

Project Report On

# “Employee Management System”

***Submitted to***

Rashtrasant Tukadoji Maharaj Nagpur University, Nagpur

In partial fulfillment of the requirement of

**Bachelor of Commerce (Computer Application) (BCCA) (New) Part-III Semester-VI**

Examination Summer-2023

***Submitted By***

#### Mr. Yash S. Wadpalliwar

#### Mr. Vishesh S. Shiv

***Under the Guidance of***

**Prof. Yogesh T. Bisen**

**Department of Computer Science**

**KAMLA NEHRU MAHAVIDYALAYA**

# Sakkardara Chowk, Nagpur-24

**2022-2023**

Certificate

This is to certify that the project report on **“Employee Management System”** is submitted by **Mr. Yash S. Wadpalliwar, Mr. Vishesh S. Shiv** for partial fulfillment of the requirement of **Bachelor of Commerce (Computer Application) (BCCA) (New) Part-III Semester-VI** examination of the Rashtrasant Tukadoji Maharaj Nagpur University, Nagpur.

It is the original Software project carried out under the supervision and guidance of **Prof. Yogesh T. Bisen** and undergone requisite duration as prescribed by Rashtrasant Tukadoji Maharaj Nagpur University, Nagpur for the project work.

**Prof. Sachin Y. Zade**

**Head of the Dept. Comp. Science**

Kamla Nehru Mahavidyalaya, Nagpur

***Guide***

#### Prof. Yogesh T. Bisen

Kamla Nehru Mahavidyalaya, Nagpur

#### Internal Examiner External Examiner

**Place: Nagpur Date**:

***DECLARATION***

To,

#### The Principal,

Kamla Nehru Mahavidyalaya, Sakkardara chowk,

Nagpur-24.

Respected Sir,

We the undersigned hereby declare that the work **“Employee Management System”** developed and submitted by us is our original work. The system presented here, is developed by us independently and has not been duplicated from any other source.

We understand that any such copying is liable to be punished in any way the University deem fit.

#### Place: Nagpur (Name and Signature of Projectee)

**Date: Mr. Yash S. Wadpalliwar**

#### Mr. Vishesh S. Shiv

***ACKNOWLEDGEMENT***

We wish to express our sincere thanks to the honorable **Dr. Dilip S. Badwaik**, Principal, Kamla Nehru Mahavidyalaya, Nagpur and to the **Prof. Sachin Y. Zade** , H.O.D., Department of Computer science for providing us varieties of opportunities, infrastructural facilities and inspiration to gather professional knowledge and material without which it would have been impossible to complete this hard task.

We take this opportunity to express our deep gratitude and whole hearted thanks to our project guide **Prof. Yogesh T. Bisen** for his guidance throughout this work. We are very much thankful to their kindness, encouragement and the valuable time, which they have devoted to us.

We wish to thanks all those, who have helped us in our way or the others in bringing out this project successful.

#### Signature and Student Name’s Mr. Yash S. Wadpalliwar

**Mr. Vishesh S. Shiv**

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Chapter** | **Page no** |
|  | **Introduction** | **1-1** |
|  | **Objectives** | **2-2** |
|  | **Preliminary System Analysis.**   * **Preliminary Investigation** * **Present System in use** * **Flaws in Present System.** * **Need of New System** * **Feasibility Study** * **Project Category** | **3-10** |
|  | **Software Engineering Paradigm Applied**   * **Modules** * **System / Modular Chart.** | **8-8** |
|  | **Software and Hardware Requirement Specification** | **9-9** |
|  | **Detailed System Analysis**   * **Data Flow Diagram** * **Numbers of Modules and Process Logic** * **Data Structures and Tables** * **Entity-Relationship Diagram** | **10-15** |
|  | **System Design.**   * **Form Design** * **Source Code** * **Input screen & Output Screen** | **16-58** |
|  | **Testing and Validation Checks** | **59-59** |
|  | **System Security Measures** | **60-60** |
|  | **Implementation, Evaluation and Maintenance** | **61-61** |
|  | **Future Scope of the project** | **62-62** |
|  | **Suggestion & Conclusion** | **63-63** |
|  | **Bibliography& References** | **64-64** |

# INTRODUCTION

Everything has been digitised in our age of ever-increasing technology. The human workforce has

grown as a result of the abundance of job options. As a result, a system that can handle the data of

such a vast number of people in a company is required. Because of its user-friendly design, this

project makes the process of keeping records easier. The "EMPLOYEE MANAGEMENT

SYSTEM" was created to address the issues that plagued the previous manual system. This

programme is designed to eliminate, and in some cases, decrease, the problems that the current

system has.

To eliminate data entry mistakes, the software is kept as simple as possible. When inputting

incorrect data, it also displays an error notice. The user doesn’t require any formal expertise to

operate this system. The admin will be able to add new employees to this project. Employee data

may also be seen and printed by the administrator. Admins can also remove an employee and

change their details.

**Why You Need a Employee Management**

1. Increased productivity: When employees are managed effectively, they tend to be more productive. This is because they have a clear understanding of their roles and responsibilities, and they feel supported by their managers.

2. Improved morale: Effective employee management can also improve employee morale. When employees feel valued, supported, and motivated, they are more likely to be satisfied with their jobs and more likely to stay with the company.

3. Better communication: Good employee management requires effective communication between managers and employees. This can help to ensure that everyone is on the same page, and that any issues or concerns are addressed in a timely manner.

4. Improved performance: Effective employee management can also help to improve employee performance. By setting clear goals and providing regular feedback and support, managers can help their employees to develop the skills and knowledge they need to succeed.

5. Reduced turnover: Finally, good employee management can help to reduce turnover. When employees are happy and satisfied with their jobs, they are less likely to leave the company, which can save time and money on recruitment and training costs.

**OBJECTIVE**

The objective of this work is to give a complete approach to personnel information management.

This will be accomplished by developing and deploying an HR management system that will result

in a significant shift in the way employee data is managed.

This system's objectives include the following:

1. Design of an HR management system to meet needs such as adding and deleting

employees, viewing and printing employee data, and updating employee information.

2. Employee data is stored in a well-designed database.

3. An easy-to-use interface that will let user interact with the system.

4. Improve employee productivity and performance: This involves setting clear goals and expectations for employees, providing them with the necessary training and resources, and offering regular feedback and coaching to help them improve their performance.

5. Increase employee engagement and job satisfaction: This involves creating a positive work environment where employees feel valued, supported, and motivated to do their best work. It may include providing opportunities for career development, recognizing and rewarding good performance, and promoting work-life balance.

6. Reduce turnover and retain top talent: This involves implementing strategies to attract and retain top talent, such as offering competitive compensation and benefits packages, providing opportunities for growth and advancement, and fostering a positive company culture.

7. Ensure legal compliance: This involves staying up-to-date on employment laws and regulations, and ensuring that the company is in compliance with all applicable laws and regulations

**3. PRELIMINARY SYSTEM ANALYSIS**

**3.1 Preliminary Investigation**

In olden days the transaction took place are recorded and maintain on paper. So it becomes

very difficult to maintain the records, moreover the calculation are done manually this would creates

mistakes and errors. This generates the result less accurate and less efficient and unreliable.

All such limitation have been overcomes under this project. In this project the information of

new suppliers Employee Management System can be added, deleted, saved, updated by using the various command button provided

to the user. So it becomes easier for the user to maintain the records. Similar provisions have been

made for other detail option such as departments details, employee details etc.

3.2 Present System in Use

The present system in manually and has many flaws. The system did fulfill the requirement

of the earlier time, but today with over population and the bulky records the system has aged off.

• Less accurate

• Less efficient

• Possibility in creating mistakes in calculating

• Time consuming process

• Bulk of paper work

• The most important factor that is time and speed greatly affects

• During working in manual system

#### iii) Feasibility Study

In order to do a feasibility study, we must consider the following:

1. Technical Feasibility

The availability of hardware & Software necessary for the creation of the system, as-well-

as the compatibility and maturity of the technology planned to be used, and the availability

of the requisite technical staff to create the system, are all factors to consider.

2. Operational Feasibility

Problems that may develop during operations are the focus of operation feasibility. There

are two parts to this problem to consider:

3. Economic Feasibility

The concept of economic feasibility is determining whether or not the potential benefit of

fixing difficulties is worth while. Because member needs &alternative solutions haven’t

been specified at this point, it is difficult to estimate the cost at this level.

#### iv) Need of New System

* This system reduces paper work.
* This system reduces errors.
* It is easy to access present data.
* Direct access by operator to seat booking information of the client.
* Systematic storage of record.

Only a single person can operate this system. It also provides proprietor best tools to access all the information about his business to work on it. The application deals with all those small aspects of business. We decided to create this entire project in java because it provides graphical interface, user-friendly, easy to understand. Coding, compilation and debugging can be easily done and errors can be found more accurately and effectively. Access is used as database since it is very easy to handle and operate. In access it is also easy to access data into front end.

The logic behind rest of the operation is varying, simple to develop, as the desired record and searching of records is conventional. This project helps client to manage its various transactions and information related to its clients and management system.

#### v) Flaws in Present System

The existing system is manual and has many flaws. The system did fulfill the requirement of the earlier time buttoday with over population and the bulky records the system has aged off.

* Less Accurate
* Less efficient
* Possibility in creating mistake in calculations
* Time consuming process.
* Bulk of paper work.

The most important factor is that time and speed greatly affects during working in manual system.

### PROJECT CATEGORY

**ABOUT JAVA**

The pages are made by JAVA, the database is handled by MySQL, and the application .

**TOOLS PLATFORMS, LANGUAGES USED**

Platform used : Microsoft Windows, commonly referred to as Windows, is a group of several proprietary graphical operating system families, all of which are developed and marketed by Microsoft. Each family caters to a certain sector of the computing industry.

Java is a high-level, object-oriented programming language that was developed by James Gosling and his team at Sun Microsystems (which was later acquired by Oracle Corporation). Java is designed to be platform-independent, meaning that it can be run on any computer or operating system that has a Java Virtual Machine (JVM) installed. This makes Java a popular choice for developing web applications, mobile applications, enterprise applications, and many other types of software.

1. Web Applications: Java is widely used in the development of web applications, so you could consider building a web application using Java. This could involve using a framework like Spring or Struts, or you could build your own application from scratch using Java Servlets and JavaServer Pages (JSP).
2. Mobile Applications: Java is also used in the development of mobile applications, particularly on the Android platform. You could consider building an Android app using Java, or you could explore other mobile development frameworks that use Java, such as JavaFX.
3. Desktop Applications: Java can also be used to build desktop applications, either as standalone applications or as applets that run within a web browser. You could consider building a desktop application using a framework like JavaFX or Swing.

1. Data Analysis: Java is also used for data analysis and machine learning, so you could consider building.



MySQL tutorial provides basic and advanced concepts of MySQL. Our MySQL tutorial is designed for beginners and professionals.

MySQL is a relational database management system based on the Structured Query Language, which in the popular language for accessing and managing the records in the database. MySQL is open- source and free software under the GNU license. It is supported by Oracle Company.

Our MySQL tutorial includes all topics of MySQL database that provides for how to manage database and to manipulate data with the help of various SQL queries. These queries are insert records, update records, delete records, select records, create tables, drop tables, etc. There are also given MySQL interview questions to help you better understand the MySQL database.

What is Database?

It is very important to understand the database before learning MySQL. A database is an application that stores the organized collection of records. It can be accessed and manage by the user very easily. It allows us to organize data into tables, rows, columns, and indexes to find the relevant information

very quickly. Each database contains distinct API for performing database operations such as creating, managing, accessing, and searching the data it stores. Today, many databases available like MySQL, Sybase, Oracle, MongoDB, PostgreSQL, SQL Server, etc. In this section, we are going to focus on MySQL mainly.

What is MySQL?

MySQL is currently the most popular database management system software used for managing the relational database. It is open-source database software, which is supported by Oracle Company. It is fast, scalable, and easy to use database management system in comparison with Microsoft SQL Server and Oracle Database. It is commonly used in conjunction with PHP scripts for creating powerful and dynamic server-side or web-based enterprise applications.

**GUI (GRAPHICAL USER INTERFACE)**

The GUI provides picture – oriented interface. Instead of typing command, it provides menus and icons (small representation pictorial images) which the programmer can select by pointing to with a mouse and then clicking one or its buttons. A GUI is software; WINDOWS present programs, procedures, files, commands, etc. as graphical symbols. The user interacts with the computer by manipulating the graphical symbols and menu instead of typing commands. WINDOWS is not an operating system. It can run under DOS. In DOS operating system the programmer has to type commands, when Windows -3 versions were introduced in 1990. Its later version is: WINDOWS -3.1, WINDOWS-3011 AND WINDOWS for Workgroups. Today the operating system WINDOWS

-95 and WINDOWS -98 include WINDOWS graphical interface for their operation.

### SOFTWARE/HARDWARE REQUIREMENT SPECIFICATION

#### Software Specifications

**Operating System:** Microsoft windows 11

**Front End:** JAVA

**Back End:** MySql

***HARDWARE SPECIFICATIONS***

Computer having: -

Processor Intel(R) Core(TM) i5

RAM 8 GB RAM

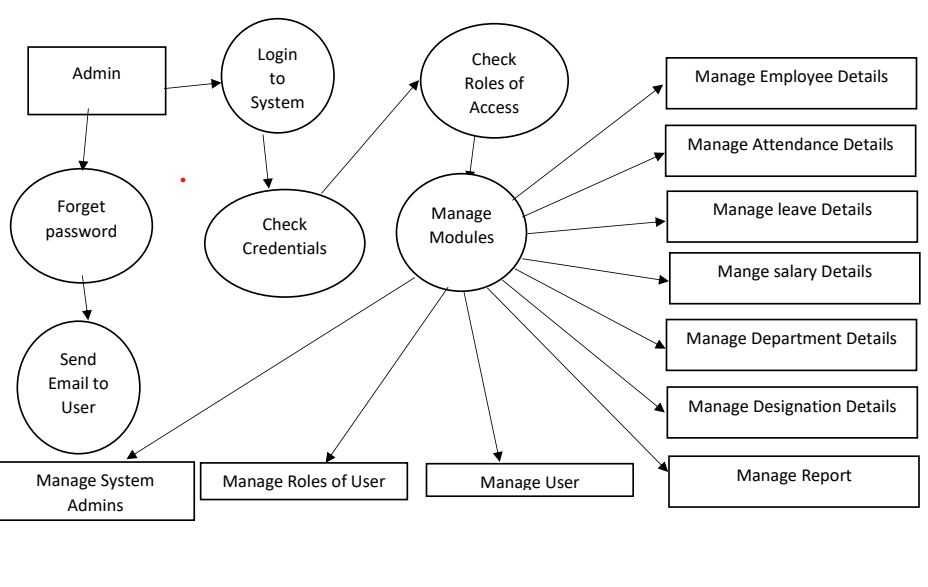
Hard Disk 1 TB SSD

Monitor 14” or above color monitor

Printer Black & White/ Color Printer

Keyboard & Mouse PS2 / USB

#### DETAILED SYSTEM ANALYSIS



1]. Employee module : -

This module contain the employee records. The sub modules are :-

• It contain details of employee who are leaved.

• The out leave module contains details of the employee who came after leave.

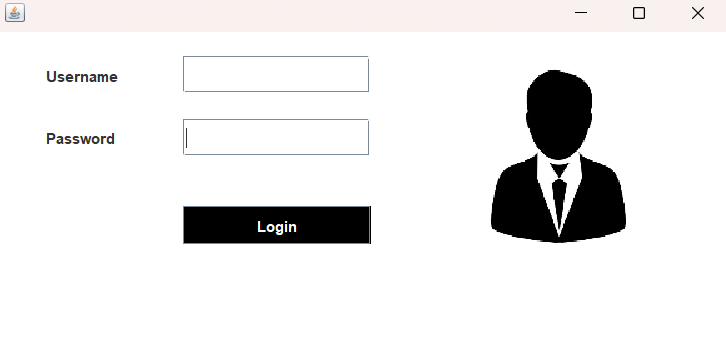
• It contains details of employee salary slip.

2]. Leave module : -

Leave module is used to generate leave reports.

Following are the screens of the Employee Management System where you can see all the featuresof this system in use and you can also see the GUI of the system:

1. Login frame – This is the login frame of this system where user have to enter the requiredcredentials to have access for the main dashboard.



**Data Structures and Tables**

**Data bases**

* **Logindata**
* **Employee**
* **Attendance**
* **Attendence**
* **Salary**

**Table: logindata**  
  
**Columns:**

|  |  |
| --- | --- |
| username | varchar(50) |
| password | varchar(50) |

**Table: employee**  
  
**Columns:**

|  |  |
| --- | --- |
| name | varchar(50) |
| fname | varchar(50) |
| age | varchar(50) |
| dod | varchar(50) |
| address | varchar(50) |
| phone | varchar(10) |
| email | varchar(50) |
| education | varchar(50) |
| designation | varchar(50) |
| aadhar | varchar(50) |
| empID | varchar(50) |

**Table:attendance**  
  
**Columns:**

|  |  |
| --- | --- |
| Eid | varchar(30) |
| name | varchar(50) |
| email | varchar(50) |
| first\_half | varchar(30) |
| second\_half | varchar(30) |
| day\_date | varchar(20) |

**Table: attendence**  
  
**Columns:**

|  |  |
| --- | --- |
| Eid | varchar(30) |
| name | varchar(50) |
| email | varchar(50) |
| first\_half | varchar(30) |
| second\_half | varchar(30) |
| day\_date | varchar(50) |

**Table: salary**  
  
**Columns:**

|  |  |
| --- | --- |
| **Sid** | int AI PK |
| Eid | varchar(10) |
| name | varchar(50) |
| email | varchar(50) |
| hra | float |
| da | float |
| mid | float |
| pf | float |
| basic | float |
| **month\_year** | varchar(50) |

1. **SYSTEM DESIGN CODING**

package Employee\_Management;

import java.awt.Color;

import java.awt.Image;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.sql.ResultSet;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JPasswordField;

import javax.swing.JTextField;

public class Loginpage extends JFrame implements ActionListener{

JTextField tfusername, tfpassword;

Loginpage(){

getContentPane().setBackground(Color.white);

setLayout(null);

JLabel lblusername = new JLabel("Username");

lblusername.setBounds(40, 20, 100, 30);

add(lblusername);

tfusername = new JTextField();

tfusername.setBounds(150, 20, 150, 30);

add(tfusername);

JLabel pass = new JLabel("Password");

pass.setBounds(40, 70, 100, 30);

add(pass);

tfpassword = new JPasswordField();

tfpassword.setBounds(150, 70, 150, 30);

add(tfpassword);

JButton Login = new JButton("Login");

Login.setBounds(150, 140, 150, 30);

Login.setBackground(Color.BLACK);

Login.setForeground(Color.WHITE);

Login.addActionListener(this);

add(Login);

ImageIcon il = new ImageIcon(ClassLoader.getSystemResource("Employee\_Management/icons/second.jpg"));

Image i2 = il.getImage().getScaledInstance(200, 200, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

JLabel image = new JLabel(i3);

image.setBounds(350, 0, 200, 200);

add(image);

setSize(600, 300);

setLocation(450, 200);

setVisible(true);

}

public void actionPerformed(ActionEvent ae){

try{

String username = tfusername.getText();

String password = tfpassword.getText();

Connectionclass c = new Connectionclass();

String query = "select \* from logindata where username = '"+username+"' and password = '"+password+"'";

ResultSet rs = c.stm.executeQuery(query);

if (rs.next()){

setVisible(false);

new HomePage();

}else {

JOptionPane.showMessageDialog(null, "Invalid username or password");

setVisible(false);

}

}catch (Exception e){

e.printStackTrace();

}

}

public static void main(String[] args){

new Loginpage();

}

}

2. Main Dashboard – After login in, user is directed to the main dashboard of

this system where user can perform various operations like adding an

employee, deleting an employee.



package Employee\_Management;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

public class HomePage extends JFrame implements ActionListener{

JLabel l1,l2,l3,l4,l5,l6,l7,l8,l9,l10;

Font f,f1,f2;

JPanel p1;

HomePage(){

super("Employee Home Page");

setLocation(0, 0);

setSize(1550,900);

f=new Font("Lucida Fax", Font.BOLD,20);

f2=new Font("Gadugi",Font.BOLD,35);

f1=new Font("MS UI Gothic",Font.BOLD,18);

ImageIcon il = new ImageIcon(ClassLoader.getSystemResource("Employee\_Management/icons/Im.jpg"));

Image i2 = il.getImage().getScaledInstance(1550, 1000, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

l1=new JLabel(i3);

JMenuBar m1=new JMenuBar();

JMenu men1=new JMenu("Profile");

JMenuItem ment1=new JMenuItem("Complete Profile");

JMenuItem ment2=new JMenuItem("View Profile");

ment1.addActionListener(this);

ment2.addActionListener(this);

JMenu men2=new JMenu("Manage");

JMenuItem ment3=new JMenuItem("Update Details");

ment3.addActionListener(this);

JMenu men3=new JMenu("Attendance");

JMenuItem ment5=new JMenuItem("Take Attendance");

JMenuItem ment6=new JMenuItem("View Attendance");

ment5.addActionListener(this);

ment6.addActionListener(this);

JMenu men5=new JMenu("Salary");

JMenuItem ment9=new JMenuItem("Add Salary");

JMenuItem ment10=new JMenuItem("Generate Salary Slip");

ment9.addActionListener(this);

ment10.addActionListener(this);

JMenu men6=new JMenu("Exit");

JMenuItem ment11=new JMenuItem("Logout");

ment11.addActionListener(this);

JMenu men7=new JMenu("Delete");

JMenuItem ment12=new JMenuItem("Delete Employee");

ment12.addActionListener(this);

men1.add(ment1);

men1.add(ment2);

men2.add(ment3);

men3.add(ment5);

men3.add(ment6);

men5.add(ment9);

men5.add(ment10);

men6.add(ment11);

men7.add(ment12);

m1.add(men1);

m1.add(men2);

m1.add(men3);

m1.add(men5);

m1.add(men7);

m1.add(men6);

men1.setFont(f);

men2.setFont(f);

men3.setFont(f);

men5.setFont(f);

men6.setFont(f);

men7.setFont(f);

m1.setBackground(Color.BLACK);

men1.setForeground(Color.GRAY);

men2.setForeground(Color.GRAY);

men3.setForeground(Color.GRAY);

men5.setForeground(Color.GRAY);

men7.setForeground(Color.GRAY);

men6.setForeground(Color.RED);

ment1.setBackground(Color.BLACK);

ment2.setBackground(Color.BLACK);

ment3.setBackground(Color.BLACK);

ment5.setBackground(Color.BLACK);

ment6.setBackground(Color.BLACK);

ment6.setBackground(Color.BLACK);

ment9.setBackground(Color.BLACK);

ment10.setBackground(Color.BLACK);

ment11.setBackground(Color.BLACK);

ment12.setBackground(Color.BLACK);

ment1.setForeground(Color.YELLOW);

ment2.setForeground(Color.YELLOW);

ment3.setForeground(Color.YELLOW);

ment5.setForeground(Color.YELLOW);

ment6.setForeground(Color.YELLOW);

ment9.setForeground(Color.YELLOW);

ment10.setForeground(Color.YELLOW);

ment11.setForeground(Color.YELLOW);

ment12.setForeground(Color.YELLOW);

setJMenuBar(m1);

add(l1);

setVisible(true);

}

public void actionPerformed(ActionEvent e){

String comnd=e.getActionCommand();

if(comnd.equals("Complete Profile")){

new AddEmployee();

}

else if(comnd.equals("View Profile")){

new View\_Employee();

}

else if(comnd.equals("Update Details")){

new Update\_Details\_Data().setVisible(true);

}

else if(comnd.equals("Take Attendance")){

new Employee\_Attendance().setVisible(true);

}

else if(comnd.equals("View Attendance")){

new View\_Attendance().setVisible(true);

}

else if(comnd.equals("Add Salary")){

new Salary().setVisible(true);

}

else if(comnd.equals("Generate Salary Slip")){

new Generate\_PaySlip().setVisible(true);

}

else if(comnd.equals("Delete Employee")){

new Delete\_Employee().setVisible(true);

}

else if(comnd.equals("Logout")){

System.exit(0);

}

}

public static void main(String args[]){

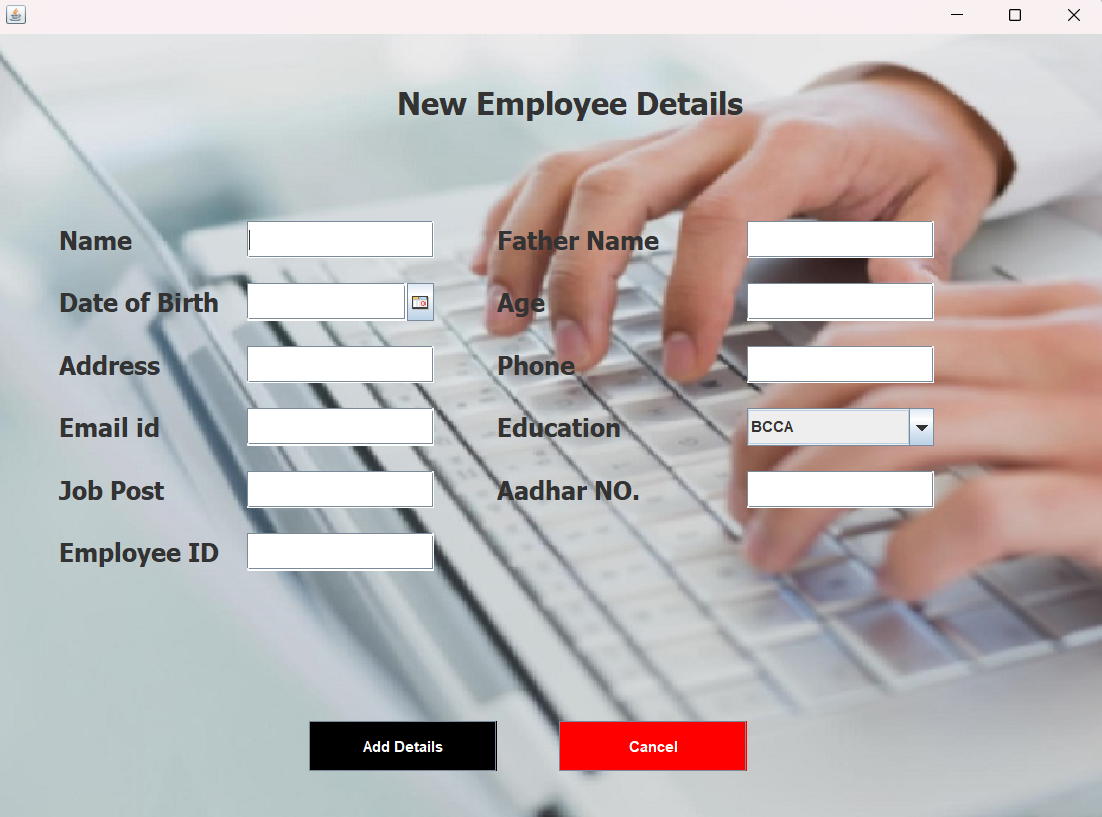
new HomePage();

}

}

3) Add employee – Here user have to enter all the required credentials to add a

new employee to the system.



package Employee\_Management;

import java.awt.\*;

import javax.swing.\*;

import com.toedter.calendar.JDateChooser;

import java.awt.event.\*;

import java.util.\*;

public class AddEmployee extends JFrame implements ActionListener{

Random ran = new Random();

int number = ran.nextInt(999999);

JTextField tfname, tffname, tfaddress, tfphone, tfaadhar, tfemail, tfage, tfdesignation,tfempid ;

JDateChooser dcdob;

JComboBox cbeducation;

JLabel lblempID,b1;

JButton add, back;

AddEmployee() {

getContentPane().setBackground(Color.white);

setLayout(null);

ImageIcon il = new ImageIcon(ClassLoader.getSystemResource("Employee\_Management/icons/add\_employee.jpg"));

Image i2 = il.getImage().getScaledInstance(1120, 630, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

JLabel image = new JLabel(i3);

image.setBounds(0, 0, 1120, 630);

add(image);

JLabel heading = new JLabel("New Employee Details");

heading.setBounds(320, 30, 500, 50);

heading.setFont(new Font("TAHOMA", Font.BOLD,25));

image.add(heading);

JLabel labelname = new JLabel("Name");

labelname.setBounds(50, 150, 150, 30);

labelname.setFont(new Font("TAHOMA", Font.BOLD,20));

image.add(labelname);

tfname = new JTextField();

tfname.setBounds(200, 150, 150, 30);

image.add(tfname);

JLabel labelfname = new JLabel("Father Name");

labelfname.setBounds(400, 150, 150, 30);

labelfname.setFont(new Font("TAHOMA", Font.BOLD,20));

image.add(labelfname);

tffname = new JTextField();

tffname.setBounds(600, 150, 150, 30);

image.add(tffname);

JLabel labelsalary = new JLabel("Age");

labelsalary.setBounds(400, 200, 150, 30);

labelsalary.setFont(new Font("TAHOMA", Font.BOLD,20));

image.add(labelsalary);

tfage = new JTextField();

tfage.setBounds(600, 200, 150, 30);

image.add(tfage);

JLabel labeldob = new JLabel("Date of Birth");

labeldob.setBounds(50, 200, 150, 30);

labeldob.setFont(new Font("TAHOMA", Font.BOLD,20));

image.add(labeldob);

dcdob = new JDateChooser();

dcdob.setBounds(200, 200, 150, 30);

image.add(dcdob);

JLabel labeladdress = new JLabel("Address");

labeladdress.setBounds(50, 250, 150, 30);

labeladdress.setFont(new Font("TAHOMA", Font.BOLD,20));

image.add(labeladdress);

tfaddress = new JTextField();

tfaddress.setBounds(200, 250, 150, 30);

image.add(tfaddress);

JLabel labelphone = new JLabel("Phone");

labelphone.setBounds(400, 250, 150, 30);

labelphone.setFont(new Font("TAHOMA", Font.BOLD,20));

image.add(labelphone);

tfphone = new JTextField();

tfphone.setBounds(600, 250, 150, 30);

image.add(tfphone);

JLabel labelemail = new JLabel("Email id");

labelemail.setBounds(50, 300, 150, 30);

labelemail.setFont(new Font("TAHOMA", Font.BOLD,20));

image.add(labelemail);

tfemail = new JTextField();

tfemail.setBounds(200, 300, 150, 30);

image.add(tfemail);

JLabel labeleducation = new JLabel("Education");

labeleducation.setBounds(400, 300, 150, 30);

labeleducation.setFont(new Font("TAHOMA", Font.BOLD,20));

image.add(labeleducation);

String courses[] = {"BCCA", "BCA", "B.Com", "BA", "BTech", "BSC", "MSC", "MBA", "MCA", "MA", "MTech"};

cbeducation = new JComboBox(courses);

cbeducation.setBounds(600, 300, 150, 30);

image.add(cbeducation);

JLabel labeldesignation = new JLabel("Job Post");

labeldesignation.setBounds(50, 350, 150, 30);

labeldesignation.setFont(new Font("TAHOMA", Font.BOLD,20));

image.add(labeldesignation);

tfdesignation = new JTextField();

tfdesignation.setBounds(200, 350, 150, 30);

image.add(tfdesignation);

JLabel labelaadhar = new JLabel("Aadhar NO.");

labelaadhar.setBounds(400, 350, 150, 30);

labelaadhar.setFont(new Font("TAHOMA", Font.BOLD,20));

image.add(labelaadhar);

tfaadhar = new JTextField();

tfaadhar.setBounds(600, 350, 150, 30);

image.add(tfaadhar);

JLabel labelempID = new JLabel("Employee ID");

labelempID.setBounds(50, 400, 150, 30);

labelempID.setFont(new Font("TAHOMA", Font.BOLD,20));

image.add(labelempID);

tfempid = new JTextField();

tfempid.setBounds(200,400, 150, 30);

image.add(tfempid);

add = new JButton("Add Details");

add.setBounds(250, 550, 150, 40);

add.addActionListener(this);

add.setBackground(Color.BLACK);

add.setForeground(Color.WHITE);

image.add(add);

back = new JButton("Cancel");

back.setBounds(450, 550, 150, 40);

back.addActionListener(this);

back.setBackground(Color.RED);

back.setForeground(Color.WHITE);

image.add(back);

setSize(900, 700);

setLocation(300, 50);

setVisible(true);

}

public void actionPerformed(ActionEvent ae){

if (ae.getSource() == add){

String name = tfname.getText();

String fname = tffname.getText();

String dod = ((JTextField) dcdob.getDateEditor().getUiComponent()).getText();

String age = tfage.getText();

String address = tfaddress.getText();

String phone = tfphone.getText();

String email = tfemail.getText();

String education = (String)cbeducation.getSelectedItem();

String designation = tfdesignation.getText();

String aadhar = tfaadhar.getText();

String empID = tfempid.getText();

try{

Connectionclass conn = new Connectionclass();

String query = "insert into employee values('"+name+"', '"+fname+"', '"+age+"', '"+dod+"', '"+address+"', '"+phone+"', '"+email+"', '"+education+"', '"+designation+"', '"+aadhar+"', '"+empID+"')";

conn.stm.executeUpdate(query);

JOptionPane.showMessageDialog(null, "Detail added Successfully");

setVisible(false);

new HomePage();

}catch (Exception e){

e.printStackTrace();

}

}if(ae.getSource() == back) {

setVisible(false);

new HomePage();

}

}

public static void main(String[] args){

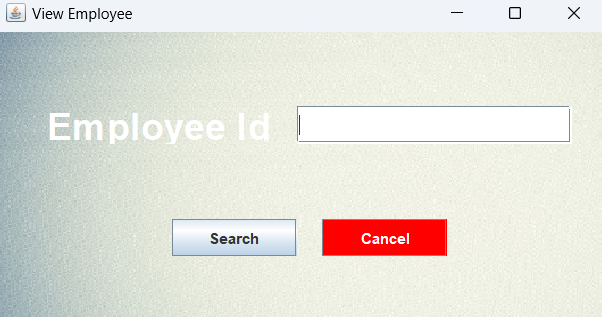
new AddEmployee();

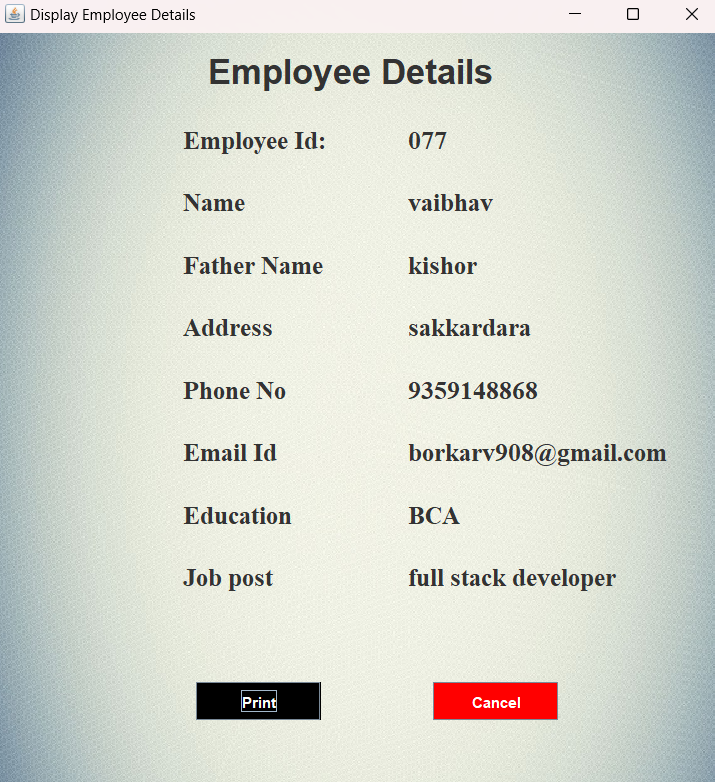
}

}

1. View and update employee – In order to view and update employee

information, the user have to enter employee ID.





package Employee\_Management;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

public class View\_Employee extends JFrame implements ActionListener{

JFrame f;

JTextField t;

JLabel l1,l2;

JButton b,b2;

View\_Employee(){

f=new JFrame("View Employee");

f.setBackground(Color.GREEN);

f.setLayout(null);

l1=new JLabel();

l1.setBounds(0,0,500,270);

l1.setLayout(null);

ImageIcon il = new ImageIcon(ClassLoader.getSystemResource("Employee\_Management/icons/print.jpg"));

Image i2 = il.getImage().getScaledInstance(700, 370, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

l1.setIcon(i3);

l2=new JLabel("Employee Id");

l2.setVisible(true);

l2.setBounds(40,60,250,30);

l2.setForeground(Color.white);

Font F1 =new Font("Airal",Font.BOLD,30);

l2.setFont(F1);

l1.add(l2);

f.add(l1);

t=new JTextField();

t.setBounds(240,60,220,30);

l1.add(t);

b=new JButton("Search");

b.setBounds(140,150,100,30);

b.addActionListener(this);

l1.add(b);

b2=new JButton("Cancel");

b2.setBackground(Color.red);

b2.setForeground(Color.white);

b2.setBounds(260,150,100,30);

b2.addActionListener(this);

l1.add(b2);

f.setSize(500, 270);

f.setLocation(450, 250);

f.setVisible(true);

}

public void actionPerformed(ActionEvent e){

if(e.getSource()==b){

f.setVisible(false);

new View\_Employee\_Data(t.getText());

}

if(e.getSource()==b2){

f.setVisible(false);

new HomePage();

}

}

public static void main(String[] args){

new View\_Employee();

}

}

package Employee\_Management;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JFrame;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

public class View\_Employee\_Data extends JFrame implements ActionListener{

JFrame f;

JLabel id8,id,id1,aid,aid1,id2,aid2,id3,aid3,id4,aid4,id5,aid5,id6,aid6,id7,aid7,aid8,id9;

String emp\_id,name,father,address,phone,email,education,post,age,dob,aadhar;

JButton b1,b2;

ImageIcon icon;

View\_Employee\_Data(String empID){

try{

Connectionclass obj = new Connectionclass();

String s = "select \* from employee where empID='"+empID+"'";

ResultSet rs=obj.stm.executeQuery(s);

while(rs.next()){

name=rs.getString("name");

father=rs.getString("fname");

age=rs.getString("age");

address=rs.getString("address");

phone=rs.getString("Phone");

email=rs.getString("email");

education=rs.getString("Education");

post=rs.getString("designation");

aadhar=rs.getString("aadhar");

emp\_id=rs.getString("empID");

}

}

catch(Exception ex){

ex.printStackTrace();

}

f=new JFrame("Display Employee Details");

f.setVisible(true);

f.setSize(595,642);

f.setLocation(450,200);

f.setBackground(Color.white);

f.setLayout(null);

id9=new JLabel();

id9.setBounds(0,0,595,642);

id9.setLayout(null);

ImageIcon img=new ImageIcon(ClassLoader.getSystemResource("Employee\_Management/icons/print.jpg"));

id9.setIcon(img);

id8=new JLabel("Employee Details");

id8.setBounds(170,10,250,40);

f.add(id8);

id8.setFont(new Font("Airal",Font.BOLD,28));

id9.add(id8);

f.add(id9);

id=new JLabel("Employee Id:");

id.setBounds(150,70,120,30);

id.setFont(new Font("serif",Font.BOLD,20));

id9.add(id);

aid=new JLabel(emp\_id);

aid.setBounds(330,70,200,30);

aid.setFont(new Font("serif",Font.BOLD,20));

id9.add(aid);

id1=new JLabel("Name");

id1.setBounds(150,120,100,30);

id1.setFont(new Font("serif",Font.BOLD,20));

id9.add(id1);

aid1=new JLabel(name);

aid1.setBounds(330,120,300,30);

aid1.setFont(new Font("serif",Font.BOLD,20));

id9.add(aid1);

id2=new JLabel("Father Name");

id2.setBounds(150,170,200,30);

id2.setFont(new Font("serif",Font.BOLD,20));

id9.add(id2);

aid2=new JLabel(father);

aid2.setBounds(330,170,300,30);

aid2.setFont(new Font("serif",Font.BOLD,20));

id9.add(aid2);

id3=new JLabel("Address");

id3.setBounds(150,220,100,30);

id3.setFont(new Font("serif",Font.BOLD,20));

id9.add(id3);

aid3=new JLabel(address);

aid3.setBounds(330,220,300,30);

aid3.setFont(new Font("serif",Font.BOLD,20));

id9.add(aid3);

id4=new JLabel("Phone No");

id4.setBounds(150,270,100,30);

id4.setFont(new Font("serif",Font.BOLD,20));

id9.add(id4);

aid4=new JLabel(phone);

aid4.setBounds(330,270,300,30);

aid4.setFont(new Font("serif",Font.BOLD,20));

id9.add(aid4);

id5=new JLabel("Email Id");

id5.setBounds(150,320,100,30);

id5.setFont(new Font("serif",Font.BOLD,20));

id9.add(id5);

aid5=new JLabel(email);

aid5.setBounds(330,320,300,30);

aid5.setFont(new Font("serif",Font.BOLD,20));

id9.add(aid5);

id6=new JLabel("Education");

id6.setBounds(150,370,100,30);

id6.setFont(new Font("serif",Font.BOLD,20));

id9.add(id6);

aid6=new JLabel(education);

aid6.setBounds(330,370,300,30);

aid6.setFont(new Font("serif",Font.BOLD,20));

id9.add(aid6);

id7=new JLabel("Job post");

id7.setBounds(150,420,100,30);

id7.setFont(new Font("serif",Font.BOLD,20));

id9.add(id7);

aid7=new JLabel(post);

aid7.setBounds(330,420,300,30);

aid7.setFont(new Font("serif",Font.BOLD,20));

id9.add(aid7);

b1 = new JButton("Print");

b1.setBounds(160, 520, 100, 30);

b1.addActionListener(this);

b1.setBackground(Color.BLACK);

b1.setForeground(Color.WHITE);

id9.add(b1);

b2 = new JButton("Cancel");

b2.setBounds(350, 520, 100, 30);

b2.addActionListener(this);

b2.setBackground(Color.RED);

b2.setForeground(Color.WHITE);

id9.add(b2);

}

public void actionPerformed(ActionEvent e){

if(e.getSource()==b1){

JOptionPane.showMessageDialog(null, "printed Successfully");

f.setVisible(false);

new HomePage();

}

if(e.getSource()==b2){

f.setVisible(false);

new View\_Employee();

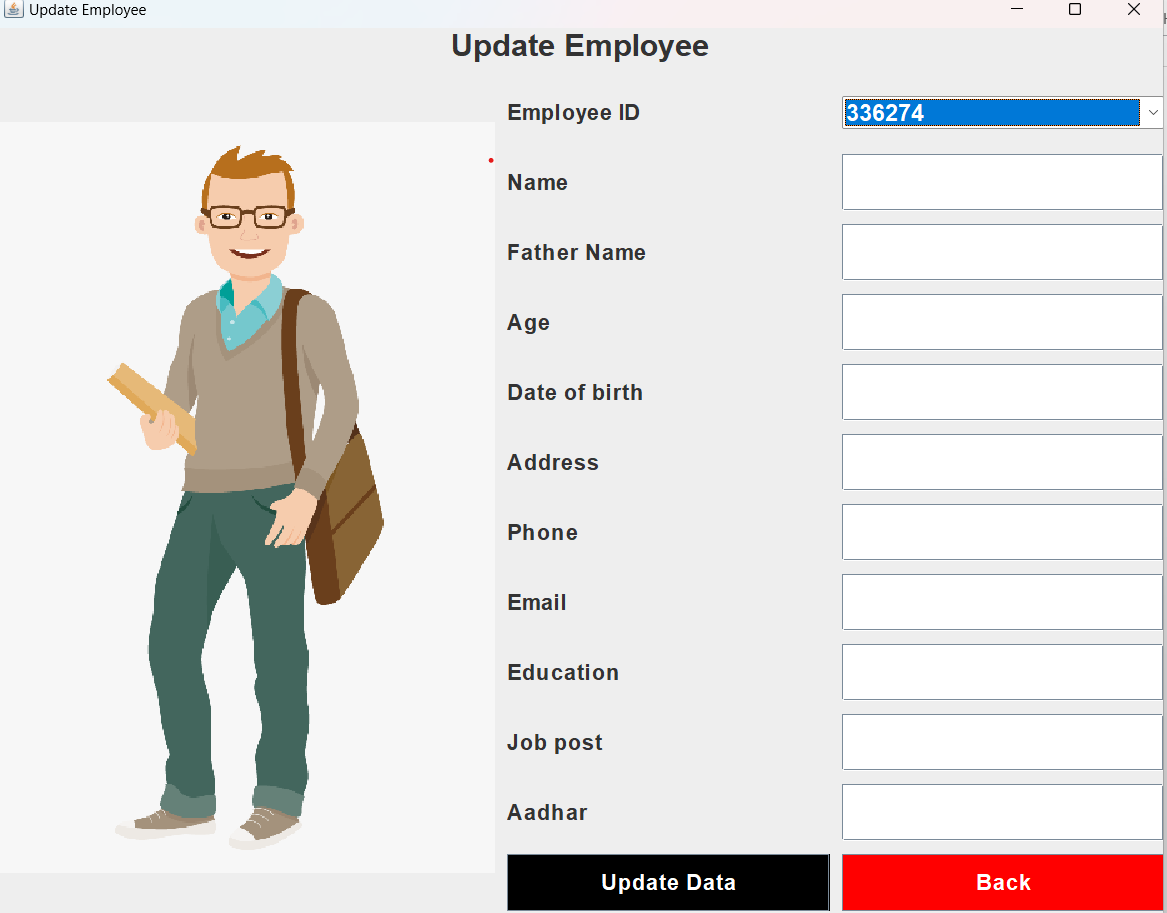
}

}

}

1. Update employee – In order to view and update employee

information, the user have to enter employee ID.



package Employee\_Management;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JFrame;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

public class Update\_Details\_Data extends JFrame implements ActionListener{

JLabel l1,l2,l3,l4,l5,l6,l7,l8,l9,l10,l11,l12,l13;

JButton b1,b2;

JPanel p1,p2,p3;

JTextField tf1, tf10, tf2, tf3, tf4, tf5, tf6, tf7,tf8,tf9;

Font f,f1;

Choice ch;

Update\_Details\_Data(){

super("Update Employee");

setLocation(450,50);

setSize(950,750);

f=new Font("Arial",Font.BOLD,25);

f1=new Font("Arial",Font.BOLD,18);

ch=new Choice();

try {

Connectionclass obj = new Connectionclass();

String q = "select empID from employee";

ResultSet rest=obj.stm.executeQuery(q);

while(rest.next())

{

ch.add(rest.getString("empID"));

}

}

catch(Exception e){

e.printStackTrace();

}

l1=new JLabel("Update Employee");

l2=new JLabel("Name");

l3=new JLabel("Father Name");

l4=new JLabel("Age");

l5=new JLabel("Date of birth");

l6=new JLabel("Address");

l7=new JLabel("Phone");

l8=new JLabel("Email");

l9=new JLabel("Education");

l10=new JLabel("Job post");

l11=new JLabel("Aadhar");

l12=new JLabel("Employee ID");

tf1=new JTextField();

tf2=new JTextField();

tf3=new JTextField();

tf4=new JTextField();

tf5=new JTextField();

tf6=new JTextField();

tf7=new JTextField();

tf8=new JTextField();

tf9=new JTextField();

tf10=new JTextField();

b1=new JButton("Update Data");

b2=new JButton("Back");

l1.setHorizontalAlignment(JLabel.CENTER);

b1.addActionListener(this);

b2.addActionListener(this);

l1.setFont(f);

l2.setFont(f1);

l3.setFont(f1);

l4.setFont(f1);

l5.setFont(f1);

l6.setFont(f1);

l7.setFont(f1);

l8.setFont(f1);

l9.setFont(f1);

l10.setFont(f1);

l11.setFont(f1);

l12.setFont(f1);

ch.setFont(f1);

tf1.setFont(f1);

tf2.setFont(f1);

tf3.setFont(f1);

tf4.setFont(f1);

tf5.setFont(f1);

tf6.setFont(f1);

tf7.setFont(f1);

tf8.setFont(f1);

tf9.setFont(f1);

tf10.setFont(f1);

b1.setFont(f1);

b2.setFont(f1);

b1.setBackground(Color.BLACK);

b2.setBackground(Color.red);

b1.setForeground(Color.WHITE);

b2.setForeground(Color.WHITE);

p1=new JPanel();

p1.setLayout(new GridLayout(1,1,10,10));

p1.add(l1);

p2=new JPanel();

p2.setLayout(new GridLayout(12,2,10,10));

p2.add(l12);

p2.add(ch);

p2.add(l2);

p2.add(tf1);

p2.add(l3);

p2.add(tf2);

p2.add(l4);

p2.add(tf3);

p2.add(l5);

p2.add(tf4);

p2.add(l6);

p2.add(tf5);

p2.add(l7);

p2.add(tf6);

p2.add(l8);

p2.add(tf7);

p2.add(l9);

p2.add(tf8);

p2.add(l10);

p2.add(tf9);

p2.add(l11);

p2.add(tf10);

p2.add(b1);

p2.add(b2);

p3=new JPanel();

p3.setLayout(new GridLayout(1,1,10,10));

ImageIcon il = new ImageIcon(ClassLoader.getSystemResource("Employee\_Management/icons/empng.png"));

Image i2 = il.getImage().getScaledInstance(400, 600, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

l13=new JLabel(i3);

p3.add(l13);

setLayout(new BorderLayout(10,10));

add(p1,"North");

add(p2,"Center");

add(p3,"West");

ch.addMouseListener(new MouseAdapter ()

{

public void mouseClicked(MouseEvent arg0)

{

try

{

Connectionclass obj2 = new Connectionclass();

String empID=ch.getSelectedItem();

String q1 = "select \* from employee where empID='"+empID+"'";

ResultSet rest1=obj2.stm.executeQuery(q1);

while(rest1.next()){

tf1.setText(rest1.getString("name"));

tf2.setText(rest1.getString("fname"));

tf3.setText(rest1.getString("age"));

tf4.setText(rest1.getString("dod"));

tf5.setText(rest1.getString("address"));

tf6.setText(rest1.getString("phone"));

tf7.setText(rest1.getString("email"));

tf8.setText(rest1.getString("education"));

tf9.setText(rest1.getString("designation"));

tf10.setText(rest1.getString("aadhar"));

}

}

catch(Exception exx){

exx.printStackTrace();

}

}

});

}

public void actionPerformed(ActionEvent e){

String id=ch.getSelectedItem();

String name=tf1.getText();

String fname=tf2.getText();

String age=tf3.getText();

String dod=tf4.getText();

String address=tf5.getText();

String phone=tf6.getText();

String email=tf7.getText();

String education=tf8.getText();

String designation=tf9.getText();

String aadhar=tf10.getText();

if(e.getSource()==b1){

try{

Connectionclass conn = new Connectionclass();

String q1 = "update employee set name ='"+name+"', fname = '"+fname+"', age = '"+age+"', dod = '"+dod+"', address = '"+address+"', phone = '"+phone+"', email = '"+email+"', education = '"+education+"', designation = '"+designation+"', aadhar = '"+aadhar+"' where empID = '"+id+"'";

int aa=conn.stm.executeUpdate(q1);

if(aa == 1)

{

JOptionPane.showMessageDialog(null, "Data Successfully Updated");

this.setVisible(false);

new Update\_Details\_Data();

}

else{

JOptionPane.showMessageDialog(null, "Please fill all details carefully");

}

}

catch(Exception ee){

ee.printStackTrace();

}

}if(e.getSource() == b2) {

this.setVisible(false);

}

}

public static void main(String[] args){

new Update\_Details\_Data().setVisible(true);

}

}

package Employee\_Management;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JFrame;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

public class Employee\_Attendance extends JFrame implements ActionListener{

JLabel l1,l2,l3,l4,l5;

Choice c1,c2,c3;

JButton bt1,bt2;

JTextField tf1,tf2;

Font f;

JPanel p;

Employee\_Attendance(){

super("Employee Attendance");

setLocation(300,200);

setSize(450,300);

setResizable(false);

f=new Font("senserif",Font.BOLD,15);

l1=new JLabel("Select Employee ID");

l2=new JLabel("First Half");

l3=new JLabel("Second Half");

l4=new JLabel("Name");

l5=new JLabel("Email");

l1.setFont(f);

l2.setFont(f);

l3.setFont(f);

l4.setFont(f);

l5.setFont(f);

tf1=new JTextField();

tf2=new JTextField();

tf1.setFont(f);

tf2.setFont(f);

tf1.setEditable(false);

tf2.setEditable(false);

c2=new Choice();

c2.add("Present");

c2.add("Absent");

c3=new Choice();

c3.add("Present");

c3.add("Absent");

c2.setFont(f);

c3.setFont(f);

bt1=new JButton("Submit");

bt2=new JButton("Close");

bt1.setBackground(Color.BLACK);

bt1.setForeground(Color.WHITE);

bt2.setBackground(Color.BLACK);

bt2.setForeground(Color.WHITE);

bt1.setFont(f);

bt2.setFont(f);

bt1.addActionListener(this);

bt2.addActionListener(this);

c1=new Choice();

try {

Connectionclass obj = new Connectionclass();

String q = "select \* from employee";

ResultSet rest=obj.stm.executeQuery(q);

while(rest.next())

{

c1.add(rest.getString("empID"));

}

}

catch(Exception ex)

{

ex.printStackTrace();

}

p=new JPanel();

p.setLayout(new GridLayout(6,2,10,10));

p.add(l1);

p.add(c1);

p.add(l4);

p.add(tf1);

p.add(l5);

p.add(tf2);

p.add(l2);

p.add(c2);

p.add(l3);

p.add(c3);

p.add(bt1);

p.add(bt2);

add(p);

c1.addMouseListener(new MouseAdapter()

{

@Override

public void mouseClicked(MouseEvent arg0)

{

try

{

Connectionclass obj2 = new Connectionclass();

String eid=c1.getSelectedItem();

String q2 = "select \* from employee where empID='"+eid+"'";

ResultSet rest1=obj2.stm.executeQuery(q2);

while(rest1.next()){

tf1.setText(rest1.getString("name"));

tf2.setText(rest1.getString("email"));

}

}

catch(Exception exx){

exx.printStackTrace();

}

}

});

}

public void actionPerformed(ActionEvent ev){

if(ev.getSource()==bt1)

{

String ch\_eid=c1.getSelectedItem();

String ch\_first\_half=c2.getSelectedItem();

String ch\_second\_half=c3.getSelectedItem();

String name=tf1.getText();

String email=tf2.getText();

String dt=new java.util.Date().toString();

try{

Connectionclass obj1 = new Connectionclass();

String q1="insert into attendence values('"+ch\_eid+"','"+name+"','"+email+"','"+ch\_first\_half+"','"+ch\_second\_half+"','"+dt+"')";

obj1.stm.executeUpdate(q1);

JOptionPane.showMessageDialog(null, "Data inserted");

setVisible(false);

}

catch(Exception ex1){

ex1.printStackTrace();

}

}if(ev.getSource() == bt2) {

JOptionPane.showMessageDialog(null, "Are you sure to close?");

setVisible(false);

}

}

public static void main(String[] args){

new Employee\_Attendance().setVisible(true);

}

}



**All employee attendance record**

package Employee\_Management;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JFrame;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

public class View\_Attendance extends JFrame implements ActionListener{

JTable t;

JButton bt1;

JTextField tf1;

JPanel p1,p2,p3;

String x[]={"Employee ID","Name","Email","First Haif","Second Haif","date"};

String y[][]=new String [20][6];

int i=0,j=0;

Font f,f1;

JLabel l1,l2;

View\_Attendance(){

super("All Employee Attendance records");

setLocation(0,10);

setSize(1480,400);

f=new Font("MS UI Gothic",Font.BOLD,17);

try

{

Connectionclass obj = new Connectionclass();

String q = "select \* from attendence";

ResultSet rest=obj.stm.executeQuery(q);

while(rest.next()){

y[i][j++]=rest.getString("Eid");

y[i][j++]=rest.getString("name");

y[i][j++]=rest.getString("Email");

y[i][j++]=rest.getString("first\_half");

y[i][j++]=rest.getString("second\_half");

y[i][j++]=rest.getString("day\_date");

i++;

j=0;

}

t=new JTable(y,x);

}

catch(Exception ex){

ex.printStackTrace();

}

t.setBackground(Color.BLACK);

t.setForeground(Color.WHITE);

t.setFont(f);

JScrollPane js=new JScrollPane(t);

f1=new Font("Lucida Fax",Font.BOLD,25);

l1=new JLabel("Secarch any Employee");

l1.setHorizontalAlignment(JLabel.CENTER);

l1.setFont(f1);

l1.setForeground(Color.YELLOW);

l2=new JLabel("Employee ID");

l2.setFont(f1);

l2.setForeground(Color.GRAY);

tf1=new JTextField();

bt1=new JButton("Search Employee");

tf1.setFont(f);

bt1.setFont(f);

bt1.setBackground(Color.BLACK);

bt1.setForeground(Color.red);

bt1.addActionListener(this);

p1=new JPanel();

p1.setLayout(new GridLayout(1,1,10,10));

p1.add(l1);

p2=new JPanel();

p2.setLayout(new GridLayout(1,3,10,10));

p2.add(l2);

p2.add(tf1);

p2.add(bt1);

p3=new JPanel();

p3.setLayout(new GridLayout(2,1,10,10));

p3.add(p1);

p3.add(p2);

p1.setBackground(Color.BLACK);

p2.setBackground(Color.BLACK);

p3.setBackground(Color.BLACK);

add(p3,"South");

add(js);

}

public void actionPerformed(ActionEvent e){

String eid=tf1.getText();

if(e.getSource()==bt1)

{

new View\_Attendance\_single(eid).setVisible(true);

}

}

public static void main(String[] args){

new View\_Attendance().setVisible(true);

}

}

**Add all employee salaries report**

package Employee\_Management;

import java.awt.event.ActionListener;

import javax.swing.JFrame;

import javax.swing.JFrame;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

public class Salary extends JFrame implements ActionListener{

JLabel l1,l2,l3,l4,l5,l6,l7,l8,l9,l10,l11,l12;

Font f,f1;

JTextField t1, t2, t3, t4, t5, t6, t7;

Choice ch1,ch2,ch3;

JButton bt1,bt2;

JPanel p1,p2,p3;

Salary(){

super("Salary");

setLocation(100,100);

setSize(950,700);

setResizable(false);

f=new Font("MS UI Gothic",Font.BOLD,18);

f=new Font("Arial",Font.BOLD,18);

f1=new Font("Arial",Font.BOLD,25);

l1=new JLabel("Select Employee ID");

l2=new JLabel("Name");

l3=new JLabel("Email");

l4=new JLabel("HRA");

l5=new JLabel("DA");

l6=new JLabel("MID");

l7=new JLabel("PF");

l8=new JLabel("Basic Salary");

l9=new JLabel("Select Month");

l10=new JLabel("Select Year");

l12=new JLabel("Employee Salary");

l12.setHorizontalAlignment(JLabel.CENTER);

ch1=new Choice();

try {

Connectionclass obj = new Connectionclass();

String q = "select empID from employee";

ResultSet rest=obj.stm.executeQuery(q);

while(rest.next())

{

ch1.add(rest.getString("empID"));

}

}

catch(Exception e){

e.printStackTrace();

}

ch2=new Choice();

ch2.add("January");

ch2.add("February");

ch2.add("March");

ch2.add("April");

ch2.add("May");

ch2.add("June");

ch2.add("July");

ch2.add("August");

ch2.add("September");

ch2.add("October");

ch2.add("November");

ch2.add("December");

ch3=new Choice();

ch3.add("2015");

ch3.add("2016");

ch3.add("2017");

ch3.add("2018");

ch3.add("2019");

ch3.add("2020");

ch3.add("2021");

ch3.add("2022");

ch3.add("2023");

ch3.add("2024");

l1.setFont(f);

l2.setFont(f);

l3.setFont(f);

l4.setFont(f);

l5.setFont(f);

l6.setFont(f);

l7.setFont(f);

l8.setFont(f);

l9.setFont(f);

l10.setFont(f);

l12.setFont(f1);

ch1.setFont(f);

ch2.setFont(f);

ch3.setFont(f);

t1=new JTextField();

t2=new JTextField();

t3=new JTextField();

t4=new JTextField();

t5=new JTextField();

t6=new JTextField();

t7=new JTextField();

t1.setFont(f);

t2.setFont(f);

t3.setFont(f);

t4.setFont(f);

t5.setFont(f);

t6.setFont(f);

t7.setFont(f);

t1.setEditable(false);

t2.setEditable(false);

ImageIcon img = new ImageIcon(ClassLoader.getSystemResource("Employee\_Management/icons/empng.png"));

Image image = img.getImage().getScaledInstance(400, 600, Image.SCALE\_DEFAULT);

ImageIcon img2 = new ImageIcon(image);

l11=new JLabel(img2);

bt1=new JButton("Submit");

bt2=new JButton("Close");

bt1.setFont(f);

bt2.setFont(f);

bt1.setBackground(Color.BLACK);

bt1.setForeground(Color.WHITE);

bt2.setBackground(Color.BLACK);

bt2.setForeground(Color.WHITE);

bt1.addActionListener(this);

bt2.addActionListener(this);

p1=new JPanel();

p1.setLayout(new GridLayout(11,2,10,10));

p1.add(l1);

p1.add(ch1);

p1.add(l2);

p1.add(t1);

p1.add(l3);

p1.add(t2);

p1.add(l4);

p1.add(t3);

p1.add(l5);

p1.add(t4);

p1.add(l6);

p1.add(t5);

p1.add(l7);

p1.add(t6);

p1.add(l8);

p1.add(t7);

p1.add(l9);

p1.add(ch2);

p1.add(l10);

p1.add(ch3);

p1.add(bt1);

p1.add(bt2);

p2=new JPanel();

p2.setLayout(new GridLayout(1,1,10,10));

p2.add(l11);

p3=new JPanel();

p3.setLayout(new GridLayout(1,1,10,10));

p3.add(l12);

setLayout(new BorderLayout(30,30));

add(p1,"Center");

add(p2,"West");

add(p3,"North");

ch1.addMouseListener(new MouseAdapter ()

{

@Override

public void mouseClicked(MouseEvent arg0)

{

try

{

Connectionclass obj2 = new Connectionclass();

String eid2=ch1.getSelectedItem();

String q3 = "select \* from employee where empID='"+eid2+"'";

ResultSet rest1=obj2.stm.executeQuery(q3);

while(rest1.next()){

t1.setText(rest1.getString("name"));

t2.setText(rest1.getString("email"));

}

}

catch(Exception exx){

exx.printStackTrace();

}

}

});

}

public void actionPerformed(ActionEvent e){

if(e.getSource()==bt1){

String empID=ch1.getSelectedItem();

String name=t1.getText();

String email=t2.getText();

float hra=Float.parseFloat(t3.getText());

float da=Float.parseFloat(t4.getText());

float mid=Float.parseFloat(t5.getText());

float pf=Float.parseFloat(t6.getText());

float basic=Float.parseFloat(t7.getText());

String month=ch2.getSelectedItem()+" "+ch3.getSelectedItem();

try{

Connectionclass obj1 = new Connectionclass();

String q1 = "insert into salary values('"+0+"','"+empID+"','"+name+"','"+email+"','"+hra+"','"+da+"','"+mid+"','"+pf+"','"+basic+"','"+month+"')";

int aa=obj1.stm.executeUpdate(q1);

if(aa==1){

JOptionPane.showMessageDialog(null, "Data inserted");

this.setVisible(false);

}

else{

JOptionPane.showMessageDialog(null, "Please !, Fill all details carefully");

this.setVisible(false);

this.setVisible(true);

}

}

catch(Exception exx){

exx.printStackTrace();

}

}

if(e.getSource() == bt2) {

JOptionPane.showMessageDialog(null, "Are you sure to close?");

setVisible(false);

}

}

public static void main(String[] args)

{

new Salary().setVisible(true);

}

}

**Delete any Employee**

**package Employee\_Management;**

**import java.awt.event.ActionEvent;**

**import java.awt.event.ActionListener;**

**import javax.swing.JFrame;**

**import javax.swing.\*;**

**import java.awt.\*;**

**import java.awt.event.\*;**

**import java.sql.\*;**

**public class Delete\_Employee extends JFrame implements ActionListener{**

**JTable t;**

**JButton bt1;**

**JTextField tf1;**

**JPanel p1,p2,p3;**

**String x[]={"Employee ID","Name","Email","Age","Date of Birth","Post"};**

**String y[][]=new String [20][6];**

**int i=0,j=0;**

**Font f,f1;**

**JLabel l1,l2;**

**Delete\_Employee(){**

**super("Delete any Employee");**

**setLocation(0,10);**

**setSize(1480,400);**

**f=new Font("MS UI Gothic",Font.BOLD,17);**

**try**

**{**

**Connectionclass obj = new Connectionclass();**

**String q = "select \* from employee";**

**ResultSet rest=obj.stm.executeQuery(q);**

**while(rest.next()){**

**y[i][j++]=rest.getString("empID");**

**y[i][j++]=rest.getString("name");**

**y[i][j++]=rest.getString("Email");**

**y[i][j++]=rest.getString("Age");**

**y[i][j++]=rest.getString("dod");**

**y[i][j++]=rest.getString("designation");**

**i++;**

**j=0;**

**}**

**t=new JTable(y,x);**

**}**

**catch(Exception ex){**

**ex.printStackTrace();**

**}**

**t.setBackground(Color.BLACK);**

**t.setForeground(Color.WHITE);**

**t.setFont(f);**

**JScrollPane js=new JScrollPane(t);**

**f1=new Font("Lucida Fax",Font.BOLD,25);**

**l1=new JLabel("Delete any Employee");**

**l1.setHorizontalAlignment(JLabel.CENTER);**

**l1.setFont(f1);**

**l1.setForeground(Color.YELLOW);**

**l2=new JLabel("Employee ID");**

**l2.setFont(f1);**

**l2.setForeground(Color.GRAY);**

**tf1=new JTextField();**

**bt1=new JButton("Delete Employee");**

**tf1.setFont(f);**

**bt1.setFont(f);**

**bt1.setBackground(Color.BLACK);**

**bt1.setForeground(Color.red);**

**bt1.addActionListener(this);**

**p1=new JPanel();**

**p1.setLayout(new GridLayout(1,1,10,10));**

**p1.add(l1);**

**p2=new JPanel();**

**p2.setLayout(new GridLayout(1,3,10,10));**

**p2.add(l2);**

**p2.add(tf1);**

**p2.add(bt1);**

**p3=new JPanel();**

**p3.setLayout(new GridLayout(2,1,10,10));**

**p3.add(p1);**

**p3.add(p2);**

**p1.setBackground(Color.BLACK);**

**p2.setBackground(Color.BLACK);**

**p3.setBackground(Color.BLACK);**

**add(p3,"South");**

**add(js);**

**}**

**public void actionPerformed(ActionEvent e){**

**if(e.getSource()==bt1)**

**{**

**String eid=tf1.getText();**

**try{**

**Connectionclass obj1 = new Connectionclass();**

**String q="delete from employee where empID='"+eid+"'";**

**String q1="delete from attendence where Eid='"+eid+"'";**

**String q2="delete from salary where Eid='"+eid+"'";**

**int res=obj1.stm.executeUpdate(q);**

**if(res==1){**

**JOptionPane.showMessageDialog(null, "Data Successfully deleted");**

**obj1.stm.executeUpdate(q1);**

**obj1.stm.executeUpdate(q2);**

**this.setVisible(false);**

**new Delete\_Employee().setVisible(true);**

**}**

**else{**

**JOptionPane.showMessageDialog(null, "did't delete sorry !!!");**

**this.setVisible(false);**

**new Delete\_Employee().setVisible(true);**

**}**

**}**

**catch(Exception ex){**

**ex.printStackTrace();**

**}**

**}**

**}**

**public static void main(String[] args){**

**new Delete\_Employee().setVisible(true);**

**}**

**}**

**package Employee\_Management;**

**import java.awt.event.ActionListener;**

**import javax.swing.JFrame;**

**import javax.swing.JFrame;**

**import javax.swing.\*;**

**import java.awt.\*;**

**import java.awt.event.\*;**

**import java.sql.\*;**

**public class Generate\_PaySlip extends JFrame implements ActionListener {**

**JLabel l1,l2,l3,l4,l5,l6;**

**Font f;**

**JTextArea ta;**

**Choice ch1,ch2,ch3;**

**JButton bt1,bt2;**

**JPanel p1;**

**Generate\_PaySlip(){**

**super("Generate pay Slip");**

**setLocation(100,100);**

**setSize(500,500);**

**setResizable(false);**

**f=new Font("Arial",Font.BOLD,16);**

**l1=new JLabel("Employee Id");**

**l2=new JLabel("Month");**

**l3=new JLabel("Year");**

**l1.setFont(f);**

**l2.setFont(f);**

**l3.setFont(f);**

**ch1=new Choice();**

**try {**

**Connectionclass obj = new Connectionclass();**

**String q = "select \* from employee";**

**ResultSet rest=obj.stm.executeQuery(q);**

**while(rest.next())**

**{**

**ch1.add(rest.getString("empID"));**

**}**

**}**

**catch(Exception e){**

**e.printStackTrace();**

**}**

**ch2=new Choice();**

**ch2.add("January");**

**ch2.add("February");**

**ch2.add("March");**

**ch2.add("April");**

**ch2.add("May");**

**ch2.add("June");**

**ch2.add("July");**

**ch2.add("August");**

**ch2.add("September");**

**ch2.add("October");**

**ch2.add("November");**

**ch2.add("December");**

**ch3=new Choice();**

**ch3.add("2015");**

**ch3.add("2016");**

**ch3.add("2017");**

**ch3.add("2018");**

**ch3.add("2019");**

**ch3.add("2020");**

**ch3.add("2021");**

**ch3.add("2022");**

**ch3.add("2023");**

**ch3.add("2024");**

**ch1.setFont(f);**

**ch2.setFont(f);**

**ch3.setFont(f);**

**bt1=new JButton("Print");**

**bt1.setBackground(Color.BLACK);**

**bt1.setForeground(Color.WHITE);**

**bt1.addActionListener(this);**

**bt1.setFont(f);**

**bt2=new JButton("Close");**

**bt2.setBackground(Color.BLACK);**

**bt2.setForeground(Color.WHITE);**

**bt2.addActionListener(this);**

**bt2.setFont(f);**

**p1=new JPanel();**

**p1.setLayout(new GridLayout(4,2,10,10));**

**p1.add(l1);**

**p1.add(ch1);**

**p1.add(l2);**

**p1.add(ch2);**

**p1.add(l3);**

**p1.add(ch3);**

**p1.add(bt1);**

**p1.add(bt2);**

**ta=new JTextArea();**

**JScrollPane sp=new JScrollPane(ta);**

**ta.setEditable(false);**

**ta.setFont(f);**

**setLayout(new BorderLayout ());**

**add(sp,"Center");**

**add(p1,"South");**

**}**

**public void actionPerformed(ActionEvent e){**

**if(e.getSource()==bt1){**

**ta.setText("-----------------------pay slip-------------------------");**

**try{**

**Connectionclass obj1 = new Connectionclass();**

**String id=ch1.getSelectedItem();**

**String month\_year=ch2.getSelectedItem()+" "+ch3.getSelectedItem();**

**String q1="select \* from salary where Eid='"+id+"'";**

**ResultSet rest1=obj1.stm.executeQuery(q1);**

**while(rest1.next()){**

**ta.append("\n\nEmployee Id : "+Integer.parseInt(rest1.getString("Eid")));**

**ta.append("\nEmployee Name :"+rest1.getString("name"));**

**ta.append("\nEmployee Name :"+rest1.getString("email"));**

**ta.append("\n-------------------------------------------\n\n");**

**}**

**String q2="select \* from salary where month\_year='"+month\_year+"' and Eid='"+id+"'";**

**ResultSet rest2=obj1.stm.executeQuery(q2);**

**if(rest2.next()==false){**

**ta.append("\n-----------------------------------------------------------\n\n");**

**ta.append("Record Not Found of this Month and year.\n");**

**ta.append("\n-----------------------------or------------------------------\n\n");**

**ta.append("Please add the salary of this month and year for this record.\n");**

**}**

**else{**

**ta.append("\n\nHRA : "+Float.parseFloat(rest2.getString("hra")));**

**ta.append("\nDA : "+Float.parseFloat(rest2.getString("da")));**

**ta.append("\nMED : "+Float.parseFloat(rest2.getString("mid")));**

**ta.append("\nPF : "+Float.parseFloat(rest2.getString("pf")));**

**ta.append("\nBASIC Salary : "+Float.parseFloat(rest2.getString("basic")));**

**}**

**}**

**catch(Exception exx){**

**exx.printStackTrace();**

**}**

**}**

**if(e.getSource()==bt2){**

**JOptionPane.showMessageDialog(null, "Are you sure to close?");**

**setVisible(false);**

**}**

**}**

**public static void main(String[] args)**

**{**

**new Generate\_PaySlip().setVisible(true);**

**}**

**}**

**package Employee\_Management;**

**import java.awt.Color;**

**import java.awt.Font;**

**import java.awt.event.ActionEvent;**

**import java.awt.event.ActionListener;**

**import javax.swing.ImageIcon;**

**import javax.swing.JButton;**

**import javax.swing.JFrame;**

**import javax.swing.JLabel;**

**public class EmployeeFirstpage extends JFrame implements ActionListener{**

**EmployeeFirstpage(){**

**setBounds(100, 100, 1322, 566);**

**setLayout(null);**

**ImageIcon il = new ImageIcon(ClassLoader.getSystemResource("Employee\_Management/icons/emplo.jpg"));**

**JLabel image = new JLabel(il);**

**image.setBounds(0, 0, 1366, 565);**

**add(image);**

**JLabel heading = new JLabel("EMPLOYEE MANAGEMENT SYSTEM");**

**heading.setBounds(155, 30, 1200, 60);**

**heading.setFont(new Font("serif", Font.PLAIN,60));**

**heading.setForeground(Color.blue);**

**add(heading);**

**JButton next = new JButton("Next");**

**next.setBounds(1150, 470, 150, 50);**

**next.setBackground(Color.WHITE);**

**next.setForeground(Color.MAGENTA);**

**next.addActionListener(this);**

**next.setFont(new Font("serif",Font.PLAIN, 24));**

**image.add(next);**

**setVisible(true);**

**}**

**public void actionPerformed(ActionEvent ae){**

**setVisible(false);**

**new Loginpage();**

**}**

**public static void main(String[] args) {**

**new EmployeeFirstpage();**

**}**

**}**

**2) Output Screen**



First page

* **TESTING AND VALIDATION CHECKS**

Software Testing is a method to check whether the actual software product matches expected requirements and to ensure that software product is Defect free. It involves execution of software/system components using manual or automated tools to evaluate one or more properties of interest.

Testing provides ultimate support for Visual Basic applications. For these applications you can perform almost any kind of tests: unit, functional (or user interface), regression, data- driven, etc.In software project management, software testing, and software engineering, verification and validation (V&V) is the process of checking that a software system meets specifications and that it fulfills its intended purpose. It may also be referred to as software quality control. It is normally the responsibility of software testers as part of the software development lifecycle. Validation checks that

the product design satisfies or fits the intended use (high-level checking), i.e., the software meets the user requirements. This is done through dynamic testing and other forms of review.

Verification and validation are not the same thing, although they are often confused. Boehm succinctly expressed Software Verification: The process of evaluating software to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase.

Software Validation: The process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements.

As this project has made the use of the "JAVA", data is stored in the form of the frame that distinguishes it from another. The user should know such data which is not repeated and the errors are minimized. Another validation check is that proper data needs to inputted so that the user can get proper output. Users have to need fill the every filled compulsorily without that he cannot move to other filled in payment form only numbers can be inputted.

* **SYSTEM SECURITY MEASURES**

**Security** is an important term of any project (system) development. Security provides a protection to the computer, database, and project from unauthorized user. Security of computer facilities stored data and the information generated is a part of successful conversion. Recognition of the need for security is a natural outgrowthof the belief that information is a key organizational resource. It is useful to think of security of the project data and information on an imaginary continuum from totally secure to totally open. Security is the responsibility of all thosewho come into context with the system. Security has three interrelated aspects i.e. physical, logical and behavioral.

When you run this software, very first form which you find is a password frame consisting of two textboxes for username and password. As soon as frame open and ask for password from the user, user enters the correct password and will move on the next form otherwise move will not be allowed until and unless user types the correct password.

* **IMPLEMENTATION, EVALUATION AND MAINTENANCE**

Implementation phase is mainly concerned with user training, site preparations & file conversions. It also involves final testing of the system, During Implementation the components build during development are put into operational use. Brief Reference of the points that should be addressed during implementation:

a) Writing, testing, Debugging & Documenting program.

b) Converting data from the old to new system.

e) Giving training to user about how to operate the system. d) Developing operating procedures for the computer center staff.

e) Establishing a maintenance procedure to repair & Enhance the system.

1) Completing system Documentation.

g) Reviewing the administrative plan, personnel requirement plan, and hardware plan.

**Evaluation and Maintenance**

After the Implementation stage, another important stage in project development is evaluation & maintenance. After keeping the project in the working condition for some time, all the errors that are shown in the computer program should be removed. The programmer needs to correct them so that the same errors should not be repeated. We should also get the feedback from the users which are using it and ask them whether it is user-friendly or not. After evaluating the program & satisfying the needs of the user the program is maintained fully to give the same functionality for what it was

intended to be. This stage should be implemented so as to regular check-ups of errors with error handling techniques. This stage is the updating & correcting of the program to account for changing conditions or field experience. Proper testing & documentation significantly reduce the frequency and extent of the required maintenance.

1. Correcting Errors.
2. Newly discovered bugs.

c) Resolving necessary changes.

1. Specification changes.

e) Enhances or modify the system maintenance.

* + **FUTURE SCOPE OF THE PROJECT**

The GUI and the features added to this system are the basic ones.

In future, there will be a better Graphical User Interface and there will be more features added to this system.

If Graphical User Interface is improved then this system will be more user friendly and more features added will make this system a lot better and HR will be able to perform more operations.

The future scope of employee management systems is vast and promising. As technology continues to advance, these systems will become even more powerful and useful for businesses of all sizes. Some potential areas of growth and development include:

Integration with other business systems: Employee management systems will likely become more integrated with other business systems, such as payroll and accounting software. This will allow for more streamlined and efficient management of employee data across multiple platforms.

Enhanced automation: Automation is already a key feature of many employee management systems, but it will likely become even more advanced in the future. Tasks such as onboarding, time tracking, and performance evaluations may become fully automated, freeing up time for HR staff to focus on more strategic initiatives.

Improved analytics: Employee management systems generate a wealth of data that can be used to improve business performance. In the future, these systems may become even more sophisticated in terms of data analysis and reporting, allowing businesses to make more informed decisions about their workforce.

* **SUGGESTION & CONCLUSION**

Employee management systems are important tools for organizations to streamline their HR processes and optimize employee performance.

Here are some suggestions and conclusions for effective employee management system:

Use a cloud-based employee management system: Cloud-based systems are accessible from anywhere and provide real-time updates. They also provide more security and backup options.

Use an integrated system: An integrated system can manage all HR functions in one place, including payroll, performance, benefits, and employee data.

Define clear policies and procedures: Clear policies and procedures help employees understand what is expected of them and what they can expect from the organization.

Provide regular training: Regular training can help employees stay up-to-date on company policies and procedures, as well as new technologies and skills.

Encourage feedback: Encouraging feedback from employees can help identify areas for improvement in the employee management system and in the organization as a whole.

* **BIBLIOGRAPHY & REFERENCES**

**References:**

1. **https://www.codecademy.com/learn/learn-java**
2. **https://www.codecademy.com/learn/learn-java**
3. **https://www.udemy.com/topic/java/**
4. **https://www.coursera.org/courses?query=java**
5. **https://www.baeldung.com/**
6. [**https://www.sololearn.com/Course/Java/**](https://www.sololearn.com/Course/Java/)
7. **https://www.youtube**