**RAJALAKSHMI ENGINEERING COLLEGE**

**RAJALAKSHMI NAGAR, THANDALAM – 602 105**

****

|  |
| --- |
| **CS19541**  **COMPUTER NETWORKS LABORATORY** |
| **Laboratory Manual Note Book** |

|  |
| --- |
| **Name : Mohan Vaithya.E**  **Year/Branch/Section : 3rd year - Computer Science and Design**  **Register No. : 221701037**  **Semester : 5**  **Academic Year : 2024-2025** |

**EXP NO : 8 TCP SOCKET**

**DATE : 09/09/24**

**AIM:**

To write a Java program to implement echo client server using TCP sockets.

**Software Used:**

IntelliJ IDEA

**PROCEDURE:**

### Echo Server:

1. Initialize a ServerSocket on a specific port.
2. Use accept() on the ServerSocket to wait for a client.
3. Obtain InputStream and OutputStream from the client Socket.
4. Continuously read data from the client.
5. Send the received message back to the client as an echo.
6. If the client sends "exit", close the connection.
7. Close the InputStream, OutputStream, client Socket, and ServerSocket.

Echo Client:

1. Connect to the server’s IP address and port.
2. Get OutputStream to send data and InputStream to receive data.
3. Take user input and send it to the server.
4. Receive and display the server's response.
5. If the user inputs "exit", close the connection.
6. Close the InputStream, OutputStream, and Socket.

**PROGRAM:**

HelloWorld.java:

import java.net.\*;  
import java.io.\*;  
  
public class HelloWorld {  
 public static void main(String[] args) {  
 try (ServerSocket ss = new ServerSocket(5000)) {  
 System.*out*.println("Waiting for Client...");  
  
 *// Accept client connection* try (Socket client = ss.accept();  
 DataInputStream input = new DataInputStream(client.getInputStream());  
 DataOutputStream output = new DataOutputStream(client.getOutputStream())) {  
  
 System.*out*.println("Client connected!");  
  
 *// Receive and send back a message from the client* String message = input.readUTF();  
 System.*out*.println("Received from client: " + message);  
  
 output.writeUTF("Echo: " + message);  
 }  
 } catch (IOException e) {  
 System.*out*.println("Server error: " + e.getMessage());  
 }  
 }  
}

**OUTPUT**:



Client.java:

import java.net.\*;  
import java.io.\*;  
  
public class Client {  
 public static void main(String[] args) {  
 try (Socket client = new Socket("localhost", 5000);  
 DataOutputStream output = new DataOutputStream(client.getOutputStream());  
 DataInputStream input = new DataInputStream(client.getInputStream())) {  
  
 System.*out*.println("Client is connected to server!");  
  
 *// Send a message to the server* output.writeUTF("Hello, Server!");  
  
 *// Receive a response from the server* String response = input.readUTF();  
 System.*out*.println("Received from server: " + response);  
  
 } catch (IOException e) {  
 System.*out*.println("Error: " + e.getMessage());  
 }  
 }  
}

**OUTPUT:**



**RESULT:**

Thus, the Java program to implement echo client server using TCP sockets has been executed successfully.