Practical No. 2

Aim :- Create, debug and execute program based on TCP client server communication.

Theory :-

To examine a socket at any time for address and port information by using following methods.

① InetAddress getInetAddress()

Returns InetAddress associated with socket IP address.

② Int getPort()

Return the remote port which socket object is connected.

③ int getLocalhost()

Returns the local port to which socket object is connected

④ InputStream getInputStream()

Return the input stream associated with invoking socket.

⑤ ~~End~~ getOutputStream()

Returns the output stream associated with invoking socket.

Establishing a TCP connection using TCP socket.

① Server need to instantiate server socket object

② Server invokes an accept() by server socket class

③ client → socket object which has server name & port Number

④ Attempt by client to connect

Some methods used in server sockets

i accept() - it listen for connection & accepts it.

ii close() - closes this socket

iii getLocalPort() - returns port on which this socket is listening

Program:-

client.java 1 ×

client.java > client > main(String[])

```java
import java.net.*;
import java.io.*;

public class client {
    Run | Debug
    public static void main(String[] args) throws IOException {
        Socket s = new Socket("localhost", 4999);

        PrintWriter pr = new PrintWriter(s.getOutputStream());
        pr.println("Is it working?");
        pr.flush();

        InputStreamReader in = new InputStreamReader(s.getInputStream());
        BufferedReader bf = new BufferedReader(in);

        String str = bf.readLine();
        System.out.println("Server:"+str);
    }
}
```

server.java 1 ×

server.java > server > main(String[])

```java
import java.net.*;
import java.io.*;

public class server {
    Run | Debug
    public static void main(String[] args) throws IOException {
        ServerSocket ss = new ServerSocket(4999);
        Socket s = ss.accept();

        System.out.println("Client is Connected");

        InputStreamReader in = new InputStreamReader(s.getInputStream());
        BufferedReader bf = new BufferedReader(in);

        String str = bf.readLine();
        System.out.println("Client:"+str);

        PrintWriter pr = new PrintWriter(s.getOutputStream());
        pr.println("Yess!");
        pr.flush();
    }
}
```