Atharva Yadav

Roll No. 127

Batch : s23

Network Lab Assignment No.8

AIM: SOCKET PROGRAMMING USING TCP/IP PROTOCOL

Theory: To connect to another machine we need a socket connection. A socket connection means the two machines have information about each other's network location (IP Address) and TCP port. The java.net.Socket class represents a Socket. environments.

Java Program for Client application

CODE:

```
import java.io.*; import java.net.*; public class Client {
public static void main(String[] args) {  try {
 Socket socket = new Socket("localhost", 5000);  System.out.println("Connected to server.");
 BufferedReader reader = new BufferedReader(new
InputStreamReader(socket.getInputStream()));
 PrintWriter writer = new PrintWriter(socket.getOutputStream(), true);
 BufferedReader consoleReader = new BufferedReader(new
InputStreamReader(System.in));

 String inputLine, outputLine;  while (true) {
 System.out.print("Client: ");  outputLine =
consoleReader.readLine();  writer.println(outputLine);  if
(outputLine.equalsIgnoreCase("bye")) break;  inputLine =
reader.readLine();
 System.out.println("Server: " + inputLine);  if
(inputLine.equalsIgnoreCase("bye")) break;
}  writer.close();  reader.close();
socket.close();  } catch (IOException
e) {
 e.printStackTrace();
 }
 }
}
```

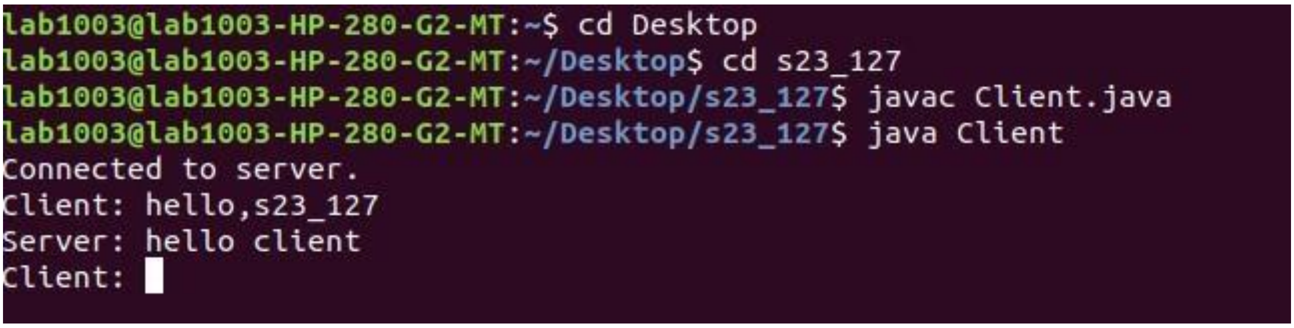
Java Program for Server application

CODE:

```
import java.io.*; import java.net.*; public class Server {
public static void main(String[] args) {  try {
 ServerSocket = new ServerSocket(5000);  System.out.println("Server started, waiting for client...");
 Socket clientSocket = serverSocket.accept();
 System.out.println("Client connected: " + clientSocket);
 BufferedReader reader = new BufferedReader(new
InputStreamReader(clientSocket.getInputStream()));
 PrintWriter writer = new PrintWriter(clientSocket.getOutputStream(), true);
 BufferedReader consoleReader = new BufferedReader(new
InputStreamReader(System.in));  String inputLine,
outputLine;
 while ((inputLine = reader.readLine()) != null) {
System.out.println("Client: " + inputLine);  if
(inputLine.equalsIgnoreCase("bye")) break; System.out.print("Server:
");  outputLine = consoleReader.readLine();  writer.println(outputLine);
if (outputLine.equalsIgnoreCase("bye")) break;
 }  writer.close();  reader.close();
clientSocket.close();
serverSocket.close();  } catch
(IOException e) {
 e.printStackTrace();
 }
 }
}
```

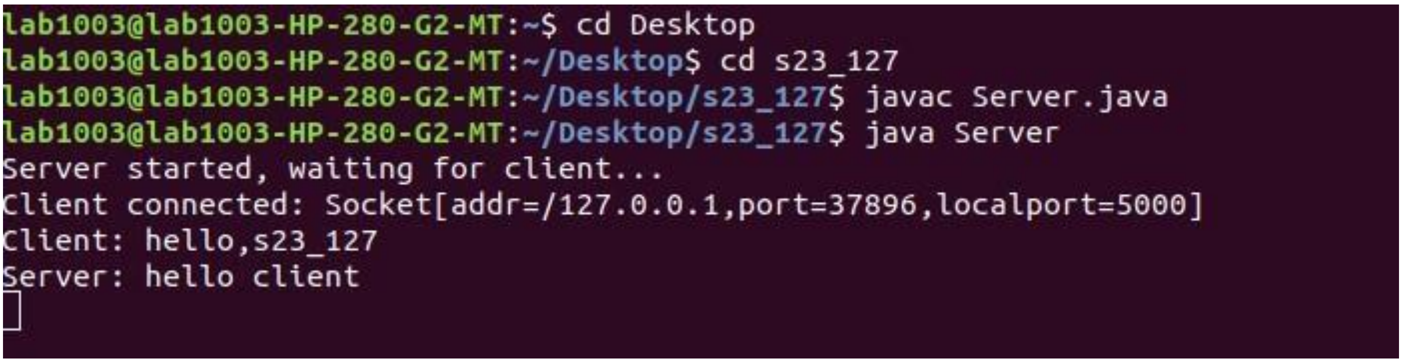
OUTPUT:

Client:



```
lab1003@lab1003-HP-280-G2-MT:~$ cd Desktop
lab1003@lab1003-HP-280-G2-MT:~/Desktop$ cd s23_127
lab1003@lab1003-HP-280-G2-MT:~/Desktop/s23_127$ javac Client.java
lab1003@lab1003-HP-280-G2-MT:~/Desktop/s23_127$ java Client
Connected to server.
Client: hello,s23_127
Server: hello client
Client: 
```

Server:



```
lab1003@lab1003-HP-280-G2-MT:~$ cd Desktop
lab1003@lab1003-HP-280-G2-MT:~/Desktop$ cd s23_127
lab1003@lab1003-HP-280-G2-MT:~/Desktop/s23_127$ javac Server.java
lab1003@lab1003-HP-280-G2-MT:~/Desktop/s23_127$ java Server
Server started, waiting for client...
Client connected: Socket[addr=/127.0.0.1,port=37896,localport=5000]
Client: hello,s23_127
Server: hello client
```

Conclusion:TCP protocol, ensuring reliable data transfer between programs on different machines. applications.We likely explored the concepts of sockets, which act as endpoints for communication, and delved into the The network assignment on socket programming using TCP equips us with a foundational skill for building network

Based On IO 6: To design and configure a network for an organization