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

```
package com.insertion;

import java.sql.Connection;
import java.sql.DriverManager;

public class ConnectionTest {

	Connection connection = null;

	public Connection getConnectionDetails() { //utility
		try {
			Class.forName("com.mysql.jdbc.Driver");

			connection =
DriverManager.getConnection("jdbc:mysql://localhost:3306/test",
"root", "root");
		} catch (Exception e) {
			e.printStackTrace();
		}

		return connection;
	}
}


package com.insertion;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.util.Scanner;

public class UserInput {

	Connection connection = null;
```

```
	PreparedStatement ps = null;

	private void insertStudentData(String name, String city,
String mobile) throws SQLException {

		try {

			ConnectionTest connectionTest = new
ConnectionTest();
			connection =
connectionTest.getConnectionDetails();

			ps = connection.prepareStatement("insert into
student(name,city,mobile)values(?,?,?)");
			ps.setString(1, name);
			ps.setString(2, city);
			ps.setString(3, mobile);

			int i = ps.executeUpdate();
			System.out.println("Record is inserted
successfully.." + i);
		} catch (Exception e) {
			e.printStackTrace();
		} finally {
			connection.close();
			ps.close();
		}
	}

	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 name>>");
			String name = sc.next();
			System.out.println("Enter city>>");
			String city = sc.next();
```

```
                System.out.println("Enter mobile>>");
                String mobile = sc.next();
                UserInput userInput = new UserInput();
                userInput.insertStudentData(name, city, mobile);
            }
            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.