

# **EMPLOYEE MANAGEMENT SYSTEM**

**An Embedded Project Report**

**(19CS391 Programming with JAVA)**

***Submitted by***

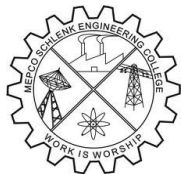
**HARRISH RAGAVENDAR S (202004044)**

***in partial fulfilment for the award of the degree  
of***

**BACHELOR OF ENGINEERING**

**in**

**COMPUTER SCIENCE AND ENGINEERING**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**MEPCO SCHLENK ENGINEERING COLLEGE, SIVAKASI**

**(An Autonomous Institution affiliated to Anna University Chennai)**



**December 2021**



## **BONAFIDE CERTIFICATE**

Certified that this project report titled **EMPLOYEE MANAGEMENT SYSTEM** is the bonafide work of **HARRISH RAGAVENDAR S (202004044)** who carried out project under our supervision in the **19CS391 Programming with JAVA (Embedded Project)**.

**Dr.K.Thirumoorthy , M.E.,Ph.D**

Course Instructor

Assistant Professor (Sr.Grade),

Department of Computer Science &Engg,

Mepco Schlenk Engineering College,

Sivakasi.

**Dr. J. Raja Sekar, M.E.,Ph.D.**

Head of the Department

Professor,

Department of Computer Science & Engg,

Mepco Schlenk Engineering College,

Sivakasi.

Submitted for Viva-Voce Examination held at **MEPCO SCHLENK ENGINEERING COLLEGE, SIVAKASI**

**(AUTONOMOUS)** on .....

**Internal Examiner**

**External Examiner**



## **ABSTRACT**

Employees are the backbone of any company therefore their management plays a major role in deciding the success of an organization. Employee Management Software makes it easy for the employer to keep track of all records. In this world of growing technologies everything is being computerized. With large number of work opportunities, human workforce is increasing day by day. Thus there is a need of a system which can handle the data of such a large number of employees in an organization. This project simplifies the task of maintaining large employee records. This project allows only the administrator to add new employees after proper authentication. All the details of the employees will be stored properly in a database. It is an application developed in Java GUI technology and database used is MySQL. It contains employee information like Employee ID, First name, Surname, and Age. It is an easy to use application and has a user-friendly interface. It is totally built at the administrative end which means that only the admin has access rights to change or modify any records. So this makes it safe and reliable application to use. The main aim of developing this application was to reduce the errors that occur in the manual system. One can search the details easily by just entering Employee ID. In earlier systems, there was not such a facility to do so. All the details are stored in an MySQL database. It is easy to update any employee details. All the employee records are integrated and so this makes it user-friendly and easy to use application. Employee Management System Project is developed completely using Java SWING, solely using Notepad++ without the use of an IDE. Moreover, this system is designed for the particular need of the company to carry out operations in a smooth and effective manner. No formal technical knowledge is needed for the operator to use this system. Thus by all these features, the product will be an user-friendly one.

## ACKNOWLEDGEMENT

First of all I thank **LORD ALMIGHTY** for his grace and abundant blessings on me to make this work a great success.

I sincerely thank our respected principal **Dr. S. ARIVAZHAGAN**, for providing necessary facilities to carry out this work successfully.

I wish to express my sincere gratitude to **Dr. J. Raja Sekar**, Professor and Head of the Computer Science and Engineering department for his stimulating support and encouragement for the completion of this project work.

I am graceful to our Project guides **Mr.K.Thirumoorthy**, Assistant Professor (Sr. Grade), Computer Science and Engineering department for their insightful comments and valuable suggestions helped me to complete this project work successfully.

My sincere thanks go to our revered faculty members and lab technicians for their help over this project work.

Last but not the least, I extend my indebtedness towards my beloved family and my friends for their support which made this project a successful one.

## **Table of Contents**

1. Problem Statement
2. Objectives and Outcomes
3. Existing Model
4. Proposed Model
5. Software and Hardware requirements
6. System Design
  - 6.1. Class Diagram
  - 6.2. Database Design
  - 6.3. Product features
7. System Implementation
  - 7.1. Source Code
  - 7.2. Database Queries
8. Results and Discussion
  - 8.1. Output Screenshots
  - 8.2. Reports
9. Summary
10. References

## **1. Problem Statement**

To manage all the details of an employee such as Name, Gender, address, Contact details, Salary and Attendance information and store them in a database. To generate Employee ID automatically starting from default value 1. To set salary for an employee and to update employee details and salary details. To generate pay slip for an employee for each month. Also provide productivity shortcuts such as Notepad and a Calculator within the application.

## **2. Objectives and Outcomes**

The objective of this project is to provide a comprehensive approach towards the management of employee information. With large number of work opportunities, the human workforce has increased. Thus there is a need of a system which can handle the data of such a large number of employees. This project simplifies the task of maintaining records with its user friendly nature. A flexible and easy to use Employee Management software solution for small and medium sized companies provides modules for personnel information management, thereby organizations and companies are able to manage their crucial data. The combination of these modules into one application assures the perfect platform for re-engineering and aligning Human Resource processes along with the organizational goals. This system brings about an easy way of maintaining the details of employees working in any organization. It is simple to understand and can be used by anyone who is not even familiar with simple Employee Management System.



### **3. Existing Model**

Earlier systems were manual where there was no way of proper storing of information. Employee records were stored manually which lead to errors. There was no proper way of tracking employee records. Searching and updating the details of a particular employee was also a tedious task. It was very difficult and required a lot of paperwork which made the work time consuming and less secure. There was no administrator who could handle the records. So there was a need to develop a system which could manage all these things and reduce the paperwork.

### **4. Proposed Model**

Employee Management System is an easy to use application created to manage the employee data. The system will eliminate the manual work with automation. It is created to record the details of the employees. This reduces the dependency on the manual system which is error prone. This system can easily help in tracking employee records. There is a search feature which allows getting records of a specific employee. There is also a feature for generating Pay Slips for the employees every month. The Pay Slips can be printed in PDF Format. There is an administrator who can add, edit, delete and save records in a database. All the entered and updated employee data are stored in the backend in a MySQL database.

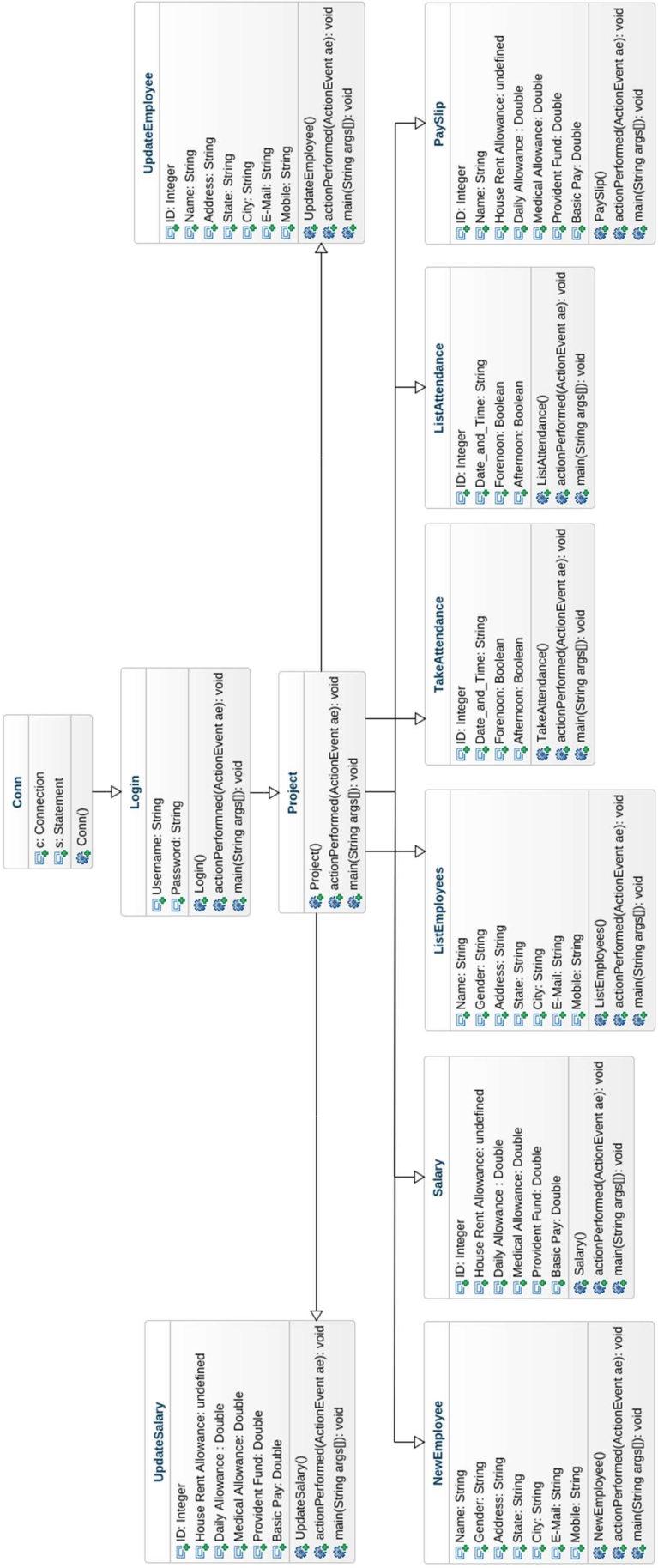
## 5. Software and Hardware requirements

Software Requirements				
Operating System	Editor	Compiler	Database	Database Server
Windows / Linux	Notepad++	Java JDK 8.1 or higher	MySQL	WampServer

Hardware Requirements		
Processor	RAM	Disk Space
Dual Core or above	2 GB or higher	Minimum 5GB of available disk space

## 6. System Design

### 6.1. Class Diagram



## 6.2. Database Design

**Database Name :** Project

**Tables :** Login, Employee, Salary, Attendance

### Description of Tables

#### a. Login

S.No	Field	Datatype	Constraint	Description
1	Username	VARCHAR	DEFAULT	Describes the username of the administrator.
2	Password	VARCHAR	DEFAULT	Describes the password of the corresponding Username.

#### b. Employee

S.No	Field	Datatype	Constraint	Description
1	ID	INTEGER	AUTO_INCREMENT	Describes the unique ID of an employee.
2	Name	VARCHAR	DEFAULT	Describes the name of an employee.
3	Gender	VARCHAR	DEFAULT	Describes the gender of an employee.
4	Address	VARCHAR	DEFAULT	Describes the address of an employee.
5	State	VARCHAR	DEFAULT	Describes the state where an employee is from.

6	City	VARCHAR	DEFAULT	Describes the city where an employee is from.
7	Email	VARCHAR	DEFAULT	Describes the contact e-mail of the employee.
8	Mobile	VARCHAR	DEFAULT	Describes the contact number of the employee.

### c. Salary

S.No	Field	Datatype	Constraint	Description
1	ID	VARCHAR	DEFAULT	Describes the unique ID of an employee.
2	House_Rent_Allowance	VARCHAR	DEFAULT	Describes the House Rent Allowance given to an employee.
3	Daily_Allowance	VARCHAR	DEFAULT	Describes the Daily Allowance given to an employee.
4	Medical_Allowance	VARCHAR	DEFAULT	Describes the Medical Allowance given to an employee.
5	Provident_Fund	VARCHAR	DEFAULT	Describes the Provident Fund allocated to an employee.
6	Basic_Pay	VARCHAR	DEFAULT	Describes the Basic Salary of an employee.

#### **d. Attendance**

<b>S.No</b>	<b>Field</b>	<b>Datatype</b>	<b>Constraint</b>	<b>Description</b>
1	ID	VARCHAR	DEFAULT	Describes the unique ID of an employee.
2	Date_and_Time	VARCHAR	DEFAULT	Describes the date and time at which the attendance was recorded.
3	Forenoon	VARCHAR	DEFAULT	Describes whether the employee was present at the forenoon session or not.
4	Afternoon	VARCHAR	DEFAULT	Describes whether the employee was present at the afternoon session or not.

### **6.3. Product Features**

The Employee Management System Software which highly automates the workflow in an organization, has the below mentioned features :

- Administrative end based security, which means that the employees cannot access or change the data in the database.
- New employee details can be added to the database when some new employee joins the company.
- Employee's will be assigned with an unique ID which will be generated automatically.

- Salary can be set to the employee, with the help of Employee ID of an employee.
- Employee Details and Salary Details can be updated as well as deleted, with the help of the unique ID of an employee.
- Attendance of an employee can be marked, which will be recorded with the exact date and time of marking of the attendance.
- Pay Slip for an employee can be generated automatically, by knowing the Employee ID, and it can also be printed in PDF format, which can also be used as a soft copy.
- All the employee details and attendance details can be printed in a spreadsheet like table, which can also be saved in PDF format.
- Handy and useful shortcuts, which one will need in urge, such as Notepad and Calculator are also integrated within the application, to enhance productivity.
- Shortcut Keys are also provided for each and every option, so that the administrator can solely work with the help of keyboard, without looking for a mouse.

## 7. System Implementation

### 7.1. Source Code

#### a. Conn.java

```
import java.sql.*;

public class Conn
{
    public Connection c;
    public Statement s;
    public Conn()
    {
        try
        {
            Class.forName("com.mysql.jdbc.Driver");
            c= DriverManager.getConnection("jdbc:mysql:///project",
            "root","");
            s = c.createStatement();
        }
        catch(Exception e)
        {
            e.printStackTrace();
        }
    }
}
```



## **b. Login.java**

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

public class Login extends JFrame
{
    JLabel l1,l2,l3;
    JPanel p1,p2;
    JTextField t1;
    JPasswordField t2;
    JButton b1,b2;
    Login()
    {
        super("Login Page");
        setLayout(new BorderLayout());
        ImageIcon icon = new
        ImageIcon(ClassLoader.getResource("icons/logo.png"));
        Image ic = icon.getImage();
        setIconImage(ic);
        t1 = new JTextField(10);
        t2 = new JPasswordField(10);
        l3 = new JLabel(new
        ImageIcon(ClassLoader.getResource("icons/user.png")));
```

```
b1 = new JButton("Submit", new
ImageIcon(ClassLoader.getResource("icons/ok.png")));
b1.setBackground(new Color(0,204,0));
b1.setForeground(Color.BLACK);
b2 = new JButton("Cancel", new ImageIcon
(ClassLoader.getResource("icons/delete.png")));
b2.setBackground(Color.RED);
b2.setForeground(Color.WHITE);
p1 = new JPanel();
p2 = new JPanel();
p1.setBackground(new Color(204,204,204));
p2.setBackground(new Color(204,204,204));
add(l3,BorderLayout.WEST);
p1.add(new JLabel("Username "));
p1.add(t1);
p1.add(new JLabel("Password "));
p1.add(t2);
add(p1,BorderLayout.CENTER);
p2.add(b1);
p2.add(b2);
add(p2,BorderLayout.SOUTH);
setSize(400,250);
setLocation(500,200);
setVisible(true);
setResizable(false);
```

```

b1.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent ae)
    {
        try
        {
            Conn c1=new Conn();
            String u=t1.getText();
            String v=t2.getText();
            String q="select * from login where
username='"+u+"' and password='"+v+"'";
            ResultSet rs=c1.s.executeQuery(q);
            if(rs.next())
            {
                new Project().setVisible(true);
                setVisible(false);
            }
            else
            {
                JOptionPane.showMessageDialog(null,
"Invalid login");
                setVisible(false);
            }
        }
        catch(Exception e)
        {

```

```

        e.printStackTrace();
    }
}
});
b2.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent ae)
    {
        try
        {
            JOptionPane.showMessageDialog(null,
"Cancelled");

            setVisible(false);
        }
        catch(Exception e)
        {
            e.printStackTrace();
        }
    }
});
}

public static void main(String[] args)
{
    Login l = new Login();
    l.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

```

```
}  
}
```

### **c. NewEmployee.java**

```
import javax.swing.*;  
import java.awt.*;  
import java.awt.event.*;  
public class NewEmployee extends JFrame  
{  
    JLabel l1,l2,l3,l4,l5,l6,l7;  
    JTextField t1,t2,t3,t4,t5,t6,t7;  
    JButton b1,b2;  
    JPanel p1;  
    Choice c1;  
    NewEmployee()  
    {  
        super("New Employee");  
        setSize(600,650);  
        setLocation(400,30);  
        setResizable(false);  
        ImageIcon icon = new  
ImageIcon(ClassLoader.getResource("icons/logo.png"));  
        Image ic = icon.getImage();  
        setIconImage(ic);  
        getContentPane().setBackground(Color.WHITE);  
    }  
}
```

```
p1= new JPanel();
p1.setBackground(new Color(255,255,153));
p1.setLayout(new GridLayout(8,2,10,20));
l1 = new JLabel("    Name : ");
l1.setFont(new Font("Arial",Font.BOLD,22));
t1 = new JTextField(15);
p1.add(l1);
p1.add(t1);
c1 = new Choice();
c1.add("Male");
c1.add("Female");
l2 = new JLabel("    Gender : ");
l2.setFont(new Font("Arial",Font.BOLD,22));
p1.add(l2);
p1.add(c1);
l3 = new JLabel("    Address : ");
l3.setFont(new Font("Arial",Font.BOLD,22));
t3 = new JTextField(15);
p1.add(l3);
p1.add(t3);
l4 = new JLabel("    State : ");
l4.setFont(new Font("Arial",Font.BOLD,22));
t4 = new JTextField(15);
p1.add(l4);
```

```
p1.add(t4);
l5 = new JLabel("    City : ");
l5.setFont(new Font("Arial",Font.BOLD,22));
t5 = new JTextField(15);
p1.add(l5);
p1.add(t5);
l6 = new JLabel("    E-Mail ID : ");
l6.setFont(new Font("Arial",Font.BOLD,22));
t6 = new JTextField(15);
p1.add(l6);
p1.add(t6);
l7 = new JLabel("    Mobile : ");
l7.setFont(new Font("Arial",Font.BOLD,22));
t7= new JTextField(15);
p1.add(l7);
p1.add(t7);
b1 =new JButton("SUBMIT", new
ImageIcon(ClassLoader.getResource("icons/ok.png")));
b1.setFont(new Font("Arial",Font.BOLD,16));
b1.setBackground(new Color(0,204,0));
b1.setForeground(Color.BLACK);
b2 = new JButton("CANCEL", new ImageIcon
(ClassLoader.getResource("icons/delete.png")));
b2.setFont(new Font("Arial",Font.BOLD,16));
b2.setBackground(Color.RED);
```

```

b2.setForeground(Color.WHITE);
p1.add(b1);
p1.add(b2);
setLayout(new BorderLayout());
add(p1,"Center");
b1.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent ae)
    {
        String n = t1.getText();
        String g = c1.getSelectedItem();
        String a = t3.getText();
        String s = t4.getText();
        String c = t5.getText();
        String e = t6.getText();
        String p = t7.getText();
        String qry = "insert into employee
values(null,'" + n + "','" + g + "','" + a + "','" + s + "','" + c + "','" + e + "','" + p + "')";
        System.out.println(qry);
        try
        {
            Conn c1 = new Conn();
            c1.s.executeUpdate(qry);

            JOptionPane.showMessageDialog(null,"Employee
Created");

```



```

        setVisible(false);
    }
    catch(Exception ee)
    {
        ee.printStackTrace();
    }
}

});

b2.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent ae)
    {
        try
        {
            JOptionPane.showMessageDialog(null,
"Cancelled");

            setVisible(false);
        }
        catch(Exception e)
        {
            e.printStackTrace();
        }
    }
});
}

```

```

public static void main(String s[])
{
    NewEmployee ne = new NewEmployee();
    ne.setVisible(true);
    ne.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
}

```

#### **d. Salary.java**

```

import java.sql.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class Salary extends JFrame
{
    JLabel l1,l2,l3,l4,l5,l6,l7;
    JTextField t1,t2,t3,t4,t5,t6,t7;
    JButton b1,b2;
    Choice c2;
    Salary()
    {
        super("Set Salary");
        JPanel p1= new JPanel();
        p1.setBackground(new Color(255,255,153));
        p1.setLayout(new GridLayout(7,2,10,20));
    }
}

```

```

setSize(600,650);
setLocation(400,30);
setVisible(true);
setResizable(false);

ImageIcon icon = new
ImageIcon(ClassLoader.getResource("icons/logo.png"));

Image ic = icon.getImage();
setIconImage(ic);

c2 = new Choice();

try
{
    Conn c = new Conn();

    ResultSet rs = c.s.executeQuery("select * from
employee");

    while(rs.next())
    {
        c2.add(rs.getString("id"));
    }
}

catch(Exception e){ }

l1 = new JLabel(" Select Employee Number");
l1.setFont(new Font("Arial",Font.BOLD,22));
p1.add(l1);

p1.add(c2);

l2 = new JLabel(" House Rent Allowance : ");

```

```
l2.setFont(new Font("Arial",Font.BOLD,22));
t1 = new JTextField(15);
p1.add(l2);
p1.add(t1);
l3 = new JLabel(" Daily Allowance : ");
l3.setFont(new Font("Arial",Font.BOLD,22));
t3 = new JTextField(15);
p1.add(l3);
p1.add(t3);
l4 = new JLabel(" Medical Allowance : ");
l4.setFont(new Font("Arial",Font.BOLD,22));
t4 = new JTextField(15);
p1.add(l4);
p1.add(t4);
l5 = new JLabel(" Provident Fund : ");
l5.setFont(new Font("Arial",Font.BOLD,22));
t5 = new JTextField(15);
p1.add(l5);
p1.add(t5);
l6 = new JLabel(" Basic Salary : ");
l6.setFont(new Font("Arial",Font.BOLD,22));
t6 = new JTextField(15);
p1.add(l6);
p1.add(t6);
```

```

        b1 =new JButton("SUBMIT", new
ImageIcon(ClassLoader.getResource("icons/ok.png")));
        b1.setFont(new Font("Arial",Font.BOLD,16));
        b1.setBackground(new Color(0,204,0));
        b1.setForeground(Color.BLACK);
        b2 = new JButton("CANCEL", new
ImageIcon(ClassLoader.getResource("icons/delete.png")))
;
        b2.setFont(new Font("Arial",Font.BOLD,16));
        b2.setBackground(Color.RED);
        b2.setForeground(Color.WHITE);
        p1.add(b1);
        p1.add(b2);
        add(p1);
        b1.addActionListener(new ActionListener(){
            public void actionPerformed(ActionEvent ae)
            {
                String hra = t1.getText();
                String id = c2.getSelectedItem();
                String da = t3.getText();
                String med = t4.getText();
                String pf = t5.getText();
                String basic = t6.getText();
                String qry = "insert into salary values("+ id
+", "+hra+", "+da+", "+med+", "+pf+", "+basic+")";

```

```

        System.out.println(qry);
        try
        {
            Conn c1 = new Conn();
            c1.s.executeUpdate(qry);

JOptionPane.showMessageDialog(null,"Salary Entered.");
            setVisible(false);
        }
        catch(Exception e)
        {
            e.printStackTrace();
        }
    }

});

b2.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent ae)
    {
        try
        {
            JOptionPane.showMessageDialog(null,
"Cancelled");

            setVisible(false);
        }
        catch(Exception e)

```

```

        {
            e.printStackTrace();
        }
    }
});

getContentPane().setBackground(Color.WHITE);
}

public static void main(String[] args)
{
    Salary s = new Salary();
    s.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
}

```

### **e. ListEmployees.java**

```

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

public class ListEmployees extends JFrame
{
    JTable j1;
    JButton b1;

    String h[]={"Employee
ID","Name","Gender","Address","State","City","E-Mail
ID","Mobile"};

```

```
String d[][]=new String[20][8];
int i=0,j=0;
ListEmployees()
{
    super("EMPLOYEE DETAILS");
    setSize(1000,400);
    setLocation(200,100);
    ImageIcon icon = new
ImageIcon(ClassLoader.getResource("icons/logo.png"));
    Image ic = icon.getImage();
    setIconImage(ic);
    try
    {
        String q="select * from employee";
        Conn c1 = new Conn();
        ResultSet rs=c1.s.executeQuery(q);

        while(rs.next())
        {
            d[i][j++]=rs.getString("ID");
            d[i][j++]=rs.getString("Name");
            d[i][j++]=rs.getString("Gender");
            d[i][j++]=rs.getString("Address");
            d[i][j++]=rs.getString("State");
```



```

        d[i][j++]=rs.getString("City");
        d[i][j++]=rs.getString("EMail");
        d[i][j++]=rs.getString("Mobile");
        i++;
        j=0;
    }
    j1=new JTable(d,h);
}
catch(Exception e){}

b1=new JButton("Print",new
ImageIcon(ClassLoader.getResource("icons/print.png")));
b1.setFont(new Font("Arial",Font.BOLD,22));
b1.setBackground(Color.GREEN);
b1.setForeground(Color.BLACK);
add(b1,"South");

JScrollPane s1=new JScrollPane(j1);
add(s1);

b1.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent ae)
    {
        try
        {
            j1.print();
        }
    }
});

```

```

        catch(Exception e){}

    }

});

}

public static void main(String s[])
{
    ListEmployees le = new ListEmployees();
    le.setVisible(true);
    le.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
}

```

#### **f. UpdateEmployee.java**

```

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

public class UpdateEmployee extends JFrame implements
ActionListener,ItemListener
{
    JLabel l1,l2,l3,l4,l5,l6,l7,emp;
    JTextField t1,t2,t3,t4,t5,t6,t7;
    JButton b1,b2;
    Choice c1,c2;
}

```

```

UpdateEmployee()
{
    super("Update Employee Details");

    ImageIcon icon = new
UpdateEmployee(ClassLoader.getResource("icons/logo.png"));
    Image ic = icon.getImage();
    setIconImage(ic);
    setSize(600,650);
    setLocation(400,30);
    setResizable(false);
    c2 = new Choice();
    try
    {
        Conn c = new Conn();
        ResultSet rs = c.s.executeQuery("select * from
employee");
        while(rs.next())
        {
            c2.add(rs.getString("ID"));
        }
    }
    catch(Exception e){}

    JPanel p1= new JPanel();
    p1.setBackground(new Color(51,170,255));
    p1.setLayout(new GridLayout(9,2,10,20));

```

```
emp = new JLabel(" Select Employee ID");
emp.setFont(new Font("Arial",Font.BOLD,22));
p1.add(emp);
p1.add(c2);
l1 = new JLabel(" Name : ");
l1.setFont(new Font("Arial",Font.BOLD,22));
t1 = new JTextField(15);
p1.add(l1);
p1.add(t1);
c1 = new Choice();
c1.add("Male");
c1.add("Female");
l2 = new JLabel(" Gender : ");
l2.setFont(new Font("Arial",Font.BOLD,22));
p1.add(l2);
p1.add(c1);
l3 = new JLabel(" Address : ");
l3.setFont(new Font("Arial",Font.BOLD,22));
t3 = new JTextField(15);
p1.add(l3);
p1.add(t3);
l4 = new JLabel(" State : ");
l4.setFont(new Font("Arial",Font.BOLD,22));
t4 = new JTextField(15);
```

```
p1.add(l4);
p1.add(t4);
l5 = new JLabel("    City : ");
l5.setFont(new Font("Arial",Font.BOLD,22));
t5 = new JTextField(15);
p1.add(l5);
p1.add(t5);
l6 = new JLabel("    E-Mail ID : ");
l6.setFont(new Font("Arial",Font.BOLD,22));
t6 = new JTextField(15);
p1.add(l6);
p1.add(t6);
l7 = new JLabel("    Mobile : ");
l7.setFont(new Font("Arial",Font.BOLD,22));
t7= new JTextField(15);
p1.add(l7);
p1.add(t7);
b1 =new JButton("UPDATE",new
Imagelcon(ClassLoader.getResource("icons/ok.png")));
b2 = new JButton("DELETE",new Imagelcon
(ClassLoader.getResource("icons/delete.png")));
p1.add(b1);
p1.add(b2);
setLayout(new BorderLayout());
add(p1,"Center");
```

```

b1.addActionListener(this);
b1.setFont(new Font("Arial",Font.BOLD,16));
b1.setBackground(new Color(0,204,0));
b1.setForeground(Color.BLACK);
b2.addActionListener(this);
b2.setFont(new Font("Arial",Font.BOLD,16));
b2.setBackground(Color.RED);
b2.setForeground(Color.WHITE);
}
public void actionPerformed(ActionEvent ae)
{
    if(ae.getSource()==b1)
    {
        String n = t1.getText();
        String g = c1.getSelectedItem();
        String a = t3.getText();
        String s = t4.getText();
        String c = t5.getText();
        String e = t6.getText();
        String p = t7.getText();

        String qry = "update employee set
Name='"+n+"',Gender='"+g+"',Address='"+a+"',State='"+s+"',City
='"+c+"',EMail='"+e+"',Mobile='"+p+"' where
ID="+c2.getSelectedItem();

        System.out.println(qry);
    }
}

```

```

try
{
    Conn c1 = new Conn();
    c1.s.executeUpdate(qry);
    JOptionPane.showMessageDialog(null,"Employee
Details Updated.");
    this.setVisible(false);
}
catch(Exception ee)
{
    ee.printStackTrace();
}
}
if(ae.getSource()==b2)
{
    try
    {
        Conn c1 = new Conn();
        c1.s.executeUpdate("delete from employee where
ID="+c2.getSelectedItemId());
        c1.s.executeUpdate("delete from salary where
ID="+c2.getSelectedItemId());
        c1.s.executeUpdate("delete from attendance where
ID="+c2.getSelectedItemId());
    }
}

```

```

        JOptionPane.showMessageDialog(null,"Employee
Deleted.");
        this.setVisible(false);
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
}
public void itemStateChanged(ItemEvent ie)
{
    try
    {
        Conn c1 = new Conn();
        ResultSet rs = c1.s.executeQuery("select * from employee
where id="+c2.getSelectedItemId());
        if(rs.next())
        {
            t1.setText(rs.getString("Name"));
            t3.setText(rs.getString("Address"));
            t4.setText(rs.getString("State"));
            t5.setText(rs.getString("City"));
            t6.setText(rs.getString("EMail"));
            t7.setText(rs.getString("Mobile"));
        }
    }
}

```



```

        }
    }
    catch(Exception ee)
    {
        ee.printStackTrace();
    }
}

public static void main(String[] args)
{
    UpdateEmployee ue = new UpdateEmployee();
    ue.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
}

```

#### **g. UpdateSalary.java**

```

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

public class UpdateSalary extends JFrame implements
ActionListener,ItemListener
{
    JLabel l1,l2,l3,l4,l5,l6,l7;
    JTextField t1,t2,t3,t4,t5;
    JButton b1,b2;
}

```

```

Choice c2;
UpdateSalary()
{
    super("Update Salary");
    setResizable(false);
    ImageIcon icon = new
UpdateSalary(ClassLoader.getResource("icons/logo.png"));
    Image ic = icon.getImage();
    setIconImage(ic);
    c2 = new Choice();
    try
    {
        Conn c = new Conn();
        ResultSet rs = c.s.executeQuery("select * from
employee");
        while(rs.next())
        {
            c2.add(rs.getString("ID"));
        }
    }
    catch(Exception e){}
    JPanel p1= new JPanel();
    p1.setBackground(new Color(51,170,255));
    p1.setLayout(new GridLayout(7,2,10,20));
    l1 = new JLabel(" Select Employee ID");

```

```
l1.setFont(new Font("Arial",Font.BOLD,22));
p1.add(l1);
p1.add(c2);
l2 = new JLabel(" House Rent Allowance : ");
l2.setFont(new Font("Arial",Font.BOLD,22));
t1 = new JTextField(15);
p1.add(l2);
p1.add(t1);
l3 = new JLabel(" Daily Allowance : ");
l3.setFont(new Font("Arial",Font.BOLD,22));
t2 = new JTextField(15);
p1.add(l3);
p1.add(t2);
l4 = new JLabel(" Medical Allowance : ");
l4.setFont(new Font("Arial",Font.BOLD,22));
t3 = new JTextField(15);
p1.add(l4);
p1.add(t3);
l5 = new JLabel(" Provident Fund : ");
l5.setFont(new Font("Arial",Font.BOLD,22));
t4 = new JTextField(15);
p1.add(l5);
p1.add(t4);
l6 = new JLabel(" Basic Salary : ");
```

```
l6.setFont(new Font("Arial",Font.BOLD,22));
t5 = new JTextField(15);
p1.add(l6);
p1.add(t5);
b1 =new JButton("UPDATE",new
ImageIcon(ClassLoader.getResource("icons/ok.png")));
b1.setFont(new Font("Arial",Font.BOLD,16));
b1.setBackground(new Color(0,204,0));
b1.setForeground(Color.BLACK);
b2 = new JButton("DELETE",new
(ClassLoader.getResource("icons/delete.png")));
b2.setFont(new Font("Arial",Font.BOLD,16));
b2.setBackground(Color.RED);
b2.setForeground(Color.WHITE);
p1.add(b1);
p1.add(b2);
b1.addActionListener(this);
b2.addActionListener(this);
setSize(600,550);
setLocation(400,50);
setVisible(true);
add(p1);
getContentPane().setBackground(Color.WHITE);
}
```

```

public void actionPerformed(ActionEvent ae)
{
    if(ae.getSource()==b1)
    {
        String hra = t1.getText();
        String id = c2.getSelectedItem();
        String da = t2.getText();
        String med = t3.getText();
        String pf = t4.getText();
        String basic = t5.getText();

        String qry = "update salary set
House_Rent_Allowance="+hra+",Daily_Allowance="+da+",Medic
al_Allowance="+med+",Provident_Fund="+pf+",Basic_Pay="+bas
ic+" where ID="+id;

        System.out.println(qry);

        try
        {
            Conn c1 = new Conn();
            c1.s.executeUpdate(qry);

            JOptionPane.showMessageDialog(null,"Salary
Updated.");

            this.setVisible(false);
        }
        catch(Exception e)
        {

```

```

        e.printStackTrace();
    }
}
if(ae.getSource()==b2)
{
    try
    {
        Conn c1 = new Conn();
        c1.s.executeUpdate("delete from salary where
ID="+c2.getSelectedItem());
        JOptionPane.showMessageDialog(null,"Salary
Deleted.");
        this.setVisible(false);
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
}
public void itemStateChanged(ItemEvent ie)
{
    try
    {
        Conn c1 = new Conn();

```

```
        ResultSet rs = c1.s.executeQuery("select * from salary
where id="+c2.getSelectedItemId());
        if(rs.next())
        {
            t1.setText(rs.getString("House_Rent_Allowance"));
            t2.setText(rs.getString("Daily_Allowance"));
            t3.setText(rs.getString("Medical_Allowance"));
            t4.setText(rs.getString("Provident_Fund"));
            t5.setText(rs.getString("Basic_Pay"));
        }
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}

public static void main(String[] args)
{
    UpdateSalary us = new UpdateSalary();
    us.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
}
```

## **h. TakeAttendance.java**

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

public class TakeAttendance extends JFrame
{
    JLabel l1,l2,l3;
    JTextField t1,t2,t3,t4,t5,t6,t7;
    JButton b1,b2;
    JPanel p1;
    Choice c2,fh,sh;
    TakeAttendance()
    {
        setTitle("Attendance");
        setLocation(200,300);
        setSize(400,400);
        setResizable(false);

        ImageIcon icon = new
        ImageIcon(ClassLoader.getResource("icons/logo.png"));
        Image ic = icon.getImage();
        setIconImage(ic);
        c2 = new Choice();
        p1= new JPanel();
    }
}
```



```
p1.setBackground(new Color(51,170,255));
p1.setLayout(new GridLayout(4,2,10,20));
try
{
    Conn c = new Conn();
    ResultSet rs = c.s.executeQuery("select * from
employee");
    while(rs.next())
    {
        c2.add(rs.getString("ID"));
    }
}
catch(Exception e){}
l1 = new JLabel("    Select Employee ID");
p1.add(l1);
l1.setFont(new Font("Arial",Font.BOLD,22));
p1.add(c2);
l2 = new JLabel("    Forenoon");
l2.setFont(new Font("Arial",Font.BOLD,22));
fh = new Choice();
fh.add("Present");
fh.add("Absent");
fh.add("Leave");
p1.add(l2);
```

```
p1.add(fh);
l3 = new JLabel("  Afternoon");
l3.setFont(new Font("Arial",Font.BOLD,22));
sh = new Choice();
sh.add("Present");
sh.add("Absent");
sh.add("Leave");
p1.add(l3);
p1.add(sh);
b1 =new JButton("SUBMIT",new
ImageIcon(ClassLoader.getResource("icons/ok.png")));
b1.setFont(new Font("Arial",Font.BOLD,21));
b1.setBackground(new Color(0,204,0));
b1.setForeground(Color.BLACK);
b2 = new JButton("CANCEL",new ImageIcon
(ClassLoader.getResource("icons/delete.png")));
b2.setFont(new Font("Arial",Font.BOLD,21));
b2.setBackground(Color.RED);
b2.setForeground(Color.WHITE);
p1.add(b1);
p1.add(b2);
add(p1);
b1.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent ae)
    {
```

```

        String f = fh.getSelectedItemAt();
        String s = sh.getSelectedItemAt();
        String dt = new java.util.Date().toString();
        String id=c2.getSelectedItemAt();
        String qry = "insert into attendance
values("+ id +",""+dt+"",""+f+"",""+s+"")";
        System.out.println(qry);
        try
        {
            Conn c1 = new Conn();
            c1.s.executeUpdate(qry);

            JOptionPane.showMessageDialog(null,"Attendance
Entered");

            setVisible(false);
        }
        catch(Exception ee)
        {
            ee.printStackTrace();
        }
    }

});

b2.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent ae)
    {

```

```

        try
        {
            JOptionPane.showMessageDialog(null,
"Cancelled");

            setVisible(false);
        }
        catch(Exception e)
        {
            e.printStackTrace();
        }
    }

});

getContentPane().setBackground(Color.WHITE);
setVisible(true);
setSize(600,450);
setLocation(600,200);
}

public static void main(String[] args)
{
    TakeAttendance ta = new TakeAttendance();
    ta.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
}

```

### **i. PaySlip.java**

```
import java.sql.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.util.Date;
public class PaySlip extends JFrame
{
    Choice c1;
    JTextArea t1;
    JButton b1;
    PaySlip()
    {
        setTitle("Pay Slip");
        setSize(800,700);
        setLocation(300,10);
        setResizable(false);
        ImageIcon icon = new
        ImageIcon(ClassLoader.getResource("icons/logo.png"));
        Image ic = icon.getImage();
        setIconImage(ic);
        c1 = new Choice();
        try
        {
```

```
Conn c = new Conn();
ResultSet rs = c.s.executeQuery("select * from salary");
while(rs.next())
{
    c1.add(rs.getString("ID"));
}
}
catch(Exception e){}
setLayout(new BorderLayout());
JPanel p1 = new JPanel();
p1.add(new JLabel("Select ID"));
p1.add(c1);
add(p1,"North");
t1 = new JTextArea(30,80);
JScrollPane jsp = new JScrollPane(t1);
Font f1 = new Font("TimesRoman",Font.BOLD,14);
t1.setFont(f1);
b1 = new JButton("Generate Pay Slip",new
ImageIcon(ClassLoader.getResource("icons/print.png")));
b1.setFont(new Font("Arial",Font.BOLD,22));
b1.setBackground(new Color(0,204,0));
b1.setForeground(Color.BLACK);
add(b1,"South");
add(jsp,"Center");
```

```

b1.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent ae)
    {
        try
        {
            Conn c = new Conn();
            ResultSet rs = c.s.executeQuery("select * from
employee where ID="+c1.getSelectedItemId());
            rs.next();
            String name = rs.getString("Name");
            rs.close();
            rs = c.s.executeQuery("select * from salary where
ID="+c1.getSelectedItemId());
            double gross=0;
            double net=0;
            Date d1 = new Date();
            int month = d1.getMonth() + 1;
            int year = d1.getYear() + 1900;
            t1.setText("-----  PAY SLIP FOR THE
MONTH OF "+month+" ," +year+ " -----");
            t1.append("\n");
            if(rs.next())
            {
                t1.append("\n  Employee ID  :
"+rs.getString("ID"));
            }
        }
    }
}

```

```

t1.append("\n Employee Name : "+name);
t1.append("\n-----");
-----");

t1.append("\n");
double hra =
rs.getDouble("House_Rent_Allowance");
t1.append("\n House Rent Allowance : "+hra);
double da = rs.getDouble("Daily_Allowance");
t1.append("\n Daily Allowance : "+da);
double med = rs.getDouble("Medical_Allowance");
t1.append("\n Medical Allowance : "+med);
double pf = rs.getDouble("Provident_Fund");
t1.append("\n Provident Fund : "+pf);
double basic = rs.getDouble("Basic_Pay");
gross = hra+da+med+pf+basic;
net = gross - pf;
t1.append("\n Basic Pay : "+basic);
t1.append("\n-----");
-----");

t1.append("\n");
t1.append("\n GROSS SALARY : "+gross+" \n
NET SALARY : "+net);

t1.append("\n Tax : 2.1% of GROSS = "+
(gross*2.1/100));
t1.append("\n-----");
-----");

```



```

        t1.append("\n");
        t1.append("\n");
        t1.append("\n");
        t1.append("
( Signature )  ");
    }

    t1.print();
}
catch(Exception e)
{
    e.printStackTrace();
}
});
}

public static void main(String[] args)
{
    PaySlip ps = new PaySlip();
    ps.setVisible(true);
    ps.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
}

```

## **j. ListAttendance.java**

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

public class ListAttendance extends JFrame
{
    JTable j1;
    JButton b1;
    String h[]={"Employee ID","Date &
Time","Forenoon","Afternoon"};
    String d[][]=new String[15][4];
    int i=0,j=0;
    ListAttendance()
    {
        super("View Employees Attendance");
        ImageIcon icon = new
ImageIcon(ClassLoader.getResource("icons/logo.png"));
        Image ic = icon.getImage();
        setIconImage(ic);
        setSize(1000,400);
        setLocation(200,100);
        try
        {
            String q="select * from attendance";
```

```

Conn c1=new Conn();
ResultSet rs=c1.s.executeQuery(q);
while(rs.next())
{
    d[i][j++]=rs.getString("ID");
    d[i][j++]=rs.getString("Date_and_Time");
    d[i][j++]=rs.getString("Forenoon");
    d[i][j++]=rs.getString("Afternoon");
    i++;
    j=0;
}
j1=new JTable(d,h);
}
catch(Exception e){}

b1=new JButton("Print",new
ImageIcon(ClassLoader.getResource("icons/print.png")));
b1.setBackground(new Color(0,189,0));
b1.setForeground(Color.BLACK);
b1.setFont(new Font("Arial",Font.BOLD,22));
add(b1,"South");
JScrollPane s1=new JScrollPane(j1);
add(s1);
b1.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent ae)

```

```

        {
            try
            {
                j1.print();
            }
            catch(Exception e){}
        }
    });
}

public static void main(String[] args)
{
    ListAttendance la = new ListAttendance();
    la.setVisible(true);
    la.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
}

```

#### **k. Project.java**

```

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class Project extends JFrame implements ActionListener
{
    Project()
    {

```

```
super("Employee Management System");
setExtendedState(JFrame.MAXIMIZED_BOTH);
ImageIcon icon = new
ImageIcon(ClassLoader.getResource("icons/logo.png"));
Image i = icon.getImage();
setIconImage(i);
getContentPane().setBackground(new Color(153,102,0));
ImageIcon i1 = new ImageIcon
(ClassLoader.getResource("icons/header.png"));
Image i2 = i1.getImage().getScaledInstance
(1280,720,Image.SCALE_DEFAULT);
ImageIcon i3 = new ImageIcon(i2);
JLabel l1 = new JLabel(i3);
add(l1);
JMenuBar mb = new JMenuBar();
setJMenuBar(mb);
mb.add(Box.createRigidArea(new Dimension(200,100)));
mb.setBackground(Color.WHITE);
JMenu m1 = new JMenu("Employee's Information");
m1.setFont(new Font("TimesRoman",Font.BOLD,25));
JMenuItem t1 = new JMenuItem("Add New Employee");
t1.setFont(new Font("TimesRoman",Font.PLAIN,15));
t1.setMnemonic('N');
t1.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_N,
ActionEvent.CTRL_MASK));
```

```
t1.setIcon(new
ImageIcon(ClassLoader.getResource("icons/New.png")));

JMenuItem t2 = new JMenuItem("Salary Entry");
t2.setFont(new Font("TimesRoman",Font.PLAIN,15));
t2.setMnemonic('S');

t2.setIcon(new
ImageIcon(ClassLoader.getResource("icons/schedreport.p
ng")));

t2.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_S,
ActionEvent.CTRL_MASK));

JMenuItem t3 = new JMenuItem("List of Employees");
t3.setFont(new Font("TimesRoman",Font.PLAIN,15));
t3.setMnemonic('L');

t3.setIcon(new
ImageIcon(ClassLoader.getResource("icons/newinvoice.p
ng")));

t3.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_L,
ActionEvent.CTRL_MASK));

m1.add(t1);
m1.add(t2);
m1.add(t3);
mb.add(m1);

t1.addActionListener(this);
t2.addActionListener(this);
t3.addActionListener(this);

JMenu edit =new JMenu("Update / Delete Details");
```

```
edit.setForeground(new Color(255,102,0));
edit.setFont(new Font("TimesRoman", Font.BOLD, 25));
mb.add(edit);

JMenuItem s1 = new JMenuItem("Update / Delete Employee
Details");

s1.setFont(new Font("TimesRoman",Font.PLAIN,15));
s1.setMnemonic('p');

s1.setIcon(new ImageIcon
(ClassLoader.getResource("icons/empreport.png")));

s1.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_P,
ActionEvent.CTRL_MASK));

edit.add(s1);

JMenuItem s2 = new JMenuItem("Update / Delete Salary
Details of Employee");

s2.setFont(new Font("TimesRoman",Font.PLAIN,15));
s2.setMnemonic('U');

s2.setIcon(new ImageIcon
(ClassLoader.getResource("icons/EditOpen.png")));

s2.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_U,
ActionEvent.CTRL_MASK));

edit.add(s2);

JMenuItem s3 = new JMenuItem("Attendance Entry");
s3.setFont(new Font("TimesRoman",Font.PLAIN,15));
s3.setMnemonic('T');

s3.setIcon(new ImageIcon
(ClassLoader.getResource("icons/EXPENSE.PNG")));
```

```
s3.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_T,
ActionEvent.CTRL_MASK));
edit.add(s3);
s1.addActionListener(this);
s2.addActionListener(this);
s3.addActionListener(this);
JMenu rep =new JMenu("Employee's Report");
rep.setFont(new Font("TimesRoman", Font.BOLD, 25));
mb.add(rep);
JMenuItem p1 = new JMenuItem("Generate Salary Slip");
p1.setFont(new Font("TimesRoman",Font.PLAIN,15));
p1.setMnemonic('G');
p1.setIcon(new ImageIcon
(ClassLoader.getResource("icons/payments.png")));
p1.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_G,
ActionEvent.CTRL_MASK));
rep.add(p1);
JMenuItem p2 = new JMenuItem("Attendance Record");
p2.setFont(new Font("TimesRoman",Font.PLAIN,15));
p2.setMnemonic('A');
p2.setIcon(new ImageIcon
(ClassLoader.getResource("icons/empreport.png")));
p2.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_A,
ActionEvent.CTRL_MASK));
rep.add(p2);
```



```
p1.addActionListener(this);
p2.addActionListener(this);
JMenu util =new JMenu("Shortcuts");
util.setForeground(new Color(255,102,0));
util.setFont(new Font("TimesRoman", Font.BOLD, 25));
mb.add(util);

JMenuItem u1 = new JMenuItem("Notepad");
u1.setIcon(new ImageIcon
(ClassLoader.getResource("icons/New.png")));
u1.setFont(new Font("TimesRoman",Font.PLAIN,15));
u1.setMnemonic('o');
u1.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_O,
ActionEvent.CTRL_MASK));
util.add(u1);

JMenuItem u2 = new JMenuItem("Calculator");
u2.setIcon(new
ImageIcon(ClassLoader.getResource("icons/calc.png")));
u2.setFont(new Font("TimesRoman",Font.PLAIN,15));
u2.setMnemonic('C');
u2.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_C,
ActionEvent.CTRL_MASK));
util.add(u2);

u1.addActionListener(this);
u2.addActionListener(this);

JMenu m8=new JMenu("Exit");
```

```
m8.setFont(new Font("TimesRoman", Font.BOLD, 25));
mb.add(m8);
JMenuItem m8i1=new JMenuItem("Exit");
m8.add(m8i1);
m8i1.setFont(new Font("TimesRoman", Font.PLAIN, 15));
m8i1.setMnemonic('X');
m8i1.setIcon(new
ImageIcon(ClassLoader.getResource("icons/exit.PNG")));
    m8i1.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_X,
ActionEvent.CTRL_MASK));
    m8i1.addActionListener(this);
}
public void actionPerformed(ActionEvent ae)
{
    String msg= ae.getActionCommand();
    if(msg.equals("Add New Employee"))
        new NewEmployee().setVisible(true);
        else if(msg.equals("Salary Entry"))
            new Salary().setVisible(true);
    else if(msg.equals("List of Employees"))
        new ListEmployees().setVisible(true);
    else if(msg.equals("Update / Delete Employee Details"))
        new UpdateEmployee().setVisible(true);
    else if(msg.equals("Update / Delete Salary Details of
Employee"))
```

```
new UpdateSalary().setVisible(true);
    else if(msg.equals("Attendance Entry"))
new TakeAttendance().setVisible(true);
    else if(msg.equals("Generate Salary Slip"))
new PaySlip().setVisible(true);
else if(msg.equals("Attendance Record"))
    new ListAttendance().setVisible(true);
else if(msg.equals("Notepad"))
{
    try
    {
        Runtime.getRuntime().exec("notepad.exe");
    }
    catch(Exception e){}
}
else if(msg.equals("Calculator"))
{
    try
    {
        Runtime.getRuntime().exec("calc.exe");
    }
    catch(Exception e){ }
}
else if(msg.equals("Exit"))
```

```

        System.exit(0);
    }
    public static void main(String[] args)
    {
        Project p = new Project();
        p.setVisible(true);
        p.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
}

```

## 7.2. Database Queries

### a. Creating Database

- i. The database **Project** is created by the query  
CREATE DATABASE project;
- ii. The **project** database can be used by the  
query : USE project;
- iii. There are four tables available in the database,  
which are **login**, **employee**, **salary** and  
**attendance**.
- iv. **Query used to create the table login :**

```

CREATE TABLE login(username VARCHAR(255),
password VARCHAR(255));

```

**v. Query to add an Administrator :**

```
INSERT INTO login values('admin','admin');
```

**vi. Query used to create the table employee :**

```
CREATE TABLE employee(ID INT(11) NOT NULL  
AUTO_INCREMENT, Name VARCHAR(50),  
Gender VARCHAR(10), Address VARCHAR(500),  
State VARCHAR(20), City VARCHAR(20), EMail  
VARCHAR(50), Mobile VARCHAR(20),  
CONSTRAINT employee_pk PRIMARY KEY(ID));
```

**vii. Query used to create the table salary :**

```
CREATE TABLE salary(ID VARCHAR(20),  
House_Rent_Allowance VARCHAR(20),  
Medical_Allowance VARCHAR(20),  
Provident_Fund VARCHAR(20), Basic_Pay  
VARCHAR(20));
```

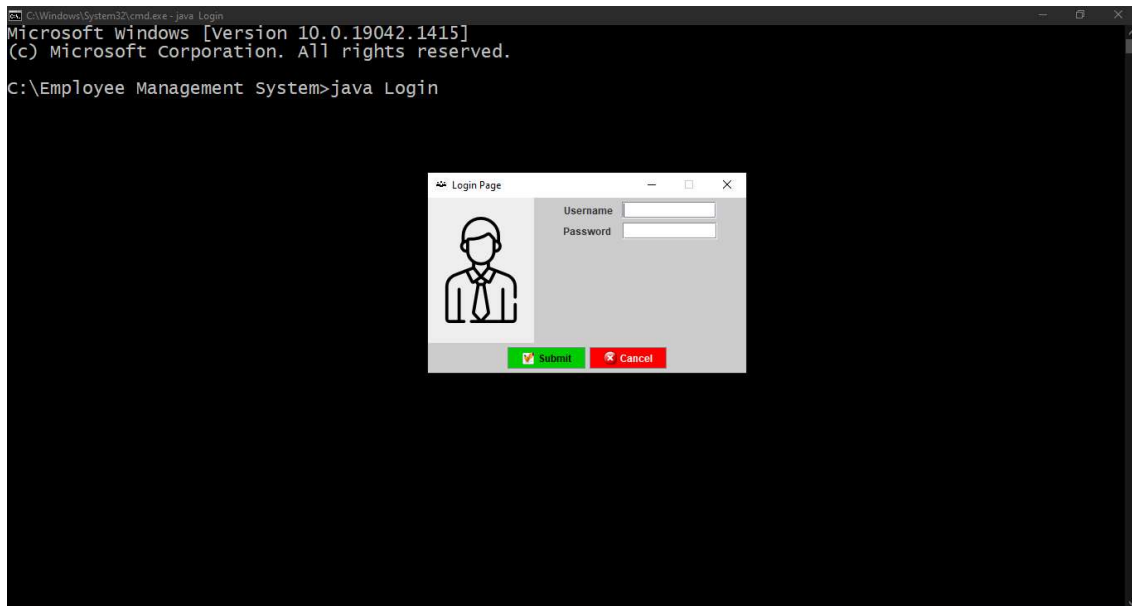
**viii. Query used to create the table attendance :**

```
CREATE TABLE attendance(ID VARCHAR(20),  
Date_and_Time VARCHAR(200), Forenoon  
VARCHAR(20), Afternoon VARCHAR(20));
```

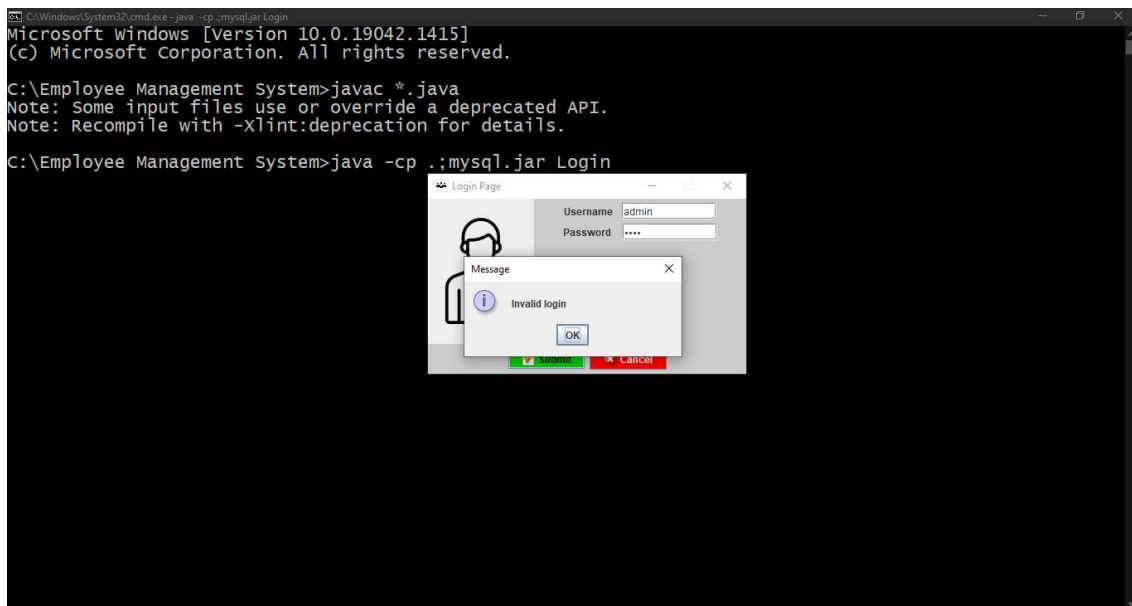
## 8. Results and Discussion

### 8.1. Output Screenshots

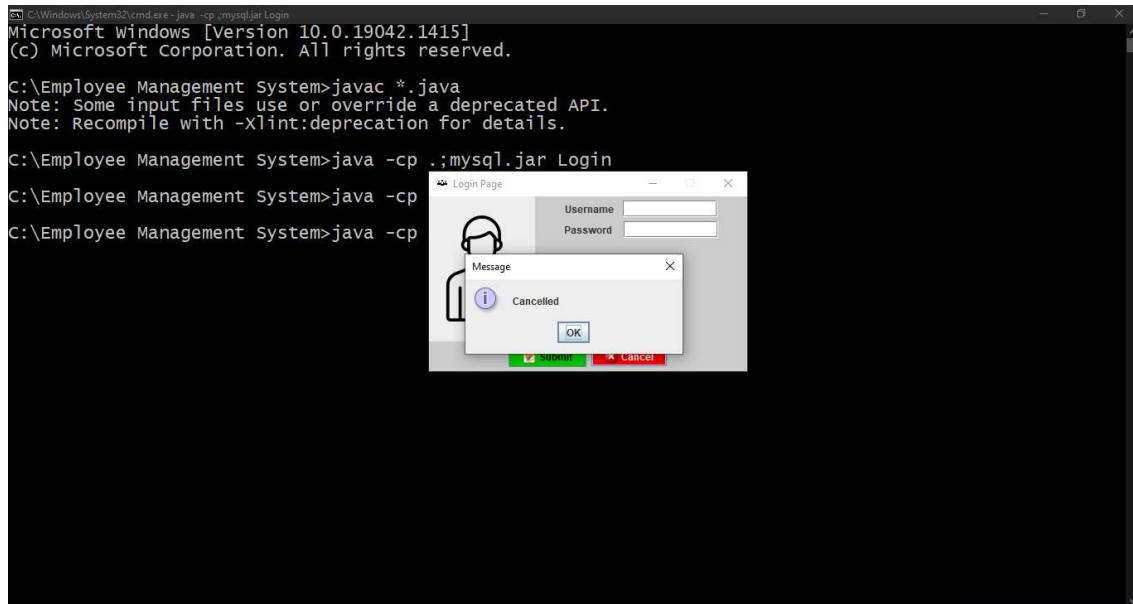
#### 1. Login Panel



#### 2. Invalid Login Message



### 3. Clicking Cancel Button



### 4. Main Window – On Successful Login



## 5. Adding New Employee

Employee Management System

Employee's Information

New Employee

Name : Ragav S

Gender : Male

Address : 17, Green Street, Sivakasi 626-005.

State : TamilNadu

City : Virudhunagar

E-Mail ID : ragav@outlook.in

Mobile : 7890987654

SUBMIT CANCEL

## 6. Pop Up message on successful creation

Employee Management System

Employee's Information

New Employee

Name : Ragav S

Gender : Male

Address : 17, Green Street, Sivakasi 626-005.

State : TamilNadu

City : Virudhunagar

E-Mail ID : ragav@outlook.in

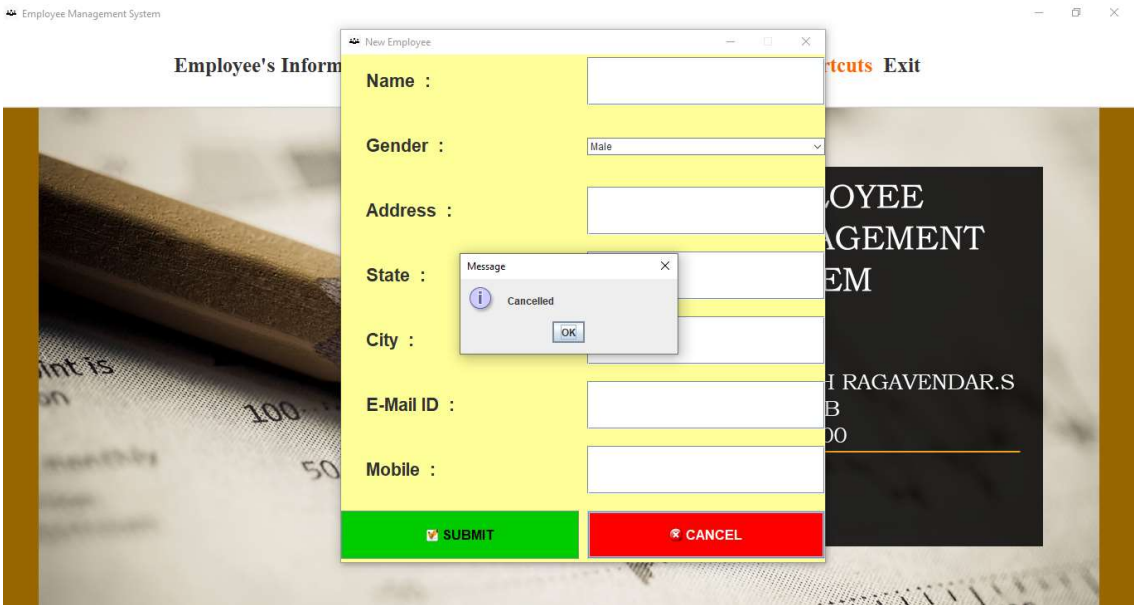
Mobile : 7890987654

SUBMIT CANCEL

Message  
Employee Created  
OK



## 7. Hitting the Cancel Button



## 8. Data Insertion at Back-End

```
c:\wamp\bin\mysql\mysql5.7.36\bin\mysql.exe
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 228
Server version: 5.7.36 MySQL Community Server (GPL)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use project
Database changed
mysql> select * from employee;
```

ID	Name	Gender	Address	State	City	EMail	Mobile
1	Harrish S	Male	15, Park Street, Madurai 625-016.	TamilNadu	Madurai	harrish@gmail.com	7010787159
2	Ragav S	Male	17, Green Street, Sivakasi 626-005.	TamilNadu	Virudhunagar	ragav@outlook.in	7890987654

```
2 rows in set (0.03 sec)

mysql>
```

## 9. Setting the salary for the employees

Employee Management System

Employee's Information

Set Salary

Select Employee Number: 1

House Rent Allowance : 3000

Daily Allowance : 500

Medical Allowance : 700

Provident Fund : 2000

Basic Salary : 24000

☒ SUBMIT ☐ CANCEL

Employee Management System

Employee's Information

Set Salary

Select Employee Number: 1

House Rent Allowance : 3000

Daily Allowance : 500

Medical Allowance : 700

Provident Fund : 2000

Basic Salary : 24000

☒ SUBMIT ☐ CANCEL

Employee Management System

Employee's Information

Set Salary

Select Employee Number: 1

House Rent Allowance : 3000

Daily Allowance : 500

Medical Allowance : 700

Provident Fund : 2000

Basic Salary : 24000

☒ SUBMIT ☐ CANCEL

Message

Salary Entered.

OK

Employee Management System

Employee's Information

Set Salary

Select Employee Number: 1

House Rent Allowance : 3000

Daily Allowance : 500

Medical Allowance : 700

Provident Fund : 2000

Basic Salary : 24000

☒ SUBMIT ☐ CANCEL

## 10. Salary data at Back-End

```
c:\wamp\bin\mysql\mysql5.7.30\bin\mysql.exe
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 228
Server version: 5.7.30 MySQL Community Server (GPL)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use project
Database changed
mysql> select * from employee;

+----+-----+-----+-----+-----+-----+-----+
| ID | Name      | Gender | Address                                     | State | City           | EMail              | Mobile |
+----+-----+-----+-----+-----+-----+-----+
| 1  | Harrish S | Male   | 15, Park Street, Madurai 625-016.         | TamilNadu | Madurai        | harrish@gmail.com  | 7010787159 |
| 2  | Ragav S   | Male   | 17, Green Street, Sivakasi 626-005.       | TamilNadu | Virudhunagar   | ragav@outlook.in   | 7890987654 |
+----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.03 sec)

mysql> select * from salary;

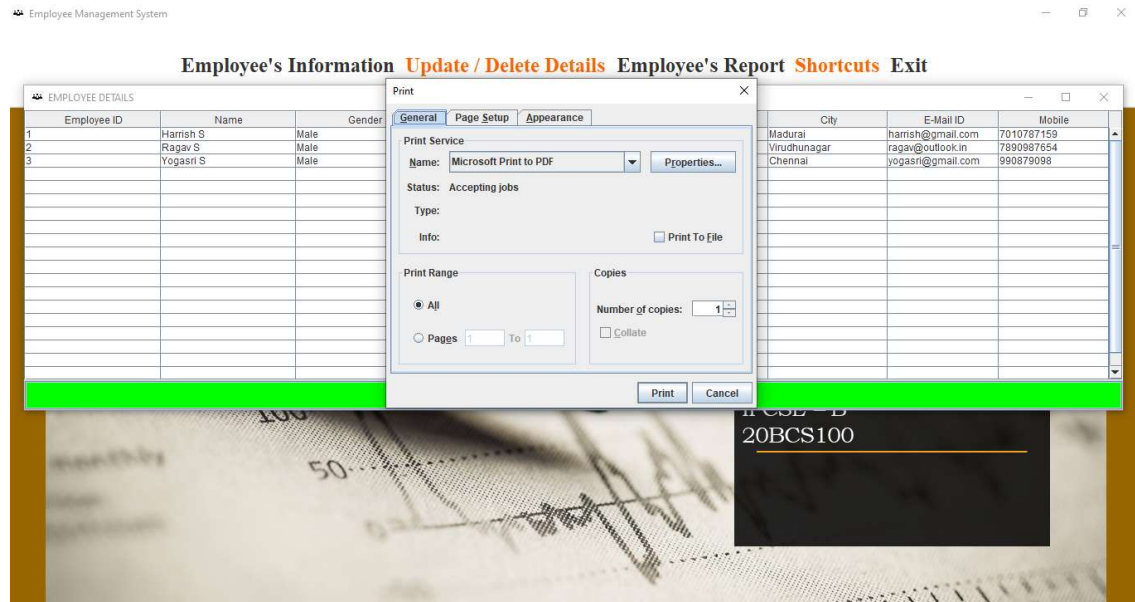
+----+-----+-----+-----+-----+-----+
| ID | House_Rent_Allowance | Daily_Allowance | Medical_Allowance | Provident_Fund | Basic_Pay |
+----+-----+-----+-----+-----+-----+
| 1  | 3000                 | 500              | 700                | 2000            | 24000     |
| 2  | 5000                 | 450              | 800                | 1700            | 25000     |
| 3  | 4000                 | 650              | 1000               | 3000            | 30000     |
+----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

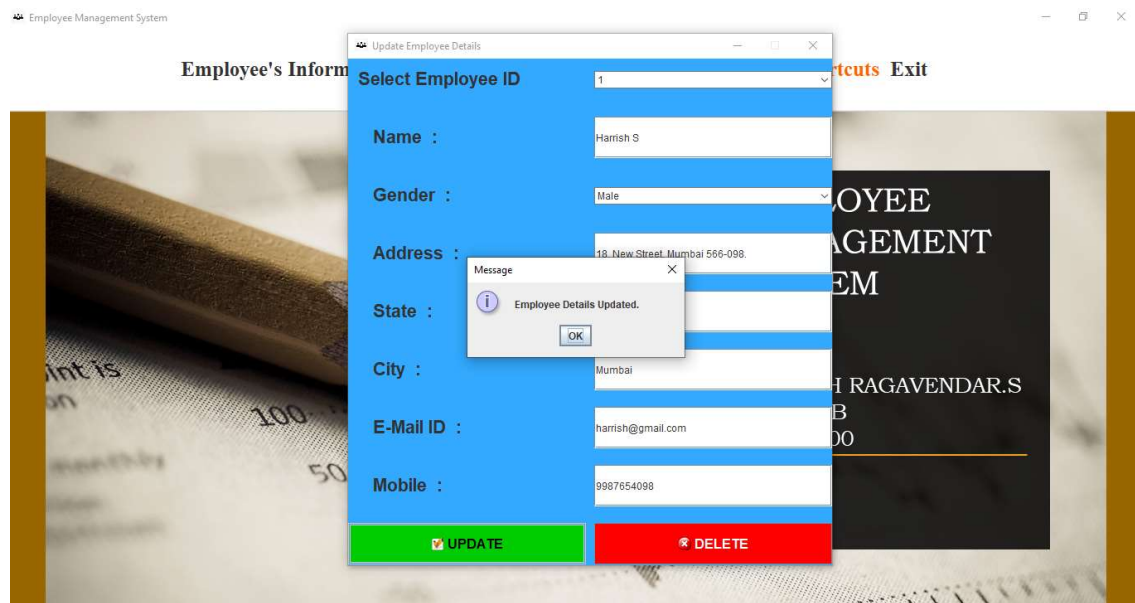
## 11. Displaying the list of employees

[illegible]

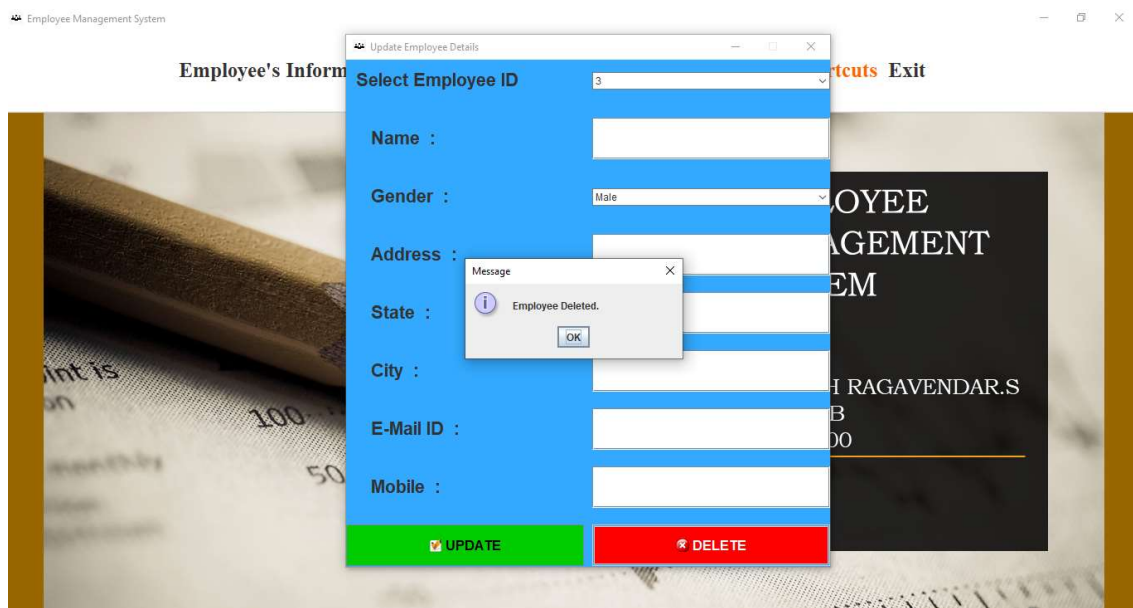
## 12. Printing Employee Details in PDF Format



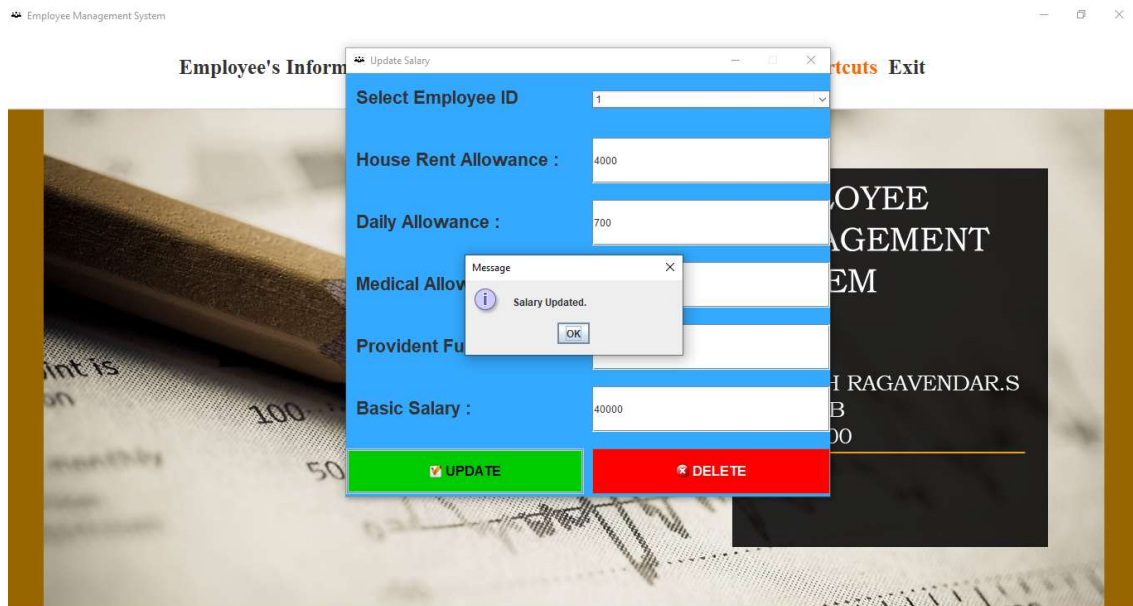
## 13. Updating Employee Details



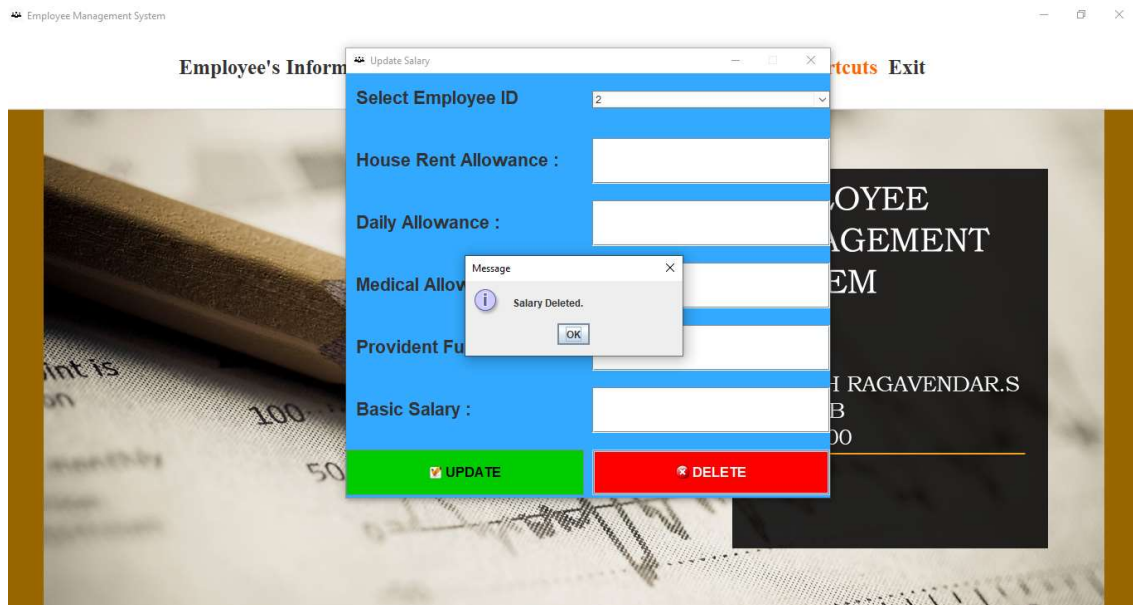
## 14. Deleting Employee Details



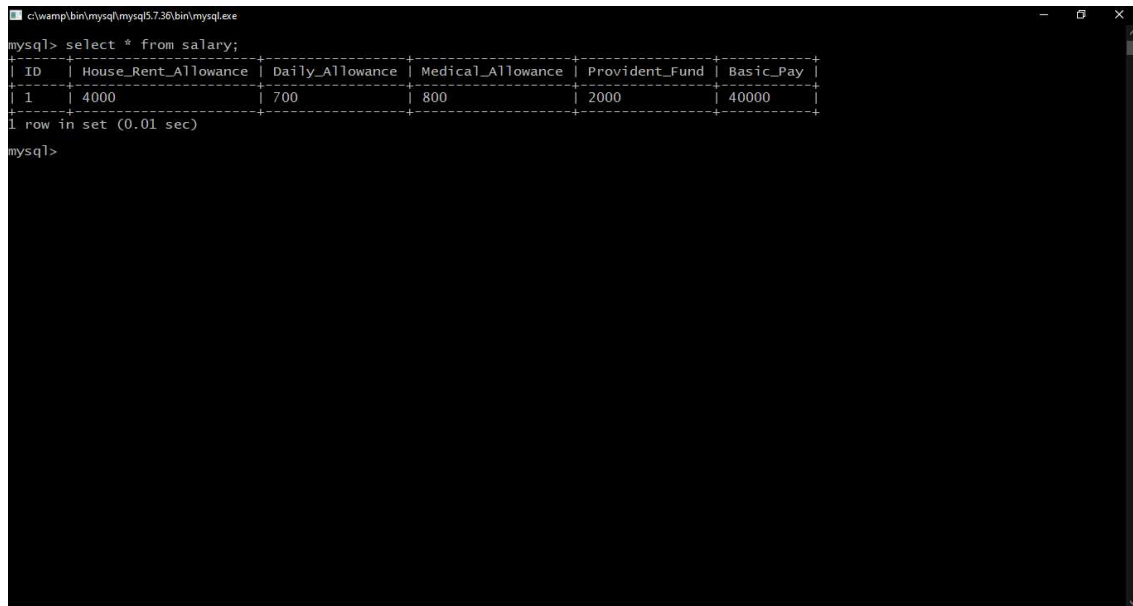
## 15. Updating Salary Details



## 16. Deleting Salary of Employee

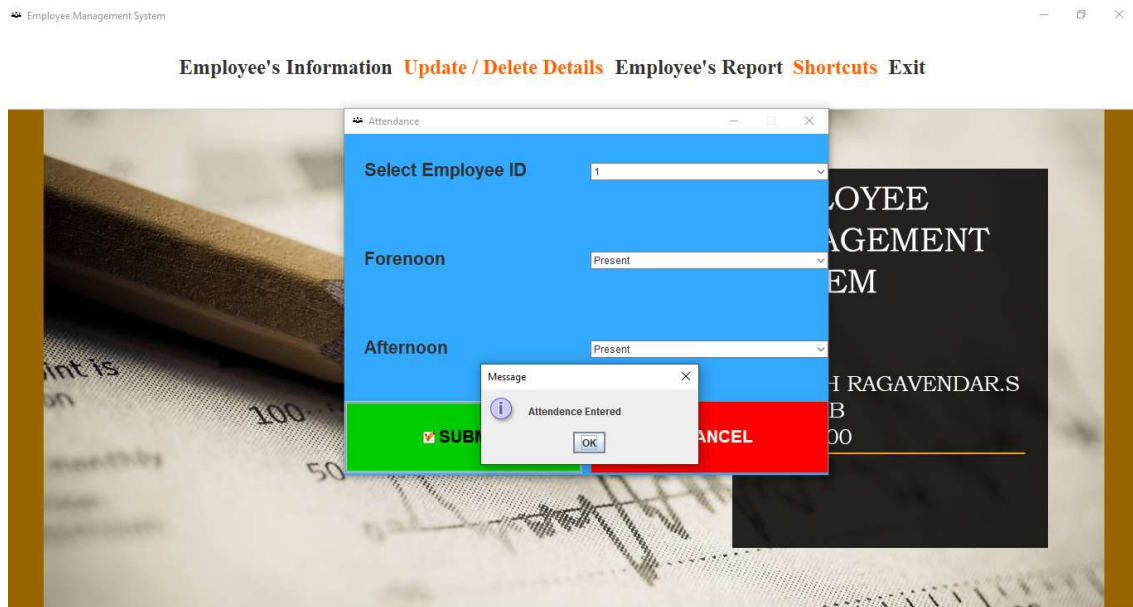


## 17. Deletion at Back-End





## 18. Attendance Entry



## 19. Entry at Back-End

```
c:\wamp\bin\mysql\mysql5.7.30\bin\mysql.exe
mysql> select * from attendance;
+----+-----+-----+-----+-----+-----+-----+
| ID | Date_and_Time | Forenoon | Afternoon |
+----+-----+-----+-----+-----+-----+
| 1  | Sun Dec 26 10:26:59 IST 2021 | Present | Present |
| 2  | Sun Dec 26 10:28:02 IST 2021 | Absent  | Present |
| 4  | Sun Dec 26 10:28:11 IST 2021 | Present | Leave   |
| 5  | Sun Dec 26 10:28:17 IST 2021 | Present | Present |
+----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> _
```

## 20. Generating Pay Slip

Employee Management System

Pay Slip

Select ID: 1

PAY SLIP FOR THE MONTH OF 12, 2021

Employee ID : 1  
Employee Name : Harrih S

---


House Rent Allowance : 4000.0  
Daily Allowance : 700.0  
Medical Allowance : 800.0  
Provident Fund : 2000.0  
Basic Pay : 40000.0

---

GROSS SALARY : 47500.0  
NET SALARY : 45500.0  
Tax : 2.1% of GROSS = 997.5

---

( Signature )



## 21. Printing Pay Slip in PDF Format

Employee Management System

Pay Slip

Select ID: 1

PAY SLIP FOR THE MONTH OF 12, 2021

Employee ID : 1  
Employee Name : Harrih S

---


House Rent Allowance : 4000.0  
Daily Allowance : 700.0  
Medical Allowance : 800.0  
Provident Fund : 2000.0  
Basic Pay : 40000.0

---

GROSS SALARY : 47500.0  
NET SALARY : 45500.0  
Tax : 2.1% of GROSS = 997.5

---

( Signature )



Print

General Page Setup Appearance

Print Service

Name: Microsoft Print to PDF Properties...

Status: Accepting jobs

Type:

Info: ☐ Print To File

Print Range

☒ All

☐ Pages 1 To 1

Copies

Number of copies: 1

☐ Collate

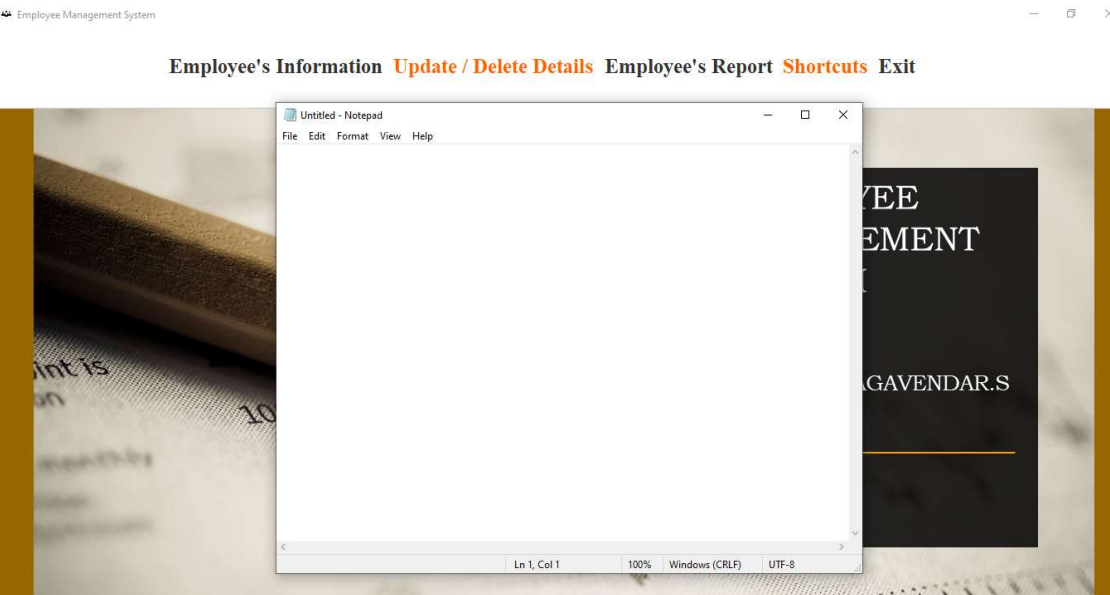
Print Cancel



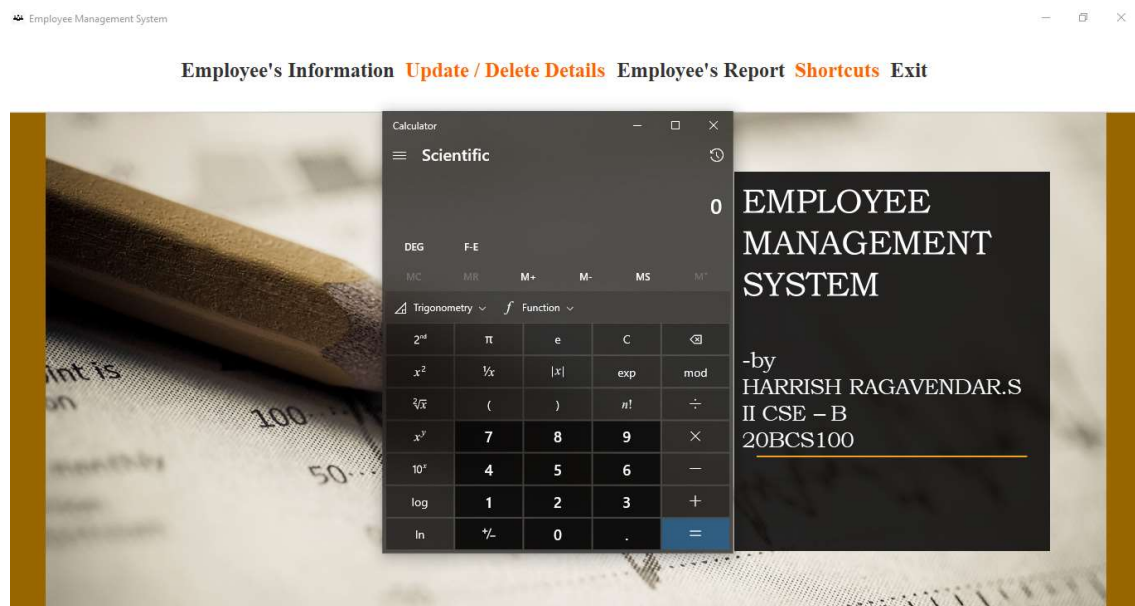
## 22. Displaying Attendance Details

[illegible]

## 23. Shortcuts – Opening Notepad



## 24. Shortcuts – Opening Calculator



## 25. Exit



## **8.2. Reports**

### **Available Classes :**

1. Connection
2. Login
3. NewEmployee
4. Salary
5. ListEmployees
6. UpdateEmployee
7. UpdateSalary
8. TakeAttendance
9. PaySlip
10. ListAttendance
11. Project (Main Frame)

#### **I. Connection**

This class establishes the connection between the project and MySQL Database.

#### **II. Login**

This class is used to create the Login Panel. If the username or password is wrong, it displays a prompt as Invalid Login. If the credentials are valid, then the administrator can login.

#### **III. NewEmployee**

This class is used to add the details of a new employee into the database. Once the employee is added, a unique ID will be generated automatically for the employee.

#### **IV. Salary**

This class is used to set Salary to an employee, using the unique ID of the employee.

#### **V. ListEmployees**

This class lists all the details of the employees in a spreadsheet format, which can also be printed, or converted to PDF format.

#### **VI. UpdateEmployee**

This class is used to Update or Delete the details of the employees using the Employee ID.

#### **VII. UpdateSalary**

This class is used to Update or Delete the Salary details of the employees using the Employee ID.

#### **VIII. TakeAttendance**

This class is used to mark attendance for the employees, in both the Forenoon and Afternoon Sessions, using the Employee ID.

#### **IX. PaySlip**

This class is used to generate the PaySlip for the employee every Month, by using their Employee ID. The PaySlip can also be printed in PDF format.

#### **X. ListAttendance**

This class is used to list the Attendance Details of the employees in a spreadsheet format, which can also be printed, or converted to PDF format.

## **XI. Project**

This class is the heart or core of the Employee Management System. It contains a menu bar, which contains a lot of menus, and each menu has various menu items. It is designed in such a way that it has shortcuts for all features, and the UI is also made so colourful and pleasing to the end user.

## **9. Summary**

Thus the Employee Management System project automates the entire tasks of storing and updating each and every details of an employee. It comes handy in tracking employee records and generating pay slips. It also generates the details of all the employees, which can be printed in a PDF Format. All the employee records are integrated and so this makes it user-friendly and easy to use application.

## **10. References**

Geeks for Geeks : <https://www.geeksforgeeks.org/java/>

StackOverflow : <https://stackoverflow.com>

Java course by freeCodeCamp : <https://youtu.be/grEKMHGyYns>