



Birla Institute of Technology, Mesra

Master OF COMPUTER APPLICATION

CERTIFICATE

This is to certify that this investigatory Project Entitled as “**HUMAN RESOURCE MANAGEMENT SYSTEM**” has being carried out by **RAHUL KR CHAUDHARY** of MCA Part-5 under the guidance of “**Prof NAGENDRA PRASAD TIWARY**”.

This Project work is carried out with the consultation of supervisor Prof Nagendra Prasad Tiwary.

It has been successfully completed as per the requirements and hope to complete in future and bring to its full operational stage.

Name: Rahul Kr Chaudhary
Roll: MCA/40023/19

SIGNATURE OF EXTERNAL: _____

SIGNATURE OF PROJECT GUIDE: _____

SIGNATURE OF LAB-INCHARGE: _____

DATE: _____

ACKNOWLEDGEMENT

It is our privilege to express our sincerest regards to our project coordinator, Prof Nagendra Prasad Tiwary sir, for their valuable inputs, able guidance, encouragement, whole-hearted cooperation and constructive criticism throughout the duration of our project.

We deeply express our sincere thanks to our Head of for encouraging and allowing us to present the project on the topic “**HUMAN RESOURCE MANAGEMENT SYSTEM**” at our department premises for the partial fulfilment of the requirements leading to the award of MCA degree. We take this opportunity to thank all our lecturers who have directly or indirectly helped our project.

We pay our respects and love to our parents and all other family members and friends for their love and encouragement throughout our career. Last but not the least we express our thanks to our friends for their cooperation and support.

I express my deep sense of appreciation to my known, without their effort and zeal, this significant project would not have been seen in the light of the day.

CONTENTS

❑ ACKNOWLEDGEMENT	3
❑ INTRODUCTION	6
❑ PROBLEMS AND SOLUTION OF THE PROJECT	7
❑ OBJECTIVE	9
❑ HARDWARE AND SOFTWARE REQUIREMENTS	12
❑ MODULES DESCRIPTION	13
❑ DATABASES	14
❑ CONTEXT DIAGRAM/ 0 LEVEL DFD	17
❑ 1 LEVEL DFD	18
❑ 2 LEVEL DFD	19
❑ ER DIAGRAM	20

❑ TESTING	21
❑ CODING.....	23
❑ FUTURE IMPROVEMENT.....	102
❑ CONCLUSION	103
❑ BIBLIOGRAPHY	104

INTRODUCTION

Human Resource Management System is a distributed application, developed to maintain the details of employees working in any organization. It maintains the information about the personal details of their employees, also the details about the payroll system which enable to generate the pay slip

The main aim of the Human Resource Management System software is to maintain all the sections of the organization easily. Also to develop a software application that supports the application specific to the HR automation in an intranet specific to a company there by allowing the integration of all the employees pertaining to that organization. To keep track of all the other departments related to that organization such as marketing, development etc. To allow the HR of an organization to update the employee details whenever there is a change in the employee profile pertaining to that organization. It also brings onto a string the employee specific suggestions and makes them free to post their requirements to the HR for making the organization more specific regarding the maintenance of the organization. The application should be able to store, monitor, and access huge amount of data with better performance and which should be error free.

The HR center is a powerful application designed to allow companies to streamline their human resource tasks and manage their employees more efficiently (Employee and Company Information, Employee Time, Attendance, and Leave Request HR Documentation Management (i.e. insurance forms, etc.). The HR Center includes a comprehensive employee information database, work information database, beneficiary information, and more for each employee. It comes standard with employee self-service access allowing employees to update their personal information request time off or input their daily timesheet entries. It also has role based access level control that is functionally based on whether a user is an employee, a manager, or an HR administrator. With HR center managers and HR administrators can manage the track.

PROBLEMS AND SOLUTION OF THE PROJECT

1. Problems with Existing system:

Manually it is very difficult to manage the human resource. With the help of computer it becomes easy and faster to manage the system.

In this project it is the facilities to storing and managing all the information about the employee working in that company and the projects handle by the company.

The HR administration fails short of controlling the Employees activities in analysing his/her his strength and weakness. That decision for appraisal of assigning next project to the employee or to train him/her to enhance the skills lies with proper

projection. It is not provided with the detailed project information done or to be assigned based on application.

DRAWBACKS IN EXERCISE EXISTING SYSTEM

- Need of extra manual effort
- It used to take much time to find any Employee
- Not very much accurate
- Danger of losing the files in some cases

2. Proposed system:

In the Previous System, Details are Stored Manually in papers, to share the details between employees was a financial drawback. Updations in the details is a tedious task.

But a new system was proposed to overcome the above drawbacks.

Functionalities and advantages of proposed system are:

- provide information in a quick time according to the requirements that are to be Fulfilled.
- Easier retrieval will be possible as multiple search facilities will be available.

- Security of data.
- The important process is integrated together for a proper functioning.
- Redundancy of data is removed.
- Inconsistency of data is removed.
- The Paper Work would be reduced to a greater extent.

OBJECTIVE

The name of objectives of human resource management system is to manage the details of employee department designation leave interview it manages all the information about employee event interview employee the project is totally built at administrative and thus only the administrator is guaranteed the access the purpose of the project is to build an application program to reduce the manual work for managing the employee department event designation it tracks all the details about the designation, leave, interview.

The aim is to develop a software application that supports the application specific to the HR automation in an intranet specific to a company there by allowing the integration of all the employees pertaining to that organization. To keep track of all the other departments related to that organization such as marketing, development etc.

To allow the HR of an organization to update the employee details whenever there is a change in the employee profile pertaining to that organization. It also brings onto a string the employee specific suggestions and

makes them free to post their requirements to the HR for making the organization more specific regarding the maintenance of the organization.

To develop a software application that supports Specific to the HR Automation in an intranet to a company there by allowing the interaction of all the employees pertaining to that organization to keep track of all the other departments related to that organization like marketing, research etc. and to allow the HR department of an organization to update

FEASIBILITY STUDY

The feasibility studies are undergone as follows:

1. Economic Feasibility:

More commonly known as **Cost/Benefit Analysis**. The procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs. If the benefits outweigh costs, then decision is made to design and implement the system. Considering the facts it is becoming evident that the system will be economically feasible both for developer as well as for client's respect.

2. Technical Feasibility:

Technical feasibility centers on the existing computer system (hardware, software, etc.) and to what extent it can support the proposed addition. If the budget is a serious constraint, then the project is judged not feasible. In our case this does not become an obstacle.

3. Operational feasibility:

The management & operators desire to be well acquainted with the requisite skill needed. Here most of the members in development team having technical expatriation.

4. Time feasibility:

The management & operators here concern about whether the project will completed timely or not. But considering the facts and figures collected by us regarding our project it can be easily assumed that the project will be completed within the specified time frame.

HARDWARE AND SOFTWARE REQUIREMENT

RAM	500 MB and Above
HARD DISK	2 GB Hard Disk Space and Above
PROCESSOR	I3 Processor and Above
PRINTER	Laser Printer

OPERATING SYSTEM	WINDOWS OS
DATABASE SERVER	MySQL
FRAMEWORK SERVER	Java
JDK	1.2.7

MODULE DESCRIPTION

- **Employee Entry:** It is a master record and it is used to maintain all the details of employees, i.e., name, address, basic salary, etc.
- **Privileges:** It is also a master record and it is used to maintain the records of different privileges provided to certain employees, i.e., no. of sick leave, pf, hra, etc.
- **Performance:** This module is also for master handling which details the skills of employees in various fields.
- **Attendance:** This is of transactional level. It maintains records of the daily attendance of the employees.

- **Leave:** This module is of transactional level as well and is used for requesting for leaves by employees.
- **Holiday:** This is also of transactional level. It maintains record of all the holidays.

Reports involved in the systems

- **Pay slip:** This module is used for keeping and monitoring the payments being made. It is used to show the all the payments done on the specific month. This report is helpful for the management when they want all details of specific month in a single place.

Receipts: All the payments are acknowledged and the records of the payment kept.

DATABASES

Databases to be expected in the proposed system:

1. LoginTbl

FIELD NAME	DATA TYPE	Size	CONSTRAINTS
UserId	VarChar	50	
Pass	Varchar	20	

2. RegisterTbl

FIELD NAME	DATA TYPE	Size	CONSTRAINTS
UserId	VarChar	50	
Password	Varchar	20	
Confirm Password	Varchar	20	
Email	Varchar	25	
Phone	Varchar	12	

3. RegisterTbl

FIELD NAME	DATA TYPE	Size	CONSTRAINTS
Eid	VarChar	50	
empname	Varchar	20	
Dept	Varchar	20	
Gen	Varchar	25	
Dob	Varchar	12	
Con	VarChar	25	

Adhar	VarChar	25	
Address	VarChar	25	
Designation	VarChar	25	
Doj	VarChar	25	
Last	VarChar	25	
Exp	VarChar	25	
Doc	VarChar	25	
Da	VarChar	25	
Hra	VarChar	25	
Wa	VarChar	25	
Tot_allowancw	VarChar	25	
Gpf	VarChar	25	
It	VarChar	25	
Gs	VarChar	25	
Pf	VarChar	25	
Lic	VarChar	25	
Tot_deduction	VarChar	25	
Casual	VarChar	25	
medical	VarChar	25	
Annual	VarChar	25	

4. PerformanceTbl

FIELD NAME	DATA TYPE	Size	CONSTRAINTS
Eid	VarChar	50	
empname	Varchar	20	
Designation	Varchar	20	
Training	Varchar	25	
Cstatus	Varchar	12	
Cfrom	VarChar	25	
Cto	VarChar	25	
Paid	VarChar	25	
Reimbursed	VarChar	25	
Address	VarChar	25	

5. AttTbl

FIELD NAME	DATA TYPE	Size	CONSTRAINTS
Emp_code	VarChar	50	
Dday	Varchar	20	
Mmonth	Varchar	20	
Yyear	Varchar	25	
Cstatus	Varchar	12	

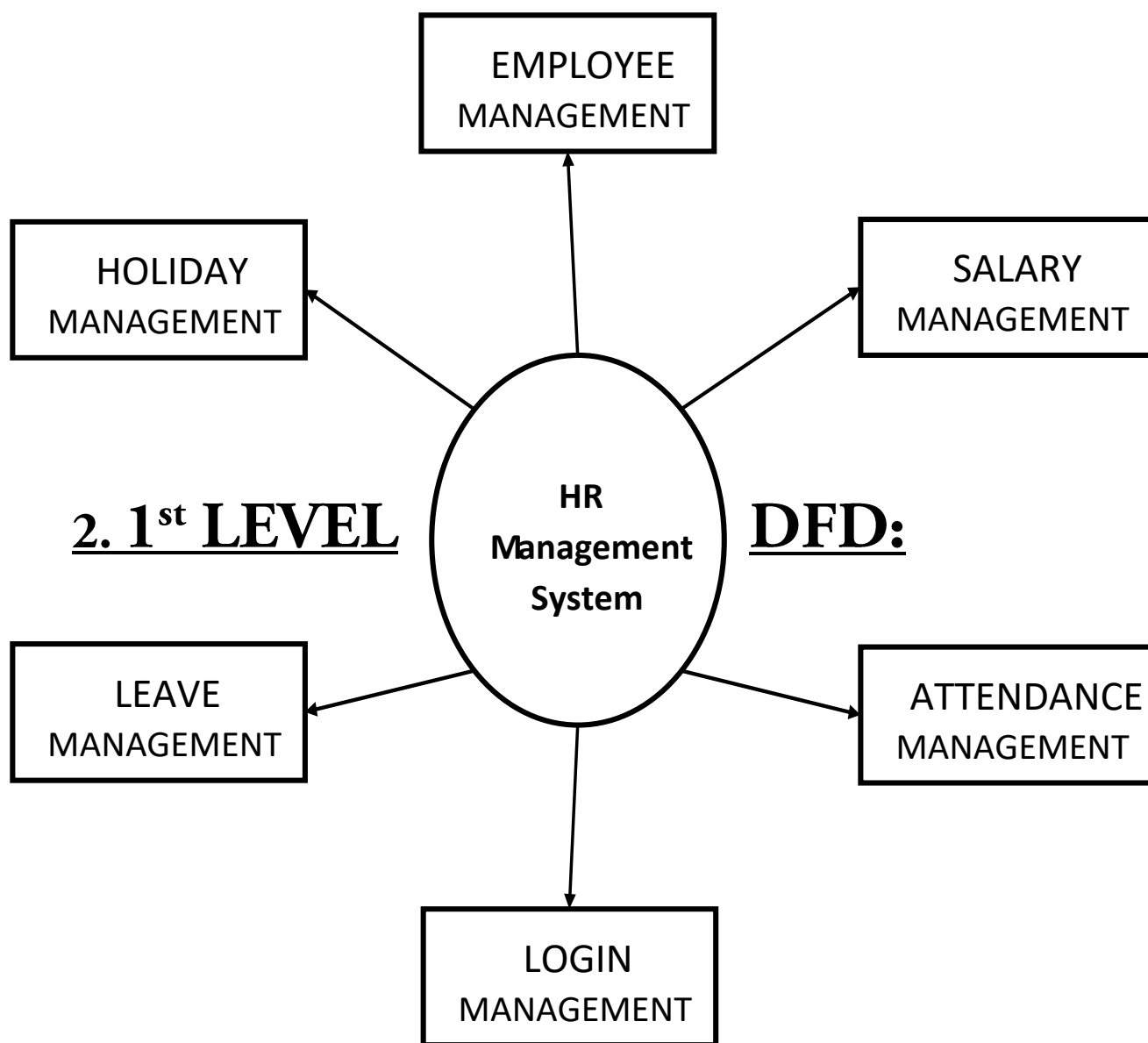
6. LeaveTbl

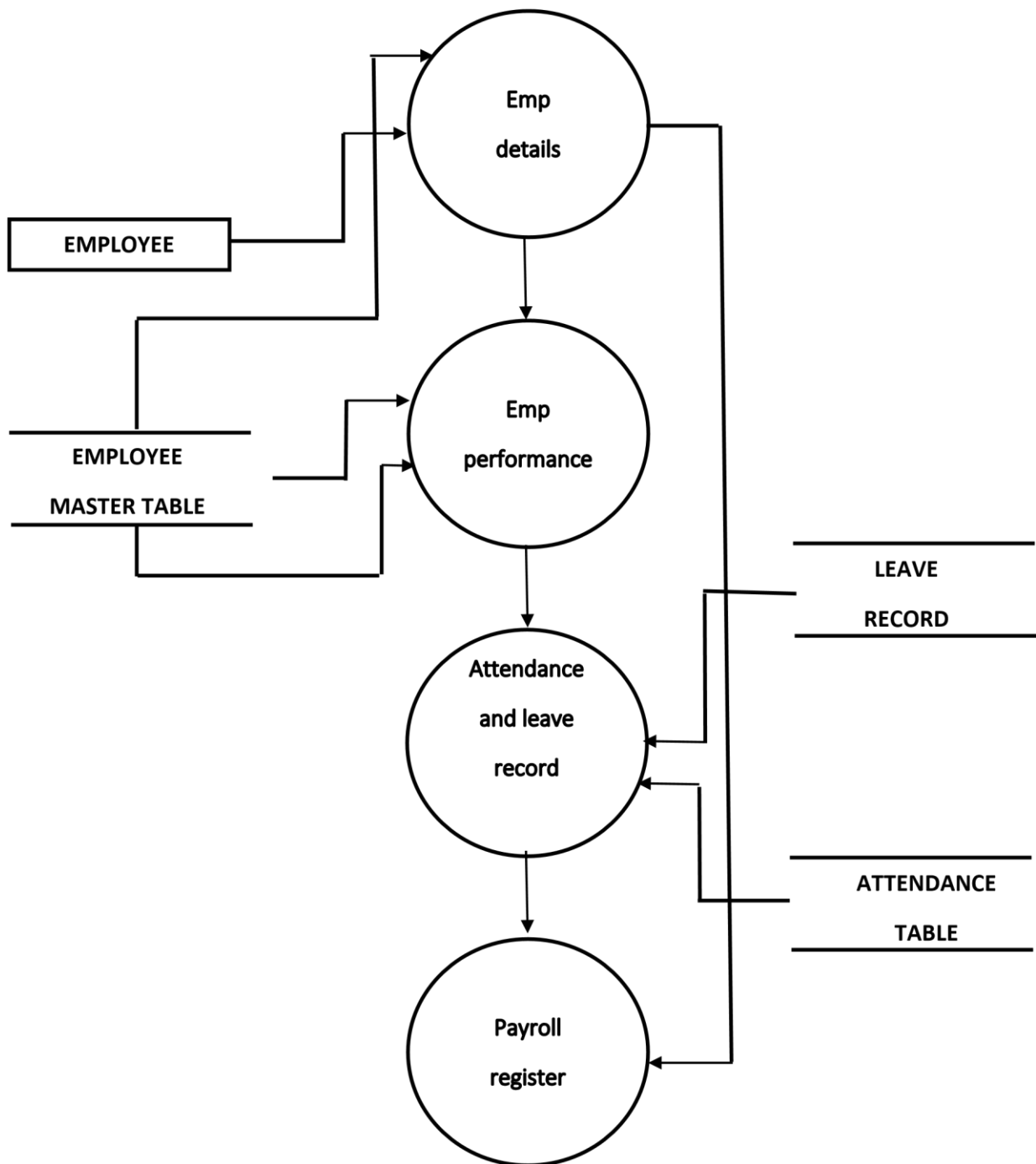
FIELD NAME	DATA TYPE	Size	CONSTRAINTS
Eid	VarChar	50	
Ename	Varchar	20	
Designation	Varchar	20	
Leave_code	Varchar	25	
Start_date	Varchar	12	
End_date	VarChar	25	
No_of_days	VarChar	25	

11. HolidayTbl

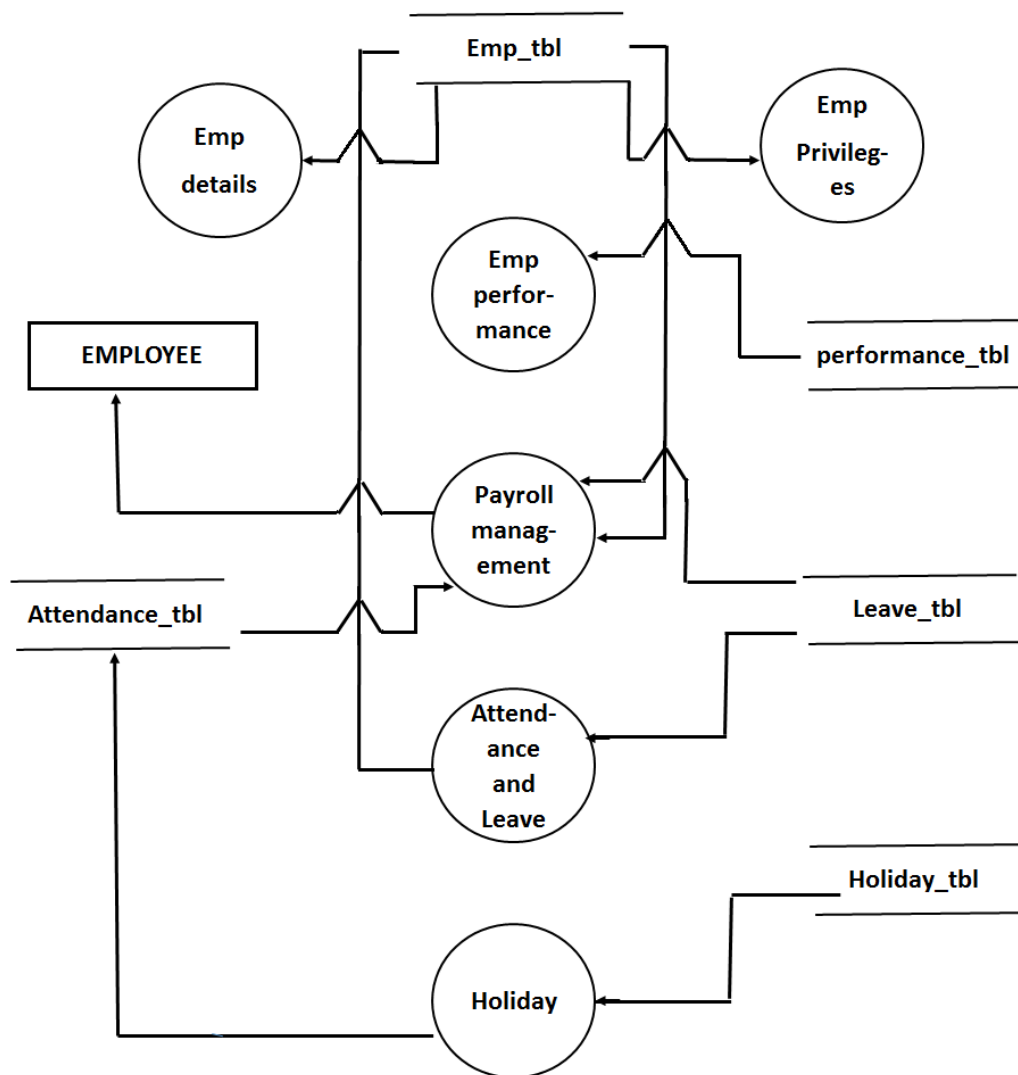
FIELD NAME	DATA TYPE	Size	CONSTRAINTS
Hdate	VarChar	50	
hname	Varchar	20	

1. CONTEXT DIAGRAM / 0-LEVEL DFD:

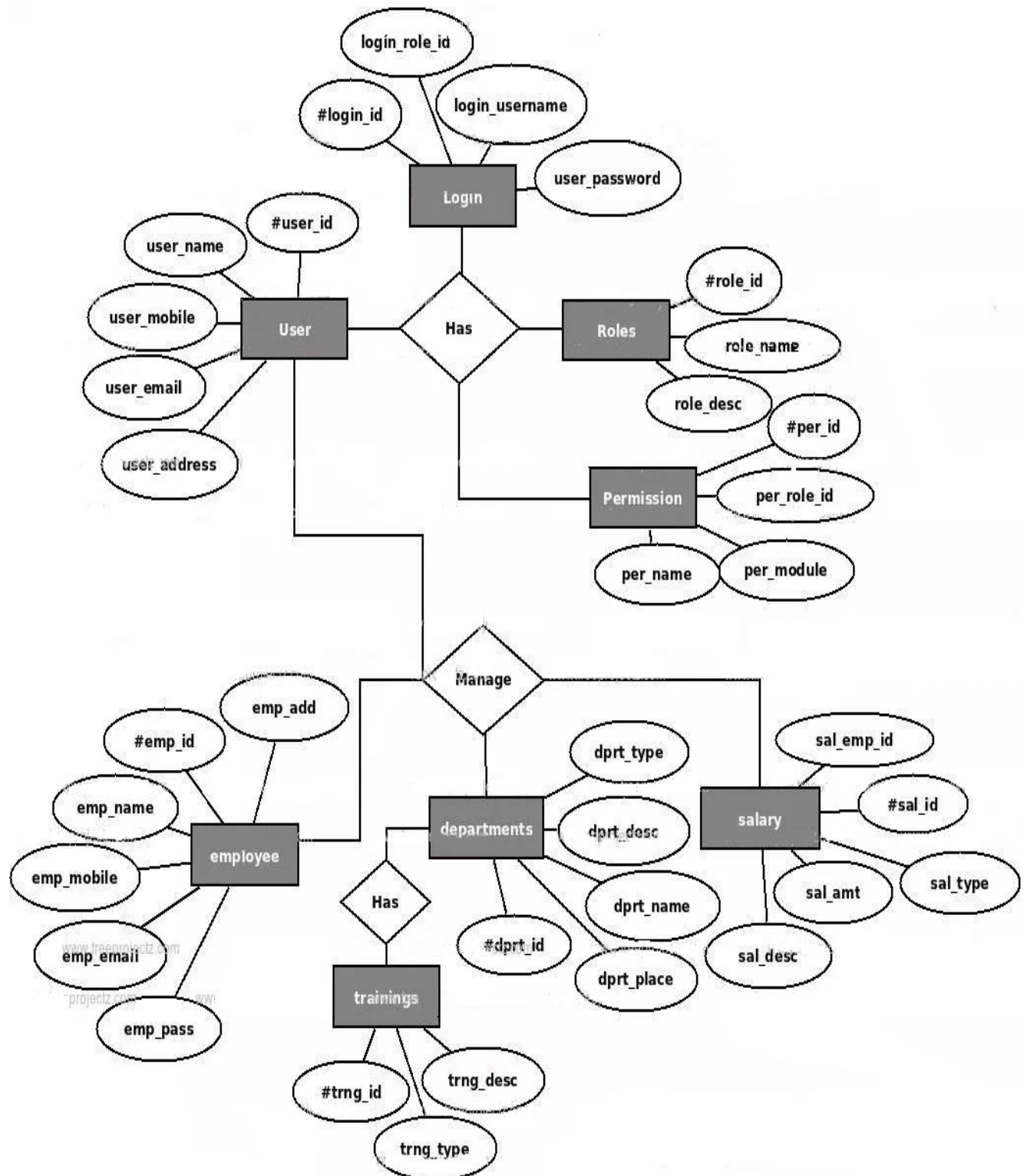




3. 2nd -LEVEL DFD:



4. ER DIAGRAM:



TESTING PROCEDURE AND SYSTEM IMPLEMENTATION PHASES

System testing is an important but critical part of system. Module testing involves the process of testing the logical units of program individually and integrating the individual modules, to test the overall system. The objective of module testing is to determine whether the module meets its specifications.

In order to perform module testing two things are to be considered: the design of test cases and the coordination of the multiple modules. Test cases may be constructed from specification or by analysing the module code. The testing strategies corresponding to tests two approaches are called as black box and white box testing. In black box testing the internal structure and behaviour of the program is not considered. In white box testing the structure of the program is examined and test data are derived from the program's logic. In non-incremental approach testing modules independently and then combining them to the program without further testing test a program. In the incremental approach a module is tested in combinations with the set of previously tested modules.

The testing plans areas that are to be tested.

1) Testing for data entry error: -

Incorrect data entry, leads to incorrect results. Sometimes it may create run time errors. So data should be checked for validity.

2) Testing for basic principles: -

If all processing is in order but display of the screens is disorderly then that has to be rectified and such screen can confuse the user.

User will be left wondering on the next actions to be taken. In some of the cases operator/user has the habit of certain key for particular option. In such cases user requirements are taken into consideration.

3) Testing for suitable error messages: -

If the error messages are not suitable phases or incorrect then there can be log off problems to the user who is non-professional won't be able to figure out what exactly the error is.

4) Testing for suitable response time: -

In some cases, the results of a particular query may take out a lot of system resources and also a longer time to process. In such cases program logic has to be rewritten. In some cases changes have been made in table.

One major factor, which has influenced the functionality of the system, is the valid data. As and when the master files were completed, the user added valid data and thus the expected results could also be verified. During testing I used incremental approach, black box testing and program instrumentation.

These processes continue till the user is fully satisfied. The data are again tested and the reports are compared with the original ones that the system should produce. The testing is done when the original data for guarantee of the outputs before it is implemented.

1. Intro Form:



```
import java.io.*;
```

```
import java.awt.*;
```

```
import java.applet.*;
```

```
import java.awt.event.*;
```

```
import javax.swing.*;
```

```
import java.sql.*;
```

```
public class intro extends JFrame implements ActionListener
```

```
{
```

```
    JButton bnext=new JButton();
```

```
    JButton bcancel=new JButton();
```

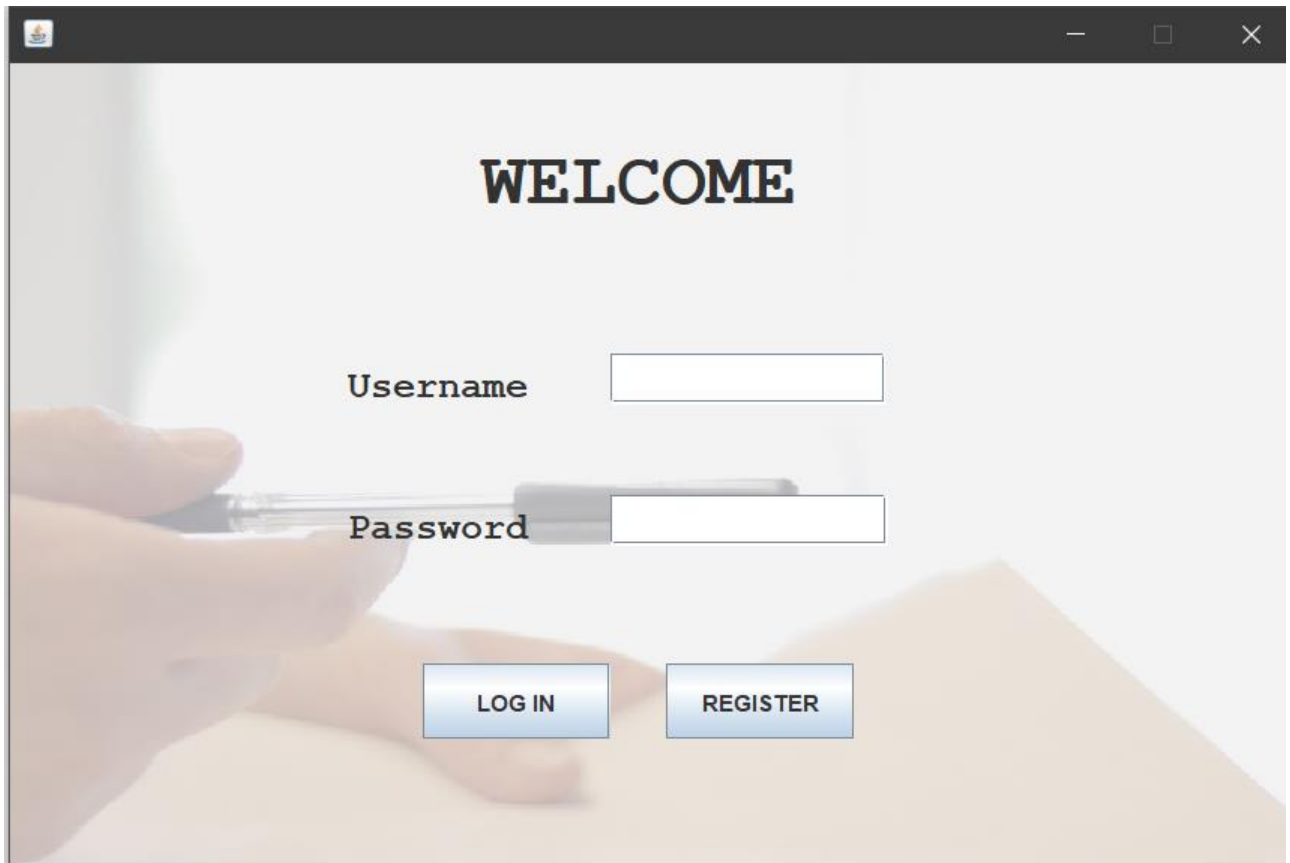
```
public intro()
```

```
{
add(bnext);
add(bcancel);
bnext.addActionListener(this);
bcancel.addActionListener(this);
bnext.setBounds(1099,1042,185,50);
bcancel.setBounds(1778,990,98,80);
}

public void actionPerformed(ActionEvent e)
{
if(e.getSource()==bnext)
{
new login();
this.dispose();
}
if(e.getSource()==bcancel)
{
this.dispose();
}
}

public static void main(String args[])throws IOException
{
intro obj=new intro();
}
}
```


2. Login Form:



```
import java.io.*;
import java.awt.*;
import java.applet.*;
import java.sql.*;
import java.util.*;
public class login extends JFrame implements ActionListener
{
    JTextField t_name=new JTextField();
    JPasswordField t_pass=new JPasswordField();
    JButton blogin=new JButton("");
    JButton breg=new JButton("");
    JButton bcan=new JButton("");
    public login()
    {
        setSize(700,465);
        add(t_pass);
```

```
        add(t_name);
        add(blogin);
        add(breg);
        add(bcan);
        blogin.addActionListener(this);
        breg.addActionListener(this);
        bcan.addActionListener(this);

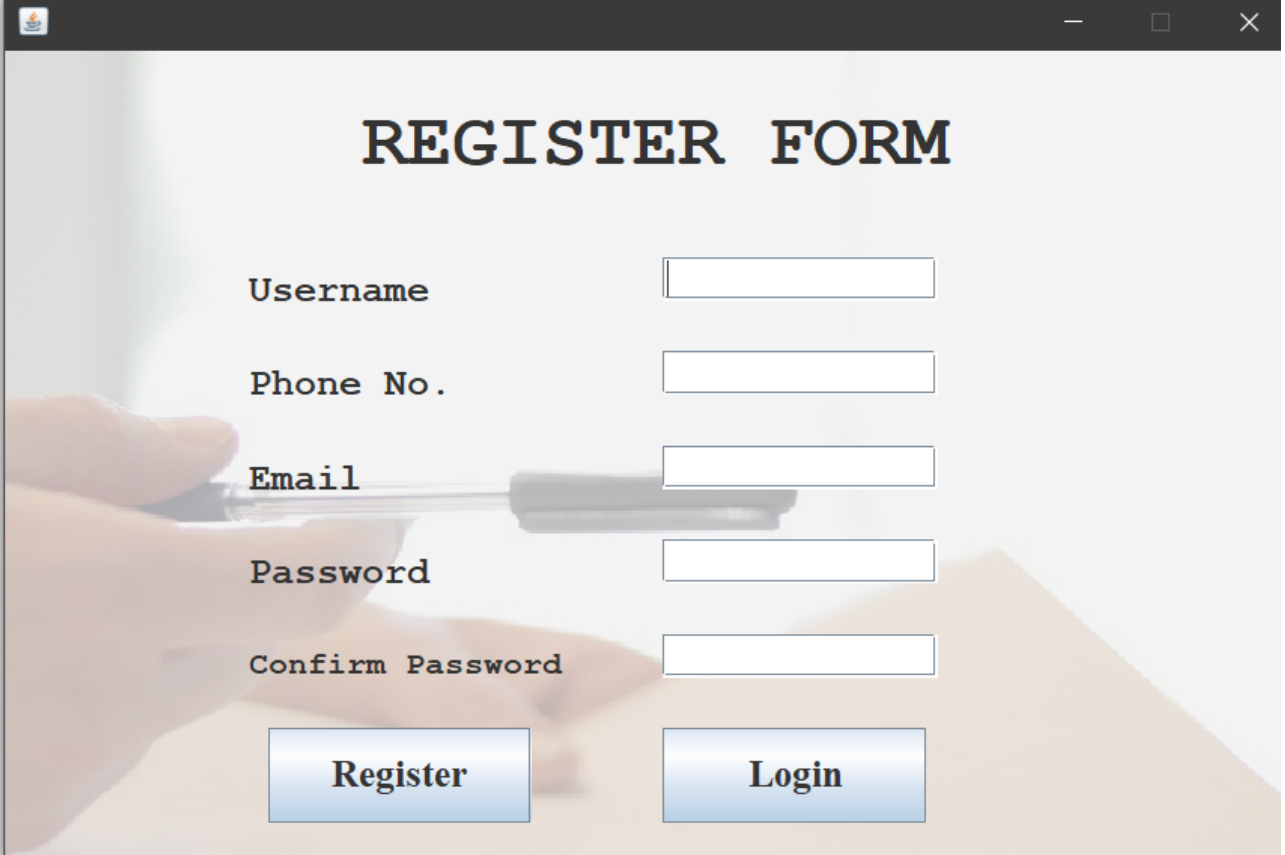
        t_pass.setEchoChar('*');
        t_name.setBounds(497,157,147,27);
        t_pass.setBounds(497,233,148,27);
        blogin.setBounds(395,332,82,28);
        breg.setBounds(500,330,83,31);
        bcan.setBounds(663,9,30,32);
    }

    public void actionPerformed(ActionEvent e)
    {
        if(e.getSource()==breg)
        {
            new register();
        }
        if(e.getSource()==bcan)
        {
            this.dispose();
        }
        if(e.getSource()==blogin)
        {
            try
            {
                Connection con=DriverManager.getConnection("jdbc:odbc:hrpdsn");
                Statement st=con.createStatement();
```

```
ResultSet res=st.executeQuery("select * from login where
username='"+t_name.getText()+"' and
password='"+t_pass.getText()+"'");
if(res.next())
{
    ctr=0;
    JOptionPane.showMessageDialog(null,"welcome");
    new menu();
}
else
{
    JOptionPane.showMessageDialog(null,"Invalid Username");
    ctr++;
}
}
catch(Exception ex)
{
    System.out.print(ex);
}
}

public static void main(String args[])throws IOException
{
    login obj=new login();
}
}
```

3. Register Form:



REGISTER FORM

Username

Phone No.

Email

Password

Confirm Password

Register **Login**

```
import java.io.*;
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
import java.sql.*;
public class register extends JFrame implements ActionListener
{
    JTextField t_name=new JTextField();
    JTextField t_email=new JTextField();
    JTextField t_phn=new JTextField();
    JPasswordField t_pass=new JPasswordField();
    JPasswordField t_con=new JPasswordField();
    JButton breg=new JButton("");
    JButton bcancel=new JButton("");
```

```
        JButton blog=new JButton("");
        public register()
        {
            setSize(700,465);
            add(t_pass);
            add(t_con);
            add(t_name);
            add(t_email);
            add(t_phn);
            add(blog);
            add(breg);
            add(bcancel);
            breg.addActionListener(this);
            bcancel.addActionListener(this);
            t_name.setBounds(498,137,146,23);
            t_phn.setBounds(498,173,146,23);

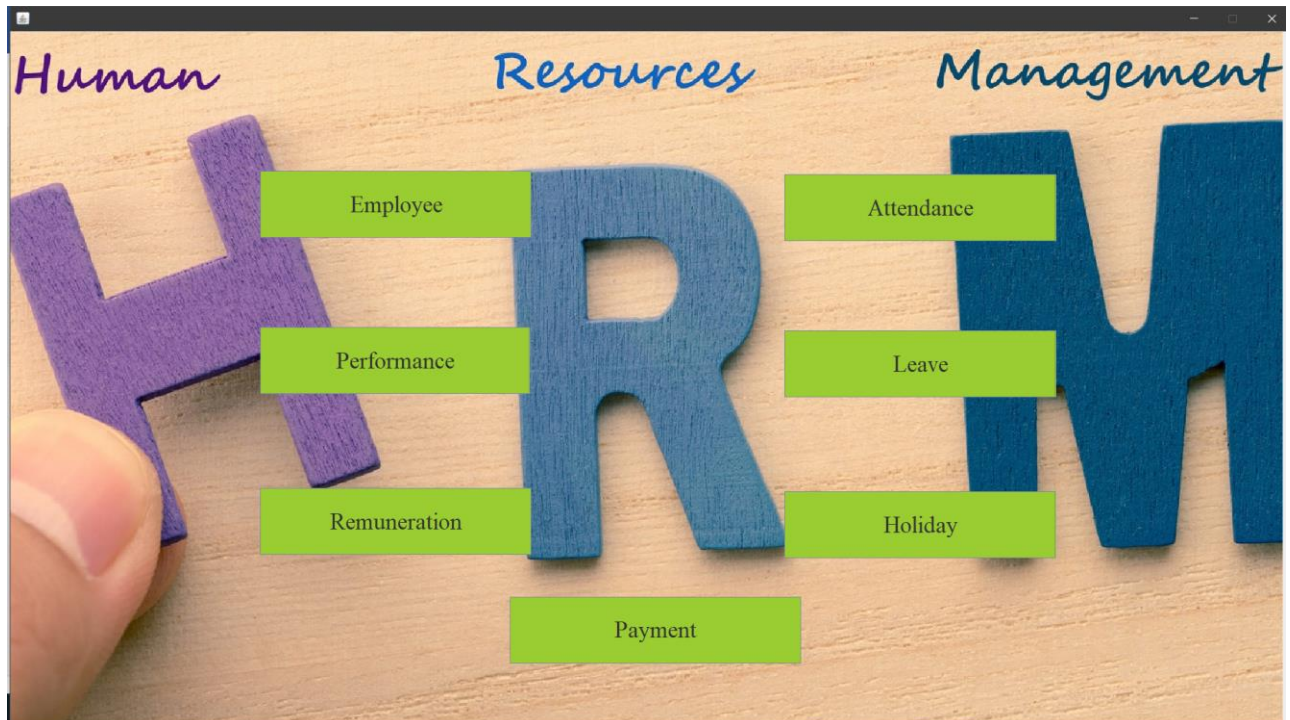
            t_con.setBounds(498,281,146,23);
            breg.setBounds(450,335,84,31);
            blog.setBounds(423,395,135,25);
            bcancel.setBounds(663,9,30,32);
        }
        public void actionPerformed(ActionEvent e)
        {
            if(e.getSource()==bcancel)
            {
                this.dispose();
            }
            if(e.getSource()==blog)
            {
                new login();
            }
        }
    }
}
```

```

    }
    if(e.getSource()==breg)
    {
        Try
        {
            if((t_name.getText().equals(""))||t_pass.getText().equals(""))||t_con.getText().equals(
            "")||t_email.getText().equals(""))||t_phn.getText().equals(""))
                if(t_pass.getText().equals(t_con.getText()))
                {
                    Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
                    String str="Insert into login(username,password,email,phone)values(?,?,?,?)";
                    PreparedStatement ps=con.prepareStatement(str);
                    ps.setString(1,t_name.getText());
                    ps.setString(2,t_pass.getText());
                    ps.setString(3,t_email.getText());
                    ps.setString(4,t_phn.getText());
                    ps.executeUpdate();
                }
            else
            {
                JOptionPane.showMessageDialog(null,"Password Not Matched!");
            }
        }
        catch(Exception ex)
        {
            System.out.print(ex);
        }
    }
    public static void main(String args[])throws IOException
    {
        register obj=new register();
    }
}

```

4. Master Form:



```
import java.io.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class masters extends JFrame implements ActionListener
{
    JButton bemp=new JButton();
    JButton bper=new JButton();
    JButton brem=new JButton();
    JButton batt=new JButton();
    JButton bleave=new JButton();
    JButton bhol=new JButton();
    JButton bpay=new JButton();
    public masters()
    {
        add(bemp);
        add(bper);
```

```

        add(brem);
        add(bleave);
        add(bhol);
        add(bpay);
        add(bcancel);
        bemp.addActionListener(this);
        bper.addActionListener(this);
        batt.addActionListener(this);
        bleave.addActionListener(this);
        bhol.addActionListener(this);
        bemp.setBounds(653,518,326,60);
        bper.setBounds(652,703,324,60);
        brem.setBounds(648,897,326,60);
        batt.setBounds(1518,520,326,60);
        bleave.setBounds(1517,707,326,60);
    }
    public void actionPerformed(ActionEvent e)
    {
        if(e.getSource()==bemp)
        {
            new emp();
        }
        if(e.getSource()==bper)
        {
            new performance();
        }
        if(e.getSource()==brem)
        {
            new remuneration();
        }
        if(e.getSource()==batt)

```



```
        {
            new attendance();
        }
        if(e.getSource()==bleave)
        {
            new leave();
        }
        if(e.getSource()==bhol)
        {
            new holiday();
        }
        if(e.getSource()==bpay)
        {
            new payslip();
        }
        if(e.getSource()==bcancel)
        {
            this.dispose();
        }
    }
    public static void main(String args[])throws IOException
    {
        masters obj=new masters();
    }
}
```

5. Employee Form:

EMPLOYEE DETAILS

Employee ID	<input type="text"/>	Department	<input type="text" value="Select Department"/>
Name	<input type="text"/>	Contact No.	<input type="text"/>
Gender	<input type="radio"/> M <input type="radio"/> F	DOB	<input type="text" value="dd"/> <input type="text" value="mm"/> <input type="text" value="yyyy"/>
Designation	<input type="text" value="Select"/>	Aadhar No.	<input type="text"/>
Address	<input type="text"/>	DOJ	<input type="text" value="dd"/> <input type="text" value="mm"/> <input type="text" value="yyyy"/>
Basic	<input type="text"/>	Experience	<input type="text" value="Select"/>
		Previous Workplace	<input type="text"/>

```

import java.io.*;
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
import javax.swing.*;
import java.sql.*;

public class emp extends JFrame implements ActionListener
{
    JTextField t_eid=new JTextField();
    JTextField t_ename=new JTextField();
    JTextField t_cont=new JTextField();
    JTextField t_adh=new JTextField();
    JTextField t_addr=new JTextField();
    JTextField t_last=new JTextField();
    JTextField t_doc=new JTextField();

    String dept[]={"select
department", "Sales", "Analyst", "Purchase", "EDP", "Developer", "Programmer", "Desig
ner"};

```

```

JComboBox cmbdpt=new JComboBox(dept);
JComboBox cmbdojd=new JComboBox();
JComboBox cmbdojm=new JComboBox();
JComboBox cmbdojy=new JComboBox();
JComboBox cmbdobd=new JComboBox();
JComboBox cmbdobm=new JComboBox();
JComboBox cmbdoby=new JComboBox();
JComboBox cmbdes=new JComboBox(des);
String exp[]={"Select","04","05","06","07","08","09","10+"};
JComboBox cmbexp=new JComboBox(exp);
JRadioButton rbmale=new JRadioButton();
JRadioButton rbfemale=new JRadioButton();
ButtonGroup gen=new ButtonGroup();
JButton bnew=new JButton();
JButton bsave=new JButton();
JButton bsearch=new JButton();
JButton bedit=new JButton();
JButton bcancel=new JButton();
JButton epri=new JButton();
public emp()
{
setSize(2500,1500);
cmbdobd.addItem("dd");
for(int d=1;d<=31;d++)
{
cmbdobd.addItem(""+d);
}
cmbdobm.addItem("mm");
for(int m=1;m<=12;m++)
{
cmbdobm.addItem(""+m);
}

```

```
}
cmbdoby.addItem("yyyy");
for(int y=1985;y<=2019;y++)
{
cmbdoby.addItem(""+y);
}
cmbdojd.addItem("dd");
for(int d=1;d<=31;d++)
{
cmbdojd.addItem(""+d);
}
cmbdojm.addItem("mm");
for(int m=1;m<=12;m++)
{
cmbdojm.addItem(""+m);
}
cmbdojy.addItem("yyyy");
for(int y=2005;y<=2019;y++)
{
cmbdojy.addItem(""+y);
}
add(t_eid);
add(t_ename);
add(t_cont);
add(t_adh);
add(t_addr);
add(t_last);
add(t_doc);
add(cmbdpt);
add(cmbdobd);
add(cmbdobm);
```

```
add(cmbdoby);  
add(cmbdojd);  
add(cmbdojm);  
add(cmbdojy);  
add(cmbdes);  
add(cmbexp);  
add(rbmale);  
add(rbfemale);  
gen.add(rbmale);  
gen.add(rbfemale);
```

```
add(bnew);  
add(bsave);  
add(bsearch);  
add(bedit);  
add(bcancel);  
add(epri);  
bnew.addActionListener(this);  
bsave.addActionListener(this);  
bsearch.addActionListener(this);  
bedit.addActionListener(this);  
bcancel.addActionListener(this);  
t_eid.setBounds(923,530,180,34);  
t_ename.setBounds(923,590,180,33);  
t_cont.setBounds(1560,589,180,34);  
t_adh.setBounds(1560,706,180,34);  
t_addr.setBounds(928,763,317,83);  
cmbdobd.setBounds(1562,645,61,35);  
cmbdobm.setBounds(1634,645,61,35);  
cmbdoby.setBounds(1708,645,69,35);  
cmbdojd.setBounds(1562,762,60,35);
```

```

cmbdojm.setBounds(1634,761,61,35);
cmbdojy.setBounds(1708,762,69,35);
cmbdes.setBounds(923,704,180,33);
cmbexp.setBounds(1562,820,98,40);
rbmale.setBounds(935,648,20,20);
rbfemale.setBounds(1013,648,20,20);
bnew.setBounds(812,1003,125,41);
bsave.setBounds(981,1003,125,41);
bsearch.setBounds(1148,1003,125,41);
bedit.setBounds(1316,1001,125,41);
bcancel.setBounds(1484,1000,125,41);
epri.setBounds(1753,405,120,100);
}
public void actionPerformed(ActionEvent e)
{
if(e.getSource()==bnew)
{
try
{
Connection con=DriverManager.getConnection("jdbc:odbc:empdsn");String
str="select * from emp";
Statement st=con.createStatement();
ResultSet res=st.executeQuery(str);
while(res.next())
{
ctr++;
}
t_eid.setText("E00"+ctr);
t_ename.setText("");
t_cont.setText("");
t_adh.setText("");

```

```

t_addr.setText("");
gen.clearSelection();
}
catch(Exception ex)
{
    System.out.print(ex);
}
}
if(e.getSource()==bsave)
{
    try
    {
        Connection con=DriverManager.getConnection("jdbc:odbc:empdsn");
        String str="Insert into emp(eid,empname,dept,gen,dob,con,adhar,addr,des,doj,
last,exp,doc)values(?,?,?,?,?,?,?,?,?,?,?,?,?)";
        PreparedStatement ps=con.prepareStatement(str);ps.setString(1,t_eid.getText());
        ps.setString(2,t_ename.getText());
        ps.setString(3,""+cmbdpt.getSelectedItem());
        String gen="";
        if(rbmale.isSelected())
        gen="MALE";
        else
        gen="FEMALE";
        ps.setString(4,gen);
        String
        dob=cmbdobd.getSelectedItem()+"/"+cmbdobm.getSelectedItem()+"/"+cmbdoby.ge
tSelectedItem();
        ps.setString(5,dob);
        ps.setString(6,t_cont.getText());
        ps.setString(7,t_adh.getText());
        ps.setString(8,t_addr.getText());
        ps.setString(9,""+cmbdes.getSelectedItem());
    }
    catch(Exception ex)
    {
        System.out.print(ex);
    }
}
}

```

```

String doj=cmbdojd.getSelectedItemAt()+"/"+cmbdojm.getSelectedItemAt()+
"/"+cmbdojy.getSelectedItemAt();
ps.setString(10,doj);
ps.setString(11,t_last.getText());
ps.setString(12,""+cmbexp.getSelectedItemAt());
ps.setString(13,t_doc.getText());
ps.executeUpdate();
JOptionPane.showMessageDialog(null,"Saved");
}
catch(Exception ex)
{
System.out.print(ex);
}
}
if(e.getSource()==bsearch)
{
try
{
Connection con=DriverManager.getConnection("jdbc:odbc:empdsn");
Statement st=con.createStatement();
ResultSet res=st.executeQuery("select * from emp where eid=
"+t_eid.getText()+"");
while(res.next())
{
t_ename.setText(res.getString(2));
cmbdpt.setSelectedItem(res.getString(3));
String gen=res.getString(4);
if(gen.equals("MALE"))
rbmale.setSelected(true);
else
rbfemale.setSelected(true);
}
}
}

```



```

String arr[]=res.getString(5).toString().split("/");
cmbdobd.setSelectedItem(arr[0]);
cmbdobm.setSelectedItem(arr[1]);
cmbdoby.setSelectedItem(arr[2]);
t_cont.setText(res.getString(6));
t_adh.setText(res.getString(7));
t_addr.setText(res.getString(8));
cmbdes.setSelectedItem(res.getString(9));
String arr1[]=res.getString(10).toString().split("/");
cmbdojd.setSelectedItem(arr1[0]);
cmbdojm.setSelectedItem(arr1[1]);
cmbdojy.setSelectedItem(arr1[2]);
}
catch(Exception ex)
{
System.out.print(ex);
}
}
if(e.getSource()==bedit)
{
try
{
int flag=1;
if(cmbdojy.getSelectedItem().equals("yyyy")||cmbdoby.getSelectedItem().equals("y
yyy")||cmbdojm.getSelectedItem().equals("mm")||cmbdobm.getSelectedItem().equa
ls("mm")||cmbdojd.getSelectedItem().equals("dd")||cmbdobd.getSelectedItem().equ
als("dd"))
{
flag=0;
}
if(flag==1)
{

```

```

Connection con=DriverManager.getConnection("jdbc:odbc:empdsn");
Statement st=con.createStatement();
String g;
if(rbmale.isSelected())
g="MALE";
else
g="FEMALE";
String str="Update emp set
empname='"+t_ename.getText()+"',dept='"+cmbdpt.getSelectedItemId()+"',dob='"+cm
bdobd.getSelectedItemId()+"/"+cmbdobm.getSelectedItemId()+"/"+cmbdoby.getSelecte
dItemId()+"',con='"+t_cont.getText()+"',adhar='"+t_adh.getText()+"',addr='"+t_addr.ge
tText()+"',des='"+cmbdes.getSelectedItemId()+"',doj='"+cmbdojd.getSelectedItemId()+/
"+cmbdojm.getSelectedItemId()+"/"+cmbdojy.getSelectedItemId()+"',last='"+t_last.getT
ext()+"',exp='"+cmbexp.getSelectedItemId()+"',doc='"+t_doc.getText()+"',gen='"+g+"
where eid='"+t_eid.getText()+"'";
st.executeUpdate(str);
}
}
catch(Exception ex)
{
System.out.print(ex);
}
}
if(e.getSource()==bcancel)
{
this.dispose();
}
}
public static void main(String args[])throws IOException
{
emp obj=new emp();
}
}

```

6. Privileges Form:

PRIVILEGES

Employee ID Name Department

DA GPF Casual Leave

HRA IT Sick Leave

WA GS Medical Leave

PF Annual Leave

LIC

Total Allowances Total Deductions

```
import java.io.*;
```

```
import java.awt.*;
```

```
import java.applet.*;
```

```
import java.awt.event.*;
```

```
import java.sql.*;
```

```
public class privileges extends JFrame implements  
ActionListener,FocusListener,ItemListener
```

```
{
```

```
    JTextField t_eid=new JTextField();
```

```
    JTextField t_ename=new JTextField();
```

```
    JTextField t_da=new JTextField();
```

```
    JTextField t_hra=new JTextField();
```

```
    JTextField t_wa=new JTextField();
```

```
    JTextField t_totall=new JTextField(" ");
```

```
    JTextField t_gpf=new JTextField();
```

```
    JTextField t_it=new JTextField();
```

```
    JTextField t_gs=new JTextField();
```

```
    JTextField t_pf=new JTextField();
```

```

JTextField t_lic=new JTextField();
JTextField t_totded=new JTextField(" ");
JTextField t_cas=new JTextField();
JTextField t_sick=new JTextField();

String dept[]={"Select
Department","Sales","Analyst","Purchase","EDP","Developer","Programmer","Desig
ner"};

JComboBox cmbdpt=new JComboBox(dept);
JButton bnew=new JButton();
JButton bsave=new JButton();
JButton bsearch=new JButton();
JButton bcancel=new JButton();
JButton btotall=new JButton();

public privileges()
{
add(t_eid);
add(t_ename);
add(t_da);
add(t_hra);
add(t_wa);
add(t_totall);
add(t_gpf);
add(t_it);
add(t_gs);
add(t_pf);
add(t_lic);
add(t_totded);
add(t_cas);
add(t_sick);
add(bnew);
add(bsave);
add(bsearch);

```

```
add(bcancel);
add(bttotal);
add(cmbdpt);
cmbdpt.addItemListener(this);
t_eid.setBounds(834,530,181,33);
    t_da.setBounds(826,642,181,33);
t_hra.setBounds(825,700,181,33);
t_wa.setBounds(825,754,181,33);
t_totall.setBounds(1047,920,181,32);
t_ename.setBounds(1190,530,181,33);
t_gpf.setBounds(1200,642,181,33);
t_it.setBounds(1200,698,181,33);
t_gs.setBounds(1202,753,181,33);
t_pf.setBounds(1200,808,181,33);
t_lic.setBounds(1200,861,181,33);
t_totded.setBounds(1425,920,181,32);
t_cas.setBounds(1639,642,181,33);
t_sick.setBounds(1639,698,181,33);
t_med.setBounds(1639,753,181,33);
t_ann.setBounds(1639,807,181,33);
cmbdpt.setBounds(1637,531,181,33);
bnew.addActionListener(this);
bsave.addActionListener(this);
bsearch.addActionListener(this);
bcancel.addActionListener(this);
bttotal.addActionListener(this);
t_eid.addFocusListener(this);
t_totall.addFocusListener(this);
t_totded.addFocusListener(this);
bnew.setBounds(812,1010,123,42);
bsave.setBounds(980,1010,123,42);
```

```
bsearch.setBounds(1148,1010,123,42);
bcancel.setBounds(1316,1010,123,42);
bttotal.setBounds(675,920,123,42);
}

public void actionPerformed(ActionEvent e)
{
    if(e.getSource()==bnew)
    {
        try
        {
            Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
            String str="select * from emp";
            PreparedStatement ps=con.prepareStatement(str);
            Statement st=con.createStatement();
            ResultSet res=st.executeQuery(str);
            while(res.next())
            {
                ctr++;
            }
            if(ctr>0&&ctr<=9)
            {
                t_eid.setText("E00"+ctr);
            }
            else if(ctr>=10&&ctr<=99)
            {
                t_eid.setText("E0"+ctr);
            }
            else if(ctr>=100)
            {
                t_eid.setText("E"+ctr);
            }
        }
    }
}
```

```

t_ename.setText("");
t_da.setText("");
t_hra.setText("");
t_wa.setText("");
t_totall.setText("");
t_gpf.setText("");
t_it.setText("");
t_gs.setText("");
t_pf.setText("");
t_lic.setText("");
t_totded.setText("");
t_cas.setText("");
t_sick.setText("");
t_med.setText("");
t_ann.setText("");
cmbdpt.setSelectedIndex(0);
}
catch(Exception ex)
{
System.out.print(ex);
}
if(e.getSource()==bsave)
{
try
{
Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
Statement st=con.createStatement();

String str="Update emp set
da='"+t_da.getText()+"',hra='"+t_hra.getText()+"',wa='"+t_wa.getText()+"',tot_allowa
nce='"+t_totall.getText()+"',gpf='"+t_gpf.getText()+"',it='"+t_it.getText()+"',gs='"+t_gs
.getText()+"',pf='"+t_pf.getText()+"',lic='"+t_lic.getText()+"',tot_deduction='"+t_totde

```

```

d.getText()+"",casual=""+t_cas.getText()+"",sick=""+t_sick.getText()+"",medical=""+t_
med.getText()+"",annual=""+t_ann.getText()+" where eid=""+t_eid.getText()+"";
st.executeUpdate(str);
JOptionPane.showMessageDialog(null,"Saved");
}
}
catch(Exception ex)
{
System.out.print(ex);
}
}
if(e.getSource()==bsearch)
{
try
{
int flg=0;
        Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
Statement st=con.createStatement();
String eid=JOptionPane.showInputDialog(null,"Enter Employee ID:");
ResultSet res=st.executeQuery("Select * from emp where eid="+eid+"");
while(res.next())
{
t_eid.setText(res.getString(1));
t_ename.setText(res.getString(2));
cmbdpt.setSelectedItem(res.getString(3));
        t_da.setText(res.getString(14));
        t_hra.setText(res.getString(15));
        t_wa.setText(res.getString(16));
        t_totall.setText(res.getString(17));
        t_gpf.setText(res.getString(18));
        t_it.setText(res.getString(19));

```



```

        t_gs.setText(res.getString(20));
        t_pf.setText(res.getString(21));
        t_lic.setText(res.getString(22));
        t_totded.setText(res.getString(23));
        t_cas.setText(res.getString(24));
        t_sick.setText(res.getString(25));
        t_med.setText(res.getString(26));
    }
}
catch(Exception ee)
{
    System.out.println(ee);
}
}
if(e.getSource()==bcancel)
{
    this.dispose();
}
if(e.getSource()==bttotal)
{
    int da=Integer.parseInt(t_da.getText());
    int hra=Integer.parseInt(t_hra.getText());
    int totall=da+hra+wa;
    int gpf=Integer.parseInt(t_gpf.getText());
    int it=Integer.parseInt(t_it.getText());
    int gs=Integer.parseInt(t_gs.getText());
    int pf=Integer.parseInt(t_pf.getText());
    int lic=Integer.parseInt(t_lic.getText());
    int totded=gpf+it+gs+pf+lic;
}
}

```

```
public void focusLost(FocusEvent f)
{
    if(f.getComponent()==t_eid)
    {
        try
        {
            Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
            Statement st=con.createStatement();
            ResultSet res=st.executeQuery("select * from emp where id='"+t_eid.getText()+"");
            while(res.next())
            {
                t_ename.setText(res.getString(2));
                cmbdpt.setSelectedItem(res.getString(3));
            }
        }
        catch(Exception ex)
        {
            System.out.print(ex);
        }
    }

    public void focusGained(FocusEvent f)
    {}

    public void itemStateChanged(ItemEvent ie)
    {
        if(ie.getSource()==cmbdpt)
        {
            try
            {
                int flg=1;
                Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
                Statement st=con.createStatement();
```

```

String tempcode=""+cmbdpt.getSelectedItem();
        ResultSet res=st.executeQuery("Select * from remuneration where
        Emp_Type='"+tempcode+"'");
while(res.next())
{
t_hra.setText(res.getString(2));
    t_da.setText(res.getString(3));
        t_wa.setText(res.getString(4));
        t_gpf.setText(res.getString(5));
        t_it.setText(res.getString(6));
        t_gs.setText(res.getString(7));
        t_pf.setText(res.getString(8));
        t_lic.setText(res.getString(9));
        t_cas.setText(res.getString(10));
        t_sick.setText(res.getString(11));
        t_med.setText(res.getString(12));
        t_ann.setText(res.getString(13));
    }
}
catch(Exception e)
{
System.out.println(e);
}
}
}

public static void main(String args[])throws IOException
{
privileges obj=new privileges();
}
}

```

7. Performance Form:

```

import java.io.*;
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
import java.sql.*;

public class performance extends JFrame implements
ActionListener,FocusListener
{
    JTextField t_eid=new JTextField();
    JTextField t_ename=new JTextField();
    JTextField t_training=new JTextField();
    JTextField t_paid=new JTextField();
    JTextField t_reimbursed=new JTextField();
    JTextField t_addr=new JTextField();
    String
des[]={"Select","BA","BCOM","BSC","BBA","BCA","BTECH","MA","MCOM","MSC","
MBA","MCA","MTECH"};
    JComboBox cmbdes=new JComboBox(des);
    JComboBox cmbfromd=new JComboBox();

```

```

JComboBox cmbfromm=new JComboBox();
JComboBox cmbfromy=new JComboBox();
JComboBox cmbtod=new JComboBox();
JComboBox cmbtom=new JComboBox();
JComboBox cmbtoy=new JComboBox();
String stat[]={"--SELECT--","CERTIFIED","NOT CERTIFIED","INCOMPLETE"};
JComboBox cmbstatus=new JComboBox(stat);
JButton bnew=new JButton();
JButton bsubmit=new JButton();
JButton bcancel=new JButton();
public performance()
{
    cmbfromd.addItem("dd");
    for(int d=1;d<=31;d++)
    {
        cmbfromd.addItem(""+d);
    }
    cmbfromm.addItem("mm");
    for(int m=1;m<=12;m++)
    {
        cmbfromm.addItem(""+m);
    }
    cmbfromy.addItem("yyyy");
    for(int y=2010;y<=2019;y++)
    {
        cmbfromy.addItem(""+y);
    }
    cmbtod.addItem("dd");
    for(int d=1;d<=31;d++)
    {
        cmbtod.addItem(""+d);
    }

```

```
}  
cmbtom.addItem("mm");  
for(int m=1;m<=12;m++)  
{  
    cmbtom.addItem(""+m);  
}  
  
cmbtoy.addItem("yyyy");  
for(int y=2010;y<=2019;y++)  
{  
    cmbtoy.addItem(""+y);  
}  
add(t_eid);  
add(t_ename);  
add(t_training);  
add(t_paid);  
add(t_reimbursed);  
add(t_addr);  
add(cmbdes);  
add(cmbfromd);  
add(cmbfromm);  
add(cmbfromy);  
add(cmbtod);  
add(cmbtom);  
add(cmbtoy);  
add(cmbstatus);  
add(bnew);  
add(bsubmit);  
add(bcancel);  
bnew.addActionListener(this);  
bsubmit.addActionListener(this);
```

```

bcancel.addActionListener(this);
t_eid.setBounds(924,532,181,33);
t_ename.setBounds(924,600,181,33);
t_training.setBounds(924,667,181,35);
t_paid.setBounds(925,804,181,35);
t_reimbursed.setBounds(1565,804,181,35);
t_addr.setBounds(1560,600,317,83);
cmbdes.setBounds(1560,532,181,33);
cmbtod.setBounds(922,732,60,35);
cmbtom.setBounds(993,732,61,35);
cmbtoy.setBounds(1067,733,69,35);
cmbfromd.setBounds(1562,731,60,35);
cmbfromm.setBounds(1634,731,61,35);
cmbfromy.setBounds(1708,732,69,35);
cmbstatus.setBounds(924,875,181,35);
bnew.setBounds(965,995,125,41);
bsubmit.setBounds(1147,996,125,41);
t_eid.addFocusListener(this);
t_ename.addFocusListener(this);
t_paid.addFocusListener(this);
t_reimbursed.addFocusListener(this);
}
public void actionPerformed(ActionEvent e)
{
    if(e.getSource()==bnew)
    {
        try
        {
            Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
            String str="select * from performance";
            PreparedStatement ps=con.prepareStatement(str);

```

```
Statement st=con.createStatement();
ResultSet res=st.executeQuery(str);
while(res.next())
{
    ctr++;
}
if(ctr>0&&ctr<=9)
{
    t_eid.setText("E00"+ctr);
}
else if(ctr>=10&&ctr<=99)
{
    t_eid.setText("E0"+ctr);
}
else if(ctr>=100)
{
    t_eid.setText("E"+ctr);
}
t_ename.setText("");
t_training.setText("");
t_paid.setText("");
cmbtod.setSelectedIndex(0);
cmbtom.setSelectedIndex(0);
cmbtoy.setSelectedIndex(0);
cmbstatus.setSelectedIndex(0);
}
catch(Exception ex)
{
    System.out.print(ex);
}
}
```



```

if(e.getSource()==bsubmit)
{
try
{
Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");

    String str="insert into
    performance(eid,ename,des,training,cstatus,cfrom,cto,paid,reimbursed,addr)
    values(?,?,?,?,?,?,?,?,?,?,?)";

PreparedStatement ps=con.prepareStatement(str);
ps.setString(1,t_eid.getText());
ps.setString(2,t_ename.getText());
ps.setString(3,""+cmbdes.getSelectedItem());
ps.setString(4,t_training.getText());
ps.setString(5,""+cmbstatus.getSelectedItem());
ps.setString(6,from);

    String
    to=cmbtod.getSelectedItem()+"/"+cmbtom.getSelectedItem()+"/"+cmbtoy.get
    SelectedItem();

ps.setString(7,to);
ps.setString(8,t_paid.getText());
ps.setString(9,t_reimbursed.getText());
ps.setString(10,t_addr.getText());
ps.executeUpdate();
}
}
catch(Exception ex)
{
System.out.print(ex);
}
}
if(e.getSource()==bcancel)
{

```

```

this.dispose();
}
}

public void focusLost(FocusEvent f)
{
if(f.getComponent()==t_eid)
{
try
{
Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
Statement st=con.createStatement();

        ResultSet res=st.executeQuery("select * from emp where
        eid='"+t_eid.getText()+"'");
while(res.next())
{
t_ename.setText(res.getString(2));
t_addr.setText(res.getString(8));
}
}
catch(Exception ex)
{
System.out.print(ex);
}
}

public void focusGained(FocusEvent f)
{

}

public static void main(String args[])throws IOException
{
performance obj=new performance();

```

```
}
}
```

8. Remuneration Form:

```
import java.io.*;
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
import java.sql.*;

public class remuneration extends JFrame implements
ActionListener,ItemListener,FocusListener
{
String dept[]={"Select
Department","Sales","Analyst","Purchase","EDP","Developer","Programmer","Desig
ner"};

JComboBox cmbdpt=new JComboBox(dept);
JTextField t_da=new JTextField();
JTextField t_hra=new JTextField();
JTextField t_wa=new JTextField();
JTextField t_gpf=new JTextField();
JTextField t_it=new JTextField();
```

```
JTextField t_gs=new JTextField();
JTextField t_pf=new JTextField();
JTextField t_lic=new JTextField();
JTextField t_cas=new JTextField();
JTextField t_sick=new JTextField();
JTextField t_med=new JTextField();
JTextField t_ann=new JTextField();
JButton bnew=new JButton();
JButton bsave=new JButton();
JButton bedit=new JButton();
JButton bsearch=new JButton();
JButton bcancel=new JButton();
public remuneration()
{
    add(cmbdpt);
    add(t_da);
    add(t_hra);
    add(t_wa);
    add(t_gpf);
    add(t_it);
    add(t_gs);
    add(t_pf);
    add(t_lic);
    add(t_cas);
    add(t_sick);
    add(t_med);
    add(t_ann);
    add(bnew);
    add(bsave);
    add(bsearch);
    add(bedit);
```

```
add(bcancel);
bnew.addActionListener(this);
bsave.addActionListener(this);
bsearch.addActionListener(this);
bedit.addActionListener(this);
bcancel.addActionListener(this);
t_med.addFocusListener(this);
t_ann.addFocusListener(this);
t_sick.addFocusListener(this);
cmbdpt.addItemListener(this);
cmbdpt.setBounds(833,550,181,33);
t_da.setBounds(826,662,181,33);
t_hra.setBounds(825,720,181,33);
t_wa.setBounds(825,774,181,33);
t_gpf.setBounds(1200,662,181,33);
t_it.setBounds(1200,718,181,33);
t_gs.setBounds(1202,773,181,33);
t_pf.setBounds(1200,828,181,33);
t_lic.setBounds(1200,881,181,33);
t_cas.setBounds(1639,662,181,33);
t_sick.setBounds(1639,718,181,33);
t_med.setBounds(1639,773,181,33);
t_ann.setBounds(1639,827,181,33);
bnew.setBounds(813,1005,123,42);
bsave.setBounds(980,1004,123,42);
bsearch.setBounds(1148,1003,123,42);
bedit.setBounds(1319,1002,123,42);
bcancel.setBounds(1485,1001,123,42);
}
public void actionPerformed(ActionEvent e)
{
```

```
if(e.getSource()==bnew)
{
try
{
t_da.setText("");
t_hra.setText("");
t_wa.setText("");
t_gpf.setText("");
t_it.setText("");
t_gs.setText("");
t_pf.setText("");
t_lic.setText("");
t_cas.setText("");
t_sick.setText("");
t_med.setText("");
t_ann.setText("");
}
catch(Exception ex)
{
System.out.print(ex);
}
}
if(e.getSource()==bsearch)
{
try
{
bedit.setEnabled(true);
int flg=0;
Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
Statement st=con.createStatement();
String tempcode=JOptionPane.showInputDialog(null,"Enter Employee Type:");
```

```

ResultSet res=st.executeQuery("Select * from remuneration where
Emp_Type='"+tempcode+"'");
while(res.next())
{
cmbdpt.setSelectedItem(res.getString(1));
t_hra.setText(res.getString(2));
        t_da.setText(res.getString(3));
        t_wa.setText(res.getString(4));
        t_gpf.setText(res.getString(5));
        t_it.setText(res.getString(6));
        t_gs.setText(res.getString(7));
        t_pf.setText(res.getString(8));
        t_lic.setText(res.getString(9));
        t_cas.setText(res.getString(10));
        t_sick.setText(res.getString(11));
        t_med.setText(res.getString(12));
        t_ann.setText(res.getString(13));
        flg=1;
    }
    if(flg==0)
JOptionPane.showMessageDialog(null,"No such Record present");
    }
    catch(Exception ee)
    {
System.out.println(ee);
    }
}
if(e.getSource()==bcancel)
{
        this.dispose();
    }
}

```

```

if(e.getSource()==bedit)
{
try
{
Connection con;
con=DriverManager.getConnection("jdbc:odbc:hrdsn");
ResultSet res;
Statement st=con.createStatement();
res=st.executeQuery("select * from remuneration");
String Str="update remuneration set
hra="+Integer.parseInt(t_hra.getText())+",da="+Integer.parseInt(t_da.getText())+",w
a="+Integer.parseInt(t_wa.getText())+",gpf="+Integer.parseInt(t_gpf.getText())+",it=
"+Integer.parseInt(t_it.getText())+",gis="+Integer.parseInt(t_gs.getText())+",pf="+Int
eger.parseInt(t_pf.getText())+",lic="+Integer.parseInt(t_lic.getText())+",casual="+Int
eger.parseInt(t_cas.getText())+",sick="+Integer.parseInt(t_sick.getText())+",medical
="+Integer.parseInt(t_med.getText())+",annual="+Integer.parseInt(t_ann.getText())
+" where Emp_Type="+cmbdpt.getSelectedItem()+"";
st.executeUpdate(Str);
JOptionPane.showMessageDialog(null,"Updated Successfully");
}
}

        catch(Exception ex)
        {
System.out.println(ex);
        }
    }
    if(e.getSource()==bsave)
    {
try
{
Connection con;
con=DriverManager.getConnection("jdbc:odbc:hrdsn");
Statement st=con.createStatement();

```



```
String str="insert into
remuneration(Emp_Type,Hra,Da,Wa,Gpf,It,Gis,Pf,Lic,casual,sick,medi
cal,annual)values(?,?,?,?,?,?,?,?,?,?,?,?,?)";
```

```
PreparedStatement ps=con.prepareStatement(str);
ps.setString(1,""+cmbdpt.getSelectedItem());
ps.setString(2,""+t_hra.getText());
ps.setString(3,t_da.getText());
ps.setString(4,t_wa.getText());
ps.setString(5,t_gpf.getText());
ps.setString(6,t_it.getText());
ps.setString(7,t_gs.getText());
ps.setString(8,t_pf.getText());
ps.setString(9,t_lic.getText());
ps.setString(10,t_cas.getText());
ps.setString(11,t_sick.getText());
ps.setString(12,t_med.getText());
ps.setString(13,t_ann.getText());
ps.executeUpdate();
JOptionPane.showMessageDialog(null,"Saved Successfully");
}
}
catch(Exception ee)
{
System.out.println(ee);
}
}
}
public void itemStateChanged(ItemEvent ie)
{
if(ie.getSource()==cmbdpt)
{
try
```

```

{
Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
Statement st=con.createStatement();
String tempcode="" +cmbdpt.getSelectedItem();
ResultSet res=st.executeQuery("Select * from remuneration where
Emp_Type='"+tempcode+"'");
while(res.next())
{
t_hra.setText(res.getString(2));
        t_da.setText(res.getString(3));
        t_wa.setText(res.getString(4));
        t_gpf.setText(res.getString(5));
        t_it.setText(res.getString(6));
        t_gs.setText(res.getString(7));
        t_pf.setText(res.getString(8));
        t_lic.setText(res.getString(9));
        t_cas.setText(res.getString(10));
        t_sick.setText(res.getString(11));
        t_med.setText(res.getString(12));
        t_ann.setText(res.getString(13));
catch(Exception e)
{
System.out.println(e);
}
}
public void focusGained(FocusEvent f)
{
}

    public static void main(String argv[])throws IOException
    {
remuneration obj=new remuneration();

```

```
}
}
```

9. Attendance Form:

The screenshot shows a Java Swing window titled "TIMEOUT". It features a "Date" text field containing "2021/9/30" and a "Department" dropdown menu with "SELECT". Below these are two tables. The first table has columns "ID" and "NAME". The second table has columns "ID", "NAME", and "TIME OUT". A "Timeout" button is located between the two tables. At the bottom of the window are three buttons: "Refresh", "Save", and "Cancel".

```
import java.io.*;
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
import java.sql.*;
import javax.swing.table.DefaultTableModel;
public class attendance extends JFrame implements ActionListener,ItemListener
{
    JTextField t_date=new JTextField();
    JButton brefresh=new JButton();
    JButton bsave=new JButton();
    JButton bcancel=new JButton();
    JButton bright=new JButton();
    JButton btimeout=new JButton();
    String dept[]={"--SELECT--"};
```

```

JComboBox cmbdept=new JComboBox(dept);
String col1[]={ "ID","NAME","STATUS"};
Object[][] data1={{ "l_eid","l_ename","l_status"}};
DefaultTableModel model1=new DefaultTableModel(col1,0);
JTable tbl1;
JScrollPane scrln1;
String col2[]={ "ID","NAME","TIME IN"};
Object[][] data2={{ "r_eid","r_ename","time_in"}};
DefaultTableModel model2=new DefaultTableModel(col2,0);
JTable tbl2;
JScrollPane scrln2;
String l_eid,l_ename,l_status,r_eid,r_ename,time_in;
public attendance()
{
try
{
Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
Statement st=con.createStatement();
ResultSet res=st.executeQuery("select distinct dept from emp");
while(res.next())
{
cmbdept.addItem(res.getString(1));
}
}
catch(Exception e)
{
System.out.println(e);
}
add(t_date);
add(cmbdept);
add(brefresh);

```

```

add(bsave);
add(bcancel);
add(bright);
add(btimeout);
brefresh.addActionListener(this);
bsave.addActionListener(this);
bcancel.addActionListener(this);
bright.addActionListener(this);
btimeout.addActionListener(this);
cmbdept.addItemListener(this);
t_date.setBounds(820,555,181,33);
cmbdept.setBounds(1525,555,181,33);
bright.setBounds(1135,760,142,41);
btimeout.setBounds(1135,845,142,41);
brefresh.setBounds(920,1056,125,41);
bsave.setBounds(1132,1056,125,41);
bcancel.setBounds(1334,1056,125,41);
Calendar cal=Calendar.getInstance();
String
sdate=cal.get(Calendar.DATE)+"/"+cal.get(Calendar.MONTH)+"/"+cal.get(Calendar
.YEAR);
t_date.setText(sdate);
tbl1=new JTable(data1,col1);
scrln1=new JScrollPane(tbl1);
add(scrln1);
scrln1.setBounds(674,710,433,215);
tbl2=new JTable(data2,col2);
scrln2=new JScrollPane(tbl2);
add(scrln2);
scrln2.setBounds(1308,710,456,215);
tbl2.setModel(model2);
}

```

```
public void itemStateChanged(ItemEvent ie)
{
    if(ie.getSource()==cmbdept)
    {
        try
        {
            Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
            Statement st=con.createStatement();
            ResultSet res=st.executeQuery("select * from emp where
            dept='"+cmbdept.getSelectedItem()+"");
            model1.setRowCount(0);
            model2.setRowCount(0);
            while(res.next())
            {
                Vector v=new Vector();
                v.add(res.getString(1));
                v.add(res.getString(2));
                model1.addRow(v);
            }
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
    }
}

public void actionPerformed(ActionEvent e)
{
    if(e.getSource()==brefresh)
    {
        cmbdept.setSelectedIndex(0);
        model1.setRowCount(0);
    }
}
```

```

model2.setRowCount(0);
}
if(e.getSource()==bsave)
{
try
{
Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
String str="select * from holiday where hdate='"+t_date.getText()+"'";
Statement st=con.createStatement();
ResultSet res=st.executeQuery(str);
while(res.next())
{}
else
{
String str2="insert into
attendance(cdate,dept,eid,ename,time_in)values(?,?,?,?,?)";
PreparedStatement ps=con.prepareStatement(str2);
for(int i=0;i<tbl2.getRowCount();i++)
{
ps.setString(1,t_date.getText());
ps.setString(2,""+cmbdept.getSelectedItem());
ps.setString(3,""+tbl1.getValueAt(i,0));
ps.setString(4,""+tbl1.getValueAt(i,1));
ps.setString(5,""+tbl2.getValueAt(i,2));
ps.executeUpdate();
}
JOptionPane.showMessageDialog(null,"Submitted!");
}
}
}
catch(Exception ex)

```

```

{
    System.out.print(ex);
}
}
if(e.getSource()==bcancel)
{
    this.dispose();
}
if(e.getSource()==bright)
{
    int k=tbl1.getSelectedRow();
    if((flag==1)|| (tbl2.getRowCount()==0))
    {
        Calendar cal=Calendar.getInstance();
        String
        tin=cal.get(Calendar.HOUR)+":"+cal.get(Calendar.MINUTE)+":"+cal.get(Calendar.S
        ECOND);
        Vector v2=new Vector();
        v2.add(tbl1.getValueAt(k,0));
        v2.add(tbl1.getValueAt(k,1));
        v2.add(tin);
        model2.addRow(v2);
        tbl1.setValueAt("PRESENT",k,2);
    }
}
if(e.getSource()==btimeout)
{
    new timeout();
}
}
public static void main(String args[])throws IOException
{

```



```
attendance obj=new attendance();
}
}
```

10.Timeout Form:

The screenshot shows a Java Swing window titled "TIMEOUT". It features a "Date" text field containing "2021/9/30" and a "Department" dropdown menu with "SELECT". Below these are two empty tables. The first table has columns "ID" and "NAME". The second table has columns "ID", "NAME", and "TIME OUT". A "Timeout" button is located between the two tables. At the bottom of the window are three buttons: "Refresh", "Save", and "Cancel".

```
import java.io.*;
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
import java.sql.*;
import javax.swing.table.DefaultTableModel;
public class timeout extends JFrame implements ActionListener,ItemListener
{
    JTextField t_date=new JTextField();
    JButton brefresh=new JButton();
    JButton bsave=new JButton();
    JButton bcancel=new JButton();
    JButton bleft=new JButton();
    String des[]={"--SELECT--"};
```

```

JComboBox cmbdept=new JComboBox(des);
String col1[]={ "ID","NAME"};
Object[][] data1={{ "l_eid","l_ename"}};
DefaultTableModel model1=new DefaultTableModel(col1,0);
JTable tbl1;
JScrollPane scrln1;
String col2[]={ "ID","NAME","TIME OUT"};
Object[][] data2={{ "r_eid","r_ename","time_out"}};
DefaultTableModel model2=new DefaultTableModel(col2,0);
JTable tbl2;
JScrollPane scrln2;
String l_eid,l_ename,r_eid,r_ename,time_out;
public timeout()
{
try
{
Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
Statement st=con.createStatement();
ResultSet res=st.executeQuery("select distinct dept from emp");
while(res.next())
{
cmbdept.addItem(res.getString(1));
}
}
catch(Exception e)
{
System.out.println(e);
}
add(t_date);
add(cmbdept);
add(brefresh);

```

```

add(bsave);
add(bcancel);
add(bleft);
t_date.setBounds(820,555,181,33);
cmbdept.setBounds(1525,555,181,33);
brefresh.setBounds(920,1056,125,41);
Calendar cal=Calendar.getInstance();
String
sdate=cal.get(Calendar.DATE)+"/"+cal.get(Calendar.MONTH)+"/"+cal.get(Calendar
.YEAR);
t_date.setText(sdate);
tbl1=new JTable(data1,col1);
scripn1=new JScrollPane(tbl1);
add(scripn1);
scripn1.setBounds(708,710,312,215);
tbl1.setModel(model1);
tbl2=new JTable(data2,col2);
scripn2=new JScrollPane(tbl2);
add(scripn2);
scripn2.setBounds(1308,710,456,215);
tbl2.setModel(model2);
}
public void itemStateChanged(ItemEvent ie)
{
if(ie.getSource()==cmbdept)
{
try
{
Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
Statement st=con.createStatement();
ResultSet res=st.executeQuery("select * from attendance where
dept='"+cmbdept.getSelectedItem()+" and time_out is null");

```

```

model1.setRowCount(0);
model2.setRowCount(0);
while(res.next())
{
    Vector v=new Vector();
    v.add(res.getString(3));
    v.add(res.getString(4));
    model1.addRow(v);
}
}
catch(Exception e)
{
    System.out.println(e);
}
}
public void actionPerformed(ActionEvent e)
{
    if(e.getSource()==bsave)
    {
        try
        {
            Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
            Statement st=con.createStatement();
            for(int i=0;i<tbl1.getRowCount();i++)
            {
                String t=""+tbl2.getValueAt(i,2);
                String str="Update attendance set time_out='"+tbl2.getValueAt(i,2)+"' where
eid='"+tbl1.getValueAt(i,0)+"' and time_out is null";
                st.executeUpdate(str);
            }
        }
    }
}

```

```

JOptionPane.showMessageDialog(null,"RECORD UPDATED");
}
}
catch(Exception ex)
{
System.out.print(ex);
}
if(e.getSource()==bcancel)
{
this.dispose();
}
if(e.getSource()==bleft)
{
int k=tbl1.getSelectedRow();
for(int j=0;j<tbl1.getRowCount();j++)
{
Calendar cal=Calendar.getInstance();

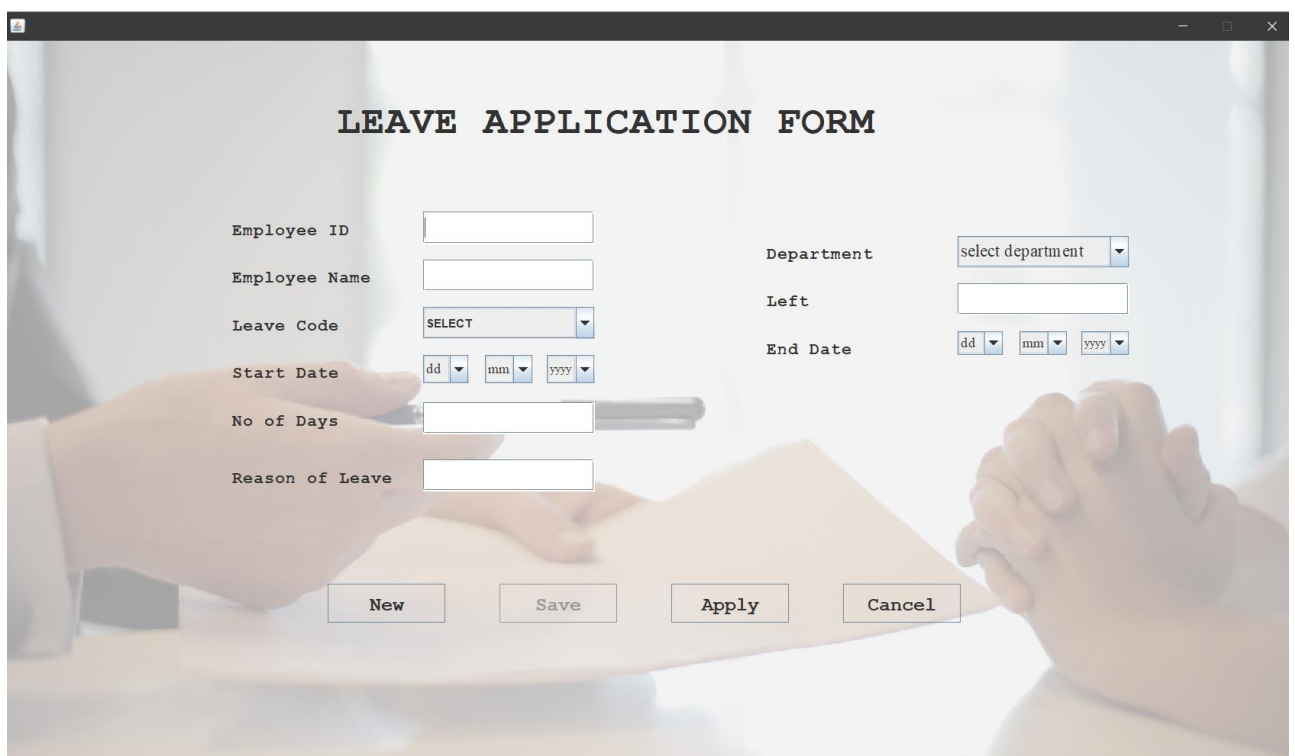
String
tout=cal.get(Calendar.HOUR)+":"+cal.get(Calendar.MINUTE)+":"+cal.get(Calendar.
SECOND);
Vector v2=new Vector();
v2.add(tbl1.getValueAt(k,0));
v2.add(tbl1.getValueAt(k,1));
v2.add(tout);
model2.addRow(v2);
}
}
}

public static void main(String args[])throws IOException
{
timeout obj=new timeout();

```

```
}  
}
```

11. Leave Form:



The screenshot displays a web-based "LEAVE APPLICATION FORM" interface. The form is set against a background image of hands holding a pen over a document. The title "LEAVE APPLICATION FORM" is centered at the top in a bold, monospace font. Below the title, the form is organized into two columns of input fields. The left column contains: "Employee ID" (text input), "Employee Name" (text input), "Leave Code" (dropdown menu with "SELECT" as the current value), "Start Date" (date picker with "dd", "mm", and "yyyy" dropdowns), "No of Days" (text input), and "Reason of Leave" (text input). The right column contains: "Department" (dropdown menu with "select department" as the current value), "Left" (text input), and "End Date" (date picker with "dd", "mm", and "yyyy" dropdowns). At the bottom of the form, there are four buttons: "New", "Save", "Apply", and "Cancel", each enclosed in a light gray rectangular box.

LEAVE APPLICATION FORM			
Employee ID	<input type="text"/>	Department	<input type="text" value="select department"/>
Employee Name	<input type="text"/>	Left	<input type="text"/>
Leave Code	<input type="text" value="SELECT"/>	End Date	<input type="text" value="dd mm yyyy"/>
Start Date	<input type="text" value="dd mm yyyy"/>		
No of Days	<input type="text"/>		
Reason of Leave	<input type="text"/>		
<div><input type="button" value="New"/> <input type="button" value="Save"/> <input type="button" value="Apply"/> <input type="button" value="Cancel"/></div>			

```

import java.io.*;
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
import javax.swing.*;
import java.sql.*;

public class leave extends JFrame implements
ActionListener,FocusListener,ItemListener
{
    JTextField t_eid=new JTextField();
    JTextField t_ename=new JTextField();
    JTextField t_no=new JTextField();
    JTextField t_rea=new JTextField();
    JTextField t_left=new JTextField();
    String leave_code[]={"SELECT","CASUAL","SICK","MEDICAL","ANNUAL"};
    JComboBox cmbcode=new JComboBox(leave_code);
    JComboBox cmbsd=new JComboBox();
    JComboBox cmbsm=new JComboBox();
    JComboBox cmbsy=new JComboBox();
    JComboBox cmbed=new JComboBox();
    JComboBox cmbem=new JComboBox();
    JComboBox cmbey=new JComboBox();
    String des[]={"select
department","Sales","Purchase","EDP","Developer","Programmer","Designer"};
    JComboBox cmbdpt=new JComboBox(des);
    JButton bnew=new JButton();
    JButton bsave=new JButton();
    JButton bapply=new JButton();
    JButton bcancel=new JButton();
    public leave()
    {
        for(int d=1;d<=31;d++)

```

```
{
cmbbsd.addItem(""+d);
}
for(int m=1;m<=12;m++)
{
cmbsm.addItem(""+m);
}
for(int y=1995;y<=2019;y++)
{
cmbsy.addItem(""+y);
}
for(int d=1;d<=31;d++)
{
cmbed.addItem(""+d);
}
for(int m=1;m<=12;m++)
{
cmbem.addItem(""+m);
}
for(int y=2010;y<=2019;y++)
{
cmbey.addItem(""+y);
}
add(t_eid);
add(t_ename);
add(t_no);
add(t_rea);
add(t_left);
add(cmbcode);
add(cmbdpt);
add(cmbbsd);
```



```
add(cmbsm);
add(cmbsy);
add(cmbed);
add(cmbem);
add(cmbey);
add(bnew);
add(bsave);
add(bapply);
add(bcancel);
bnew.addActionListener(this);
bsave.addActionListener(this);
bapply.addActionListener(this);
bcancel.addActionListener(this);
t_eid.addFocusListener(this);
t_ename.addFocusListener(this);
t_left.addFocusListener(this);
t_no.addFocusListener(this);
t_rea.addFocusListener(this);
cmbcode.addItemListener(this);
t_eid.setBounds(902,567,180,34);
t_ename.setBounds(902,630,180,33);
t_no.setBounds(902,817,180,33);
t_rea.setBounds(902,878,180,33);
t_left.setBounds(1525,689,180,33);
cmbcode.setBounds(902,692,180,33);
cmbdpt.setBounds(1525,566,180,33);
cmbsd.setBounds(901,753,60,35);
cmbsm.setBounds(973,753,61,35);
cmbsy.setBounds(1047,753,69,35);
cmbed.setBounds(1524,752,60,35);
cmbem.setBounds(1597,752,61,35);
```

```

cmbey.setBounds(1671,752,69,35);
bnew.setBounds(923,1002,124,41);
bsave.setBounds(1127,1002,124,41);
bapply.setBounds(1319,1002,124,41);
bcancel.setBounds(1521,1002,124,41);
}
public void actionPerformed(ActionEvent e)
{
if(e.getSource()==bnew)
{
try
{
Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
String str="select * from emp";
PreparedStatement ps=con.prepareStatement(str);
Statement st=con.createStatement();
ResultSet res=st.executeQuery(str);
while(res.next())
{
ctr++;
}
if(ctr>0&&ctr<=9)
{
t_eid.setText("E00");
}
else if(ctr>=10&&ctr<=99)
{
t_eid.setText("E0");
}
else if(ctr>=100)
{

```

```

t_eid.setText("E");
}
t_ename.setText("");
t_no.setText("");
t_rea.setText("");
cmbcode.setSelectedIndex(0);
cmbdpt.setSelectedIndex(0);
cmbsd.setSelectedIndex(0);
cmbsm.setSelectedIndex(0);
cmbsy.setSelectedIndex(0);
cmbed.setSelectedIndex(0);
cmbem.setSelectedIndex(0);
cmbey.setSelectedIndex(0);
}
catch(Exception ex)
{
System.out.print(ex);
}
}
if(e.getSource()==bsave)
{
try
{
Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
String str="Insert into
leave(eid,ename,des,leave_code,start_date,end_date,no_of_days,reason)va
lues(?,?,?,?,?,?,?,?)";
PreparedStatement ps=con.prepareStatement(str);
ps.setString(1,t_eid.getText());
ps.setString(2,t_ename.getText());
ps.setString(3,""+cmbdpt.getSelectedItem());
ps.setString(4,""+cmbcode.getSelectedItem());

```

```

        String
        start=cmbbsd.getSelectedItemId()+"/"+cmbbsm.getSelectedItemId()+"/"+cmbbsy.getSelectedItemId();
ps.setString(5,start);

        String
        end=cmbbed.getSelectedItemId()+"/"+cmbbem.getSelectedItemId()+"/"+cmbbey.getSelectedItemId();
ps.setString(6,end);
ps.setString(7,t_no.getText());
ps.setString(8,t_rea.getText());
ps.executeUpdate();
JOptionPane.showMessageDialog(null,"Saved");
}
}
catch(Exception ex)
{
    System.out.print(ex);
}
}
if(e.getSource()==bapply)
{
    try
    {
        Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
        Statement st=con.createStatement();

        ResultSet res=st.executeQuery("select * from emp where
        eid='"+t_eid.getText()+"'");
        int l=Integer.parseInt(t_left.getText());
        String ltype="";
        String code=cmbcode.getSelectedItemId()+"";
        while(res.next())
        {

```

```
if(code.equals("CASUAL"))
{
ltype=res.getString(24);
}
else if(code.equals("SICK"))
{
ltype=res.getString(25);
}
else if(code.equals("MEDICAL"))
{
ltype=res.getString(26);
}
else
{
ltype=res.getString(27);
}
}
int a=Integer.parseInt(ltype);
int b=Integer.parseInt(t_no.getText());
int c;
c=a-b;
if(a<=b)
{}
else
{
String str="Update emp set "+cmbcode.getSelectedItem()+"="+c+" where
eid='"+t_eid.getText()+"'";
st.executeUpdate(str);
}
}
catch(Exception ex)
```

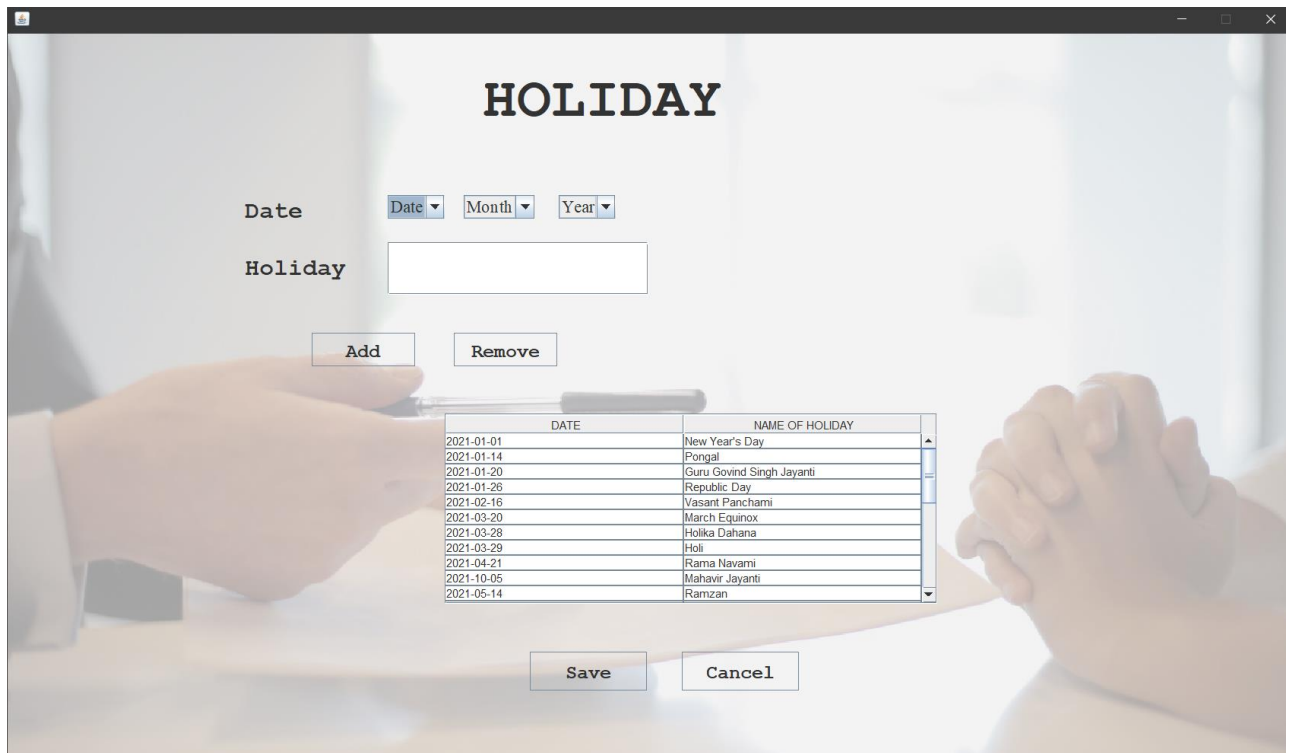
```

{
System.out.print(ex);
}
}
if(e.getSource()==bcancel)
{
this.dispose();
}
}
public void focusLost(FocusEvent f)
{
if(f.getComponent()==t_ename)
{
String s=t_ename.getText();
if(s.equals(""))
{
}
else
{
for( int i=0;i<s.length();i++)
{
if(s.charAt(i)>='a' && s.charAt(i)<='z' || s.charAt(i)>='A' &&
s.charAt(i)<='Z' || s.charAt(i)==' ')
{
}
}
}
public void itemStateChanged(ItemEvent ie)
{
if(ie.getSource()==cmbcode)
{
try

```

```
{
Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
Statement st=con.createStatement();
ResultSet res=st.executeQuery("select "+cmbcode.getSelectedItemId()+" from emp
where eid='"+t_eid.getText()+"'");
while(res.next())
{
t_left.setText(res.getString(1));
}
}
catch(Exception e)
{
System.out.println(e);
}
}
public static void main(String args[])throws IOException
{
leave obj=new leave();
}
}
```

12.Holiday Form:



```

import java.io.*;
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
import java.sql.*;
import javax.swing.table.DefaultTableModel;
public class holiday extends JFrame implements ActionListener
{
    JTextField thol=new JTextField();
    JButton badd=new JButton();
    JButton brem=new JButton();
    JButton bsave=new JButton();
    JButton bcancel=new JButton();
    JComboBox cmbyear=new JComboBox();
    JComboBox cmbd=new JComboBox();
    JComboBox cmbm=new JComboBox();
    String col1[]={"DATE","NAME OF HOLIDAY"};
    Object[][] data1={{"hdate","hname"}};
    DefaultTableModel model1=new DefaultTableModel(col1,0);

```



```
JTable tbl1;  
JScrollPane scrpn1;  
String hdate,hname;  
public holiday()  
{cmbd.addItem("--D--");  
for(int d=1;d<=31;d++)  
{  
cmbd.addItem(""+d);  
}  
  
cmbm.addItem("--M--");  
for(int d=1;d<=12;d++)  
{  
cmbm.addItem(""+d);  
}  
cmbyear.addItem("--Y--");  
for(int d=1999;d<=2050;d++)  
{  
cmbyear.addItem(""+d);  
}  
add(thol);  
add(badd);  
add(brem);  
add(bsave);  
add(bcancel);  
add(cmbd);  
add(cmbm);  
add(cmbyear);  
badd.addActionListener(this);  
brem.addActionListener(this);  
bsave.addActionListener(this);
```

```

bcancel.addActionListener(this);
cmbd.setBounds(819,493,60,35);
cmbm.setBounds(891,493,61,35);
cmbyear.setBounds(965,493,69,35);
thol.setBounds(819,558,275,55);
badd.setBounds(817,635,109,41);
brem.setBounds(975,635,109,41);
bsave.setBounds(1109,1074,123,41);
bcancel.setBounds(1311,1075,123,41);
tbl1=new JTable(data1,col1);
scripn1=new JScrollPane(tbl1);
add(scripn1);
scripn1.setBounds(1000,720,519,320);
tbl1.setModel(model1);
thol.setFont(new Font("Adobe Fan Heiti Std B",Font.PLAIN,18));
}public void actionPerformed(ActionEvent e)
{
if(e.getSource()==badd)
{
try
{
Vector v1=new Vector();
String
d=cmbd.getSelectedItem()+"/"+cmbm.getSelectedItem()+"/"+cmbyear.getSelectedItem();
v1.add(d);
v1.add(thol.getText());
model1.addRow(v1);
}
catch(Exception ex)
{
System.out.print(ex);

```

```
}  
}  
if(e.getSource()==brem)  
{  
    try  
    {  
        model1.removeRow(tbl1.getSelectedRow());  
    }  
    catch(Exception ex)  
    {  
        System.out.print(ex);  
    }  
}  
if(e.getSource()==bsave)  
{  
    try  
    {  
        Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");  
        String str="Insert into holiday(hdate,hname)values(?,?)";  
        PreparedStatement ps=con.prepareStatement(str);  
        for(int i=0;i<tbl1.getRowCount();i++)  
        {  
            ps.setString(1,""+tbl1.getValueAt(i,0));  
            ps.setString(2,""+tbl1.getValueAt(i,1));  
            ps.executeUpdate();  
        }  
        JOptionPane.showMessageDialog(null,"Saved!");  
    }  
}  
catch(Exception ex)  
{
```

```

System.out.print(ex);
}
}
if(e.getSource()==bcancel)
{
this.dispose();
}
}
public void focusLost(FocusEvent f)
{
if(f.getComponent()==badd)
{
try
{
thol.setText("");
}
catch(Exception ex)
{
System.out.print(ex);
}
}
}
public void focusGained(FocusEvent f)
{
}
public static void main(String args[])throws IOException
{
holiday obj=new holiday();
}
}

```

13.Pay Slip Form:

```

import java.awt.*;
import java.awt.event.*;
import java.applet.*;
import java.io.*;
import java.sql.*;

public class payslip extends JFrame implements ActionListener,FocusListener
{
    String dpt[]={ " Select
Department","Sales","Purchase","EDP","Developer","Programmer","Designer"};

    JComboBox cmbdept=new JComboBox(dpt);

    JTextField teid=new JTextField();

    JTextField tname=new JTextField();

    JTextField tbasic=new JTextField();

    JComboBox cmbdobd=new JComboBox();

    String mon[]={ "--SELECT--
","JANUARY","FEBRUARY","MARCH","APRIL","MAY","JUNE","JULY","AUGUST","
SEPTEMBER","OCTOBER","NOVEMBER","DECEMBER"};

```

```
JComboBox cmbmonth=new JComboBox(mon);
JButton bnew=new JButton();
JButton bsearch=new JButton();
JButton bok=new JButton();
JButton bcancel=new JButton();
public payslip()
{
    for(int d=1;d<=31;d++)
    {
        cmbdobd.addItem(""+d);
    }
    for(int m=1;m<=12;m++)
    {
        cmbdobm.addItem(""+m);
    }
    for(int y=2019 ;y>=1850;y--)
    {
        cmbdoby.addItem(""+y);
    }
    add(teid);
    add(tname);
    add(cmbmonth);
    add(tbasic);
    add(cmbdoby);
    add(cmbdobm);
    add(cmbdobd);
    add(cmbdept);
```

```
add(bnew);
add(bsearch);
add(bok);
add(bcancel);
add(l_back);
teid.setBounds(210,154,170,35);
cmbmonth.setBounds(210,270,100,35);
tbasic.setBounds(210,330,170,35);
tname.setBounds(670,150,170,35);
teid.setBounds(923,600,180,34);
tname.setBounds(923,692,180,33);
tbasic.setBounds(923,870,180,34);
cmbdobd.setBounds(1562,691,61,35);
cmbdobm.setBounds(1634,691,61,35);
cmbdoby.setBounds(1708,691,69,35);
cmbmonth.setBounds(923,777,180,34);
cmbdept.setBounds(1560,601,180,34);
bnew.setBounds(834,1039,125,44);
bcancel.setBounds(1459,1039,125,44);
bnew.addActionListener(this);
bsearch.addActionListener(this);
bok.addActionListener(this);
bcancel.addActionListener(this);
teid.addFocusListener(this);
tname.addFocusListener(this);
tbasic.addFocusListener(this);
teid.setFont(new Font("Adobe Fan Heiti Std B",Font.PLAIN,18));
```

```
tname.setFont(new Font("Adobe Fan Heiti Std B",Font.PLAIN,18));
tbasic.setFont(new Font("Adobe Fan Heiti Std B",Font.PLAIN,18));
}
public void actionPerformed(ActionEvent e)
{
    if(e.getSource()==bnew)
    {
        try
        {
            int ctr=1;
            Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
            String str="select * from emp";
            PreparedStatement ps=con.prepareStatement(str);
            Statement st=con.createStatement();
            ResultSet res=st.executeQuery(str);
            while(res.next())
            {
                ctr++;
            }
            if(ctr>0&&ctr<=9)
            {
                t_eid.setText("E00");
            }
            else if(ctr>=10&&ctr<=99)
            {
                t_eid.setText("E0");
            }
        }
    }
}
```



```

        else if(ctr>=100)
        {
            t_eid.setText("E");
        }
        tname.setText("");
        tbasic.setText("");
        cmbdept.setSelectedIndex(0);
        cmbdobd.setSelectedIndex(0);
        cmbdobm.setSelectedIndex(0);
        cmbdoby.setSelectedIndex(0);
        cmbmonth.setSelectedIndex(0);
    }
    catch(Exception ex)
    {
        System.out.print(ex);
    }
}
if(e.getSource()==bsearch)
{ try
    {
        Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
        Statement st=con.createStatement();
        String eid=JOptionPane.showInputDialog(null,"Enter Employee Type:");
        ResultSet res=st.executeQuery("select * from emp where eid='"+eid+"'");
        while(res.next())
        {
            teid.setText(res.getString(1));

```

```

        tname.setText(res.getString(2));
        cmbdept.setSelectedItem(res.getString(3));
        String arr[]=res.getString(5).toString().split("/");
        cmbdobd.setSelectedItem(arr[0]);
        cmbdobm.setSelectedItem(arr[1]);
        cmbdoby.setSelectedItem(arr[2]);
        String year=(arr[2]);
        tbasic.setText(res.getString(13));
    }
}
catch(Exception ex)
{
    System.out.print(ex);
}

if(e.getSource()==bok)
{
    try
    {
        Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
        Statement st=con.createStatement();
        String str="select * from emp where eid='"+teid.getText()+"'";
        ResultSet res=st.executeQuery(str);
        PrintWriter pw=new PrintWriter("report.txt");
        while(res.next())
        {
            da=da*basic/100;
            hra=hra*basic/100;

```

```

wa=wa*basic/100;

totall=totall*basic/100;

gpf=gpf*basic/100;

it=it*basic/100;

gs=gs*basic/100;

pf=pf*basic/100;

lic=lic*basic/100;

int net=totall-totded+basic;

pw.println("          HR MANAGEMENT SYSTEM");
pw.println("          _____");
pw.println();

pw.println("EMPLOYEE CODE:"+res.getString(1)+"
EMPLOYEE NAME:"+res.getString(2));

pw.println("DESIGNATION:"+res.getString(3)+"          FOR
MONTH:"+cmbmonth.getSelectedItem()+"          YEAR:2019");
pw.println("_____");

//pw.println();

pw.println("EARNINGS:  AMOUNT");

pw.println("DA :      "+da);
pw.println("HRA:      "+hra);
pw.println("WA :      "+wa);

pw.println("TOTAL EARNINGS: "+totall);

pw.println("DEDUCTIONS:  AMOUNT");

pw.println("GPF:      "+gpf);
pw.println("IT :      "+it);
pw.println("GS :      "+gs);
pw.println("PF :      "+pf);

```

```

        pw.println("LIC:      "+lic);
        pw.println("TOTAL DEDUCTIONS: "+totded);
        pw.println("_____");
        pw.println("NET SALARY: "+net);
    }
    pw.close();
    Process p=Runtime.getRuntime().exec("Notepad report.txt");
}
}
catch(Exception ee)
{
    System.out.print(e);
}
}
if(e.getSource()==bcancel)
{
    this.dispose();
}
}

public void focusGained(FocusEvent f)
{
}

public void focusLost(FocusEvent f)
{
    if(f.getComponent()==teid)
    {

```

```
try
{
    Connection con=DriverManager.getConnection("jdbc:odbc:hrdsn");
    Statement st=con.createStatement();
    ResultSet res=st.executeQuery("select * from emp where
eid='"+teid.getText()+"'");
    while(res.next())
    {
        tname.setText(res.getString(2));
        cmbdept.setSelectedItem(res.getString(3));

        tbasic.setText(res.getString(13));
    }
}
catch(Exception ex)
{
    System.out.print(ex);
}
if(f.getComponent()==tname)
{
    String s=tname.getText();
}
else
{
    for( int i=0;i<s.length();i++)
    {
```

```

        if(s.charAt(i)>='a' && s.charAt(i)<='z' || s.charAt(i)>='A' &&
s.charAt(i)<='Z' || s.charAt(i)==' ')
    }
}
if(f.getComponent()==tbasic)
{
    String s=tbasic.getText();
    if(s.equals(""))
    {
    }
    else
    {
        for( int i=0;i<s.length();i++)
        {

if(s.charAt(i)=='0' || s.charAt(i)=='1' || s.charAt(i)=='2' || s.charAt(i)=='3' || s.charAt(i)=='4' ||
s.charAt(i)=='5' || s.charAt(i)=='6' || s.charAt(i)=='7' || s.charAt(i)=='8' || s.charAt(i)=='9')

            {
            }

        }

    public static void main(String args[])throws IOException
    {
        payslip obj=new payslip();
    }
}

```

FUTURE IMPROVEMENT

This software manages company's employee details and applications such as attendance details, employee leaves, Employee skills, department and many more.

It provides the Manager with effective tools to keep track of the records and easily manipulate them.

The software has wide scope in the future as its vast calculation of salary earnings and deduction.

In future it can be enhanced with some more features like:

- Distributed systems.
- External modules like TDS.
- Extra modules like overtime.
- Full financial accounting system.
- A complete payroll software.

CONCLUSION

Our project is a humble venture to satisfy the needs in an institution. The system is developed with much care that is free from errors and at the same time, it is efficient and less time consuming. Our aim is to provide a very user friendly human resource management system which is helpful in managing details of employees, their functions and in various other aspects. In this application only one person has been given access rights and it is restricted up to the functionalities, so that the data is maintained securely and redundant data is prevented.

Several user-friendly coding are adopted. The important thing is that the system is robust. Avoid malfunctioning from outsiders. It goes through all phases of software development cycle. So, product is accurate. Also provision is provided for future development in the system.

A special thanks to our mentor for the guidance and blessing which was provided for the completion of our project. We welcome your cooperation and feedback.

BIBLIOGRAPHY

The main aim of developing HUMAN RESOURCE MANAGEMENT SYSTEM is to provide all information that is required by the companies about their employees. User friendliness is a must that is the user must get the details without complicated searching procedures. Other important requirements of software are data security, extensibility and maintainability. All these features are included in this web application. The project greatly helped in understanding the various phases in website development and exposure to a new developer platform JAVA and SQL.

REFERENCE:

BOOKS-

- “SQL, PL/SQL, The Programming Language of Oracle”-Ivan Bayross
- “Data System Concept”-Abraham Silverschatz, Henry K. Forth and D. b Sudarshan

WEBSITES-

- www.slideshare.com
- www.scribd.com
- www.wikipidiea.com
- www.code-project.com

Developed By Rahul Kr Chaudhary