12.mysql connectivity

package session;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.Statement;

public class Connect {

public static void main(String[] args) {

// TODO Auto-generated method stub

try {

Class.forName("oracle.jdbc.driver.OracleDriver");//Instantiation of the driver

Connection con = DriverManager.getConnection("jdbc:oracle:thin:@127.0.0.1:1521:xe",

"xe", "SYSTEM123"); //creation of statement for execution

Statement st = con.createStatement(); //opening the connection with db

// create table

ResultSet rs = st.executeQuery("create table conndemo(name varchar(25), roll int,marks int)");

ResultSet rs1 = st.executeQuery("insert into conndemo values('lmn',2,21)"); //inserted to table

ResultSet rs2 = st.executeQuery("insert into conndemo values('abc',3,25)");

ResultSet rs3 = st.executeQuery("insert into conndemo values('xyz',4,22)");

ResultSet rs4 = st.executeQuery("insert into conndemo values('def',5,27)");

ResultSet rs5 = st.executeQuery("select \* from conndemo"); //fetching data

System.out.println("executed successfully");

while (rs5.next()) {

System.out.println(rs5.getString("name") + " " + rs5.getInt("roll") + " " +

rs5.getInt(3));

}

int rs6 = st.executeUpdate("update studjdbcdemo set name='tina' where roll=3");

ResultSet rs7 = st.executeQuery("drop table studjdbcdemo"); //Deleting the database table view

ResultSet rs8 = st.executeQuery("create view viewJDBCNamee as select name from studjdbcdemo"); //creating view

ResultSet rs9 = st.executeQuery("select \* from viewJDBCNamee");

while (rs9.next()) {

System.out.println("" + rs9.getString(1));

}

ResultSet rs10 = st.executeQuery("alter table studjdbcdemo add admissionDate Date");

//alter table structure

ResultSet rs11 = st.executeQuery("alter table studjdbcdemo drop column admissionDate");

// rs.close();

st.close();

con.close();

} catch (Exception ex) {

System.out.println("Error: " + ex);

}

}

}

Alternate code

CREATE TABLE Employee (

emp\_id NUMBER PRIMARY KEY,

emp\_name VARCHAR2(50),

emp\_salary NUMBER

);

import java.sql.\*;

import java.util.Scanner;

public class OracleDBNavigation {

private static final String DB\_URL = "jdbc:oracle:thin:@localhost:1521:xe"; // Modify as per your DB

private static final String USER = "your\_username";

private static final String PASS = "your\_password";

private Connection connection;

private Scanner scanner;

public OracleDBNavigation() {

try {

// Establish database connection

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

System.out.println("Connected to Oracle database.");

} catch (SQLException e) {

e.printStackTrace();

}

scanner = new Scanner(System.in);

}

// Method to add a new employee

public void addEmployee() {

try {

System.out.print("Enter Employee ID: ");

int empId = scanner.nextInt();

scanner.nextLine(); // Consume newline

System.out.print("Enter Employee Name: ");

String empName = scanner.nextLine();

System.out.print("Enter Employee Salary: ");

double empSalary = scanner.nextDouble();

String sql = "INSERT INTO Employee (emp\_id, emp\_name, emp\_salary) VALUES (?, ?, ?)";

PreparedStatement preparedStatement = connection.prepareStatement(sql);

preparedStatement.setInt(1, empId);

preparedStatement.setString(2, empName);

preparedStatement.setDouble(3, empSalary);

int rows = preparedStatement.executeUpdate();

if (rows > 0) {

System.out.println("Employee added successfully.");

}

} catch (SQLException e) {

e.printStackTrace();

}

}

// Method to update employee details

public void updateEmployee() {

try {

System.out.print("Enter Employee ID to update: ");

int empId = scanner.nextInt();

scanner.nextLine(); // Consume newline

System.out.print("Enter New Employee Name: ");

String empName = scanner.nextLine();

System.out.print("Enter New Employee Salary: ");

double empSalary = scanner.nextDouble();

String sql = "UPDATE Employee SET emp\_name = ?, emp\_salary = ? WHERE emp\_id = ?";

PreparedStatement preparedStatement = connection.prepareStatement(sql);

preparedStatement.setString(1, empName);

preparedStatement.setDouble(2, empSalary);

preparedStatement.setInt(3, empId);

int rows = preparedStatement.executeUpdate();

if (rows > 0) {

System.out.println("Employee updated successfully.");

} else {

System.out.println("Employee not found.");

}

} catch (SQLException e) {

e.printStackTrace();

}

}

// Method to delete an employee

public void deleteEmployee() {

try {

System.out.print("Enter Employee ID to delete: ");

int empId = scanner.nextInt();

String sql = "DELETE FROM Employee WHERE emp\_id = ?";

PreparedStatement preparedStatement = connection.prepareStatement(sql);

preparedStatement.setInt(1, empId);

int rows = preparedStatement.executeUpdate();

if (rows > 0) {

System.out.println("Employee deleted successfully.");

} else {

System.out.println("Employee not found.");

}

} catch (SQLException e) {

e.printStackTrace();

}

}

// Method to list all employees

public void listEmployees() {

try {

String sql = "SELECT \* FROM Employee";

Statement statement = connection.createStatement();

ResultSet resultSet = statement.executeQuery(sql);

System.out.println("Employee List:");

while (resultSet.next()) {

System.out.println("ID: " + resultSet.getInt("emp\_id") +

", Name: " + resultSet.getString("emp\_name") +

", Salary: " + resultSet.getDouble("emp\_salary"));

}

} catch (SQLException e) {

e.printStackTrace();

}

}

// Main method to run the program

public static void main(String[] args) {

OracleDBNavigation dbNav = new OracleDBNavigation();

Scanner scanner = new Scanner(System.in);

while (true) {

System.out.println("\nChoose an operation:");

System.out.println("1. Add Employee");

System.out.println("2. Update Employee");

System.out.println("3. Delete Employee");

System.out.println("4. List Employees");

System.out.println("5. Exit");

System.out.print("Enter choice: ");

int choice = scanner.nextInt();

switch (choice) {

case 1:

dbNav.addEmployee();

break;

case 2:

dbNav.updateEmployee();

break;

case 3:

dbNav.deleteEmployee();

break;

case 4:

dbNav.listEmployees();

break;

case 5:

System.out.println("Exiting...");

System.exit(0);

break;

default:

System.out.println("Invalid choice. Try again.");

}

}

}

}