**NALANDA COLLEGE OF ENGINEERING**

*Department Of Computer Science and Engineering ,AKU University*



Bachelor In Computer Science and Eng.

**A Project Report On Bihar State Power Holding**

**Company Ltd.**

Submitted By:

**Reg. no. Name**

15105109043 RANJAN KUMAR MANDAL

15105109183 RAJU RANJAN

15105109140 NAHEED PRAVEEN

**SESSION:-2015-2019**

## Certificate of Originality

I hereby declare that the Project entitled “**Electricity Bill Management System**” submitted to the Depatment of Information Technology, **NALANDA COLLEGE OF ENGINEERING, CHANDI** (NALANDA) in partial fulfilment for the award of the Degree of **COMPUTER SCIENCE AND ENGINEERING** in session 2015-2019 in an authentic record of my own work carried out under the guidance of **Mr. RAJEEV SINGH** and that the Project has not previously formed the basis for the award of any other degree.

Place: chandi Signature of the candidate.

Date: 30/04/2019 Ranjan KumarMandal Raju Ranjan

Naheed Praveen

This is to certify that the above statement made by the candidate is correct

to the best of my knowledge.

1. Signature of Internal guide 2. Signature of External Guide

Name Name

Designation Designation

### ACKNOWLEDGEMENT

With Candor and Pleasure, we take opportunity to express our sincere thanks and obligation to our esteemed guide or Sir **Mr. Rajeev Singh**. It is because of his able and mature guidance and co-operation without which it would not have been possible for us to complete our project.

It is our pleasant duty to thank all the staff member of the computer centre who never hesitated us from time during the project.

Finally, we gratefully acknowledge the support, encouragement & patience of our family, and as always, nothing in our life would be possible without God, Thank You!

Date: 30/04/2019 **Signature**

### INDEX

1. Introduction
2. Objective
3. System Analysis
4. Feasibility Study

* Technical Feasibility
* Operational Feasibility
* Economic Feasibility

1. Analysis

* DFD 0 Level,1 Level,2 Level
* E-R Diagram
* Table Structure

1. H/W and S/W Requirement
2. System Design
3. Coding
4. Implementation and Maintenance
5. Testing
6. Modules
7. Scope of future application
8. Conclusion
9. Bibliography/References

****

***Electricity***

***BILL Management System***

### Introduction of the Project Electricity Bill Management System:

The **"Electricity Bill Management System"** has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and in some cases reduce the hardships faced by this existing system. Moreover, this system is designed for the particular need of the company to carry out operations in a smooth and effective manner.

The application is reduced as much as possible to avoid errors while entering the data. It also provides error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus by this all it proves it is user-friendly. Electricity Bill Management System, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus, it will help organization in better utilization of resources.

Every organization, whether big or small, has challenges to overcome and managing the information of Unit of Energy, Electricity, Store Record, Connections, and Electricity Board. Every Electricity Bill Management System has different Electricity needs; therefore, we design exclusive employee management systems that are adapted to your managerial requirements. This is designed to assist in strategic planning, and will help you ensure that your organization is equipped with the right level of information and details for your future goals. Also, for those busy executive who are always on the go, our systems come with remote access features, which will allow you to manage your workforce anytime, at all times. These systems will ultimately allow you to better manage resources.

##### **Objective of Project Electricity Bill Management System:**

In the context of the requirement for partial fulfillment of **COMPUTER SCIENCE AND ENGINEERING** from **Nalanda College of Engineering**, **Chandi**(Nalanda) this project was undertaken to justify the knowledge gained in four years course and to prove the ability to develop real life project in Electricity Bill Management System.

The objective of the Project on Electricity Bill Management System is to manage the details of Electricity, Unit of Energy, Bill, Store Record, and Electricity Board. It manages all the information about Electricity, Connections, Electricity Board, and Electricity.

My Job is to convert these entire manual jobs to a computerized system to make the task of transaction and keeping record more easy and efficient software is the outcome of the this jobs.

The system is implemented in a client server architecture form. It runs on the client machine .There is centralized database (ORACLE) that is portable across different platforms. It runs on the client machine. There are possibilities when more than one user want to access the same table or the same record of the table. Record locking and the table locking.

### System Analysis

**PROPOSED SYSTEM:**

* System needs store information about new entry of Electricity.
* System needs to help the internal staff to keep information of Unit of Energy and find them as per various queries.
* System need to maintain quantity record.
* System need to keep the record of Bill.
* System need to update and delete the record.
* System also needs a search area.
* It also needs a security system to prevent data.

**Identification of need:**

The old manual system was suffering from a series of drawbacks. Since whole of the system was to be maintained with hands the process of keeping, maintaining and retrieving the information was very tedious and lengthy. The records were never used to be in a systematic order. there used to be lots of difficulties in associating any particular transaction with a particular context. If any information was to be found it was required to go through the different registers, documents there would never exist anything like report generation. There would always be unnecessary consumption of time while entering records and retrieving records. One more problem was that it was very difficult to find errors while entering the records. Once the records were entered it was very difficult to update these records.

The reason behind it is that there is lot of information to be maintained and have to be kept in mind while running the business .For this reason we have provided features Present system is partially automated (computerized), actually existing system is quite laborious as one has to enter same information at three different places.

**Following points should be well considered:**

Documents and reports that must be provided by the new system

there can also be few reports, which can help management in decision-

making and cost controlling, but since these reports do not get required

attention, such kind of reports and information were also identified and

given required attention.

Details of the information needed for each document and report.

•The required frequency and distribution for each document.

•Probable sources of information for each document and report.

•With the implementation of computerized system, the task of keeping records in an organized manner will be solved. The greatest of all is the retrieval of information, which will be at the click of the mouse. So the proposed system helps in saving the time in different operations and making information flow easy giving valuable reports.

The Software Requirements Specification is produced at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description, a detailed functional and behavioral description, an indication of performance requirements and design constraints, appropriate validation criteria, and other data pertinent to requirements.

### Feasibility Study:

After doing the project Electricity Bill Management System, study and analyzing all the existing or required functionalities of the system, the next task is to do the feasibility study for the project. All projects are feasible - given unlimited resources and infinite time.

Feasibility study includes consideration of all the possible ways to provide a solution to the given problem. The proposed solution should satisfy all the user requirements and should be flexible enough so that future changes can be easily done based on the future upcoming requirements.

**A. Economic Feasibility**

This is a very important aspect to be considered while developing a project. We decided the technology based on minimum possible cost factor.

* All hardware and software cost has to be borne by the organization.
* Overall we have estimated that the benefits the organization is going to receive from the proposed system will surely overcome the initial costs and the later on running cost for system.

**B. Technical Feasibility**

This is concerned with specifying equipment and software that will successfully satisfy the user requirement. The assessment of technical feasibility must be based on an outline design of the system requirements in terms of input, output, files, programming procedures and staff. This can be qualified in terms of volume of data, trends, frequency of updating, circle of operation, etc. In order to give an indirection of scale and magnitude of the technical requirements. The technical feasibility centers on the existing system and to what extent it can support the proposed system. Previously,**“ MANAGEMENT”** was running with a cumbersme manual system. Now as they are well equipped with the hardware, software and human ware to run this package, the proposed system is technically feasible.

**C. Operational Feasibility**

No doubt the proposed system is fully GUI based that is very user friendly and all inputs to be taken all self-explanatory even to a layman. Besides, a proper training has been conducted to let know the essence of the system to the users so that they feel comfortable with new system. As far our study is concerned the clients are comfortable and happy as the system has cut down their loads and doing.

### 

## 

## 

## Analysis

DATA FLOW DIAGRAMS

**Zero Level DFD**

Bill Generated

Empoyee

ADMIN

Customer

**1-Level DFD**

Generate System User Report

System user

Management

Login Management

Connection Management

Bill Management

Customer Management

Generate Customer Report

Generate Bill Report

Generate Customer Report

Check User Login Details

**2-Level DFD**

ADMIN

Manage Paid Record Details

Manage System Admains

Manage electricity Details

Manage Bill Details

Manage Customer Details

Manage Connection Details

Manage Payment Details

**Table Structure**

1. TABLE NAME : - ADMINISTRATOR

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTRIBUTE** | **DATA TYPE** | **SIZE** | **CONSTRAINTS** |
| **ADMIN\_ID** | **VARCHAR** | **7** | **Primary key** |
| **ADMIN\_PWD** | **VARCAHR** | **7** | **Not null** |

1. TABLE NARE : - LOG\_TABLE

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTRIBUTE** | **DATA TYPE** | **SIZE** | **CONSTRAINTS** |
| **EMP\_ID** | **VARCHAR** | **5** | **Primary key** |
| **EMP\_PWD** | **VARCHAR** | **15** | **Not null** |

1. TABLE NARE : - NEW\_CONN

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTRIBUTE** | **DATA TYPE** | **SIZE** | **CONSTRAINTS** |
| **CON\_ID** | **VARCHAR** | **5** | **Primary key** |
| **CON\_NAME** | **VARCHAR** | **20** | **Not null** |
| **CONN\_TYPE** | **VARCHAR** | **10** | **Not null** |
| **CONN\_STREET** | **VARCHAR** | **20** | **Not null** |
| **CONN\_PLOT** | **VARCHAR** | **20** |  |
| **CONN\_CITY** | **VARCHAR** | **20** |  |
| **CONN\_PIN** | **VARCHAR** | **6** |  |
| **CONN\_STATE** | **VARCHAR** | **15** |  |
| **CONN\_MOB** | **VARCHAR** | **11** |  |
| **LANDLINE** | **VARCHAR** | **12** |  |
| **CON\_EMAIL** | **VARCAHR** | **30** |  |
| **CON\_SAL** | **VARCAHR** | **5** |  |
| **CON\_FATHER** | **VARCAHR** | **30** |  |
| **APPLY\_DATE** | **VARCHAR** | **20** |  |
| **CON\_DOB** | **DATE** |  |  |
| **CON\_GEN** | **VARCHAR** | **20** |  |
| **PHOTO** | **BLOB** |  |  |
| **DUE\_AMOUNT** | **NUMBER** | **15** |  |
| **PRE\_UNIT** | **NUMBER** | **16** |  |
| **PAID\_AMOUNT** | **NUMBER** | **15** |  |
| **PAID\_DATE** | **VARCAHR** | **20** |  |
| **BILL\_NO** | **VARCHAR** | **6** |  |

1. TABLE NARE : - CONTACT\_US

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTRIBUTE** | **DATA TYPE** | **SIZE** | **CONSTRAINTS** |
| **NAME\_UN** | **VARCHAR** | **15** |  |
| **EMAIL\_ID** | **VARCHAR** | **15** |  |
| **MOBILE** | **NUMBER** | **12** |  |
| **SUB** | **VARCHAR** | **15** |  |
| **MSG** | **VARCHAR** | **100** |  |

1. TABLE NARE : - AUTOMATIC

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTRIBUTE** | **DATA TYPE** | **SIZE** | **CONSTRAINTS** |
| **AEMP\_ID** | **NUMBER** | **4** |  |
| **ACONN\_ID** | **NUMBER** | **5** |  |
| **ABILL\_NO** | **NUMBER** | **5** |  |
| **APAY\_NO** | **NUMBER** | **5** |  |
| **AMETER\_NO** | **NUMBER** | **4** |  |

1. TABLE NARE : - SALARY\_PAYMENT

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTRIBUTE** | **DATA TYPE** | **SIZE** | **CONSTRAINTS** |
| **EMP\_ID** | **VARCHAR** | **5** |  |
| **HRA** | **NUMBER** | **5** |  |
| **BONUS** | **NUMBER** | **5** |  |
| **PF** | **NUMBER** | **5** |  |
| **INCOME\_TAX** | **NUMBER** | **5** |  |
| **TOTAL\_SAL** | **NUMBER** | **12** |  |

### Tools And Platform USED:

**Hardware and Software Requirement specifications:**

**Software Requirements:**

|  |  |
| --- | --- |
| Name of component | Specification |
| Operating System | Windows 8, Windows XP, Windows7, |
| Language | Java 2 Runtime Environment |
| Database | Oracle 10g |
| Software Development Kit | Java JDK 1.7 or Above |
| Database JDBC Driver | JDBC thin driver |

**Hardware Requirements:**

|  |  |
| --- | --- |
| Name of component | Specification |
| Processor | Pentium III 630MHz |
| RAM | 500 MB |
| Hard disk | 512 MB |
| Monitor | 15” color monitor |
| Keyboard | 122 keys |

* **About Java**

Java is a general purpose, object-oriented [programming language](http://en.wikipedia.org/wiki/Programming_language) developed by [**Sun Microsystems**](http://en.wikipedia.org/wiki/Sun_Microsystems) ofUSA in 1991.Originally called OAK by **James Gosling** who is theinventor of the language,java was designed for development of software for consumer electronic devices like TVs,VCRs and such other electronic machines. The goal had a strong impact on the development team to make the language simple, portable and highly reliable. the java team which include Patrick naught on discovered that the existing language c/c++ had limitation In terms of both reliability and portability ,their new language java on c/c++ but removed a number of features of c/c++ that were considered as sources of problems and thus made java a really simple ,reliable ,portable and powerful language .

Either usually computer language is compiled or interpreted .java combines both these approach. Thus making java a two-stage system first, java compiler translated source code into byte code instruction. Bytecodes are not machine instructions and therefore, in the second stage java interpreter generates machine code that can be directly executed by the machine that is running the java program.so we could say java is both a compile and interpreted language.

Java is a platform-independent language.java is a highly secure language on internet

**Major release versions of Java:**

* ***JDK 1.0***
* ***JDK 1.1***
* ***JDK 1.2***
* ***JDK 1.3***
* ***JDK 1.4***
* ***JDK 1.5***
* ***JDK 1.7***
* ***JDK 1.8***
* 
* **About Oracle10g**

Oracle is one such RDBMS package which was invented by E.F.Codd and is based on the simple concept i.e. Table. It provides a set of functional programs that we use as tools to build structure and perform tasks. In ORACLE, data is stored and displayed in tables. A table is a data structure that holds data in a relational database. A table comprises of rows and columns. Table can also show relationship between entities. The formal name of a table is relation, hence the name RELATIONAL DATABASE MANAGEMENT SYSTEM.

**Feature of ORACLE10g:**

***1****>****Application development***

These sections describe application development features for oracle database 11g. Application program interface provides features i;e oracle open database connectivity (ODBC) drive support for Linux, oracle call interface client version API,start up and shut down of Database OCI API Instant client Software Development kit(SDK) Java, Java Database Connectivity(JDBC),and web services.

***2> Availability***

It describe Availability features for oracle database 10g. fast-start failover to standby database in oracle database 10g. Data Guard offers a compelling set of capabilities that provide unparalleled levels of data availability, data protection, and data recovery. For example, on an outage of the primary server, site of network, the new fast-start failover feature allow Data Guard to rapidly and automatically failover to the standby database, without requiring manual intervention

***3>SQL Support for Analytic Application***

Oracle extends the SQL language to support more powerful analytics. The MODEL clause of SELECT statement, which provides a powerful new approach to analytic calculations, adds to its expressiveness and scalability in this release.

**4>Clustering**

The following section describe new features for oracle database 10g in clustering area. Such as :-Real Application Clusters(RAC) Performance this release provides session management capabilities that improve the ability to balance workloads across instance based on real time information.

**6>Windows**

The following sections describe new features for oracle Database 10g in the windows area . Improve Availability and Scalability with oracle Real Application Clusters.ODP.NET has been enhanced with greater ability to intelligently use RAC.when Optimizing their application*.*

### System Design

Login Window:



Start Window:



Entry Form:



Bill Generate Form:



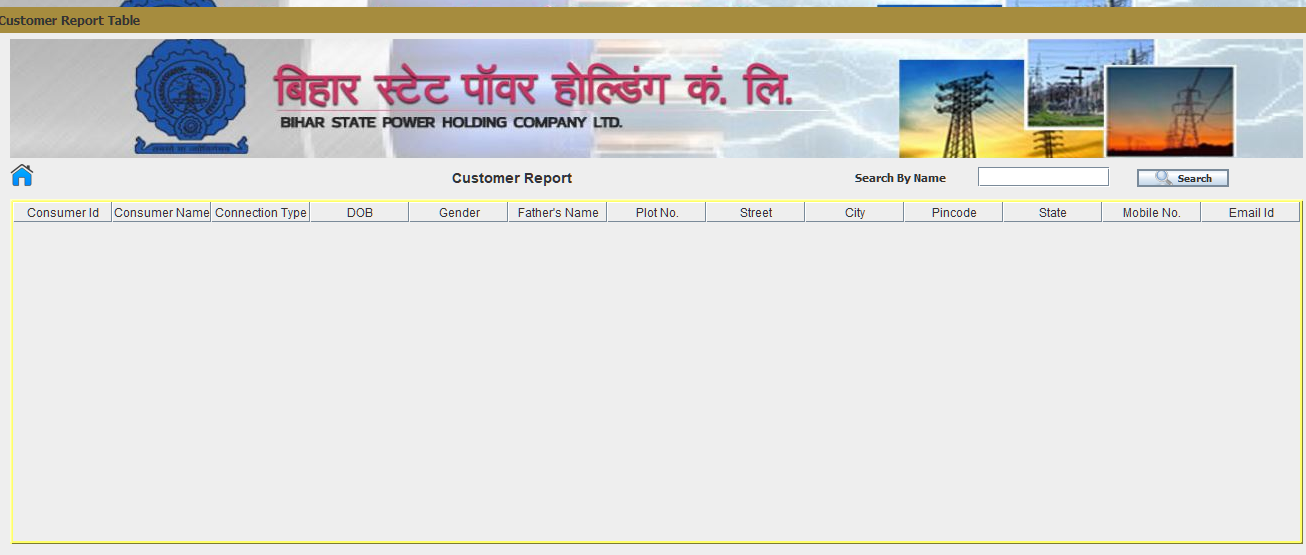
Payment Form:

****

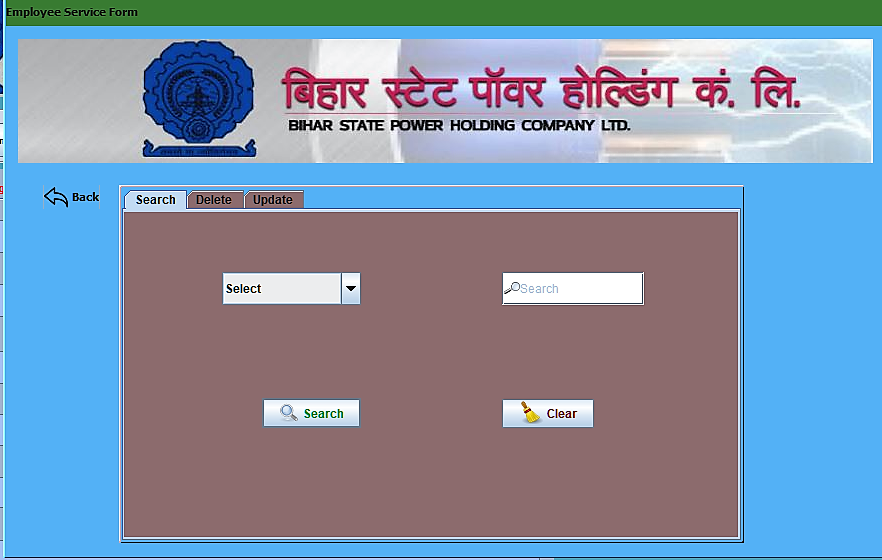
Enquiry Form:



Report Form:



Service Form:



Admin Window:

****

User Creation:



### Coding

***Add Customer***

package Module;

import java.awt.Image;

import java.awt.Toolkit;

import java.awt.event.KeyAdapter;

import java.awt.event.KeyEvent;

import java.io.File;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import java.io.IOException;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.ImageIcon;

import javax.swing.JFileChooser;

import javax.swing.JOptionPane;

public class ConnectionForm extends javax.swing.JFrame {

Connection con;

PreparedStatement stmt;

int cno;

boolean flag;

ResultSet rs;

String cnostr;

File f;

StartForm m;

public ConnectionForm(StartForm m) throws SQLException, ClassNotFoundException {

initComponents();

con=ConnDb.conLink();

this.m=m;

stmt=con.prepareStatement("select aconn\_id from automatic");

rs=stmt.executeQuery();

rs.next();

cno=rs.getInt(1);

cno++;

stmt=con.prepareStatement("Select sysdate from dual");

rs=stmt.executeQuery();

rs.next();

adate\_tf.setText(""+rs.getDate(1));

auto();

conmob\_tf.addKeyListener(new KeyAdapter(){

@Override

public void keyTyped(KeyEvent e){

if(conmob\_tf.getText().length()>=10)

e.consume();

}

});

conpin\_tf.addKeyListener(new KeyAdapter(){

@Override

public void keyTyped(KeyEvent e){

if(conpin\_tf.getText().length()>=6)

e.consume();

}

});

conland\_tf.addKeyListener(new KeyAdapter(){

@Override

public void keyTyped(KeyEvent e){

if(conland\_tf.getText().length()>=10)

e.consume();

}

});

}

private void auto()

{

if(cno<10)

cnostr="C000"+cno;

else if(cno<100)

cnostr="C00"+cno;

else if(cno<1000)

cnostr="C0"+cno;

else if(cno<10000)

cnostr="C"+cno;

else

JOptionPane.showMessageDialog(this,"CONSUMER id exceed its value");

conid\_tf.setText(cnostr);

}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

JFileChooser ch=new JFileChooser();

ch.showOpenDialog(this);

f=ch.getSelectedFile();

ImageIcon img=new ImageIcon(new ImageIcon(""+f).getImage().getScaledInstance(ph.getWidth(),ph.getHeight(),Image.SCALE\_DEFAULT));

ph.setIcon(img);

}

private void conn\_cbActionPerformed(java.awt.event.ActionEvent evt) {

switch (conn\_cb.getSelectedItem().toString()) {

case "Domestic":

price\_tf.setText("10");

break;

case "Commercial":

price\_tf.setText("15");

break;

case "Industrial":

price\_tf.setText("20");

break;

default:

price\_tf.setText(" ");

break;

}

}

private void check() throws SQLException{

if(flag)

{

conname\_tf.setText("");

conn\_cb.setSelectedIndex(0);

plot\_tf.setText("");

conadd\_tf.setText("");

connvill\_tf.setText("");

conpin\_tf.setText("");

constate\_cb.setSelectedIndex(0);

conland\_tf.setText("");

conmob\_tf.setText("");

price\_tf.setText("");

conemail\_tf.setText("");

fcus\_tf.setText("");

con\_dob.setSelectedDate(null);

occu\_tf.setText("");

sal\_cb.setSelectedIndex(0);

gen\_cb.setSelectedIndex(0);

ph.setIcon(null);

next\_bt.setEnabled(false);

stmt=con.prepareStatement("Update automatic set aconn\_id=?");

stmt.setInt(1,cno);

stmt.executeUpdate();

flag=false;

cno++;

auto();

}

}

private void next\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(conn\_cb.getSelectedIndex()==0)

{

JOptionPane.showMessageDialog(this,"Select Connection type!");

}

else if(sal\_cb.getSelectedIndex()==0)

{

JOptionPane.showMessageDialog(this,"Salution must be selected");

}

else if(conname\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Name must be not be empty");

conname\_tf.requestFocus();

}

else if(gen\_cb.getSelectedIndex()==0)

{

JOptionPane.showMessageDialog(this,"Gender must be selected");

gen\_cb.requestFocus();

}

else if(fcus\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Father Name must not be empty");

fcus\_tf.requestFocus();

}

else if(con\_dob.getSelectedDate()==null)

{

JOptionPane.showMessageDialog(this,"Date must be Selected");

con\_dob.requestFocus();

}

else if(occu\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Occupation must not be empty");

occu\_tf.requestFocus();

}

else if(plot\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Plot No. must not be empty");

plot\_tf.requestFocus();

}

else if(conadd\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Street Name must not be empty");

conadd\_tf.requestFocus();

}

else if(connvill\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"City must not be empty");

connvill\_tf.requestFocus();

}

else if(conpin\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Pincode must not be empty");

conpin\_tf.requestFocus();

}

else if(constate\_cb.getSelectedIndex()==0)

{

JOptionPane.showMessageDialog(this,"State must be Selected");

constate\_cb.requestFocus();

}

else if(conemail\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Email must not be empty");

conemail\_tf.requestFocus();

}

else if(conmob\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Mobile No. must not be empty");

conmob\_tf.requestFocus();

}

else if(conland\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Landline must not be empty");

conland\_tf.requestFocus();

}

else if(ph.getIcon()==null)

{

JOptionPane.showMessageDialog(this,"Insert a Photo");

}

else

{

try {

stmt=con.prepareStatement("insert into new\_conn (con\_id,con\_name,conn\_type,conn\_street,conn\_plot,conn\_city,conn\_pin,conn\_state,conn\_mob,occu,landline,con\_email,con\_sal,con\_father,apply\_date,con\_dob,con\_gen,bill\_unit,photo) values(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)");

stmt.setString(1,conid\_tf.getText());

String con\_type=conn\_cb.getSelectedItem().toString();

stmt.setString(3,con\_type);

stmt.setString(2,conname\_tf.getText());

stmt.setString(5,plot\_tf.getText());

stmt.setString(4,conadd\_tf.getText());

stmt.setString(6,connvill\_tf.getText());

stmt.setString(7,conpin\_tf.getText());

String state=constate\_cb.getSelectedItem().toString();

stmt.setString(8,state);

FileInputStream fin= new FileInputStream(f);

stmt.setBinaryStream(19,fin,fin.available());

stmt.setString(11,conland\_tf.getText());

stmt.setString(9,(conmob\_tf.getText()));

stmt.setInt(18,Integer.parseInt(price\_tf.getText()));

stmt.setString(12,conemail\_tf.getText());

stmt.setString(14, fcus\_tf.getText());

stmt.setDate(16, new java.sql.Date(con\_dob.getSelectedDate().getTimeInMillis()));

stmt.setString(10,occu\_tf.getText());

String sal=sal\_cb.getSelectedItem().toString();

stmt.setString(13,sal);

String gen=gen\_cb.getSelectedItem().toString();

stmt.setString(17,gen);

stmt.setString(15,adate\_tf.getText());

int a=JOptionPane.showConfirmDialog(this,"Do you want to insert the record","Record Inserted",JOptionPane.YES\_NO\_OPTION,JOptionPane.INFORMATION\_MESSAGE);

if(a==0)

{

stmt.executeUpdate();

JOptionPane.showMessageDialog(this,"Record Sucessfully Inserted","Inserted",JOptionPane.INFORMATION\_MESSAGE);

stmt=con.prepareStatement("Update new\_conn set due\_amount=0,pre\_unit=0,paid\_amount=0,paid\_date=0 where con\_id=?");

stmt.setString(1,conid\_tf.getText().toUpperCase());

stmt.executeUpdate();

flag=true;

stmt=con.prepareStatement("Select con\_name,con\_id,con\_gen,con\_father,con\_dob,occu,conn\_plot,conn\_street,conn\_city,conn\_pin,conn\_state,conn\_mob,landline,con\_email,conn\_type,bill\_unit from new\_conn where upper(con\_id)=?");

stmt.setString(1,conid\_tf.getText().toUpperCase());

rs=stmt.executeQuery();

new CustomerAccount(rs).setVisible(true);

check();

}

else

{

JOptionPane.showMessageDialog(this,"Record Not Inserted","Inserted",JOptionPane.ERROR\_MESSAGE);

}

} catch (SQLException ex) {

JOptionPane.showMessageDialog(this,"Record Not Sucessfully Inserted","Entery Problem",JOptionPane.ERROR\_MESSAGE);

} catch (FileNotFoundException ex) {

Logger.getLogger(ConnectionForm.class.getName()).log(Level.SEVERE, null, ex);

} catch (IOException | ClassNotFoundException ex) {

Logger.getLogger(ConnectionForm.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

private void clear\_btActionPerformed(java.awt.event.ActionEvent evt) {

conname\_tf.setText("");

conn\_cb.setSelectedIndex(0);

plot\_tf.setText("");

conadd\_tf.setText("");

connvill\_tf.setText("");

conpin\_tf.setText("");

constate\_cb.setSelectedIndex(0);

conland\_tf.setText("");

conmob\_tf.setText("");

price\_tf.setText("");

conemail\_tf.setText("");

fcus\_tf.setText("");

con\_dob.setSelectedDate(null);

occu\_tf.setText("");

sal\_cb.setSelectedIndex(0);

gen\_cb.setSelectedIndex(0);

ph.setIcon(null);

next\_bt.setEnabled(false);

}

private void conname\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume(); }

}

private void fcus\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();

}

}

private void occu\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();

}

}

private void plot\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE|| Character.isDigit(c) ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();

}

}

private void conadd\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||Character.isDigit(c)||c==KeyEvent.VK\_BACK\_SPACE ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();

}

}

private void connvill\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();

}

}

private void conpin\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isDigit(c)||c==KeyEvent.VK\_BACK\_SPACE ||c==KeyEvent.VK\_DELETE )){

evt.consume();

}

}

private void conemail\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if((c==KeyEvent.VK\_SPACE)){

JOptionPane.showMessageDialog(this,"Space Not allowed");

evt.consume();

}

}

private void conmob\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isDigit(c)||c==KeyEvent.VK\_BACK\_SPACE ||c==KeyEvent.VK\_DELETE )){

Toolkit.getDefaultToolkit().beep();

evt.consume();}

}

private void conland\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isDigit(c)||c==KeyEvent.VK\_BACK\_SPACE ||c==KeyEvent.VK\_DELETE )){

evt.consume();

}

}

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

m.con\_bt.addActionListener(m.cons[0]);

m.cf=null;

this.dispose();

}

private void conland\_tfFocusLost(java.awt.event.FocusEvent evt) {

if(conland\_tf.getText().length()==0)

next\_bt.setEnabled(false);

else

next\_bt.setEnabled(true);

}

***Add Employee***

package Module;

import java.awt.event.KeyAdapter;

import java.awt.event.KeyEvent;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

public class Employee extends javax.swing.JFrame {

PreparedStatement stmt;

Connection con;

int ano;

String enostr;

ResultSet rs;

boolean flag;

StartForm m;

public Employee(StartForm m) throws SQLException, ClassNotFoundException {

initComponents();

con=ConnDb.conLink();

this.m=m;

stmt=con.prepareStatement("select aemp\_id from automatic");

rs=stmt.executeQuery();

rs.next();

ano=rs.getInt(1);

ano++;

auto();

emp\_mob.addKeyListener(new KeyAdapter(){

@Override

public void keyTyped(KeyEvent e){

if(emp\_mob.getText().length()>=10)

e.consume();

}

});

emp\_pin.addKeyListener(new KeyAdapter(){

@Override

public void keyTyped(KeyEvent e){

if(emp\_pin.getText().length()>=6)

e.consume();

}

});

}

private void auto()

{

if(ano<10)

enostr="E00"+ano;

else if(ano<100)

enostr="E0"+ano;

else if(ano<1000)

enostr="E"+ano;

else

JOptionPane.showMessageDialog(this,"Employee id exceed its value");

empid\_tf.setText(enostr);

}

private void save\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(emp\_sal.getSelectedIndex()==0)

{

JOptionPane.showMessageDialog(this,"Salutation must Selected!");

}

else if(emp\_first.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Name must not be empty!");

emp\_first.requestFocus();

}

else if(emp\_dob.getSelectedDate()==null)

{

JOptionPane.showMessageDialog(this,"Date of Birth must be selected");

emp\_dob.requestFocus();

}

else if(emp\_nat.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Nationality must not be empty!");

emp\_nat.requestFocus();

}

else if(emp\_post.getSelectedIndex()==0)

{

JOptionPane.showMessageDialog(this,"Desgination must be Selected");

}

else if(emp\_plot.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Plot no. must not be empty");

emp\_plot.requestFocus();

}

else if(emp\_add.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Street Name must not be empty");

emp\_add.requestFocus();

}

else if(emp\_vil.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Village must not be empty");

emp\_vil.requestFocus();

}

else if(emp\_pin.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Pincode must not be empty");

emp\_pin.requestFocus();

}

else if(emp\_state.getSelectedIndex()==0)

{

JOptionPane.showMessageDialog(this,"State must be Selected!");

}

else if(emp\_mob.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Mobile No. must not be empty");

emp\_mob.requestFocus();

}

else if(emp\_email.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Email id must not be empty");

emp\_email.requestFocus();

}

else

{

try

{

stmt=con.prepareStatement("insert into emp\_table values(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)");

stmt.setString(1,empid\_tf.getText());

String sal=emp\_sal.getSelectedItem().toString();

stmt.setString(2,sal);

stmt.setString(3,emp\_first.getText());

String gencb=emp\_gencb.getSelectedItem().toString();

stmt.setString(4,gencb);

stmt.setString(5,emp\_nat.getText());

String post=emp\_post.getSelectedItem().toString();

stmt.setString(15,post);

stmt.setLong(14,Long.parseLong(emp\_bsal.getText()));

stmt.setString(9,emp\_plot.getText());

stmt.setString(8,emp\_add.getText());

stmt.setString(10,emp\_vil.getText());

stmt.setInt(11,Integer.parseInt(emp\_pin.getText()));

String state=emp\_state.getSelectedItem().toString();

stmt.setString(12,state);

stmt.setString(7,emp\_email.getText());

stmt.setLong(6,(Long.parseLong(emp\_mob.getText())));

stmt.setDate(13,new java.sql.Date(emp\_dob.getSelectedDate().getTimeInMillis()));

stmt.executeUpdate();

flag=true;

new\_bt.setEnabled(true);

this.getRootPane().setDefaultButton(new\_bt);

JOptionPane.showMessageDialog(this,"Record Sucessful inserted!","Record enter",JOptionPane.INFORMATION\_MESSAGE);

save\_bt.setEnabled(false);

}

catch(SQLException ex)

{

JOptionPane.showMessageDialog(this, "Record not inserted","Problmes", JOptionPane.ERROR\_MESSAGE);

}

}

}

private void emp\_postActionPerformed(java.awt.event.ActionEvent evt) {

switch (emp\_post.getSelectedItem().toString()) {

case "Managing Director":

emp\_bsal.setText("100000");

break;

case "Engineering Director":

emp\_bsal.setText("90000");

break;

case "Chief Accounts Officer":

emp\_bsal.setText("80000");

break;

case "Chief Engineer":

emp\_bsal.setText("70000");

break;

case "Assitant Engineer":

emp\_bsal.setText("60000");

break;

case "Cheif Water Analyst":

emp\_bsal.setText("50000");

break;

case "Lab Attender":

emp\_bsal.setText("40000");

break;

case "Administrative Officer":

emp\_bsal.setText("250000");

break;

case "Steno\_Typist":

emp\_bsal.setText("20000");

break;

case "Driver":

emp\_bsal.setText("20000");

break;

case "Watchman":

emp\_bsal.setText("3000");

break;

case "Sweeper":

emp\_bsal.setText("2000");

break;

case "Programmer":

emp\_bsal.setText("20000");

break;

default:

emp\_bsal.setText("40000");

break;

}

}

private void new\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(flag)

{

try

{

emp\_sal.setSelectedIndex(0);

emp\_first.setText("");

emp\_gencb.setSelectedIndex(0);

emp\_nat.setText("");

emp\_post.setSelectedIndex(0);

emp\_plot.setText("");

emp\_add.setText("");

emp\_vil.setText("");

emp\_pin.setText("");

emp\_state.setSelectedIndex(0);

emp\_email.setText("");

emp\_mob.setText("");

emp\_bsal.setText("");

emp\_first.requestFocus();

emp\_dob.setSelectedDate(null);

stmt=con.prepareStatement("Update automatic set aemp\_id=?");

stmt.setInt(1,ano);

stmt.executeUpdate();

flag=false;

ano++;

auto();

new\_bt.setEnabled(false);

} catch (SQLException ex) {

Logger.getLogger(Employee.class.getName()).log(Level.SEVERE, null, ex);

}

}

else

JOptionPane.showMessageDialog(this,"Please! Fill the following detail");

}

private void emp\_firstKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();

}

}

private void emp\_natKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();

}

}

private void emp\_plotKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE|| Character.isDigit(c) ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();}

}

private void emp\_addKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();}

}

private void emp\_vilKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();

}

}

private void emp\_pinKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isDigit(c)||c==KeyEvent.VK\_BACK\_SPACE ||c==KeyEvent.VK\_DELETE )){

evt.consume();

}

}

private void emp\_mobKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isDigit(c)||c==KeyEvent.VK\_BACK\_SPACE ||c==KeyEvent.VK\_DELETE )){

evt.consume();}

}

private void emp\_emailKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if((c==KeyEvent.VK\_SPACE)){

JOptionPane.showMessageDialog(this,"Space Not allowed");

evt.consume();

}

}

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

m.newemp\_bt.addActionListener(m.em[0]);

m.emp=null;

this.dispose();

}

private void emp\_emailFocusLost(java.awt.event.FocusEvent evt) {

if(emp\_email.getText().length()==0)

save\_bt.setEnabled(false);

else

{

save\_bt.setEnabled(true);

this.getRootPane().setDefaultButton(save\_bt);

}

}

***Pay Salary***

package Module;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import javax.swing.JOptionPane;

public class Emp\_Salary extends javax.swing.JFrame {

PreparedStatement stmt;

Connection con;

ResultSet rs;

long sal,hra,bonus,pf,it,tsal;

boolean f;

String empid;

StartForm m;

public Emp\_Salary(StartForm m) throws SQLException, ClassNotFoundException {

initComponents();

this.m=m;

empid\_tf.requestFocus();

con=ConnDb.conLink();

}

private void pay\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(empid\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Employee id must not be empty","Entry problem",JOptionPane.ERROR\_MESSAGE);

empid\_tf.requestFocus();

}

else

{

try {

stmt=con.prepareStatement("insert into salary\_payment values(?,?,?,?,?,?)");

stmt.setString(1,empid\_tf.getText());

stmt.setLong(2,Long.parseLong(emphra\_tf.getText()));

stmt.setLong(3,Long.parseLong(empbonus\_tf.getText()));

stmt.setLong(4,Long.parseLong(emppf\_tf.getText()));

stmt.setLong(5,Long.parseLong(empit\_tf.getText()));

stmt.setLong(6,Long.parseLong(empts\_tf.getText()));

stmt.executeUpdate();

JOptionPane.showMessageDialog(this,"Payment sucessful of Employee: "+empid);

pay\_bt.setEnabled(false);

cal\_bt.setEnabled(false);

new\_bt.setEnabled(true);

}

catch (SQLException ex) {

JOptionPane.showMessageDialog(this,"Payment not sucessful of Employee "+empid );

}

}

}

private void new\_btActionPerformed(java.awt.event.ActionEvent evt) {

clear();

new\_bt.setEnabled(false);

}

private void empid\_tfFocusLost(java.awt.event.FocusEvent evt) {

if(empid\_tf.getText().length()==0)

empid\_tf.requestFocus();

else

{

try {

stmt=con.prepareStatement("Select emp\_name,emp\_street,emp\_plot,emp\_city,emp\_pincode,emp\_state,emp\_mob,emp\_sal,emp\_dob,emp\_gen,desg,emp\_basic\_sal from emp\_table where upper(emp\_id)=?");

stmt.setString(1,empid\_tf.getText().toUpperCase());

rs=stmt.executeQuery();

rs.next();

empsalary\_tf.setText(""+rs.getString(12));

empname\_tf.setText(rs.getString(8)+" "+rs.getString(1));

empgen\_tf.setText(rs.getString(10));

empdob\_tf.setText(rs.getString(9));

emppost\_tf.setText(rs.getString(11));

empplot\_tf.setText(rs.getString(3));

empstreet\_tf.setText(rs.getString(2));

emptown\_tf.setText(rs.getString(4));

emppin\_tf.setText(""+rs.getString(5));

empmob\_tf.setText(""+rs.getString(7));

empstate\_tf.setText(rs.getString(6));

sal=Long.parseLong(empsalary\_tf.getText());

hra=(sal\*10)/100;

bonus=(sal\*11)/100;

pf=(sal\*12)/100;

it=(sal\*9)/100;

emphra\_tf.setText(""+hra);

empbonus\_tf.setText(""+bonus);

emppf\_tf.setText(""+pf);

empit\_tf.setText(""+it);

empid=empid\_tf.getText();

cal\_bt.setEnabled(true);

}

catch (SQLException ex)

{

JOptionPane.showMessageDialog(this,"Employee ID not found");

empid\_tf.setText("");

empid\_tf.requestFocus();

}

}

}

private void cal\_btActionPerformed(java.awt.event.ActionEvent evt) {

tsal=(sal+hra+bonus)-(pf+it);

empts\_tf.setText(""+tsal);

}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {

m.empsal\_bt.addActionListener(m.esal[0]);

m.emp\_sal=null;

this.dispose();

}

private void cal\_btMouseClicked(java.awt.event.MouseEvent evt) {

pay\_bt.setEnabled(true);

}

private void clear()

{

empid\_tf.setText("");

empname\_tf.setText("");

empgen\_tf.setText("");

empdob\_tf.setText("");

emppost\_tf.setText("");

empplot\_tf.setText("");

empstreet\_tf.setText("");

emptown\_tf.setText("");

emppin\_tf.setText("");

empmob\_tf.setText("");

empstate\_tf.setText("");

emphra\_tf.setText("");

empbonus\_tf.setText("");

emppf\_tf.setText("");

empit\_tf.setText("");

empts\_tf.setText("");

empsalary\_tf.setText("");

}

private void clear\_btActionPerformed(java.awt.event.ActionEvent evt) {

clear();

}

***Bill Generate***

package Module;

import java.awt.Component;

import java.awt.Dimension;

import java.awt.Graphics;

import java.awt.Graphics2D;

import java.awt.event.KeyEvent;

import java.awt.print.PageFormat;

import java.awt.print.Printable;

import java.awt.print.PrinterException;

import java.awt.print.PrinterJob;

import java.sql.Connection;

import java.sql.Date;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

public class Bill\_generate\_form extends javax.swing.JFrame {

PreparedStatement stmt,stmt1;

Connection con;

ResultSet rs,rs1;

int bno;

String bnostr,contype;

StartForm m;

public Bill\_generate\_form(StartForm m) throws SQLException, ClassNotFoundException {

initComponents();

con=ConnDb.conLink();

this.m=m;

consumer\_tf.requestFocus();

stmt=con.prepareStatement("select abill\_no from automatic");

rs=stmt.executeQuery();

rs.next();

bno=rs.getInt(1);

bno++;

auto();

stmt=con.prepareStatement("Select sysdate from dual");

rs=stmt.executeQuery();

rs.next();

month\_tf.setText(""+rs.getDate(1));

}

private void auto()

{

if(bno<10)

bnostr="B000"+bno;

else if(bno<100)

bnostr="B00"+bno;

else if(bno<1000)

bnostr="B0"+bno;

else if(bno<10000)

bnostr="B"+bno;

else

JOptionPane.showMessageDialog(this,"Bill id exceed its value");

bill\_tf.setText(bnostr);

}

private void submit\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(consumer\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Cosnumer Id must not be empty");

consumer\_tf.requestFocus();

}

else if(cur\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Current reading must be filled");

cur\_tf.requestFocus();

}

else

{

try

{

stmt=con.prepareStatement("insert into bill\_report values(?,?,?,?)");

stmt.setString(1,bill\_tf.getText());

stmt.setString(2, consumer\_tf.getText());

stmt.setDate(3,Date.valueOf(month\_tf.getText()));

stmt.setInt(4,(Integer.parseInt(netpay\_tf.getText())));

stmt.executeUpdate();

submit\_bt.setEnabled(false);

new\_bt.setEnabled(true);

stmt=con.prepareStatement("Update new\_conn set pre\_unit=?,due\_amount=?,bill\_no=? where upper(con\_id)='"+consumer\_tf.getText()+"'");

stmt.setInt(1,Integer.parseInt(cur\_tf.getText()));

stmt.setInt(2,Integer.parseInt(netpay\_tf.getText()));

stmt.setString(3,(bill\_tf.getText()));

stmt.executeUpdate();

JOptionPane.showMessageDialog(this,"Bill Sucessfully generated","record enter",JOptionPane.INFORMATION\_MESSAGE);

int r=JOptionPane.showConfirmDialog(null,"DO U WANT TO PRINT THE RECORD","Confirmation Window",JOptionPane.YES\_NO\_OPTION);

if(r==0)

{

submit\_bt.setVisible(false);

new\_bt.setVisible(false);

jButton3.setVisible(false);

PrinterJob pjob = PrinterJob.getPrinterJob();

PageFormat preformat = pjob.defaultPage();

preformat.setOrientation(PageFormat.LANDSCAPE);

PageFormat postformat = pjob.pageDialog(preformat);

//If user does not hit cancel then print.

if (preformat != postformat) {

//Set print component

pjob.setPrintable(new PaymentForm.Printer(this), postformat);

if (pjob.printDialog()) {

pjob.print();

}

}

submit\_bt.setVisible(true);

new\_bt.setVisible(true);

jButton3.setVisible(true);

}

}catch (SQLException ex) {

JOptionPane.showMessageDialog(this,"Bill sucessfully not generated" );

} catch (PrinterException ex) {

Logger.getLogger(PaymentForm.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

private void consumer\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE|| Character.isDigit(c) ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();

}

}

private void cur\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isDigit(c)||c==KeyEvent.VK\_BACK\_SPACE ||c==KeyEvent.VK\_DELETE )){

evt.consume();

}

}

private void consumer\_tfFocusLost(java.awt.event.FocusEvent evt) {

try {

stmt=con.prepareStatement("Select con\_id,con\_name,conn\_type,due\_amount,pre\_unit,paid\_amount,paid\_date from new\_conn where upper(con\_id)=?");

stmt.setString(1,consumer\_tf.getText().toUpperCase());

rs=stmt.executeQuery();

rs.next();

stmt1=con.prepareStatement("Select meter\_no from customer\_account where upper(con\_id)=?");

stmt1.setString(1,consumer\_tf.getText().toUpperCase());

rs1=stmt1.executeQuery();

rs1.next();

conname\_bill.setText(rs.getString(2));

meter\_tf.setText(rs1.getString(1));

conn\_tf.setText(rs.getString(3));

pre\_tf.setText(""+rs.getString(5));

dueamount\_tf.setText(""+rs.getString(4));

lastpaid\_tf.setText(""+rs.getString(7));

lastpaidamount\_tf.setText(""+rs.getString(6));

cur\_tf.requestFocus();

} catch (SQLException ex) {

JOptionPane.showMessageDialog(this,"Consumer Not found");

}

}

private void cur\_tfFocusLost(java.awt.event.FocusEvent evt) {

int a;

if(cur\_tf.getText().length()==0)

submit\_bt.setEnabled(false);

else

{

contype=conn\_tf.getText();

int cur=Integer.parseInt(cur\_tf.getText());

int pre=Integer.parseInt(pre\_tf.getText());

if(cur>=pre)

{

int unit=cur-pre;

total\_tf.setText(""+unit);

switch (contype) {

case "Domestic":

fixed\_tf.setText("25");

mrent\_tf.setText("20");

a=(int)(unit\*10);

break;

case "Commercial":

fixed\_tf.setText("35");

mrent\_tf.setText("30");

a=(int)(unit\*15);

break;

default:

fixed\_tf.setText("45");

mrent\_tf.setText("40");

a=(int)(unit\*20);

break;

}

int fixed=Integer.parseInt(fixed\_tf.getText());

int mrent=Integer.parseInt(mrent\_tf.getText());

int ta=a+fixed+mrent;

ta\_tf.setText(""+ta);

int due=Integer.parseInt(dueamount\_tf.getText());

int net=(int)ta+due;

netpay\_tf.setText(""+net);

submit\_bt.setEnabled(true);

this.getRootPane().setDefaultButton(submit\_bt);

}

else

{

JOptionPane.showMessageDialog(this,"Current reading is smaller than previous reading");

cur\_tf.setText("");

cur\_tf.requestFocus();

}

}

}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

m.bill\_bt.addActionListener(m.bg[0]);

m.bill=null;

this.dispose();

}

private void submit\_btMouseClicked(java.awt.event.MouseEvent evt) {

new\_bt.setEnabled(true);

}

private void new\_btMouseClicked(java.awt.event.MouseEvent evt) {

new\_bt.setEnabled(false);

}

private void new\_btActionPerformed(java.awt.event.ActionEvent evt) {

try {

dueamount\_tf.setText("");

netpay\_tf.setText("");

conname\_bill.setText("");

lastpaidamount\_tf.setText("");

lastpaid\_tf.setText("");

ta\_tf.setText("");

mrent\_tf.setText("");

fixed\_tf.setText("");

total\_tf.setText("");

cur\_tf.setText("");

pre\_tf.setText("");

meter\_tf.setText("");

conn\_tf.setText("");

consumer\_tf.setText("");

stmt=con.prepareStatement("Update automatic set abill\_no=?");

stmt.setInt(1,bno);

stmt.executeUpdate();

bno++;

auto();

} catch (SQLException ex) {

Logger.getLogger(Bill\_generate\_form.class.getName()).log(Level.SEVERE, null, ex);

}

}

public static class Printer implements Printable {

final Component comp;

public Printer(Component comp){

this.comp = comp;

}

@Override

public int print(Graphics g, PageFormat format, int page\_index)

throws PrinterException {

if (page\_index > 0) {

return Printable.NO\_SUCH\_PAGE;

}

// get the bounds of the component

Dimension dim = comp.getSize();

double cHeight = dim.getHeight();

double cWidth = dim.getWidth();

// get the bounds of the printable area

double pHeight = format.getImageableHeight();

double pWidth = format.getImageableWidth();

double pXStart = format.getImageableX();

double pYStart = format.getImageableY();

double xRatio = pWidth / cWidth;

double yRatio = pHeight / cHeight;

Graphics2D g2 = (Graphics2D) g;

g2.translate(pXStart, pYStart);

g2.scale(xRatio, yRatio);

comp.paint(g2);

return Printable.PAGE\_EXISTS;

}

***Bill Search***

*package bsphcl;*

*import java.sql.ResultSet;*

*import java.sql.SQLException;*

*public class Bill\_Search\_Show extends javax.swing.JFrame {*

*ResultSet rs;*

*ResultSet rs1;*

*public Bill\_Search\_Show(ResultSet rs,ResultSet rs1) throws SQLException {*

*this.rs=rs;*

*this.rs1=rs1;*

*rs.next();*

*rs1.next();*

*initComponents();*

*disp();*

*this.getRootPane().setDefaultButton(back\_bt);*

*}*

*@SuppressWarnings("unchecked")*

*// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents*

*private void initComponents() {*

*jPanel5 = new javax.swing.JPanel();*

*jPanel3 = new javax.swing.JPanel();*

*jPanel1 = new javax.swing.JPanel();*

*jLabel5 = new javax.swing.JLabel();*

*netpay\_show = new javax.swing.JTextField();*

*billno\_show = new javax.swing.JTextField();*

*jLabel15 = new javax.swing.JLabel();*

*jLabel4 = new javax.swing.JLabel();*

*jLabel16 = new javax.swing.JLabel();*

*conname\_show = new javax.swing.JTextField();*

*jLabel13 = new javax.swing.JLabel();*

*jLabel2 = new javax.swing.JLabel();*

*pdate\_tf = new javax.swing.JTextField();*

*jLabel10 = new javax.swing.JLabel();*

*prereading\_show = new javax.swing.JTextField();*

*conid\_show = new javax.swing.JTextField();*

*jLabel7 = new javax.swing.JLabel();*

*damount\_show1 = new javax.swing.JTextField();*

*netpay\_show2 = new javax.swing.JTextField();*

*jLabel8 = new javax.swing.JLabel();*

*jLabel1 = new javax.swing.JLabel();*

*bdate\_tf = new javax.swing.JTextField();*

*dues\_tf = new javax.swing.JTextField();*

*back\_bt = new javax.swing.JButton();*

*jLabel12 = new javax.swing.JLabel();*

*jPanel6 = new javax.swing.JPanel();*

*jLabel17 = new javax.swing.JLabel();*

*jPanel2 = new javax.swing.JPanel();*

*jLabel3 = new javax.swing.JLabel();*

*setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE\_ON\_CLOSE);*

*setTitle("Bill Search");*

*setUndecorated(true);*

*setResizable(false);*

*jPanel5.setBackground(new java.awt.Color(0, 153, 255));*

*jPanel3.setBackground(new java.awt.Color(204, 204, 255));*

*jPanel1.setBackground(new java.awt.Color(136, 87, 87));*

*jPanel1.setBorder(javax.swing.BorderFactory.createEtchedBorder());*

*jLabel5.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N*

*jLabel5.setText("Previous Reading");*

*netpay\_show.setEditable(false);*

*billno\_show.setEditable(false);*

*jLabel15.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N*

*jLabel15.setText("Last Paid Date");*

*jLabel4.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N*

*jLabel4.setText("Consumer Id");*

*jLabel16.setText("Bill Date");*

*conname\_show.setEditable(false);*

*jLabel13.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N*

*jLabel13.setText("Last Paid Amount");*

*jLabel2.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N*

*jLabel2.setText("Bill No.");*

*pdate\_tf.setEditable(false);*

*jLabel10.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N*

*jLabel10.setText("Net Payable");*

*prereading\_show.setEditable(false);*

*conid\_show.setEditable(false);*

*jLabel7.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N*

*jLabel7.setText("Dues");*

*damount\_show1.setEditable(false);*

*netpay\_show2.setEditable(false);*

*jLabel8.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N*

*jLabel8.setText("Consumer Name");*

*jLabel1.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N*

*jLabel1.setText("Bill Date");*

*bdate\_tf.setEditable(false);*

*dues\_tf.setEditable(false);*

*javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);*

*jPanel1.setLayout(jPanel1Layout);*

*jPanel1Layout.setHorizontalGroup(*

*jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGroup(jPanel1Layout.createSequentialGroup()*

*.addGap(28, 28, 28)*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)*

*.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGroup(jPanel1Layout.createSequentialGroup()*

*.addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED\_SIZE, 114, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)*

*.addComponent(billno\_show, javax.swing.GroupLayout.PREFERRED\_SIZE, 116, javax.swing.GroupLayout.PREFERRED\_SIZE))*

*.addGroup(jPanel1Layout.createSequentialGroup()*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()*

*.addComponent(jLabel10, javax.swing.GroupLayout.PREFERRED\_SIZE, 77, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addGap(41, 41, 41))*

*.addGroup(jPanel1Layout.createSequentialGroup()*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addComponent(jLabel5, javax.swing.GroupLayout.PREFERRED\_SIZE, 114, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(jLabel15, javax.swing.GroupLayout.PREFERRED\_SIZE, 89, javax.swing.GroupLayout.PREFERRED\_SIZE))*

*.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)))*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addComponent(netpay\_show, javax.swing.GroupLayout.PREFERRED\_SIZE, 116, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(prereading\_show, javax.swing.GroupLayout.PREFERRED\_SIZE, 116, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(pdate\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 116, javax.swing.GroupLayout.PREFERRED\_SIZE))))*

*.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()*

*.addComponent(jLabel16, javax.swing.GroupLayout.PREFERRED\_SIZE, 77, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)*

*.addComponent(netpay\_show2, javax.swing.GroupLayout.PREFERRED\_SIZE, 116, javax.swing.GroupLayout.PREFERRED\_SIZE))*

*.addGroup(jPanel1Layout.createSequentialGroup()*

*.addComponent(jLabel8)*

*.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)*

*.addComponent(conname\_show, javax.swing.GroupLayout.PREFERRED\_SIZE, 116, javax.swing.GroupLayout.PREFERRED\_SIZE)))*

*.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 138, Short.MAX\_VALUE)*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addComponent(jLabel13, javax.swing.GroupLayout.PREFERRED\_SIZE, 100, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addGroup(jPanel1Layout.createSequentialGroup()*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addComponent(jLabel7, javax.swing.GroupLayout.PREFERRED\_SIZE, 76, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(jLabel4, javax.swing.GroupLayout.PREFERRED\_SIZE, 74, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 74, javax.swing.GroupLayout.PREFERRED\_SIZE))*

*.addGap(66, 66, 66)*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addComponent(bdate\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)*

*.addComponent(damount\_show1, javax.swing.GroupLayout.Alignment.LEADING, javax.swing.GroupLayout.PREFERRED\_SIZE, 116, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(conid\_show, javax.swing.GroupLayout.PREFERRED\_SIZE, 116, javax.swing.GroupLayout.PREFERRED\_SIZE))*

*.addComponent(dues\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))))*

*.addGap(50, 50, 50))*

*);*

*jPanel1Layout.linkSize(javax.swing.SwingConstants.HORIZONTAL, new java.awt.Component[] {bdate\_tf, conid\_show, dues\_tf});*

*jPanel1Layout.setVerticalGroup(*

*jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGroup(jPanel1Layout.createSequentialGroup()*

*.addContainerGap()*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)*

*.addComponent(billno\_show, javax.swing.GroupLayout.PREFERRED\_SIZE, 31, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED\_SIZE, 25, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(conid\_show, javax.swing.GroupLayout.PREFERRED\_SIZE, 31, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(jLabel4, javax.swing.GroupLayout.PREFERRED\_SIZE, 25, javax.swing.GroupLayout.PREFERRED\_SIZE))*

*.addGroup(jPanel1Layout.createSequentialGroup()*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGroup(jPanel1Layout.createSequentialGroup()*

*.addGap(121, 121, 121)*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)*

*.addComponent(jLabel5, javax.swing.GroupLayout.PREFERRED\_SIZE, 25, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(prereading\_show, javax.swing.GroupLayout.PREFERRED\_SIZE, 31, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(jLabel7)*

*.addComponent(dues\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)))*

*.addGroup(jPanel1Layout.createSequentialGroup()*

*.addGap(58, 58, 58)*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)*

*.addComponent(jLabel8)*

*.addComponent(conname\_show, javax.swing.GroupLayout.PREFERRED\_SIZE, 31, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(jLabel1)*

*.addComponent(bdate\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))))*

*.addGap(36, 36, 36)*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)*

*.addComponent(netpay\_show, javax.swing.GroupLayout.PREFERRED\_SIZE, 31, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(jLabel10, javax.swing.GroupLayout.PREFERRED\_SIZE, 25, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(damount\_show1, javax.swing.GroupLayout.PREFERRED\_SIZE, 31, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(jLabel13, javax.swing.GroupLayout.PREFERRED\_SIZE, 25, javax.swing.GroupLayout.PREFERRED\_SIZE))*

*.addGap(30, 30, 30)*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)*

*.addComponent(jLabel15)*

*.addComponent(pdate\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 31, javax.swing.GroupLayout.PREFERRED\_SIZE))*

*.addGap(91, 91, 91)*

*.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)*

*.addComponent(netpay\_show2, javax.swing.GroupLayout.PREFERRED\_SIZE, 31, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addComponent(jLabel16))))*

*.addContainerGap())*

*);*

*jPanel1Layout.linkSize(javax.swing.SwingConstants.VERTICAL, new java.awt.Component[] {bdate\_tf, conid\_show, dues\_tf});*

*back\_bt.setForeground(new java.awt.Color(204, 0, 51));*

*back\_bt.setIcon(new javax.swing.ImageIcon("E:\\pro\_ele\\pro\_module\\Project\\src\\Image\\if\_Arrow\_Left\_1063872.png")); // NOI18N*

*back\_bt.setText("Back");*

*back\_bt.setOpaque(false);*

*back\_bt.addActionListener(new java.awt.event.ActionListener() {*

*public void actionPerformed(java.awt.event.ActionEvent evt) {*

*back\_btActionPerformed(evt);*

*}*

*});*

*jLabel12.setFont(new java.awt.Font("Arial", 1, 14)); // NOI18N*

*jLabel12.setText("Bill Details");*

*javax.swing.GroupLayout jPanel3Layout = new javax.swing.GroupLayout(jPanel3);*

*jPanel3.setLayout(jPanel3Layout);*

*jPanel3Layout.setHorizontalGroup(*

*jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGroup(jPanel3Layout.createSequentialGroup()*

*.addContainerGap(26, Short.MAX\_VALUE)*

*.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel3Layout.createSequentialGroup()*

*.addComponent(jLabel12)*

*.addGap(263, 263, 263)*

*.addComponent(back\_bt)*

*.addGap(28, 28, 28))*

*.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel3Layout.createSequentialGroup()*

*.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addGap(18, 18, 18))))*

*);*

*jPanel3Layout.setVerticalGroup(*

*jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel3Layout.createSequentialGroup()*

*.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)*

*.addGroup(jPanel3Layout.createSequentialGroup()*

*.addComponent(jLabel12)*

*.addGap(19, 19, 19))*

*.addGroup(jPanel3Layout.createSequentialGroup()*

*.addComponent(back\_bt, javax.swing.GroupLayout.PREFERRED\_SIZE, 24, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)))*

*.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 315, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addGap(12, 12, 12))*

*);*

*jLabel17.setBackground(new java.awt.Color(204, 153, 0));*

*jLabel17.setIcon(new javax.swing.ImageIcon("E:\\pro\_ele\\pro\_module\\Project\\src\\Image\\Capture7.JPG")); // NOI18N*

*javax.swing.GroupLayout jPanel6Layout = new javax.swing.GroupLayout(jPanel6);*

*jPanel6.setLayout(jPanel6Layout);*

*jPanel6Layout.setHorizontalGroup(*

*jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGroup(jPanel6Layout.createSequentialGroup()*

*.addComponent(jLabel17, javax.swing.GroupLayout.PREFERRED\_SIZE, 858, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addGap(0, 0, Short.MAX\_VALUE))*

*);*

*jPanel6Layout.setVerticalGroup(*

*jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addComponent(jLabel17, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)*

*);*

*jPanel2.setBackground(new java.awt.Color(140, 85, 142));*

*jLabel3.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N*

*jLabel3.setText("Bill Search");*

*javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);*

*jPanel2.setLayout(jPanel2Layout);*

*jPanel2Layout.setHorizontalGroup(*

*jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGroup(jPanel2Layout.createSequentialGroup()*

*.addComponent(jLabel3)*

*.addGap(0, 0, Short.MAX\_VALUE))*

*);*

*jPanel2Layout.setVerticalGroup(*

*jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGroup(jPanel2Layout.createSequentialGroup()*

*.addComponent(jLabel3)*

*.addGap(0, 5, Short.MAX\_VALUE))*

*);*

*javax.swing.GroupLayout jPanel5Layout = new javax.swing.GroupLayout(jPanel5);*

*jPanel5.setLayout(jPanel5Layout);*

*jPanel5Layout.setHorizontalGroup(*

*jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)*

*.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel5Layout.createSequentialGroup()*

*.addContainerGap()*

*.addComponent(jPanel6, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)*

*.addContainerGap())*

*.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel5Layout.createSequentialGroup()*

*.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)*

*.addComponent(jPanel3, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addGap(51, 51, 51))*

*);*

*jPanel5Layout.setVerticalGroup(*

*jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel5Layout.createSequentialGroup()*

*.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)*

*.addComponent(jPanel6, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)*

*.addComponent(jPanel3, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)*

*.addGap(23, 23, 23))*

*);*

*javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());*

*getContentPane().setLayout(layout);*

*layout.setHorizontalGroup(*

*layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addComponent(jPanel5, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)*

*);*

*layout.setVerticalGroup(*

*layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addComponent(jPanel5, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)*

*);*

*pack();*

*setLocationRelativeTo(null);*

*}// </editor-fold>//GEN-END:initComponents*

*private void back\_btActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_back\_btActionPerformed*

*this.dispose();*

*}//GEN-LAST:event\_back\_btActionPerformed*

*public void disp() throws SQLException*

*{*

*billno\_show.setText(rs.getString(1));*

*conid\_show.setText(rs.getString(2));*

*bdate\_tf.setText(rs.getString(3));*

*netpay\_show.setText(rs.getString(4));*

*conname\_show.setText(rs1.getString(1));*

*prereading\_show.setText(rs1.getString(3));*

*dues\_tf.setText(rs1.getString(2));*

*damount\_show1.setText(rs1.getString(4));*

*pdate\_tf.setText(rs1.getString(5));*

*}*

*// Variables declaration - do not modify//GEN-BEGIN:variables*

*private javax.swing.JButton back\_bt;*

*private javax.swing.JTextField bdate\_tf;*

*private javax.swing.JTextField billno\_show;*

*private javax.swing.JTextField conid\_show;*

*private javax.swing.JTextField conname\_show;*

*private javax.swing.JTextField damount\_show1;*

*private javax.swing.JTextField dues\_tf;*

*private javax.swing.JLabel jLabel1;*

*private javax.swing.JLabel jLabel10;*

*private javax.swing.JLabel jLabel12;*

*private javax.swing.JLabel jLabel13;*

*private javax.swing.JLabel jLabel15;*

*private javax.swing.JLabel jLabel16;*

*private javax.swing.JLabel jLabel17;*

*private javax.swing.JLabel jLabel2;*

*private javax.swing.JLabel jLabel3;*

*private javax.swing.JLabel jLabel4;*

*private javax.swing.JLabel jLabel5;*

*private javax.swing.JLabel jLabel7;*

*private javax.swing.JLabel jLabel8;*

*private javax.swing.JPanel jPanel1;*

*private javax.swing.JPanel jPanel2;*

*private javax.swing.JPanel jPanel3;*

*private javax.swing.JPanel jPanel5;*

*private javax.swing.JPanel jPanel6;*

*private javax.swing.JTextField netpay\_show;*

*private javax.swing.JTextField netpay\_show2;*

*private javax.swing.JTextField pdate\_tf;*

*private javax.swing.JTextField prereading\_show;*

*// End of variables declaration//GEN-END:variables*

*}*

***Customer Service***

package Module;

import java.awt.event.KeyEvent;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

public class CustomerSF extends javax.swing.JFrame {

String s1,s2;

ResultSet rs,rs1;

Connection con;

PreparedStatement stmt;

StartForm m;

public CustomerSF(StartForm m) throws SQLException, ClassNotFoundException {

initComponents();

this.m=m;

con=ConnDb.conLink();

}

private void search\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(searchcon\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Search field must not be empty");

searchcon\_tf.requestFocus();

}

else

{

try{

s1=(String)search\_con.getSelectedItem();

if(s1.equalsIgnoreCase("Consumer Id"))

{

s1="select con\_name,conn\_type,con\_id,conn\_street,conn\_plot,conn\_city,conn\_pin,conn\_state,conn\_mob,con\_email,con\_sal,con\_father,con\_dob,con\_gen,bill\_unit,occu from new\_conn where upper(con\_id)=?";

s2="Select meter\_no from customer\_account where upper(con\_id)=?";

}

else

{

s1="select con\_name,conn\_type,con\_id,conn\_street,conn\_plot,conn\_city,conn\_pin,conn\_state,conn\_mob,con\_email,con\_sal,con\_father,con\_dob,con\_gen,bill\_unit,occu from new\_conn where upper(con\_name)=?";

s2="Select meter\_no from customer\_account where upper(con\_name)=?";

}

stmt=con.prepareStatement(s1,ResultSet.TYPE\_SCROLL\_SENSITIVE,ResultSet.CONCUR\_READ\_ONLY);

stmt.setString(1,searchcon\_tf.getText().toUpperCase());

rs=stmt.executeQuery();

stmt=con.prepareStatement(s2,ResultSet.TYPE\_SCROLL\_SENSITIVE,ResultSet.CONCUR\_READ\_ONLY);

stmt.setString(1,searchcon\_tf.getText().toUpperCase());

rs1=stmt.executeQuery();

new Customer\_Search\_Show\_Sf(rs,rs1).setVisible(true);

} catch(SQLException ex)

{

JOptionPane.showMessageDialog(this,"Consumer Not Found!","Record Enter",JOptionPane.ERROR\_MESSAGE);

searchcon\_tf.setText("");

searchcon\_tf.requestFocus();

} catch (ClassNotFoundException ex) {

Logger.getLogger(CustomerSF.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

private void update\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(update\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Update field must not be empty");

update\_tf.requestFocus();

} else

{

try{

s1=(String)search\_update.getSelectedItem();

if(s1.equalsIgnoreCase("Consumer Id"))

s1="select con\_name,conn\_type,con\_id,conn\_street,conn\_plot,conn\_city,conn\_pin,conn\_mob,con\_email,bill\_unit from new\_conn where upper(con\_id)=?";

else

s1="select con\_name,conn\_type,con\_id,conn\_street,conn\_plot,conn\_city,conn\_pin,conn\_mob,con\_email,bill\_unit from new\_conn where upper(con\_name)=?";

stmt=con.prepareStatement(s1,ResultSet.TYPE\_SCROLL\_SENSITIVE,ResultSet.CONCUR\_READ\_ONLY);

stmt.setString(1,update\_tf.getText().toUpperCase());

rs=stmt.executeQuery();

new Customer\_Update\_Show(rs).setVisible(true);

} catch(SQLException ex)

{

JOptionPane.showMessageDialog(this,"Consumer Not Found!","record enter",JOptionPane.ERROR\_MESSAGE);

update\_tf.setText("");

update\_tf.requestFocus();

} catch (ClassNotFoundException ex) {

Logger.getLogger(CustomerSF.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

private void del\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(condel\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this," Field must not be empty!");

condel\_tf.requestFocus();

}

else

{

try{

s1=(String)condel\_cb.getSelectedItem();

if(s1.equalsIgnoreCase("Consumer Id"))

{

s1="select con\_name,conn\_type,con\_id,conn\_street,conn\_plot,conn\_city,conn\_pin,conn\_state,conn\_mob,con\_sal,con\_father,con\_dob,con\_gen,occu from new\_conn where upper(con\_id)=?";

s2="select meter\_no from customer\_account where upper(con\_id)=?";

}

else

{

s1="select con\_name,conn\_type,con\_id,conn\_street,conn\_plot,conn\_city,conn\_pin,conn\_state,conn\_mob,con\_sal,con\_father,con\_dob,con\_gen,occu from new\_conn where upper(con\_name)=?";

s2="select meter\_no from customer\_account where upper(con\_name)=?";

}stmt=con.prepareStatement(s1,ResultSet.TYPE\_SCROLL\_SENSITIVE,ResultSet.CONCUR\_READ\_ONLY);

stmt.setString(1,condel\_tf.getText().toUpperCase());

rs=stmt.executeQuery();

stmt=con.prepareStatement(s2,ResultSet.TYPE\_SCROLL\_SENSITIVE,ResultSet.CONCUR\_READ\_ONLY);

stmt.setString(1,condel\_tf.getText().toUpperCase());

rs1=stmt.executeQuery();

new Custormer\_Del\_form(rs,rs1).setVisible(true);

} catch(SQLException ex)

{

JOptionPane.showMessageDialog(this,"Record Not Found!","Record enter",JOptionPane.ERROR\_MESSAGE);

condel\_tf.setText("");

condel\_tf.requestFocus();

} catch (ClassNotFoundException ex) {

Logger.getLogger(CustomerSF.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

private void search\_clearActionPerformed(java.awt.event.ActionEvent evt) {

searchcon\_tf.setText("");

}

private void tabpaneStateChanged(javax.swing.event.ChangeEvent evt) {

switch(tabpane.getSelectedIndex())

{

case 0: this.getRootPane().setDefaultButton(search\_bt); break;

case 1: this.getRootPane().setDefaultButton(del\_bt); break;

case 2: this.getRootPane().setDefaultButton(update\_bt); break;

}

}

private void del\_clearActionPerformed(java.awt.event.ActionEvent evt) {

condel\_tf.setText("");

}

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

m.conser\_bt.addActionListener(m.cs[0]);

m.csf=null;

this.dispose();

}

private void condel\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE|| Character.isDigit(c) ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();

}

}

private void update\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE|| Character.isDigit(c) ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();

}

}

private void searchcon\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE|| Character.isDigit(c) ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();

}

}

private void update\_clearActionPerformed(java.awt.event.ActionEvent evt) {

update\_tf.setText("");

}

***Employee Service***

package Module;

import java.awt.Container;

import java.awt.KeyboardFocusManager;

import java.awt.event.KeyEvent;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.HashSet;

import java.util.Set;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

import javax.swing.KeyStroke;

public class EmployeeSF extends javax.swing.JFrame {

PreparedStatement stmt;

Connection con;

ResultSet rs;

String s1;

Container c;

StartForm m;

public EmployeeSF(StartForm m) throws SQLException, ClassNotFoundException {

initComponents();

con=ConnDb.conLink();

this.m=m;

Set newkey=new HashSet();

newkey.add(KeyStroke.getKeyStroke("TAB"));

tabpane.setFocusTraversalKeys(KeyboardFocusManager.FORWARD\_TRAVERSAL\_KEYS,newkey);

}

private void empsearch\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(search\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Search field must not be empty");

search\_tf.requestFocus();

}

else

{

try{

s1=(String)cb\_emp.getSelectedItem();

if(s1.equalsIgnoreCase("Employee Id"))

s1="select \* from emp\_table where upper(emp\_id)=?";

else

s1="select \* from emp\_table where upper(emp\_name)=?";

stmt=con.prepareStatement(s1,ResultSet.TYPE\_SCROLL\_SENSITIVE,ResultSet.CONCUR\_READ\_ONLY);

stmt.setString(1,search\_tf.getText().toUpperCase());

rs=stmt.executeQuery();

new Emp\_Search\_form(rs).setVisible(true);

} catch(SQLException ex)

{

JOptionPane.showMessageDialog(this,"Record Not Found!","recod enter",JOptionPane.ERROR\_MESSAGE);}}

}

private void empdel\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(empdel\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Delete field must not be empty");

empdel\_tf.requestFocus();

}

else

{

try{

s1=(String)empdel\_cb.getSelectedItem();

if(s1.equalsIgnoreCase("Employee Id"))

s1="select \* from emp\_table where upper(emp\_id)=?";

else

s1="select \* from emp\_table where upper(emp\_name)=?";

stmt=con.prepareStatement(s1,ResultSet.TYPE\_SCROLL\_SENSITIVE,ResultSet.CONCUR\_READ\_ONLY);

stmt.setString(1,empdel\_tf.getText().toUpperCase());

rs=stmt.executeQuery();

new Emp\_Del\_form(rs).setVisible(true);

} catch(SQLException ex)

{

JOptionPane.showMessageDialog(this,"Record Not Found!","recod enter",JOptionPane.ERROR\_MESSAGE);

} catch (ClassNotFoundException ex) {

Logger.getLogger(EmployeeSF.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

private void empupdate\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(update\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Update field must not be empty");

update\_tf.requestFocus();

}

else

{

try{

s1=(String)update\_cb.getSelectedItem();

if(s1.equalsIgnoreCase("Employee Id"))

s1="select \* from emp\_table where upper(emp\_id)=?";

else

s1="select \* from emp\_table where upper(emp\_name)=?";

stmt=con.prepareStatement(s1,ResultSet.TYPE\_SCROLL\_SENSITIVE,ResultSet.CONCUR\_READ\_ONLY);

stmt.setString(1,update\_tf.getText().toUpperCase());

rs=stmt.executeQuery();

new Emp\_Update\_form(rs).setVisible(true);

} catch(SQLException ex)

{ JOptionPane.showMessageDialog(this,"Record Not Found!","recod enter",JOptionPane.ERROR\_MESSAGE);} catch (ClassNotFoundException ex) {

Logger.getLogger(EmployeeSF.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

private void update\_clearActionPerformed(java.awt.event.ActionEvent evt) {

update\_tf.setText("");

}

private void tabpaneStateChanged(javax.swing.event.ChangeEvent evt) {

switch(tabpane.getSelectedIndex())

{

case 0: this.getRootPane().setDefaultButton(empsearch\_bt); break;

case 1: this.getRootPane().setDefaultButton(empdel\_bt); break;

case 2: this.getRootPane().setDefaultButton(empupdate\_bt); break;

}

}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

m.empser\_bt.addActionListener(m.es[0]);

m.esf=null;

this.dispose();

}

private void search\_clearActionPerformed(java.awt.event.ActionEvent evt) {

search\_tf.setText("");

}

private void del\_clearActionPerformed(java.awt.event.ActionEvent evt) {

empdel\_tf.setText("");

}

private void search\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE|| Character.isDigit(c) ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();

}

}

private void empdel\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE|| Character.isDigit(c) ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();

}

}

private void update\_tfKeyTyped(java.awt.event.KeyEvent evt) {

char c=evt.getKeyChar();

if(!(Character.isAlphabetic(c)||c==KeyEvent.VK\_BACK\_SPACE|| Character.isDigit(c) ||c==KeyEvent.VK\_DELETE ||c==KeyEvent.VK\_SPACE)){

evt.consume();

}

***Employee Salary***

package bsphcl;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

public class Emp\_Salary extends javax.swing.JFrame {

PreparedStatement stmt;

Connection con;

ResultSet rs;

long sal,hra,bonus,pf,it,tsal;

boolean f;

String empid;

StartForm m;

public Emp\_Salary(StartForm m) throws SQLException, ClassNotFoundException {

initComponents();

this.m=m;

empid\_tf.requestFocus();

con=ConnDb.conLink();

}

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents

private void initComponents() {

jPanel5 = new javax.swing.JPanel();

jPanel6 = new javax.swing.JPanel();

jPanel2 = new javax.swing.JPanel();

jLabel11 = new javax.swing.JLabel();

empstate\_tf = new javax.swing.JTextField();

emptown\_tf = new javax.swing.JTextField();

empstreet\_tf = new javax.swing.JTextField();

jLabel21 = new javax.swing.JLabel();

jSeparator1 = new javax.swing.JSeparator();

emppost\_tf = new javax.swing.JTextField();

jLabel22 = new javax.swing.JLabel();

emppin\_tf = new javax.swing.JTextField();

jLabel2 = new javax.swing.JLabel();

jLabel23 = new javax.swing.JLabel();

jLabel4 = new javax.swing.JLabel();

jLabel5 = new javax.swing.JLabel();

jLabel6 = new javax.swing.JLabel();

empmob\_tf = new javax.swing.JTextField();

jLabel7 = new javax.swing.JLabel();

empid\_tf = new javax.swing.JTextField();

empname\_tf = new javax.swing.JTextField();

empgen\_tf = new javax.swing.JTextField();

empdob\_tf = new javax.swing.JTextField();

empplot\_tf = new javax.swing.JTextField();

jLabel8 = new javax.swing.JLabel();

jLabel9 = new javax.swing.JLabel();

jLabel10 = new javax.swing.JLabel();

jButton4 = new javax.swing.JButton();

jPanel1 = new javax.swing.JPanel();

jLabel12 = new javax.swing.JLabel();

empsalary\_tf = new javax.swing.JTextField();

jLabel14 = new javax.swing.JLabel();

jLabel15 = new javax.swing.JLabel();

emphra\_tf = new javax.swing.JTextField();

jLabel16 = new javax.swing.JLabel();

empbonus\_tf = new javax.swing.JTextField();

jLabel17 = new javax.swing.JLabel();

jLabel18 = new javax.swing.JLabel();

emppf\_tf = new javax.swing.JTextField();

jLabel19 = new javax.swing.JLabel();

empit\_tf = new javax.swing.JTextField();

jLabel20 = new javax.swing.JLabel();

empts\_tf = new javax.swing.JTextField();

cal\_bt = new javax.swing.JButton();

jSeparator3 = new javax.swing.JSeparator();

jLabel13 = new javax.swing.JLabel();

jPanel4 = new javax.swing.JPanel();

new\_bt = new javax.swing.JButton();

pay\_bt = new javax.swing.JButton();

clear\_bt = new javax.swing.JButton();

jPanel3 = new javax.swing.JPanel();

jLabel25 = new javax.swing.JLabel();

jPanel8 = new javax.swing.JPanel();

jLabel3 = new javax.swing.JLabel();

setDefaultCloseOperation(javax.swing.WindowConstants.DO\_NOTHING\_ON\_CLOSE);

setTitle("Employee Salary");

setUndecorated(true);

setResizable(false);

jPanel6.setBackground(new java.awt.Color(119, 196, 182));

jPanel2.setBackground(new java.awt.Color(119, 181, 198));

jPanel2.setBorder(new javax.swing.border.SoftBevelBorder(javax.swing.border.BevelBorder.RAISED));

jLabel11.setText("State");

empstate\_tf.setEditable(false);

emptown\_tf.setEditable(false);

empstreet\_tf.setEditable(false);

jLabel21.setText("Desgination");

emppost\_tf.setEditable(false);

jLabel22.setText("Pincode");

emppin\_tf.setEditable(false);

jLabel2.setText("Employee Id");

jLabel23.setText("Mobile No.");

jLabel4.setText("Employee Name");

jLabel5.setText("Gender");

jLabel6.setText("Date Of Birth");

empmob\_tf.setEditable(false);

jLabel7.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N

jLabel7.setText("Address");

empid\_tf.addFocusListener(new java.awt.event.FocusAdapter() {

public void focusLost(java.awt.event.FocusEvent evt) {

empid\_tfFocusLost(evt);

}

});

empname\_tf.setEditable(false);

empgen\_tf.setEditable(false);

empdob\_tf.setEditable(false);

empplot\_tf.setEditable(false);

jLabel8.setText("Plot No.");

jLabel9.setText("Street Name");

jLabel10.setText("Village/Town");

jButton4.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N

jButton4.setIcon(new javax.swing.ImageIcon("E:\\pro\_ele\\pro\_module\\Project\\src\\Image\\if\_Arrow\_Left\_1063872.png")); // NOI18N

jButton4.setText("Back");

jButton4.setBorder(null);

jButton4.setBorderPainted(false);

jButton4.setContentAreaFilled(false);

jButton4.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton4ActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);

jPanel2.setLayout(jPanel2Layout);

jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addContainerGap(36, Short.MAX\_VALUE)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(jLabel11, javax.swing.GroupLayout.PREFERRED\_SIZE, 67, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(empstate\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 105, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(jLabel8, javax.swing.GroupLayout.PREFERRED\_SIZE, 67, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(empplot\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 105, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(jLabel10, javax.swing.GroupLayout.PREFERRED\_SIZE, 73, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(emptown\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 105, javax.swing.GroupLayout.PREFERRED\_SIZE))))

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGap(127, 127, 127)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(jLabel21, javax.swing.GroupLayout.PREFERRED\_SIZE, 67, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(emppost\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(jLabel5, javax.swing.GroupLayout.PREFERRED\_SIZE, 67, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(empgen\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 105, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addGap(44, 44, 44))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel2Layout.createSequentialGroup()

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jLabel22, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jLabel9, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jLabel23, javax.swing.GroupLayout.PREFERRED\_SIZE, 67, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(27, 27, 27)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(empmob\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(empstreet\_tf, javax.swing.GroupLayout.DEFAULT\_SIZE, 105, Short.MAX\_VALUE)

.addComponent(emppin\_tf))

.addGap(45, 45, 45))))

.addGroup(jPanel2Layout.createSequentialGroup()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED\_SIZE, 78, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(26, 26, 26)

.addComponent(empid\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 105, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(jLabel4, javax.swing.GroupLayout.PREFERRED\_SIZE, 94, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(empname\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 105, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(jLabel6, javax.swing.GroupLayout.PREFERRED\_SIZE, 86, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addComponent(empdob\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 105, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addComponent(jLabel7, javax.swing.GroupLayout.PREFERRED\_SIZE, 67, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jSeparator1, javax.swing.GroupLayout.PREFERRED\_SIZE, 509, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addContainerGap())))

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(jButton4)

.addGap(0, 0, Short.MAX\_VALUE))

);

jPanel2Layout.linkSize(javax.swing.SwingConstants.HORIZONTAL, new java.awt.Component[] {empdob\_tf, empgen\_tf, empid\_tf, empmob\_tf, empname\_tf, emppin\_tf, empplot\_tf, emppost\_tf, empstate\_tf, empstreet\_tf, emptown\_tf});

jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(jButton4, javax.swing.GroupLayout.PREFERRED\_SIZE, 25, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(empid\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 29, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel2)

.addComponent(jLabel21)

.addComponent(emppost\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(41, 41, 41)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(empname\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 29, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel4)

.addComponent(empgen\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 29, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel5))

.addGap(40, 40, 40)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(empdob\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 29, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel6))

.addGap(18, 18, 18)

.addComponent(jLabel7)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jSeparator1, javax.swing.GroupLayout.PREFERRED\_SIZE, 10, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(16, 16, 16)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(empplot\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 29, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel8)

.addComponent(jLabel9)

.addComponent(empstreet\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 29, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(37, 37, 37)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel10)

.addComponent(emptown\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 29, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel22)

.addComponent(emppin\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(39, 39, 39)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(empstate\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 29, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel11)

.addComponent(jLabel23)

.addComponent(empmob\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

jPanel2Layout.linkSize(javax.swing.SwingConstants.VERTICAL, new java.awt.Component[] {empdob\_tf, empgen\_tf, empid\_tf, empmob\_tf, empname\_tf, emppin\_tf, empplot\_tf, emppost\_tf, empstate\_tf, empstreet\_tf, emptown\_tf});

jPanel1.setBackground(new java.awt.Color(107, 140, 129));

jPanel1.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED));

jLabel12.setText("Basic Salary");

empsalary\_tf.setEditable(false);

jLabel14.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N

jLabel14.setForeground(new java.awt.Color(0, 0, 153));

jLabel14.setText("Add");

jLabel15.setText("HRA 10%");

emphra\_tf.setEditable(false);

jLabel16.setText("BONUS 11%");

empbonus\_tf.setEditable(false);

jLabel17.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N

jLabel17.setForeground(new java.awt.Color(0, 0, 153));

jLabel17.setText("Less");

jLabel18.setText("Provident Fund 12%");

emppf\_tf.setEditable(false);

jLabel19.setText("Income Tax 9%");

empit\_tf.setEditable(false);

jLabel20.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N

jLabel20.setText("Total Salary");

empts\_tf.setEditable(false);

cal\_bt.setForeground(new java.awt.Color(0, 102, 255));

cal\_bt.setIcon(new javax.swing.ImageIcon("E:\\pro\_ele\\pro\_module\\Project\\src\\Image\\if\_4\_2431401.png")); // NOI18N

cal\_bt.setText("CALCULATE");

cal\_bt.setEnabled(false);

cal\_bt.setOpaque(false);

cal\_bt.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

cal\_btMouseClicked(evt);

}

});

cal\_bt.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

cal\_btActionPerformed(evt);

}

});

jLabel13.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N

jLabel13.setForeground(new java.awt.Color(204, 0, 0));

jLabel13.setText("Salary Details");

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addComponent(jLabel14, javax.swing.GroupLayout.PREFERRED\_SIZE, 39, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(34, 34, 34)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel16)

.addComponent(jLabel15)))

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addComponent(jLabel17, javax.swing.GroupLayout.PREFERRED\_SIZE, 30, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(19, 19, 19)

.addComponent(jLabel12, javax.swing.GroupLayout.PREFERRED\_SIZE, 89, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(35, 35, 35)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel19, javax.swing.GroupLayout.PREFERRED\_SIZE, 99, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel18, javax.swing.GroupLayout.PREFERRED\_SIZE, 110, javax.swing.GroupLayout.PREFERRED\_SIZE))))

.addGap(87, 87, 87)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(empsalary\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 105, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(empbonus\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 105, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(emphra\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 105, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(emppf\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 105, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(empit\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 105, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(0, 27, Short.MAX\_VALUE))

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addComponent(jSeparator3)))

.addContainerGap())

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()

.addGap(37, 37, 37)

.addComponent(jLabel20, javax.swing.GroupLayout.PREFERRED\_SIZE, 76, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(empts\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 99, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(65, 65, 65))

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addComponent(jLabel13)

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addComponent(cal\_bt, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addContainerGap())

);

jPanel1Layout.linkSize(javax.swing.SwingConstants.HORIZONTAL, new java.awt.Component[] {empbonus\_tf, emphra\_tf, empit\_tf, emppf\_tf, empsalary\_tf, empts\_tf});

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addComponent(jLabel13)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel12)

.addComponent(empsalary\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 29, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jLabel14)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(emphra\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 25, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel15))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(empbonus\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel16))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jLabel17)

.addGap(17, 17, 17)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(emppf\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 28, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel18))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel19)

.addComponent(empit\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 28, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jSeparator3, javax.swing.GroupLayout.PREFERRED\_SIZE, 10, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(cal\_bt)

.addGap(18, 18, 18)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(empts\_tf, javax.swing.GroupLayout.PREFERRED\_SIZE, 28, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel20))

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

jPanel1Layout.linkSize(javax.swing.SwingConstants.VERTICAL, new java.awt.Component[] {empbonus\_tf, emphra\_tf, empit\_tf, emppf\_tf, empsalary\_tf, empts\_tf});

jPanel4.setBorder(new javax.swing.border.SoftBevelBorder(javax.swing.border.BevelBorder.RAISED));

new\_bt.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N

new\_bt.setForeground(new java.awt.Color(0, 102, 204));

new\_bt.setIcon(new javax.swing.ImageIcon("E:\\pro\_ele\\pro\_module\\Project\\src\\Image\\if\_add\_134224.png")); // NOI18N

new\_bt.setText("New");

new\_bt.setEnabled(false);

new\_bt.setOpaque(false);

new\_bt.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

new\_btActionPerformed(evt);

}

});

pay\_bt.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N

pay\_bt.setForeground(new java.awt.Color(0, 102, 102));

pay\_bt.setIcon(new javax.swing.ImageIcon("E:\\pro\_ele\\pro\_module\\Project\\src\\Image\\if\_Mastercard\_345560.png")); // NOI18N

pay\_bt.setText("Pay");

pay\_bt.setEnabled(false);

pay\_bt.setOpaque(false);

pay\_bt.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

pay\_btActionPerformed(evt);

}

});

clear\_bt.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N

clear\_bt.setForeground(new java.awt.Color(153, 0, 0));

clear\_bt.setIcon(new javax.swing.ImageIcon("E:\\pro\_ele\\pro\_module\\Project\\src\\Image\\if\_edit-clear\_118917.png")); // NOI18N

clear\_bt.setText("Clear");

clear\_bt.setOpaque(false);

clear\_bt.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

clear\_btActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel4Layout = new javax.swing.GroupLayout(jPanel4);

jPanel4.setLayout(jPanel4Layout);

jPanel4Layout.setHorizontalGroup(

jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel4Layout.createSequentialGroup()

.addGap(28, 28, 28)

.addComponent(new\_bt)

.addGap(29, 29, 29)

.addComponent(pay\_bt)

.addGap(18, 21, Short.MAX\_VALUE)

.addComponent(clear\_bt)

.addGap(29, 29, 29))

);

jPanel4Layout.setVerticalGroup(

jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel4Layout.createSequentialGroup()

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(new\_bt)

.addComponent(pay\_bt, javax.swing.GroupLayout.PREFERRED\_SIZE, 24, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(clear\_bt, javax.swing.GroupLayout.PREFERRED\_SIZE, 23, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

jLabel25.setIcon(new javax.swing.ImageIcon("E:\\pro\_ele\\pro\_module\\Project\\src\\Image\\Capture7.JPG")); // NOI18N

javax.swing.GroupLayout jPanel3Layout = new javax.swing.GroupLayout(jPanel3);

jPanel3.setLayout(jPanel3Layout);

jPanel3Layout.setHorizontalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel25, javax.swing.GroupLayout.PREFERRED\_SIZE, 1093, javax.swing.GroupLayout.PREFERRED\_SIZE)

);

jPanel3Layout.setVerticalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel25, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

jPanel8.setBackground(new java.awt.Color(173, 150, 86));

jLabel3.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N

jLabel3.setText(" Employee Salary");

javax.swing.GroupLayout jPanel8Layout = new javax.swing.GroupLayout(jPanel8);

jPanel8.setLayout(jPanel8Layout);

jPanel8Layout.setHorizontalGroup(

jPanel8Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel8Layout.createSequentialGroup()

.addComponent(jLabel3)

.addGap(0, 0, Short.MAX\_VALUE))

);

jPanel8Layout.setVerticalGroup(

jPanel8Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT\_SIZE, 26, Short.MAX\_VALUE)

);

javax.swing.GroupLayout jPanel6Layout = new javax.swing.GroupLayout(jPanel6);

jPanel6.setLayout(jPanel6Layout);

jPanel6Layout.setHorizontalGroup(

jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel6Layout.createSequentialGroup()

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel6Layout.createSequentialGroup()

.addGap(43, 43, 43)

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(51, 51, 51)

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jPanel4, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addGroup(jPanel6Layout.createSequentialGroup()

.addGap(16, 16, 16)

.addComponent(jPanel3, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addContainerGap(17, Short.MAX\_VALUE))

.addComponent(jPanel8, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

jPanel6Layout.setVerticalGroup(

jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel6Layout.createSequentialGroup()

.addComponent(jPanel8, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(jPanel3, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(33, 33, 33)

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel6Layout.createSequentialGroup()

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(31, 31, 31)

.addComponent(jPanel4, javax.swing.GroupLayout.PREFERRED\_SIZE, 32, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addContainerGap(14, Short.MAX\_VALUE))

);

javax.swing.GroupLayout jPanel5Layout = new javax.swing.GroupLayout(jPanel5);

jPanel5.setLayout(jPanel5Layout);

jPanel5Layout.setHorizontalGroup(

jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel6, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

jPanel5Layout.setVerticalGroup(

jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel6, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(jPanel5, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(0, 0, Short.MAX\_VALUE))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel5, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

);

pack();

setLocationRelativeTo(null);

}// </editor-fold>//GEN-END:initComponents

private void pay\_btActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_pay\_btActionPerformed

if(empid\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Employee id must not be empty","Entry problem",JOptionPane.ERROR\_MESSAGE);

empid\_tf.requestFocus();

}

else

{

try {

stmt=con.prepareStatement("insert into salary\_payment values(?,?,?,?,?,?)");

stmt.setString(1,empid\_tf.getText());

stmt.setLong(2,Long.parseLong(emphra\_tf.getText()));

stmt.setLong(3,Long.parseLong(empbonus\_tf.getText()));

stmt.setLong(4,Long.parseLong(emppf\_tf.getText()));

stmt.setLong(5,Long.parseLong(empit\_tf.getText()));

stmt.setLong(6,Long.parseLong(empts\_tf.getText()));

stmt.executeUpdate();

con.close();

JOptionPane.showMessageDialog(this,"Payment sucessful of Employee: "+empid);

pay\_bt.setEnabled(false);

cal\_bt.setEnabled(false);

new\_bt.setEnabled(true);

}

catch (SQLException ex) {

JOptionPane.showMessageDialog(this,"Payment not sucessful of Employee "+empid );

}

}

}//GEN-LAST:event\_pay\_btActionPerformed

private void new\_btActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_new\_btActionPerformed

try {

clear();

new\_bt.setEnabled(false);

con=ConnDb.conLink();

} catch (SQLException | ClassNotFoundException ex) {

Logger.getLogger(Emp\_Salary.class.getName()).log(Level.SEVERE, null, ex);

}

}//GEN-LAST:event\_new\_btActionPerformed

private void empid\_tfFocusLost(java.awt.event.FocusEvent evt) {//GEN-FIRST:event\_empid\_tfFocusLost

if(empid\_tf.getText().length()==0)

empid\_tf.requestFocus();

else

{

try {

stmt=con.prepareStatement("Select emp\_name,emp\_street,emp\_plot,emp\_city,emp\_pincode,emp\_state,emp\_mob,emp\_sal,emp\_dob,emp\_gen,desg,emp\_basic\_sal from emp\_table where upper(emp\_id)=?");

stmt.setString(1,empid\_tf.getText().toUpperCase());

rs=stmt.executeQuery();

rs.next();

empsalary\_tf.setText(""+rs.getString(12));

empname\_tf.setText(rs.getString(8)+" "+rs.getString(1));

empgen\_tf.setText(rs.getString(10));

empdob\_tf.setText(rs.getString(9));

emppost\_tf.setText(rs.getString(11));

empplot\_tf.setText(rs.getString(3));

empstreet\_tf.setText(rs.getString(2));

emptown\_tf.setText(rs.getString(4));

emppin\_tf.setText(""+rs.getString(5));

empmob\_tf.setText(""+rs.getString(7));

empstate\_tf.setText(rs.getString(6));

sal=Long.parseLong(empsalary\_tf.getText());

hra=(sal\*10)/100;

bonus=(sal\*11)/100;

pf=(sal\*12)/100;

it=(sal\*9)/100;

emphra\_tf.setText(""+hra);

empbonus\_tf.setText(""+bonus);

emppf\_tf.setText(""+pf);

empit\_tf.setText(""+it);

empid=empid\_tf.getText();

cal\_bt.setEnabled(true);

}

catch (SQLException ex)

{

JOptionPane.showMessageDialog(this,"Employee ID not found");

empid\_tf.setText("");

empid\_tf.requestFocus();

}

}

}//GEN-LAST:event\_empid\_tfFocusLost

private void cal\_btActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_cal\_btActionPerformed

tsal=(sal+hra+bonus)-(pf+it);

empts\_tf.setText(""+tsal);

}//GEN-LAST:event\_cal\_btActionPerformed

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton4ActionPerformed

m.empsal\_bt.addActionListener(m.esal[0]);

m.emp\_sal=null;

this.dispose();

}//GEN-LAST:event\_jButton4ActionPerformed

private void cal\_btMouseClicked(java.awt.event.MouseEvent evt) {//GEN-FIRST:event\_cal\_btMouseClicked

pay\_bt.setEnabled(true);

}//GEN-LAST:event\_cal\_btMouseClicked

private void clear()

{

empid\_tf.setText("");

empname\_tf.setText("");

empgen\_tf.setText("");

empdob\_tf.setText("");

emppost\_tf.setText("");

empplot\_tf.setText("");

empstreet\_tf.setText("");

emptown\_tf.setText("");

emppin\_tf.setText("");

empmob\_tf.setText("");

empstate\_tf.setText("");

emphra\_tf.setText("");

empbonus\_tf.setText("");

emppf\_tf.setText("");

empit\_tf.setText("");

empts\_tf.setText("");

empsalary\_tf.setText("");

}

private void clear\_btActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_clear\_btActionPerformed

clear();

}//GEN-LAST:event\_clear\_btActionPerformed

// Variables declaration - do not modify//GEN-BEGIN:variables

private javax.swing.JButton cal\_bt;

private javax.swing.JButton clear\_bt;

private javax.swing.JTextField empbonus\_tf;

private javax.swing.JTextField empdob\_tf;

private javax.swing.JTextField empgen\_tf;

private javax.swing.JTextField emphra\_tf;

private javax.swing.JTextField empid\_tf;

private javax.swing.JTextField empit\_tf;

private javax.swing.JTextField empmob\_tf;

private javax.swing.JTextField empname\_tf;

private javax.swing.JTextField emppf\_tf;

private javax.swing.JTextField emppin\_tf;

private javax.swing.JTextField empplot\_tf;

private javax.swing.JTextField emppost\_tf;

private javax.swing.JTextField empsalary\_tf;

private javax.swing.JTextField empstate\_tf;

private javax.swing.JTextField empstreet\_tf;

private javax.swing.JTextField emptown\_tf;

private javax.swing.JTextField empts\_tf;

private javax.swing.JButton jButton4;

private javax.swing.JLabel jLabel10;

private javax.swing.JLabel jLabel11;

private javax.swing.JLabel jLabel12;

private javax.swing.JLabel jLabel13;

private javax.swing.JLabel jLabel14;

private javax.swing.JLabel jLabel15;

private javax.swing.JLabel jLabel16;

private javax.swing.JLabel jLabel17;

private javax.swing.JLabel jLabel18;

private javax.swing.JLabel jLabel19;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel20;

private javax.swing.JLabel jLabel21;

private javax.swing.JLabel jLabel22;

private javax.swing.JLabel jLabel23;

private javax.swing.JLabel jLabel25;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JLabel jLabel6;

private javax.swing.JLabel jLabel7;

private javax.swing.JLabel jLabel8;

private javax.swing.JLabel jLabel9;

private javax.swing.JPanel jPanel1;

private javax.swing.JPanel jPanel2;

private javax.swing.JPanel jPanel3;

private javax.swing.JPanel jPanel4;

private javax.swing.JPanel jPanel5;

private javax.swing.JPanel jPanel6;

private javax.swing.JPanel jPanel8;

private javax.swing.JSeparator jSeparator1;

private javax.swing.JSeparator jSeparator3;

private javax.swing.JButton new\_bt;

private javax.swing.JButton pay\_bt;

// End of variables declaration//GEN-END:variables

}

***Forget Password***

package Module;

import java.awt.Image;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.ImageIcon;

import javax.swing.JOptionPane;

public class UserForgetPwd extends javax.swing.JFrame {

PreparedStatement stmt;

Connection con;

ResultSet rs;

boolean flag=false;

ImageIcon img;

public UserForgetPwd() throws SQLException, ClassNotFoundException {

initComponents();

con=ConnDb.conLink();

img=new ImageIcon(new ImageIcon(getClass().getResource("eye3.png")).getImage().getScaledInstance(eye\_bt.getWidth(),eye\_bt.getHeight(),Image.SCALE\_SMOOTH));

eye\_bt.setIcon(img);

}

private void confirm\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(new\_tf.getText().length()==0)

{

new\_tf.requestFocus();

JOptionPane.showMessageDialog(this,"new password cannot empty!");

}

else if( confirm\_tf.getText().length()==0)

{

confirm\_tf.requestFocus();

JOptionPane.showMessageDialog(this,"confirm password cannot empty!");

}

else if(!(new\_tf.getText().equals(confirm\_tf.getText())))

{

new\_tf.requestFocus();

JOptionPane.showMessageDialog(this,"New password and confirm password must be same");

new\_tf.setText("");

confirm\_tf.setText("");

}

else

{

try {

stmt=con.prepareStatement("update log\_tab set user\_pwd=? where upper(usr\_id)='"+user\_tf.getText().toUpperCase()+"'");

stmt.setString(1,confirm\_tf.getText());

stmt.executeUpdate();

try {

JOptionPane.showMessageDialog(this,"Password successfully changed");

new LoginForm1().setVisible(true);

con.close();

this.dispose();

} catch (ClassNotFoundException ex) {

Logger.getLogger(UserForgetPwd.class.getName()).log(Level.SEVERE, null, ex);

}

} catch (SQLException ex) {

JOptionPane.showMessageDialog(this,"Password successfully not changed");

}

}

}

private void close\_btActionPerformed(java.awt.event.ActionEvent evt) {

try {

new LoginForm1().setVisible(true);

this.dispose();

} catch (SQLException | ClassNotFoundException ex) {

Logger.getLogger(UserForgetPwd.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void cur\_tfFocusLost(java.awt.event.FocusEvent evt) {

try {

stmt=con.prepareStatement("Select usr\_id,user\_pwd from log\_tab where upper(usr\_id)='"+user\_tf.getText().toUpperCase()+"'");

rs=stmt.executeQuery();

if(rs.next())

{

if(rs.getString(1).equalsIgnoreCase(user\_tf.getText()))

{

if(rs.getString(2).equalsIgnoreCase(cur\_tf.getText()))

{

ans\_tf.requestFocus();

}

else

{

JOptionPane.showMessageDialog(this,"Invalid Password ");

cur\_tf.setText("");

cur\_tf.requestFocus();

}

}

else

{

cur\_tf.setText("");

user\_tf.setText("");

user\_tf.requestFocus();

JOptionPane.showMessageDialog(this,"Invalid User ID");

}

}

else

{

JOptionPane.showMessageDialog(this,"Not available this User ID!");

user\_tf.setText("");

user\_tf.requestFocus();

cur\_tf.setText("");

}

} catch (SQLException ex) {

Logger.getLogger(UserForgetPwd.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void ans\_tfFocusLost(java.awt.event.FocusEvent evt) {

try {

stmt=con.prepareStatement("Select usr\_id,user\_ans from log\_tab where upper(usr\_id)='"+user\_tf.getText().toUpperCase()+"'");

rs=stmt.executeQuery();

if(rs.next())

{

if(rs.getString(1).equals(user\_tf.getText()))

{

if(rs.getString(2).equalsIgnoreCase(ans\_tf.getText()))

{

new\_tf.requestFocus();

new\_tf.setEnabled(true);

confirm\_tf.setEnabled(true);

}

else

{

ans\_tf.setText("");

ans\_tf.requestFocus();

JOptionPane.showMessageDialog(this,"Invalid Answer!! ");

}

}

else

{

ans\_tf.setText("");

user\_tf.setText("");

user\_tf.requestFocus();

JOptionPane.showMessageDialog(this,"Invalid User ID");

}

} else

{

JOptionPane.showMessageDialog(this,"Not available this User ID!");

user\_tf.setText("");

ans\_tf.setText("");

user\_tf.requestFocus();

}

} catch (SQLException ex) {

Logger.getLogger(UserForgetPwd.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void eye\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(flag){

img=new ImageIcon(new ImageIcon(getClass().getResource("eye3.png")).getImage().getScaledInstance(eye\_bt.getWidth(),eye\_bt.getHeight(),Image.SCALE\_SMOOTH));

eye\_bt.setIcon(img);

flag=false;

new\_tf.setEchoChar('\*');

}

else{

img=new ImageIcon(new ImageIcon(getClass().getResource("e1.png")).getImage().getScaledInstance(eye\_bt.getWidth(),eye\_bt.getHeight(),Image.SCALE\_SMOOTH));

eye\_bt.setIcon(img);

flag=true;

new\_tf.setEchoChar((char)0);

}

}

***Login Window***

package Module;

import java.awt.Image;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.ButtonGroup;

import javax.swing.ImageIcon;

import javax.swing.JOptionPane;

public class LoginForm1 extends javax.swing.JFrame {

ResultSet rs;

Connection con;

PreparedStatement stmt;

ImageIcon img;

ButtonGroup bg;

boolean flag=false;

public LoginForm1() throws SQLException, ClassNotFoundException {

initComponents();

bg=new ButtonGroup();

bg.add(rb1);

bg.add(rb2);

con=ConnDb.conLink();

ques\_cb.setVisible(false);

ans\_tf.setVisible(false);

this.getRootPane().setDefaultButton(login\_bt);

img=new ImageIcon(new ImageIcon(getClass().getResource("eye3.png")).getImage().getScaledInstance(eye\_bt.getWidth(),eye\_bt.getHeight(),Image.SCALE\_SMOOTH));

eye\_bt.setIcon(img);

}

private void login\_btActionPerformed(java.awt.event.ActionEvent evt) {

try

{

if(rb1.isSelected()){

if(rs.getString(3).equalsIgnoreCase("null"))

{

if(ans\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(null,"Question value must not be empty!");

ans\_tf.requestFocus();

}

else

{

stmt=con.prepareStatement("update administrator set admin\_ans=?");

stmt.setString(1,ans\_tf.getText());

stmt.executeUpdate();

try {

new AdminWindow().setVisible(true);

this.dispose();

} catch (ClassNotFoundException ex) {

Logger.getLogger(LoginForm1.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

else{

if(rs.getString(1).equalsIgnoreCase(user\_tf.getText()))

if(rs.getString(2).equalsIgnoreCase(pwd\_tf.getText()))

{

try {

new AdminWindow().setVisible(true);

this.dispose();

} catch (ClassNotFoundException ex) {

Logger.getLogger(LoginForm1.class.getName()).log(Level.SEVERE, null, ex);

}

}

else

{

pwd\_tf.setText("");

pwd\_tf.requestFocus();

JOptionPane.showMessageDialog(this,"Invalid Password ");

}

else

{

user\_tf.setText("");

user\_tf.requestFocus();

JOptionPane.showMessageDialog(this,"Invalid User name");

}

}

}

else {

stmt=con.prepareStatement("Select user\_pwd from log\_tab where upper(usr\_id)='"+user\_tf.getText().toUpperCase()+"'");

rs=stmt.executeQuery();

if(rs.next())

{

if(rs.getString(1).equals(pwd\_tf.getText()))

{

try {

new StartForm(1).setVisible(true);

this.dispose();

} catch (ClassNotFoundException ex) {

Logger.getLogger(LoginForm1.class.getName()).log(Level.SEVERE, null, ex);

}

}

else

{

pwd\_tf.setText("");

pwd\_tf.requestFocus();

JOptionPane.showMessageDialog(this,"Password wrong");

}

}

else

JOptionPane.showMessageDialog(this, "User Id Incorrect","Problems", JOptionPane.ERROR\_MESSAGE);

}

}

catch(SQLException ex)

{

JOptionPane.showMessageDialog(this,"Select Login type","Problmes", JOptionPane.ERROR\_MESSAGE);

}

}

private void forget\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(bg.getSelection()==null){

JOptionPane.showMessageDialog(this,"Select the Login type");

}

else if(rb2.isSelected())

{

try {

new UserForgetPwd().setVisible(true);

this.dispose();

} catch (SQLException | ClassNotFoundException ex) {

Logger.getLogger(LoginForm1.class.getName()).log(Level.SEVERE, null, ex);

}

}

else

{

try {

new AdminForgetPwd().setVisible(true);

this.dispose();

} catch (SQLException | ClassNotFoundException ex) {

Logger.getLogger(LoginForm1.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

private void close\_btActionPerformed(java.awt.event.ActionEvent evt) {

System.exit(0);

}

private void eye\_btMouseClicked(java.awt.event.MouseEvent evt) {

if(flag){

img=new ImageIcon(new ImageIcon(getClass().getResource("eye3.png")).getImage().getScaledInstance(eye\_bt.getWidth(),eye\_bt.getHeight(),Image.SCALE\_SMOOTH));

eye\_bt.setIcon(img);

flag=false;

pwd\_tf.setEchoChar('\*');

}

else{

img=new ImageIcon(new ImageIcon(getClass().getResource("e1.png")).getImage().getScaledInstance(eye\_bt.getWidth(),eye\_bt.getHeight(),Image.SCALE\_SMOOTH));

eye\_bt.setIcon(img);

flag=true;

pwd\_tf.setEchoChar((char)0);

}

}

private void rb1ActionPerformed(java.awt.event.ActionEvent evt) {

try {

stmt=con.prepareStatement("select \* from administrator");

rs=stmt.executeQuery();

if(rs.next())

if(rs.getString(3).equalsIgnoreCase("null")){

ques\_cb.setVisible(true);

ans\_tf.setVisible(true);

}

} catch (SQLException ex) {

Logger.getLogger(LoginForm1.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void rb2ActionPerformed(java.awt.event.ActionEvent evt) {

ques\_cb.setVisible(false);

ans\_tf.setVisible(false);

}

private void ques\_cbActionPerformed(java.awt.event.ActionEvent evt) {

if(ques\_cb.getSelectedIndex()==-1)

JOptionPane.showMessageDialog(this,"Select the question");

}

***Payment Form***

package Module;

import java.awt.Component;

import java.awt.Dimension;

import java.awt.Graphics;

import java.awt.Graphics2D;

import java.awt.print.PageFormat;

import java.awt.print.Printable;

import java.awt.print.PrinterException;

import java.awt.print.PrinterJob;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

public class PaymentForm extends javax.swing.JFrame {

PreparedStatement stmt,stmt1;

Connection con;

int pno,dues,bno;

String pnostr;

ResultSet rs,rs1;

String bnostr;

StartForm m;

public PaymentForm(StartForm m) throws SQLException, ClassNotFoundException {

con=ConnDb.conLink();

initComponents();

this.m=m;

stmt=con.prepareStatement("Select sysdate from dual");

rs=stmt.executeQuery();

rs.next();

pdate\_pay.setText(""+rs.getDate(1));

stmt=con.prepareStatement("select apay\_no from automatic");

rs=stmt.executeQuery();

rs.next();

bno=rs.getInt(1);

bno++;

auto();

}

private void auto()

{

if(bno<10)

bnostr="P000"+bno;

else if(bno<100)

bnostr="P00"+bno;

else if(bno<1000)

bnostr="P0"+bno;

else if(bno<10000)

bnostr="P"+bno;

else

JOptionPane.showMessageDialog(this,"Bill id exceed its value");

payid\_tf.setText(bnostr);

}

private void pay\_btActionPerformed(java.awt.event.ActionEvent evt) {

try {

stmt=con.prepareStatement("Update new\_conn set due\_amount=?,paid\_amount=?,paid\_date=? where upper(con\_id)='"+conid\_pay.getText()+"'");

stmt.setInt(1, Integer.parseInt(dueam\_pay.getText()));

stmt.setInt(2,Integer.parseInt(paidamou\_pay.getText()));

stmt.setString(3,pdate\_pay.getText());

stmt.executeUpdate();

stmt=con.prepareStatement("Update bill\_report set net\_payable=? where upper(bill\_no)='"+billno\_pay.getText()+"'");

stmt.setInt(1, Integer.parseInt(dueam\_pay.getText()));

stmt.executeUpdate();

JOptionPane.showMessageDialog(this,"Payment sucessfully completed" );

pay\_bt.setEnabled(false);

new\_bt.setEnabled(true);

int r=JOptionPane.showConfirmDialog(null,"DO U WANT TO PRINT THE RECORD","Confirmation Window",JOptionPane.YES\_NO\_OPTION);

if(r==0)

{

pay\_bt.setVisible(false);

new\_bt.setVisible(false);

jButton2.setVisible(false);

PrinterJob pjob = PrinterJob.getPrinterJob();

PageFormat preformat = pjob.defaultPage();

preformat.setOrientation(PageFormat.PORTRAIT);

PageFormat postformat = pjob.pageDialog(preformat);

//If user does not hit cancel then print.

if (preformat != postformat) {

//Set print component

pjob.setPrintable(new Printer(this), postformat);

if (pjob.printDialog()) {

pjob.print();

pay\_bt.setVisible(true);

new\_bt.setVisible(true);

jButton2.setVisible(true);

}

}

}

}catch (SQLException ex) {

JOptionPane.showMessageDialog(this,"Payment sucessfully not completed" );

} catch (PrinterException ex) {

Logger.getLogger(PaymentForm.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void paidamou\_payFocusLost(java.awt.event.FocusEvent evt) {

try

{

int net=Integer.parseInt(netpay\_tf.getText());

int paid=Integer.parseInt(paidamou\_pay.getText());

if(net>=paid)

{

dues=net-paid;

dueam\_pay.setText(""+dues);

pay\_bt.setEnabled(true);

}

else

JOptionPane.showMessageDialog(this,"Not greater than Netpayable Amount");}

catch (NumberFormatException e) {

JOptionPane.showMessageDialog(this,"Paid amount not be empty","Entry problem",JOptionPane.ERROR\_MESSAGE);}

}

private void conid\_payFocusLost(java.awt.event.FocusEvent evt) {

if(conid\_pay.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Consumer id must not be empty","Entry problem",JOptionPane.ERROR\_MESSAGE);

conid\_pay.requestFocus();

}

else

{

try {

stmt=con.prepareStatement("Select con\_name,conn\_mob from new\_conn where upper(con\_id)=?");

stmt.setString(1,conid\_pay.getText().toUpperCase());

rs=stmt.executeQuery();

rs.next();

stmt1=con.prepareStatement("Select bill\_no,bill\_month,net\_payable from bill\_report where upper(con\_id)=?",ResultSet.TYPE\_SCROLL\_SENSITIVE,ResultSet.CONCUR\_READ\_ONLY);

stmt1.setString(1,conid\_pay.getText().toUpperCase());

rs1=stmt1.executeQuery();

rs1.last();

conname\_pay.setText(rs.getString(1));

mob\_pay.setText(rs.getString(2));

billno\_pay.setText(rs1.getString(1));

billmon\_pay.setText(rs1.getString(2));

netpay\_tf.setText(rs1.getString(3));

paidamou\_pay.requestFocus();

} catch (SQLException ex) {

JOptionPane.showMessageDialog(this,"Customer ID not found");

conname\_pay.setText("");

conname\_pay.requestFocus();

}

}

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

m.payment\_bt.addActionListener(m.pay[0]);

m.pf=null;

this.dispose();

}

private void new\_btMouseClicked(java.awt.event.MouseEvent evt) {

try

{

conname\_pay.setText("");

mob\_pay.setText("");

billno\_pay.setText("");

billmon\_pay.setText("");

netpay\_tf.setText("");

paidamou\_pay.setText("");

dueam\_pay.setText("");

paidamou\_pay.setText("");

pdate\_pay.setText("");

stmt=con.prepareStatement("Update automatic set apay\_no=?");

stmt.setInt(1,bno);

stmt.executeUpdate();

bno++;

auto();

new\_bt.setEnabled(false);

} catch (SQLException ex) {

Logger.getLogger(PaymentForm.class.getName()).log(Level.SEVERE, null, ex);

}

}

public static class Printer implements Printable {

final Component comp;

public Printer(Component comp){

this.comp = comp;

}

@Override

public int print(Graphics g, PageFormat format, int page\_index)

throws PrinterException {

if (page\_index > 0) {

return Printable.NO\_SUCH\_PAGE;

}

// get the bounds of the component

Dimension dim = comp.getSize();

double cHeight = dim.getHeight();

double cWidth = dim.getWidth();

// get the bounds of the printable area

double pHeight = format.getImageableHeight();

double pWidth = format.getImageableWidth();

double pXStart = format.getImageableX();

double pYStart = format.getImageableY();

double xRatio = pWidth / cWidth;

double yRatio = pHeight / cHeight;

Graphics2D g2 = (Graphics2D) g;

g2.translate(pXStart, pYStart);

g2.scale(xRatio, yRatio);

comp.paint(g2);

return Printable.PAGE\_EXISTS;

}

}

***Start Window***

package Module;

import java.awt.event.ActionListener;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.text.DateFormat;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

public class StartForm extends javax.swing.JFrame {

PreparedStatement stmt;

Connection con;

ResultSet rs;

PaymentForm pf;

AboutUs about;

Bill\_generate\_form bill;

ContactUs cu;

ConnectionForm cf;

Emp\_Report\_Table emp\_report;

Employee emp;

Customer\_Report\_Table cus\_report;

Emp\_Salary emp\_sal;

CustomerSF csf;

EmployeeSF esf;

BillReportSF bsf;

ActionListener []pay,em,cons,mp,au,bg,esal,cs,es,cr,er,bs,cont;

int r;

public StartForm(int r) throws SQLException, ClassNotFoundException {

initComponents();

con=ConnDb.conLink();

this.r=r;

if(r==1)

{

empser\_bt.setEnabled(false);

empsal\_bt.setEnabled(false);

}

Thread clock=new Thread()

{

public void run(){

for(;;){

java.util.Date d=new java.util.Date();

d1.setText(DateFormat.getDateTimeInstance().format(d));

try{

sleep(1000);

}catch(InterruptedException ex)

{

Logger.getLogger(StartForm.class.getName()).log(Level.SEVERE,null,ex);

}

}

}

};

clock.start();

}

private void logout\_btActionPerformed(java.awt.event.ActionEvent evt) {

try {

if(pf!=null)

pf.dispose();

if(about!=null)

about.dispose();

if(bill!=null)

bill.dispose();

if(cu!=null)

cu.dispose();

if(csf!=null)

csf.dispose();

if(cf!=null)

cf.dispose();

if(emp\_report!=null)

emp\_report.dispose();

if(emp!=null)

emp.dispose();

if(cus\_report!=null)

cus\_report.dispose();

if(emp\_sal!=null)

emp\_sal.dispose();

if(esf!=null)

esf.dispose();

if(bsf!=null)

bsf.dispose();

this.dispose();

con.close();

JOptionPane.showMessageDialog(this,"Logout Sucessful","Logout",JOptionPane.INFORMATION\_MESSAGE);

new LoginForm1().setVisible(true);

} catch (SQLException | NullPointerException |ClassNotFoundException ex) {

Logger.getLogger(StartForm.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void conser\_btActionPerformed(java.awt.event.ActionEvent evt) {

try {

cs= conser\_bt.getActionListeners();

conser\_bt.removeActionListener(cs[0]);

csf=new CustomerSF(this);

csf.setVisible(true);

} catch (SQLException | ClassNotFoundException ex) {

Logger.getLogger(StartForm.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void payment\_btActionPerformed(java.awt.event.ActionEvent evt) {

try{ pay=payment\_bt.getActionListeners();

payment\_bt.removeActionListener(pay[0]);

pf=new PaymentForm(this);

pf.setVisible(true);

} catch (ClassNotFoundException | SQLException ex) {

Logger.getLogger(StartForm.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void empreport\_btActionPerformed(java.awt.event.ActionEvent evt) {

try {

con=ConnDb.conLink();

stmt=con.prepareStatement("Select \* from emp\_table order by emp\_id asc");

rs=stmt.executeQuery();

emp\_report=new Emp\_Report\_Table(rs);

emp\_report.setVisible(true);

} catch (SQLException |ClassNotFoundException ex) {

Logger.getLogger(BillReportSF.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void newemp\_btActionPerformed(java.awt.event.ActionEvent evt) {

try {

em=newemp\_bt.getActionListeners();

newemp\_bt.removeActionListener(em[0]);

emp=new Employee(this);

emp.setVisible(true);

} catch (SQLException | ClassNotFoundException ex) {

Logger.getLogger(StartForm.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void con\_btActionPerformed(java.awt.event.ActionEvent evt) {

try {

cons=con\_bt.getActionListeners();

con\_bt.removeActionListener(cons[0]);

cf=new ConnectionForm(this);

cf.setVisible(true);

} catch (SQLException | ClassNotFoundException ex) {

Logger.getLogger(StartForm.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void contact\_btActionPerformed(java.awt.event.ActionEvent evt) {

try {

cont=contact\_bt.getActionListeners();

contact\_bt.removeActionListener(cont[0]);

cu= new ContactUs(this);

cu.setVisible(true);

} catch (SQLException | ClassNotFoundException ex) {

Logger.getLogger(StartForm.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void aboutus\_btActionPerformed(java.awt.event.ActionEvent evt) {

try{

au=aboutus\_bt.getActionListeners();

aboutus\_bt.removeActionListener(au[0]);

about=new AboutUs(this);

about.setVisible(true);

}catch(Exception e)

{}

}

private void bill\_btActionPerformed(java.awt.event.ActionEvent evt) {

try {

bg=bill\_bt.getActionListeners();

bill\_bt.removeActionListener(bg[0]);

bill= new Bill\_generate\_form(this);

bill.setVisible(true);

} catch (SQLException | ClassNotFoundException ex) {

Logger.getLogger(StartForm.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void conreport\_btActionPerformed(java.awt.event.ActionEvent evt) {

try {

con=ConnDb.conLink();

stmt=con.prepareStatement("Select con\_id,con\_name,conn\_type,con\_dob,con\_gen,con\_father,conn\_plot,conn\_street,conn\_city,conn\_pin,conn\_state,conn\_mob,con\_email from new\_conn order by con\_id asc");

rs=stmt.executeQuery();

cus\_report=new Customer\_Report\_Table(rs);

cus\_report.setVisible(true);

} catch (SQLException |ClassNotFoundException ex) {

Logger.getLogger(BillReportSF.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void empsal\_btActionPerformed(java.awt.event.ActionEvent evt) {

try {

esal=empsal\_bt.getActionListeners();

empsal\_bt.removeActionListener(esal[0]);

emp\_sal=new Emp\_Salary(this);

emp\_sal.setVisible(true);

} catch (SQLException | ClassNotFoundException ex) {

Logger.getLogger(StartForm.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void billservice\_btActionPerformed(java.awt.event.ActionEvent evt) {

try {

bs=billservice\_bt.getActionListeners();

billservice\_bt.removeActionListener(bs[0]);

bsf= new BillReportSF(this);

bsf.setVisible(true);

} catch (SQLException | ClassNotFoundException ex) {

Logger.getLogger(StartForm.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void empser\_btActionPerformed(java.awt.event.ActionEvent evt) {

try {

es=empser\_bt.getActionListeners();

empser\_bt.removeActionListener(es[0]);

esf=new EmployeeSF(this);

esf.setVisible(true);

} catch (SQLException | ClassNotFoundException ex) {

Logger.getLogger(StartForm.class.getName()).log(Level.SEVERE, null, ex);

}

}

***Employee Report***

package Module;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

public class Emp\_Report\_Table extends javax.swing.JFrame {

ResultSet rs;

Connection con;

PreparedStatement stmt;

public Emp\_Report\_Table(ResultSet rs1) throws SQLException, ClassNotFoundException {

initComponents();

con=ConnDb.conLink();

rs=rs1;

this.getRootPane().setDefaultButton(search\_bt);

DefaultTableModel model=(DefaultTableModel)emp\_tab.getModel();

while(rs.next())

model.addRow(new Object[]{rs.getString(1),rs.getString(2)+" "+rs.getString(3),rs.getString(4),rs.getString(13),rs.getString(5),rs.getString(15),rs.getLong(14),rs.getString(9),rs.getString(8),rs.getString(10),rs.getInt(11),rs.getString(12),rs.getLong(6),rs.getString(7)});

emp\_tab.setModel(model);

}

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

this.dispose();

}

private void search\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(search\_tf.getText().length()==0)

{

JOptionPane.showMessageDialog(this,"Enter consumer name");

search\_tf.requestFocus();

}

else

{

try {

stmt=con.prepareStatement("Select \* from emp\_table where upper(emp\_name)=?");

stmt.setString(1,search\_tf.getText().toUpperCase());

rs=stmt.executeQuery();

DefaultTableModel mod=new DefaultTableModel(new String[]{"Employee Id", "Employee Name", "Gender", "DOB", "Nationality", "Designation", " Basic Salary", "Plot No.", "Street", "Village/Town", "Pincode", "State", "Mobile", "Email Id"},0);

while(rs.next())

mod.addRow(new Object[]{rs.getString(1),rs.getString(2)+" "+rs.getString(3),rs.getString(4),rs.getString(13),rs.getString(5),rs.getString(15),rs.getLong(14),rs.getString(9),rs.getString(8),rs.getString(10),rs.getInt(11),rs.getString(12),rs.getLong(6),rs.getString(7)});

emp\_tab.setModel(mod);

} catch (SQLException ex) {

JOptionPane.showMessageDialog(this,"Record Not Found","Record",JOptionPane.ERROR\_MESSAGE);

}

}

}

***New User Creation***

package Module;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

public class UserCreation extends javax.swing.JFrame {

Connection con;

PreparedStatement stmt;

boolean flag=false;

public UserCreation() throws SQLException, ClassNotFoundException {

initComponents();

con=ConnDb.conLink();

this.getRootPane().setDefaultButton(create\_bt);

}

private void close\_btActionPerformed(java.awt.event.ActionEvent evt) {

try {

new AdminWindow().setVisible(true);

this.dispose();

} catch (SQLException | ClassNotFoundException ex) {

Logger.getLogger(UserCreation.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void create\_btActionPerformed(java.awt.event.ActionEvent evt) {

if(uid\_tf.getText().length()==0)

{

uid\_tf.requestFocus();

JOptionPane.showMessageDialog(this,"User Id not be empty");

}

else if(user\_tf.getText().length()==0)

{

user\_tf.requestFocus();

JOptionPane.showMessageDialog(this,"User name not be empty");

}

else if(pwd\_tf.getText().length()==0)

{

pwd\_tf.requestFocus();

JOptionPane.showMessageDialog(this,"Password field not be empty");

}

else if(confirm\_tf.getText().length()==0)

{

confirm\_tf.requestFocus();

JOptionPane.showMessageDialog(this,"Confirm Password not be empty");

}

else if(ques\_cb.getSelectedIndex()==-1)

{

JOptionPane.showMessageDialog(this,"Question should be selected");

}

else if(ans\_tf.getText().length()==0)

{

ans\_tf.requestFocus();

JOptionPane.showMessageDialog(this,"Answer field not be empty");

}

else

{

try {

stmt=con.prepareStatement("insert into log\_tab values(?,?,?,?)");

stmt.setString(1, uid\_tf.getText());

stmt.setString(2, user\_tf.getText());

stmt.setString(4, ans\_tf.getText());

stmt.setString(3, pwd\_tf.getText());

stmt.executeUpdate();

JOptionPane.showMessageDialog(this,"User successfully created");

new AdminWindow().setVisible(true);

this.dispose();

} catch (SQLException ex) {

JOptionPane.showMessageDialog(this,"User sucessfully not created");

} catch (ClassNotFoundException ex) {

Logger.getLogger(UserCreation.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

private void confirm\_tfFocusLost(java.awt.event.FocusEvent evt) {

if(!(pwd\_tf.getText().equals(confirm\_tf.getText())))

{

JOptionPane.showMessageDialog(this,"New password and confirm password must be same");

confirm\_tf.requestFocus();

confirm\_tf.setText("");

create\_bt.setEnabled(false);

}

else

create\_bt.setEnabled(true)

}

***User details***

*package bsphcl;*

*import java.sql.ResultSet;*

*import java.sql.SQLException;*

*import java.util.logging.Level;*

*import java.util.logging.Logger;*

*import javax.swing.table.DefaultTableModel;*

*public class UserDetail extends javax.swing.JFrame {*

*ResultSet rs;*

*int r;*

*public UserDetail(ResultSet rs) throws SQLException {*

*initComponents();*

*this.rs=rs;*

*DefaultTableModel model=(DefaultTableModel)user\_table.getModel();*

*while(rs.next())*

*model.addRow(new Object[]{rs.getString(1),rs.getString(2),rs.getString(3),rs.getString(4)});*

*user\_table.setModel(model);*

*}*

*@SuppressWarnings("unchecked")*

*// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents*

*private void initComponents() {*

*jPanel1 = new javax.swing.JPanel();*

*jPanel2 = new javax.swing.JPanel();*

*jLabel1 = new javax.swing.JLabel();*

*close\_bt = new javax.swing.JButton();*

*jScrollPane1 = new javax.swing.JScrollPane();*

*user\_table = new javax.swing.JTable();*

*setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);*

*setUndecorated(true);*

*jPanel1.setBackground(new java.awt.Color(0, 102, 102));*

*jPanel2.setBackground(new java.awt.Color(145, 101, 99));*

*jLabel1.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N*

*jLabel1.setText("User Details");*

*close\_bt.setIcon(new javax.swing.ImageIcon("E:\\pro\_ele\\pro\_module\\Project\\src\\Image\\if\_f-cross\_256\_282471.png")); // NOI18N*

*close\_bt.setBorder(null);*

*close\_bt.addActionListener(new java.awt.event.ActionListener() {*

*public void actionPerformed(java.awt.event.ActionEvent evt) {*

*close\_btActionPerformed(evt);*

*}*

*});*

*javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);*

*jPanel2.setLayout(jPanel2Layout);*

*jPanel2Layout.setHorizontalGroup(*

*jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGroup(jPanel2Layout.createSequentialGroup()*

*.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 84, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)*

*.addComponent(close\_bt, javax.swing.GroupLayout.PREFERRED\_SIZE, 27, javax.swing.GroupLayout.PREFERRED\_SIZE))*

*);*

*jPanel2Layout.setVerticalGroup(*

*jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)*

*.addComponent(close\_bt, javax.swing.GroupLayout.DEFAULT\_SIZE, 29, Short.MAX\_VALUE)*

*);*

*user\_table.setBorder(javax.swing.BorderFactory.createEtchedBorder());*

*user\_table.setForeground(new java.awt.Color(0, 0, 153));*

*user\_table.setModel(new javax.swing.table.DefaultTableModel(*

*new Object [][] {*

*},*

*new String [] {*

*"User Id", "User Name", "Password", "Ques/Answer"*

*}*

*) {*

*Class[] types = new Class [] {*

*java.lang.String.class, java.lang.String.class, java.lang.String.class, java.lang.String.class*

*};*

*public Class getColumnClass(int columnIndex) {*

*return types [columnIndex];*

*}*

*});*

*jScrollPane1.setViewportView(user\_table);*

*javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);*

*jPanel1.setLayout(jPanel1Layout);*

*jPanel1Layout.setHorizontalGroup(*

*jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT\_SIZE, 532, Short.MAX\_VALUE)*

*.addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)*

*);*

*jPanel1Layout.setVerticalGroup(*

*jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGroup(jPanel1Layout.createSequentialGroup()*

*.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)*

*.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))*

*);*

*javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());*

*getContentPane().setLayout(layout);*

*layout.setHorizontalGroup(*

*layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)*

*);*

*layout.setVerticalGroup(*

*layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)*

*);*

*pack();*

*setLocationRelativeTo(null);*

*}// </editor-fold>//GEN-END:initComponents*

*private void close\_btActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_close\_btActionPerformed*

*try {*

*new AdminWindow().setVisible(true);*

*this.dispose();*

*} catch (SQLException | ClassNotFoundException ex) {*

*Logger.getLogger(UserDetail.class.getName()).log(Level.SEVERE, null, ex);*

*}*

*}//GEN-LAST:event\_close\_btActionPerformed*

*// Variables declaration - do not modify//GEN-BEGIN:variables*

*private javax.swing.JButton close\_bt;*

*private javax.swing.JLabel jLabel1;*

*private javax.swing.JPanel jPanel1;*

*private javax.swing.JPanel jPanel2;*

*private javax.swing.JScrollPane jScrollPane1;*

*private javax.swing.JTable user\_table;*

*// End of variables declaration//GEN-END:variables*

*}*

***Connection code***

package Module;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

public class ConnDb {

public static Connection conLink() throws SQLException, ClassNotFoundException

{

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

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

return(con);

}

}

### IMPLEMENTATION AND MAINTENANCE

The platform is the hardware and software combination that the Client/Server runs on. While hardware systems vary widely in features and capabilities, certain common features are needed for the operating system software.

**Hardware Specifications**

Hardware is a set of physical components, which performs the functions of applying appropriate, predefined instructions. In other words, one can say that electronic and mechanical parts of computer constitute hardware.

This package is designed on a powerful programming language JAVA(Core). It is a powerful Graphical User Interface. The backend is MS SQL Server, which is used to maintain database. It can run on almost all the popular microcomputers.

**Software Requirements**

The software is a set of procedures of coded information or a program which when fed into the computer hardware, enables the computer to perform the various tasks. Software is like a current inside the wire, which cannot be seen but its effect can be felt.

Application software : JDK 1.5, Ms SQL Server 2005.

**Maintenance**

After the installation phase is completed and the user staff is adjusted to the changes created by the candidate system, evaluation and maintenance begin. Like any system, there is an aging process that requires periodic maintenance of hardware and software. If the new information is inconsistent with the design specifications, then changes have to be made. Hardware also requires periodic maintenance to keep in tune with design specifications. The importance of maintenance is to continue to bring the new system to standards.

User priorities, changes in organizational requirements, or environmental factors also call for system enhancements.

### Testing

Testing of a system before implementation is good strategy to ensure error free handling of software and data in future. Necessary care must be taken to avoid run time errors. After completing coding and development part of computer system, it needs to be tested to verify its proper functioning. Testing forms is one of the most important phases of the Complete **SDLC.** The general types of testing are the **system testing** followed by **unit testing**.

Testing a system with raw data is a very complicated task. The raw data is passed to the system as input and the output is verified according to the process logic. If the output is of according to the process logic then the system can be implemented on actual location. If the output is not according to the process, logic than the unit/system is revisited and modified until the optimum output does not arrives.

Here are the few sample run time screens with raw data, which is given there in after successful processing of the System.

### TYPES OF TESTING:

**Functional Testing:**

In functional testing the structure of the program is not considered. Test cases are decided solely on the basis of requirements or specifications of the program or module and the internals of the module or the program are not considered for selection of test cases. Due to its nature, functional testing is often called “black box testing”.

The basis for deciding test cases in functional testing is the requirements or specifications of the system or module. For the entire system, the test cases are designed from the requirement specification document for the system. For modules created during design, test cases for functional testing are decided from the module specifications produced during the design.

**Integral Testing:**

Unit testing ensures that all modules have been tested and each of them works properly individually. Unit testing does not guarantee if these modules will work fine if these are integrated together as a whole system. It is observed that many errors crop up when the modules are joined together. Integration testing uncovers errors that arises when modules are integrated to build the overall system.

Following types of errors may arise:

* Data can be lost across an interface: That is data coming out of a module is not a going into the desired module.
* Sub-functions, when combined, may not produce the desired major function.
* Individually acceptable imprecision may be magnified to unacceptable levels. For example, in module there is error-precision taken as +-10 units. In other module same error-precision is used. Now these modules are combined. Suppose the error-precision from both modules needs to be multiplied then the error precision would be +-100 which would not be acceptable of the system.
* Global data structures can present problems: For example, in a system there is a global memory. Now these modules are combined. All are accessing the same global memory. Because so many functions are accessing that memory, low memory problem arises.

**System Testing:**

Software is only one element of a larger computer-based system. Ultimately, software is incorporated with other system elements and a series of system integration and a validation test are conducted. These tests fall outside the scope of software engineering process and are not conducted solely by the software developer.

System testing is actually a series of different test whose primary purpose is to fully exercise the computer-based system. Although each test has a different purpose, all work to verify that all system elements have been properly integrated and perform allocated functions.

### MODULE

**No. of Modules :-**

The System after careful analysis has been identified to be presented with the following modules: -

1. Authentication and Security Module
2. Employee Module
3. Bill Management Module

**Details of Modules :-**

1. **Authentication and Security Module :-**

The user details should be verified against the details in the user tables and if it is valid user, they should be entered into the system. Once entered based on user type access to the different modules to be enabled / disabled and individuals user can change their default password or old password.

1. **Admin Module :-**

This module consists of the following sub modules:-

1. **User Management Modules:** Here a main admin can access any user related information, for examples their contact details ...etc. and requirements come he can delete a user.
2. **Others:** In this module admin uses services of other different module for efficient management of the application.
3. **Employee Module :-**

In this module user can register and modify. And can take services of other modules.

1. **User Management Module :-**

In this module new user details can be added, modified, and also status can be viewed. In this module user can be pay bill according to his needs like view bill,pay bill etc.

### Future Scope of the Project:

* In a nutshell, it can be summarized that the future scope of the project circles around maintaining information regarding:
* We can add printer in future.
* We can give more advance software for Electricity Bill Management System including more facilities
* We will host the platform on online servers to make it accessible worldwide
* Integrate multiple load balancers to distribute the loads of the system
* Create the master and slave database structure to reduce the overload of the database queries
* Implement the backup mechanism for taking backup of codebase and database on regular basis on different servers

The above-mentioned points are the enhancements, which can be done to increase the applicability and usage of this project. Here we can maintain the records of Electricity and Unit of Energy. Enhancements can be done to maintain all the Electricity, Unit of Energy, Bill, Store Record, and Electricity Board.

We have left all the options open so that if there is any other future requirement in the system by the user for the enhancement of the system then it is possible to implement them. In the last we would like to thanks all the persons involved in the development of the system directly or indirectly. We hope that the project will serve its purpose for which it is develop there by underlining success of process.

### CONCLUSION

This project is designed to meet the requirements of a Electricity Bill Management System. It has been developed in JAVA, keeping in mind the specifications of the system.

For designing the system we have used simple data flow diagrams.

Overall the project teaches us the essential skills like:

Using system analysis and design techniques like data flow diagram in designing the system.

Understanding the database handling and query processing using Access.

### References and Bibliography:

* Google for problem solving
* Head First Java 2nd Edition
* http://www.jdbc-tutorial.com
* http://www.javatpoint.com/java-tutorial
* http://www.tutorialspoint.com/mysql/