//Program for insert the multiple student data using JDBC into database.

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**public** **class** ConnectionTest {

Connection connection=**null**;

**public** Connection getConnectionDetails() {

**try** {

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

connection=DriverManager.*getConnection*("jdbc:mysql://localhost:3306/demo","root","root");

}**catch**(Exception e) {

e.printStackTrace();

}

**return** connection;

}

**import** java.sql.Connection;

**import** java.sql.PreparedStatement;

**import** java.sql.SQLException;

**import** java.util.Scanner;

**public** **class** UserInput {

PreparedStatement prs = **null**;

Connection con = **null**;

**public** **void** insertStudentData(String firstName,String lastName,String city,String sallery) {

**try** {

ConnectionTest connectionTest=**new** ConnectionTest();

con=connectionTest.getConnectionDetails();

prs=con.prepareStatement("insert into user(firstName,lastName,city,sallery)values(?,?,?,?)");

prs.setString(1, firstName);

prs.setString(2, lastName);

prs.setString(3, city);

prs.setString(4, sallery);

**int** i=prs.executeUpdate();

System.***out***.println("Record is inserted successfully.." + i);

}**catch**(Exception e) {

e.printStackTrace();

}

}

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

Scanner sc = **new** Scanner(System.***in***);

**for** (**int** i = 0; i < 3; i++) {

System.***out***.println("Enter first name>>");

String firstName = sc.next();

System.***out***.println("Enter last name>>");

String lastName = sc.next();

System.***out***.println("Enter city>>");

String city = sc.next();

System.***out***.println("Enter sallery>>");

String sallery = sc.next();

UserInput userInput = **new** UserInput();

userInput.insertStudentData(firstName, lastName, city, sallery);

}

sc.close();

}

}

**Connection Pooling:**

* If we required to communicate with database multiple times then it is not recommended to create separate Connection object every time, because creating and destroying Connection object every time creates performance problems.
* Also physical connection are limited in count.
* To overcome this problem, we should go for Connection Pool.
* Connection Pool is a pool of already created Connection objects which are ready to use.It creates logical connection.
* If we want to communicate with database then we request Connection pool to provide Connection. Once we got the Connection, by using that we can communicates with database. After completing our work, we can return Connection to the pool instead of destroying.
* Hence the main advantage of Connection Pool is we can reuse same Connection object multiple times, so that overall performance of application will be improved.

package com.test;

import java.sql.Connection;

import java.sql.DriverManager;

public class ConnectionDemo {

// Designing the method for establishing the connection

public Connection getConnectionDemo() {

Connection connection = null;

try {

// Loading the driver class

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

// Establishing the connection

connection = DriverManager.getConnection("jdbc:mysql://localhost:3306/test", "root", "root");

} catch (Exception e) {

e.printStackTrace();

}

return connection;

}

}

package com.test;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.SQLException;

import java.util.Scanner;

public class UserDemo {

// LastName, FirstName, Address, City, Salary

private void insertUserDetails(String LastName, String FirstName, String Address, String City, String Salary)

throws SQLException {

Connection connection = null;

PreparedStatement ps = null;

try {

ConnectionDemo connectionDemo = new ConnectionDemo();

connection = connectionDemo.getConnectionDemo();

ps = connection

.prepareStatement("insert into user(LastName,FirstName, Address, City, Salary) values (?,?,?,?,?)");

ps.setString(1, LastName);

ps.setString(2, FirstName);

ps.setString(3, Address);

ps.setString(4, City);

ps.setString(5, Salary);

int i = ps.executeUpdate();

System.out.println("Record Inserted>>" + i);

} catch (Exception e) {

e.printStackTrace();

} finally {

connection.close();

ps.close();

}

}

public static void main(String[] args) throws SQLException {

Scanner scanner = new Scanner(System.in);

for (int i = 1; i <= 2; i++) {

System.out.println("Enter Last Name>>");

String lastName = scanner.next();

System.out.println("Enter first Name>>");

String firstName = scanner.next();

System.out.println("Enter Address>>");

String address = scanner.next();

System.out.println("Enter City>>");

String city = scanner.next();

System.out.println("Enter salary>>");

String salary = scanner.next();

UserDemo userDemo = new UserDemo();

userDemo.insertUserDetails(lastName, firstName, address, city, salary);

}

}

}