

git 설치와 기본 설정



About..

컴퓨터소프트웨어공학과
김 원 일

❖ 목 차



- git 다운로드 및 설치
- github 계정 만들기
- 나의 저장소 관리하기
- 저장소 로컬로 가져오기
- git 명령어 동작 위치
- 저장소 상태 정보 확인 / 수정
- github 인증 토큰 생성
- github 인증 토큰 사용
- github 인증 토큰 삭제



- git 설치 프로그램 다운로드

- <http://git-scm.com/> 사이트 접속 및 최신 버전 다운로드



The screenshot shows the Git website homepage. At the top, there's a navigation bar with the Git logo and the tagline "--local-branching-on-the-cheap". A search bar is on the right. The main content area has two paragraphs describing Git as a free and open source distributed version control system. To the right of the text is a diagram showing a branching model with multiple stacks of code blocks connected by colored lines. Below the text are four circular icons with labels: 'About' (gears), 'Documentation' (book), 'Downloads' (downward arrow), and 'Community' (speech bubbles). On the right side, there's a monitor displaying the 'Latest source Release 2.30.1' with a red box highlighting the 'Download 2.30.1 for Windows' button.

git --local-branching-on-the-cheap

Search entire site...

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 is **easy to learn** and has a **tiny footprint with lightning fast performance**. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like **cheap local branching**, convenient staging areas, and **multiple workflows**.

About
The advantages of Git compared to other source control systems.

Documentation
Command reference pages, Pro Git book content, videos and other material.

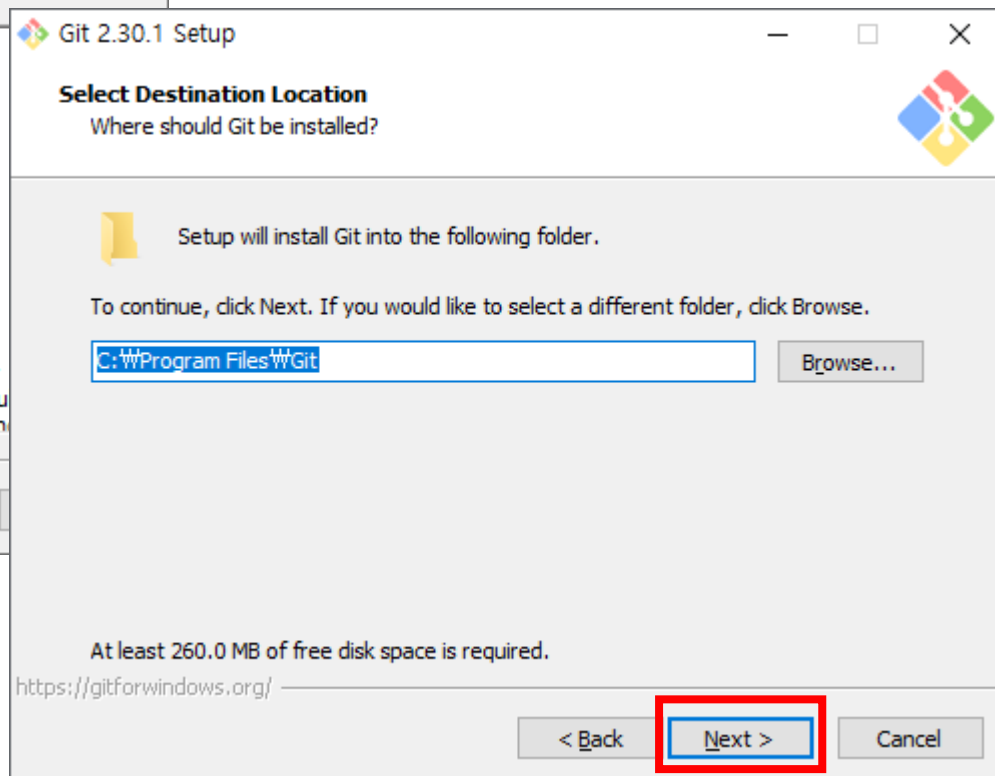
Downloads
GUI clients and binary releases for all major platforms.

Community
Get involved! Bug reporting, mailing list, chat, development and more.

Latest source Release
2.30.1
[Release Notes \(2021-02-08\)](#)
Download 2.30.1 for Windows



• 라이선스 동의와 설치 경로 설정

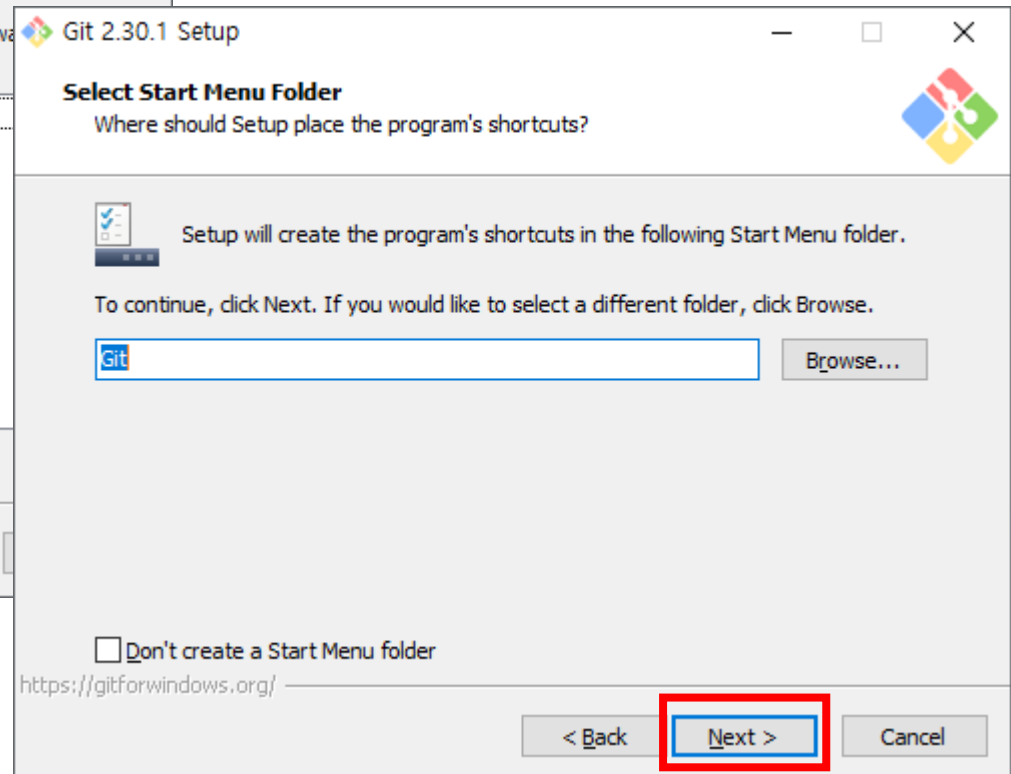
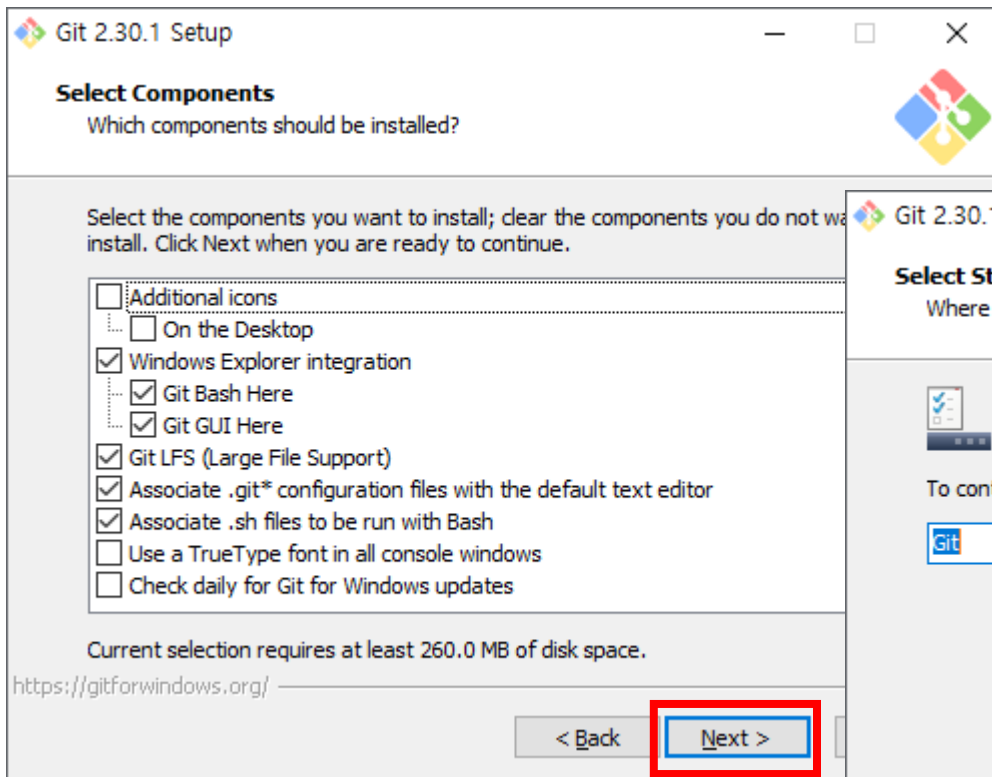




git 설치 - 2



- 설치 도구와 시작 메뉴 이름 설정
 - 설치 도구는 그대로 진행

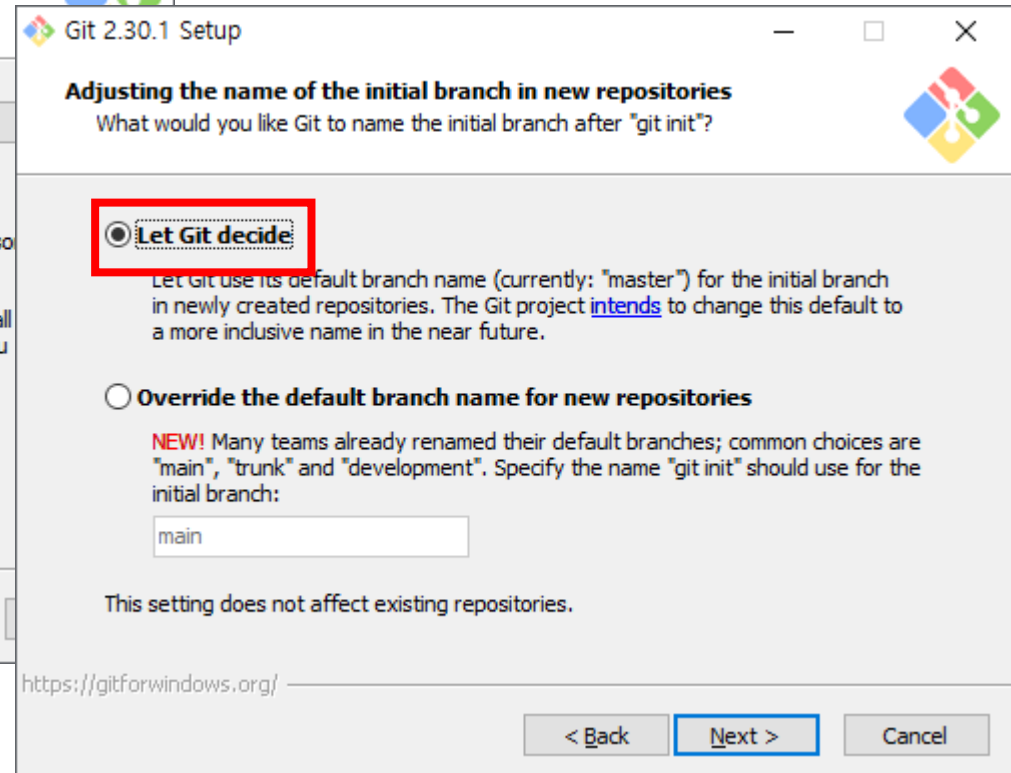
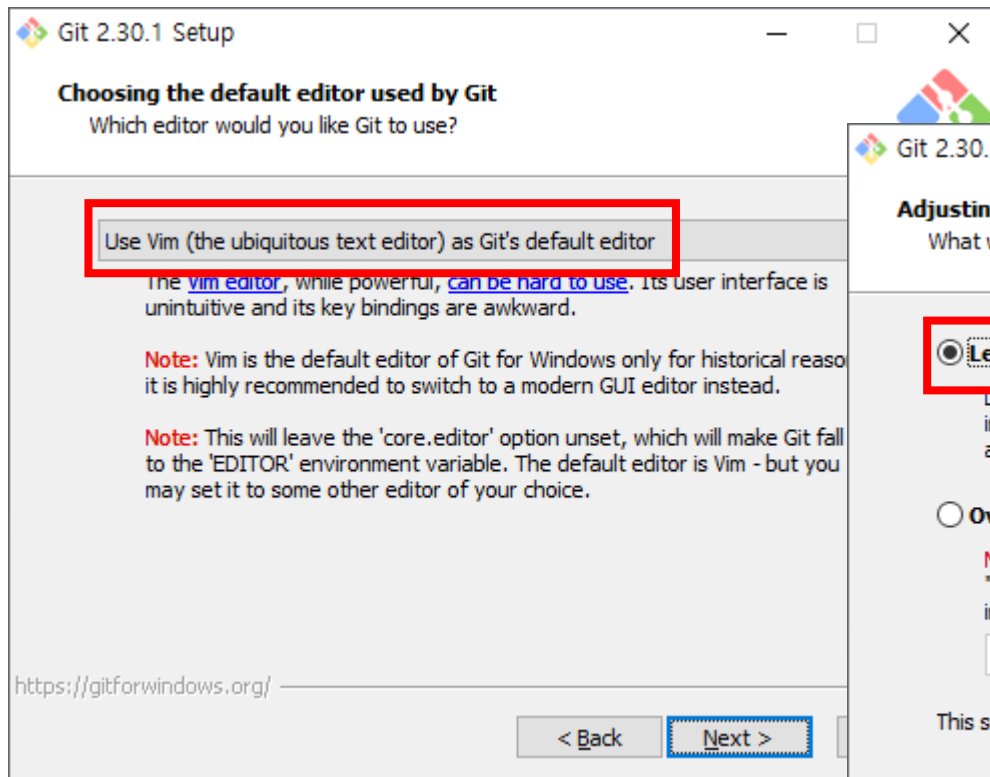




git 설치 - 3

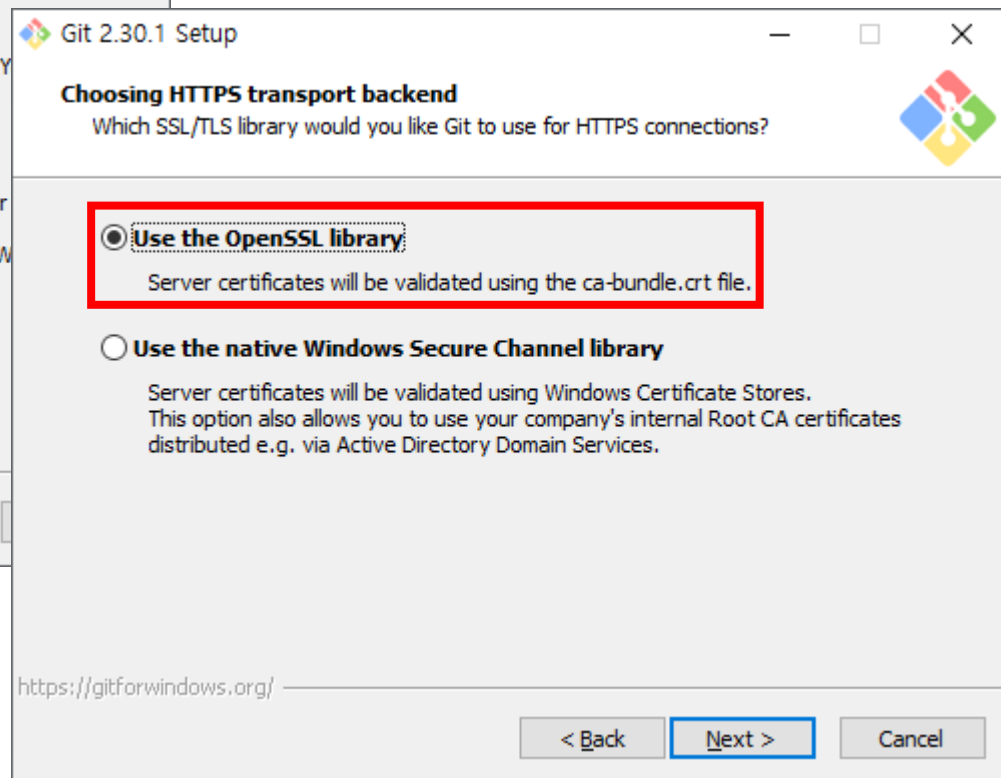
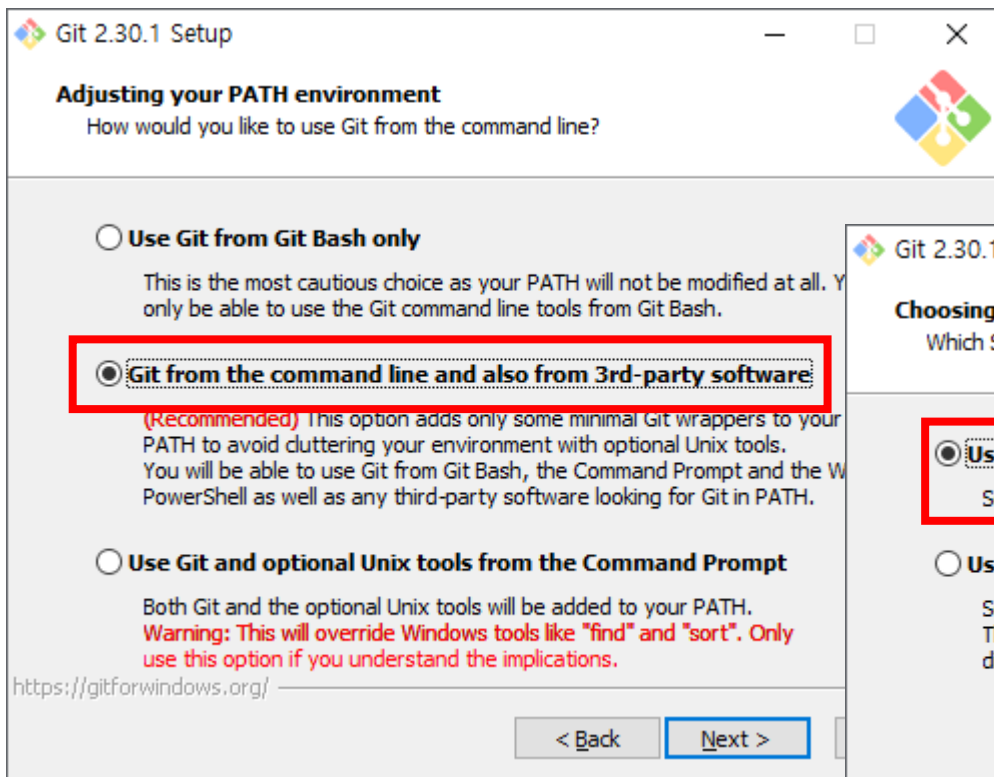
- 문서 편집기와 브랜치 처리

- linux 수업과 연계를 위해 Vim을 편집기로 설정
- git에서 관리하도록 진행을 권장



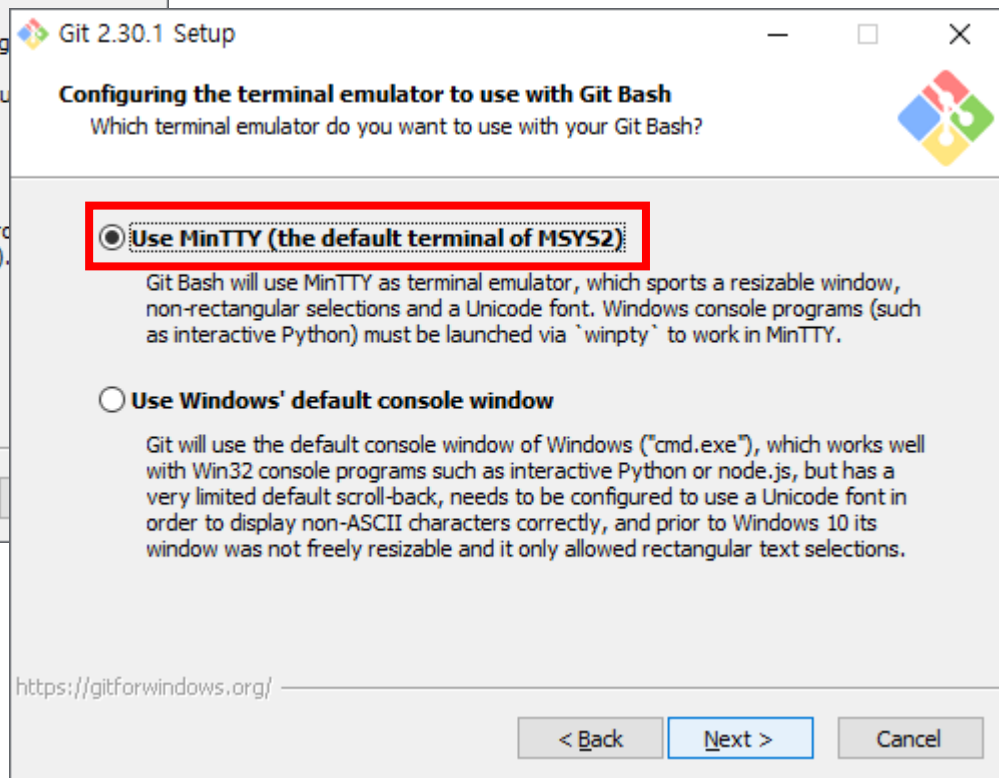
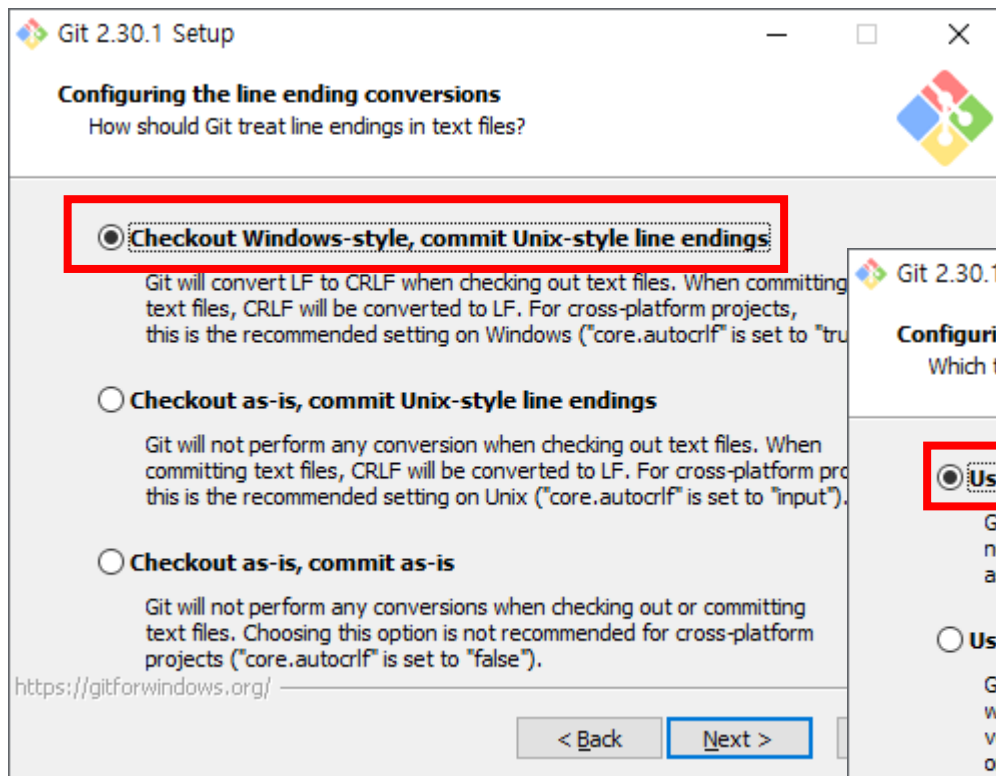


- 시스템 경로에 git 등록과 보안 연결 방식 설정



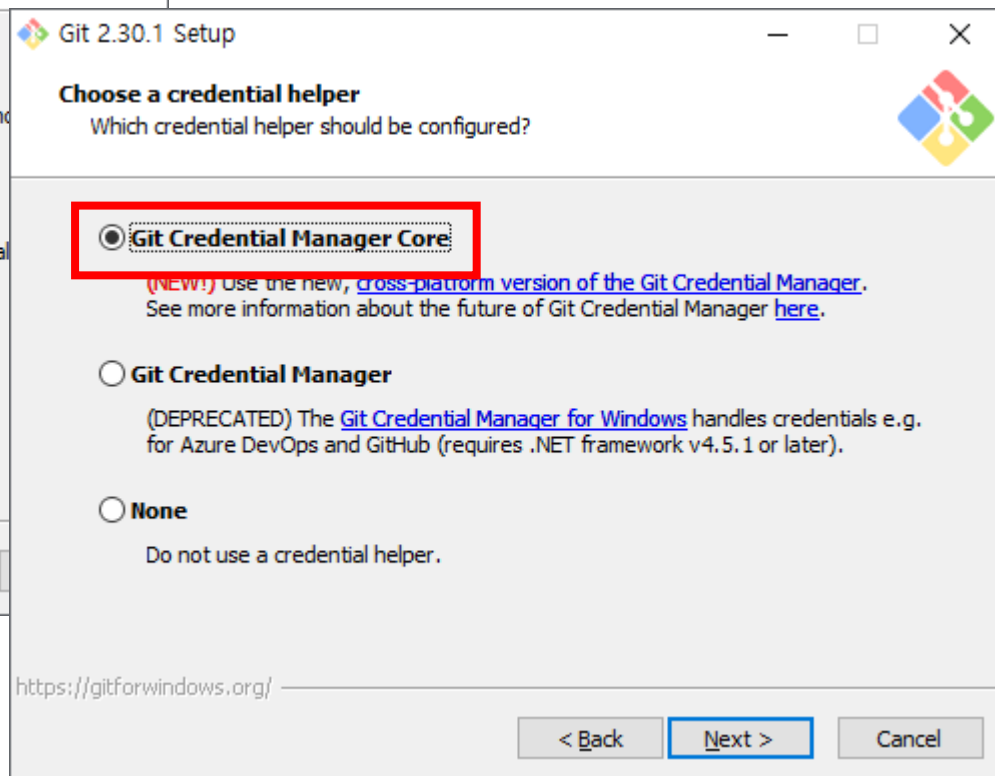
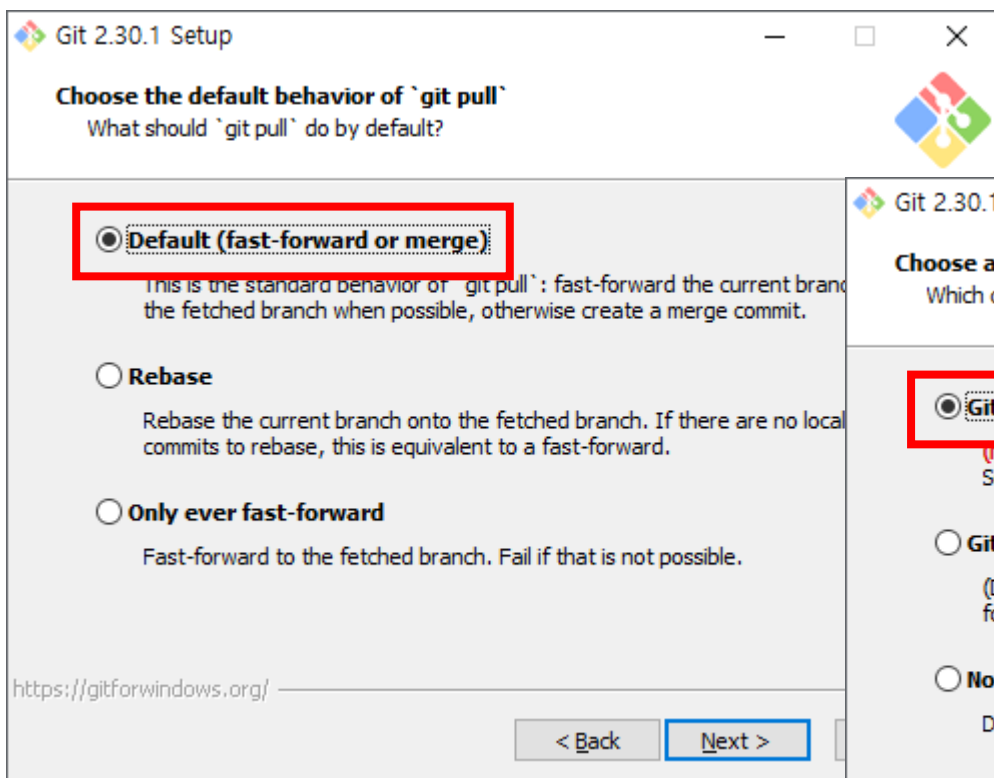


• 소스코드 처리 방식 설정



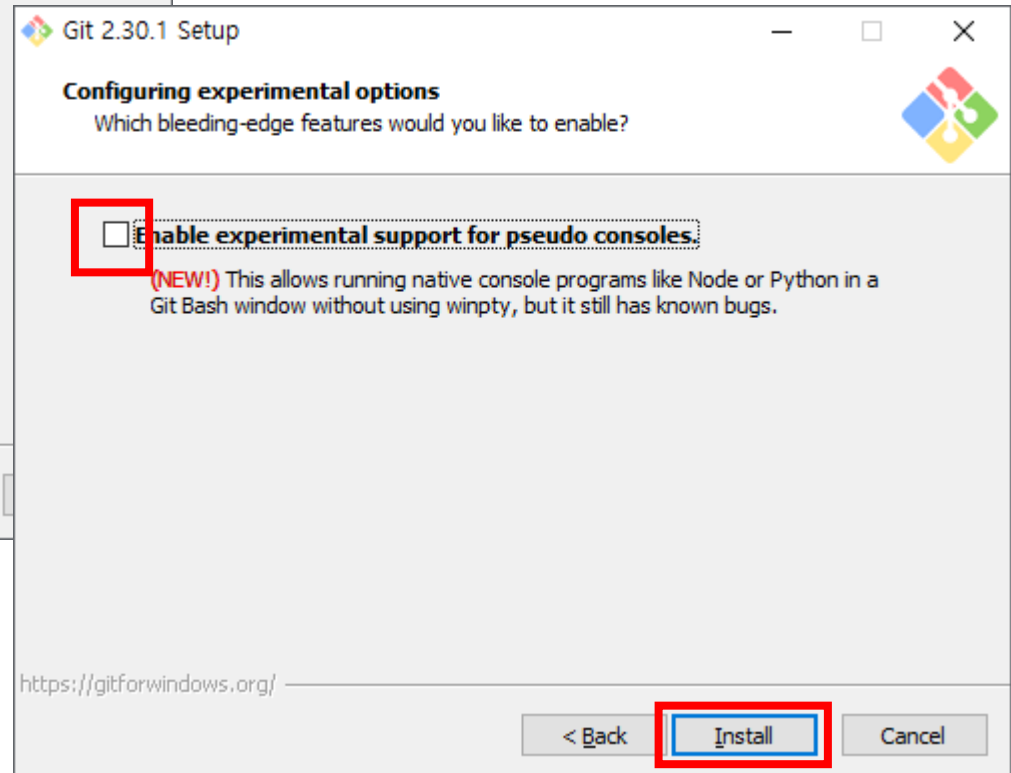
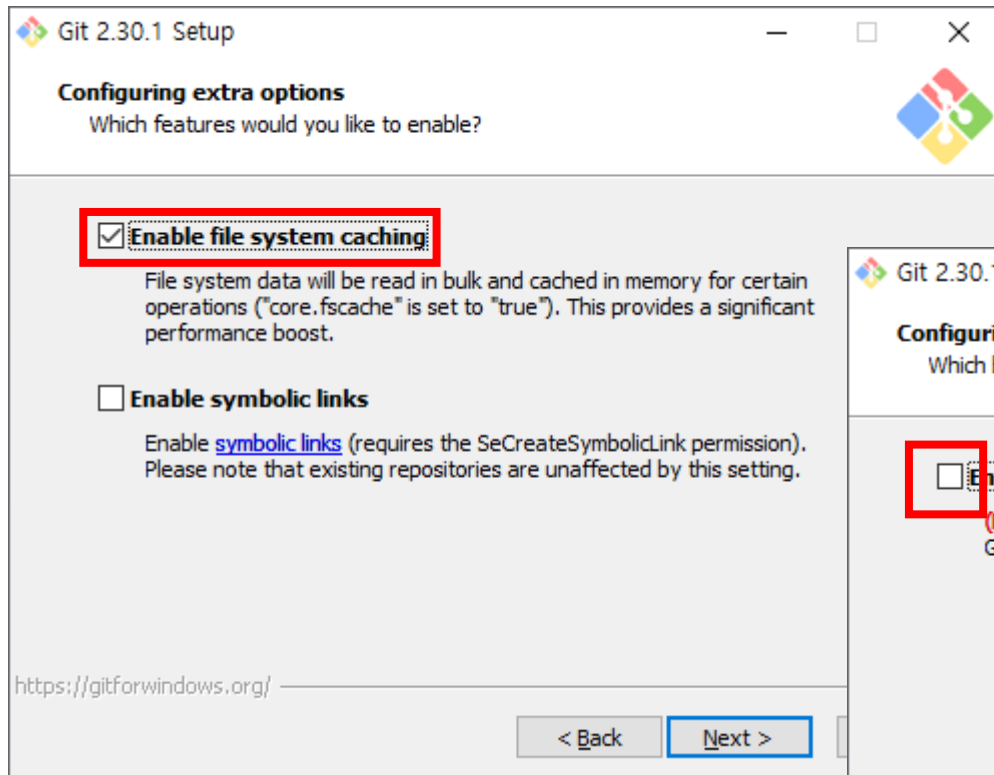


- 소스 다운로드 시의 동작 설정과 자격증명 설정
 - git bash에서 로그인 수행을 위한 설정



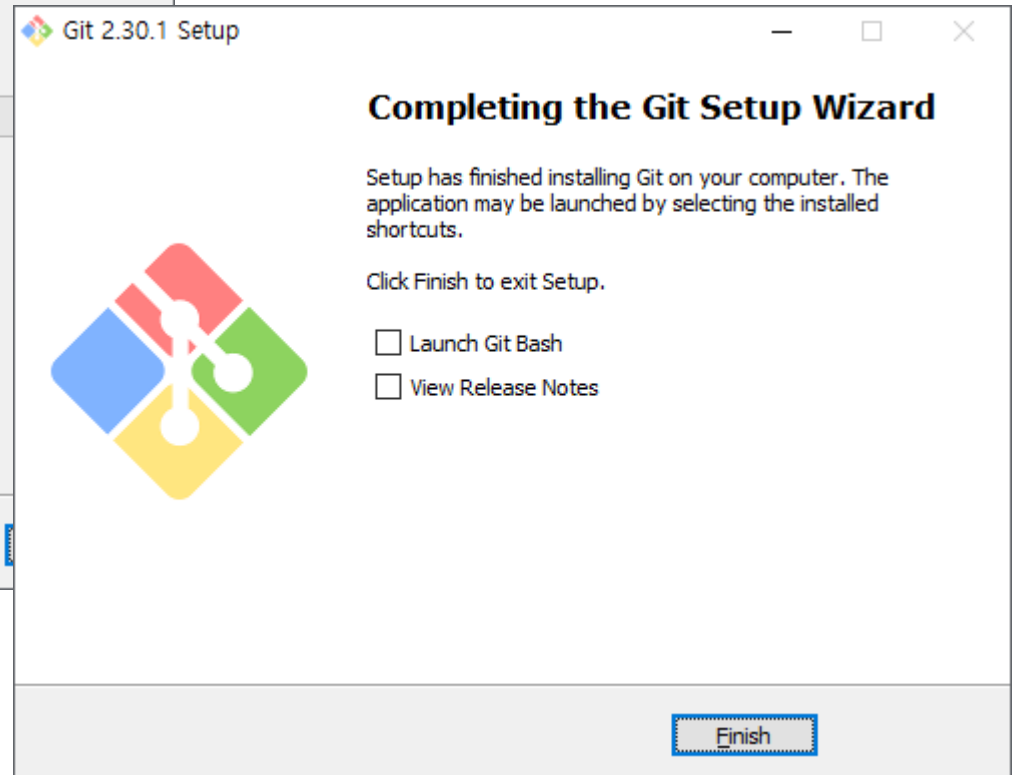
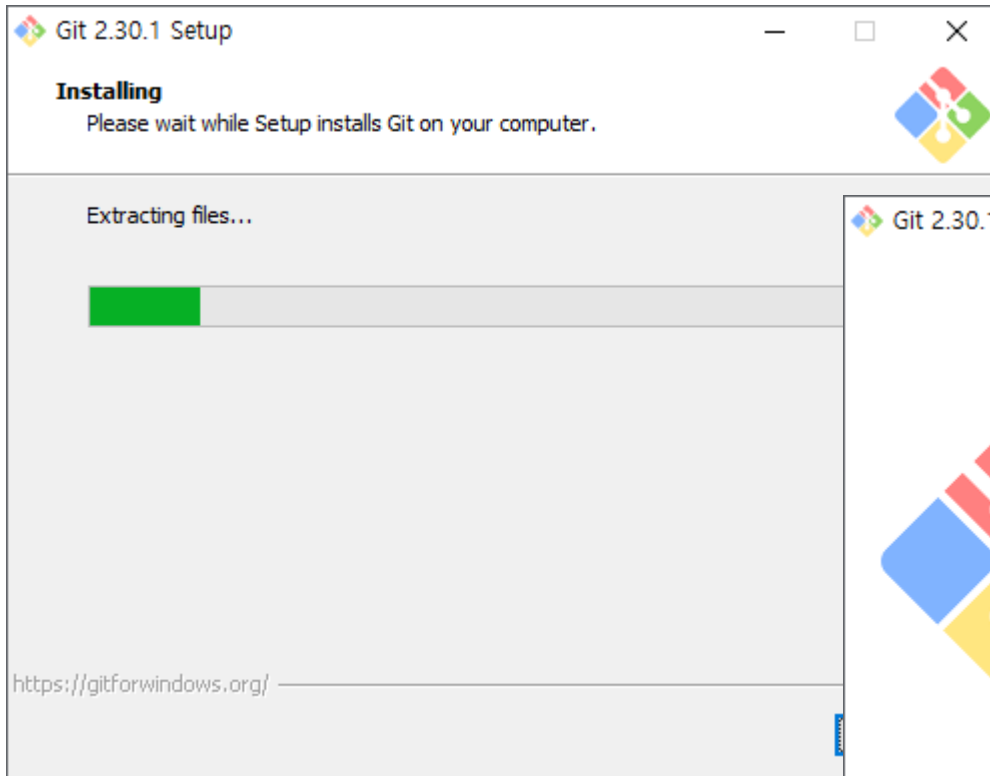


- 추가 옵션과 테스트 중인 옵션 설정





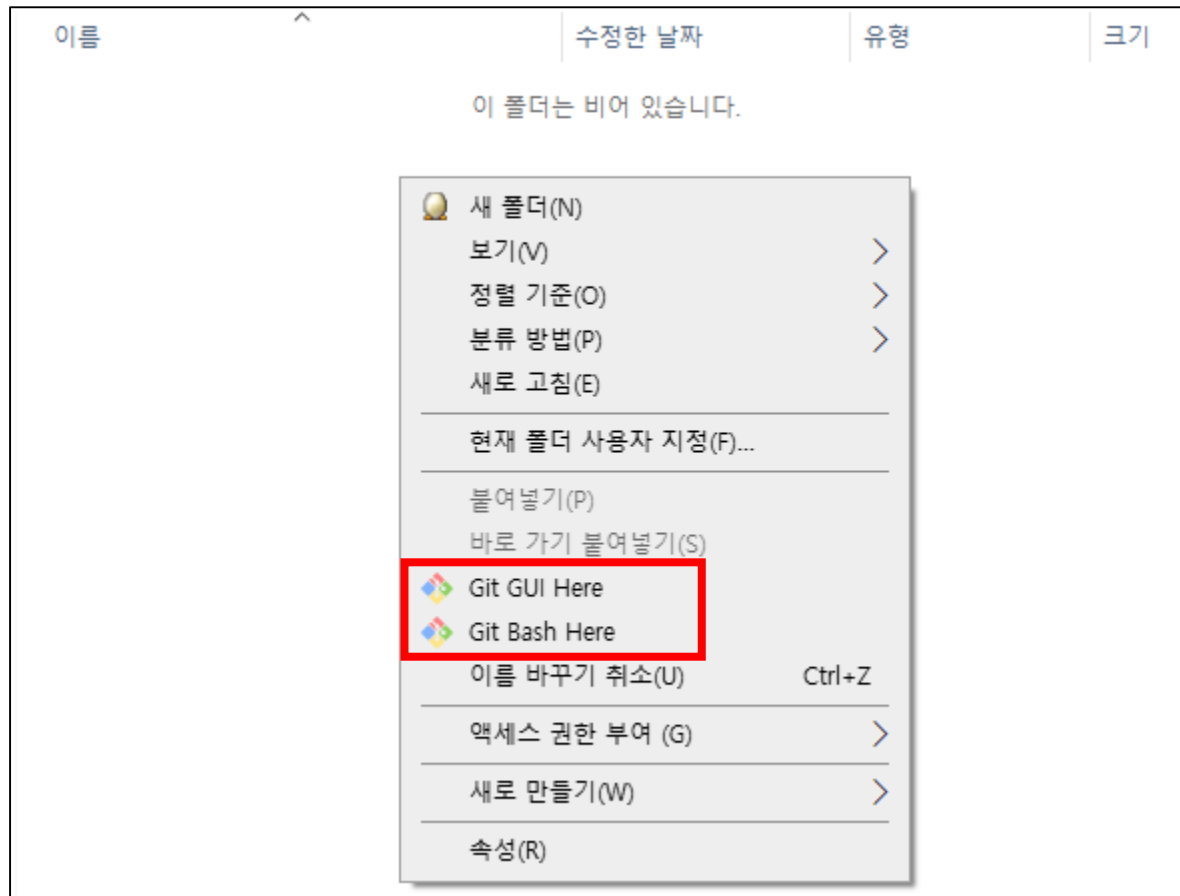
- 설치 및 종료





• 탐색기에서 설치 확인

- 빈 곳에서 오른쪽 클릭 시, git 메뉴의 추가 여부 확인 가능



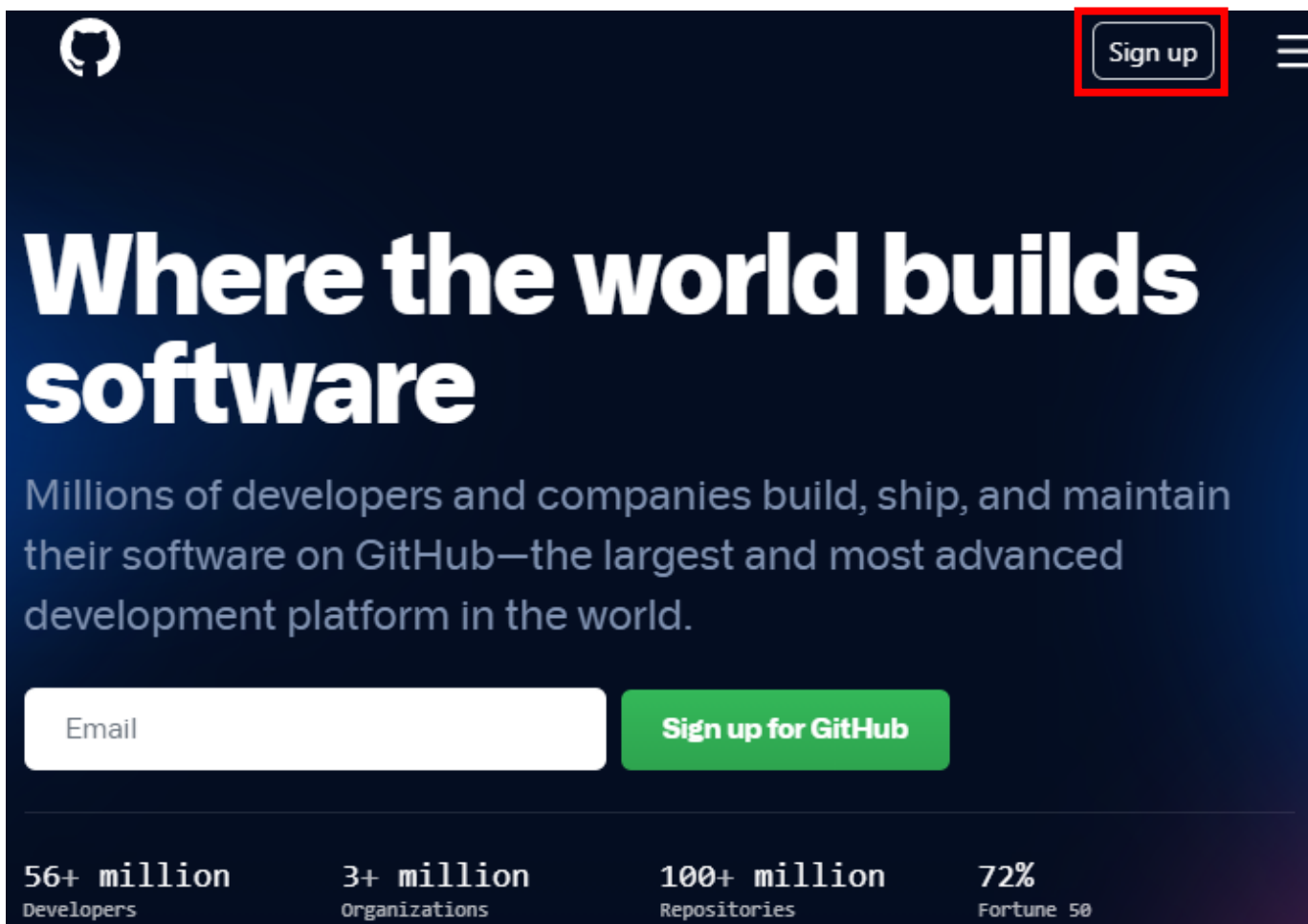


github 계정 만들기 - 1



- github 사이트 접속

- <https://github.com/> 에서 "Sign up" 선택





github 계정 만들기 - 2



- 계정 명, 메일주소, 비밀번호 입력으로 계정 생성
- 메일 인증 후, "Sign in"으로 로그인

Join GitHub

Create your account

Username *


Email address *

Password *

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter.
[Learn more.](#)

Email preferences

☐ Send me occasional product updates, announcements, and offers.



Sign in to GitHub

Username or email address

Password [Forgot password?](#)

[Sign in](#)

New to GitHub? [Create an account.](#)

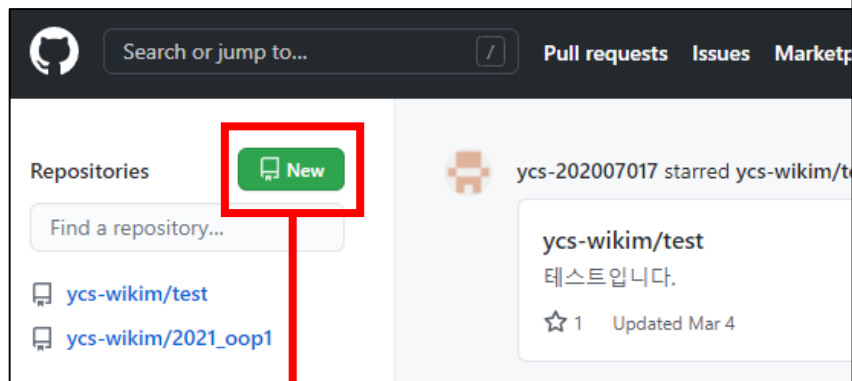
[Terms](#) [Privacy](#) [Security](#) [Contact GitHub](#)



나의 저장소 관리하기 - 1




• 나의 메인 페이지



Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)


Owner * Repository name *

 ycs-wikim ▼ /

Great repository names are short and memorable. Need inspiration? How about [miniature-train](#)?

Description (optional)

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

☐  **Private**
You choose who can see and commit to this repository.

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

☐ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

☐ **Add .gitignore**
Choose which files not to track from a list of templates. [Learn more.](#)

☐ **Choose a license**
A license tells others what they can and can't do with your code. [Learn more.](#)




나의 저장소 관리하기 - 2



- 저장소 이름

- 저장소는 항상 본인 계정 명 뒤에 입력한 이름으로 생성


Owner * Repository name *


 ycs-wikim / 2021_oop2 ✓

Great repository names are ... 2021_oop2 is available. ... Need inspiration? How about [bookish-bassoon](#)?

Description (optional)

- 저장소의 권한 설정

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

☐  **Private**
You choose who can see and commit to this repository.



나의 저장소 관리하기 - 3



• 저장소 초기 설정

- Add a README file
 - 저장소 설명 파일 추가 여부 설정
- Add .gitignore
 - 저장소가 관리하지 않을 파일 정보 설정
- Choose a license
 - 저장소의 라이선스 설정

Initialize this repository with:

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

☐ Add a README file
This is where you can write a long description for your project. [Learn more.](#)

☐ Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)

☐ Choose a license
A license tells others what they can and can't do with your code. [Learn more.](#)


Create repository



나의 저장소 관리하기 - 4




• 저장소 정보 확인

 Search or jump to... / Pull requests Issues Marketplace Explore 🔔 + 🌐


📁 ycs-wikim / 2021_oop2 👁 Unwatch 1 ★ Star 0 🍴 Fork 0

<> Code ! Issues 🔗 Pull requests ▶ Actions 📁 Projects 📖 Wiki 🛡 Security 📊 Insights ⚙ Settings


🔗 main 🔗 1 branch 🏷 0 tags Go to file Add file 📄 Code

 ycs-wikim Initial commit 213bfba 3 days ago 🕒 1 commit

📄 .gitignore	Initial commit	3 days ago
📄 LICENSE	Initial commit	3 days ago
📄 README.md	Initial commit	3 days ago

README.md 

2021_oop2

About 

No description, website, or topics provided.

📖 Readme
📄 GPL-3.0 License

Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

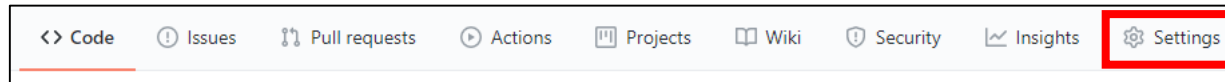


나의 저장소 관리하기 - 5



• 저장소 삭제하기

- 저장소 상단의 "Settings" 메뉴 클릭
- 클릭한 화면 가장 아래 "Danger Zone"에서 삭제 가능



Danger Zone

Change repository visibility

This repository is currently public.

[Change visibility](#)

Transfer ownership

Transfer this repository to another user or to an organization where you have the ability to create repositories.

[Transfer](#)

Archive this repository

Mark this repository as archived and read-only.

[Archive this repository](#)

Delete this repository

Once you delete a repository, there is no going back. Please be certain.

[Delete this repository](#)



저장소 로컬로 가져오기



- 탐색기에서 “git bash here” 실행

- 계정명 : ycs-wikim, 저장소명 : 2021_oop2 이라면

- 실행된 CUI 에서 아래 명령 실행

- **git clone https://github.com/ycs-wikim/2021_oop2.git**

- git clone https://github.com/계정명/저장소명.git 형식
- clone은 복제해서 가져오라는 명령

```
MINGW64:/d/___수업자료/2021/객체지향언어

YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어
$ git clone https://github.com/ycs-wikim/2021_oop2.git
Cloning into '2021_oop2'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (5/5), 12.78 KiB | 12.78 MiB/s, done.

YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어
$
```



git 명령어 동작 가능 위치



- git bash의 상태 정보로 확인 가능
 - 현재 디렉터리 정보 뒤에 (브랜치이름)이 보이는 경우만
 - 아래 그림에서 디렉터리 이동 전에는 (main) 표시가 없음
 - (main)이 나타난 경우에만 git 명령어가 정상적으로 동작

```
MINGW64:/d/___수업자료/2021/객체지향언어/2021_oop2
YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어
$ cd 2021_oop2/
YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어/2021_oop2 (main)
$ |
```



저장소의 상태 정보 확인 - 1



- git status
 - 현재 저장소의 상태 정보를 출력
 - 변경이 없는 경우는 아래와 같이 표시
 - nothing to commit, working tree clean
 - 변경 사항이 없으며, 추가로 처리할 사항이 없음을 나타냄

```
MINGW64:/d/___수업자료/2021/객체지향언어/2021_oop2
YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어/2021_oop2 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어/2021_oop2 (main)
$
```




저장소의 상태 정보 확인 - 2



- 저장소 정보 변경

- touch 명령으로 저장소에 파일을 생성
- 상태 정보에서 관리되지 않는 파일명을 출력하여 알림
 - 붉은 색으로 파일 명을 출력
 - git에 의해 관리되지 않는 파일을 나타냄

```
MINGW64:/d/___수업자료/2021/객체지향언어/2021_oop2
YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어/2021_oop2 (main)
$ touch report.txt

YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어/2021_oop2 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
       report.txt

nothing added to commit but untracked files present (use "git add" to track)
YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어/2021_oop2 (main)
$ |
```



저장소의 상태 정보 수정 - 1



- git add
 - 수정이 발생한 파일을 관리 대상에 포함
 - 관리 대상에 포함한 다음 상태 확인
 - 추가된 새로운 파일이 녹색으로 표시

```
MINGW64:/d/___수업자료/2021/객체지향언어/2021_oop2
YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어/2021_oop2 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
       report.txt

nothing added to commit but untracked files present (use "git add" to track)

YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어/2021_oop2 (main)
$ git add report.txt

YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어/2021_oop2 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
       new file:   report.txt

YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어/2021_oop2 (main)
$
```



저장소의 상태 정보 수정 - 2



- git commit
 - 현재 수정 상태를 저장소에 반영
 - git commit만 입력할 경우 vim 에디터가 실행
 - git commit -m "message" vim 없이 commit 수행

```
MINGW64:/d/___수업자료/2021/객체지향언어/2021_oop2
(use "git add <file>..." to include in what will be committed)
report.txt

nothing added to commit but untracked files present (use "git add" to track)
YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어/2021_oop2 (main)
$ git add report.txt

YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어/2021_oop2 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   report.txt

YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어/2021_oop2 (main)
$ git commit -m "add report file"
[main de1645a] add report file
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 report.txt

YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어/2021_oop2 (main)
$
```



저장소의 상태 정보 수정 - 3



- git push
 - commit 된 정보를 github 서버로 전달
 - 인증 창이 나타날 경우, 인증 토큰을 생성해야 함
 - ID/PWD 방식에서 Personal Access Token으로 인증 방법이 변경되었음

The screenshot shows a Windows terminal window with the title bar 'MINGW64:/d/___수업자료/2021/객체지향언어/2021_oop2'. The terminal output for the command 'git push' is as follows:

```
YUHAN@YP12624115 MINGW64 /d/___수업자료/2021/객체지향언어/2021_oop2 (main)
$ git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 506 bytes | 506.00 KiB/s, done.
Total 5 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To https://github.com/yys-wikim/2021_oop2.git
   eea5b1d..125e0e4  main -> main
```

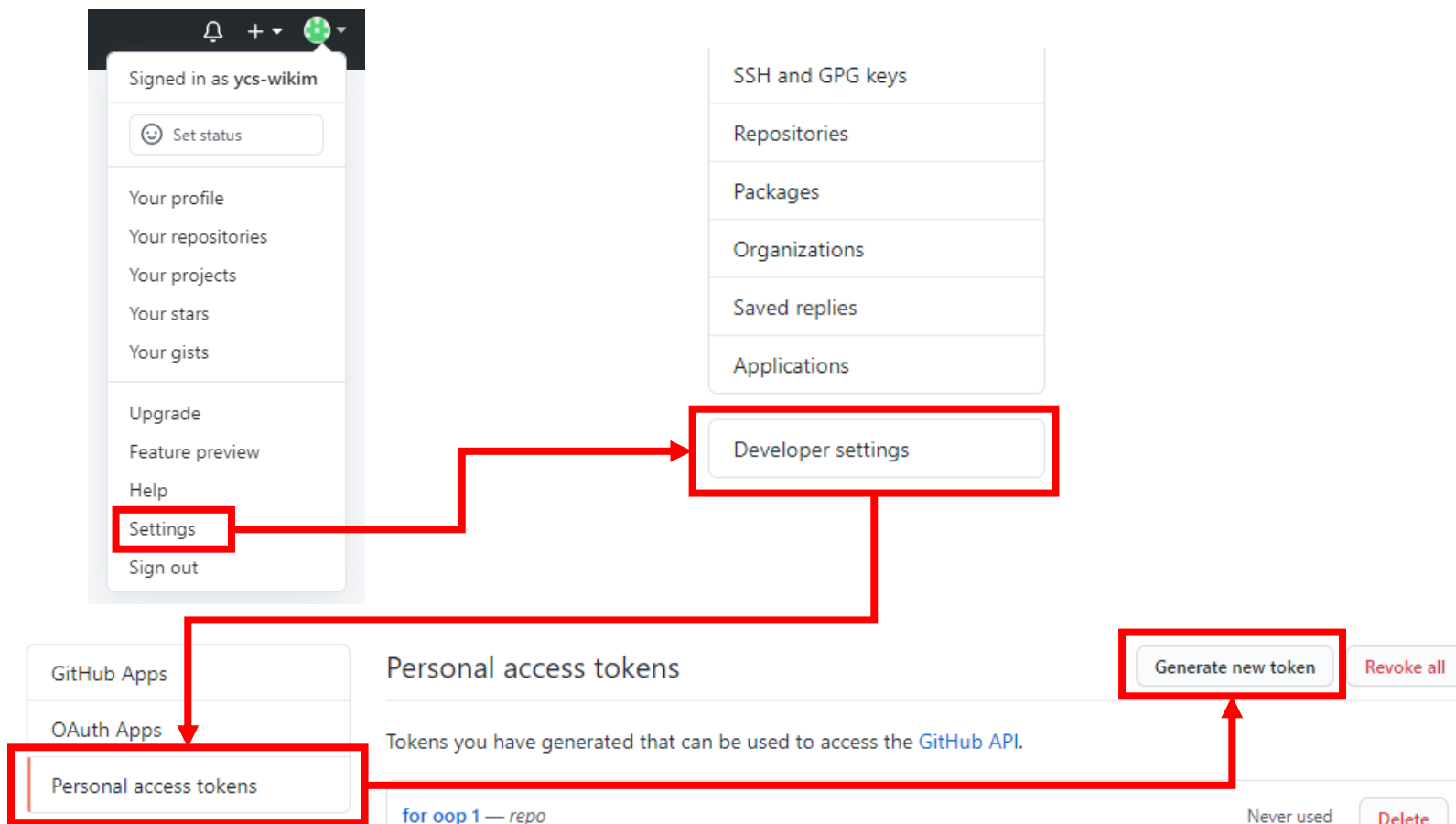
Overlaid on the terminal is a 'Connect to GitHub' dialog box. It features the GitHub logo and the text 'Sign in'. There are two main options: 'Sign in with your browser' (with an external link icon) and a 'Personal Access Token' input field. Below the input field is a 'Sign in' button. At the bottom, there is a link for 'Don't have an account? Sign up'.



github 인증 토큰 생성 - 1

- github 우측 상단의 아이콘 클릭

- “Settings” → “Developer settings” → “Personal access tokens”
→ “Generate new token” 순으로 선택





github 인증 토큰 생성 - 2



- “Note”에 간단한 설명 입력
 - 가장 위 “repo”만 선택하고, 토큰을 생성

New personal access token

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

new token

What's this token for?

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

<input checked="" type="checkbox"/> repo	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo:deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input checked="" type="checkbox"/> repo:invite	Access repository invitations
<input checked="" type="checkbox"/> security_events	Read and write security events

☐ admin:enterprise
Full control of enterprises

☐ manage_billing:enterprise
Read and write enterprise billing data

☐ read:enterprise
Read enterprise profile data

☐ admin:gpg_key
Full control of public user GPG keys ([Developer Preview](#))

☐ write:gpg_key
Write public user GPG keys

☐ read:gpg_key
Read public user GPG keys

Generate token

Cancel



github 인증 토큰 사용




- 생성된 토큰 옆의 아이콘을 클릭
 - 클립보드로 정확한 토큰을 입력
 - 복사해서 붙일 경우, 잘못된 입력이 발생할 수 있음

Personal access tokens


Tokens you have generated that can be used to access the [GitHub API](#).

Make sure to copy your new personal access token now. You won't be able to

✓ 59f951b32d85ff7dc47c7a5ca576c29d785d1228 

Connect to GitHub

GitHub
Sign in

Sign in with your browser 

or

Sign in

Don't have an account? [Sign up](#)



github 인증 토큰 삭제 - 1



- 다중 사용자 환경에서 github 인증서 변경 필요 시
 - "제어판"에서 "자격 증명"을 검색
 - "자격 증명 관리자"를 실행

The screenshot shows the Windows Settings application with the search bar set to '자격 증명' (Credentials). The search results list '자격 증명 관리자' (Credential Manager) as the top result. A red arrow points from this result to the Credential Manager window, which is open in the foreground. The Credential Manager window shows the '자격 증명 관리' (Credential Management) section, with options for '웹 자격 증명' (Web Credentials) and 'Windows 자격 증명' (Windows Credentials). The '웹 자격 증명' section is currently selected, showing a message that no web credentials are stored.

Windows 설정

자격 증명

자격 증명 관리자

사용자 계정 컨트롤 설정 변경

Windows 자격 증명 관리

웹 자격 증명 관리

시스템
디스플레이, 소리, 알림, 전원

전화
Android, iPhone 연결

네트워크 및 인터넷
Wi-Fi, 비행기 모드, VPN

개인 설정
배경, 잠금 화면, 색

자격 증명 관리자

제어판 > 모든 제어판 항목 > 자격 증명 관리자

제어판 홈

자격 증명 관리

웹 사이트, 연결된 응용 프로그램 및 네트워크에 대해 저장된 로그인 정보를 보고 삭제합니다.

웹 자격 증명

Windows 자격 증명

웹 암호

웹 암호가 없습니다.



github 인증 토큰 삭제 - 2

• 자격 증명 토큰 삭제

- “Windows 자격 증명”에서 github 자격 증명을 삭제
- github에서 서버 연결 시 인증 토큰 입력 창이 다시 나타남
- 새로운 토큰 입력 가능

