SOURCE CODE FOR HOSPITAL MANAGEMENT SYSTEM :

1 :CODE FOR TABLE CREATION :

CREATE TABLE "HOSPITAL$PATIENTS" (

"PATIENT\_ID#" VARCHAR2(20),

"NAME\_" VARCHAR2(100),

"AGE#" NUMBER(3),

"GENDER$" VARCHAR2(10),

"CONTACT\_NO" VARCHAR2(20),

CONSTRAINT "PK$PATIENTS\_ID" PRIMARY KEY ("PATIENT\_ID#")

);

CREATE TABLE "HOSPITAL$DOCTORS" (

"DOCTOR\_ID#" VARCHAR2(20),

"NAME\_" VARCHAR2(100),

"SPECIALIZATION$" VARCHAR2(100),

"PHONE\_NO" VARCHAR2(20),

CONSTRAINT "PK$DOCTORS\_ID" PRIMARY KEY ("DOCTOR\_ID#")

);

CREATE TABLE "HOSPITAL#APPOINTMENTS" (

"APPT\_ID$" VARCHAR2(20),

"PATIENT\_ID#" VARCHAR2(20),

"DOCTOR\_ID#" VARCHAR2(20),

"APPT\_DATE" DATE,

"APPT\_TIME" VARCHAR2(10),

CONSTRAINT "PK$APPT\_ID" PRIMARY KEY ("APPT\_ID$"),

CONSTRAINT "FK#APPT\_PATIENT" FOREIGN KEY ("PATIENT\_ID#")

REFERENCES "HOSPITAL$PATIENTS" ("PATIENT\_ID#"),

CONSTRAINT "FK#APPT\_DOCTOR" FOREIGN KEY ("DOCTOR\_ID#")

REFERENCES "HOSPITAL$DOCTORS" ("DOCTOR\_ID#")

);

CREATE TABLE "HOSPITAL$STAFF" (

"STAFF\_ID#" VARCHAR2(20),

"NAME\_" VARCHAR2(100),

"POSITION$" VARCHAR2(50),

"PHONE\_NO" VARCHAR2(20),

CONSTRAINT "PK$STAFF\_ID" PRIMARY KEY ("STAFF\_ID#")

);

CREATE TABLE "HOSPITAL$BILLING" (

"BILL\_ID$" VARCHAR2(20),

"PATIENT\_ID#" VARCHAR2(20),

"AMOUNT$" NUMBER(10, 2),

"BILL\_DATE" DATE,

CONSTRAINT "PK$BILL\_ID" PRIMARY KEY ("BILL\_ID$"),

CONSTRAINT "FK#BILL\_PATIENT" FOREIGN KEY ("PATIENT\_ID#")

REFERENCES "HOSPITAL$PATIENTS" ("PATIENT\_ID#")

);

2 :Code for Login Page GUI :

package hello;

import javax.swing.\*;

import javax.swing.table.DefaultTableModel;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.Connection;

import java.sql.PreparedStatement;

public class HospitalManagementSystem {

public static void main(String[] args) {

SwingUtilities.invokeLater(() -> new HospitalManagementSystem().showLoginWindow());

}

void showLoginWindow() {

JFrame loginFrame = new JFrame("Hospital Login");

loginFrame.setSize(400, 250);

loginFrame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

loginFrame.setLayout(null);

loginFrame.getContentPane().setBackground(new Color(240, 248, 255));

JLabel title = new JLabel("Hospital Management System", JLabel.CENTER);

title.setFont(new Font("Times New Roman", Font.BOLD, 18));

title.setBounds(50, 20, 300, 30);

loginFrame.add(title);

JLabel userLabel = new JLabel("USERNAME:");

userLabel.setFont(new Font("Times New Roman", Font.BOLD, 12));

userLabel.setBounds(50, 70, 100, 25);

loginFrame.add(userLabel);

JTextField userField = new JTextField();

userField.setBounds(150, 70, 180, 25);

loginFrame.add(userField);

JLabel passLabel = new JLabel("PASSWORD:");

passLabel.setFont(new Font("Times New Roman", Font.BOLD, 12));

passLabel.setBounds(50, 110, 100, 25);

loginFrame.add(passLabel);

JPasswordField passField = new JPasswordField();

passField.setBounds(150, 110, 180, 25);

loginFrame.add(passField);

JButton loginBtn = new JButton("Login");

loginBtn.setBounds(150, 160, 100, 30);

loginBtn.setBackground(new Color(70, 130, 180));

loginBtn.setForeground(Color.WHITE);

loginBtn.setFocusPainted(false);

loginBtn.addActionListener(e -> {

String username = userField.getText().trim();

String password = new String(passField.getPassword()).trim();

if ("admin".equals(username) && "admin".equals(password)) {

loginFrame.dispose();

showDashboard();

} else {

JOptionPane.showMessageDialog(loginFrame, "Invalid username or password", "Error", JOptionPane.ERROR\_MESSAGE);

}

});

loginFrame.add(loginBtn);

loginFrame.setLocationRelativeTo(null);

loginFrame.setVisible(true);

}

void showDashboard() {

JFrame frame = new JFrame("Hospital Management Dashboard");

frame.setSize(600, 450);

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setLayout(new BorderLayout(10, 10));

frame.getContentPane().setBackground(new Color(240, 248, 255));

JLabel welcomeLabel = new JLabel("Welcome to Hospital Management System", JLabel.CENTER);

welcomeLabel.setFont(new Font("Arial", Font.BOLD, 22));

welcomeLabel.setBorder(BorderFactory.createEmptyBorder(20, 10, 20, 10));

frame.add(welcomeLabel, BorderLayout.NORTH);

JPanel buttonPanel = new JPanel(new GridLayout(3, 2, 20, 20));

buttonPanel.setBackground(new Color(240, 248, 255));

buttonPanel.setBorder(BorderFactory.createEmptyBorder(10, 40, 40, 40));

String[] btnNames = {"Patients", "Doctors", "Appointments", "Staff", "Billing", "Logout"};

for (String name : btnNames) {

JButton btn = new JButton(name);

btn.setFont(new Font("Arial", Font.BOLD, 18));

btn.setBackground(new Color(70, 130, 180));

btn.setForeground(Color.WHITE);

btn.setFocusPainted(false);

buttonPanel.add(btn);

btn.addActionListener(e -> {

switch (name) {

case "Patients": showPatientsWindow(); break;

case "Doctors": showDoctorsWindow(); break;

case "Appointments": showAppointmentsWindow(); break;

case "Staff": showStaffWindow(); break;

case "Billing": showBillingWindow(); break;

case "Logout": frame.dispose(); showLoginWindow(); break;

}

});

}

frame.add(buttonPanel, BorderLayout.CENTER);

frame.setLocationRelativeTo(null);

frame.setVisible(true);

}

JFrame createDataWindow(String title, String[] columns, String[] fieldLabels) {

JFrame frame = new JFrame(title);

frame.setSize(700, 450);

frame.setLayout(new BorderLayout(10, 10));

frame.getContentPane().setBackground(new Color(245, 245, 245));

frame.setLocationRelativeTo(null);

DefaultTableModel model = new DefaultTableModel(columns, 0);

JTable table = new JTable(model);

table.setFillsViewportHeight(true);

JScrollPane scrollPane = new JScrollPane(table);

frame.add(scrollPane, BorderLayout.CENTER);

JPanel formPanel = new JPanel(new GridLayout(fieldLabels.length + 1, 2, 10, 10));

formPanel.setBackground(new Color(245, 245, 245));

JTextField[] fields = new JTextField[fieldLabels.length];

for (int i = 0; i < fieldLabels.length; i++) {

formPanel.add(new JLabel(fieldLabels[i] + ":"));

JTextField field = new JTextField();

fields[i] = field;

formPanel.add(field);

}

JButton addBtn = new JButton("Add " + title.replace(" Management", ""));

addBtn.setBackground(new Color(70, 130, 180));

addBtn.setForeground(Color.WHITE);

addBtn.setFocusPainted(false);

formPanel.add(new JLabel());

formPanel.add(addBtn);

addBtn.addActionListener(e -> {

try {

String patientId = fields[0].getText().trim();

String name = fields[1].getText().trim();

String age = fields[2].getText().trim();

String gender = fields[3].getText().trim();

String contact = fields[4].getText().trim();

if (patientId.isEmpty() || name.isEmpty() || age.isEmpty() || gender.isEmpty() || contact.isEmpty()) {

JOptionPane.showMessageDialog(frame, "Please fill all fields", "Error", JOptionPane.ERROR\_MESSAGE);

return;

}

// Connect and insert into Oracle DB

Connection conn = DBConnection.connect();

PreparedStatement stmt = conn.prepareStatement(

"INSERT INTO \"HOSPITAL$PATIENTS\" (\"PATIENT\_ID#\", \"NAME\_\", \"AGE#\", \"GENDER$\", \"CONTACT\_NO\") VALUES (?, ?, ?, ?, ?)"

);

stmt.setString(1, patientId);

stmt.setString(2, name);

stmt.setInt(3, Integer.parseInt(age))

stmt.setString(4, gender);

stmt.setString(5, contact);

stmt.executeUpdate();

stmt.close();

conn.close();

// Update GUI table

model.addRow(new Object[]{patientId, name, age, gender, contact});

for (JTextField f : fields) f.setText("");

JOptionPane.showMessageDialog(frame, "Patient added successfully to database.");

} catch (Exception ex) {

ex.printStackTrace();

JOptionPane.showMessageDialog(frame, "Error adding patient: " + ex.getMessage(), "Error", JOptionPane.ERROR\_MESSAGE);

}

});

frame.add(formPanel, BorderLayout.NORTH);

return frame;

}

void showPatientsWindow() {

String[] columns = {"Patient ID", "Name", "Age", "Gender", "Contact"};

String[] fields = {"Patient ID", "Name", "Age", "Gender", "Contact"};

JFrame frame = createDataWindow("Patients Management", columns, fields);

frame.setVisible(true);

}

void showDoctorsWindow() {

String[] columns = {"Doctor ID", "Name", "Specialization", "Phone"};

String[] fields = {"Doctor ID", "Name", "Specialization", "Phone"};

JFrame frame = createDataWindow("Doctors Management", columns, fields);

frame.setVisible(true);

}

void showAppointmentsWindow() {

String[] columns = {"Appointment ID", "Patient ID", "Doctor ID", "Date", "Time"};

String[] fields = {"Appointment ID", "Patient ID", "Doctor ID", "Date (YYYY-MM-DD)", "Time (HH:MM)"};

JFrame frame = createDataWindow("Appointments Management", columns, fields);

frame.setVisible(true);

}

void showStaffWindow() {

String[] columns = {"Staff ID", "Name", "Position", "Phone"};

String[] fields = {"Staff ID", "Name", "Position", "Phone"};

JFrame frame = createDataWindow("Staff Management", columns, fields);

frame.setVisible(true);

}

void showBillingWindow() {

String[] columns = {"Bill ID", "Patient ID", "Amount", "Date"};

String[] fields = {"Bill ID", "Patient ID", "Amount", "Date (YYYY-MM-DD)"};

JFrame frame = createDataWindow("Billing Management", columns, fields);

frame.setVisible(true);

}

}

3 :CODE FOR CONNECTION:

package hello;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

public class DBConnection {

public static Connection connect() {

Connection conn = null;

String DB\_URL = "jdbc:oracle:thin:@10.11.0.43:1521:XE";

String USER = "FA24CS049";

String PASS = "oracle";

try {

conn = DriverManager.getConnection(DB\_URL, USER, PASS);

System.out.println("Connected to the Oracle DB!");

} catch (SQLException e) {

System.err.format("SQL State: %s\n%s", e.getSQLState(), e.getMessage());

} catch (Exception e) {

e.printStackTrace();

}

return conn;}

public static void main(String[] args) {

Connection conn = connect();

if (conn != null) {

System.out.println("Connection test successful!");

} else {

System.out.println("Connection test failed.");

}

}

}