

컴퓨터공학 All in One

C/C++ 문법, 자료구조 및 심화 프로젝트 (나동빈)
제 64강 - C++ Boost.Asio 동기식 TCP 통신 예제

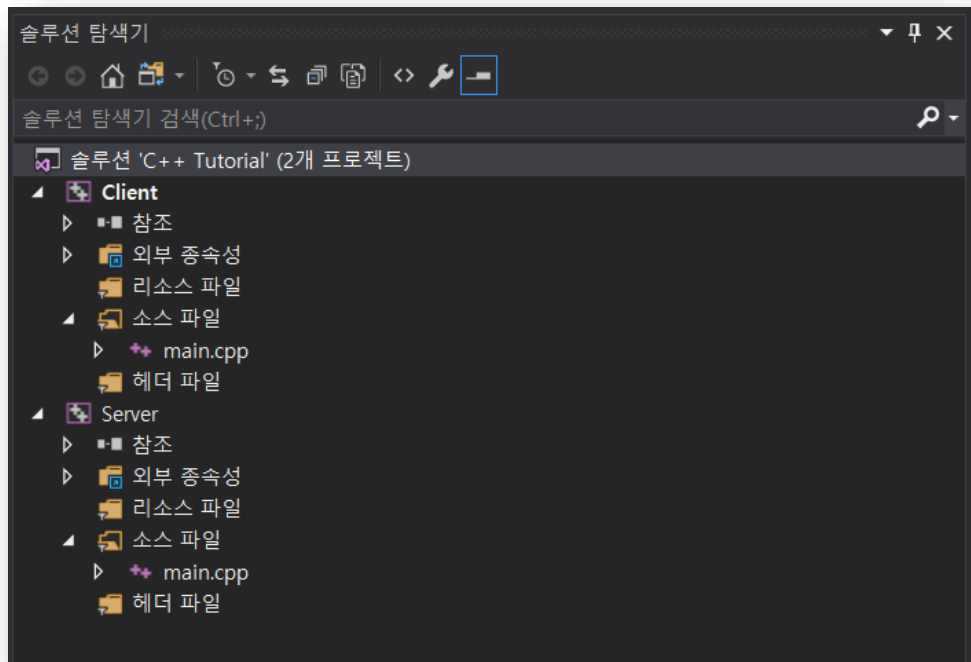
학습 목표

C++ Boost.Asio 동기식 TCP 통신 예제

- 1) Boost.Asio의 대표적인 소켓 프로그래밍 예제인 동기식 TCP 통신 예제를 구현할 수 있습니다.

C++ Boost.Asio 동기식 TCP 통신 예제

서버 & 클라이언트 구조 갖추기



C++ Boost.Asio 동기식 TCP 통신 예제

서버 프로그램 ①

```
#define _CRT_SECURE_NO_WARNINGS
#include <ctime>
#include <iostream>
#include <string>
#include <boost/asio.hpp>

using boost::asio::ip::tcp;
using namespace std;

// 서버 컴퓨터의 날짜 및 시간 정보를 반환합니다.
string make_daytime_string()
{
    time_t now = time(0);
    return ctime(&now);
}
```

C++ Boost.Asio 동기식 TCP 통신 예제

서버 프로그램 ②

```
int main()
{
    try {
        // 기본적으로 Boost Asio 프로그램은 하나의 IO Service 객체를 가집니다.
        boost::asio::io_service io_service;
        // TCP 프로토콜의 13번 포트로 연결을 받는 수동 소켓을 생성합니다.
        tcp::acceptor acceptor(io_service, tcp::endpoint(tcp::v4(), 13));
        // 모든 클라이언트에 대해 무한정 반복 수행합니다.
        while(1)
        {
            // 소켓 객체를 생성해 연결을 기다립니다.
            tcp::socket socket(io_service);
            acceptor.accept(socket);
            // 연결이 완료되면 해당 클라이언트에게 보낼 메시지를 생성합니다.
            string message = make_daytime_string();
            // 해당 클라이언트에게 메시지를 담아 전송합니다.
            boost::system::error_code ignored_error;
            boost::asio::write(socket, boost::asio::buffer(message), ignored_error);
        }
    } catch (exception& e) {
        cerr << e.what() << '\n';
    }
    return 0;
}
```

C++ Boost.Asio 동기식 TCP 통신 예제

클라이언트 프로그램 ①

```
#include <iostream>
#include <boost/array.hpp>
#include <boost/asio.hpp>

using boost::asio::ip::tcp;
using namespace std;
```

C++ Boost.Asio 동기식 TCP 통신 예제

클라이언트 프로그램 ②

```
int main()
{
    try
    {
        // 기본적으로 Boost Asio 프로그램은 하나의 IO Service 객체를 가집니다.
        boost::asio::io_service io_service;
        // 도메인 이름을 TCP 종단점으로 바꾸기 위해 Resolver를 사용합니다.
        tcp::resolver resolver(io_service);
        // 서버로는 로컬 서버, 서비스는 Daytime 프로토콜을 적어줍니다.
        tcp::resolver::query query("localhost", "daytime");
        // DNS를 거쳐 IP 주소 및 포트 번호를 얻어옵니다.
        tcp::resolver::iterator endpoint_iterator = resolver.resolve(query);
        // 소켓 객체를 초기화하여 서버에 연결합니다.
        tcp::socket socket(io_service);
        boost::asio::connect(socket, endpoint_iterator);
    }
}
```

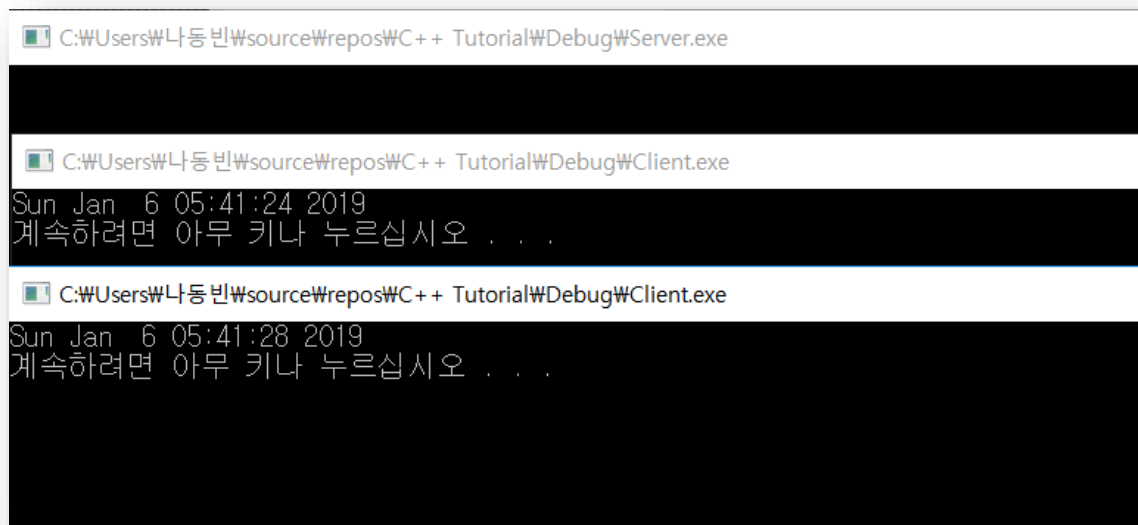
C++ Boost.Asio 동기식 TCP 통신 예제

클라이언트 프로그램 ③

```
while(1) {  
    // 버퍼 및 오류 처리 변수를 선언합니다.  
    boost::array<char, 128> buf;  
    boost::system::error_code error;  
    // 버퍼를 이용해 서버로부터 데이터를 받아옵니다.  
    size_t len = socket.read_some(boost::asio::buffer(buf), error);  
    if (error == boost::asio::error::eof)  
        break;  
    else if (error)  
        throw boost::system::system_error(error);  
    // 버퍼에 담긴 데이터를 화면에 출력합니다.  
    cout.write(buf.data(), len);  
}  
} catch (exception& e) {  
    cerr << e.what() << endl;  
}  
system("pause");  
return 0;  
}
```


C++ Boost.Asio 동기식 TCP 통신 예제

솔루션 빌드 및 실행



```
C:\Users\나동빈\source\repos\C++ Tutorial\Debug\Server.exe

C:\Users\나동빈\source\repos\C++ Tutorial\Debug\Client.exe
Sun Jan 6 05:41:24 2019
계속하려면 아무 키나 누르십시오 . . .

C:\Users\나동빈\source\repos\C++ Tutorial\Debug\Client.exe
Sun Jan 6 05:41:28 2019
계속하려면 아무 키나 누르십시오 . . .
```