

개발 환경

ECE20010

한동 대학교

김영섭 교수

idebtor@gmail.com

GCC – Gnu Compiler Collection

MSYS2 – Install 1st

MinGW-w64 – install 2nd

g++ for macOS

- Open a terminal. (command + space, type terminal in the search field.)
- Enter the following command and run it.
 - `$ xcode-select -install`
- Check the installation with the following command.
(If you see some version numbers, it was successfully installed.)
 - `$ g++ -v`
- Check the option such that all filename extensions are to be displayed.
[Finder] → [Preferences ...] → [Advanced] → Show all filename extensions

g++ for macOS

- Use this code to test that g++ can compile and run the program.

```
#include <iostream>
using namespace std;
int main() {
    cout << "Hello, World!\n";
    return 0;
}
```

1. Copy the above program and save the file as **hello.cpp** on your desktop.
2. Open a terminal and type **cd Desktop**
3. To compile, enter the following in the terminal and run it. We see the file **hello** appear on the desktop under an exec icon.

```
$ g++ -std=c++11 hello.cpp -o hello
```

4. Run the compiled program by typing the following in the terminal. You will see "Hello World!" printed in the terminal.

```
$ ./hello
```

MSYS2 우선 설치, MinGW-w64 for Windows

- 두 종류의 패키지, 즉 **MSYS2** 와 **MinGW-w64** 를 설치해야 합니다.
MSYS2를 **먼저** 설치하지 않으면 어려움을 겪는 경우가 종종 발생합니다.

MSYS2 우선 설치하기

- <http://www.MSYS2.org/>로 접속합니다.

MSYS2를 통해 MinGW-w64를 설치를 위해
아래 website를 참조해도 좋습니다.

- <https://vl0011.tistory.com/14>
- <https://stackoverflow.com/questions/30069830/how-to-install-mingw-w64-and-msys2/30071634#30071634>

MSYS2

Software Distribution and Building Platform for Windows

MSYS2 is a collection of tools and libraries providing you with an easy-to-use environment for building running native Windows software.

It consists of a command line terminal called `mintty`, `bash`, version control systems like `git` and `subversion`, and `awk` and even build systems like `autotools`, all based on a modified version of `Cygwin`. Despite some central parts being based on `Cygwin`, the main focus of MSYS2 is to provide a build environment for native software and the `Cygwin`-using parts are kept at a minimum. MSYS2 provides up-to-date native builds for `w64`, `CPython`, `CMake`, `Meson`, `OpenSSL`, `FFmpeg`, `Rust`, `Ruby`, just to name a few.

To provide easy installation of packages and a way to keep them updated it features a package manager called `Pacman`, which should be familiar to Arch Linux users. It brings many powerful features such as resolution and simple complete system upgrades, as well as straight-forward and reproducible package management. The package repository contains [more than 2900 pre-built packages](#) ready to install.

For more details see '[What is MSYS2?](#)' which also compares MSYS2 to other software distributions and environments like `Cygwin`, `WSL`, `Chocolatey`, `Scoop`, ... and '[Who Is Using MSYS2?](#)' to see which project uses MSYS2 and what for.

Installation

1. Download the installer

[msys2-x86_64-20230718.exe](#)

클릭

For more information on the installer, like command line options, or how to verify the checksum and the installer, see the [installer guide](#).

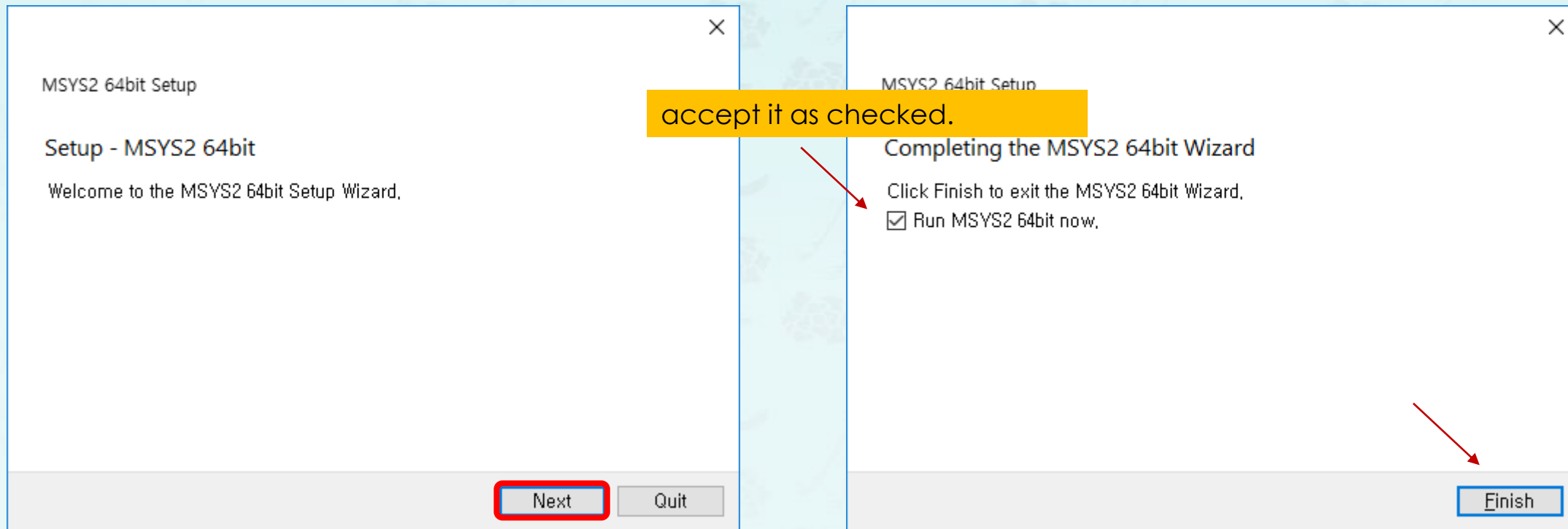
2. Run the installer. MSYS2 requires 64 bit Windows 8.1 or newer.

3. Enter your desired **Installation Folder** (short ASCII-only path on a NTFS volume, no accents, no spaces, no symlinks, no subst or network drives, no FAT).

MSYS2 우선 설치, MinGW-w64 for Windows

MSYS2 설치하기

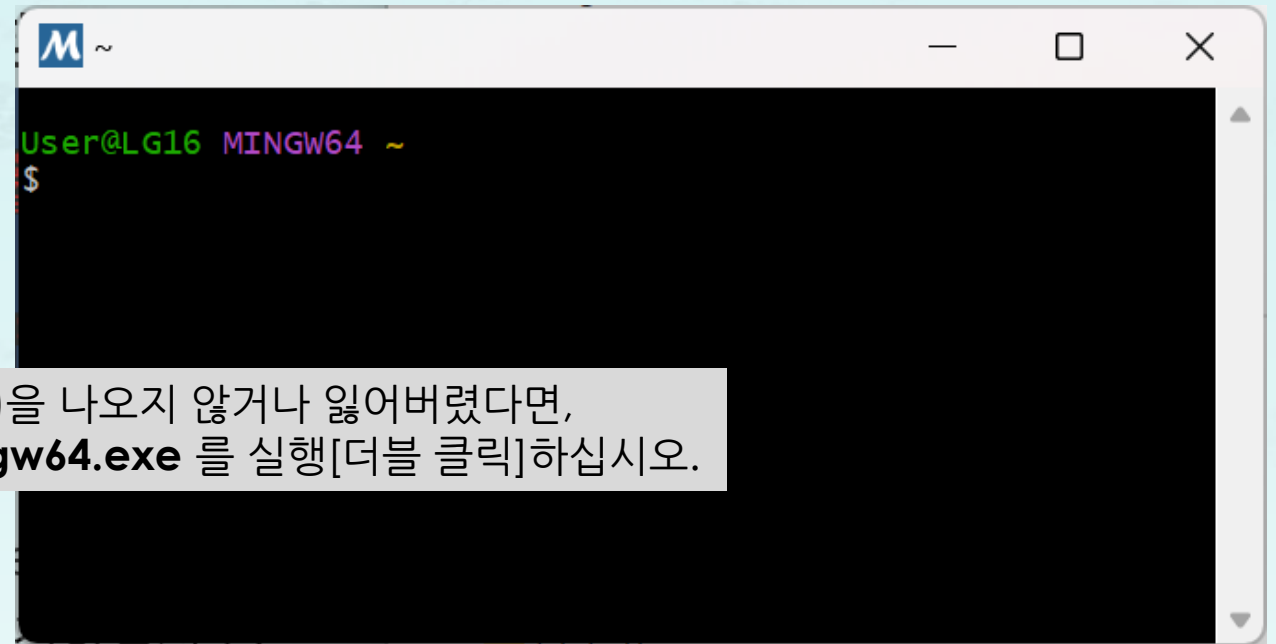
- MSYS2-x86_64-20230718.exe를 실행합니다. **혹은 최신 버전으로.**
- Finish를 클릭하여 MSYS2를 실행합니다.



MSYS2 우선 설치, MinGW-w64 for Windows

MSYS2 설치하기

- MSYS2를 실행한 화면 입니다.



- Command Shell (창) 혹은 PowerShell에서 “**pacman -Syu**”를 입력하고 Enter 키를 눌러 패키지 목록과 MSYS2를 업데이트 합니다.
- “::설치를 진행하시겠습니까? **[Y/n]**”라는 질문이 나오면, **Y** 혹은 **[Enter]**를 입력합니다.
- 실행 결과를 보면 “경고:terminate MSYS2 without returning....”, “경고: for example close your terminal window...”이라는 메시지가 나오면 경고를 무시하고 다음을 진행합니다.

MSYS2 우선 설치, MinGW-w64 for Windows

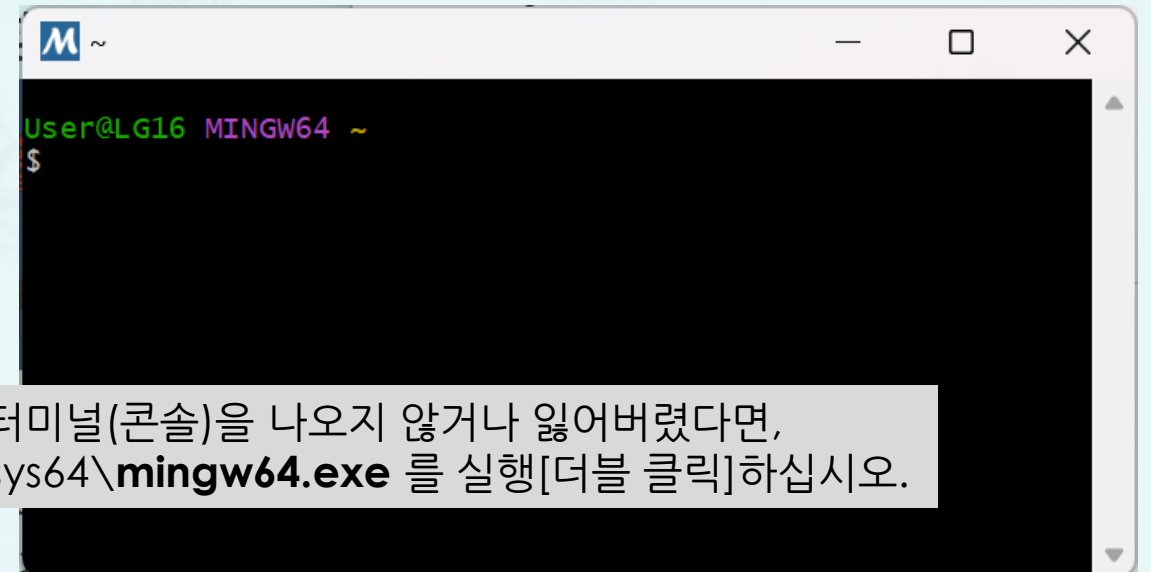
- 아래와 같은 “pacman -Syu” 실행 문제점 해결 방법

```
PS C:\Users\진형> pacman -Syu
pacman : 'pacman' 용어가 cmdlet, 함수, 스크립트 파일 또는 실행할 수 있는 프로그램 이름으로 인식되지 않습니다. 이름이 정
확한지 확인하고 경로가 포함된 경우 경로가 올바른지 검증한 다음 다시 시도하십시오.
위치 줄:1 문자:1
+ pacman -Syu
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (pacman:String) [], CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException
```

- **C:\msys64\usr\bin** (MSYS2 설치 폴더)에 packman 파일 존재 확인 하십시오
 - 존재한다면, 해당 폴더에서 powershell을 열고, **.\pacman -Syu** 를 실행하십시오.
 - 존재하지 않는다면, MSYS2를 다시 설치하십시오.

혹은

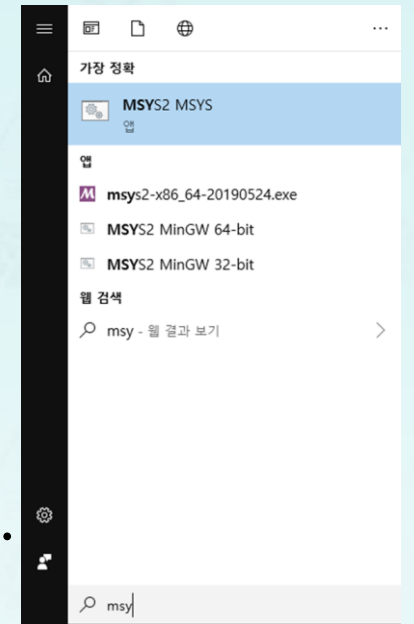
- **C:\msys64\mingw64.exe**를 실행하여 나타난 터미널에서 실행해보십시오.



MSYS2 우선 설치, MinGW-w64 for Windows

GCC(GNU Compiler Collection) 설치하기

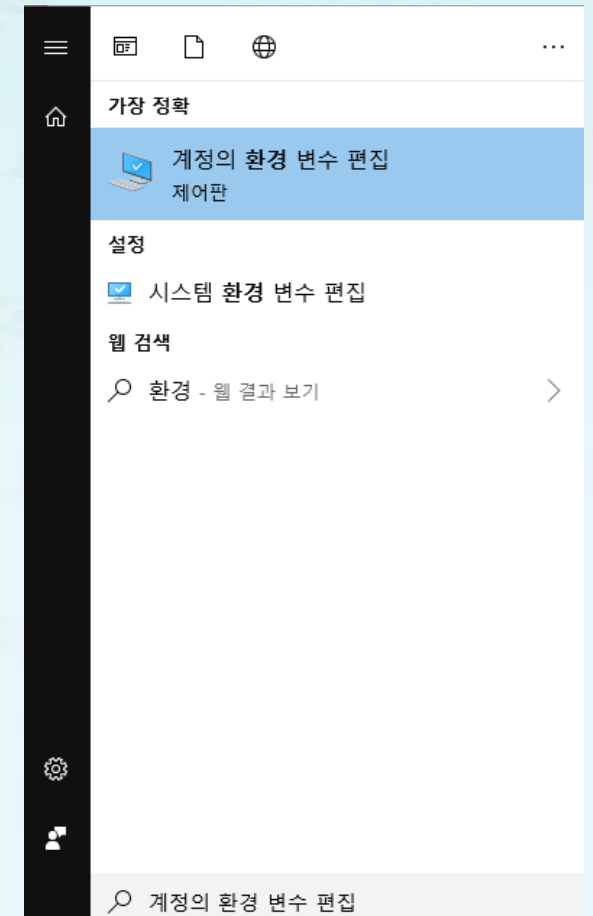
- MSYS2를 실행한 화면(콘솔)에서 다음을 입력하여 GCC를 설치하십시오.
pacman -S mingw-w64-x86_64-gcc
 - 만약, MSYS2 실행 화면(콘솔)이 없어졌으면,
MSYS2 설치된 곳에서, 예를 들면, **C:\msys64\mingw64.exe**를
실행하여 터미널(콘솔)을 다시 생성하고 명령어를 입력 하십시오.
- 설치할 때 질문이 나오면, Y 혹은 [Enter]로 응답하면 됩니다.
- Toolchain 설치 - make, gdb 등 gcc와 함께 쓰이는 툴들을 설치합니다.
pacman -S mingw-w64-x86_64-toolchain
설치할 때 질문이 나오면, <Enter>로 응답하면 됩니다.
- 하나씩 개별적으로 설치할 경우, 예를 들면, 다음과 같이 설치합니다.
pacman -S make



MSYS2 우선 설치, MinGW-w64 for Windows

환경변수 설정

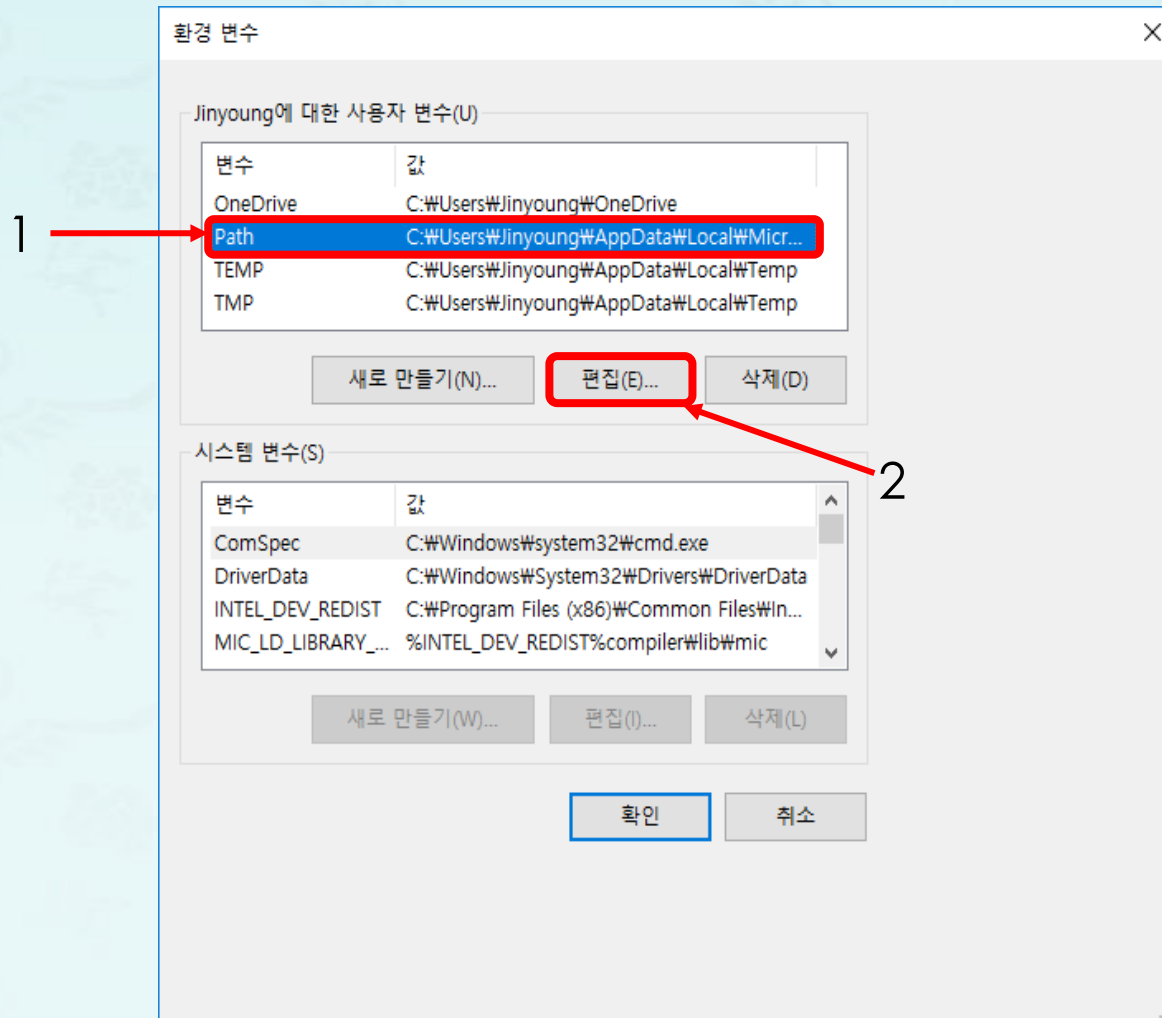
- 다음은 시스템 환경 변수를 설정해야 합니다.
- 환경변수를 설정하기 위해 윈도우 검색에서 “환경 변수”를 입력하고 계정의 환경 변수 편집으로 들어갑니다.



MSYS2 우선 설치, MinGW-w64 for Windows

환경변수 설정

- 사용자 변수 창 혹은 시스템 변수 창에서 Path를 선택한 후 편집을 클릭합니다.



MSYS2 우선 설치, MinGW-w64 for Windows

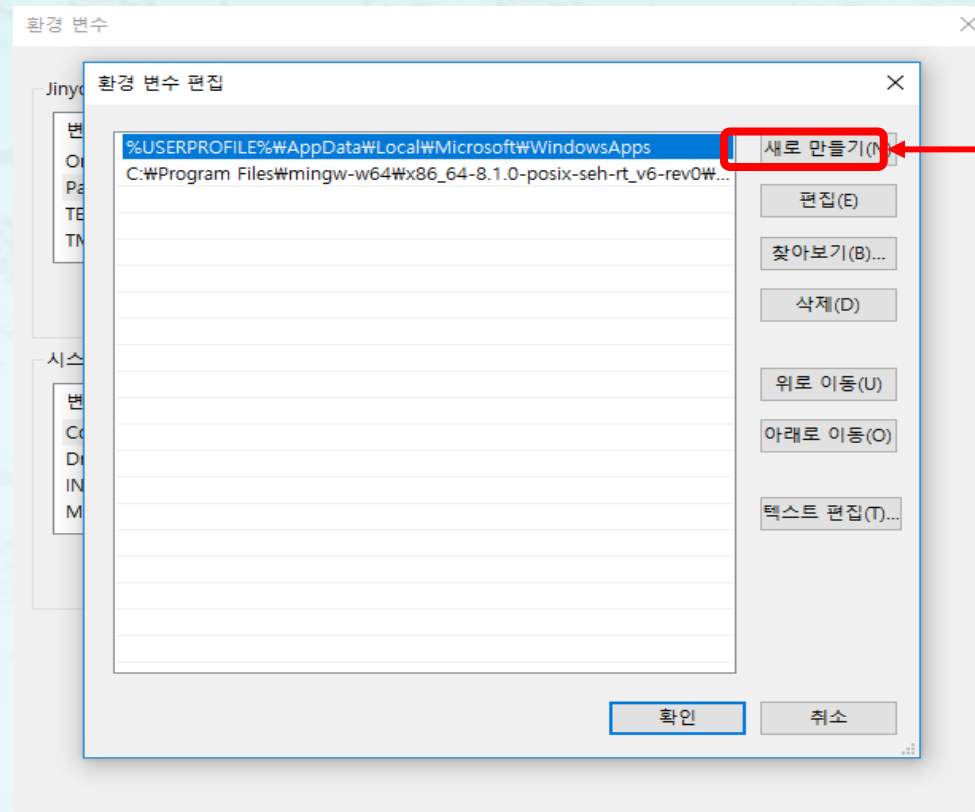
환경변수 설정

- [새로 만들기]를 클릭 후 g++.exe 가 존재하는 path를 입력하거나 [붙여넣기] 합니다.
 - msys2와 mingw64를 설치한 곳에서, **g++.exe** 가 존재하는 폴더를 찾아서, 그 **Path**를 복사하여 붙여 넣기를 하는 것이 안전합니다.
 - 만약, msys2 설치 시에 디폴트를 사용했다면, 다음과 같지만, 한번 확인해 보는 것이 좋습니다

C:\msys64\mingw64\bin

C:\msys64\usr\bin

위의 두번째 폴더는 ls, cp, rm 같은 Bash 명령어를 사용하기 위해 필요한 폴더입니다.

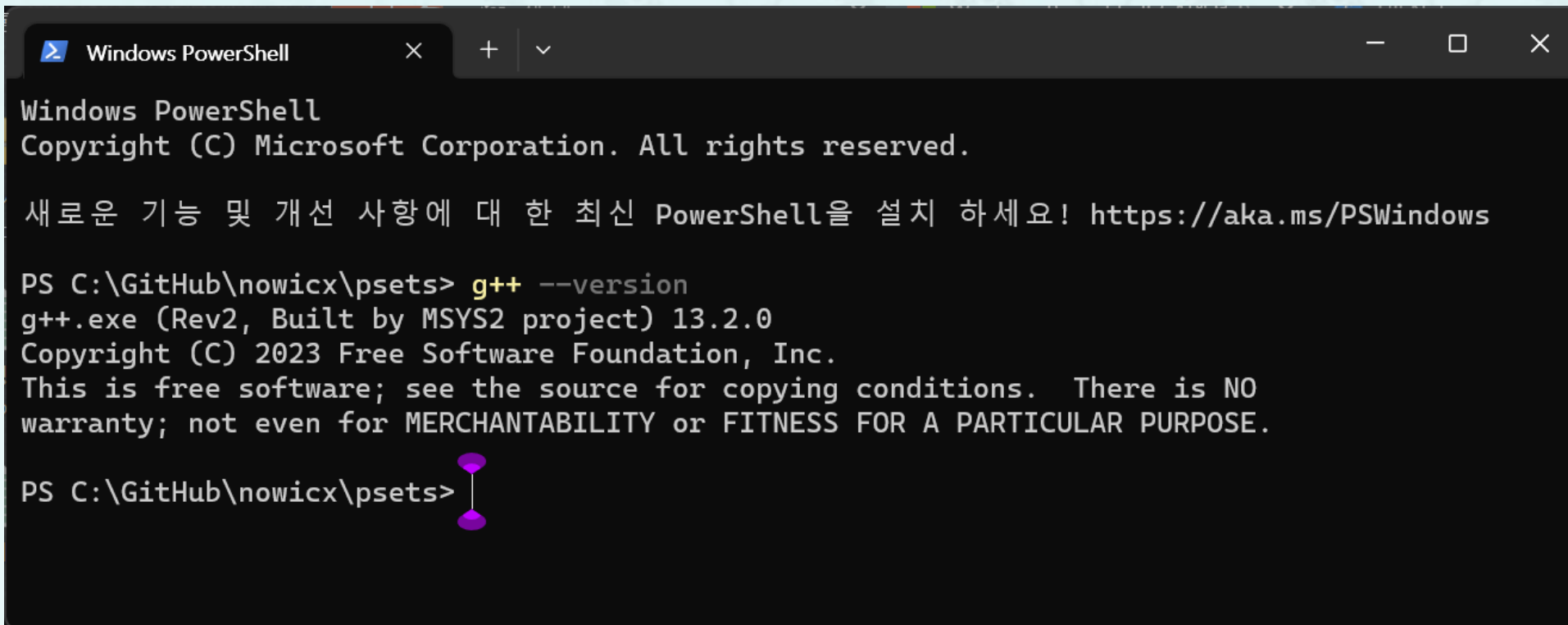


클릭 후 path 입력

MSYS2 우선 설치, MinGW-w64 for Windows

환경변수 설정

- 윈도우 검색창에 “**cmd**” or “powershell”을 실행하여, 터미널(콘솔)을 생성하십시오.
 - 혹은 **C:\msys64\mingw64.exe**를 실행하여 터미널(콘솔)을 생성하십시오.
- 프롬프트 창에 “**g++ --version**”을 입력하여 다음과 같이 화면이 나온다면 path추가가 정상적으로 된 것 입니다.

A screenshot of a Windows PowerShell terminal window. The title bar says 'Windows PowerShell'. The terminal text shows the command 'g++ --version' being executed, resulting in 'g++.exe (Rev2, Built by MSYS2 project) 13.2.0'. It also displays copyright information for the Free Software Foundation, Inc. and a disclaimer. The prompt 'PS C:\GitHub\nowicx\psets>' is visible at the bottom with a cursor.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

새로운 기능 및 개선 사항에 대한 최신 PowerShell을 설치 하세요! https://aka.ms/PSWindows

PS C:\GitHub\nowicx\psets> g++ --version
g++.exe (Rev2, Built by MSYS2 project) 13.2.0
Copyright (C) 2023 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

PS C:\GitHub\nowicx\psets>
```

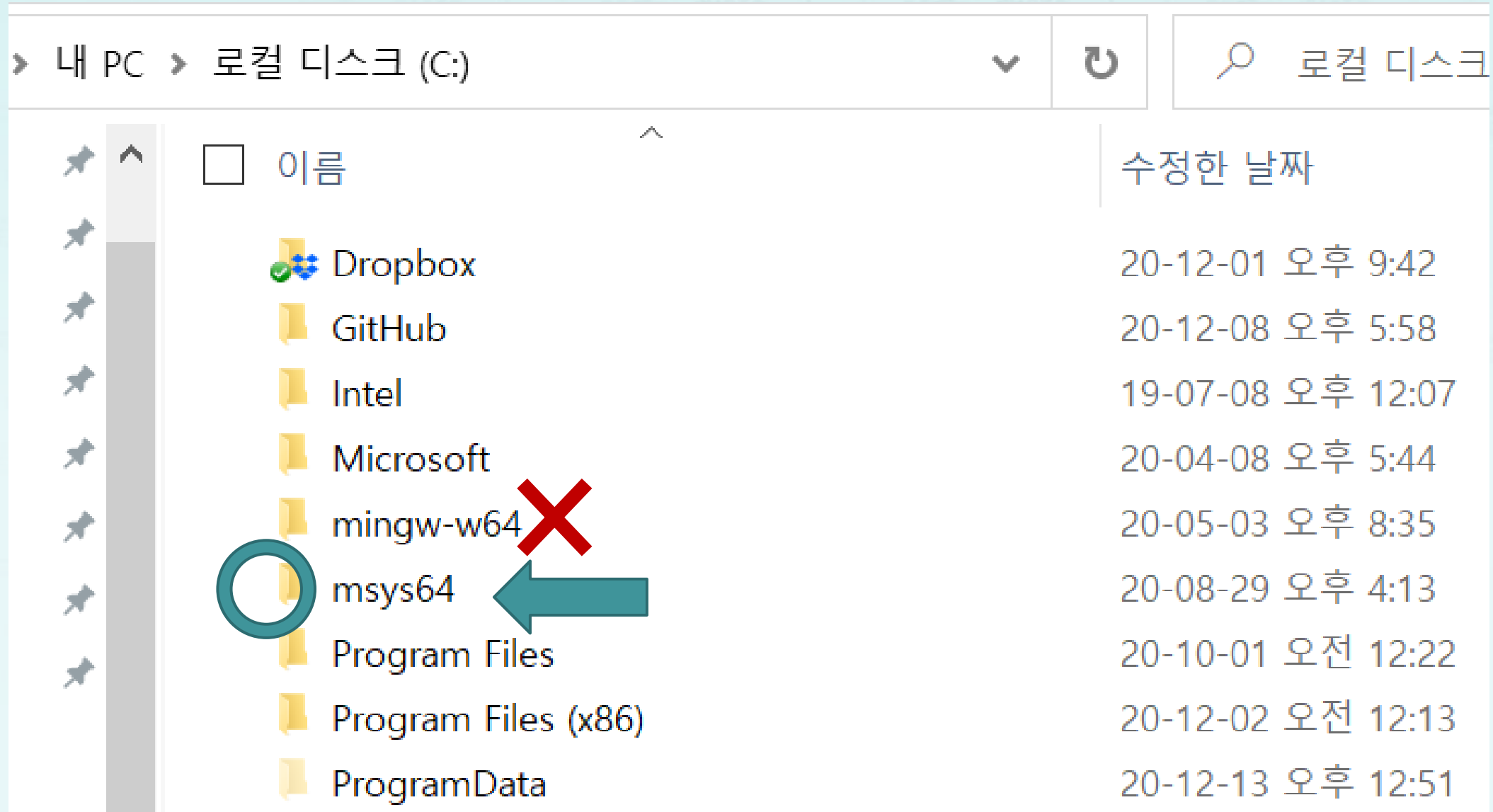
MSYS2 우선 설치, MinGW-w64 for Windows

MSYS2/MinGW-w64 설치하기 오류에 관하여...

- MSYS2가 아니라 **MinGW-w64**를 먼저 설치할 경우, 설치 오류가 발행하는 경우가 종종 있습니다.
- 또한 한글로 사용자 이름이 설정되어 있는 경우, 어려움을 많이 겪습니다. 이런 경우, 특히 블로그의 설명에 따르면 재설치하는 경우 문제가 발생한다고 합니다.
 - 이런 경우, 기존의 있는 폴더를 완전히 삭제한 후, MSYS2로 다시 설치해보십시오.
 - **MSYS2를 먼저 설치하고**, 그 안에서 MinGW-w64를 추가 설치하는 방법을 택하길 적극 권장합니다.
 - 그래도, 설치에 어려움이 있다면, (특히 한글 사용자 이름으로 인하여), 새로운 사용자를 만들어 시도해보십시오.
- <https://taeguu.tistory.com/3>

MSYS2 우선 설치, MinGW-w64 for Windows

Check C:/ folder and make sure that you Installed **msys64** first, not **mingw-w64**.



MSYS2/MinGW-w64 설치 테스트

- Use this code to test that g++ can compile and run the program.

```
#include <iostream>
using namespace std;
int main() {
    cout << "Hello, World!\n";
    return 0;
}
```

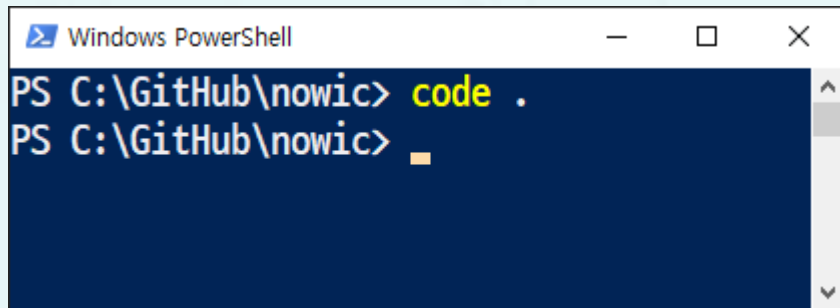
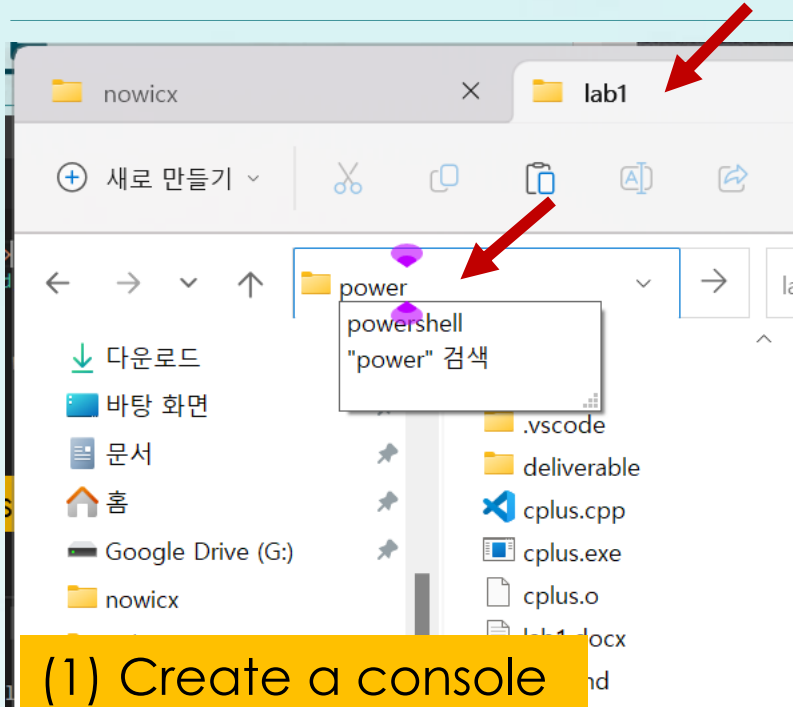
1. Copy the above program and save the file as **hello.cpp** on your folder.
2. Open a terminal at the folder where **hello.cpp** exists.
Use the [File] → [Windows PowerShell 열기] menu at the top left corner of the **File Explorer**.
3. To compile, enter the following in the terminal and run it. We see the file **hello** appear on the desktop under an exec icon.

```
$ g++ hello.cpp -o hello
```

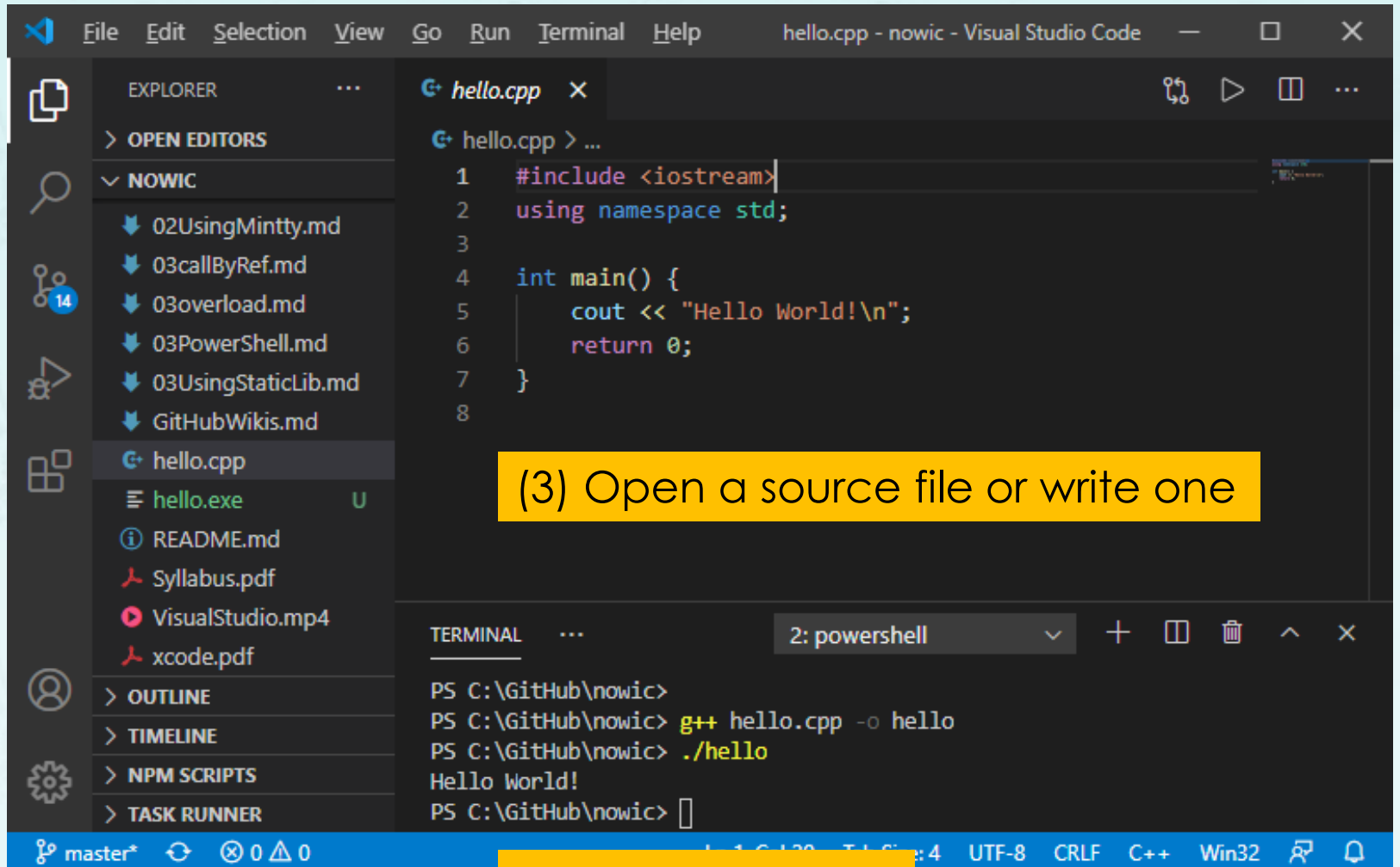
4. Run the compiled program by typing the following in the terminal. You will see "Hello World!" printed in the terminal.

```
$ ./hello
```

MSYS2/MinGW-w64 설치 테스트



(2) Start an editor or VS Code



(4) Build and run it

데이터 구조 개발환경

ECE20010

한동 대학교

김영섭 교수

idebtor@gmail.com

GCC – Gnu Compiler Collection

MSYS2 – Install 1st

MinGW-w64 – install 2nd