

# 서버프로그래밍 10주차 과제

202127064 이지호

# 1,2. 브로드캐스팅, 멀티캐스팅에 대한 간단한 설명

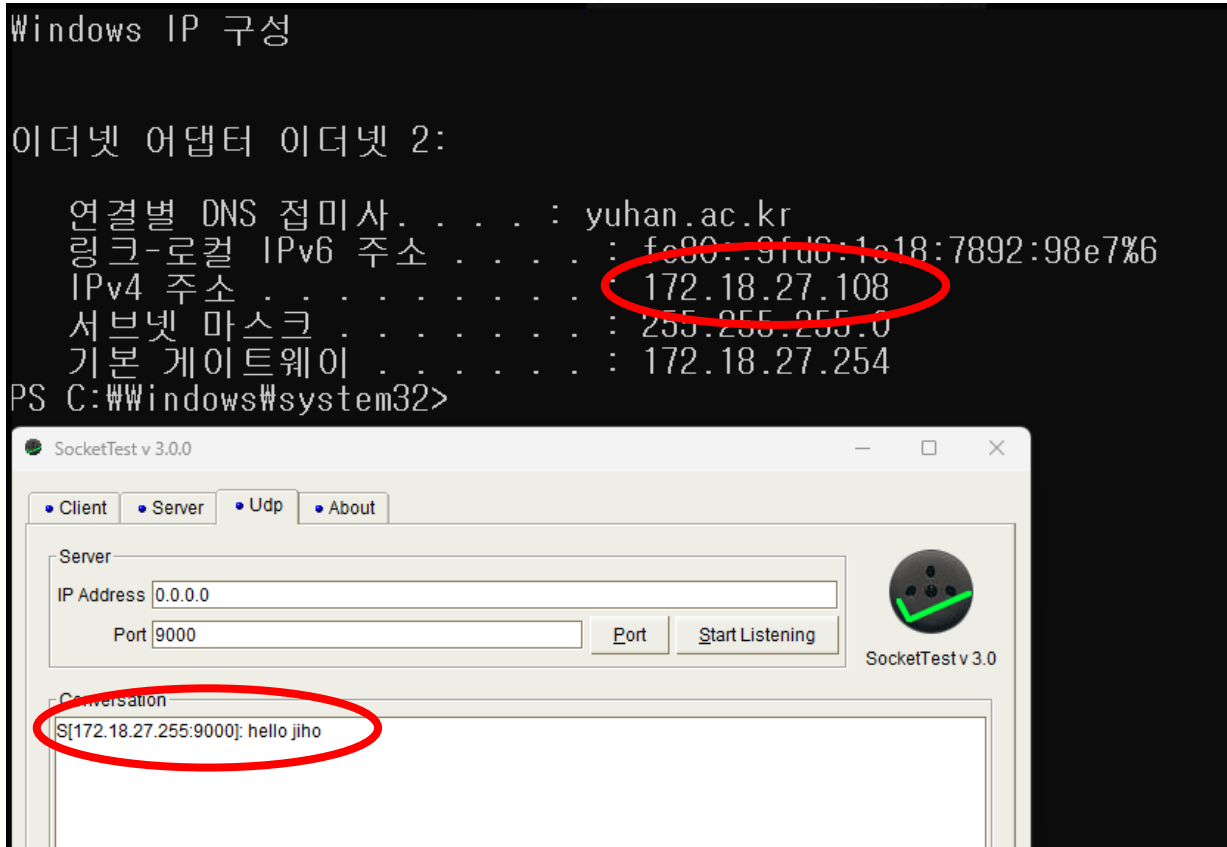
## 브로드캐스팅(Broadcasting)

	특징	사용되는 IP주소
한정 브로드캐스트	현재 서브넷 내의 모든 호스트에게 전송(라우터를 넘지 않고)	255.255.255.255
지정 브로드캐스트	해당 네트워크 내 모든 호스트에게 전송	112.37.65.255 (IP 주소 : 112.37.65.83, 서브넷 마스크 : 255.255.255.0)

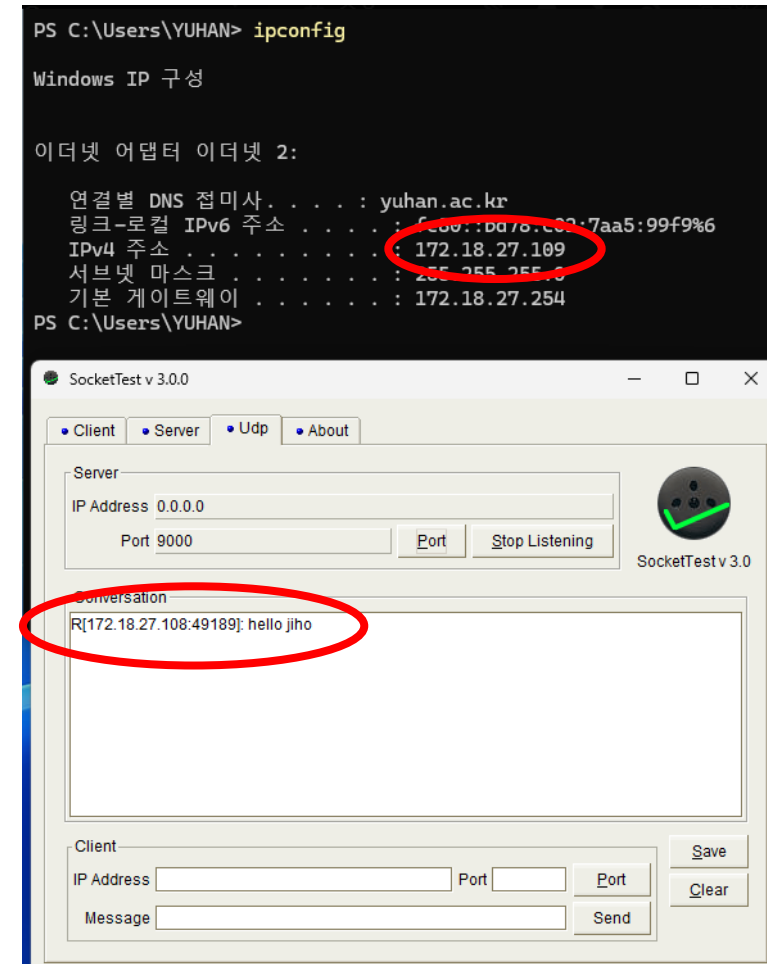
## 멀티캐스팅(Broadcasting)

	특징	사용되는 IP주소
예약된 멀티캐스트	라우팅 프로토콜 등에서 사용(라우터, 호스트 등등)	224.0.0.0 ~ 224.0.0.255
일반 멀티캐스트	어플리케이션 멀티캐스트(스트리밍 등등)	224.0.1.0 ~ 238.255.255.255
사설 멀티캐스트	조직 내부 전용	239.0.0.0 ~ 239.255.255.255

# 1. 브로드캐스팅 실습

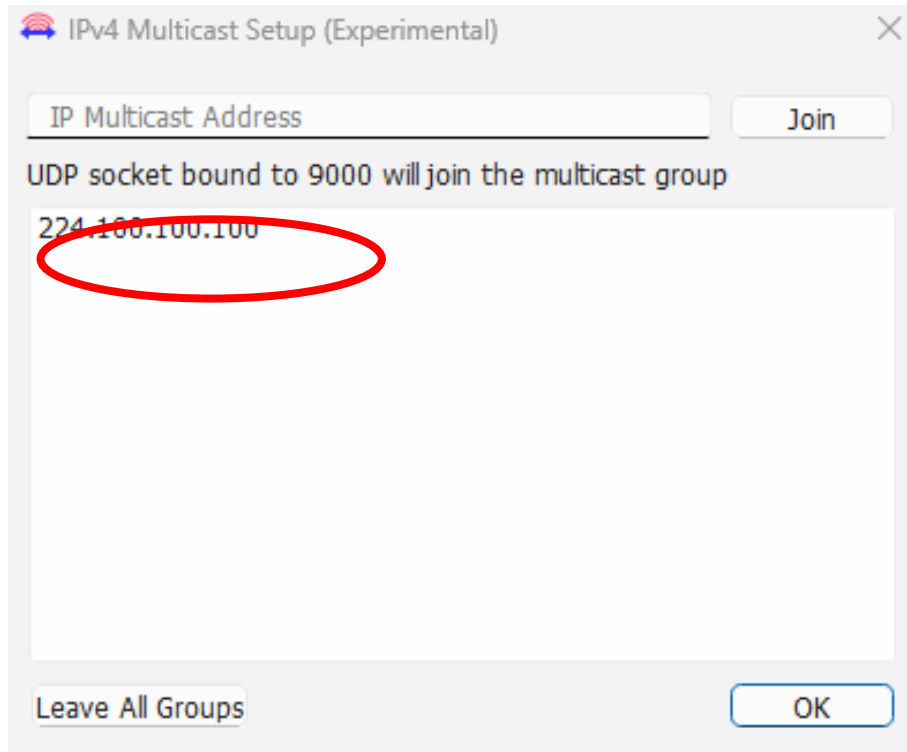


데이터를 전송한 컴퓨터

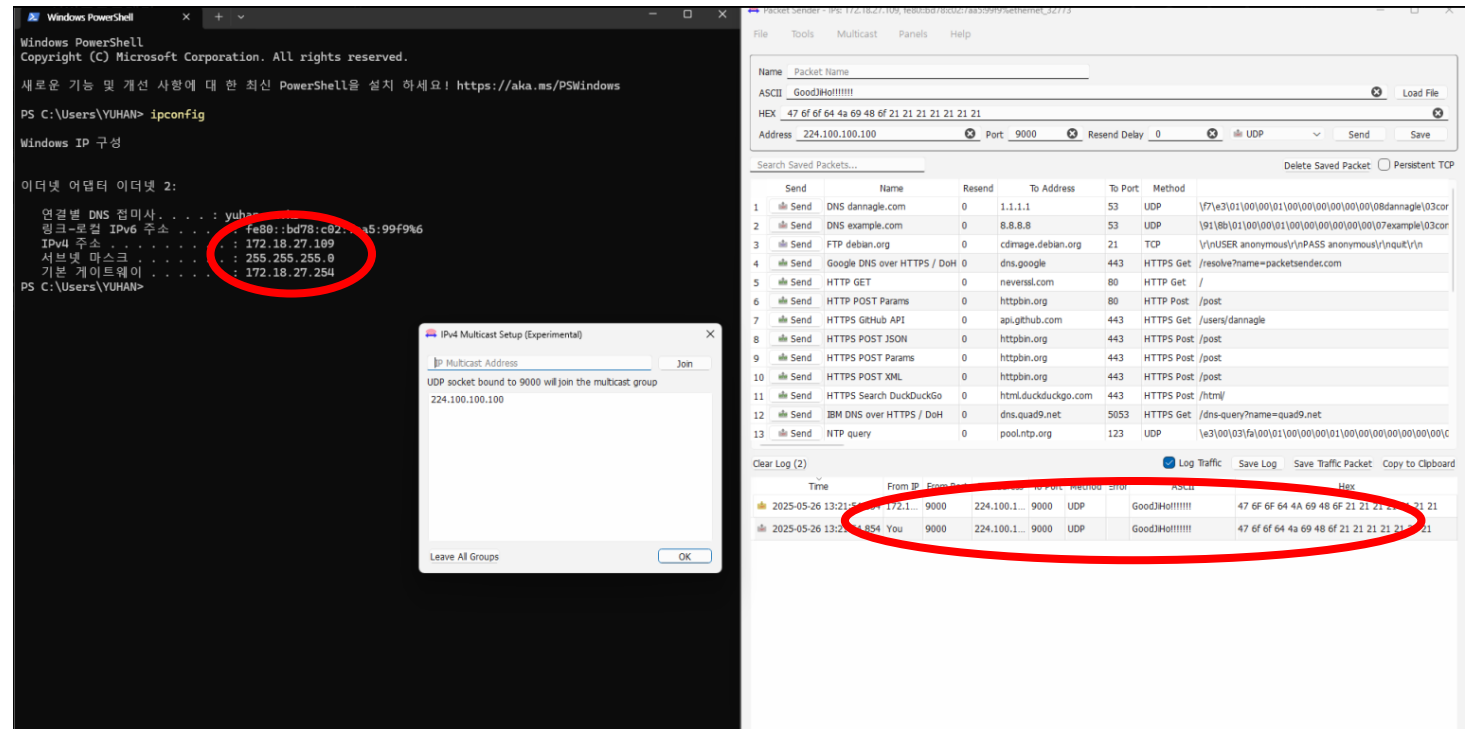


데이터를 받은 컴퓨터

## 2-1. 멀티캐스팅 실습



## 1. Packet Sender에서 멀티캐스트 주소를 설정하고 그룹에 참여



## 2. 데이터 입력 후 전송

## 2-2. 멀티캐스팅 실습

Copyright (C) Microsoft Corporation. All rights reserved.  
새로운 기능 및 개선 사항에 대한 최신 PowerShell을 설치하세요! https://aka.ms/PSWindows  
PS C:\Windows\system32> ipconfig  
Windows IP 구성  
  
이더넷 어댑터 이더넷 2:  
  
연결별 DNS 접미사. . . : yuhan.ac.kr  
링크-로컬 IPv6 주소. . . : fe80::8fdu:18:7892:98e7%6  
IPv4 주소. . . : 172.18.27.108  
서브넷 마스크. . . : 255.255.255.0  
기본 게이트웨이. . . : 172.18.27.254  
PS C:\Windows\system32>

Name Packet Name  
ASCII ASCII representation Load File  
HEX HEX representation  
Address 224.100.100.100 Port 9000 Resend Delay 0.0/blank off UDP Send Save  
Search Saved Packets... Delete Saved Packet Persistent TCP

Send	Name	Resend	To Address	To Port	Method
16	Send SSL cert mismatch	0	wrong.host.badssl.com	443	SSL
17	Send SSL HTTP example.com	0	example.com	443	SSL
18	Send SSL HTTPS packetsender.com	0	packetsender.com	443	SSL
19	Send TCP connection refused	0	1.1.1.1	200	TCP
20	Send TCP IPv6	0	2607:fb80:4002:c05:65	80	TCP
21	Send TCP ssh github.com	0	github.com	22	TCP
22	Send TCP HTTP packetsender.com	0	packetsender.com	80	TCP
23	Send Telnet RPG	0	avalon-rpg.com	23	TCP
24	Send UDP broadcast	0	255.255.255.255	5000	UDP
25	Send UDP IPv4 localhost macro	0	127.0.0.1	5000	UDP
26	Send UDP IPv6 localhost macro	0	::1	5000	UDP
27	Send UDP Multicast	0	224.0.0.51	5000	UDP
28	Send WHOIS google.com	0	192.34.234.30	43	TCP

Clear Log (1)

Time	From IP	From Port	To Address	To Port	Method	Error	ASCII	Hex
2025-05-26 13:19:58.873	172.18.27.108	9000	224.100.100.100	9000	UDP	Good!Ho!!!!!!	47 6f 6f 64 4a 69 48 6f 21 21 21 21 21	

IP Multicast Setup (Experimental)

IP Multicast Address: 224.100.100.100  
UDP socket bound to 9000 will join the multicast group

Join

Leave All Groups

OK

DTLS Server Disabled UDP:9000 TCP Server Disabled SSL Server Disabled IPv4 Mode

3. 멀티캐스트 그룹에 참여한 다른 컴퓨터

Packet Sender - IP: 172.18.27.107, fe80::285e:1b1c:8073:dd9a%6, Ethernet\_32773  
File Tools Multicast Panels Help  
Name Packet Name  
ASCII ASCII representation Load File  
HEX HEX representation  
Address 224.100.100.100 Port 9000 Resend Delay 0.0/blank off UDP Send Save  
Search Saved Packets... Delete Saved Packet Persistent TCP

Send	Name	Resend	To Address	To Port	Method
1	Send DNS damage.com	0	1.1.1.1	53	UDP
2	Send DNS example.com	0	8.8.8.8	53	UDP
3	Send FTP debian.org	0	cdmrae.debian.org	21	TCP
4	Send Google DNS over HTTPS / DoH	0	dns.google	443	HTTPS Get
5	Send HTTP GET	0	neverssl.com	80	HTTP Get
6	Send HTTP POST Params	0	httpbin.org	80	HTTP Post
7	Send HTTPS GitHub API	0	api.github.com	443	HTTPS Get
8	Send HTTPS POST JSON	0	httpbin.org	443	HTTPS Post
9	Send HTTPS POST Params	0	httpbin.org	443	HTTPS Post
10	Send HTTPS POST XML	0	httpbin.org	443	HTTPS Post
11	Send HTTP Search DuckDuckGo	0	html.duckduckgo.com	443	HTTPS Post
12	Send IBM DNS over HTTPS / DoH	0	dns.quad9.net	5053	HTTPS Get
13	Send NTP query	0	pool.ntp.org	123	UDP

Clear Log (1)

Time	From IP	From Port	To Address	To Port	Method	Error	ASCII	Hex
2025-05-26 13:20:11.440	172.18.27.109	9000	224.100.100.100	9000	UDP	Good!Ho!!!!!!	47 6f 6f 64 4a 69 48 6f 21 21 21 21 21	

IP Multicast Setup (Experimental)

IP Multicast Address: 224.100.100.100  
UDP socket bound to 9000 will join the multicast group

Join

Leave All Groups

DTLS Server Disabled UDP:9000 TCP Server Disabled

Copyright (C) Microsoft Corporation. All rights reserved.  
새로운 기능 및 개선 사항에 대한 최신 PowerShell을 설치하세요! https://aka.ms/PSWindows  
PS C:\Windows\system32> ipconfig  
Windows IP 구성  
  
이더넷 어댑터 이더넷 2:  
  
연결별 DNS 접미사. . . : yuhan.ac.kr  
링크-로컬 IPv6 주소. . . : fe80::285e:1b1c:8073:dd9a%6  
IPv4 주소. . . : 172.18.27.107  
서브넷 마스크. . . : 255.255.255.0  
기본 게이트웨이. . . : 172.18.27.254  
PS C:\Windows\system32>

4. 멀티캐스트 그룹에 참여한 또 다른 컴퓨터

### 3. UDP/TCP 포트번호 동일하게 하고 Packet Sender 통신

The image displays four screenshots related to a C# UDP/TCP communication project:

- Left Screenshot:** Shows the C# code for a client program. Key elements include the `SERVERIP` and `SERVERPORT` constants, and the `main` method where the client connects to the server. A red circle highlights the `SERVERPORT` value.
- Second Screenshot:** Shows the Packet Sender application interface. The 'Name' field is set to 'udpclient', and the 'Port' is set to 9000. A red circle highlights the 'Port' field.
- Third Screenshot:** Shows the C# code for the server program. Key elements include the `SERVERIP` and `SERVERPORT` constants, and the `main` method where the server listens for incoming connections. A red circle highlights the `SERVERPORT` value.
- Right Screenshot:** Shows the Wireshark network traffic capture. It displays a list of captured packets, with a red circle highlighting a specific packet (No. 1) and a green circle highlighting another packet (No. 2).

1. UDP, TCP 모두 9000번으로 포트 번호를 설정

2. 포트번호가 같아도 각자 다른 전송 프로토콜이기에 문제없이 통신 가능

## 4. C언어에 없는 C# 연산자들 예제

```
using System;

namespace SerPro10Weeks
{
    //202127064 이지호
    참조 3개
    class Dog
    {
        public string dogName = "포메라니안";
    }
    참조 0개
    internal class Program
    {
        참조 0개
        static void Main(string[] args)
        {
            int Num = 30123;
            if (Num is int) Console.WriteLine("Num은 정수형입니다!");
            else Console.WriteLine("Num은 문자형입니다!");

            double? dnum = Num as double?;
            if (dnum != null) Console.WriteLine("실수형으로 변환 성공");
            else Console.WriteLine("변환에 실패하여 Null 반환");

            Dog dog = new Dog();
            dog = null;
            Console.WriteLine(dog?.dogName);

            string str = null;
            //str = "하하";
            string res = str ?? "str은 널!";
            Console.WriteLine(res);

            int? a = null;
            a ??= 123;
            Console.WriteLine(a);

            Func<float, float> rect = x => x * x;
            Console.WriteLine(rect(3));

            Console.WriteLine(nameof(Dog));

            global::System.Console.WriteLine("이거슨 전역 네임스페이스 지정자이와요~");
        }
    }
}
```

Microsoft Visual Studio 디버깅

Num은 정수형입니다!  
변환에 실패하여 Null 반환

str은 널!  
123  
9  
Dog  
이거슨 전역 네임스페이스 지정자이와요~

C:\Users\user\Desktop\ServerProgrammingTest\SerPro10Weeks\bin\Debug\net8.0\SerPro10Weeks.exe(프로세스 30884)이(가) 0 코드(0x0)와 함께 종료되었습니다.  
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지되면 자동으로 콘솔 닫기]를 사용하도록 설정합니다.  
이 창을 닫으려면 아무 키나 두르세요...

# 5-1. TCP 버전

```
mespace tcpServer
C:\Users\User\Desktop>
2025-05-24 16:44 : 서버 소켓 생성
2025-05-24 16:44 : 서버 대기 시작
```

1. TCP 서버 프로그램을 실행하면 클라이언트 프로그램이 접속할 때까지 대기 상태에 진입

```
mespace tcpClient
C:\Users\User\Desktop>
2025-05-24 16:46 : Client 소켓 생성
2025-05-24 16:46 : 서버 (127.0.0.1) 접속 시도
2025-05-24 16:46 : 서버 최종 확인
2025-05-24 16:46 : 서버로부터 받은 메시지: 환영- 이치호의 TCP서버에 오신 것을 환영하네!
2025-05-24 16:46 : 데이터 전송 완료
2025-05-24 16:46 : 데이터 받음 완료: 6바이트
[보낸 데이터]
[받은 데이터]
```

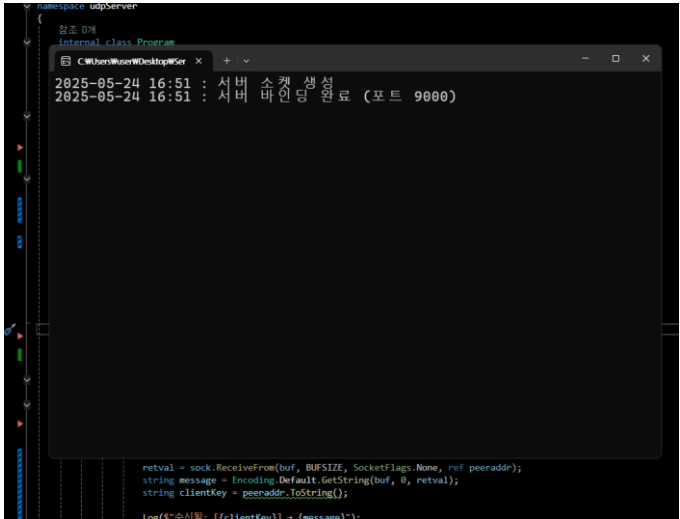
2. TCP 클라이언트 프로그램을 통해 서버에 접속하면 서버로부터 환영 메시지를 받은 뒤 서버와 통신할 수 있음.

```
mespace tcpServer
C:\Users\User\Desktop>
2025-05-24 16:44 : 서버 소켓 생성
2025-05-24 16:44 : 서버 대기 시작
2025-05-24 16:46 : Client 소켓 생성
2025-05-24 16:46 : 서버 (127.0.0.1) 접속 시도
2025-05-24 16:46 : 서버 최종 확인
2025-05-24 16:46 : 서버로부터 받은 메시지: 환영- 이치호의 TCP서버에 오신 것을 환영하네!
2025-05-24 16:47 : 데이터 전송 완료: 6바이트
2025-05-24 16:47 : 데이터 받음 완료: 6바이트
[보낸 데이터]
[받은 데이터]
```

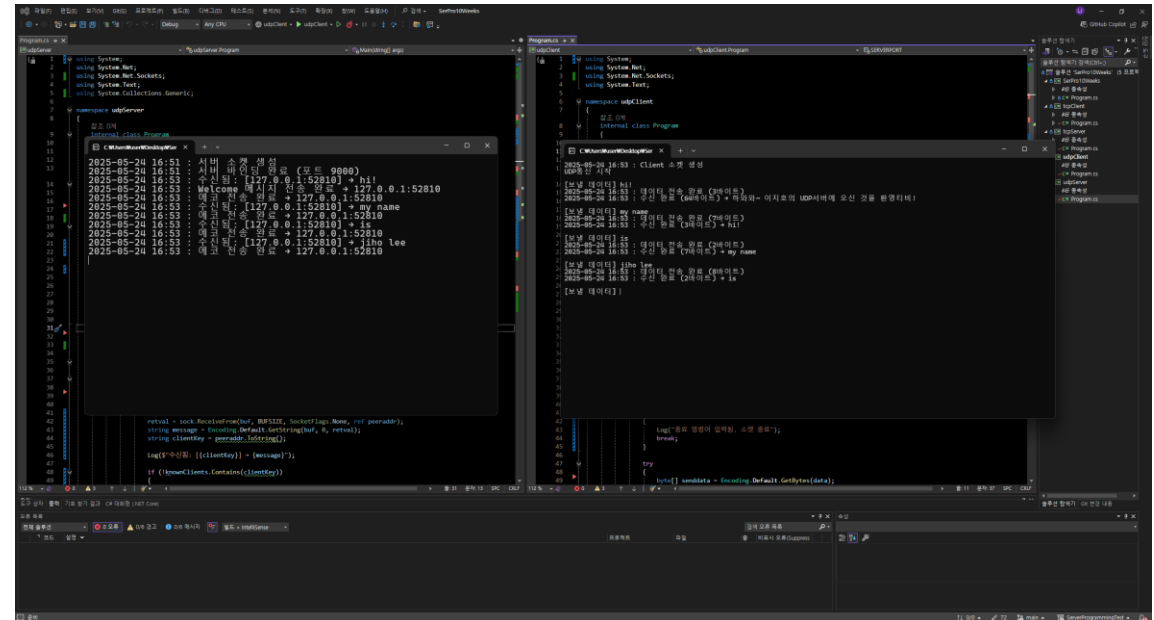
3. quit 또는 exit를 통해 클라이언트에서 서버로의 접속을 끊을 수 있음.



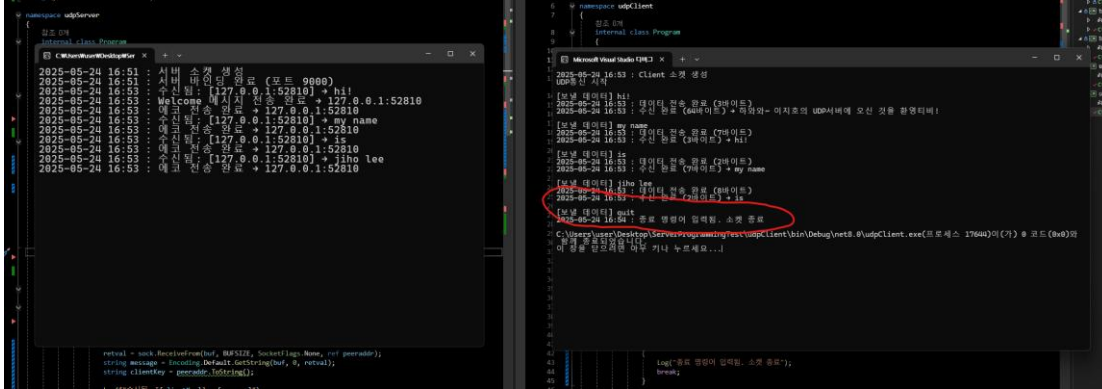
# 5-1. UDP 버전



1. UDP 서버 프로그램을 실행하면 클라이언트 프로그램이 접속할 때까지 대기 상태에 진입



2. UDP 클라이언트 프로그램을 통해 서버에 접속하면 서버로부터 환영 메시지를 받은 뒤 서버와 통신할 수 있음.



3. quit 또는 exit를 통해 클라이언트에서 서버로의 접속을 끊을 수 있음.