

-----

**git basic command**

| git --version                      |                            |  |
|------------------------------------|----------------------------|--|
| git                                |                            |  |
| git init                           | to create local repository |  |
|                                    |                            |  |
| git add [filename]                 |                            |  |
| git commit -m "---- --"            |                            |  |
|                                    |                            |  |
| git status                         |                            |  |
|                                    |                            |  |
| git log                            |                            |  |
| git clone                          |                            |  |
| git remote add origin https://---- |                            |  |
| git push origin master             |                            |  |
|                                    |                            |  |
| git branch                         |                            |  |
| git branch [jcr]                   |                            |  |
| git checkout jcr                   |                            |  |
|                                    |                            |  |

**10. pull:**

- git pull --no-rebase origin master
- git pull --rebase origin master

to check remote changes before pulling;

- git fetch origin
- git log head..origin/master --oneline

- git push origin master
- git push origin master --force

**9. branch:**

- git branch
- git branch jcr
- git checkout jcr

#### 8. how to use vim:

normal mode/insert mode/command mode:

- vim readme.txt
- i <-- insert mode
- ?????????????
- escape <--command mode
- :w <--save
- :w [new filename] <-- save as a new filename
- :q <-- quit
- :q! <-- quit by force
- :wq
- :wq!

#### 7. how to download files from a remote repository at github:

- git clone [https://github.com/cherlhee/knu\\_cpp.git](https://github.com/cherlhee/knu_cpp.git) . <--한칸띄고 마침표하기; without new 폴더 생성;
- git clone [https://github.com/cherlhee/knu\\_cpp.git](https://github.com/cherlhee/knu_cpp.git) <-- 새로운 폴더에 저장하기

#### 6. to push to a remote repository at github:

- git remote add origin [https://github.com/cherlhee/knu\\_cpp.git](https://github.com/cherlhee/knu_cpp.git)
- ///login to github site;
- git push origin master
- git push -- force origin master

#### 5. to modify files:

to modify the file of 'readme.txt';

- vim readme.txt

- ///modify

- git add readme.txt

- git commit -m "2nd update"

- git add \* <-- add all files;

- git log

- git checkout [commit number]

- git log

- git checkout [commit number]

- git checkout

#### 4. to add a file:

to create a new file of 'readme.txt';

- vim readme.txt

- git status

- git add readme.txt

- git status

to commit with memo;

- `git commit -m "---- --"`
- `git status`

### 3. to make local repository;

to make a new folder;

- `mkdir git_code`
- `cd git_code`
- `git --version`
- 

to create a local repository;

- `git init`

### 2. install git;

2005, developed by linux tobalz;

to visit the git site;

<http://git-scm.com>

to download latest source release of 2.48.1;

- download 64-bit git for windows setup;

to install;

- @adjusting your path environment; check up; git from the command line and also f

from 3<sup>rd</sup> party software;

to set identity;

caution; should be same with those used at github;

- `git config --global user.name "cherlhee"`
- `git config --global user.email "cherlhee@jcradar.com"`

-

-

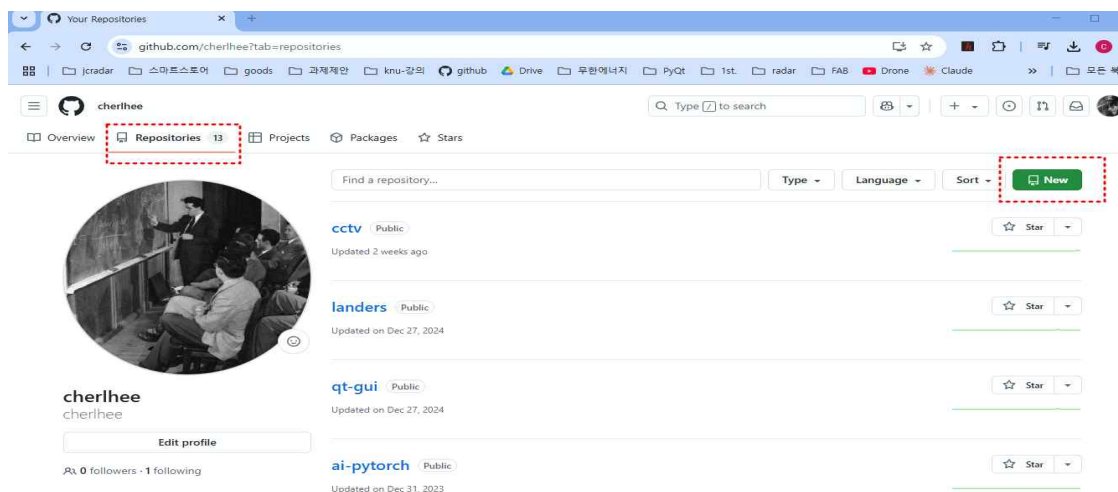
1. to visit the github site and sign up;

<https://github.com>

to create a repository;

//new repository;

-




to create repository

## Create a new repository

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



Required fields are marked with an asterisk (\*).

Owner \*  cherlhee / Repository name \* knu\_cpp  
✓ knu\_cpp is available.

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

Description (optional)

to create projects using cpp and visual cpp

- ☒  **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:

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

Add .gitignore

.gitignore template: None ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license


License: None ▾


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


 You are creating a public repository in your personal account.

Create repository


to copy the remote repository;

 **knu\_cpp** Public

 Pin

 Unwatch **1**


---



### Set up GitHub Copilot

Use GitHub's AI pair programmer to autocomplete suggestions as you code.

[Get started with GitHub Copilot](#)




### Add collaborators to this repository

Search for people using their GitHub username

[Invite collaborators](#)

---

## Quick setup — if you've done this kind of thing before

 Set up in Desktop

 or 

HTTPS

SSH

[https://github.com/cherlhee/knu\\_cpp.git](https://github.com/cherlhee/knu_cpp.git)

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

---

## ...or create a new repository on the command line

```
echo "# knu_cpp" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/cherlhee/knu_cpp.git
git push -u origin main
```