## CLIENT-TO-CLIENT CHATING

**Aim**

**Create classes needed for implementing chatting between 2 clients**

**Algorithm**

Create a class client1
Define Socket, Serversocket, bufferedreader, printwriter and 2 threads t1,t2
in constructor of client1create objects of classes Socket, Serversocket, bufferedreader, printwriter
run function
if thread is t1 read data from the keyboard and write it into the socket
if thread is t2 read data from the socket and write it into the keyboard
end run
main function
create an object of client1class
create a thread from client1 and assign it to thread t1
create a thread from client1 and assign it to thread t2
start threads t1 and t2
end of main function
end of class client1

Create a class server1
Define Socket, Serversocket, bufferedreader, printwriter and 2 threads t1,t2
in constructor of client1create objects of classes Socket, Serversocket, bufferedreader, printwriter
accept client socket using server socket
end of constructor
run function
if thread is t1 read data from the keyboard and write it into the socket
if thread is t2 read data from the socket and write it into the keyboard
end run
main function
create an object of server1class
create a thread from client1 and assign it to thread t1
create a thread from client1 and assign it to thread t2
start threads t1 and t2
end of main function
**end of class server1**

**PROGRAM**

**Client side program**

```
import java.net.*;
import java.io.*;
import java.lang.*;
class client1 implements Runnable
{
int i=0,j=0;
String msgk,msgs;
ServerSocket ser;
Thread t1=null,t2=null;
Socket soc;
```

```
Socket socc;
BufferedReader brk,brc;
PrintWriter pw;


public client1()
        {
        try{
                soc = new Socket("localhost",8000);
                brk=new BufferedReader(new InputStreamReader(System.in));
                pw=new PrintWriter(soc.getOutputStream(),true);
                brc=new BufferedReader(new InputStreamReader(soc.getInputStream()));
                }
        catch(Exception e){}
        }
public void run()
        {
        if (Thread.currentThread()==t1)
                {try
                        {for(;;)
                                {msgk=brk.readLine();
                                pw.println(msgk);
                                }
                        }
                catch(Exception e){}
                }
        else
        {
                try
                        {do
                                {msgs=brc.readLine();
                                System.out.println('from server'+msgs);

                                }
                        while(!msgs.equals("quit"));
                        }catch(Exception e){}
        }
        System.exit(0);
}

        public static void main(String args[]) {
                client1 sero =new client1();
                sero.t1=new Thread(sero);
                sero.t2=new Thread(sero);
                sero.t1.start();
                sero.t2.start();
        }
}
```

**Server side program**

```
import java.net.*;
import java.io.*;
import java.lang.*;
class Server1 implements Runnable
{
int i=0,j=0;
String msgk,msgs;
ServerSocket ser;
Thread t1=null,t2=null;
Socket soc;
Socket socc;
BufferedReader brk,brc;
PrintWriter pw;


public Server1()
        {
        try{
                ser= new ServerSocket(8000);
                soc=ser.accept();
                brk=new BufferedReader(new InputStreamReader(System.in));
                pw=new PrintWriter(soc.getOutputStream(),true);
                brc=new BufferedReader(new InputStreamReader(soc.getInputStream()));
                }
        catch(Exception e){}
        }
public void run()
        {
        if (Thread.currentThread()==t1)
                {try
                        {for(;;)
                                {msgk=brk.readLine();
                                pw.println(msgk);
                                }
                        }
                catch(Exception e){}
                }
        else
        {
                try
                        {do
                                {msgs=brc.readLine();
                                System.out.println('from client'+msgs);

                                }
                        while(!msgs.equals("quit"));
                        }catch(Exception e){}
```

```
        }
        System.exit(0);
}

        public static void main(String args[]) {
                Server1 sero =new Server1();
                sero.t1=new Thread(sero);
                sero.t2=new Thread(sero);
                sero.t1.start();
                sero.t2.start();
        }
}
```

**TEST CASE**

CASE1

Run the client program first then run the server program

CASE2

Run the server program first then the client program,and check the 2 way chatting

**OUTPUT**

server machine

```
c:>java server1
from client hai
ok
quit
```

client machine

```
c:>java client1
hai
from server ok
from server quit
```