

# 데이터 구조 개발환경

ECE20010

한동대학교  
CSEE 자료구조  
idebtor@gmail.com

g++ for macOS & Windows  
Atom  
GitHub for Desktop  
git

# 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/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 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 installer**  
One click installer for msys2

클릭

msys2-i686-20190524.exe    msys2-x86\_64-20190524.exe    View on GitHub

SHA256: c5a1881f7ac5a0449fe9b30d31...    SHA256: 2dacadcc70cc122054e60914ct...

**MSYS2 is a software distro and building platform for Windows**

At its core is an independent rewrite of MSYS, based on modern Cygwin (POSIX compatibility layer) and MinGW-w64 with the aim of better interoperability with native Windows software. It provides a bash shell, Autotools, revision control systems and the like for building native Windows applications using MinGW-w64 toolchains.

It features a package management system to provide easy installation of packages, Pacman. It brings many powerful features such as dependency resolution and simple complete system upgrades, as well as straight-forward package building.

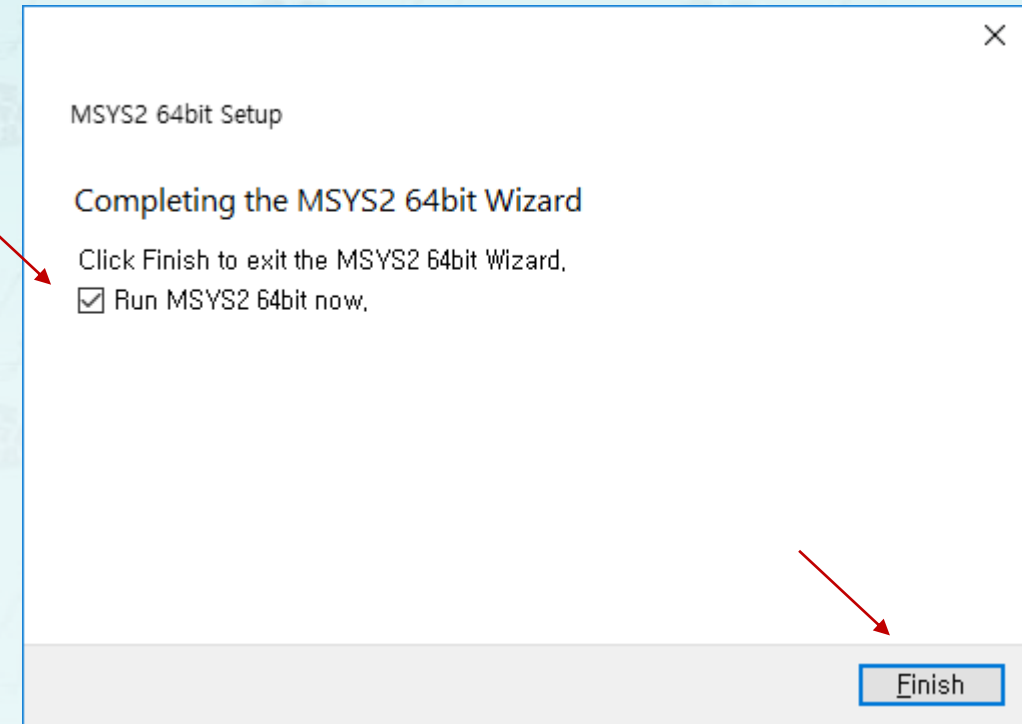
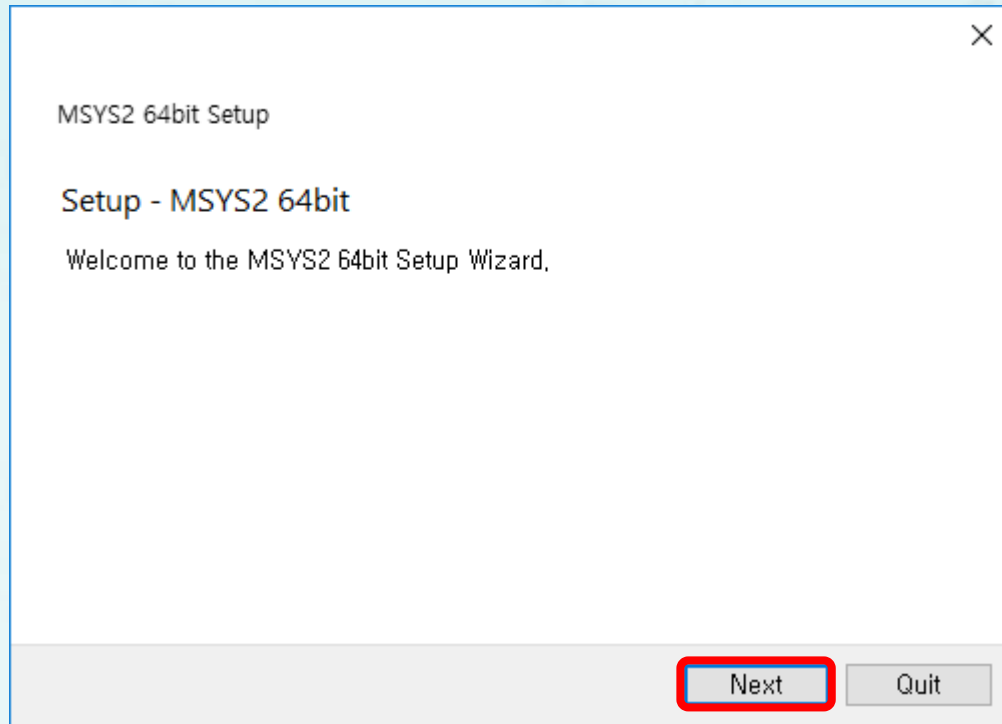
**Installation restrictions:**

- MSYS2 can't be installed on FAT\* partitions.
- Current MSYS2 can't be installed on Windows XP anymore.

# MSYS2/MinGW-w64 for Windows

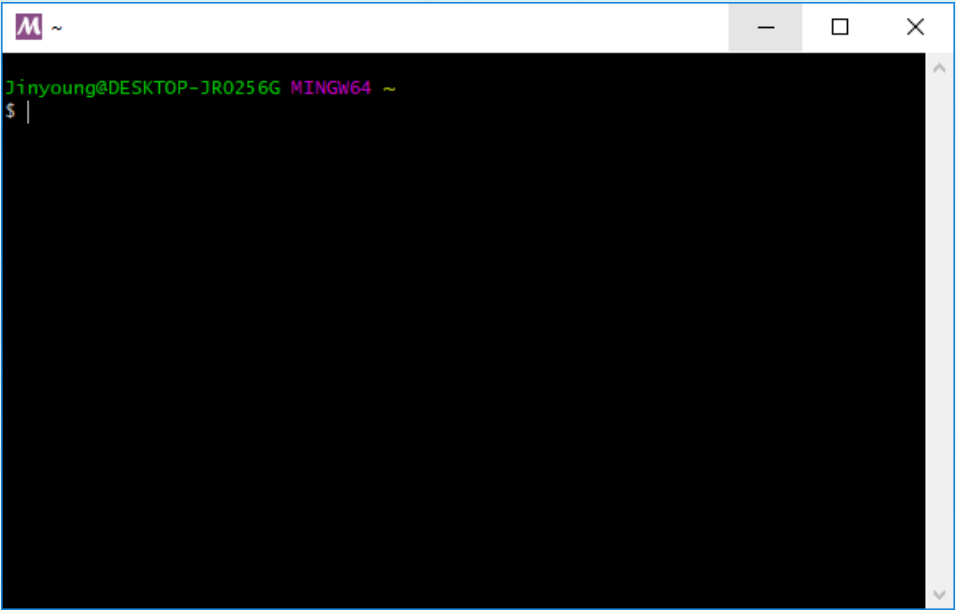
## MSYS2 설치하기

- MSYS2-x86\_64-20190524.exe를 실행합니다. 혹은 최신 버전으로.
- Finish를 클릭하여 MSYS2를 실행합니다.



# MSYS2/MinGW-w64 for Windows

## MSYS2 설치하기

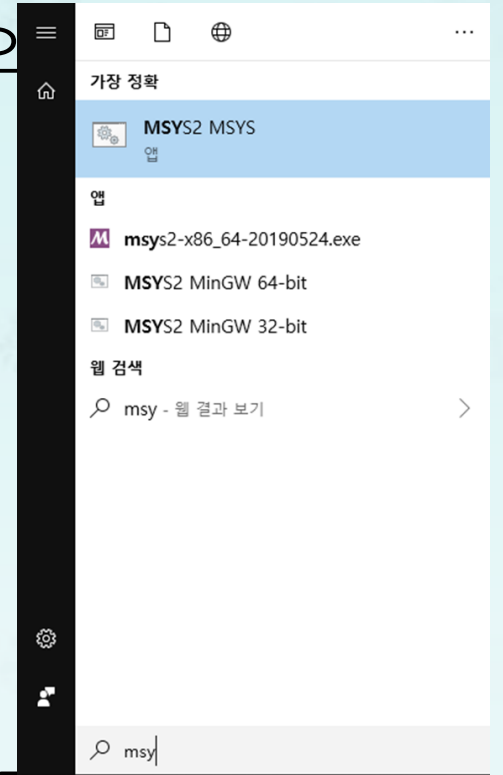
- MSYS2를 실행한 화면 입니다.
- 
- Command Shell (cmd) 혹은 PowerShell에서 “**pacman -Syu**”를 입력하고 Enter 키를 눌러 패키지 목록과 MSYS2를 업데이트 합니다.
  - “::설치를 진행하시겠습니까? [Y/N]”라는 질문이 나오면 Enter 키를 누릅니다.
  - 실행 결과를 보면 “경고:terminate MSYS2 without returning....”, “경고: for example close your terminal window...”이라는 메시지가 나오면 경고를 무시하고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



# MSYS2/MinGW-w64 for Windows

## MSYS2 설치하기 - GCC 설치하기

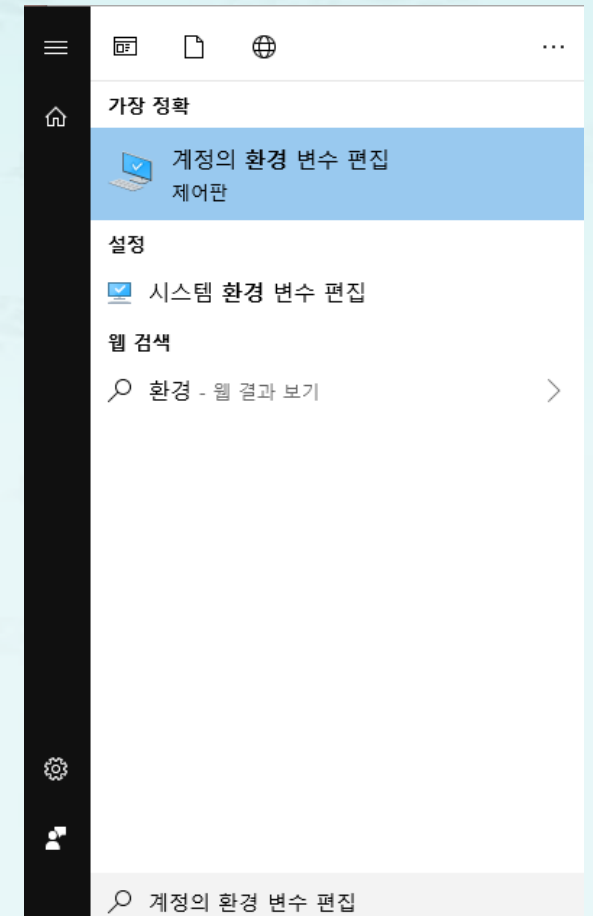
- MSYS2를 실행한 화면(콘솔)에서 다음을 입력하여 GCC를 설치하십시오  
`pacman -S mingw-w64-x86_64-gcc`
  - 만약, MSYS2 실행 화면(콘솔)이 없어졌으면, 윈도우 검색창에 “MSYS2”를 입력하거나 MSYS2를 찾아가서 MSYS2 msys를 실행하여 콘솔을 다시 생성하고 명령어를 입력 하십시오.
- 설치할 때 질문이 나오면, <Enter>로 응답하면 됩니다.
- Toolchain 설치 - make, gdb 등 gcc와 함께 쓰이는 툴들을 설치합니다.  
`pacman -S mingw-w64-x86_64-toolchain`  
설치할 때 질문이 나오면, <Enter>로 응답하면 됩니다.



# MSYS2/MinGW-w64 for Windows

## MSYS2 설치하기 - 환경변수

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

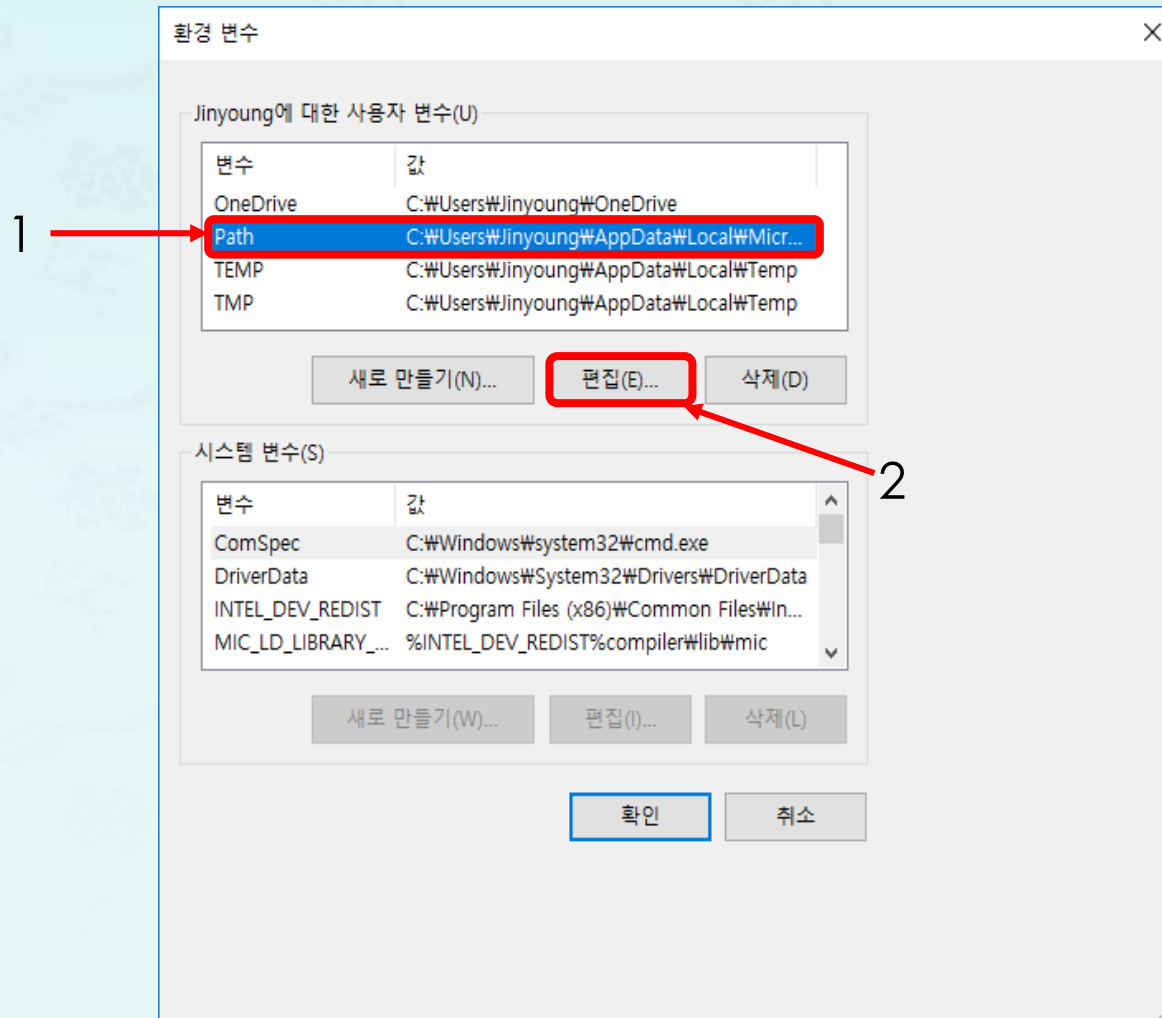




# MSYS2/MinGW-w64 for Windows

## MSYS2 설치하기 - 환경변수

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



# MSYS2/MinGW-w64 for Windows

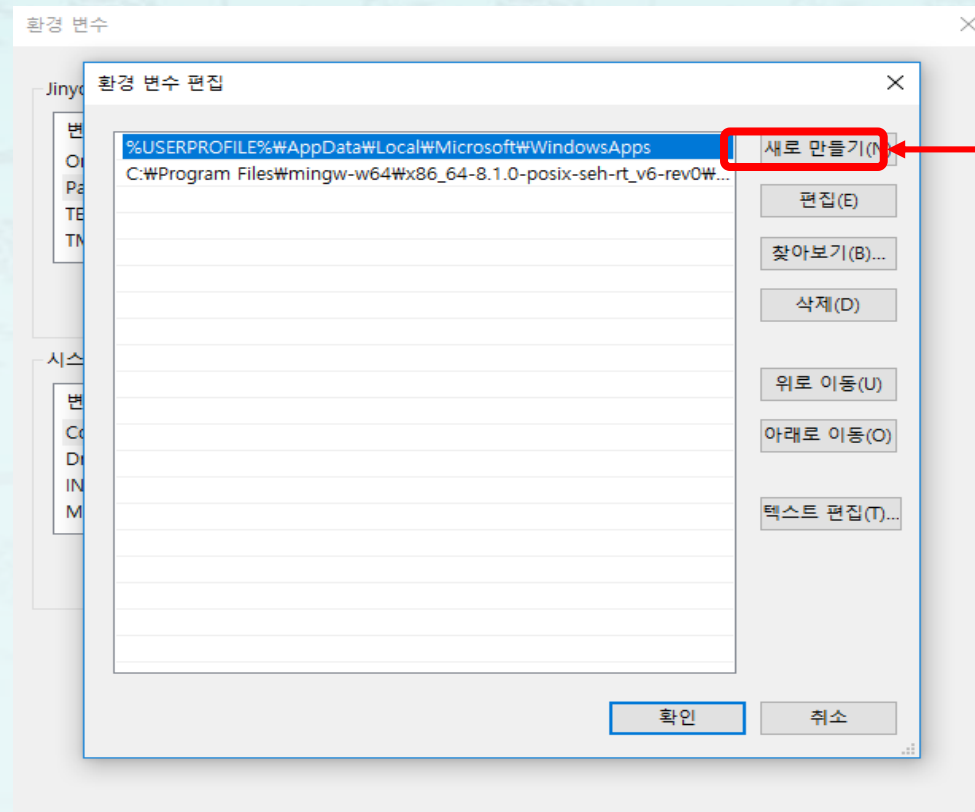
## MSYS2 설치하기 - 환경변수

- [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.

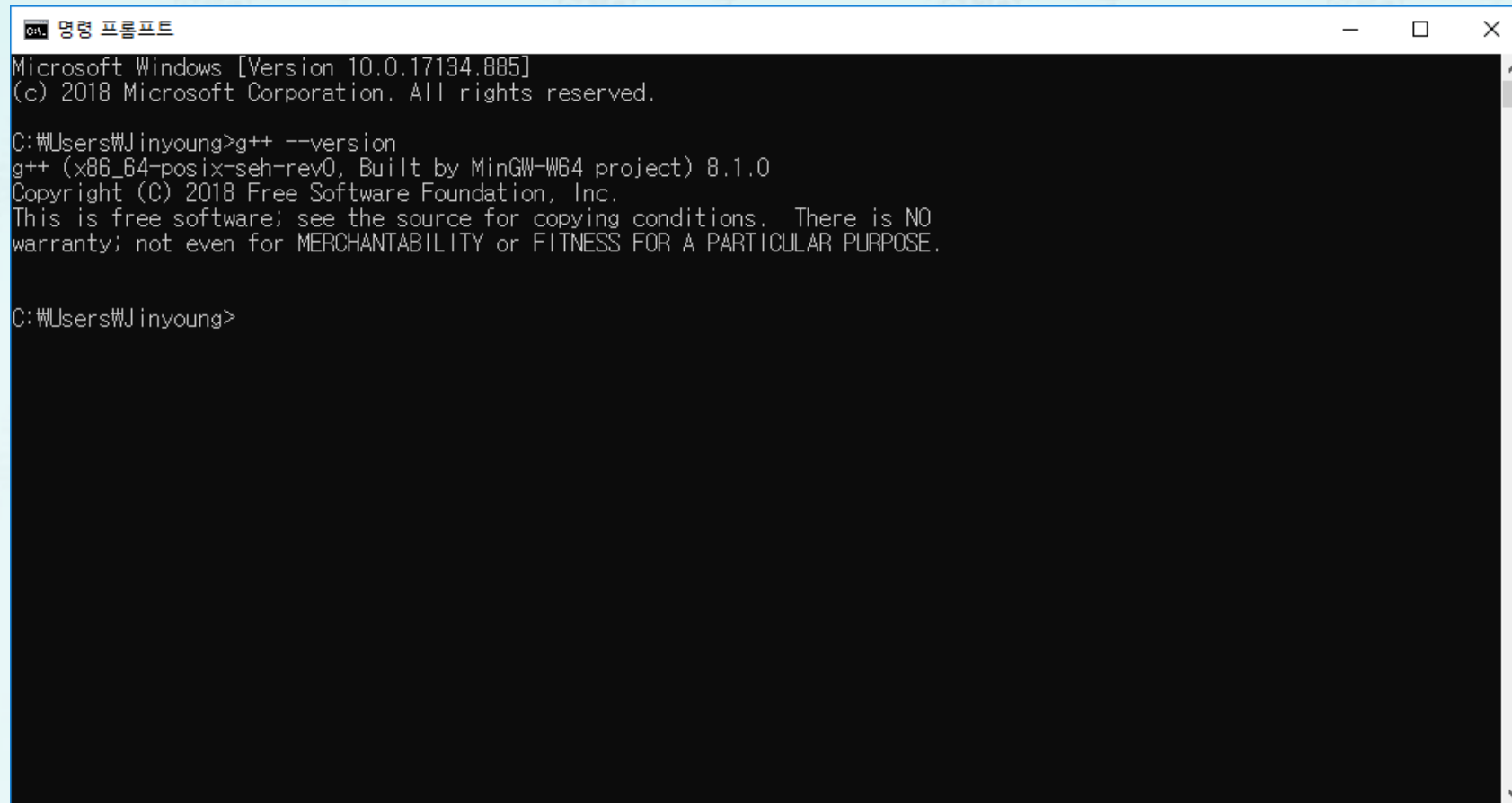


클릭 후 path 입력

# MSYS2/MinGW-w64 for Windows

## MSYS2 설치하기 - 환경변수

- 윈도우 검색창에 “**cmd**” 명령어를 실행하여, 콘솔(console or terminal)을 띄웁니다.
- 프롬프트 창에 “**g++ --version**”을 입력하여 다음과 같이 화면이 나온다면 path추가가 정상적으로 된 것 입니다.



```
CA 명령 프롬프트
Microsoft Windows [Version 10.0.17134.885]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Jinyoung>g++ --version
g++ (x86_64-posix-seh-rev0, Built by MinGW-W64 project) 8.1.0
Copyright (C) 2018 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.

C:\Users\Jinyoung>
```

# MSYS2/MinGW-w64 for Windows

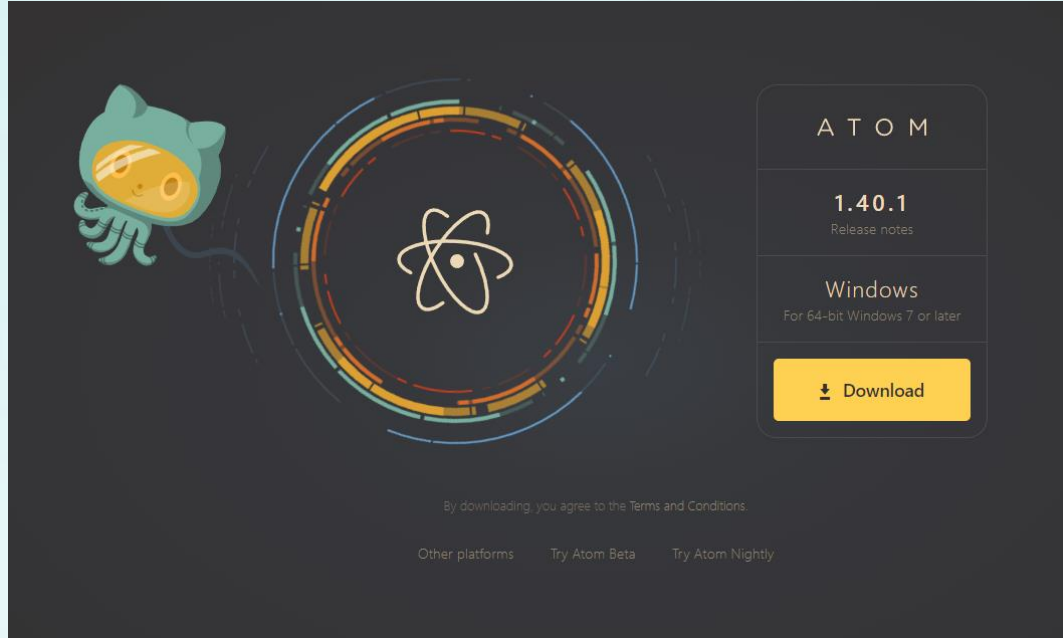
---

## MSYS2 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>

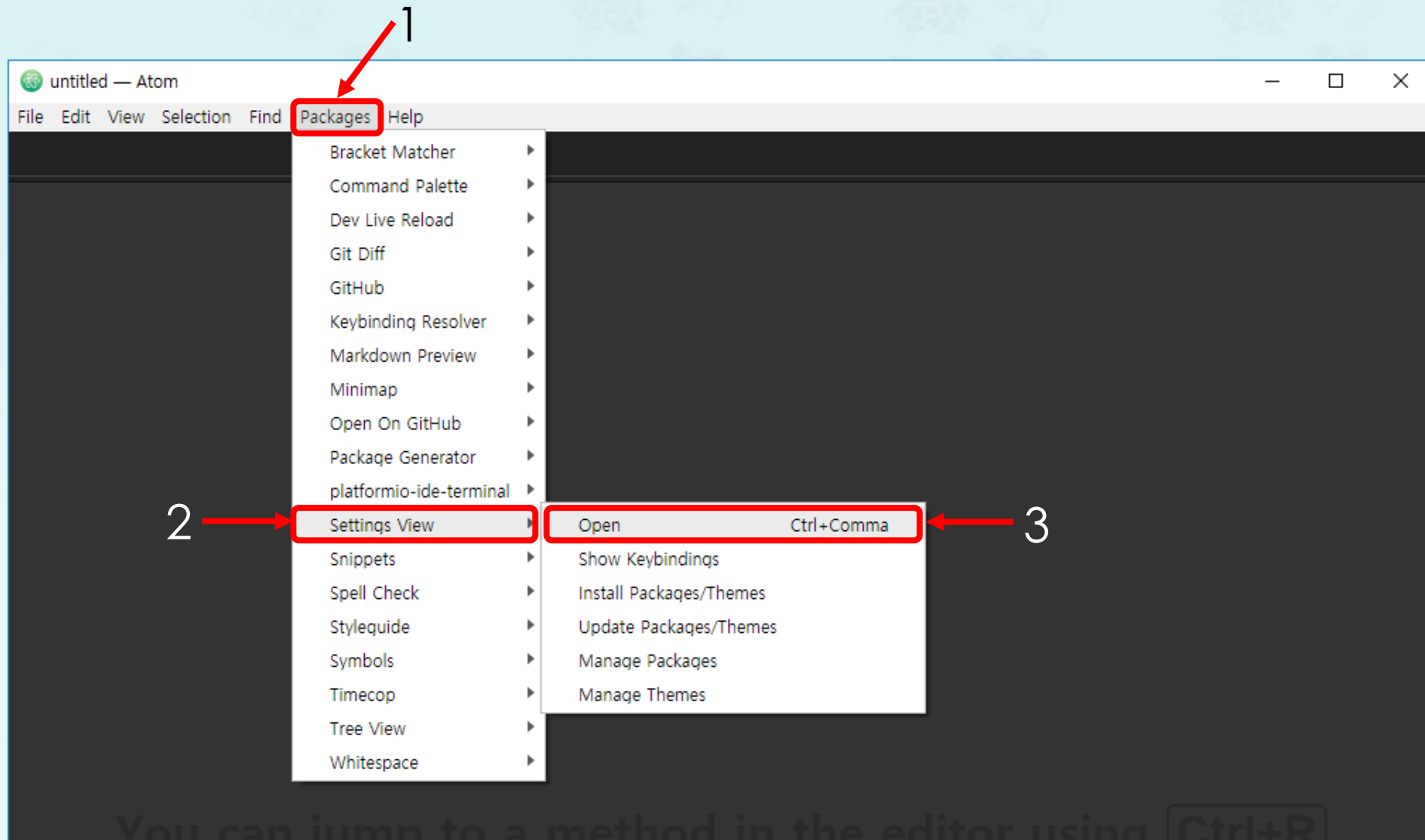
# Atom 설치하기

- <https://atom.io/>로 접속합니다.



- AtomSetup-x64.exe를 실행하면 자동으로 Atom이 설치가 완료됩니다.
- 다음 폴더를 사용자 혹은 시스템의 환경 변수 **Path**에 추가하면 편리합니다.  
**C:\Users\<username>\AppData\Local\atom\bin**

# Atom 패키지 설치하기

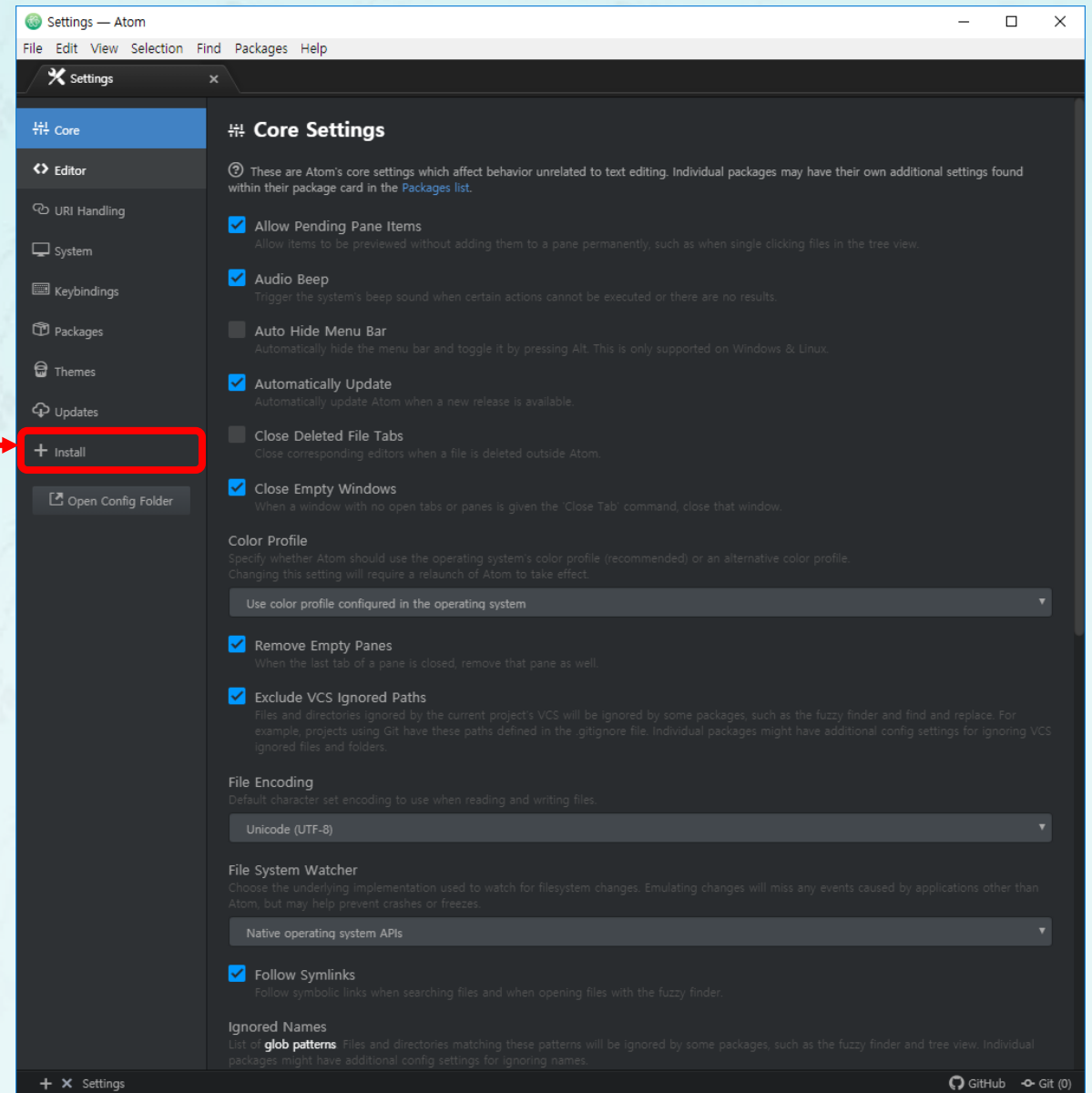




# Atom 패키지 설치하기

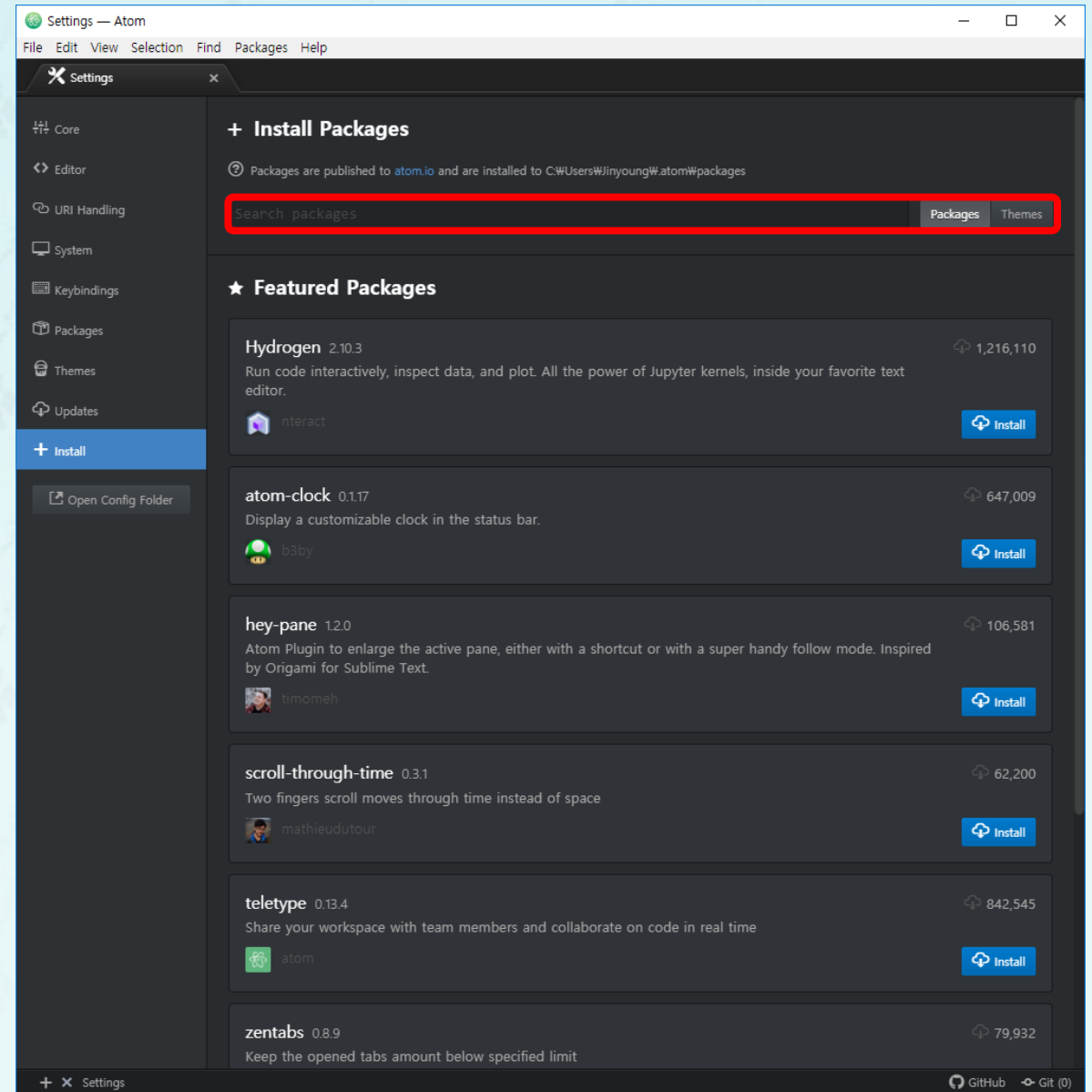
- Install를 클릭합니다.

클릭



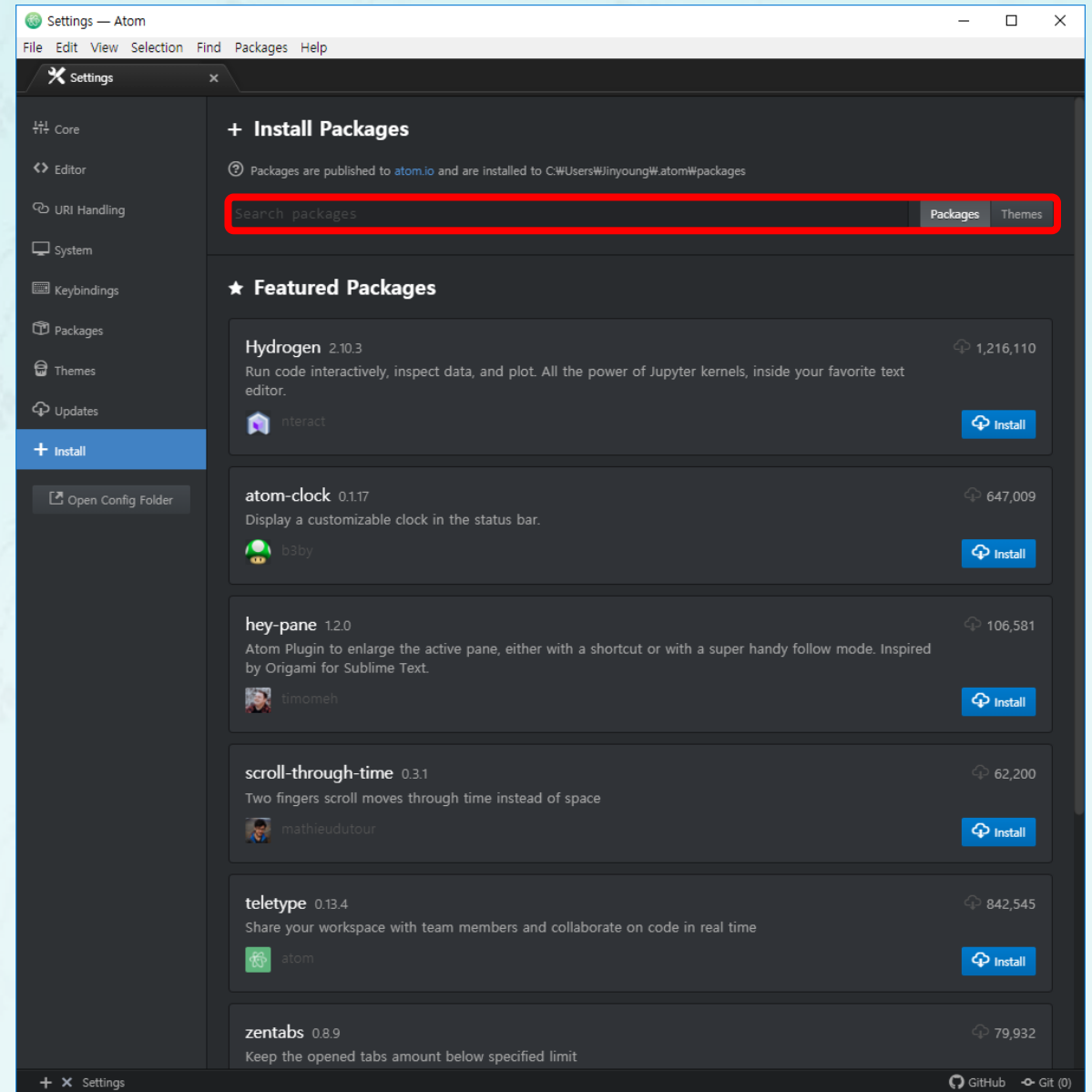
# Atom 패키지 설치하기

- 검색창에 다음의 패키지 이름들을 검색하여 설치합니다.
- platformio-ide-terminal
- file-icons
- minimap
- autocomplete-clang



# Atom 패키지 설치하기

- 검색창에 다음의 패키지 이름들을 검색하여 설치합니다.
- platformio-ide-terminal
- file-icons
- minimap
- autocomplete-clang



# MSYS2/MinGW-w64/Atom 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;
}
```

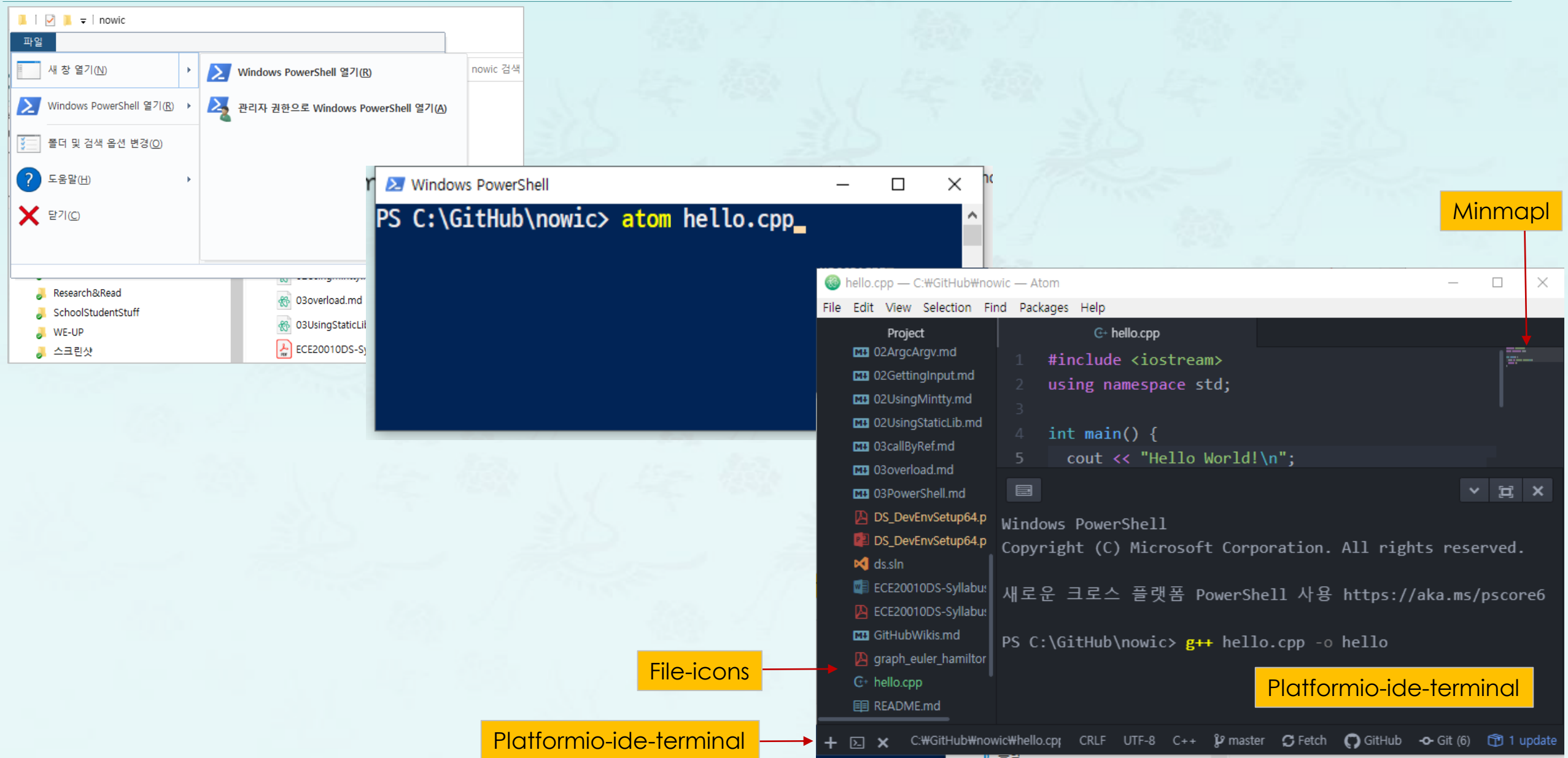
1. Copy the above program and save the file as **hello.cpp** on your folder.
2. Open a terminal at the folder where **hello.cpp** exists.  
Use the [File] → [Windows PowerShell 열기] menu at the top left corner of the **File Explorer**.
3. To compile, enter the following in the terminal and run it. We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

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

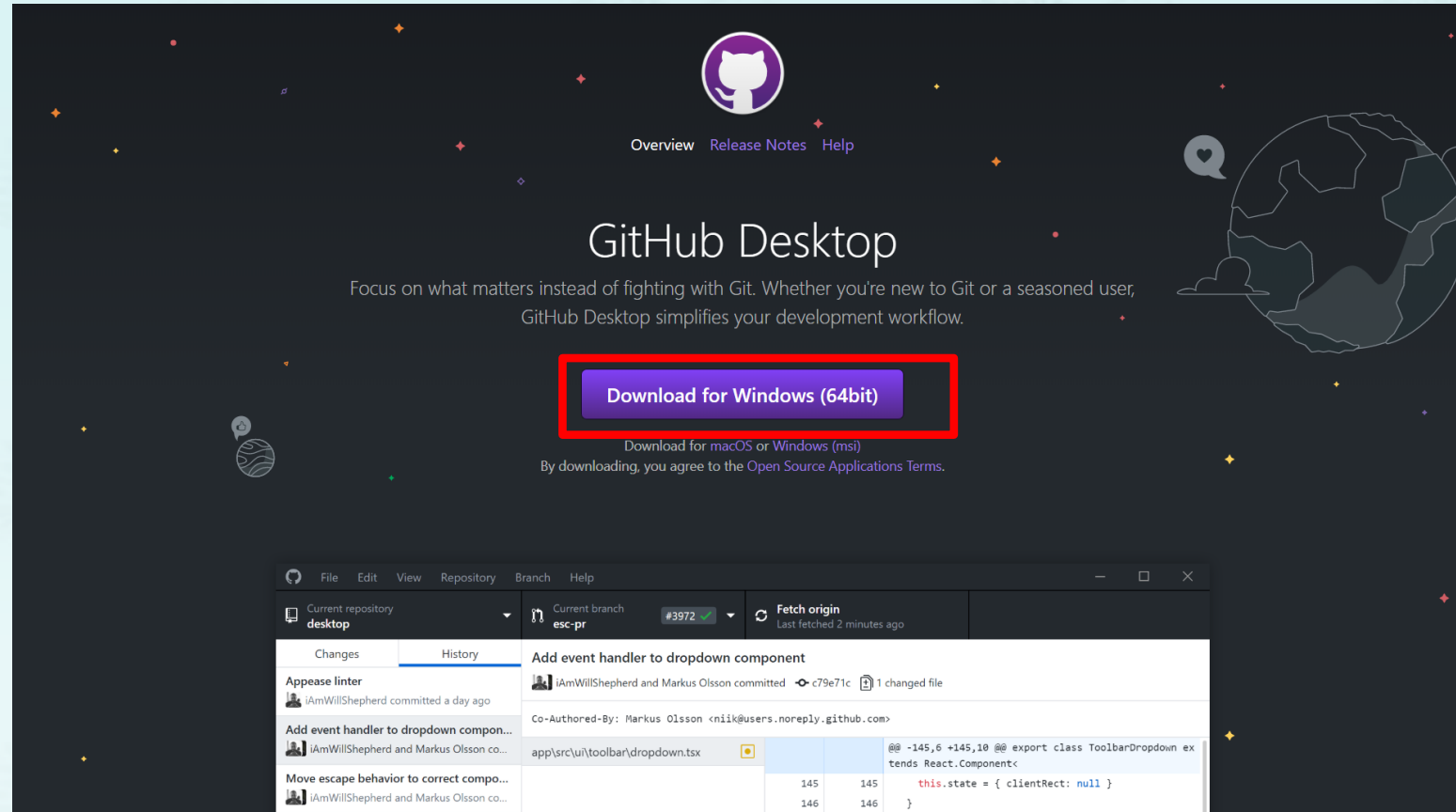
4. Run the compiled program by typing the following in the terminal. You will see "Hello World!" printed in the terminal.

```
$ ./hello
```

# MSYS2/MinGW-w64/Atom 설치 테스트



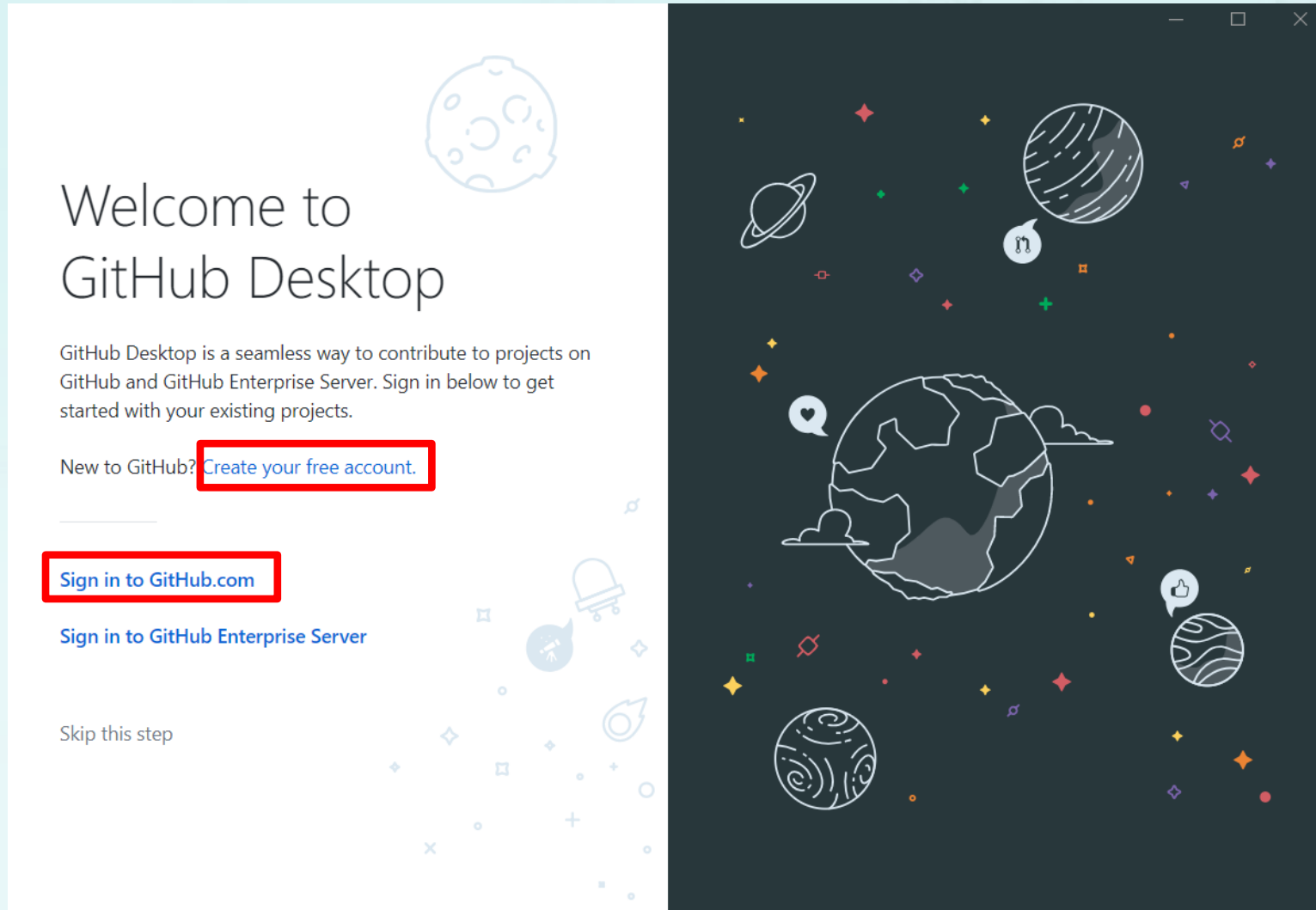
# GitHub Desktop 사용법



- <https://desktop.github.com/> 에서 다운로드 받습니다.



# GitHub Desktop 사용법



- Create your free account를 클릭하여 GitHub 가입을 합니다.
- Sign in GitHub을 클릭하여 로그인합니다.
- 다음 페이지 순서를 따라 github에서 여러분 자신의 repository 하나 만들어 봅니다.

# GitHub 첫 Repository 만들기

The screenshot shows the GitHub 'Create a new repository' page. The interface includes a top navigation bar with a search bar and links for Pulls, Issues, Marketplace, and Explore. A dropdown menu is open, showing options: New repository (highlighted), Import repository, New gist, New organization, and New project. The main form has fields for Owner (notibutj), Repository name (nowicx), and a Description (optional). Below these are radio buttons for Public and Private (selected). At the bottom, there are checkboxes for 'Initialize this repository with a README' (checked) and 'Add a license' (None selected). A green 'Create repository' button is at the bottom.

(1) New repository

Owner \* / Repository name \*

notibutj / nowicx (2)

Great repository names are short and memorable. Need inspiration? How about **verbose-fiesta**?

Description (optional)

☐ Public  
Anyone on the internet can see this repository. You choose who can commit.

☒ Private (3)  
You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

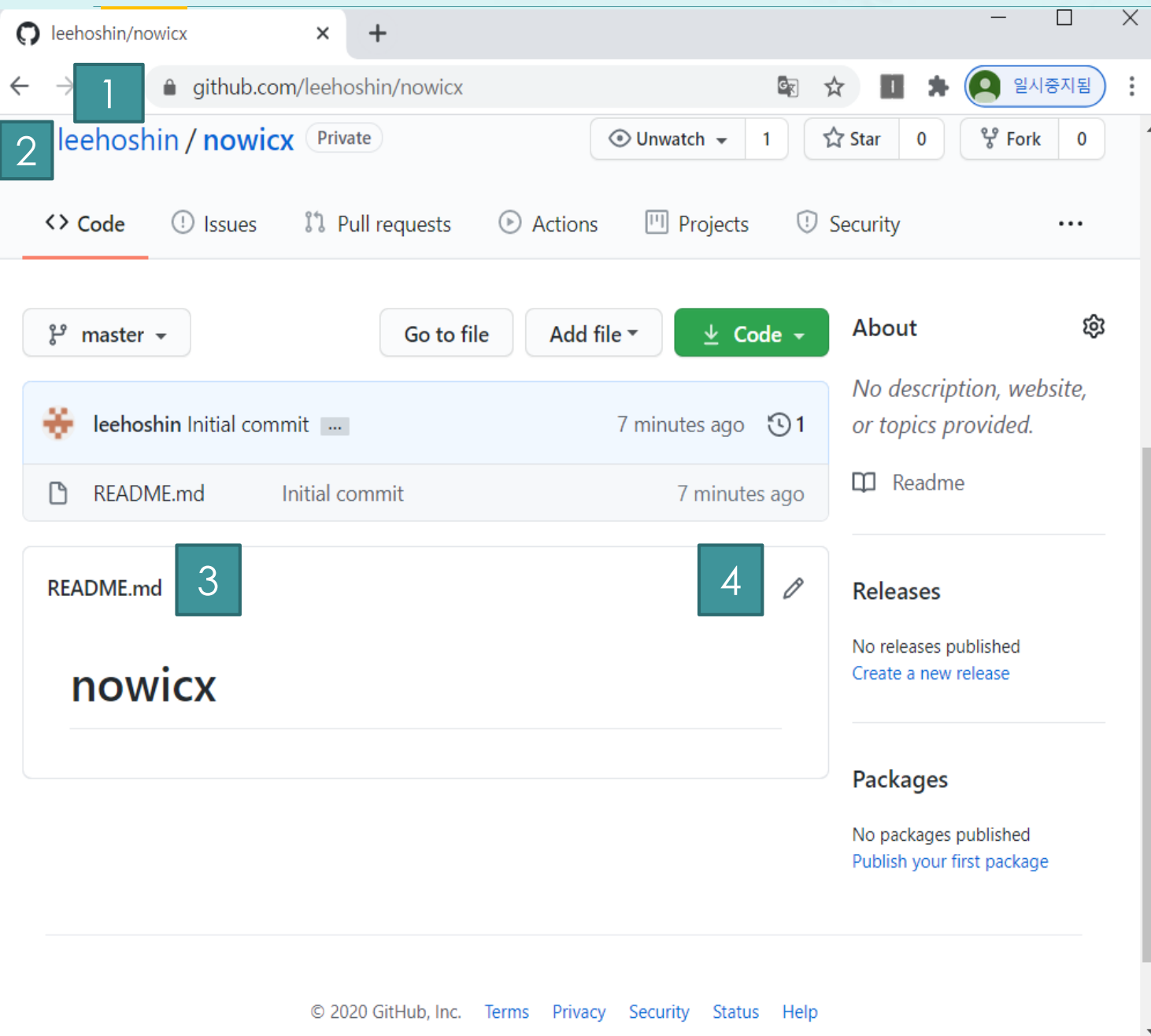
☒ Initialize this repository with a README (4)  
This will let you immediately clone the repository to your computer.

Add .gitignore: None Add a license: None (i)

Create repository (5)

1. +를 탭하여 [New Repository]를 선택합니다.
2. 여러분이 원하는 Repository 이름으로 설정합니다. 예를 들면, nowicx 입니다. (참고로, 데이터구조 클래스의 repository 이름은 nowic 입니다. nowicx라고 한 이유는 여러분이 nowic를 확장한 eXtended 한 것이란 의미입니다.)
3. Private으로 설정하여, 다른 사람들이 볼 수 없도록 합니다.
4. 체크해서 README 파일을 자동으로 생성합니다.
5. [Create repository] 을 탭하여 첫 repository를 생성합니다.

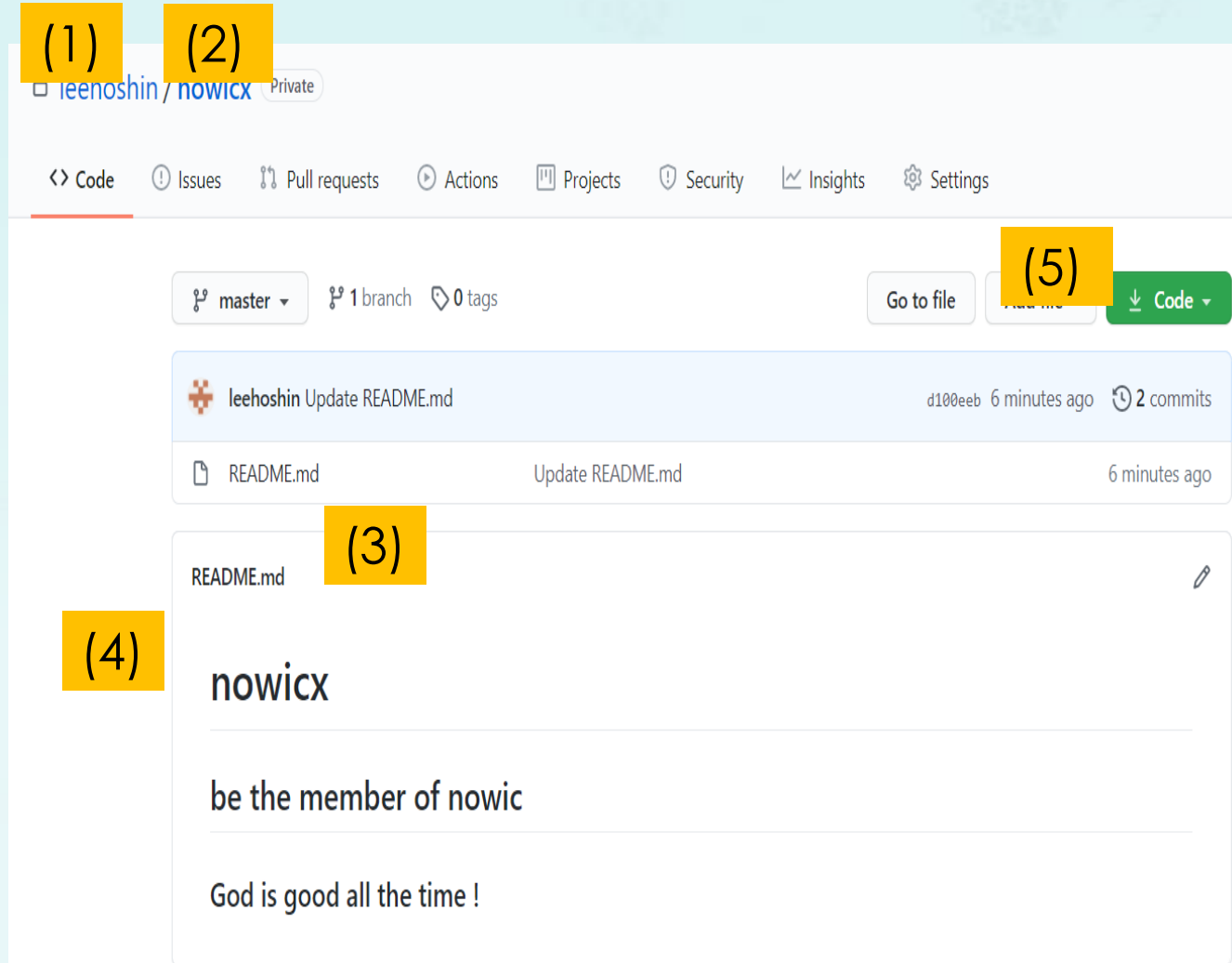
# GitHub 첫 Repository 만들기



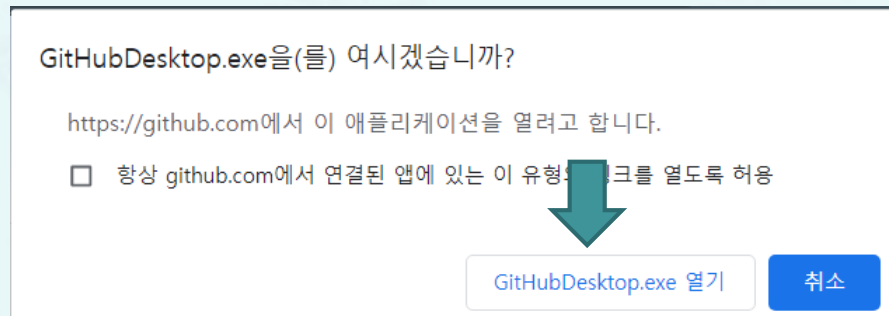
1. 클라우드(Github.com)에 여러분의 사용자 이름과 repository이름이 생성된 것을 확인할 수 있습니다.
2. 현재 여러분의 repository에 있다는 표시입니다. 여기를 탭하여 폴더들 사이를 이동할 수 있습니다.
3. 현재 repository에 있는 파일들의 리스트가 여기 나옵니다. 지금은 README 파일만 있습니다.
4. README 옆 연필을 탭하여 내용을 수정합니다

5. 수정한 후, 화면 아래 부분에 있는 [Commit changes] 를 탭하여 저장합니다.
6. 이제 github(클라우드)에 있는 repository를 여러분의 local computer에 복제를 시도하려고 합니다.

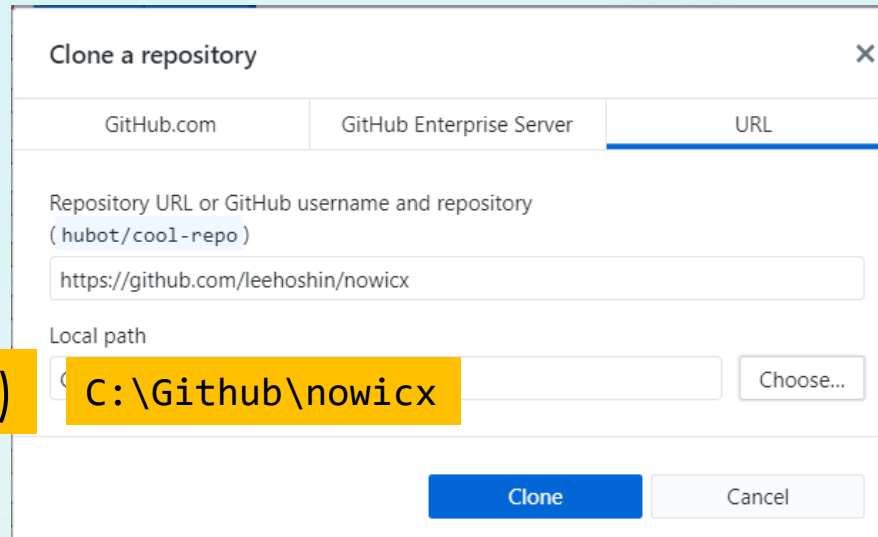
# GitHub 첫 Repository를 내 컴에 처음 복제하기



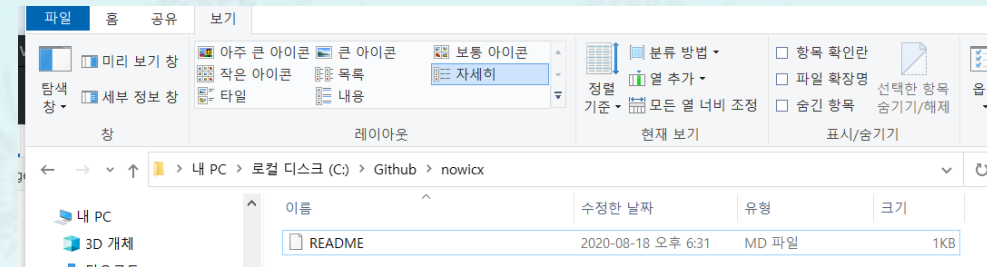
1. Github의 여러분 repository로 갑니다. 만약, 화면이 이렇게 보이지 않는다면, 사용자 이름이나 repository를 탭하여 여기로 찾아오십시오.
2. 현재 repository 이름을 보여줍니다.
3. 현재 repository에 있는 파일들의 리스트가 여기 나옵니다. 지금은 README 파일만 있습니다.
4. README 파일의 내용이 나타나 있습니다.
5. 이 repository에 있는 모든 파일을 zip파일로 다운 받거나 아니면, 있는 그대로 복제하는 방법을 선택할 수 있습니다. [Open with GitHub Desktop]을 선택합니다.
6. 또 다시 물을 때 같은 대답을 합니다.



# GitHub 첫 Repository를 내 컴에 처음 복제하기



7. "Create a repository"창에서, URL/Local Path를 지정하고 Clone하십시오.  
C Drive Root folder에 설정하면 편리합니다.  
Path는 C:\Github\nowicx  
해당 폴더에 clone(복제)된 것을 확인하십시오.



```
Windows PowerShell
PS C:\Github\nowicx> echo "be member of nowic"
be member of nowic
PS C:\Github\nowicx> echo "be member of nowic"> nowic.txt
PS C:\Github\nowicx> cat nowic.txt
be member of nowic
PS C:\Github\nowicx> dir

디렉터리: C:\Github\nowicx

Mode                LastWriteTime         Length Name
----                -
a----             2020-08-18 오후 6:33            42 nowic.txt
a----             2020-08-18 오후 6:31            69 README.md

PS C:\Github\nowicx>
```

8. echo와 stdout을 redirect 할 수 있는 > 를 사용하여 nowic.txt 파일을 생성하십시오.  
예를 들어, dir > myfiles.txt 도 가능합니다.
9. GitHub Desktop에서 [Commit to master]를 실행하고, 다음 [push origin]을 실행하여, 파일(nowic.txt)를 클라우드(github)에 추가하여 저장할 수 있습니다.

# 데이터구조 Repository를 내 컴에 복제하기

- GitHub Desktop에서 [데이터구조] 수업 자료를 가지고 오려면 Clone을 해야 합니다.
- <https://github.com/idebtor/nowic> 에 접속합니다.

Search or jump to... Pull requests Issues Marketplace Explore

idebtor / JoyAI Unwatch 26 Unstar 34 Fork 33

<> Code Issues 0 Pull requests 1 Actions Projects 0 Wiki Security Insights

모두를 위한 인공지능의 활용(GEK10109-Joyful AI for ALL)

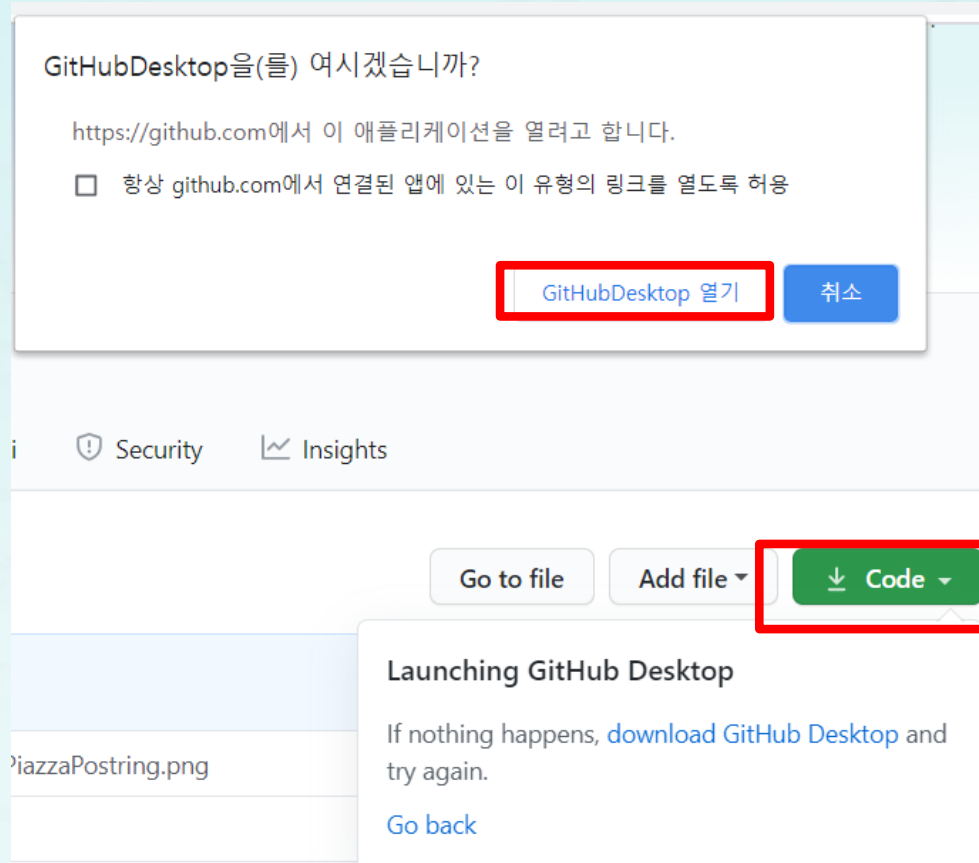
172 commits 1 branch 0 packages 0 releases 2 contributors

Branch: master New pull request Create new file Upload files Find file Clone or download

idebtor up		Latest commit 6474b78 on 8 Jan
.ipynb_checkpoints	udpated	12 months ago
book	up	2 months ago
ipynb	up	2 months ago
project_sample	Delete 8조_Emotion_Gender_Classification_프로젝트.ipynb	2 months ago
.gitignore	Create .gitignore	5 months ago
01GettingStarted.md	up	2 months ago
GEK10109Syllabus.pdf	up	2 months ago
GitHubWikis.md	Update GitHubWikis.md	5 months ago
README.md	Update README.md	12 months ago



# 데이터구조 Repository를 내 컴에 복제하기



- Code 클릭 → Open with the github Desktop 클릭
- 알림 창에서 GitHubDesktop.exe 열기 클릭

# 데이터구조 Repository를 내 컴에 복제하기

Clone a repository

GitHub.com | GitHub Enterprise Server | URL

idebtor/nowic

Your repositories

idebtor/nowic

Local path

C:\Github\nowic

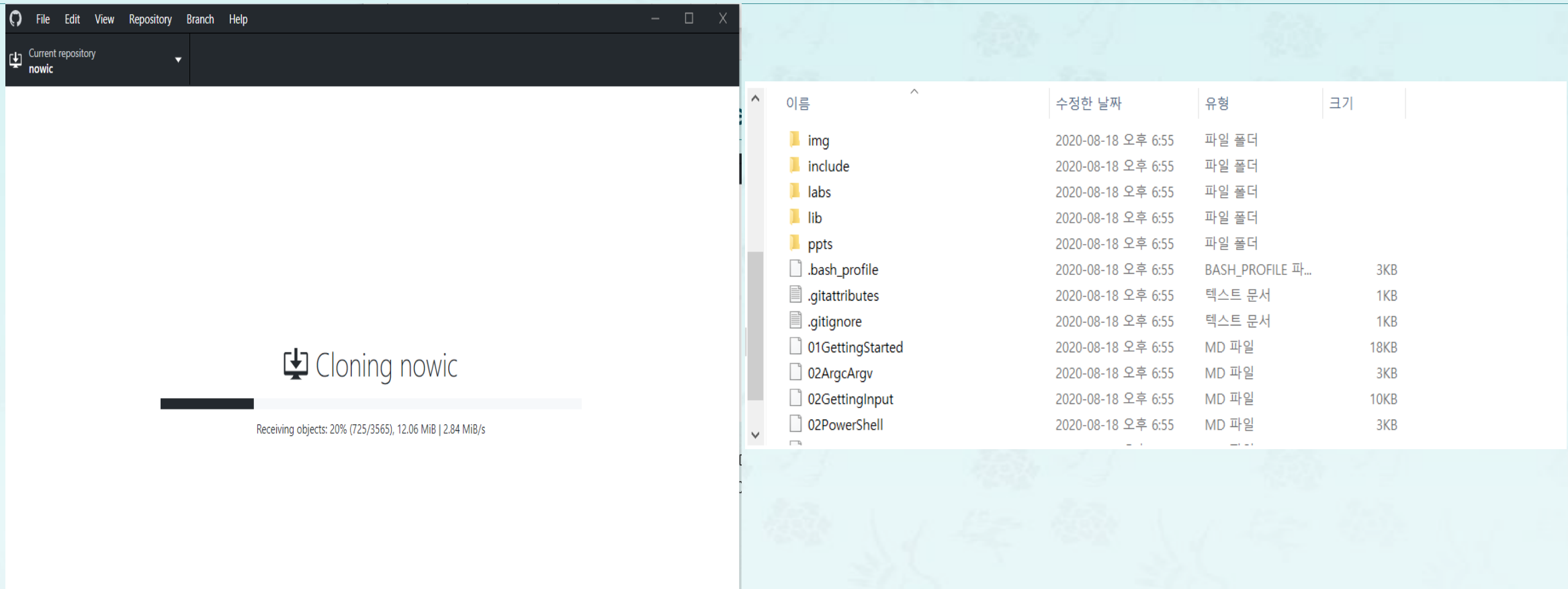
Choose...

Clone

Cancel

- 이미 내 컴에 Github가 설정이 있다면, Local Path가 저절로 설정이 되어 있으므로, [Clone]을 선택하면 됩니다.
- 맨 처음의 경우, 또한 특별히 필요한 경우, Local path는 Github에 올린 자료를 다운로드(Clone) 받는 경로를 원하는 곳으로 변경해주세요.

# 데이터구조 Repository를 내 컴에 복제하기



The screenshot shows a Git client window with the following details:

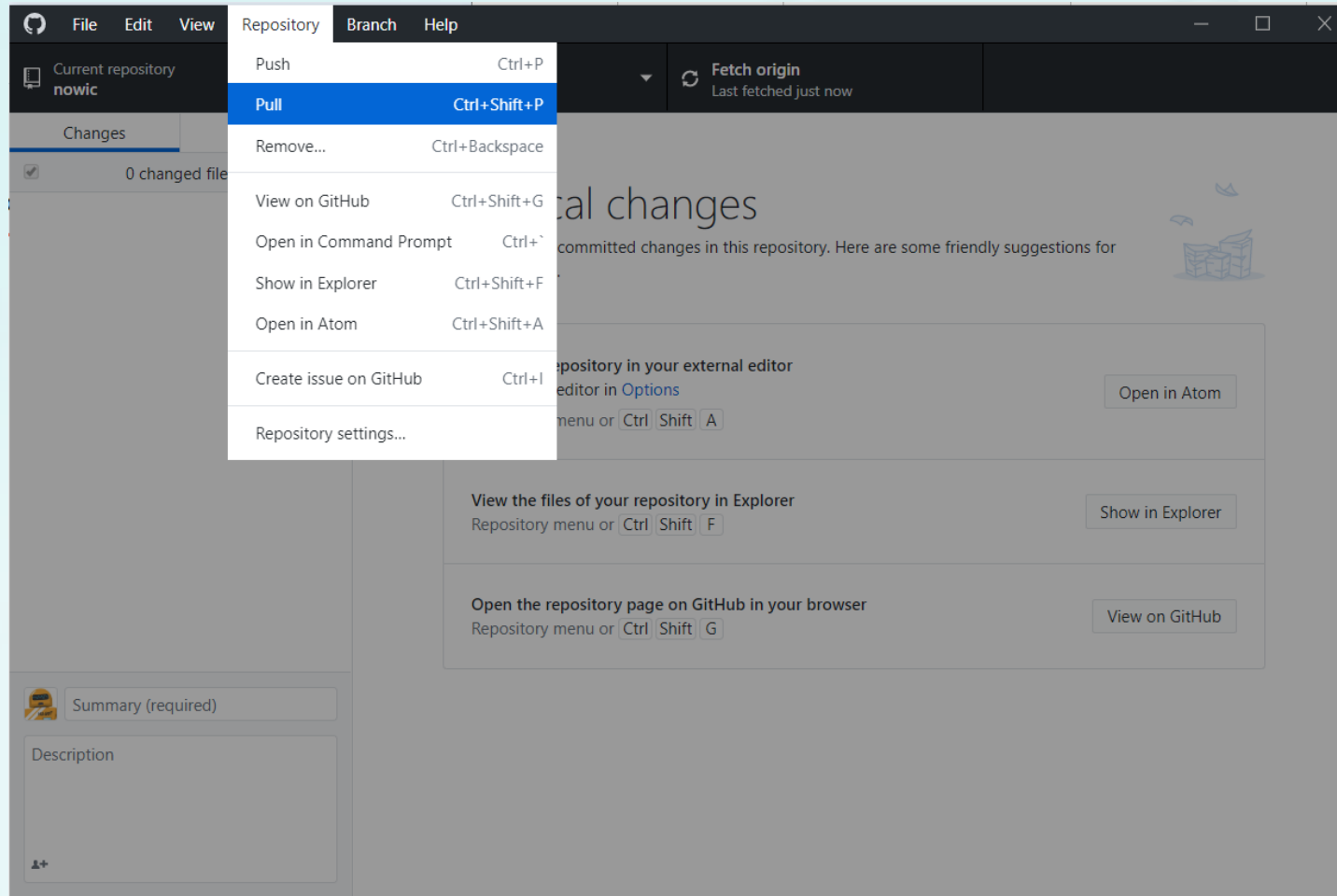
- Current repository:** nowic
- Cloning nowic**
- Progress bar:** Receiving objects: 20% (725/3565), 12.06 MiB | 2.84 MiB/s

The file explorer on the right displays the following files and folders:

이름	수정한 날짜	유형	크기
img	2020-08-18 오후 6:55	파일 폴더	
include	2020-08-18 오후 6:55	파일 폴더	
labs	2020-08-18 오후 6:55	파일 폴더	
lib	2020-08-18 오후 6:55	파일 폴더	
ppts	2020-08-18 오후 6:55	파일 폴더	
.bash_profile	2020-08-18 오후 6:55	BASH_PROFILE 파...	3KB
.gitattributes	2020-08-18 오후 6:55	텍스트 문서	1KB
.gitignore	2020-08-18 오후 6:55	텍스트 문서	1KB
01GettingStarted	2020-08-18 오후 6:55	MD 파일	18KB
02ArgcArgv	2020-08-18 오후 6:55	MD 파일	3KB
02GettingInput	2020-08-18 오후 6:55	MD 파일	10KB
02PowerShell	2020-08-18 오후 6:55	MD 파일	3KB

- Progress bar가 끝까지 도착할 때까지 기다려주세요
- Clone을 완료하면 설정한 폴더에 다음과 같은 파일이 저장됩니다.

# 데이터구조 Repository를 내 컴에 복제하기



- <https://github.com/idebtor/nowic>에 새로운 파일이 올라오면 자동으로 자신의 컴퓨터에 다운로드 받아지지 않습니다.
- Repository → Pull을 클릭하여 자신의 컴퓨터에 다운로드 받으면 됩니다.

# 내 컴의 GitHub/**nowic** & **nowicx** Repository 다루기

- C:/Github/nowicx  
데이터구조 클래스를 위한 나의 폴더이므로, 여기서 모든 작업을 합니다.  
또한 Github에 upload도 하면서 파일을 관리할 수 있습니다  
데이터구조 클래스에 필요한 자료들을 nowic 폴더에서 여기로 복사하여 사용합니다.
- C:/Github/nowic
  - 데이터구조 클래스 학습 자료(<https://github.com/idebtor/nowic>)를 내 컴에 복제한 것으로 파일 수정할 권한이 없음
  - 만약, 이 폴더에 있는 파일을 수정한다면, Github/nowic에서 새로 파일을 다시 다운받을 때 어려움을 겪을 수 있습니다.
  - (1) 이 문제의 해결 방법은 내 컴에 nowic 폴더를 삭제하고 새로 Clone하면 됩니다.
  - 또 다른 방법은 콘솔(cmd창)을 열고, github/nowic 폴더에서 다음 명령어를 실행하여 강제로 다운 받거나 수정한 파일을 다시 복구하는 방법입니다. (이를 위해서, 내 컴에 git 을 설치해 두어야 합니다)  
(2) 다음은 Local에 있는 파일들을 강제로 덮어 쓰라는 명령어입니다.  
`Git fetch -all`  
`git reset -hard origin/master`
  - (3) Local에서의 모든 수정을 취소하여 파일들을 복구한 후, 모두 원격 저장소로부터 fetch 하라는 명령어입니다.  
`git stash`  
`git fetch --all`

# Piazza App 소개

---



- 수업에서 활용하는 Piazza 사이트는 안드로이드, IOS 스토어에서 다운로드 받을 수 있습니다. 모바일에서 편리하게 이용하세요.
- 안드로이드: <https://play.google.com/store/apps/details?id=com.piazza.android&hl=ko>
- IOS: <https://apps.apple.com/us/app/piazza/id453142230>





## Installing Git – the easy way

---

- Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.



**git fast-version-control**

<https://git-scm.com/>

- If you've never used git or github before, there are a bunch of things that you need to do.
- Get a **github** account.
- Download and install **git**.

# 데이터 구조 개발환경

ECE20010

한동대학교  
CSEE 자료구조  
idebtor@gmail.com