**Experiment No: Date:**

Aim: To develop a chat application using socket

Theory:

In order to initiate a TCP session, a server and a client are required. Firstly, a server is set up to listen at a given port. The server waits and does nothing until a client attempts to connect that port. If everything goes fine, the connection is successful and both the server and client have an instance of the Socket class.

From each instance of this class, an input stream and an output stream can be obtained, and all communication is done via these streams.

The Socket class is in the java.net package that is why import java.net.\*; at the beginning of the file.

What is a socket?

We don't directly access the network interface device to send and receive packets. Instead, an intermediary connector is created to handle the programming interface to the network.Thus a socket is a connector that connects

application to a network interface of your computer.It is a method for communication between a client program and a server program in a network. A socket is defined as "the endpoint in a connection." Sockets are created and used with a set of programming methods or "function calls" sometimes called the sockets application programming interface (API). For sending and receiving data to and from the network we have to call the socket's methods.

CODE for chat application on two different hosts.

CLIENT SIDE:

**import** java.io.\*;

**import** java.net.\*;

**public** **class** client

{

**public** **static** **void** main(String a[])**throws** IOException

{

**try**

{

Socket con=**new** Socket("172.18.39.103",1295);

BufferedReader in=**new** BufferedReader(**new**

InputStreamReader(con.getInputStream()));

PrintWriter out=**new** PrintWriter(con.getOutputStream(),**true**);

**while**(**true**)

{

String s1=in.readLine();

System.*out*.println("From Server:"+s1);

System.*out*.print("Enter the messages to the server:");

BufferedReader din=**new** BufferedReader(**new**

InputStreamReader(System.*in*));

String st=din.readLine();

out.println(st);

**if**(st.equalsIgnoreCase("Bye")||st==**null**)**break**;

}

in.close();

out.close();

con.close();

}

**catch**(UnknownHostException e){ }

}

SERVER SIDE:

**import** java.io.\*;

**import** java.net.\*;

**public** **class** server

{

**public** **static** **void** main(String a[])**throws** IOException

{

**try**

{

ServerSocket s=**new** ServerSocket(1295);

System.*out*.println("Server Waiting For The Client");

Socket cs=s.accept();

InetAddress ia=cs.getInetAddress();

String cli=ia.getHostAddress();

System.*out*.println("Connected to the client with IP:"+cli);

BufferedReader in=**new** BufferedReader(**new**

InputStreamReader(cs.getInputStream()));

PrintWriter out=**new** PrintWriter(cs.getOutputStream(),**true**);

**do**

{

BufferedReader din=**new** BufferedReader(**new**

InputStreamReader(System.*in*));

System.*out*.print("To Client:");

String tocl=din.readLine();

out.println(tocl);

String st=in.readLine();

**if**(st.equalsIgnoreCase("Bye")||st==**null**)**break**;

System.*out*.println("From Client:"+st);

}**while**(**true**);

in.close();

out.close();

cs.close();

}

**catch**(IOException e) { }

}

}

OUTPUT:

SERVER SIDE:

Server Waiting For The Client

Connected to the client with IP:172.18.39.38

To Client:hi abc

From Client:hello m abc client

To Client:i m server! you plz just close the connection!

CLIENT SIDE:

From Server:hi abc

Enter the messages to the server:hello m abc client

From Server:i m server! you plz just close the connection!

Enter the messages to the server:bye

CODE for chat application on one host.

CLIENT SIDE:

**import** java.io.\*;

**import** java.net.\*

**public** **class** client

{

**public** **static** **void** main(String a[])**throws** IOException

{

**try**

{

Socket con=**new** Socket("127.0.0.1",1295);

BufferedReader in=**new** BufferedReader(**new**

InputStreamReader(con.getInputStream()));

PrintWriter out=**new** PrintWriter(con.getOutputStream(),**true**);

**while**(**true**)

{

String s1=in.readLine();

System.*out*.println("From Server:"+s1);

System.*out*.print("Enter the messages to the server:");

BufferedReader din=**new** BufferedReader(**new**

InputStreamReader(System.*in*));

String st=din.readLine();

out.println(st);

**if**(st.equalsIgnoreCase("Bye")||st==**null**)**break**;

}

in.close();

out.close();

con.close();

}

**catch**(UnknownHostException e){ }

}

SERVER SIDE:

**import** java.io.\*;

**import** java.net.\*;

**public** **class** server

{

**public** **static** **void** main(String a[])**throws** IOException

{

**try**

{

ServerSocket s=**new** ServerSocket(1295);

System.*out*.println("Server Waiting For The Client");

Socket cs=s.accept();

InetAddress ia=cs.getInetAddress();

String cli=ia.getHostAddress();

System.*out*.println("Connected to the client with IP:"+cli);

BufferedReader in=**new** BufferedReader(**new**

InputStreamReader(cs.getInputStream()));

PrintWriter out=**new** PrintWriter(cs.getOutputStream(),**true**);

**do**

{

BufferedReader din=**new** BufferedReader(**new**

InputStreamReader(System.*in*));

System.*out*.print("To Client:");

String tocl=din.readLine();

out.println(tocl);

String st=in.readLine();

**if**(st.equalsIgnoreCase("Bye")||st==**null**)**break**;

System.*out*.println("From Client:"+st);

}**while**(**true**);

in.close();

out.close();

cs.close();

}

**catch**(IOException e) { }

}

}

OUTPUT:

From Server:Hi Client, I am server here.

Enter the messages to the server:Hi Server, I am client here

From Server:Ok Client now close the connection

Enter the messages to the server:Ok server bye

From Server:No, You must enter only bye

Enter the messages to the server:Bye