

Installing git

What is Git?

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.

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

So you can track all work processes.



Access to Git official website and download the latest version

<https://git-scm.com>

or

Download from HisNet

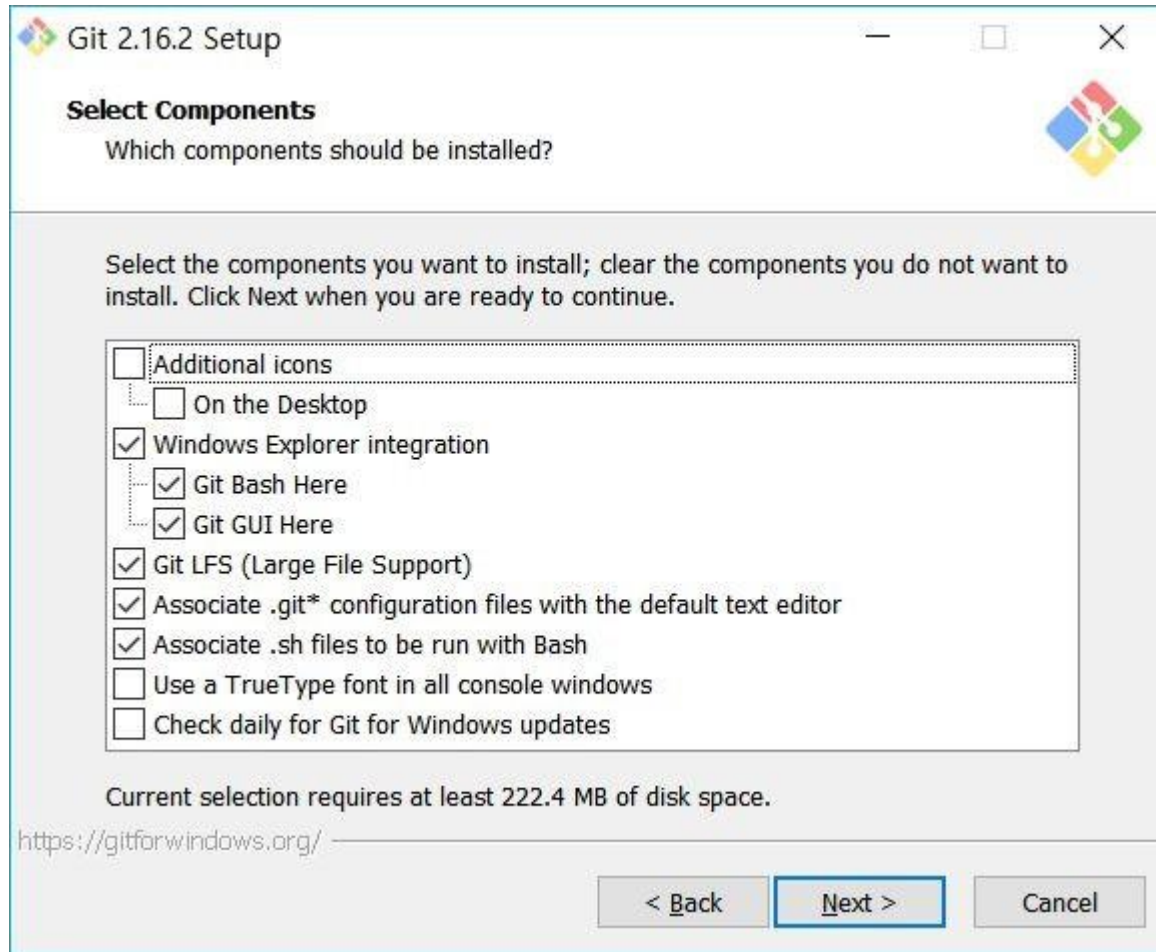
Click.
Then the installation will be
started automatically.



Installing git

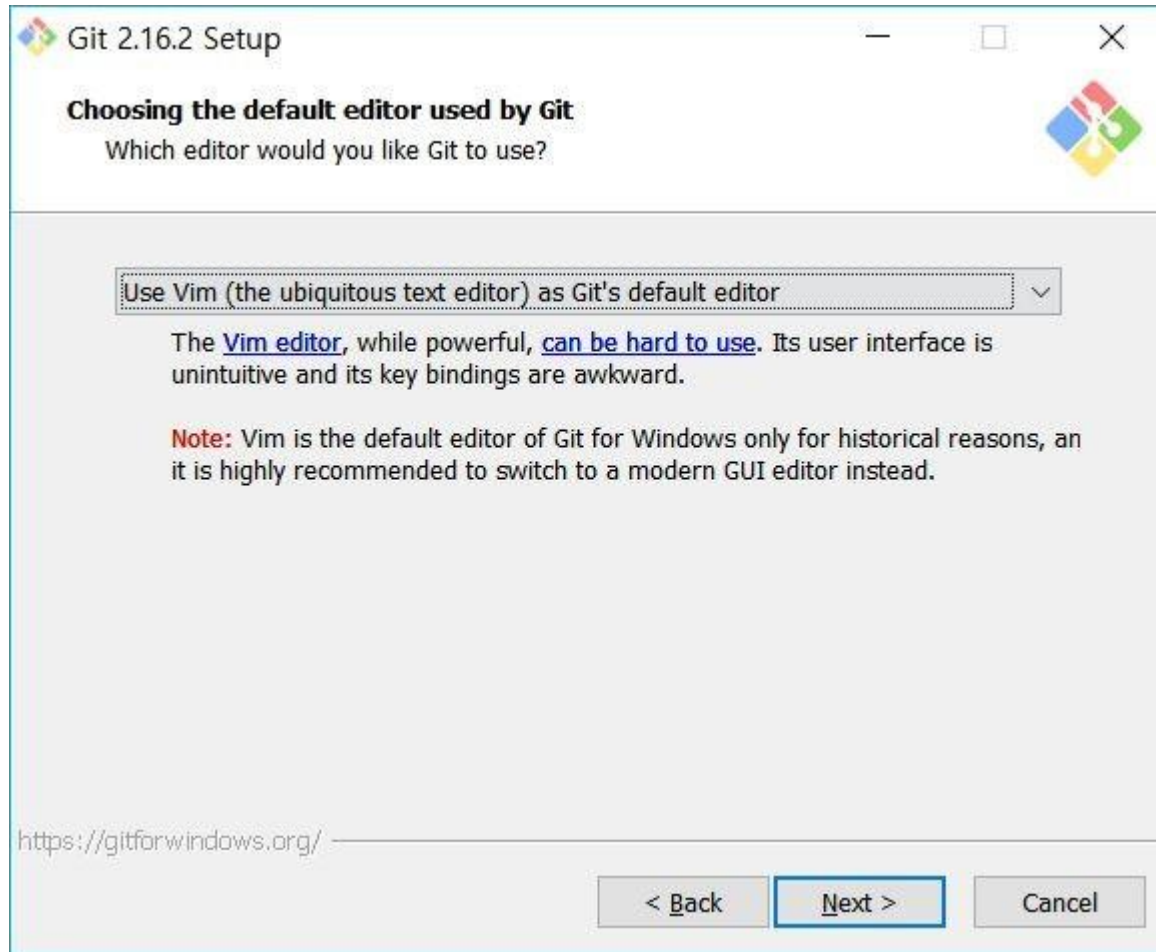


Installing git



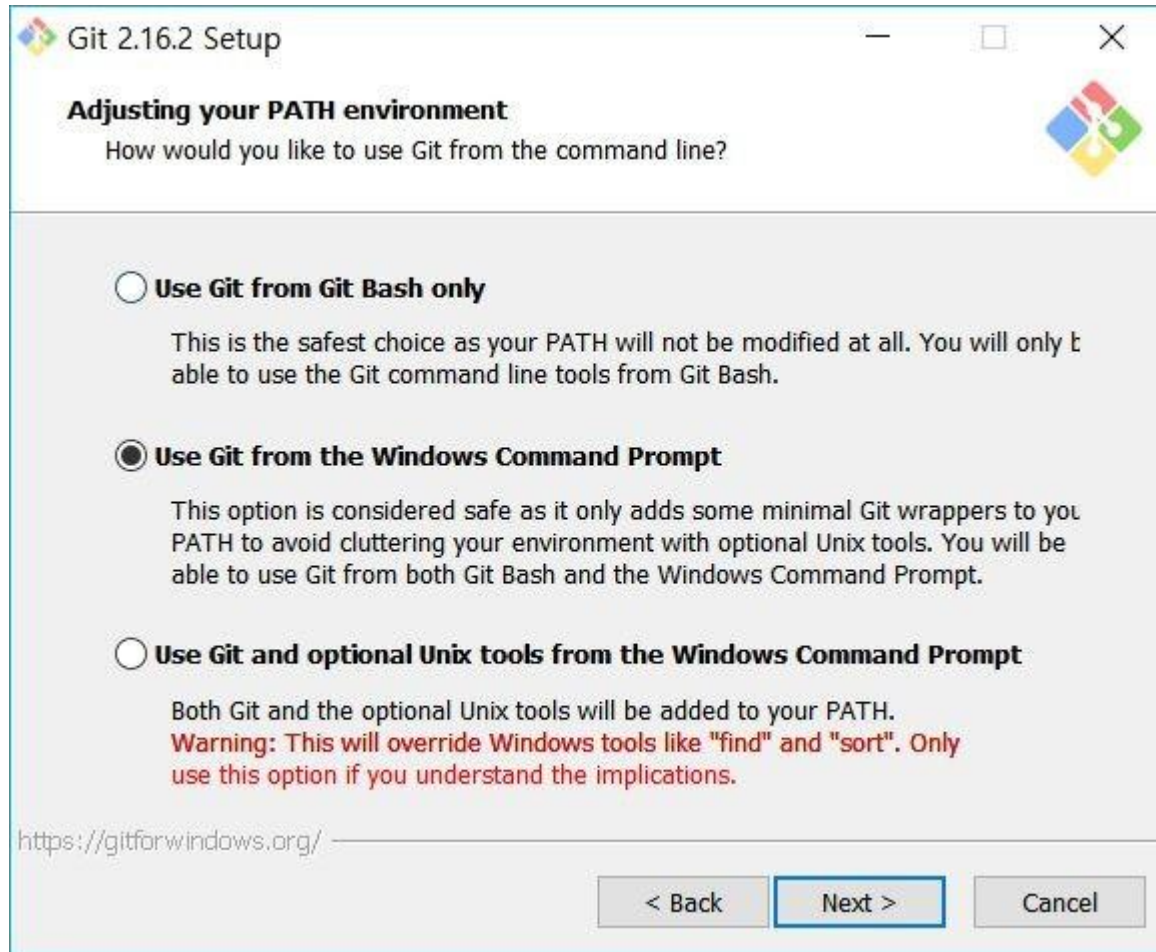
If you want add icons on your Desktop, you can check 'additional icons' and 'On the Desktop'

Installing git



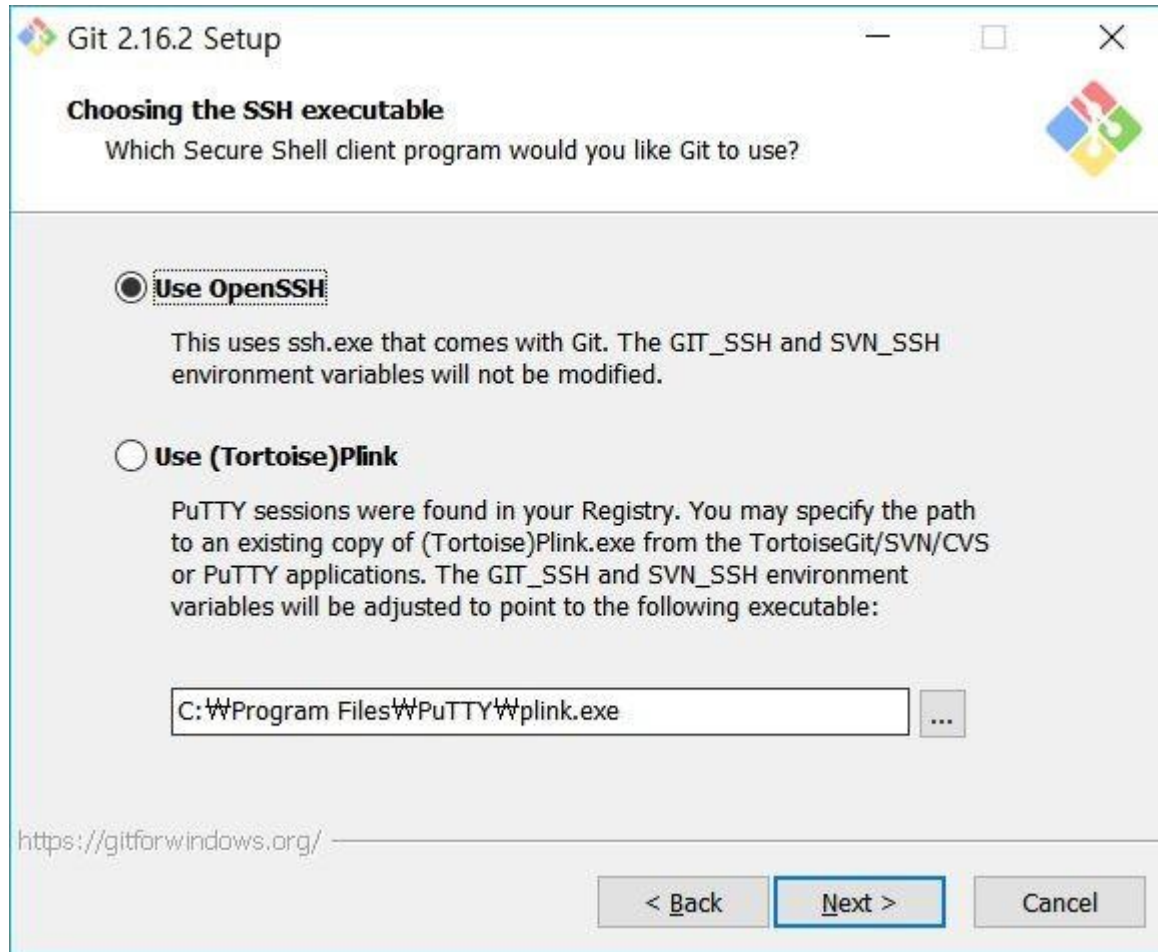
There are other options.
But using Vim is recommended.

Installing git

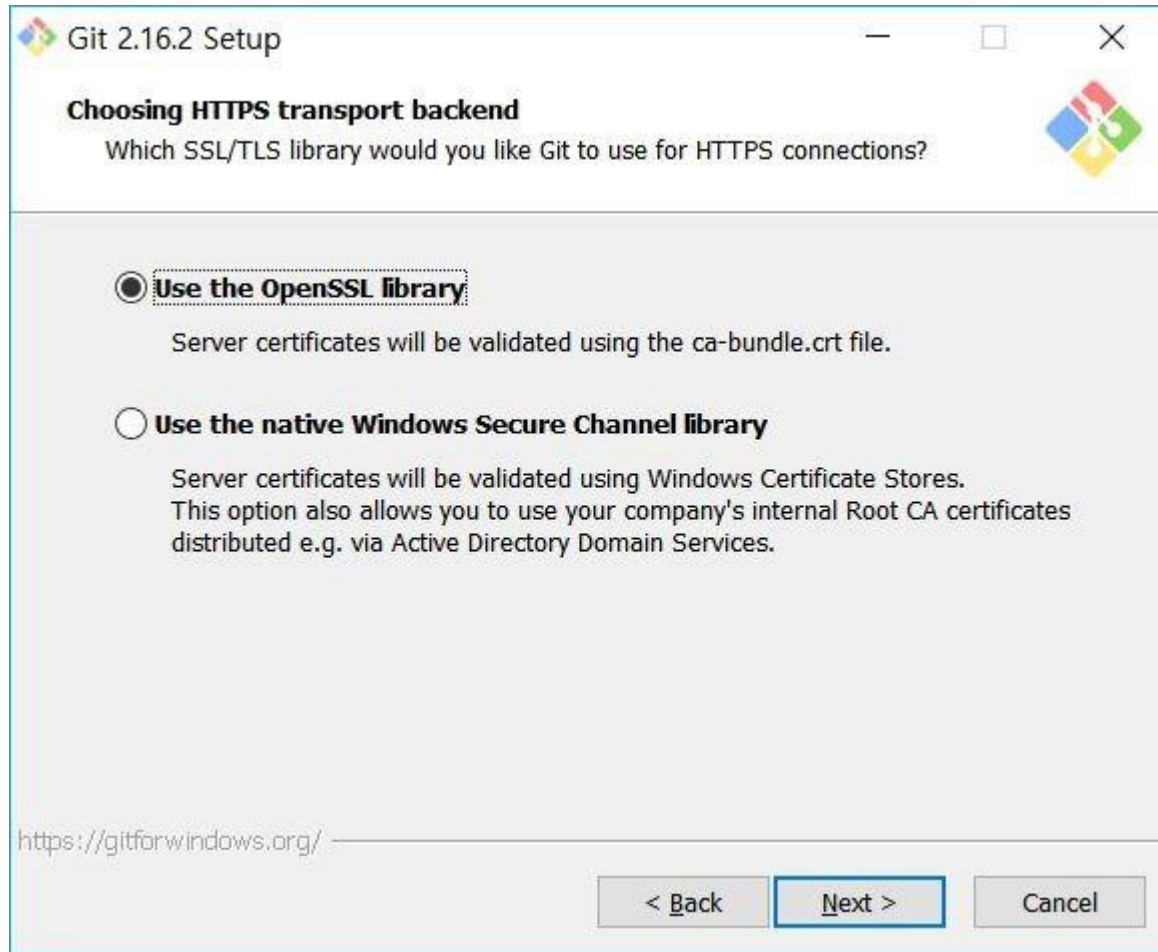


Check
'Use Git from the Windows Command Prompt'

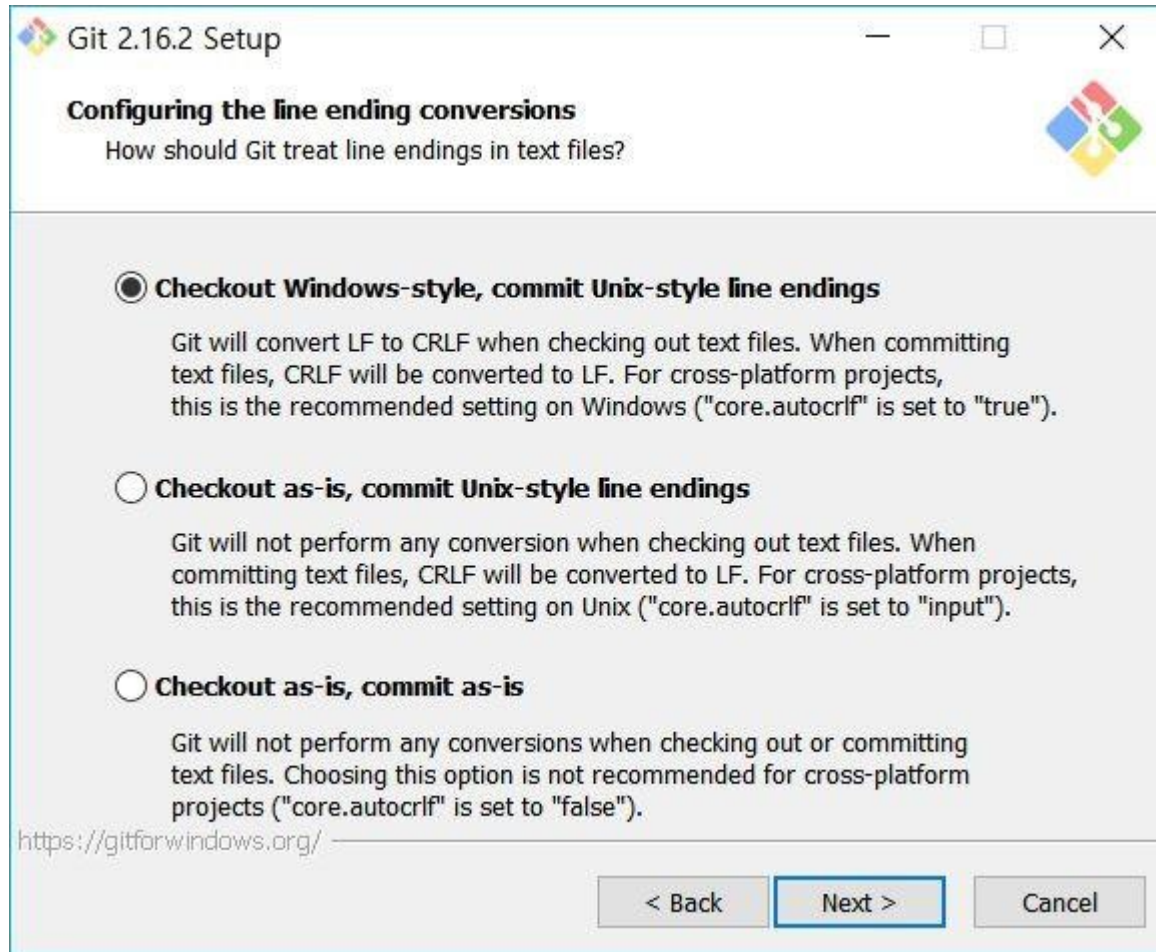
Installing git



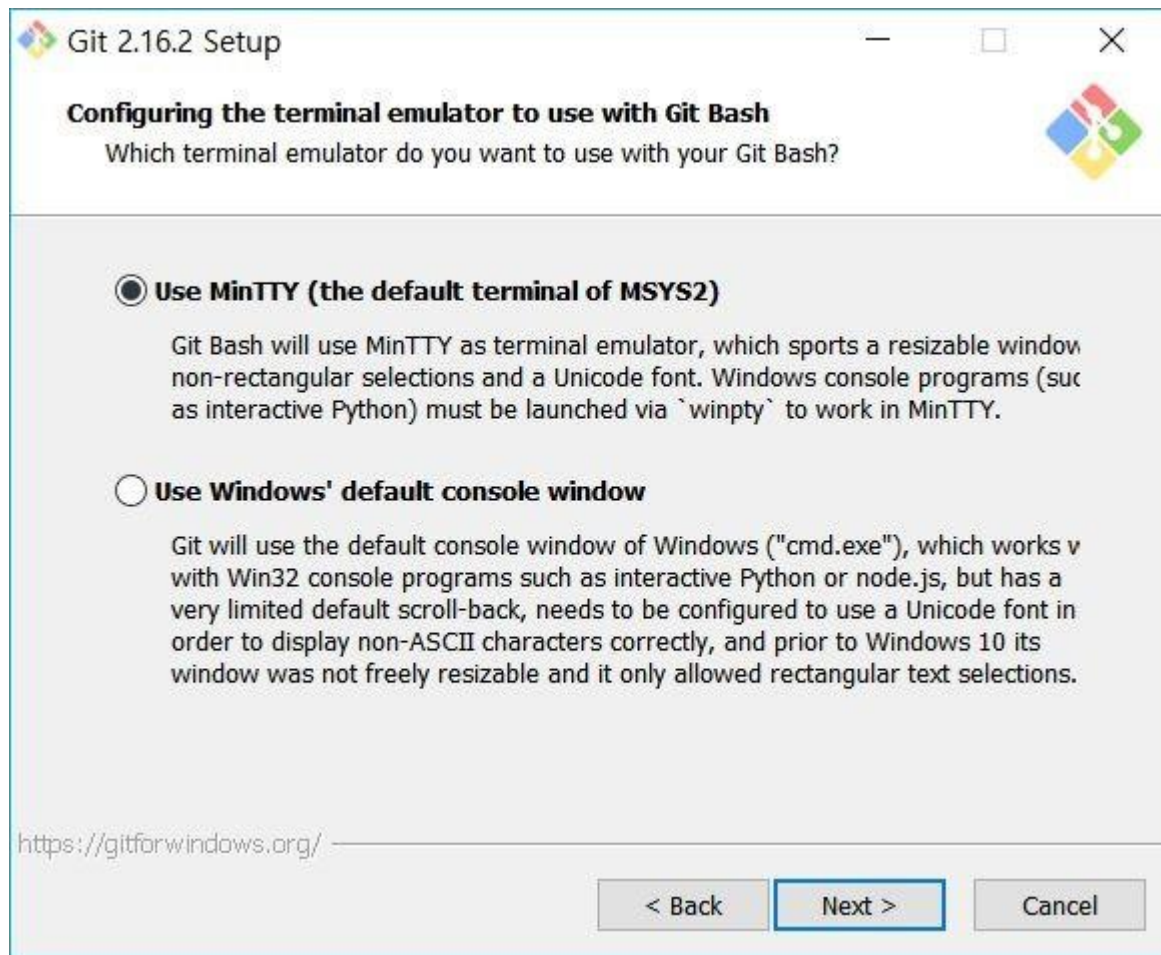
Installing git



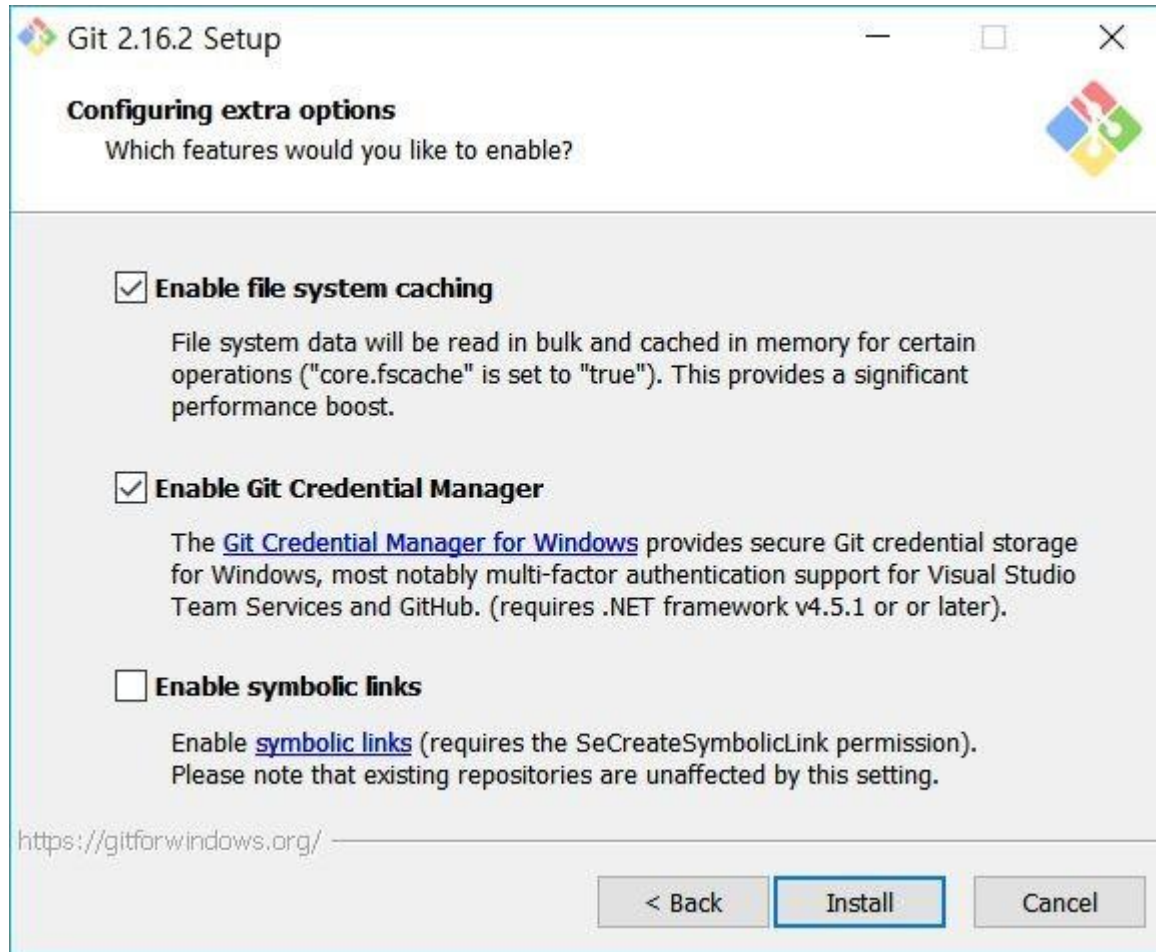
Installing git



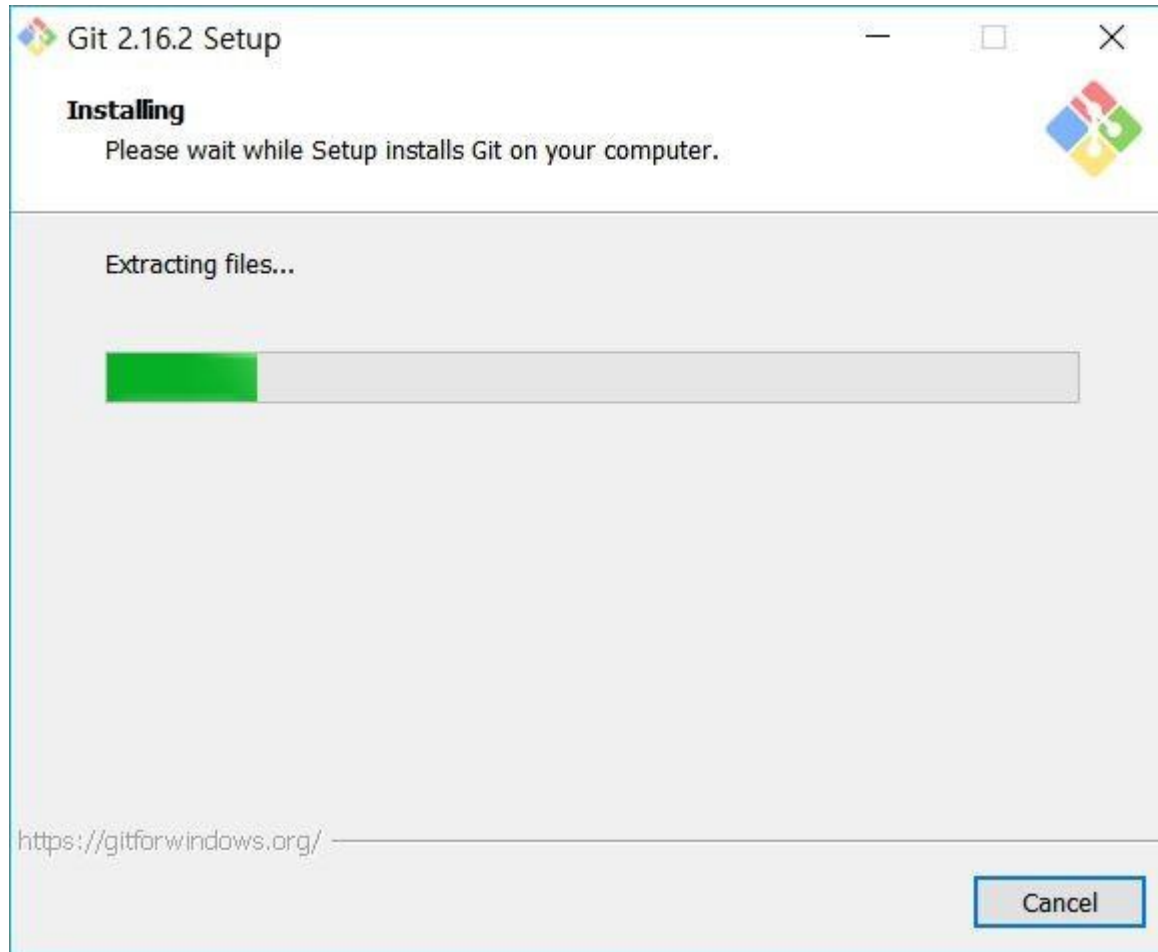
Installing git



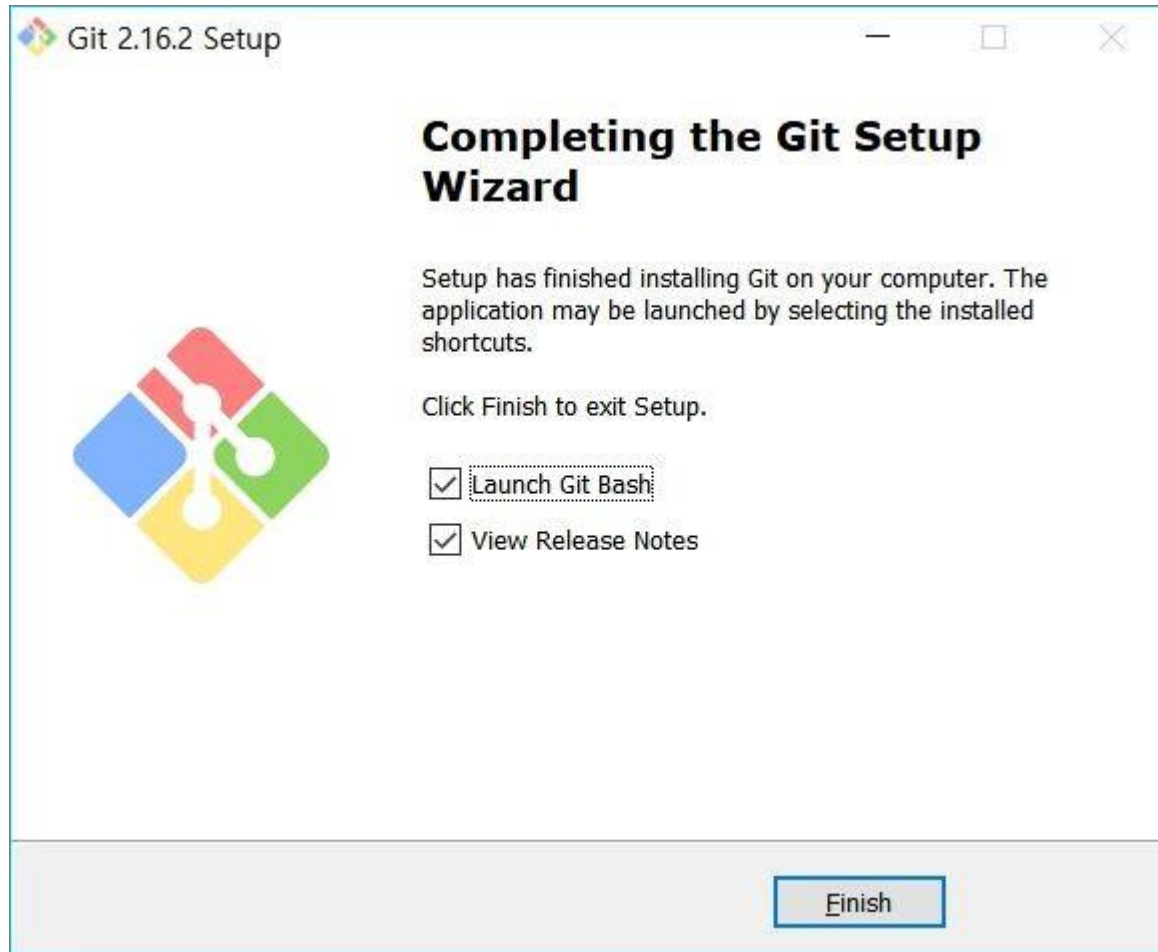
Installing git



Installing git



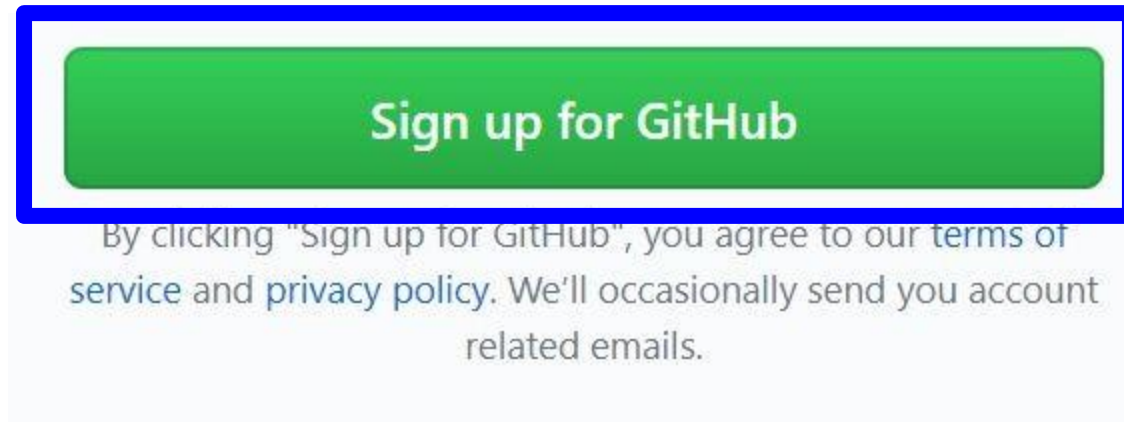
Finished installing Git



Github

Create an account

Connect to <https://github.com/> and click the green button



Create an account

Create your personal account

There were problems creating your account.

Username

enteryourname18



Email address

handongprogrammer@gmail.com



Password

.....

Use at least one lowercase letter, one numeral, and seven characters.

Create an account



Create an account



Completed

Set up a personal account



Step 2:

Choose your plan

Choose your personal plan

☒ Unlimited public repositories for free.

☐ Unlimited private repositories for \$7/month. [\(view in KRW\)](#)

Don't worry, you can cancel or upgrade at any time.

☐ Help me set up an organization next

Organizations are separate from personal accounts and are best suited for businesses who need to manage permissions for many employees.

[Learn more about organizations](#)

☐ Send me updates on GitHub news, offers, and events

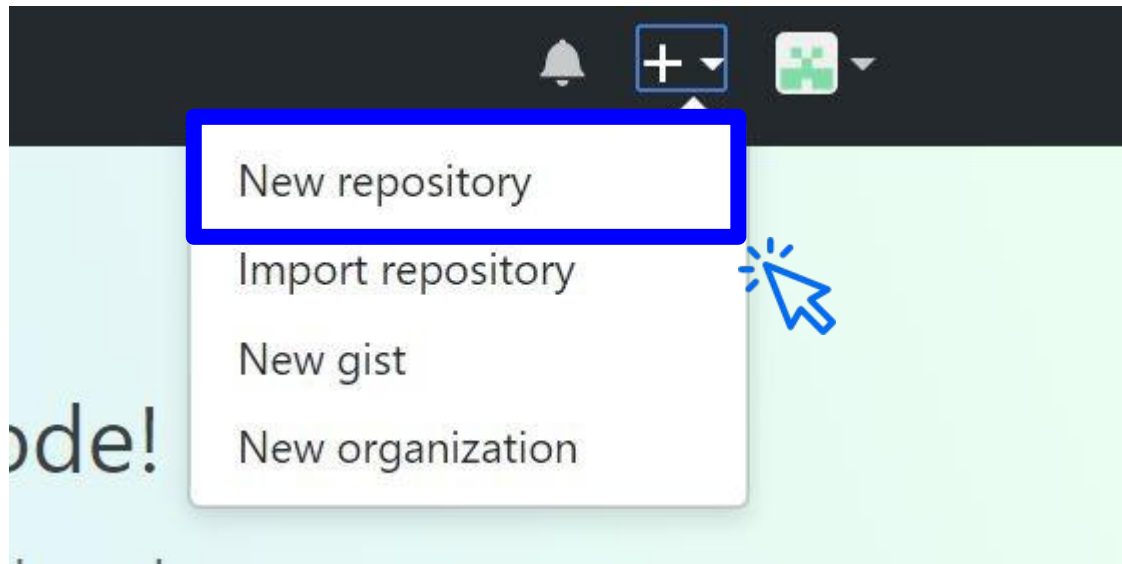
Unsubscribe anytime in your email preferences. [Learn more](#)

Continue



Then, Verify your email address.

Create your remote repository



Create your remote repository (2)

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner  / Repository name

The repository name is **YOUR CHOICE**. If you have to do labs or HWs with a specific repository name, use that name when you create the git repository!! The 'practicegit' is an example name. Do not use it for you labs and HWs. Please aware of what you are doing now.

Great repository names are short and memorable. Need inspiration? How about [cuddly-octo-garbanzo](#).

Description (optional)

to practice fork, clone command.

Create repository

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

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

☐ **Initialize this repository with a README**

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** ▼


Add a license: **None** ▼



Create your remote repository (3)

Now you can check your own remote repository address.

if you've done this kind of thing before

or ☐ HTTPS ☐ SSH 

repository include a [README](#), [LICENSE](#), and [.gitignore](#).

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

```
echo "# Test" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/lifove/Test.git
git push -u origin master
```



...or push an existing repository from the command line

```
git remote add origin https://github.com/lifove/Test.git
git push -u origin master
```



...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

[Import code](#)

Create your local repository: Do this in CMD

```
E:\> mkdir git
```

```
E:\> cd git
```

```
E:\git> mkdir Test
```

```
E:\git> cd Test
```

```
E:\git\Test> echo "# Test" >> README.md
```

```
E:\git\Test> git init
```

Initialized empty Git repository in E:/git/Test/.git/

```
E:\git\Test> git add README.md
```

```
E:\git\Test> git commit -m "first commit"
```

*** Please tell me who you are.

Run

```
git config --global user.email "you@example.com"
```

```
git config --global user.name "Your Name"
```

to set your account's default identity.

Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'user@DESKTOP-OJ184-A.(none)')

```
E:\git\Test> git config --global user.email "your@email.address"
```

```
E:\git\Test> git config --global user.name "Your Name" E:\git\Te
```

```
st> git commit -m "first commit"
```

<< Create your project directory. In this example, the directory name is "Test".
<< Change directory to "Test". Please use the git repository name you created in Github. The project name is also **YOUR CHOICE**.

Create README.md with your project name "# [your project name]". In fact you can

If you get a fatal error like this, set the email and name as follows.
Then commit again! No error then skip these steps



Create your local repository (2)


```
E:\git\Test> git commit -m "first commit" [  
master (root-commit) fc0762f] first commit  
1 file changed, 1 insertion(+)  
create mode 100644 README.md
```


```
E:\git\Test> git remote add origin https://github.com/[YOUR]/[GIT_ADDR].git
```

```
E:\git\Test> git push -u origin master
```

 GitHub Login ✕

GitHub Login

 Login

 Cancel

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

[Forgot your password?](#)

Create your local repository (3)

```
E:\git\Test> git commit -m "first commit" [  
master (root-commit) fc0762f] first commit  
1 file changed, 1 insertion(+)  
create mode 100644 README.md
```

```
E:\git\Test> git remote add origin https://github.com/[YOUR]/[GIT_ADDR].git
```

```
E:\git\Test> git push -u origin master
```

Counting objects: 3, done.

Writing objects: 100% (3/3), 213 bytes | 213.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0)

To https://github.com/lifove/Test.git

* [new branch] master -> master

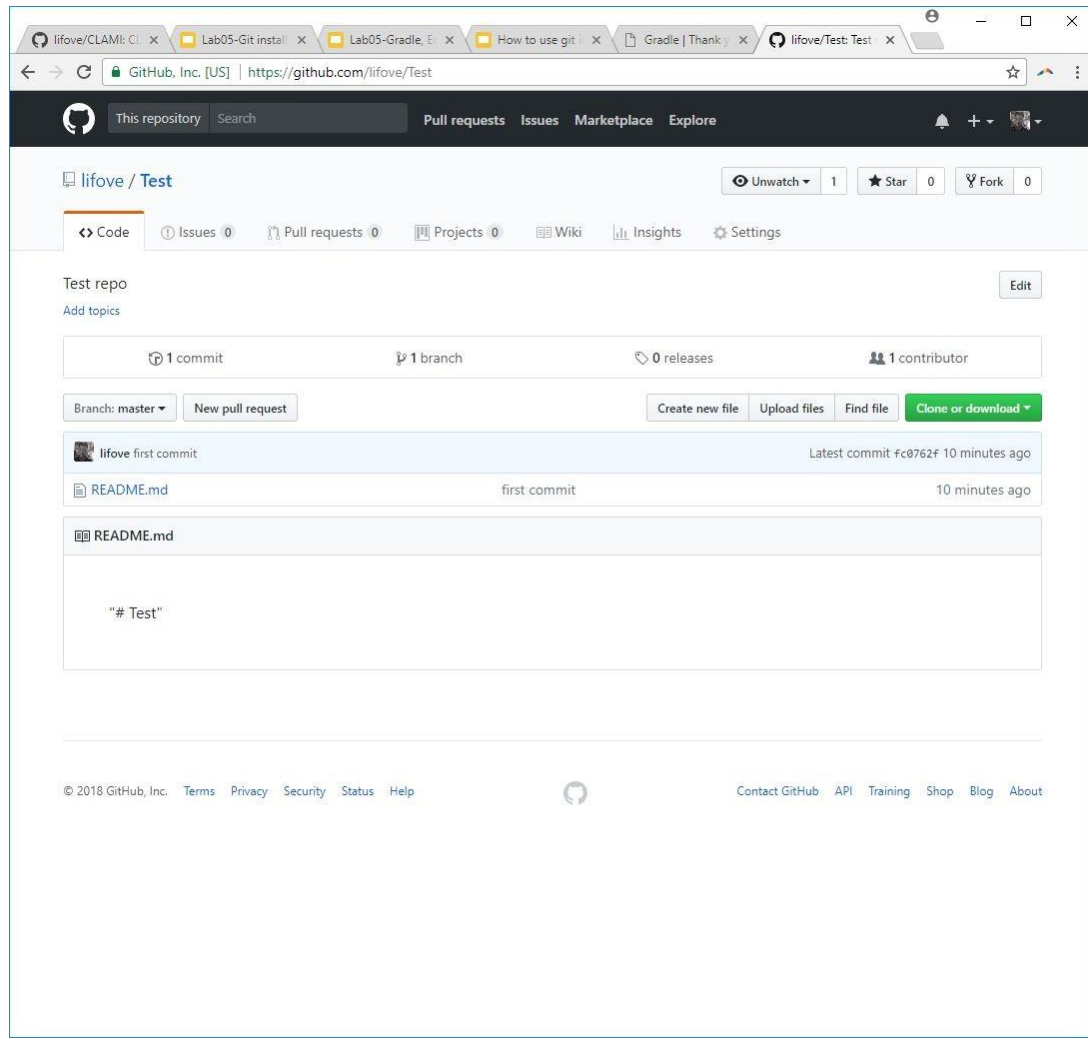
Branch 'master' set up to track remote branch 'master' from 'origin'.

```
E:\git\Test>
```

Now, you've pushed your local git into
the remote git in Github!!

Create your local repository (4)

- Check what happens in Github



Fork

(Copy a repository from other developers)

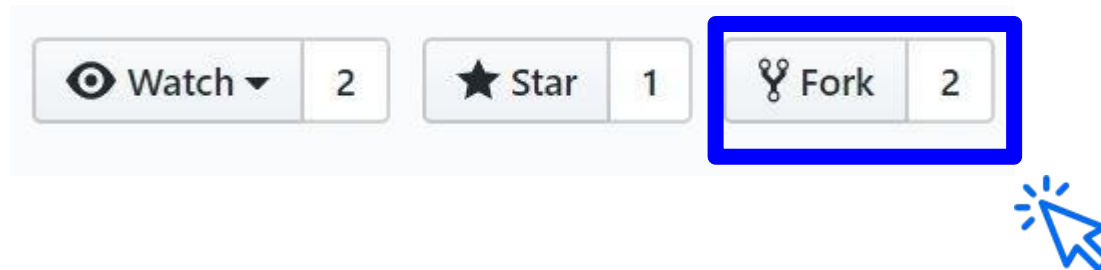
Not pork....

Forking a repository

You can **fork** to bring some source codes from other repository into your repository, without affecting the original repository.

We are going to fork this repository. ↓

<https://github.com/idebtor/HuStar-ML>



Forking a repository - result

The screenshot shows a GitHub repository page for a user's fork of 'CLAMI' from the original repository 'lifove/CLAMI'. The repository is titled 'CLAMI' and has a description 'CLAMI: an unsupervised defect prediction technique'. It shows 85 commits, 3 branches, 2 releases, 1 contributor, and is licensed under GPL-3.0. The current branch is 'develop'. The file list includes 'data', 'gradle/wrapper', 'src', '.gitignore', 'LICENSE', 'README.md', 'build.gradle', 'gradlew', and 'gradlew.bat'. The latest commit is by 'lifove' with the message 'Add an option to specify a Weka classifier for CLAMI' on 15 Feb 2017.

CLAMI
forked from lifove/CLAMI

Watch 0 Star 0 Fork 3

Code Pull requests 0 Projects 0 Wiki Insights Settings

CLAMI: an unsupervised defect prediction technique [Edit](#)

[Add topics](#)

85 commits 3 branches 2 releases 1 contributor GPL-3.0

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

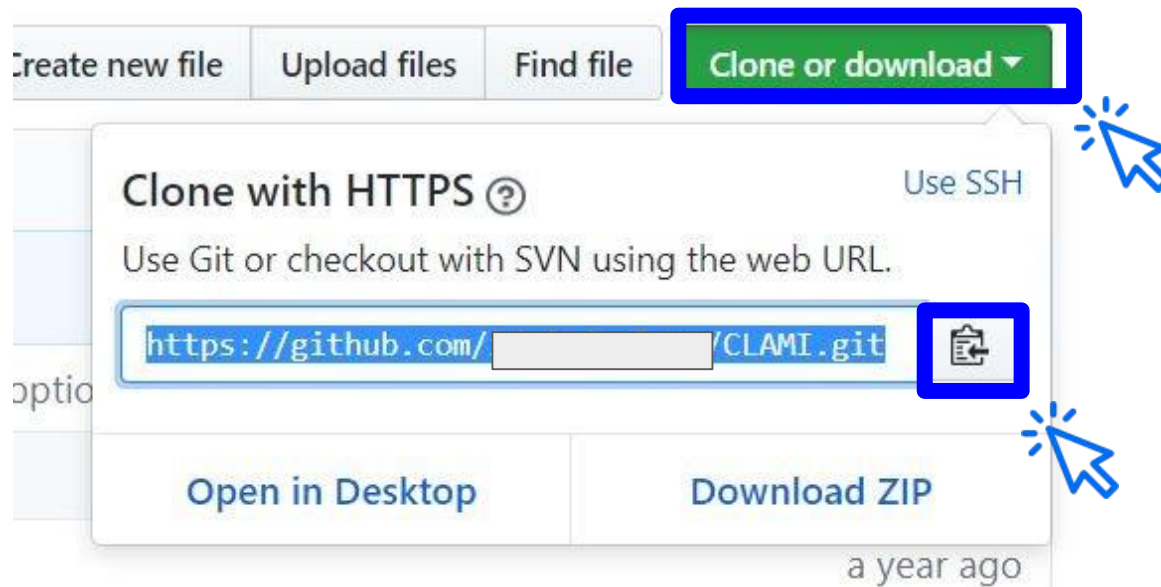
This branch is even with lifove:develop. [Pull request](#) [Compare](#)

lifove Add an option to specify a Weka classifier for CLAMI Latest commit 2d9b49f on 15 Feb 2017

data	Add printing prediction performance and implementing suppress option for	2 years ago
gradle/wrapper	gradle and project setting	2 years ago
src	Add an option to specify a Weka classifier for CLAMI	a year ago
.gitignore	Update .gitignore	2 years ago
LICENSE	Initial commit	2 years ago
README.md	Update README.md	2 years ago
build.gradle	Update build.gradle for the google guava library	2 years ago
gradlew	gradle and project setting	2 years ago
gradlew.bat	gradle and project setting	2 years ago

Cloning a repository - GUI

1. Go to your remote repository



2. And copy HTTPS

Cloning a repository - command

1. Search 'git bash' on your computer and click
2. Make sample directory : mkdir [directory_name]

```
정 찬 미 @chanshance MINGW64 ~  
$ mkdir cloningsample2
```

3. Change directory(in this case, cloningsample2): cd [directory_name]

```
정 찬 미 @chanshance MINGW64 ~  
$ cd cloningsample2/
```

4. Cloning a repository: git clone [https_address]

```
정 찬 미 @chanshance MINGW64 ~/cloningsample2  
$ git clone https://github.com/CLAMI/CLAMI.git  
Cloning into 'CLAMI'...  
remote: Counting objects: 831, done.  
remote: Total 831 (delta 0), reused 0 (delta 0), pack-reused 831  
Receiving objects: 100% (831/831), 195.77 KiB | 346.00 KiB/s, done.  
Resolving deltas: 100% (231/231), done.
```

Please check that the operation completed successfully by checking the directory.

Other useful comments in git bash

\$ ls // see the list of files in the current directory

\$ cd cloningsample2 // move to the directory, cloningsample2 in my current directory

\$ git log // see the commit history. press 'q' to exit

Questions

1. Fork idebtor/HuStar-ML git repository
2. Add files or folder (read.md or anything) and then type “git pull” to get recent resources