### **CSD420-H326 Advanced Java Programming (2251-DD)**

**Date**: 12/09/2024

**Name**: Arun Sharma

**Assignment no.**: 10 of Module 10: JDBC Part II

### **Objective**

The goal of this assignment was to create a Java GUI application (FanManager) that connects to a MySQL database to view and update records in a fans table. The application includes:

1. **Display Button**: Retrieves and displays a record based on the provided ID.
2. **Update Button**: Updates a record in the database with changes made in the GUI fields.
3. Testing invalid inputs to ensure robust error handling.

### **Initial Table State**

Before running the FanManager application, the initial state of the fans table was as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **firstname** | **lastname** | **favoriteteam** |
| 1 | John | Doe | Yankees |
| 2 | Jane | Smith | Dodgers |
| 3 | Alice | Brown | Red Sox |
| 4 | Bob | Johnson | Mets |

### **Updated Table After GUI Operations**

After running the FanManager application and performing updates through the GUI, the fans table was updated as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **firstname** | **lastname** | **favoriteteam** |
| 1 | Agam | Walia | Lakers |
| 2 | Aman | Malhotra | Indian cricket team |
| 3 | Arun | Sharma | Lakers |
| 4 | Bob | Johnson | Mets |

### **Test Results**

#### **1. Display Record**

* Tested the **Display** button with valid ID values:
  + **Input**: ID = 1
  + **Output**: Correct record retrieved:

ID: 1  
First Name: Agam  
Last Name: Walia  
Favorite Team: Lakers

#### **2. Update Record**

* Updated the record with ID = 2:
  + **Input**: Changed firstname to Aman, lastname to Malhotra, and favoriteteam to Indian cricket team.
  + **Output**: Record updated successfully in the database:

SELECT \* FROM fans WHERE ID = 2;  
| ID | firstname | lastname | favoriteteam |  
| 2 | Aman | Malhotra | Indian cricket team |

#### **3. Invalid Record Test**

* Tested the **Display** button with an invalid ID:
  + **Input**: ID = 999
  + **Output**: Error message displayed:

Record not found

### **Code**

#### **FanManager.java**

/\*

\* Date: 12/09/2024

\* Course: CSD420-H326 Advanced Java Programming

\* GitHub Repository: <https://github.com/SharmaArun017/CSD-420>

\*

\* Program: FanManager of module 10 - assignment10

\* Description: This program is a GUI-based application that allows users to

\* view and update records in the "fans" table located in the "databasedb" database.

\*

\* Functionality:

\* 1. Display Button: Retrieves and displays a record based on the provided ID.

\* 2. Update Button: Updates a record in the database with changes made in the GUI fields.

\*/

package module10.assignment10;

import java.sql.\*;

import javax.swing.\*;

import java.awt.\*;

public class FanManager extends JFrame {

private JTextField idField, firstNameField, lastNameField, favoriteTeamField;

private JButton displayButton, updateButton;

private Connection con;

public FanManager() {

// GUI Setup

setupGUI();

// Database Connection

connectToDatabase();

}

private void setupGUI() {

setTitle("Fan Manager");

setSize(400, 300);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setLayout(new GridLayout(5, 2));

add(new JLabel("ID:"));

idField = new JTextField();

add(idField);

add(new JLabel("First Name:"));

firstNameField = new JTextField();

add(firstNameField);

add(new JLabel("Last Name:"));

lastNameField = new JTextField();

add(lastNameField);

add(new JLabel("Favorite Team:"));

favoriteTeamField = new JTextField();

add(favoriteTeamField);

displayButton = new JButton("Display");

updateButton = new JButton("Update");

add(displayButton);

add(updateButton);

// Add Button Listeners

displayButton.addActionListener(e -> displayRecord());

updateButton.addActionListener(e -> updateRecord());

setVisible(true);

}

private void connectToDatabase() {

try {

Class.forName("com.mysql.cj.jdbc.Driver");

String url = "jdbc:mysql://localhost:3306/databasedb";

con = DriverManager.getConnection(url, "student1", "pass");

} catch (Exception e) {

JOptionPane.showMessageDialog(this, "Error connecting to database: " + e.getMessage());

}

}

private void displayRecord() {

try {

int id = Integer.parseInt(idField.getText());

String query = "SELECT \* FROM fans WHERE ID = ?";

PreparedStatement pstmt = con.prepareStatement(query);

pstmt.setInt(1, id);

ResultSet rs = pstmt.executeQuery();

if (rs.next()) {

firstNameField.setText(rs.getString("firstname"));

lastNameField.setText(rs.getString("lastname"));

favoriteTeamField.setText(rs.getString("favoriteteam"));

} else {

JOptionPane.showMessageDialog(this, "Record not found");

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, "Error displaying record: " + e.getMessage());

}

}

private void updateRecord() {

try {

int id = Integer.parseInt(idField.getText());

String firstName = firstNameField.getText();

String lastName = lastNameField.getText();

String favoriteTeam = favoriteTeamField.getText();

String query = "UPDATE fans SET firstname = ?, lastname = ?, favoriteteam = ? WHERE ID = ?";

PreparedStatement pstmt = con.prepareStatement(query);

pstmt.setString(1, firstName);

pstmt.setString(2, lastName);

pstmt.setString(3, favoriteTeam);

pstmt.setInt(4, id);

int rows = pstmt.executeUpdate();

if (rows > 0) {

JOptionPane.showMessageDialog(this, "Record updated successfully");

} else {

JOptionPane.showMessageDialog(this, "Record not found");

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, "Error updating record: " + e.getMessage());

}

}

public static void main(String[] args) {

new FanManager();

}

}

= Fan manager class code:

// Source code is decompiled from a .class file using FernFlower decompiler.

package module10.assignment10;

import java.awt.GridLayout;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JTextField;

public class FanManager extends JFrame {

private JTextField idField;

private JTextField firstNameField;

private JTextField lastNameField;

private JTextField favoriteTeamField;

private JButton displayButton;

private JButton updateButton;

private Connection con;

public FanManager() {

this.setupGUI();

this.connectToDatabase();

}

private void setupGUI() {

this.setTitle("Fan Manager");

this.setSize(400, 300);

this.setDefaultCloseOperation(3);

this.setLayout(new GridLayout(5, 2));

this.add(new JLabel("ID:"));

this.idField = new JTextField();

this.add(this.idField);

this.add(new JLabel("First Name:"));

this.firstNameField = new JTextField();

this.add(this.firstNameField);

this.add(new JLabel("Last Name:"));

this.lastNameField = new JTextField();

this.add(this.lastNameField);

this.add(new JLabel("Favorite Team:"));

this.favoriteTeamField = new JTextField();

this.add(this.favoriteTeamField);

this.displayButton = new JButton("Display");

this.updateButton = new JButton("Update");

this.add(this.displayButton);

this.add(this.updateButton);

this.displayButton.addActionListener((var1) -> {

this.displayRecord();

});

this.updateButton.addActionListener((var1) -> {

this.updateRecord();

});

this.setVisible(true);

}

private void connectToDatabase() {

try {

Class.forName("com.mysql.cj.jdbc.Driver");

String var1 = "jdbc:mysql://localhost:3306/databasedb";

this.con = DriverManager.getConnection(var1, "student1", "pass");

} catch (Exception var2) {

JOptionPane.showMessageDialog(this, "Error connecting to database: " + var2.getMessage());

}

}

private void displayRecord() {

try {

int var1 = Integer.parseInt(this.idField.getText());

String var2 = "SELECT \* FROM fans WHERE ID = ?";

PreparedStatement var3 = this.con.prepareStatement(var2);

var3.setInt(1, var1);

ResultSet var4 = var3.executeQuery();

if (var4.next()) {

this.firstNameField.setText(var4.getString("firstname"));

this.lastNameField.setText(var4.getString("lastname"));

this.favoriteTeamField.setText(var4.getString("favoriteteam"));

} else {

JOptionPane.showMessageDialog(this, "Record not found");

}

} catch (Exception var5) {

JOptionPane.showMessageDialog(this, "Error displaying record: " + var5.getMessage());

}

}

private void updateRecord() {

try {

int var1 = Integer.parseInt(this.idField.getText());

String var2 = this.firstNameField.getText();

String var3 = this.lastNameField.getText();

String var4 = this.favoriteTeamField.getText();

String var5 = "UPDATE fans SET firstname = ?, lastname = ?, favoriteteam = ? WHERE ID = ?";

PreparedStatement var6 = this.con.prepareStatement(var5);

var6.setString(1, var2);

var6.setString(2, var3);

var6.setString(3, var4);

var6.setInt(4, var1);

int var7 = var6.executeUpdate();

if (var7 > 0) {

JOptionPane.showMessageDialog(this, "Record updated successfully");

} else {

JOptionPane.showMessageDialog(this, "Record not found");

}

} catch (Exception var8) {

JOptionPane.showMessageDialog(this, "Error updating record: " + var8.getMessage());

}

}

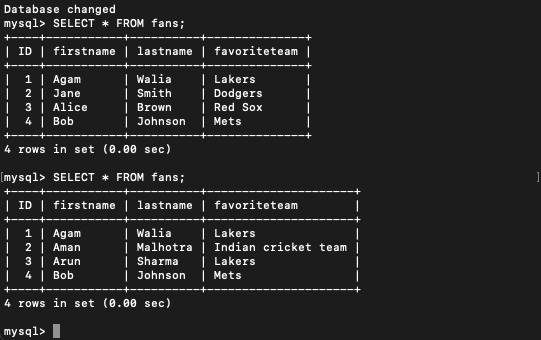
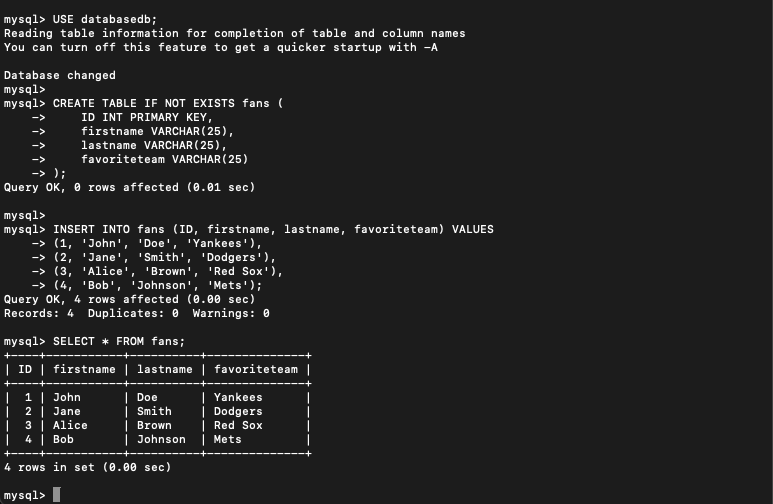
public static void main(String[] var0) {

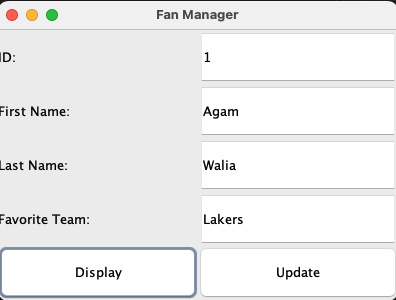
new FanManager();

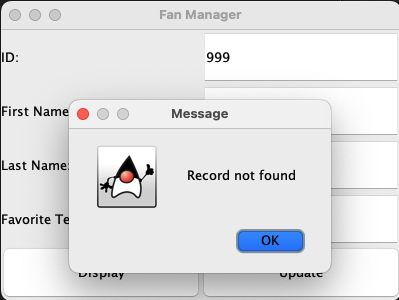
}

}

### **Screenshots**







(Performed random number id999 to test functionality )