**FULL CODE**

**\*\*StudentDatabaseManagerAWT\*\***

import java.awt.\*;

import java.awt.event.\*;

import java.util.HashMap;

import java.util.Map;

// Main class for the GUI

public class StudentDatabaseManagerAWT {

private DatabaseManager dbManager;

private TextArea displayArea;

public StudentDatabaseManagerAWT() {

dbManager = new DatabaseManager();

Frame frame = new Frame("Student Database Management System");

frame.setSize(500, 500);

frame.setLayout(new GridLayout(0, 2, 5, 5));

// Set background color and font

frame.setBackground(new Color(230, 240, 255)); // Mild blue background

Font font = new Font("Arial", Font.BOLD, 14); // Big, attractive font

// Create labels, text fields, and set fonts and sizes

Label idLabel = new Label("ID:");

idLabel.setFont(font);

TextField idField = new TextField(20);

idField.setFont(font);

Label nameLabel = new Label("Name:");

nameLabel.setFont(font);

TextField nameField = new TextField(20);

nameField.setFont(font);

Label ageLabel = new Label("Age:");

ageLabel.setFont(font);

TextField ageField = new TextField(20);

ageField.setFont(font);

Label gradeLabel = new Label("Grade:");

gradeLabel.setFont(font);

TextField gradeField = new TextField(20);

gradeField.setFont(font);

displayArea = new TextArea(10, 30);

displayArea.setEditable(false);

displayArea.setFont(font);

// Set font and background color for buttons

Button addButton = new Button("Add Student");

addButton.setFont(font);

addButton.setBackground(new Color(173, 216, 230)); // Light blue

Button searchButton = new Button("Search Student");

searchButton.setFont(font);

searchButton.setBackground(new Color(173, 216, 230));

Button deleteButton = new Button("Delete Student");

deleteButton.setFont(font);

deleteButton.setBackground(new Color(173, 216, 230));

Button displayButton = new Button("Display All Students");

displayButton.setFont(font);

displayButton.setBackground(new Color(173, 216, 230));

// Add action listeners for buttons

addButton.addActionListener(e -> {

String id = idField.getText();

String name = nameField.getText();

int age = Integer.parseInt(ageField.getText());

String grade = gradeField.getText();

dbManager.addStudent(new Student(id, name, age, grade));

displayArea.setText("Student added successfully.");

});

searchButton.addActionListener(e -> {

String id = idField.getText();

Student student = dbManager.searchStudent(id);

if (student != null) {

displayArea.setText("Student found: " + student);

} else {

displayArea.setText("Student not found.");

}

});

deleteButton.addActionListener(e -> {

String id = idField.getText();

dbManager.deleteStudent(id);

displayArea.setText("Student deleted successfully.");

});

displayButton.addActionListener(e -> {

displayArea.setText(""); // Clear previous display

for (Student student : dbManager.getAllStudents().values()) {

displayArea.append(student + "\n");

}

if (dbManager.getAllStudents().isEmpty()) {

displayArea.setText("No students to display.");

}

});

// Add components to the frame

frame.add(idLabel);

frame.add(idField);

frame.add(nameLabel);

frame.add(nameField);

frame.add(ageLabel);

frame.add(ageField);

frame.add(gradeLabel);

frame.add(gradeField);

frame.add(addButton);

frame.add(searchButton);

frame.add(deleteButton);

frame.add(displayButton);

frame.add(displayArea);

frame.setVisible(true);

frame.addWindowListener(new WindowAdapter() {

public void windowClosing(WindowEvent we) {

System.exit(0);

}

});

}

public static void main(String[] args) {

new StudentDatabaseManagerAWT();

}

}

**\*\*TeacherdatabasemanagerAWT\*\***

import java.awt.\*;

import java.awt.event.\*;

import java.util.HashMap;

import java.util.Map;

// Main class for the GUI

public class TeacherdatabasemanagerAWT {

private DatabaseManager dbManager;

private TextArea displayArea;

public TeacherdatabasemanagerAWT() {

dbManager = new DatabaseManager();

Frame frame = new Frame("Student Database Management System");

frame.setSize(500, 500);

frame.setLayout(new GridLayout(0, 2, 5, 5));

// Set background color and font

frame.setBackground(new Color(230, 240, 255)); // Mild blue background

Font font = new Font("Arial", Font.BOLD, 14); // Big, attractive font

// Create labels, text fields, and set fonts and sizes

Label idLabel = new Label("ID:");

idLabel.setFont(font);

TextField idField = new TextField(20);

idField.setFont(font);

Label nameLabel = new Label("Name:");

nameLabel.setFont(font);

TextField nameField = new TextField(20);

nameField.setFont(font);

Label ageLabel = new Label("Age:");

ageLabel.setFont(font);

TextField ageField = new TextField(20);

ageField.setFont(font);

Label gradeLabel = new Label("Grade:");

gradeLabel.setFont(font);

TextField gradeField = new TextField(20);

gradeField.setFont(font);

displayArea = new TextArea(10, 30);

displayArea.setEditable(false);

displayArea.setFont(font);

// Set font and background color for buttons

Button addButton = new Button("Add Teacher");

addButton.setFont(font);

addButton.setBackground(new Color(173, 216, 230)); // Light blue

Button searchButton = new Button("Search Teacher");

searchButton.setFont(font);

searchButton.setBackground(new Color(173, 216, 230));

Button deleteButton = new Button("Delete Teacher");

deleteButton.setFont(font);

deleteButton.setBackground(new Color(173, 216, 230));

Button displayButton = new Button("Display All Teacher");

displayButton.setFont(font);

displayButton.setBackground(new Color(173, 216, 230));

// Add action listeners for buttons

addButton.addActionListener(e -> {

String id = idField.getText();

String name = nameField.getText();

int age = Integer.parseInt(ageField.getText());

String grade = gradeField.getText();

dbManager.addStudent(new Student(id, name, age, grade));

displayArea.setText("Teacher added successfully.");

});

searchButton.addActionListener(e -> {

String id = idField.getText();

Student student = dbManager.searchStudent(id);

if (student != null) {

displayArea.setText("Teacher found: " + student);

} else {

displayArea.setText("Teacher not found.");

}

});

deleteButton.addActionListener(e -> {

String id = idField.getText();

dbManager.deleteStudent(id);

displayArea.setText("Teacher deleted successfully.");

});

displayButton.addActionListener(e -> {

displayArea.setText(""); // Clear previous display

for (Student student : dbManager.getAllStudents().values()) {

displayArea.append(student + "\n");

}

if (dbManager.getAllStudents().isEmpty()) {

displayArea.setText("No Teacher to display.");

}

});

// Add components to the frame

frame.add(idLabel);

frame.add(idField);

frame.add(nameLabel);

frame.add(nameField);

frame.add(ageLabel);

frame.add(ageField);

frame.add(gradeLabel);

frame.add(gradeField);

frame.add(addButton);

frame.add(searchButton);

frame.add(deleteButton);

frame.add(displayButton);

frame.add(displayArea);

frame.setVisible(true);

frame.addWindowListener(new WindowAdapter() {

public void windowClosing(WindowEvent we) {

System.exit(0);

}

});

}

public static void main(String[] args) {

new StudentDatabaseManagerAWT();

}

}

**\*\*UserRoleSelectionAWT\*\***

import java.awt.\*;

import java.awt.event.\*;

public class UserRoleSelectionAWT {

public UserRoleSelectionAWT() {

Frame selectionFrame = new Frame("Select User Role");

selectionFrame.setSize(400, 300);

selectionFrame.setLayout(new GridLayout(3, 1, 5, 5));

Font font = new Font("Arial", Font.BOLD, 14);

selectionFrame.setFont(font);

Button studentButton = new Button("Student");

studentButton.setFont(font);

studentButton.setBackground(new Color(173, 216, 230));

Button teacherButton = new Button("Teacher");

teacherButton.setFont(font);

teacherButton.setBackground(new Color(173, 216, 230));

studentButton.addActionListener(e -> openStudentPage());

teacherButton.addActionListener(e -> openTeacherPage());

selectionFrame.add(studentButton);

selectionFrame.add(teacherButton);

selectionFrame.setVisible(true);

selectionFrame.addWindowListener(new WindowAdapter() {

public void windowClosing(WindowEvent we) {

System.exit(0);

}

});

}

// Open the student database management page

private void openStudentPage() {

new StudentDatabaseManagerAWT(); // Opens the student page

}

// Open the teacher database management page

private void openTeacherPage() {

new TeacherdatabasemanagerAWT(); // Opens the teacher page

}

}

**\*\*LoginSignUpManagerAWT\*\***

import java.awt.\*;

import java.awt.event.\*;

public class LoginSignUpManagerAWT {

private UserDatabaseManager userDbManager;

private Frame loginFrame;

private TextField usernameField;

private TextField passwordField;

private Label statusLabel;

public LoginSignUpManagerAWT() {

userDbManager = new UserDatabaseManager();

loginFrame = new Frame("Login and Sign-Up System");

loginFrame.setSize(400, 300);

loginFrame.setLayout(new GridLayout(5, 1, 5, 5));

Font font = new Font("Arial", Font.BOLD, 14);

loginFrame.setFont(font);

Label usernameLabel = new Label("Username:");

usernameLabel.setFont(font);

usernameField = new TextField(20);

usernameField.setFont(font);

Label passwordLabel = new Label("Password:");

passwordLabel.setFont(font);

passwordField = new TextField(20);

passwordField.setFont(font);

passwordField.setEchoChar('\*');

Button loginButton = new Button("Login");

loginButton.setFont(font);

loginButton.setBackground(new Color(173, 216, 230));

Button signUpButton = new Button("Sign Up");

signUpButton.setFont(font);

signUpButton.setBackground(new Color(173, 216, 230));

statusLabel = new Label("");

statusLabel.setFont(font);

loginButton.addActionListener(e -> loginUser());

signUpButton.addActionListener(e -> signUpUser());

loginFrame.add(usernameLabel);

loginFrame.add(usernameField);

loginFrame.add(passwordLabel);

loginFrame.add(passwordField);

loginFrame.add(loginButton);

loginFrame.add(signUpButton);

loginFrame.add(statusLabel);

loginFrame.setVisible(true);

loginFrame.addWindowListener(new WindowAdapter() {

public void windowClosing(WindowEvent we) {

System.exit(0);

}

});

}

private void loginUser() {

String username = usernameField.getText();

String password = passwordField.getText();

if (userDbManager.verifyUser(username, password)) {

statusLabel.setText("Login successful!");

loginFrame.setVisible(false); // Hide the login frame

// Open the user role selection page after successful login

new UserRoleSelectionAWT(); // Opens the page with Student/Teacher options

} else {

statusLabel.setText("Invalid username or password.");

}

}

private void signUpUser() {

String username = usernameField.getText();

String password = passwordField.getText();

if (userDbManager.addUser(username, password)) {

statusLabel.setText("Sign-Up successful! Please log in.");

} else {

statusLabel.setText("Username already exists. Please try another.");

}

}

public static void main(String[] args) {

new LoginSignUpManagerAWT();

}

}