MAHARANA PRATAP EDUCATION CENTRE

Session: 2017-2018



A PROJECT REPORT ON

“ELECTRICITY BILLING SYSTEM”

Submitted By:

Utkarsh Katiyar

XII

Roll No.

# CERTIFICATE

It is hereby to certify that, the original and genuine investigation work has been carried out to investigate about the subject matter and the related data collection and investigation has been completed solely, sincerely and satisfactorily by **Anirudh Dabral** of **Class XII**, **Maharana Pratap Education Centre**, regarding the project titled **“**ELECTRICITY BILLING SYSTEM**”**

Teacher’s Signature

# ACKNOWLEDGEMENT

It’s my humble pleasure to acknowledge my deep sense of gratitude to my I.P. teacher for his valuable support, constant help and guidance at each and every stage without which it wouldn’t have been possible to complete this project.

I also register my sense of gratitude to our principal Mrs. Jyoti Das Ma’am for her immense encouragement that has made this project successful.

Anirudh Dabral

XII

.

# INTRODUCTION

ELECTRICITY BILLING SYSTEM This project has been made for storing payment and customer records like meter number, house number, name, phone number, charges for units etc. It contains addition, modification, deletion and searching of bills, generating of bills as per requirement. Total amount is being calculated automatically.

# LIMITATIONS

* This program is for test purpose only.
* No customised reports can be genrated.
* Charges per unit cannot be updated.
* Once updated, the previous details of customer is not stored.
* Payment details of a terminated meter number are still present in PAYMENT table

# THEORY

**Connecting to a MySQL Database**

This document demonstrates how to set up a connection to a MySQL database from the NetBeans IDE. Once connected, you can begin working with MySQL in the IDE's Database Explorer by creating new databases and tables, populating tables with data, and running SQL queries on database structures and content. This tutorial is designed for beginners with a basic understanding of database management, who want to apply their knowledge to working with MySQL in NetBeans IDE.

[MySQL](http://www.mysql.com/) is a popular Open Source relational database management system (RDBMS) commonly used in web applications due to its speed, flexibility and reliability. MySQL employs SQL, or Structured Query Language, for accessing and processing data contained in databases.

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 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.

A JDBC-to-ODBC bridge enables connections to any ODBC-accessible data source in the Java virtual machine (JVM) host environment.

JDBC allows multiple implementations to exist and be used by the same application. The API provides a mechanism for dynamically loading the correct Java packages and registering them with the JDBC Driver Manager. The Driver Manager is used as a connection factory for creating JDBC connections.

JDBC connections support creating and executing statements. These may be update statements such as SQL's CREATE, INSERT, UPDATE and DELETE, or they may be query statements such as SELECT. Additionally, stored procedures may be invoked through a JDBC connection. JDBC represents statements using one of the following classes:

* [Statement](https://docs.oracle.com/javase/8/docs/api/java/sql/Statement.html) – the statement is sent to the database server each and every time.
* [PreparedStatement](https://docs.oracle.com/javase/8/docs/api/java/sql/PreparedStatement.html) – the statement is cached and then the [execution path](https://en.wikipedia.org/wiki/Query_plan) is pre-determined on the database server allowing it to be executed multiple times in an efficient manner.
* [CallableStatement](https://docs.oracle.com/javase/8/docs/api/java/sql/CallableStatement.html) – used for executing [stored procedures](https://en.wikipedia.org/wiki/Stored_procedures) on the database.

Update statements such as INSERT, UPDATE and DELETE return an update count that indicates how many [rows](https://en.wikipedia.org/wiki/Row_%28database%29) were affected in the database. These statements do not return any other information.

Query statements return a JDBC row result set. The row result set is used to walk over the [result set](https://en.wikipedia.org/wiki/Result_set). Individual [columns](https://en.wikipedia.org/wiki/Column_%28database%29) in a row are retrieved either by name or by column number. There may be any number of rows in the result set. The row result set has metadata that describes the names of the columns and their types.

There is an extension to the basic JDBC API in the [javax.sql](https://docs.oracle.com/javase/8/docs/api/javax/sql/package-summary.html).

JDBC connections are often managed via a [connection pool](https://en.wikipedia.org/wiki/Connection_pool) rather than obtained directly from the driver.

# TABLE STRUCTURE OF REPORT

**DESC LOGIN;**

+----------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+----------+-------------+------+-----+---------+-------+

| USERID | int(1) | NO | PRI | NULL | |

| USERNAME | varchar(20) | YES | | NULL | |

| PASSWORD | varchar(20) | NO | | NULL | |

+----------+-------------+------+-----+---------+-------+

**DESC CUSTDETAIL;**

+-------------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+-------------+-------------+------+-----+---------+-------+

| NAME | varchar(25) | YES | | NULL | |

| HOUSE\_NO | varchar(20) | YES | | NULL | |

| PHONE | int(12) | YES | | NULL | |

| LOAD\_inWatt | int(7) | YES | | NULL | |

| METER\_NO | int(5) | NO | PRI | NULL | |

| LAST\_READ | int(4) | YES | | NULL | |

+-------------+-------------+------+-----+---------+-------+

**DESC PAYMENT;**

+----------+------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+----------+------------+------+-----+---------+-------+

| METER\_NO | int(5) | YES | | NULL | |

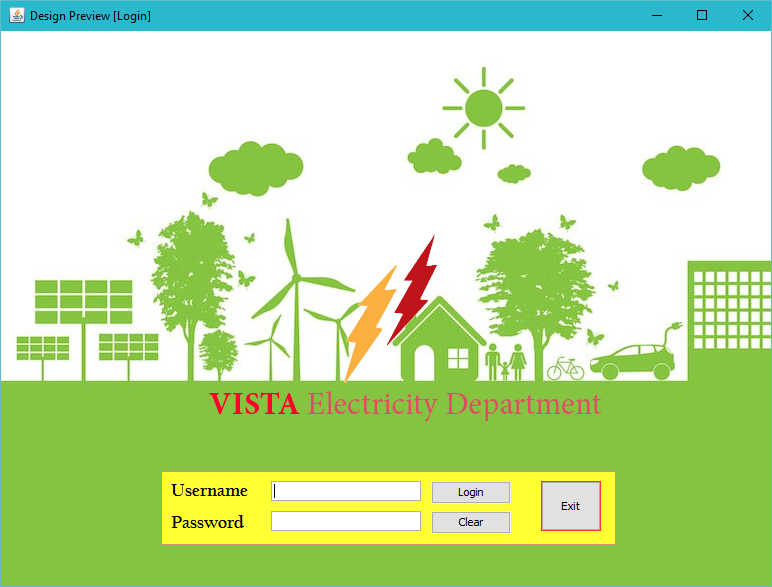
| READING | int(4) | YES | | NULL | |

| PAY\_DATE | date | YES | | NULL | |

| AMT | float(7,2) | YES | | NULL | |

+----------+------------+------+-----+---------+-------+

# CODING OF REPORT

**LOGIN**

import java.sql.\*;

import javax.swing.ImageIcon;

import javax.swing.JOptionPane;

public class Login extends javax.swing.JFrame {

Connection con;

Statement stmt,smt1;

ResultSet rs;

java.util.Date dt1;

public Login() {

initComponents();

try

{

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

}

catch(SQLException e)

{

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

}

}

**private void loginbtnActionPerformed(java.awt.event.ActionEvent evt) {**

try{

String sql="Select \* from login;";

smt1=con.createStatement();

rs=smt1.executeQuery(sql);

String log=usrnmtxt.getText();

String pass =new String(passtxt.getPassword());

while(rs.next())

{

if(log.equals(rs.getString("USERNAME")) && pass.equals(rs.getString("PASSWORD")))

{

Navigation nf=new Navigation(this);

nf.setVisible(true);

break;

}

}

}

catch(SQLException ex){

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

}

**}**

**private void ClearbtnActionPerformed(java.awt.event.ActionEvent evt) {**

usrnmtxt.setText("");

passtxt.setText("");

**}**

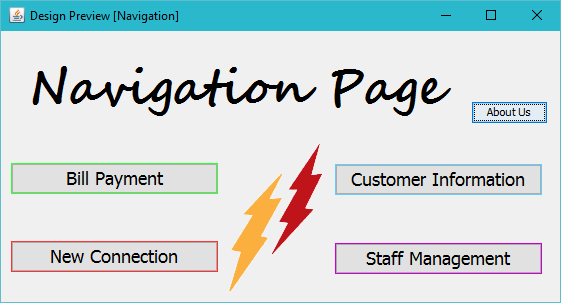
**private void exitbtnActionPerformed(java.awt.event.ActionEvent evt) {**

System.exit(0);

**}**

**}**

**NAVIGATION**



public class Navigation extends javax.swing.JFrame {

public Navigation(Login o) {

o.setVisible(false);

initComponents(); }

**private void BillBtnActionPerformed(java.awt.event.ActionEvent evt) {**

Payment pb=new Payment();

pb.setVisible(true);

**}**

**private void CInfoBtnActionPerformed(java.awt.event.ActionEvent evt) {**

Customer ci=new Customer();

ci.setVisible(true);

**}**

**private void ncbtnActionPerformed(java.awt.event.ActionEvent evt) {**

NewConnection nc=new NewConnection();

nc.setVisible(true);

**}**

**private void AbtBtnActionPerformed(java.awt.event.ActionEvent evt) {**

About ab=new About();

ab.setVisible(true);

**}**

**private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {**

String pass=JOptionPane.showInputDialog(this,"Enter Master Password");

if(pass==null) pass="";

if(pass.equals("Vista49") || pass.equals(""))

{Staff sf=new Staff();

sf.setVisible(true);

}

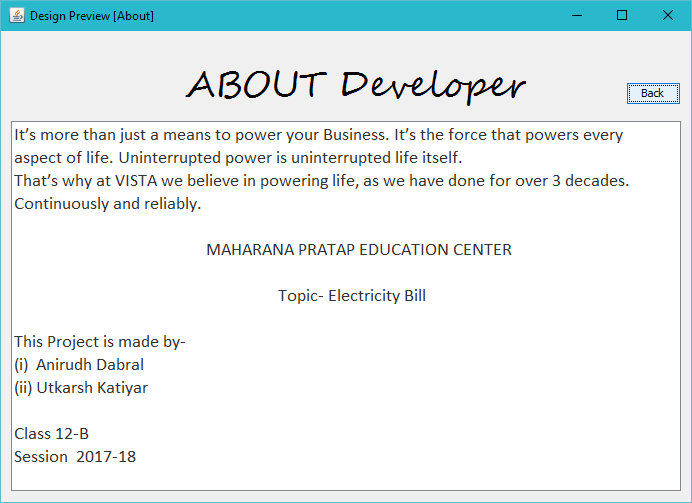
else if (pass.equals(""));

else JOptionPane.showMessageDialog(this,"Master Password Wrong");

**}**

}

**ABOUT**



public class About extends javax.swing.JFrame {

public About() {

initComponents();

}

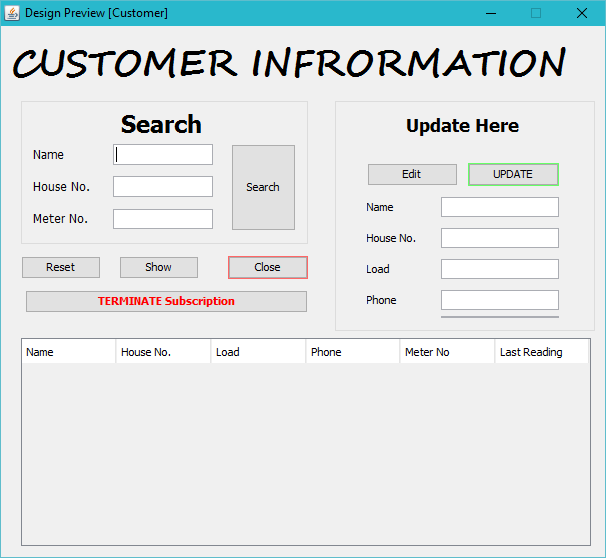
**private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {**

this.dispose();

**}**

}

**CUSTOMER**



import java.sql.\*;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

import javax.swing.table.TableModel;

public final class Customer extends javax.swing.JFrame {

Connection con;

Statement stmt,stmt1,stmt2;

ResultSet rs,rs1,rs2;

int load,phn,wno,lr;

String name,hno,query;

String N,H,L,P,M;

public Customer() {

initComponents();

reset();

try

{

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

stmt=con.createStatement();

String query="SELECT \* FROM CUSTDETAIL;";

rs=stmt.executeQuery(query);

rs.next();

}

catch(SQLException ec)

{

JOptionPane.showMessageDialog(this, ec);

}

}

public final void displayRecord(){

DefaultTableModel cust = (DefaultTableModel)

CTable.getModel();

try{con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

stmt=con.createStatement();

String query="SELECT \* FROM CUSTDETAIL;";

rs=stmt.executeQuery(query);

while(rs.next())

{

name=rs.getString("NAME");

hno=rs.getString("HOUSE\_NO");

phn=rs.getInt("PHONE");

wno=rs.getInt("METER\_NO");

load=rs.getInt("LOAD\_inWatt");

lr=rs.getInt("LAST\_READ");

cust.addRow (new Object[] {name, hno,load,phn,wno,lr});

}

}

catch(SQLException ec)

{

JOptionPane.showMessageDialog(this,ec);

}

}

public final void reset(){

updbtn.setEnabled(false);

termnbtn.setEnabled(true);

ntxt.setEditable(false);

htxt.setEditable(false);

ntxt.setEditable(false);

ltxt.setEditable(false);

ptxt.setEditable(false);

mtxt.setVisible(false);

ntxt.setText("");

htxt.setText("");

ntxt.setText("");

ltxt.setText("");

ptxt.setText("");

mtxt.setText("");

clear();

}

public final void clear(){

DefaultTableModel model = (DefaultTableModel)

CTable.getModel(); int rows=model.getRowCount();

if (rows>0)

{

for (int i=0; i<rows; i++)

model.removeRow(0);

}

}

**private void rstbtnActionPerformed(java.awt.event.ActionEvent evt) {**

NTxt1.setText("");

HnoTxt1.setText("");

MnTxt1.setText("");

reset();

**}**

**private void CTableMouseClicked(java.awt.event.MouseEvent evt) {**

int i=CTable.getSelectedRow();

TableModel vl = CTable.getModel();

ntxt.setText(vl.getValueAt(i,0).toString());

htxt.setText(vl.getValueAt(i,1).toString());

ltxt.setText(vl.getValueAt(i,2).toString());

ptxt.setText(vl.getValueAt(i,3).toString());

mtxt.setText(vl.getValueAt(i,4).toString());

**}**

**private void shbtnActionPerformed(java.awt.event.ActionEvent evt) {**

clear();

displayRecord();

**}**

**private void updbtnActionPerformed(java.awt.event.ActionEvent evt) {**

N=ntxt.getText();

H=htxt.getText();

N=ntxt.getText();

L=ltxt.getText();

P=ptxt.getText();

M=mtxt.getText();

String query2;

try

{

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

stmt1=con.createStatement();

int code=JOptionPane.showConfirmDialog(this,"Are you sure you want to update this information?","Confirm?",JOptionPane.YES\_NO\_OPTION);

if(code==JOptionPane.YES\_OPTION)

{

query2="UPDATE CUSTDETAIL SET NAME='" + N + "', HOUSE\_NO = '" + H + "',PHONE=" + P + ",LOAD\_inWatt="+ L + " WHERE METER\_NO=" + M +";";

int ss=stmt1.executeUpdate(query2);

if(ss>0) JOptionPane.showMessageDialog(this,"Updation Successful!!");

else JOptionPane.showMessageDialog(this,"Incorect values entered");

}

}

catch(SQLException e){

JOptionPane.showMessageDialog(this,e);

}

try

{

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

stmt=con.createStatement();

String query="SELECT \* FROM CUSTDETAIL;";

rs=stmt.executeQuery(query);

rs.next();

displayRecord();

}

catch(SQLException ec)

{

JOptionPane.showMessageDialog(this, ec);

}

**}**

**private void editbtnActionPerformed(java.awt.event.ActionEvent evt) {**

updbtn.setEnabled(true);

termnbtn.setEnabled(false);

ntxt.setEditable(true);

htxt.setEditable(true);

ntxt.setEditable(true);

ltxt.setEditable(true);

ptxt.setEditable(true);

}

**private void termnbtnActionPerformed(java.awt.event.ActionEvent evt) {**

reset();

M=mtxt.getText();

String query;

Statement st;

ResultSet r;

try

{

int code=JOptionPane.showConfirmDialog(this,"Are you sure you want to TERMINATE this Subscription?","Confirm?",JOptionPane.YES\_NO\_OPTION);

if(code==JOptionPane.YES\_OPTION)

{

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

query="DELETE FROM CUSTDETAIL WHERE METER\_NO='" + M +"';";

st=con.createStatement();

int c1=st.executeUpdate(query);

if(c1>0) JOptionPane.showMessageDialog(this,"Updation Successful!!");

}

}

catch (SQLException ex) {

JOptionPane.showMessageDialog(this, ex);

}

try

{

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

stmt=con.createStatement();

String quer="SELECT \* FROM CUSTDETAIL;";

rs=stmt.executeQuery(quer);

displayRecord();

}

catch(SQLException ec)

{

JOptionPane.showMessageDialog(this, ec);

}

**}**

**private void clsbtnActionPerformed(java.awt.event.ActionEvent evt) {**

this.dispose();

**}**

**private void scrbtnActionPerformed(java.awt.event.ActionEvent evt) {**

clear();

String nm=NTxt1.getText(), hou=HnoTxt1.getText(),mn=MnTxt1.getText();

Statement smt;

String query1;

query1="SELECT \* FROM CUSTDETAIL WHERE NAME LIKE '%" + nm + "%' AND HOUSE\_NO LIKE '%" + hou + "%' AND METER\_NO LIKE '%" + mn +"%';";

try

{ con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

smt=con.createStatement();

rs1=smt.executeQuery(query1);

DefaultTableModel cust = (DefaultTableModel)

CTable.getModel();

while(rs1.next())

{ name=rs1.getString("NAME");

hno=rs1.getString("HOUSE\_NO");

phn=rs1.getInt("PHONE");

wno=rs1.getInt("METER\_NO");

load=rs1.getInt("LOAD\_inWatt");

lr=rs1.getInt("LAST\_READ");

cust.addRow (new Object[] {name, hno,load,phn,wno,lr});

}

if(CTable.getRowCount()==0)JOptionPane.showMessageDialog(this,"No matching records");

}

catch(SQLException ex){

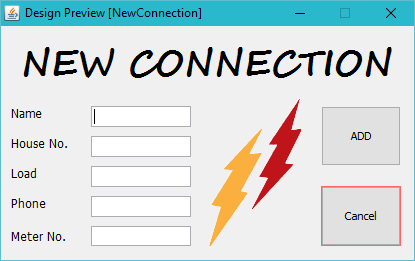
JOptionPane.showMessageDialog(this,ex);

}

**}**

}

**NEW CONNECTION**



import java.sql.\*;

import javax.swing.JOptionPane;

public final class NewConnection extends javax.swing.JFrame {

Connection con;

Statement stmt,stmt1;

ResultSet rs,rs1;

int load,phn,wno,lr;

String name,hno;

public NewConnection() {

initComponents();

try

{

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

stmt=con.createStatement();

String query="SELECT \* FROM CUSTDETAIL;";

rs=stmt.executeQuery(query);

rs.next();

}

catch(SQLException ec)

{

JOptionPane.showMessageDialog(this, ec);

}

}

public final void clear(){

NTxt.setText("");

LTxt.setText("");

HnoTxt.setText("");

PTxt.setText("");

MnTxt.setText("");

}

**private void clsbtnActionPerformed(java.awt.event.ActionEvent evt) {**

this.dispose();

**}**

**private void AddbtnActionPerformed(java.awt.event.ActionEvent evt) {**

int ramt=JOptionPane.showConfirmDialog(this,"Registration amount = Rs.1000","Confirm?",JOptionPane.YES\_NO\_OPTION);

if(ramt==JOptionPane.YES\_OPTION)

{

String nm=NTxt.getText(), hou=HnoTxt.getText();

int ld=Integer.parseInt(LTxt.getText()), ph=Integer.parseInt(PTxt.getText()), mn=Integer.parseInt(MnTxt.getText());

String query1="SELECT \* FROM CUSTDETAIL WHERE METER\_NO=" + mn + ";";

String query2;

try {

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

stmt1=con.createStatement();

rs1=stmt1.executeQuery(query1);

if(!(rs1.next()))

{

int code=JOptionPane.showConfirmDialog(this,"Insert Record?","Confirm?",JOptionPane.YES\_NO\_OPTION);

if(code==JOptionPane.YES\_OPTION)

{

query2="INSERT INTO CUSTDETAIL VALUES('" + nm + "','" + hou + "'," + ph + "," + ld + "," + mn + ",NULL);";

int ss=stmt1.executeUpdate(query2);

if(ss>0) JOptionPane.showMessageDialog(this,"Updation Successful!!");

else JOptionPane.showMessageDialog(this,"Meter number already exists");

clear();

}

}

else {JOptionPane.showMessageDialog(this,"Meter number already exists");}

}

catch(SQLException e){

JOptionPane.showMessageDialog(this,e);

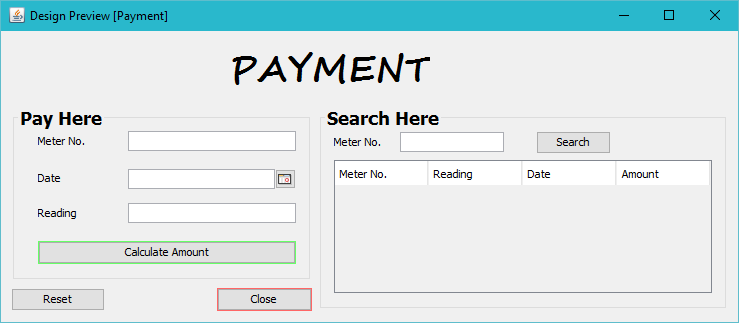
}

}

**}**

}

**PAYMENT**



import java.sql.\*;

import javax.swing.JOptionPane;

import java.sql.DriverManager;

import java.text.SimpleDateFormat;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

public class Payment extends javax.swing.JFrame {

Connection con;

Statement stmt,stmt1,stmt2,stmt3;

ResultSet rs,rs1,rs2,rs3;

int mno,read;

double am;

String date,query,query2;

public Payment() {

initComponents();

}

public final void displayrec(){

try{

while(rs.next())

{ date=rs.getString("PAY\_DATE");

read=rs.getInt("READING");

am=rs.getInt("AMT");

mno=rs.getInt("METER\_NO");

DefaultTableModel cust = (DefaultTableModel)

PayTable.getModel();

cust.addRow (new Object[] {mno, read,date,am});}

}

catch(SQLException ec) {JOptionPane.showMessageDialog(this,ec);}

}

public final void clear(){

DefaultTableModel model = (DefaultTableModel)

PayTable.getModel(); int rows=model.getRowCount();

if (rows>0)

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

model.removeRow(0);

}

}

**private void clsbtnActionPerformed(java.awt.event.ActionEvent evt) {**

this.dispose();

**}**

**private void srchbtnActionPerformed(java.awt.event.ActionEvent evt) {**

clear();

CalcAmt.setEnabled(false);

String mn=MnoTxt1.getText();

Statement smt;

String query1,query2;

query1="SELECT \* FROM PAYMENT WHERE METER\_NO LIKE '%" + mn + "%' ORDER BY PAY\_DATE DESC;";

try

{

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

smt=con.createStatement();

rs=smt.executeQuery(query1);

displayrec();

if(PayTable.getRowCount()==0)

JOptionPane.showMessageDialog(this,"No Matching Records Found!!!");

}

catch(SQLException ex){

JOptionPane.showMessageDialog(this,ex);

}

**}**

**private void resbtnActionPerformed(java.awt.event.ActionEvent evt) {**

MnoTxt1.setText("");

CalcAmt.setEnabled(true);

clear();

**}**

**private void CalcAmtActionPerformed(java.awt.event.ActionEvent evt) {**

int mno;

SimpleDateFormat dform=new SimpleDateFormat("yyyy-MM-dd");

String date = dform.format(dtchs.getDate());

double nrd,amt=0,lrd=0,rd;

mno=Integer.parseInt(MnoTxt.getText());

nrd=Double.parseDouble(RngTxt.getText());

String query="SELECT \* FROM CUSTDETAIL;";

String queryy="SELECT \* FROM PAYMENT WHERE METER\_NO=" + mno + ";";

String query1,query2;

try

{

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

stmt3=con.createStatement();

String query3="SELECT \* FROM CUSTDETAIL WHERE METER\_NO=" + mno +";";

rs3=stmt3.executeQuery(query3);

rs3.next();

String naam=rs3.getString("NAME");

lrd=rs3.getInt("LAST\_READ");

if(nrd>lrd && nrd>=0){

rd=nrd-lrd;

if (rd<=400)

{ amt=(rd\*6)+200;

JOptionPane.showMessageDialog(this,"Name of Customer = "+naam+"\nCurrent reading = "+nrd+"\nLast Reading = "+lrd+"\nUnits charged for = " + rd + "\n" + "Amount to be paid = " + amt + "\n" + "Charge per unit = Rs.6.00\nFixed Charge = Rs.200");

}

else

{ amt=(rd\*6.5)+250;

JOptionPane.showMessageDialog(this,"Name of Customer = "+naam+"\nCurrent reading = "+nrd+"\nLast Reading = "+lrd+"\nUnits charged for = " + rd + "\n" + "Amount to be paid = " + amt + "\n" + "Charge per unit = Rs.6.50\nFixed Charge = Rs.250");

}

stmt1=con.createStatement();

rs1=stmt1.executeQuery(queryy);

int code=JOptionPane.showConfirmDialog(this,"Is amount paid?","Confirm?",JOptionPane.YES\_NO\_OPTION);

if(code==JOptionPane.YES\_OPTION)

{ query1="UPDATE CUSTDETAIL SET LAST\_READ=" + nrd + "WHERE METER\_NO=" + mno +";";

query2="INSERT INTO PAYMENT VALUES("+ mno+","+rd+",'"+date+"',"+amt+");";

int ss=stmt1.executeUpdate(query1);

if(ss>0) {JOptionPane.showMessageDialog(this,"Updation Successful!!");

int st=stmt1.executeUpdate(query2);

}

else JOptionPane.showMessageDialog(this,"Meter Number not Registered");

}

}else JOptionPane.showMessageDialog(this,"Reading Incorrect");

}

catch(SQLException e){

JOptionPane.showMessageDialog(this,"Meter Number not Registered");

}

MnoTxt.setText("");

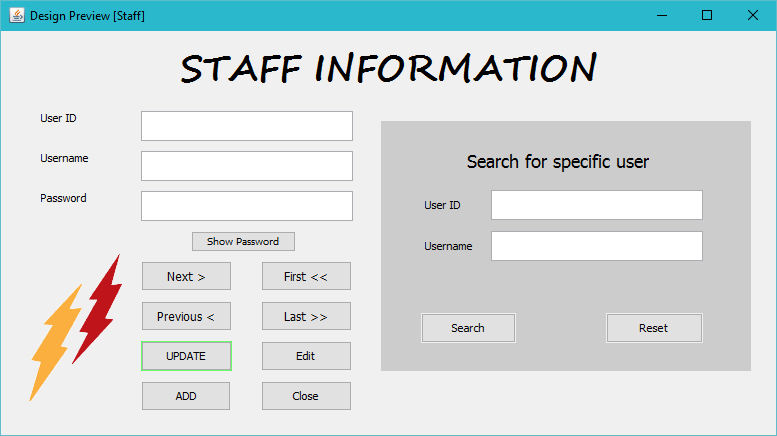
RngTxt.setText("");

dtchs.setDate(null);

**}**

}

**STAFF**



import java.sql.\*;

import javax.swing.JOptionPane;

public class Staff extends javax.swing.JFrame {

Connection con;

Statement stmt,stmt1;

ResultSet rs,rs1;

int id;

String pass,name,query;

public Staff() {

initComponents();

reset();

try

{

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

stmt=con.createStatement();

String query="SELECT \* FROM LOGIN;";

rs=stmt.executeQuery(query);

rs.next();

displayRecord();

}

catch(SQLException ec)

{

JOptionPane.showMessageDialog(this, ec);

}

}

public final void displayRecord(){

try{ id=rs.getInt("USERID");

name=rs.getString("USERNAME");

pass=rs.getString("PASSWORD");

idtxt.setText(""+id);

ntxt.setText(""+name);

ptxt.setText("\*\*\*\*\*");

}

catch(SQLException ec)

{

JOptionPane.showMessageDialog(this,ec);

}

}

public final void reset(){

ftbtn.setEnabled(true);

ltbtn.setEnabled(true);

ntbtn.setEnabled(true);

psbtn.setEnabled(true);

Addbtn.setEnabled(false);

updbtn.setEnabled(false);

editbtn.setEnabled(true);

cbtn.setEnabled(true);

idtxt.setEditable(false);

ntxt.setEditable(false);

ptxt.setEditable(false);

}

public final void clear(){

ntxt.setText("");

ptxt.setText("");

idtxt.setText("");

}

**private void editbtnActionPerformed(java.awt.event.ActionEvent evt) {**

updbtn.setEnabled(true);

Addbtn.setEnabled(true);

ftbtn.setEnabled(false);

ltbtn.setEnabled(false);

ntbtn.setEnabled(false);

psbtn.setEnabled(false);

try

{

id=rs.getInt("USERID");

}

catch(SQLException e) {

JOptionPane.showMessageDialog(this, e);

}

ntxt.setEditable(true);

idtxt.setEditable(true);

ptxt.setEditable(true);

**}**

**private void updbtnActionPerformed(java.awt.event.ActionEvent evt) {**

String naam= ntxt.getText(), pwd=ptxt.getText();

int uid=Integer.parseInt(idtxt.getText());

String query2;

try

{ con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

stmt1=con.createStatement();

int code=JOptionPane.showConfirmDialog(this,"Are you sure you want to update this information?","Confirm?",JOptionPane.YES\_NO\_OPTION);

if(code==JOptionPane.YES\_OPTION)

{

query2="UPDATE LOGIN SET USERID=" + uid + ", USERNAME='" + naam + "',PASSWORD='" + pwd + "' WHERE USERID=" + id +";";

int ss=stmt1.executeUpdate(query2);

if(ss>0) JOptionPane.showMessageDialog(this,"Updation Successful!!");

}

}

catch(SQLException e){

JOptionPane.showMessageDialog(this,e);

}

try

{ con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

stmt=con.createStatement();

String query="SELECT \* FROM LOGIN;";

rs=stmt.executeQuery(query);

rs.next();

displayRecord();

}

catch(SQLException ec)

{

JOptionPane.showMessageDialog(this, ec);

}

**}**

**private void psbtnActionPerformed(java.awt.event.ActionEvent evt) {**

reset();

try {

if (rs.previous()) displayRecord();

else

{

JOptionPane.showMessageDialog(this,"End of Results!!");

rs.first();

}

}

catch(SQLException ec)

{

JOptionPane.showMessageDialog(this, ec);

}

**}**

**private void ltbtnActionPerformed(java.awt.event.ActionEvent evt) {**

reset();

try

{ rs.last();

displayRecord();

}

catch (SQLException ex) {

JOptionPane.showMessageDialog(this, ex);

}

**}**

**private void ftbtnActionPerformed(java.awt.event.ActionEvent evt) {**

reset();

try

{

rs.first();

displayRecord();

}

catch (SQLException ex) {

JOptionPane.showMessageDialog(this, ex);

}

**}**

**private void ntbtnActionPerformed(java.awt.event.ActionEvent evt) {**

reset();

try {

if (rs.next()) displayRecord();

else

{

JOptionPane.showMessageDialog(this,"End of Results!!");

rs.last();

}

}

catch(SQLException ec)

{

JOptionPane.showMessageDialog(this, ec);

}

**}**

**private void cbtnActionPerformed(java.awt.event.ActionEvent evt) {**

this.dispose();

**}**

**private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {**

ntxt1.setText("");

idtxt1.setText("");

displayRecord();

**}**

**private void AddbtnActionPerformed(java.awt.event.ActionEvent evt) {**

String nm=ntxt.getText(), pwd=ptxt.getText();

String uid=idtxt.getText();

String query1="SELECT \* FROM LOGIN WHERE USERID=" + uid + ";";

String query2;

try {

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

stmt1=con.createStatement();

rs1=stmt1.executeQuery(query1);

if(!(rs1.next()))

{

int code=JOptionPane.showConfirmDialog(this,"Insert Record?","Confirm?",JOptionPane.YES\_NO\_OPTION);

if(code==JOptionPane.YES\_OPTION)

{

query2="INSERT INTO LOGIN VALUES(" + uid + ",'" + nm + "','" + pwd + "');";

int ss=stmt1.executeUpdate(query2);

if(ss>0) JOptionPane.showMessageDialog(this,"Updation Successful!!");

else JOptionPane.showMessageDialog(this,"User ID already exists");

clear();

}

}

else {JOptionPane.showMessageDialog(this,"User ID already exists");}

}

catch(SQLException e){

JOptionPane.showMessageDialog(this,e);

}

**}**

**private void spwbtnActionPerformed(java.awt.event.ActionEvent evt) {**

// TODO add your handling code here:

try{

pass=rs.getString("PASSWORD");

ptxt.setText(""+pass);

}

catch(SQLException ec)

{

JOptionPane.showMessageDialog(this, ec);

}

**}**

**private void sbtnActionPerformed(java.awt.event.ActionEvent evt) {**

String sname,sid;

sname=ntxt1.getText();

sid=idtxt1.getText();

Statement stmt;

String query1;

query1="SELECT \* FROM LOGIN WHERE USERID LIKE '%" + sid + "%' AND USERNAME LIKE '%" + sname + "%';";

try

{

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/vista","root","");

stmt=con.createStatement();

rs=stmt.executeQuery(query1);

if(rs.next()) displayRecord();

else

{

JOptionPane.showMessageDialog(this,"No Matching Records Found!!!");

rs=stmt.executeQuery(query);

rs.next();

displayRecord();

}

}

catch(SQLException ex){

JOptionPane.showMessageDialog(this,ex);

}

**}**

# BIBLIOGRAPHY

**Websites**

* INTERNET
* www.google.com
* www.mycbseguide.com
* “informatics practices” by CBSE