IT LAB ASSIGNMENT REPORT 5

NAME: Kshitij Dhande
ROLL NO: 16CS8076

QUESTION 5:
Create a simple “Client Server Programming” with JAVA. Here your client side has to pass a message to the server side. Server has to validate messages for proper number and return it’s double as response to the client. Make a simple GUI interface for the client.
CODE:

CLIENT:

```
// A Java program for a Client import java.net.*;
import java.io.*;
import javax.swing.JOptionPane;
public class Client
{
        // initialize socket and input output streams private Socket socket = null; private
        DataInputStream input = null; private DataOutputStream out = null; private
        DataInputStream in = null;
// constructor to put ip address and port public Client(String address, int port)
{
        // establish a connection try
        {
                socket = new Socket(address, port); System.out.println("Connected \n\n###
                Enter 'Over' to stop ###\n"); System.out.println("Enter any Number");
                // takes input from terminal
                input = new DataInputStream(System.in);
                // sends output to the socket
                out = new DataOutputStream(socket.getOutputStream()); in = new
                DataInputStream(new
                BufferedInputStream(socket.getInputStream()));
        }
```

```
  catch(UnknownHostException u)

                {
                        System.out.println(u);
                }
  catch(IOException i)
                {
                        System.out.println(i);
                }
// string to read message from input String line = "";
// keep reading until "Over" is input from terminal while (!line.equals("Over"))
{
//sending message try
        {
                enter 'Over' to stop");
        }
                // line = input.readLine();
                line = JOptionPane.showInputDialog("Enter a number or
                out.writeUTF(line);
               catch(IOException i)
                {
                        System.out.println(i);
                }
//reading message from server try
{
        line = in.readUTF();
        System.out.println(line);
        JOptionPane.showMessageDialog(null, line , "Results",
        JOptionPane.PLAIN_MESSAGE );
}
catch(IOException i)
                {
                System.out.println(i);
                }
}
// close the connection try

{
        input.close();
        out.close();
        socket.close();
}
```

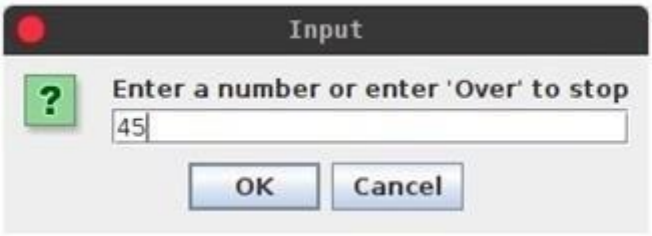
```
catch(IOException i)
    {
        System.out.println(i);
    }
}

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

SERVER:

```
// A Java program for a Client import java.net.*;
import java.io.*;
import javax.swing.JOptionPane;
public class Client
{
    // initialize socket and input output streams private Socket socket = null; private
    DataInputStream input = null; private DataOutputStream out = null; private
    DataInputStream in = null;
// constructor to put ip address and port public Client(String address, int port)
{
    // establish a connection try
{
    socket = new Socket(address, port);
    System.out.println("Connected \n\n### Enter 'Over' to stop ###\n");
    System.out.println("Enter any Number");
// takes input from terminal
    input = new DataInputStream(System.in);
// sends output to the socket
    out = new DataOutputStream(socket.getOutputStream()); in = new
    DataInputStream(new
    BufferedInputStream(socket.getInputStream()));
}
    catch(UnknownHostException u)
    {
        System.out.println(u);
    }
catch(IOException i)
    {
        System.out.println(i);
    }
// string to read message from input String line = "";
// keep reading until "Over" is input from terminal while (!line.equals("Over"))
{
//sending message try
    {
        enter 'Over' to stop");
    }
// line = input.readLine();
    line = JOptionPane.showInputDialog("Enter a number or
    out.writeUTF(line);
    catch(IOException i)
    {
        System.out.println(i);
    }
 //reading message from server try
{
    line = in.readUTF();
    System.out.println(line);
    JOptionPane.showMessageDialog(null, line , "Results",
    JOptionPane.PLAIN_MESSAGE );
}
    catch(IOException i)
    {
        System.out.println(i);
    }
}
  // close the connection 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);
    }
```

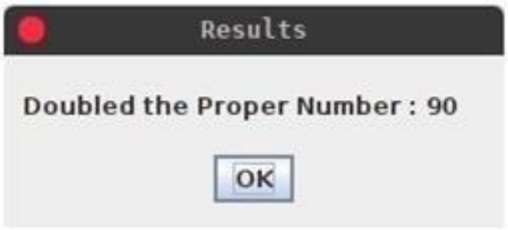
```
}
```

OUTPUT:

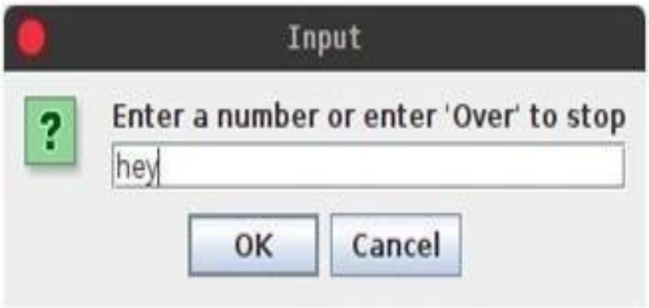

Input

Enter a number or enter 'Over' to stop

45

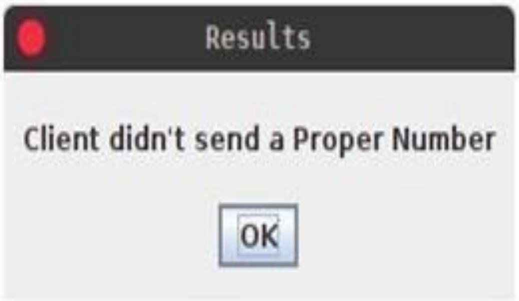

Results

Doubled the Proper Number : 90


Input

Enter a number or enter 'Over' to stop

hey


Results

Client didn't send a Proper Number