

```

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

public class client
{
    Socket socket;

    BufferedReader br;
    PrintWriter out;

    public client ()
    {
        try
        {
            System.out.println("Sending request to sever");
            socket=new Socket("127.0.0.1",7777);
            System.out.println("connection done.");

            br=new BufferedReader(new InputStreamReader(socket.getInputStream()));
            out=new PrintWriter(socket.getOutputStream());

            startReading();
            startWriting();

        }catch (Exception e)
        {
            //handle exception
        }
    }

    public void startReading()
    {
        //thread-read karke deta rahega
        Runnable r1=()->
        {
            System.out.println("reader started..");
            while(true)
            {
                try {
                    String msg = br.readLine();
                    if (msg.equals("exit")) {
                        System.out.println("server terminated the chat");
                        break;
                    }
                    System.out.println("server :" + msg);
                }catch(Exception e)
                {
                    e.printStackTrace();
                }
            }
        };
        new Thread(r1).start();
    }

    public void startWriting()
    {
        //thread - data user lega and the send karega client tk
        Runnable r2=()->
        {

```

```

        System.out.println("writer started..");
        while(true)
        {
            try
            {
                BufferedReader br1 = new BufferedReader(new
InputStreamReader(System.in));
                String content =br1.readLine();
                out.println(content);
                out.flush();
            }catch (Exception e)
            {
                //TODO:handle exception
                e.printStackTrace();
            }
        }
    };
    new Thread(r2).start();
}
public static void main (String[] args)
{
    System.out.println("this is client...");
    new client();
}
}

```

```

import java.net.*;
import java.io.*;
public class server
{
    ServerSocket server;
    Socket socket;

    BufferedReader br;
    PrintWriter out;
    //Constructor..
    public server()
    {
        try
        {
            server=new ServerSocket(7777);
            System.out.println("server is ready to accept connection");
            System.out.println("waiting...");
            socket=server.accept();

            br=new BufferedReader(new InputStreamReader(socket.getInputStream()));
            out=new PrintWriter(socket.getOutputStream());

            startReading();
            startWriting();

        } catch(Exception e)
        {
            e.printStackTrace();
        }
    }

    public void startReading()
    {
        //thread-read karke deta rahega
        Runnable r1=()->
        {
            System.out.println("reader started..");
            while(true)
            {
                try
                {
                    String msg = br.readLine();
                    if (msg.equals("exit")) {
                        System.out.println("client terminated the
chat");
                        break;
                    }
                    System.out.println("Client :" + msg);
                }catch(Exception e)
                {
                    e.printStackTrace();
                }
            }
        };
        new Thread(r1).start();
    }

    public void startWriting()
    {
        //thread - data user lega and the send karega client tk
    }
}

```

```

Runnable r2=()->
{
    System.out.println("writer started..");
    while(true)
    {
        try
        {
            BufferedReader br1 = new BufferedReader(new
InputStreamReader(System.in));
            String content =br1.readLine();
            out.println(content);
            out.flush();
        }catch (Exception e)
        {
            //TODO:handle exception
            e.printStackTrace();
        }
    }
};
new Thread(r2).start();
}
public static void main(String[] args)
{
    System.out.println("this is server..going to start server");
    new server();
}
}

```