

Experiment - 3.1

Aim:

Write a Java program to demonstrate the concept of socket programming.

Code:

Client side code:

```
import java.net.*;
import java.io.*;
class MyClient{
    public static void main(String args[])throws Exception{
        Socket s=new Socket("localhost",3333);
        DataInputStream din=new DataInputStream(s.getInputStream());
        DataOutputStream dout=new DataOutputStream(s.getOutputStream());
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));

        String str="",str2="";
        while(!str.equals("stop")){
            str=br.readLine();
            dout.writeUTF(str);
            dout.flush();
            str2=din.readUTF();
            System.out.println("Server says: "+str2);
        }

        dout.close();
        s.close();
    }
}
```

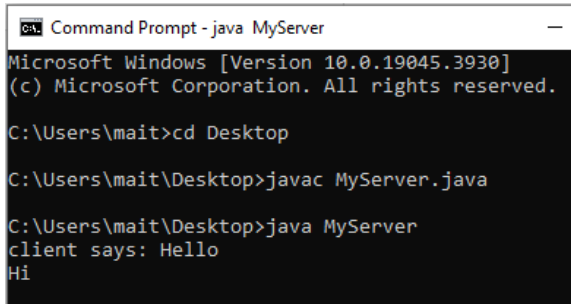
Server side code:

```
import java.net.*;
import java.io.*;
class MyServer{
    public static void main(String args[])throws Exception{
        ServerSocket ss=new ServerSocket(3333);
        Socket s=ss.accept();
        DataInputStream din=new DataInputStream(s.getInputStream());
        DataOutputStream dout=new DataOutputStream(s.getOutputStream());
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));

        String str="",str2="";
        while(!str.equals("stop")){
            str=din.readUTF();
            System.out.println("client says: "+str);
            str2=br.readLine();
        }
    }
}
```

```
dout.writeUTF(str2);
dout.flush();
}
din.close();
s.close();
ss.close();
}}
```

Output:

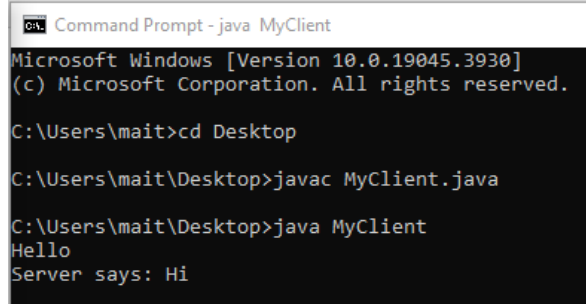


```
Command Prompt - java MyServer
Microsoft Windows [Version 10.0.19045.3930]
(c) Microsoft Corporation. All rights reserved.

C:\Users\mait>cd Desktop

C:\Users\mait\Desktop>javac MyServer.java

C:\Users\mait\Desktop>java MyServer
client says: Hello
Hi
```



```
Command Prompt - java MyClient
Microsoft Windows [Version 10.0.19045.3930]
(c) Microsoft Corporation. All rights reserved.

C:\Users\mait>cd Desktop

C:\Users\mait\Desktop>javac MyClient.java

C:\Users\mait\Desktop>java MyClient
Hello
Server says: Hi
```

Experiment - 3.2

Aim:

Write a Java program to implement Datagram UDP socket programming.

Code:

Client side code:

```
import java.io.IOException;
import java.net.DatagramPacket;
import java.net.DatagramSocket;
import java.net.InetAddress;
import java.util.Scanner;

public class Client
{
    public static void main(String args[]) throws IOException
    {
        Scanner sc = new Scanner(System.in);
        DatagramSocket ds = new DatagramSocket();
        InetAddress ip = InetAddress.getLocalHost();
        byte buf[] = null;
        while (true)
        {
            String inp = sc.nextLine();
            buf = inp.getBytes();
            DatagramPacket DpSend =
                new DatagramPacket(buf, buf.length, ip, 1234);
            ds.send(DpSend);
            if (inp.equals("bye"))
                break; } } }
```

Server side code:

```
import java.io.IOException;
import java.net.DatagramPacket;
import java.net.DatagramSocket;
import java.net.InetAddress;
import java.net.SocketException;

public class Server
{
    public static void main(String[] args) throws IOException
```

```

{
    DatagramSocket ds = new DatagramSocket(1234);
    byte[] receive = new byte[65535];

    DatagramPacket DpReceive = null;
    while (true)
    {
        DpReceive = new DatagramPacket(receive, receive.length);

        ds.receive(DpReceive);

        System.out.println("Client:-" + data(receive));

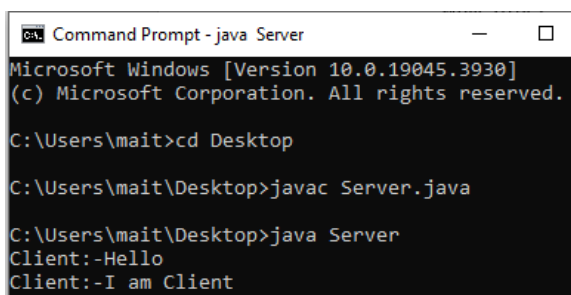
        if (data(receive).toString().equals("bye"))
        {
            System.out.println("Client sent bye.....EXITING");
            break; }

        receive = new byte[65535]; } }

public static StringBuilder data(byte[] a)
{
    if (a == null)
        return null;
    StringBuilder ret = new StringBuilder();
    int i = 0;
    while (a[i] != 0)
    {
        ret.append((char) a[i]);
        i++;
    }
    return ret; } }

```

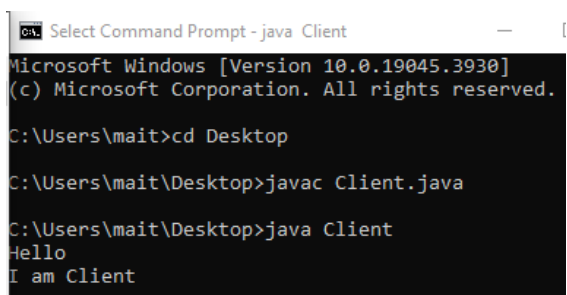
OUTPUT:



```

C:\Users\mait>cd Desktop
C:\Users\mait\Desktop>javac Server.java
C:\Users\mait\Desktop>java Server
Client:-Hello
Client:-I am Client

```



```

C:\Users\mait>cd Desktop
C:\Users\mait\Desktop>javac Client.java
C:\Users\mait\Desktop>java Client
Hello
I am Client

```

Experiment - 3.3

Aim:

Write a Java program to implement socket programming for TCP in Java Server and Client Sockets.

Code:

Client side code :

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

public class Client {
    private Socket socket = null;
    private DataInputStream input = null;
    private DataOutputStream out = null;
    public Client(String address, int port)
    {
        // establish a connection
        try {
            socket = new Socket(address, port);
            System.out.println("Connected");
            input = new DataInputStream(System.in);
            out = new DataOutputStream(
                socket.getOutputStream());
        }
        catch (UnknownHostException u) {
            System.out.println(u);
            return;
        }
        catch (IOException i) {
            System.out.println(i);
            return;
        }
        String line = "";
```

```

        while (!line.equals("Over")) {
            try {
                line = input.readLine();
                out.writeUTF(line);
            }
            catch (IOException i) {
                System.out.println(i);
            }
        }
        try {
            input.close();
            out.close();
            socket.close();
        }
        catch (IOException i) {
            System.out.println(i);
        }
    }

    public static void main(String args[])
    {
        Client client = new Client("127.0.0.1", 5000);
    }
}

```

Output:

```

C:\Users\mait>cd Desktop
C:\Users\mait\Desktop>javac Server.java
C:\Users\mait\Desktop>java Server
Server started
Waiting for a client ...
Client accepted
hello server

```

```

C:\Users\mait>cd Desktop
C:\Users\mait\Desktop>javac Client.java
Note: Client.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
C:\Users\mait\Desktop>java Client
Connected
hello server

```