

## Practical No 1B

/\*Practical 1B: A client server TCP based chatting application.\*/

### 1. ChatClient.java

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

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

```
class ChatClient
```

```
{
```

```
    public static void main(String args[])
```

```
    {
```

```
        try
```

```
        {
```

```
            Socket s = new Socket("Localhost",8000);
```

```
            BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
```

```
            DataOutputStream out = new DataOutputStream(s.getOutputStream());
```

```
            DataInputStream in = new DataInputStream(s.getInputStream());
```

```
            String msg;
```

```
            System.out.println("To stop chatting with server type STOP");
```

```
            System.out.print("Client Says: ");
```

```
            while((msg = br.readLine()) != null)
```

```
{  
    out.writeBytes(msg+"\n");  
  
    if(msg.equals("STOP"))  
  
        break;  
  
    System.out.println("Server Says : "+ in.readLine());  
  
    System.out.print("Client Says : ");  
}  
  
br.close();  
in.close();  
out.close();  
s.close();  
}  
catch(Exception e)  
{  
    e.printStackTrace();  
}  
}  
}
```

2.ChatServer.java

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

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

```
class ChatServer
```

```
{
```

```
    public static void main(String args[])
```

```
    {
```

```
        try
```

```
        {
```

```
            ServerSocket ss = new ServerSocket(8000); System.out.println("Waiting for client to connect..");
```

```
            Socket s = ss.accept();
```

```
            BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
```

```
            DataOutputStream out = new DataOutputStream(s.getOutputStream());
```

```
            DataInputStream in = new DataInputStream(s.getInputStream());
```

```
            String receive, send;
```

```
            while((receive = in.readLine()) != null)
```

```
            {
```

```
                if(receive.equals("STOP"))
```

```
                break;
```

```
                System.out.println("Client Says : "+receive);
```

```
        System.out.print("Server Says : ");  
        send = br.readLine();  
        out.writeBytes(send+"\n");  
    }  
  
    br.close();  
    in.close();  
    out.close();  
    s.close();  
}  
catch(Exception e)  
{  
    e.printStackTrace();  
}  
}  
}
```