**Practical 2.3**

**Aim:**

Write RMI application where client will send Enrolment number and Name of a student to the server. Server will Open a database connection and store the Information in Database.

**Program code:**

**RemoteInterface.java**

**import** java**.**rmi**.\*;**

public interface RemoteInterface **extends** Remote

**{**

public boolean insertInDB**(**String enroll**,** String name**)throws** Exception**;**

**}**

**ServerImplements.java**

**import** java**.**rmi**.\*;**

**import** java**.**rmi**.**server**.\*;**

**import** java**.**lang**.**String**;**

**import** java**.**sql**.\*;**

public class ServerImplements **extends** UnicastRemoteObject **implements** RemoteInterface **{**

// JDBC driver name and database URL

static final String JDBC\_DRIVER **=** "org.postgresql.Driver"**;**

static final String DB\_URL **=** "jdbc:postgresql://127.0.0.1:5433/postgres"**;**

// Database credentials

static final String USER **=** "postgres"**;**

static final String PASS **=** "12345"**;**

public ServerImplements**()throws** Exception **{**

**super();**

**}**

public boolean insertInDB**(**String enroll**,** String name**)** **{**

Connection conn **=** **null;**

Statement stmt **=** **null;**

System**.**out**.**println**(**enroll **+** name**);**

**try{**

//STEP 2: Register JDBC driver

Class**.**forName**(**JDBC\_DRIVER**);**

//STEP 3: Open a connection

System**.**out**.**println**(**"Connecting to selected database..."**);**

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

System**.**out**.**println**(**"Connected database successfully..."**);**

//STEP 4: Execute a query

System**.**out**.**println**(**"Inserting data in Table 'studentinfo' of Database 'postgres'..."**);**

stmt **=** conn**.**createStatement**();**

String sql **=** "INSERT INTO studentinfo(enrollno, name) VALUES ('"**+**enroll**+**"', '"**+**name**+**"')"**;**

stmt**.**executeUpdate**(**sql**);**

System**.**out**.**println**(**"Data Inserted successfully..."**);**

**}** **catch(**SQLException se**)** **{**

//Handle errors for JDBC

se**.**printStackTrace**();**

**}** **catch(**Exception e**)** **{**

//Handle errors for Class.forName

e**.**printStackTrace**();**

**}finally{**

//finally block used to close resources

**try{**

**if(**stmt**!=null)**

stmt**.**close**();**

**}** **catch(**SQLException se2**)** **{**

**}**// nothing we can do

**try{**

**if(**conn**!=null)**

conn**.**close**();**

**}** **catch(**SQLException se**)** **{**

se**.**printStackTrace**();**

**}**//end finally try

**}**//end try

System**.**out**.**println**(**"Goodbye!"**);**

**return** **true;**

**}**

**}**

**Server.java**

**import** java**.**rmi**.\*;**

**import** java**.**net**.\*;**

public class Server **{**

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

**try {**

ServerImplements s**=new** ServerImplements**();**

Naming**.**rebind**(**"SERVICE3"**,**s**);**

System**.**out**.**println**(**"Server Started "**);**

**}** **catch(**Exception e**)** **{**

System**.**out**.**println**(**e**.**getMessage**());**

**}**

**}**

**}**

**Client.java**

**import** java**.**rmi**.\*;**

**import** java**.**io**.\*;**

**import** java**.**util**.**Scanner**;**

public class Client **{**

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

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

String enroll**,** name**;**

**try** **{**

System**.**out**.**println**(**"Enroll no: 130050131067"**);**

String ip**=**"rmi://localhost/SERVICE3"**;**

RemoteInterface s **=** **(**RemoteInterface**)**Naming**.**lookup**(**ip**);**

System**.**out**.**println**(**"Enter Enrollno: "**);**

enroll **=** in**.**nextLine**();**

System**.**out**.**println**(**"Enter Name: "**);**

name **=** in**.**nextLine**();**

**if(**s1**.**insertInDB**(**enroll**,** name**)){**

System**.**out**.**println**(**"Data Inserted."**);**

**}else** **{**

System**.**out**.**println**(**"Data Not Inserted."**);**

**}**

**}** **catch(**Exception e**)** **{**

System**.**out**.**println**(**e**.**getMessage**());**

e**.**printStackTrace**();**

**}**

**}**

**}**

**Input Output:**

