

```

1  package ch19.sec07;
2
3  import java.io.DataInputStream;
4  import java.io.DataOutputStream;
5  import java.io.IOException;
6  import java.net.InetSocketAddress;
7  import java.net.Socket;
8
9  import org.json.JSONObject;
10
11 public class SocketClient {
12     //필드
13     ChatServer chatServer;
14     Socket socket;
15     DataInputStream dis;
16     DataOutputStream dos;
17     String clientIp;
18     String chatName;
19
20     //생성자
21     public SocketClient(ChatServer chatServer, Socket socket) {
22         try {
23             this.chatServer = chatServer;
24             this.socket = socket;
25             this.dis = new DataInputStream(socket.getInputStream());
26             this.dos = new DataOutputStream(socket.getOutputStream());
27             InetSocketAddress isa = (InetSocketAddress) socket.getRemoteSocketAddress();
28             this.clientIp = isa.getHostName();
29             receive();
30         } catch (IOException e) {
31         }
32     }
33
34     //메소드: JSON 받기
35     public void receive() {
36         chatServer.threadPool.execute(() -> {
37             try {
38                 while(true) {
39                     String receiveJson = dis.readUTF();
40
41                     JSONObject jsonObject = new JSONObject(receiveJson);
42                     String command = jsonObject.getString("command");
43
44                     switch(command) {
45                         case "incoming":
46                             this.chatName = jsonObject.getString("data");
47                             chatServer.sendToAll(this, "들어오셨습니다.");
48                             chatServer.addSocketClient(this);
49                             break;
50                         case "message":
51                             String message = jsonObject.getString("data");
52                             chatServer.sendToAll(this, message);
53                             break;
54                     }
55                 }
56             } catch (IOException e) {
57                 chatServer.sendToAll(this, "나가셨습니다.");
58                 chatServer.removeSocketClient(this);
59             }
60         });
61     }
62
63     //메소드: JSON 보내기
64     public void send(String json) {
65         try {
66             dos.writeUTF(json);
67             dos.flush();
68         } catch (IOException e) {
69         }
70     }
71
72     //메소드: 연결 종료

```

```
73     public void close() {
74         try {
75             socket.close();
76         } catch (Exception e) {}
77     }
78 }
79
80
```