) references LAN);

Table created.

SQL> create table connected_to(l_device_id Number(10),w_device_id Number(10),FOREIGN KEY(l_device_id) references LAN,FOREIGN KEY(w_device_id) references WAN);

Table created.

SQL> create table consists(cid Number(10),labname varchar2(20),FOREIGN KEY(cid) references COMPUTERS,FOREIGN KEY(labname) references labs);

Table created.

SQL> create table has(bname varchar2(20),labname varchar2(20),FOREIGN KEY(bname) references BLOCK,FOREIGN KEY(labname) references LABS);

Table created.

SQL> select * from tab;

TNAME                          TABTYPE  CLUSTERID
------------------------------ ------- ----------
BLOCK                          TABLE
COMPUTERS                      TABLE
CONNECTEDTO                    TABLE
CONNECTED_TO                   TABLE
CONSISTS                       TABLE
CONSISTS_OF                    TABLE
HAS                            TABLE
INTERNET                       TABLE
LABS                           TABLE
LAN                            TABLE
WAN                            TABLE

11 rows selected.
```

```
SQL> desc INTERNET;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 ISP                                       NOT NULL VARCHAR2(20)
 WEBSITE                                            VARCHAR2(20)

SQL> desc LAN;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 L_DEVICE_ID                               NOT NULL NUMBER(10)
 L_DEVICE_NAME                                      VARCHAR2(20)
 L_SPEED                                            VARCHAR2(10)
 L_IP_ADDRES$                                       VARCHAR2(20)

SQL> desc WAN;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 W_DEVICE_ID                               NOT NULL NUMBER(10)
 W_DEVICE_NAME                                      VARCHAR2(20)
 W_SPEED                                            VARCHAR2(10)
 W_IP_ADDRES$                                       VARCHAR2(20)

SQL> desc COMPUTERS;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 CID                                       NOT NULL NUMBER(10)
 MANUFACTURER                                       VARCHAR2(20)
 MAC_ADDRES$                                        VARCHAR2(20)

SQL> desc LABS;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 LABNAME                                   NOT NULL VARCHAR2(20)
 FLOOR                                              NUMBER(10)

SQL> desc BLOCK;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 BNAME                                     NOT NULL VARCHAR2(20)
 BRANCH                                             VARCHAR2(20)
 HOD                                                VARCHAR2(20)

SQL> desc connectedTo;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 ISP                                                VARCHAR2(20)
 L_DEVICE_ID                                        NUMBER(10)

SQL> desc consists_of;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 L_DEVICE_ID                                        NUMBER(10)
 W_DEVICE_ID                                        NUMBER(10)

SQL> desc connected_to;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 L_DEVICE_ID                                        NUMBER(10)
 W_DEVICE_ID                                        NUMBER(10)

SQL> desc consists;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 CID                                                NUMBER(10)
 LABNAME                                            VARCHAR2(20)

SQL> desc has;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 BNAME                                              VARCHAR2(20)
 LABNAME                                            VARCHAR2(20)

SQL> alter table connected_to rename column W_DEVICE_ID to cid;

Table altered.

SQL> desc connected_to;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 L_DEVICE_ID                                        NUMBER(10)
 CID                                                NUMBER(10)
```

DML COMMANDS :

```
SQL> INSERT INTO INTERNET VALUES('&ISP','&WEBSITE');
Enter value for isp: ACT
Enter value for website: www.actcorp.in
old   1: INSERT INTO INTERNET VALUES('&ISP','&WEBSITE')
new   1: INSERT INTO INTERNET VALUES('ACT','www.actcorp.in')

1 row created.

SQL> /
Enter value for isp: HATHWAY
Enter value for website: www.hathway.com
old   1: INSERT INTO INTERNET VALUES('&ISP','&WEBSITE')
new   1: INSERT INTO INTERNET VALUES('HATHWAY','www.hathway.com')

1 row created.

SQL> /
Enter value for isp: GTPL
Enter value for website: www.gtpl.net
old   1: INSERT INTO INTERNET VALUES('&ISP','&WEBSITE')
new   1: INSERT INTO INTERNET VALUES('GTPL','www.gtpl.net')

1 row created.

SQL> /
Enter value for isp: MTNL
Enter value for website: www.mtnl.net.in
old   1: INSERT INTO INTERNET VALUES('&ISP','&WEBSITE')
new   1: INSERT INTO INTERNET VALUES('MTNL','www.mtnl.net.in')

1 row created.

SQL> /
Enter value for isp: YOU
Enter value for website: www.youbroadband.in
old   1: INSERT INTO INTERNET VALUES('&ISP','&WEBSITE')
new   1: INSERT INTO INTERNET VALUES('YOU','www.youbroadband.in')

1 row created.

SQL> commit;

Commit complete.

SQL> INSERT INTO LAN VALUES(&l_device_id,'&l_device_name','&l_speed','&l_ip_address');
Enter value for l_device_id: 101
Enter value for l_device_name: Router1
Enter value for l_speed: 1425MBPS
Enter value for l_ip_address: 192.168.1.0
old   1: INSERT INTO LAN VALUES(&l_device_id,'&l_device_name','&l_speed','&l_ip_address')
new   1: INSERT INTO LAN VALUES(101,'Router1','1425MBPS','192.168.1.0')

1 row created.

SQL> /
Enter value for l_device_id: 102
Enter value for l_device_name: HUB1
Enter value for l_speed: 800MBPS
Enter value for l_ip_address: 192.168.1.1
old   1: INSERT INTO LAN VALUES(&l_device_id,'&l_device_name','&l_speed','&l_ip_address')
new   1: INSERT INTO LAN VALUES(102,'HUB1','800MBPS','192.168.1.1')

1 row created.

SQL> INSERT INTO WAN VALUES(&w_device_id,'&w_device_name','&w_speed','&w_ip_address');
Enter value for w_device_id: 201
Enter value for w_device_name: Switch1
Enter value for w_speed: 1800MBPS
Enter value for w_ip_address: 192.168.1.2
old   1: INSERT INTO WAN VALUES(&w_device_id,'&w_device_name','&w_speed','&w_ip_address')
new   1: INSERT INTO WAN VALUES(201,'Switch1','1800MBPS','192.168.1.2')

1 row created.

SQL> /
Enter value for w_device_id: 202
Enter value for w_device_name: Switch2
Enter value for w_speed: 1450MBPS
Enter value for w_ip_address: 192.168.1.3
old   1: INSERT INTO WAN VALUES(&w_device_id,'&w_device_name','&w_speed','&w_ip_address')
new   1: INSERT INTO WAN VALUES(202,'Switch2','1450MBPS','192.168.1.3')

1 row created.

SQL> commit;

Commit complete.
```

```
SQL> INSERT INTO COMPUTERS VALUES(&cid,'&Manufacturer','&mac_address');
Enter value for cid: 1
Enter value for manufacturer: DELL
Enter value for mac_address: 00-14-22-01-23-45
old   1: INSERT INTO COMPUTERS VALUES(&cid,'&Manufacturer','&mac_address')
new   1: INSERT INTO COMPUTERS VALUES(1,'DELL','00-14-22-01-23-45')

1 row created.

SQL> /
Enter value for cid: 2
Enter value for manufacturer: DELL
Enter value for mac_address: 14-CC-20-12-08-E2
old   1: INSERT INTO COMPUTERS VALUES(&cid,'&Manufacturer','&mac_address')
new   1: INSERT INTO COMPUTERS VALUES(2,'DELL','14-CC-20-12-08-E2')

1 row created.

SQL> /
Enter value for cid: 3
Enter value for manufacturer: DELL
Enter value for mac_address: 21-09-H1-25-01-F2
old   1: INSERT INTO COMPUTERS VALUES(&cid,'&Manufacturer','&mac_address')
new   1: INSERT INTO COMPUTERS VALUES(3,'DELL','21-09-H1-25-01-F2')

1 row created.

SQL> /
Enter value for cid: 4
Enter value for manufacturer: HP
Enter value for mac_address: 34-15-Z2-13-25-Y2
old   1: INSERT INTO COMPUTERS VALUES(&cid,'&Manufacturer','&mac_address')
new   1: INSERT INTO COMPUTERS VALUES(4,'HP','34-15-Z2-13-25-Y2')

1 row created.

SQL> /
Enter value for cid: 5
Enter value for manufacturer: HP
Enter value for mac_address: 58-24-R3-PR-01-24
old   1: INSERT INTO COMPUTERS VALUES(&cid,'&Manufacturer','&mac_address')
new   1: INSERT INTO COMPUTERS VALUES(5,'HP','58-24-R3-PR-01-24')

1 row created.

SQL> commit;

Commit complete.

SQL> INSERT INTO LABS VALUES('&LabName',&FLOOR);
Enter value for labname: IT LAB-1
Enter value for floor: 0
old   1: INSERT INTO LABS VALUES('&LabName',&FLOOR)
new   1: INSERT INTO LABS VALUES('IT LAB-1',0)

1 row created.

SQL> /
Enter value for labname: IT LAB-2
Enter value for floor: 0
old   1: INSERT INTO LABS VALUES('&LabName',&FLOOR)
new   1: INSERT INTO LABS VALUES('IT LAB-2',0)

1 row created.

SQL> /
Enter value for labname: IT LAB-3
Enter value for floor: 0
old   1: INSERT INTO LABS VALUES('&LabName',&FLOOR)
new   1: INSERT INTO LABS VALUES('IT LAB-3',0)

1 row created.

SQL> /
Enter value for labname: PROJECT LAB
Enter value for floor: 1
old   1: INSERT INTO LABS VALUES('&LabName',&FLOOR)
new   1: INSERT INTO LABS VALUES('PROJECT LAB',1)

1 row created.

SQL> commit
  2  ;

Commit complete.

SQL> INSERT INTO BLOCK VALUES('&bname','&branch','&hod');
Enter value for bname: RAMANUJAN
Enter value for branch: IT
Enter value for hod: RamMohanRao
old   1: INSERT INTO BLOCK VALUES('&bname','&branch','&hod')
new   1: INSERT INTO BLOCK VALUES('RAMANUJAN','IT','RamMohanRao')

1 row created.

SQL> commit;

Commit complete.

SQL> INSERT INTO connectedTo VALUES('&ISP',&l_device_id);
Enter value for isp: ACT
Enter value for l_device_id: 101
old   1: INSERT INTO connectedTo VALUES('&ISP',&l_device_id)
new   1: INSERT INTO connectedTo VALUES('ACT',101)

1 row created.

SQL> /
Enter value for isp: ACT
Enter value for l_device_id: 102
old   1: INSERT INTO connectedTo VALUES('&ISP',&l_device_id)
new   1: INSERT INTO connectedTo VALUES('ACT',102)

1 row created.

SQL> commit;

Commit complete.
```

```
SQL> select * from INTERNET;

ISP                  WEBSITE
-------------------- --------------------
ACT                  www.actcorp.in
HATHWAY              www.hathway.com
GTPL                 www.gtpl.net
MTNL                 www.mtnl.net.in
YOU                  www.youbroadband.in

SQL> select * from LAN;

L_DEVICE_ID L_DEVICE_NAME        L_SPEED    L_IP_ADDRESS
----------- -------------------- ---------- --------------------
        101 Router1              1425MBPS   192.168.1.0
        102 HUB1                 800MBPS    192.168.1.1

SQL> select * from WAN;

W_DEVICE_ID W_DEVICE_NAME        W_SPEED    W_IP_ADDRESS
----------- -------------------- ---------- --------------------
        201 Switch1              1800MBPS   192.168.1.2
        202 Switch2              1450MBPS   192.168.1.3

SQL> select * from COMPUTERS;

       CID MANUFACTURER         MAC_ADDRESS
---------- -------------------- --------------------
         1 DELL                 00-14-22-01-23-45
         2 DELL                 14-CC-20-12-08-E2
         3 DELL                 21-09-H1-25-01-F2
         4 HP                   34-15-Z2-13-25-V2
         5 HP                   58-24-R3-PR-01-24

SQL> select * from LABS;

LABNAME                   FLOOR
-------------------- ----------
IT LAB-1                      0
IT LAB-2                      0
IT LAB-3                      0
PROJECT LAB                   1

SQL> select * from BLOCK;

BNAME                BRANCH               HOD
-------------------- -------------------- --------------------
RAMANUJAN            IT                   RamMohanRao
```

Implementation

- JAVA Code : (Computers Table)

1)Insert Computer:

```
package vce.ac.in;

import java.awt.*;
import java.awt.event.*;
import java.sql.*;
public class AddComputer extends Panel
{
	Button AddComputerButton;
	TextField cidText, mnfText, macText;
	TextArea errorText;
	Connection connection;
	Statement statement;
	public AddComputer()
	{
		try
		{
			Class.forName("oracle.jdbc.driver.OracleDriver");
		}
		catch (Exception e)
		{
			System.err.println("Unable to find and load driver");
			System.exit(1);
		}
		connectToDB();
	}

	public void connectToDB()
    {
		try
		{
		  connection =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:orcl","it18737065","
vasavi");
		  statement = connection.createStatement();

		}
		catch (SQLException connectException)
		{
		  System.out.println(connectException.getMessage());
		  System.out.println(connectException.getSQLState());
		  System.out.println(connectException.getErrorCode());
		  System.exit(1);
		}
    }
	public void buildGUI()
	{
		//Handle Insert Button
		AddComputerButton = new Button("Submit");
		AddComputerButton.addActionListener(new ActionListener()
		{
			public void actionPerformed(ActionEvent e)
```

```
                    {
                            try
                            {
                              Statement statement = connection.createStatement();
                              //String query = "INSERT INTO COMPUTERS
(CID,MANUFACTURER,MAC_ADDRESS) VALUES (1,'DELL','00-14-22-01-23-45')";

                              String query= "INSERT INTO COMPUTERS VALUES(" +
cidText.getText()+", " + "'" + mnfText.getText() + "'," +"'"+ macText.getText()
+"')";
                              int i = statement.executeUpdate(query);
                              errorText.append("\nInserted " + i + " rows
successfully");
                            }
                            catch (SQLException insertException)
                            {
                              displaySQLErrors(insertException);
                            }
                    }
              });


              cidText = new TextField(15);
              mnfText = new TextField(15);
              macText = new TextField(15);


              errorText = new TextArea(10, 40);
              errorText.setEditable(false);

              Panel first = new Panel();
              first.setLayout(new GridLayout(4, 2));
              first.add(new Label("System No.:"));
              first.add(cidText);
              first.add(new Label("Manufacturer:"));
              first.add(mnfText);
              first.add(new Label("MAC ADDRESS:"));
              first.add(macText);
              first.setBounds(125,90,200,100);

              Panel second = new Panel(new GridLayout(4, 1));
              second.add(AddComputerButton);
          second.setBounds(125,220,150,100);

              Panel third = new Panel();
              third.add(errorText);
              third.setBounds(125,320,300,200);

              setLayout(null);

              add(first);
              add(second);
              add(third);

              setSize(500, 600);
              setVisible(true);

              errorText.setBackground(Color.BLACK);
              errorText.setForeground(Color.WHITE);
```

```
	}

	private void displaySQLErrors(SQLException e)
	{
		errorText.append("\nSQLException: " + e.getMessage() + "\n");
		errorText.append("SQLState:     " + e.getSQLState() + "\n");
		errorText.append("VendorError:  " + e.getErrorCode() + "\n");
	}

	public static void main(String[] args)
	{
		AddComputer adc = new AddComputer();
		adc.buildGUI();
	}
}
```

2)View Computer (Update and Delete) :

```
package vce.ac.in;

import java.awt.*;
import java.awt.event.*;
import java.sql.*;
public class ViewComputer extends Panel
{
	Button updateCompButton,removeCompButton;
	List compList;
	TextField cidText, mnfText, macText;
	TextArea errorText;
	Connection connection;
	Statement statement;
	ResultSet rs;

	public ViewComputer()
	{
		try
		{
			Class.forName("oracle.jdbc.driver.OracleDriver");
		}
		catch (Exception e)
		{
			System.err.println("Unable to find and load driver");
			System.exit(1);
		}
		connectToDB();
	}

	public void connectToDB()
    {
		try
		{
		  connection =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:orcl","it18737065","
vasavi");
		  statement = connection.createStatement();
```

```
            }
            catch (SQLException connectException)
            {
              System.out.println(connectException.getMessage());
              System.out.println(connectException.getSQLState());
              System.out.println(connectException.getErrorCode());
              System.exit(1);
            }
    }

      public void loadComputers()
      {
            try
            {
                  compList.removeAll();
              rs = statement.executeQuery("SELECT cid FROM COMPUTERS");
              while (rs.next())
              {
                    compList.add(rs.getString("cid"));
              }
            }
            catch (SQLException e)
            {
              displaySQLErrors(e);
            }
      }

      public void buildGUI()
      {
            compList = new List(10);
            loadComputers();
            add(compList);

            //When a list item is selected populate the text fields
            compList.addItemListener(new ItemListener()
            {
                  public void itemStateChanged(ItemEvent e)
                  {
                        try
                        {
                              rs = statement.executeQuery("SELECT * FROM
COMPUTERS where cid ="+compList.getSelectedItem());
                              rs.next();
                              cidText.setText(rs.getString("cid"));
                              mnfText.setText(rs.getString("manufacturer"));
                              macText.setText(rs.getString("mac_address"));
                        }
                        catch (SQLException selectException)
                        {
                              displaySQLErrors(selectException);
                        }
                  }
            });


            //Handle Update Computer Button
            updateCompButton = new Button("Modify");
            updateCompButton.addActionListener(new ActionListener()
```

```
		{
			public void actionPerformed(ActionEvent e)
			{
				try
				{
					Statement statement =
connection.createStatement();
					int i = statement.executeUpdate("UPDATE COMPUTERS
SET MANUFACTURER ='"+mnfText.getText()+"',"+ "
MAC_ADDRESS='"+macText.getText()+"'"+" WHERE cid ="+compList.getSelectedItem());
					errorText.append("\nUpdated " + i + " rows
successfully");
					//compList.removeAll();
					loadComputers();
				}
				catch (SQLException insertException)
				{
					displaySQLErrors(insertException);
				}
			}
		});

		//Handle Delete Computer Button
				removeCompButton = new Button("Delete");
				removeCompButton.addActionListener(new ActionListener()
				{
					public void actionPerformed(ActionEvent e)
					{
						try
						{
							Statement statement =
connection.createStatement();
							int i =
statement.executeUpdate("DELETE FROM COMPUTERS WHERE cid = "+
compList.getSelectedItem());
							errorText.append("\nDeleted " + i +
" rows successfully");
							cidText.setText(null);
							mnfText.setText(null);
							macText.setText(null);
							loadComputers();
						}
						catch (SQLException insertException)
						{
							displaySQLErrors(insertException);
						}
					}
				});

		cidText = new TextField(15);
		cidText.setEditable(false);
		mnfText = new TextField(15);
		macText = new TextField(15);

		errorText = new TextArea(10, 40);
		errorText.setEditable(false);

		Panel first = new Panel();
		first.setLayout(new GridLayout(4, 2));
```

```
                first.add(new Label("System no.:"));
                first.add(cidText);
                first.add(new Label("MANUFACTURER:"));
                first.add(mnfText);
                first.add(new Label("MAC_ADDRESS:"));
                first.add(macText);

                Panel second = new Panel(new GridLayout(4, 1));
                second.add(updateCompButton);
                second.add(removeCompButton);

                Panel third = new Panel();
                third.add(errorText);

                add(first);
                add(second);
                add(third);

                errorText.setBackground(Color.BLACK);
                errorText.setForeground(Color.WHITE);

                setSize(500, 600);
                setLayout(new FlowLayout());
                setVisible(true);

        }

        private void displaySQLErrors(SQLException e)
        {
                errorText.append("\nSQLException: " + e.getMessage() + "\n");
                errorText.append("SQLState:     " + e.getSQLState() + "\n");
                errorText.append("VendorError:  " + e.getErrorCode() + "\n");
        }

        public static void main(String[] args)
        {
                ViewComputer vc = new ViewComputer();
                vc.buildGUI();
        }
}
```

3)MAIN :

```
package vce.ac.in;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

class VNHBGUI extends Frame implements ActionListener
{
         String msg = "";
         Label ll;
         CardLayout cardLO;

         //Create Panels for each of the menu items, welcome screen panel and home
screen panel with CardLayout
         AddBlock adb;
```

```
        AddLab adl;
        AddComputer adc;
        AddLan adlan;
        AddWan adwan;

        ViewBlock vb;
        ViewLab vl;
        ViewComputer vc;
        ViewLan vlan;
        ViewWan vwan;

        ManageLabs ml;
        ManageComputers mc;
        ManageLan mlan;
        ManageWan mwan;

        Panel home,welcome;

        VNHBGUI()
        {
                    cardLO = new CardLayout();

                    //Create an empty home panel and set its layout to card layout
                    home = new Panel();
                    home.setLayout(cardLO);


                    ll = new Label();
                    ll.setAlignment(Label.CENTER);
                    ll.setText("VCE Network Hardware Base");

                    //Create welcome panel and add the label to it
                    welcome = new Panel();
                    welcome.add(ll);

                    //create panels for each of our menu items and build them with
respective components
                    adb = new AddBlock();            adb.buildGUI();
                    adl = new AddLab();              adl.buildGUI();
                    adc = new AddComputer();  adc.buildGUI();
                    adlan=new AddLan();              adlan.buildGUI();
                    adwan=new AddWan();              adwan.buildGUI();


                    vb = new ViewBlock();     vb.buildGUI();
                    vl = new ViewLab();       vl.buildGUI();
                    vc = new ViewComputer(); vc.buildGUI();
                    vlan=new ViewLan();              vlan.buildGUI();
                    vwan=new ViewWan();              vwan.buildGUI();

                    ml = new ManageLabs(); ml.buildGUI();
                    mc = new ManageComputers(); mc.buildGUI();
                    mlan = new ManageLan();   mlan.buildGUI();
                    mwan = new ManageWan();   mwan.buildGUI();


                    //add all the panels to the home panel which has a cardlayout
                    home.add(welcome, "Welcome");
                    home.add(adb, "AddBlock");
```

```
home.add(adl,"AddLab");
home.add(adc,"AddComputer");
home.add(adlan,"AddLan");
home.add(adwan,"AddWan");

home.add(vb,"ViewBlock");
home.add(vl,"ViewLab");
home.add(vc,"ViewComputer");
home.add(vlan,"ViewLan");
home.add(vwan,"ViewWan");

home.add(ml,"ManageLabs");
home.add(mc,"ManageComputers");
home.add(mlan,"ManageLan");
home.add(mwan,"ManageWan");

// add home panel to main frame
add(home);

// create menu bar and add it to frame
MenuBar mbar = new MenuBar();
setMenuBar(mbar);

// create the menu items and add it to Menu

Menu block = new Menu("Block");
MenuItem item1, item2;
block.add(item1 = new MenuItem("Add Block"));
block.add(item2 = new MenuItem("View Blocks"));
mbar.add(block);

Menu lab = new Menu("Labs");
MenuItem item3, item4;
lab.add(item3 = new MenuItem("Add Lab"));
lab.add(item4 = new MenuItem("View Labs"));
mbar.add(lab);

Menu computer = new Menu("Computers");
MenuItem item5, item6;
computer.add(item5 = new MenuItem("Add Computer"));
computer.add(item6 = new MenuItem("View Computers"));
mbar.add(computer);

Menu lan = new Menu("LAN");
MenuItem item7, item8;
lan.add(item7 = new MenuItem("Add LAN Device"));
lan.add(item8 = new MenuItem("View LAN Devices"));
mbar.add(lan);

Menu wan = new Menu("WAN");
MenuItem item9, item10;
wan.add(item9 = new MenuItem("Add WAN Device"));
wan.add(item10 = new MenuItem("View WAN Devices"));
mbar.add(wan);

Menu manage = new Menu("Manage");
MenuItem item11,item12,item13,item14,item15;
manage.add(item11 = new MenuItem("Labs"));
manage.add(item12 = new MenuItem("Computers"));
```

```
                manage.add(item13 = new MenuItem("LAN"));
                manage.add(item14 = new MenuItem("WAN"));
                manage.add(item15 = new MenuItem("Internet"));
                mbar.add(manage);

                Menu help = new Menu("Help");
                MenuItem About , Credits ;
                help.add(About = new MenuItem("About"));
                help.add(Credits = new MenuItem("Credits"));
                mbar.add(help);

                // register listeners
                item1.addActionListener(this);
                item2.addActionListener(this);
                item3.addActionListener(this);
                item4.addActionListener(this);
                item5.addActionListener(this);
                item6.addActionListener(this);
                item7.addActionListener(this);
                item8.addActionListener(this);
                item9.addActionListener(this);
                item10.addActionListener(this);
                item11.addActionListener(this);
                item12.addActionListener(this);
                item13.addActionListener(this);
                item14.addActionListener(this);
                item15.addActionListener(this);
                About.addActionListener(this);
                Credits.addActionListener(this);


                 // Anonymous inner class which extends WindowAdaptor to
handle the Window event: windowClosing
                addWindowListener(new WindowAdapter(){
                      public void windowClosing(WindowEvent we)
                      {
                                  System.exit(0);
                      }
                });

                //Frame properties
                setTitle("VCE Network Hardware Base");
                Color clr = new Color(230,230,230);
                setBackground(clr);
                setFont(new Font("SansSerif", Font.BOLD, 14));
                //setForeground(Color.WHITE);
                setSize(500, 600);
                setVisible(true);

        }

        public void actionPerformed(ActionEvent ae)
        {
              String arg = ae.getActionCommand();


              if(arg.equals("Add Block"))
              {
                   cardLO.show(home, "AddBlock");
```

```
        }

            else if(arg.equals("View Blocks"))
            {
                  cardLO.show(home, "ViewBlock");
                  vb.loadBlocks();
            }

            else if(arg.equals("Add Lab"))
            {
                  cardLO.show(home, "AddLab");
            }

            else if(arg.equals("View Labs"))
            {
                  cardLO.show(home, "ViewLab");
                  vl.loadLabs();
            }

            else if(arg.equals("Add Computer"))
            {
                  cardLO.show(home, "AddComputer");
            }

            else if(arg.equals("View Computers"))
            {
                  cardLO.show(home, "ViewComputer");
                  vc.loadComputers();
            }

            else if(arg.equals("Add LAN Device"))
            {
                  cardLO.show(home, "AddLan");
            }

            else if(arg.equals("View LAN Devices"))
            {
                  cardLO.show(home, "ViewLan");
                  vlan.loadLAN();
            }

            else if(arg.equals("Add WAN Device"))
            {
                  cardLO.show(home, "AddWan");
            }

            else if(arg.equals("View WAN Devices"))
            {
                  cardLO.show(home, "ViewWan");
                  vwan.loadWAN();
            }
            else if(arg.equals("Labs"))
            {
                  cardLO.show(home, "ManageLabs");
        }
            else if(arg.equals("Computers"))
            {
                  cardLO.show(home, "ManageComputers");
        }
```

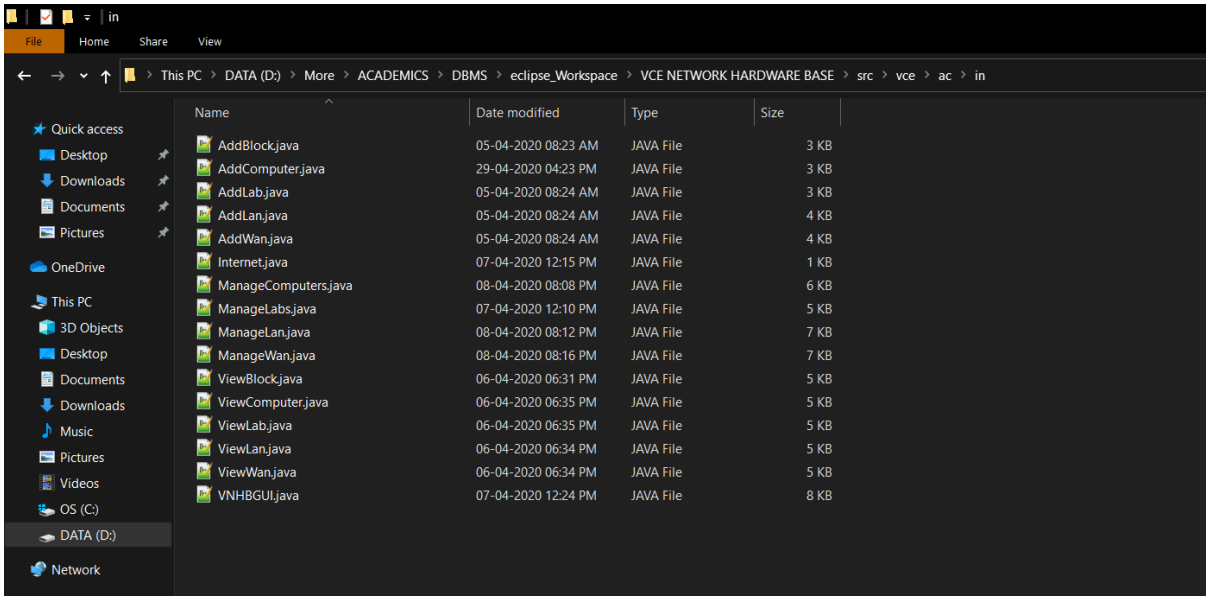
```
                else if(arg.equals("LAN"))
                {
                      cardLO.show(home, "ManageLan");
          }
                else if(arg.equals("WAN"))
                {
                      cardLO.show(home, "ManageWan");
          }
                else if(arg.equals("Internet"))
                {
                      cardLO.show(home, "Internet");
          }


                else if(arg.contentEquals("About"))
                {
                       JOptionPane.showMessageDialog(this, "                VCE
NETWORK HARDWARE BASE\n              \n              \n  ©VCE Network Hardware
Base . All Rights Reserved", "About",JOptionPane.PLAIN_MESSAGE);
                       setBackground(Color.DARK_GRAY);
                }
                else if(arg.equals("Credits"))
                {
                       JOptionPane.showMessageDialog(this, "               Name :
Anoop\n               Roll no : 1602-18-737-065\n             Section : IT-B\n
©VCE Network Hardware Base . All Rights Reserved",
"Credits",JOptionPane.PLAIN_MESSAGE);

                }
          }
          public static void main(String ... args)
          {
                      new VNHBGUI();
          }

}
```

GITHUB LINK:

1. https://github.com/ANOOP-PANDIRI/DBMS-PROJECT

FOLDER STRUCTURE :

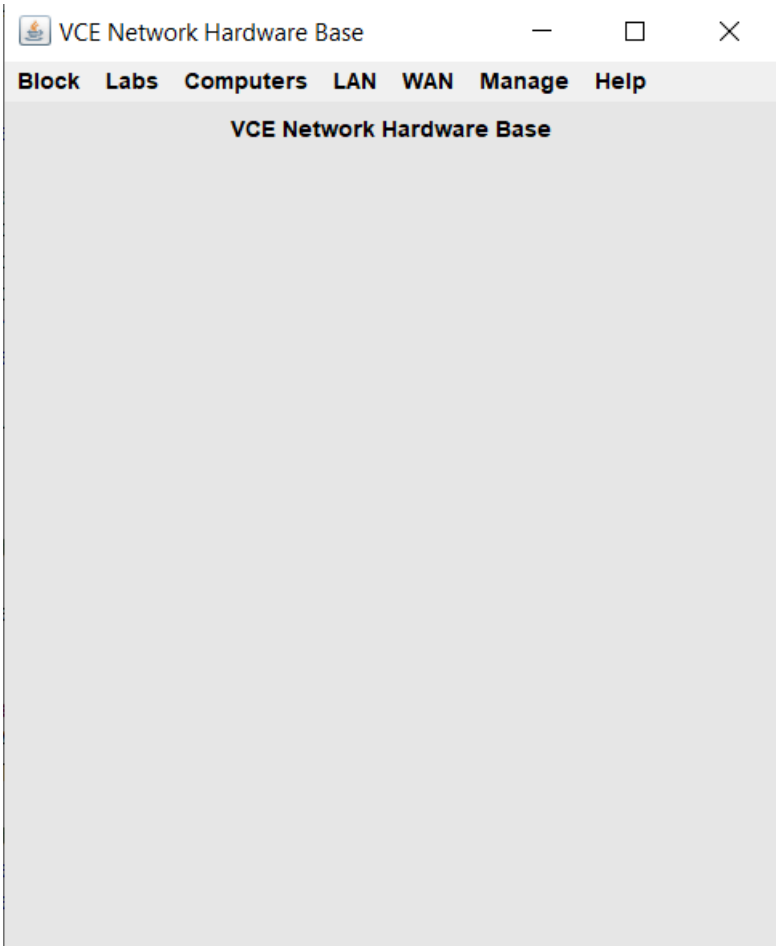
The Eclipse Project has a Folder named src . It has a package vce.ac.in which has java files. Add*** .java will perform Insert Operation and View***.java will perform Update as well as delete

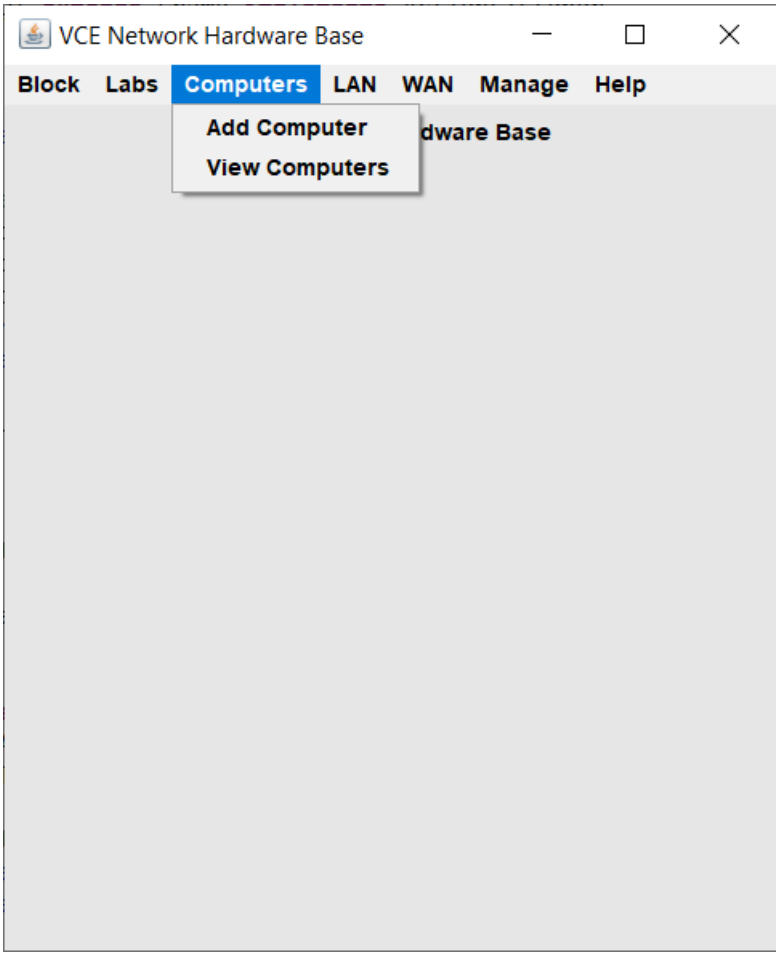


TESTING

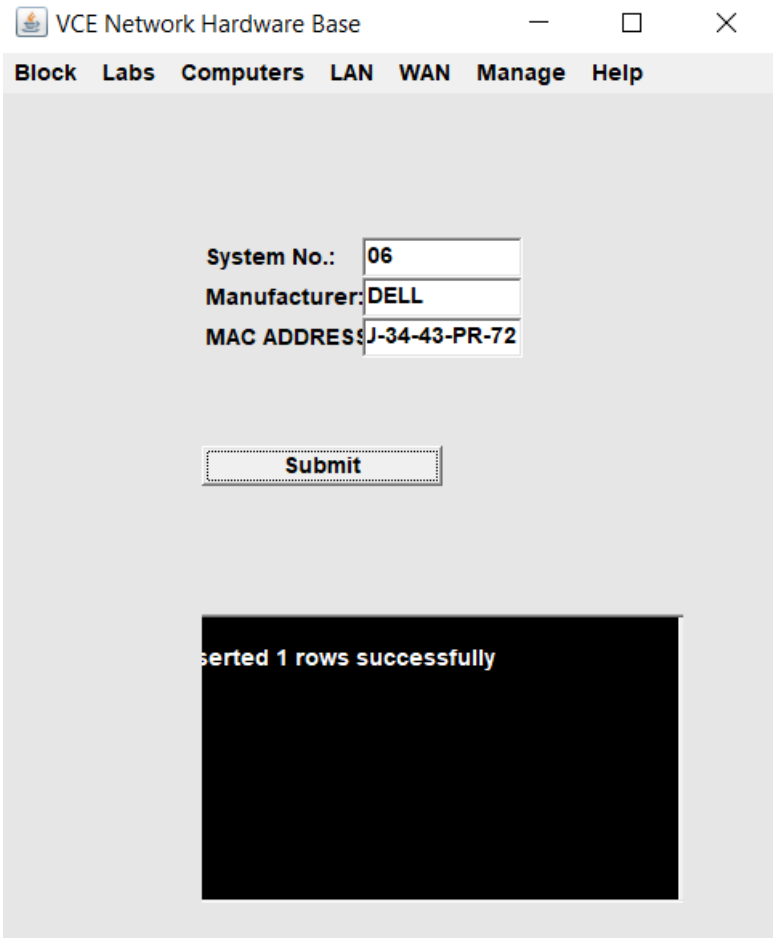
The program runs for execution of three basic operations of insertion, update and delete on 5 different table. Along with this, it also has a output column which gives the information about how many rows have been edited. Errors, syntactical or exceptional will be shown if occurred.

HOME PAGE:



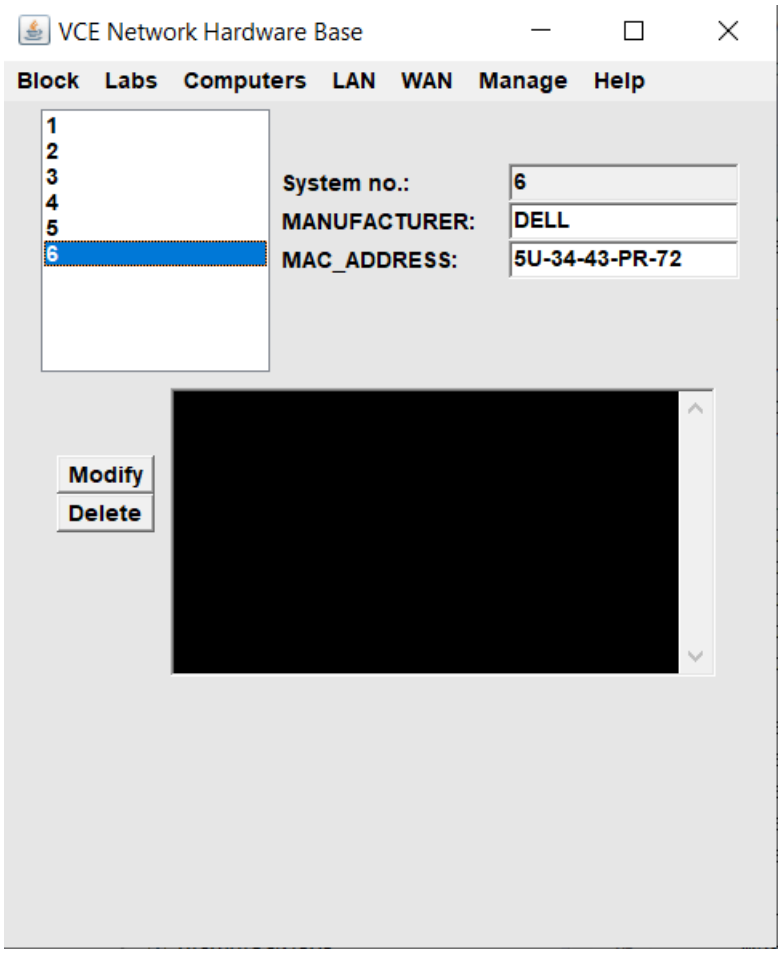

VCE Network Hardware Base
Block
Labs
Computers
LAN
WAN
Manage
Help
Add Computer
View Computers
dware Base

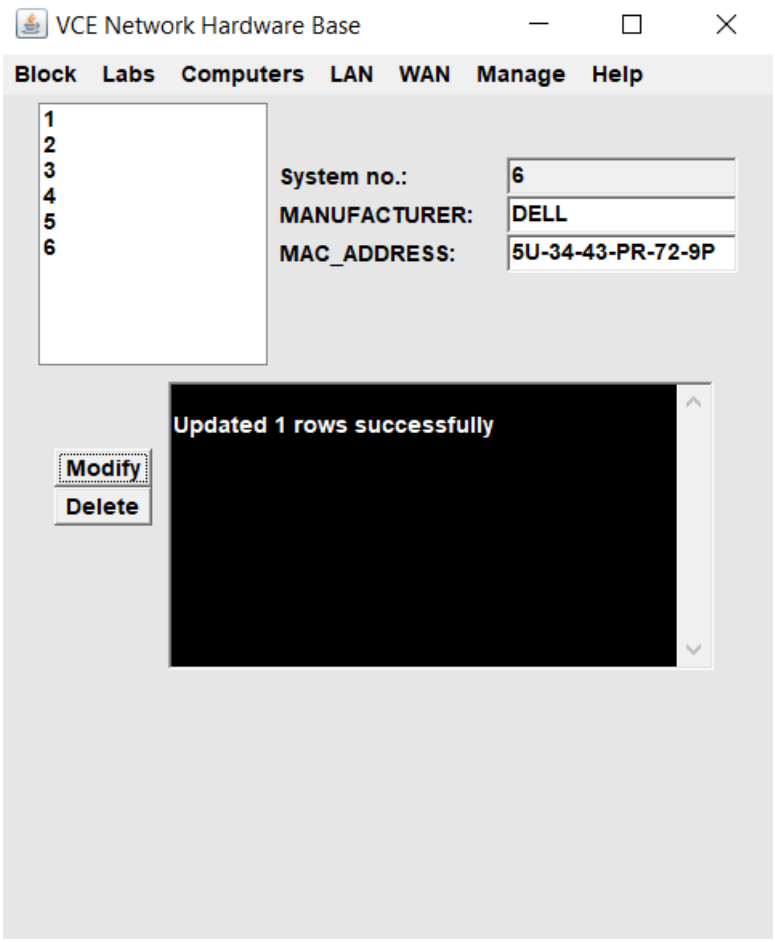
INSERT COMPUTER :




```
SQL> select * from computers;

       CID MANUFACTURER         MAC_ADDRESS
---------- -------------------- --------------------
         6 DELL                 5U-34-43-PR-72
         1 DELL                 00-14-22-01-23-49
         2 DELL                 14-CC-20-12-08-E2
         3 DELL                 21-09-H1-25-01-F2
         4 HP                   34-15-Z2-13-25-Y2
         5 HP                   58-24-R3-PR-01-24

6 rows selected.
```

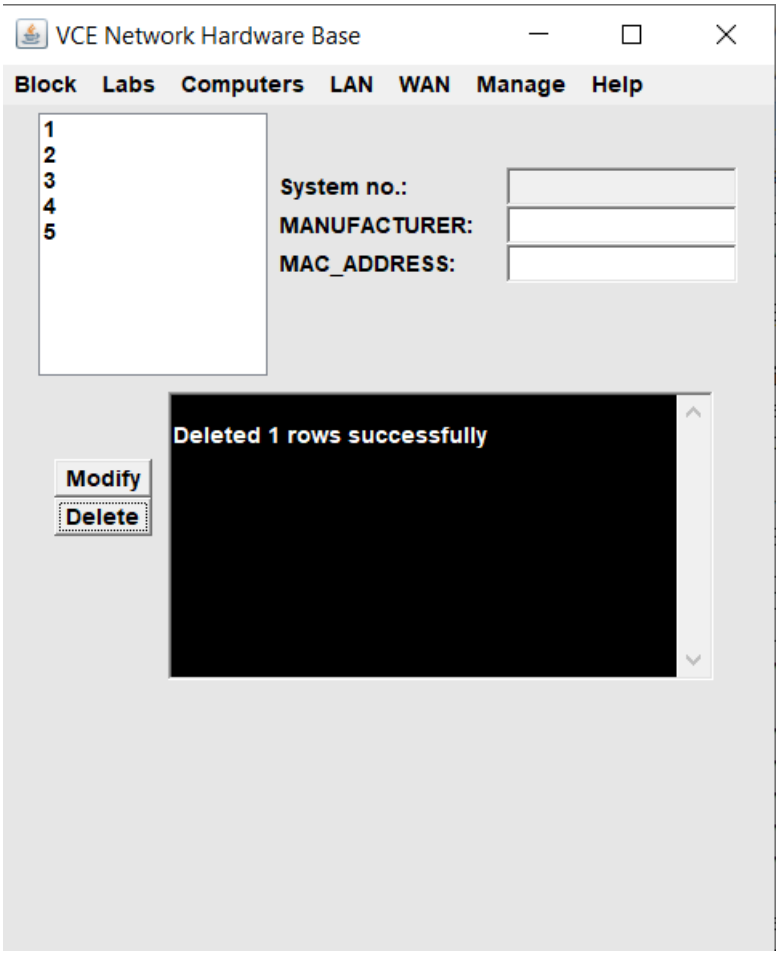
UPDATE COMPUTER :






```
SQL> select * from computers;

       CID MANUFACTURER          MAC_ADDRESS
---------- --------------------- --------------------
         6 DELL                  5U-34-43-PR-72-9P
         1 DELL                  00-14-22-01-23-49
         2 DELL                  14-CC-20-12-08-E2
         3 DELL                  21-09-H1-25-01-F2
         4 HP                    34-15-Z2-13-25-Y2
         5 HP                    58-24-R3-PR-01-24

6 rows selected.
```

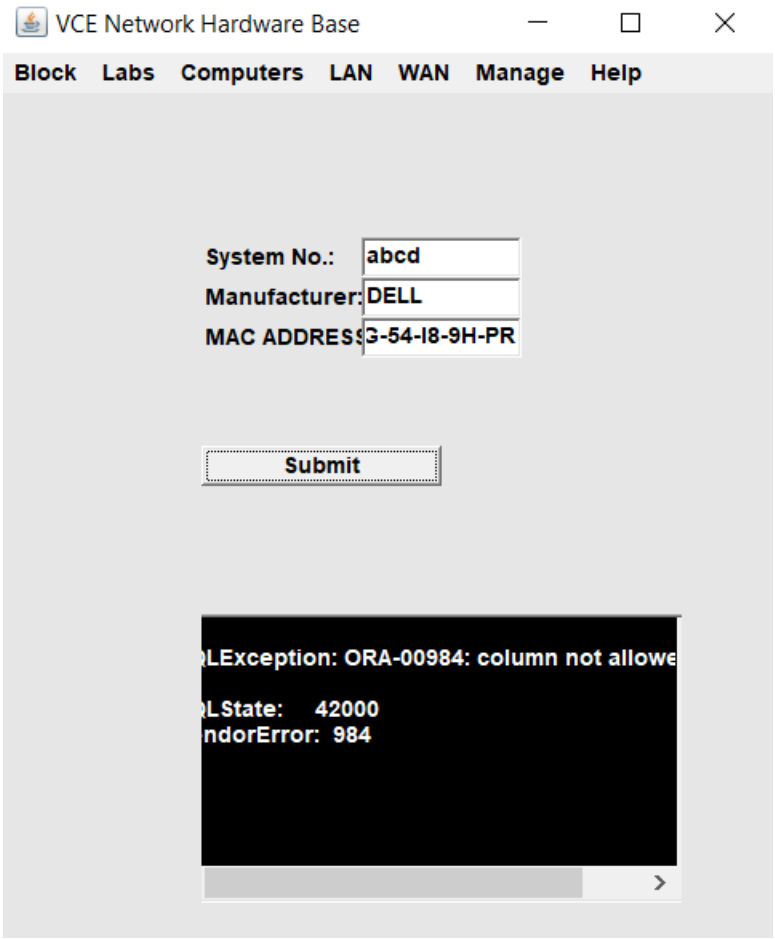
DELETE COMPUTER:



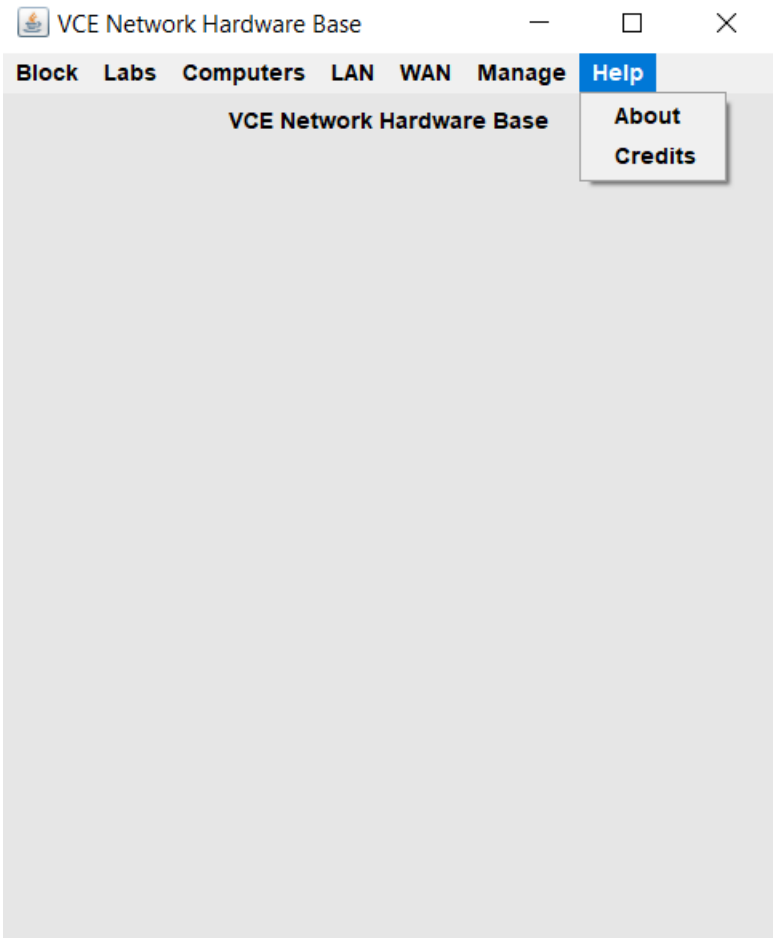

```
SQL> select * from computers;

       CID MANUFACTURER         MAC_ADDRESS
---------- -------------------- --------------------
         1 DELL                 00-14-22-01-23-49
         2 DELL                 14-CC-20-12-08-E2
         3 DELL                 21-09-H1-25-01-F2
         4 HP                   34-15-Z2-13-25-Y2
         5 HP                   58-24-R3-PR-01-24
```

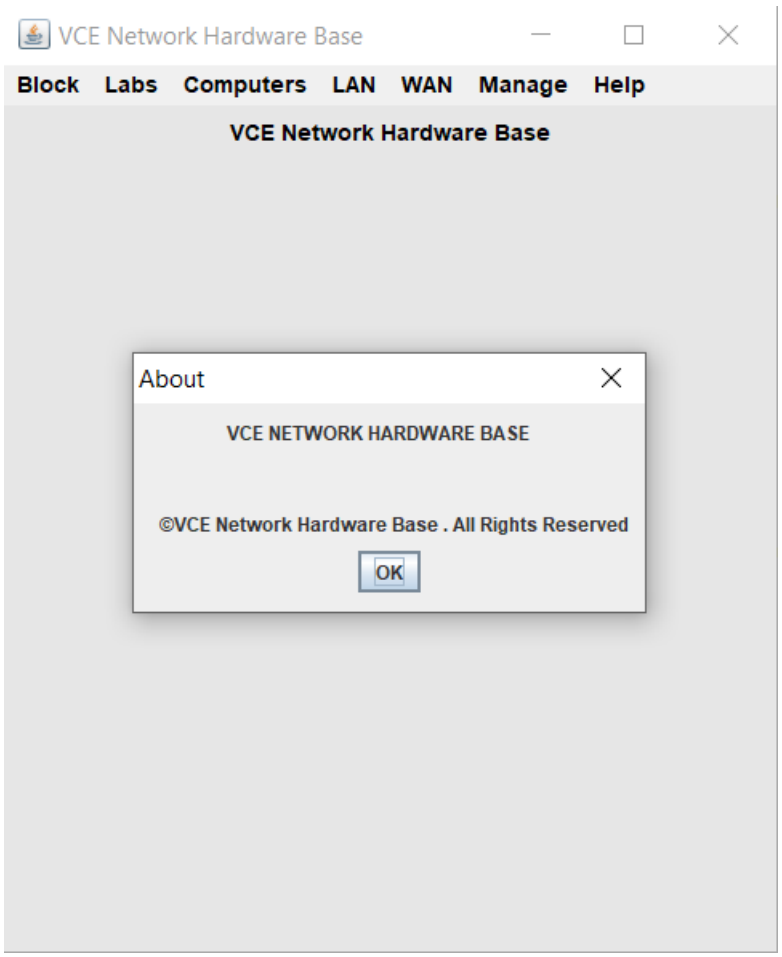
ERROR ENTRIES:

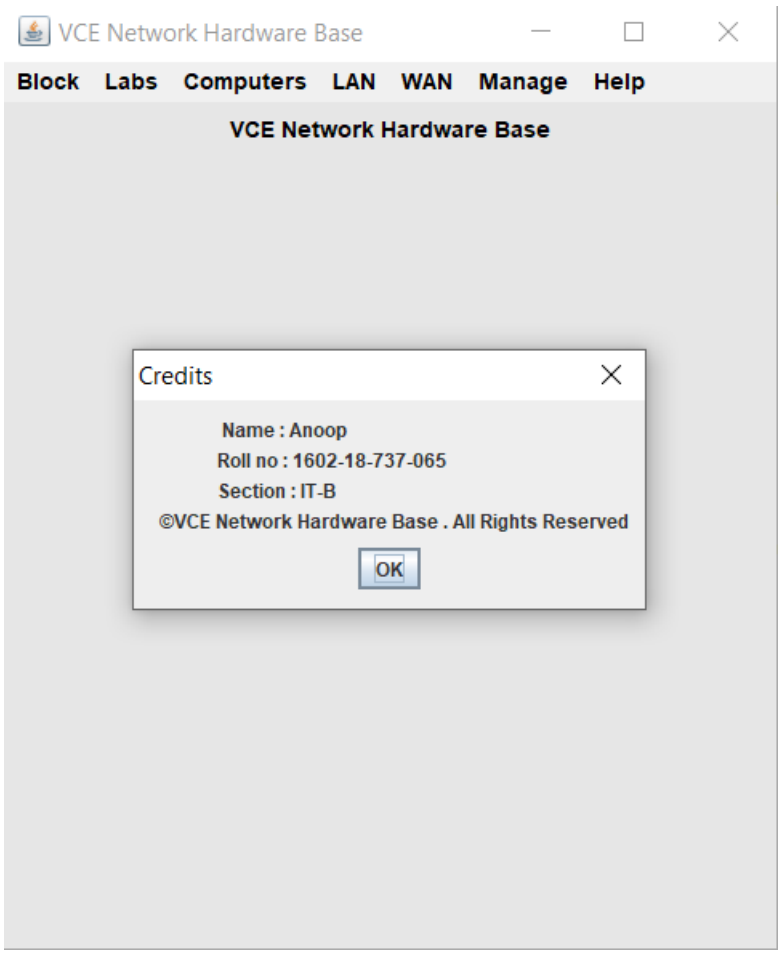
If data entered in text boxes is not according to the domain type it will lead to an error. For example entering Some Alphabets if its Domain type is Number(10).



ABOUT AND CREDITS




VCE Network Hardware Base
Block
Labs
Computers
LAN
WAN
Manage
Help
VCE Network Hardware Base
About
VCE NETWORK HARDWARE BASE
©VCE Network Hardware Base . All Rights Reserved
OK



RESULTS

The DML commands, Insert, update and delete for one of the tables in given below:

For Computers table: (in java, as per the application)

Insert: `"INSERT INTO COMPUTERS VALUES(" + cidText.getText()+", " + "'" + mnfText.getText() + "'," +"'"+ macText.getText() +"')";`

Update: `"UPDATE COMPUTERS SET MANUFACTURER ='"+mnfText.getText()+"',"+ " MAC_ADDRESS='"+macText.getText()+"'"+" WHERE cid ="+compList.getSelectedItem());`

Delete: `"DELETE FROM COMPUTERS WHERE cid = "+ compList.getSelectedItem());`

CONCLUSION:

1.Connection with database is established.

2.The values entered in Application are updated in Local Database.

REFERENCES

https://www.tutorialspoint.com/Basic-Network-Hardware

https://en.wikipedia.org/wiki/Network_switch

https://en.wikipedia.org/wiki/Router_%28computing%29

https://en.wikipedia.org/wiki/Gateway_%28telecommunications%29