



Git, Github 기초 사용법 정리



목 차

- 01 사전준비
- 02 Git 기본 사용법
- 03 Github 협업
- 04 주의할 점

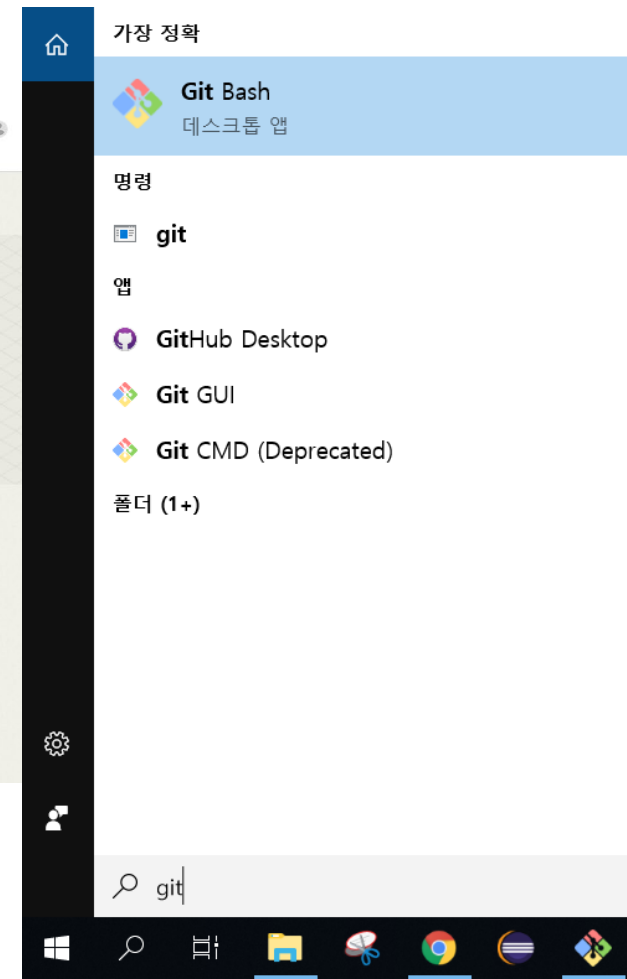
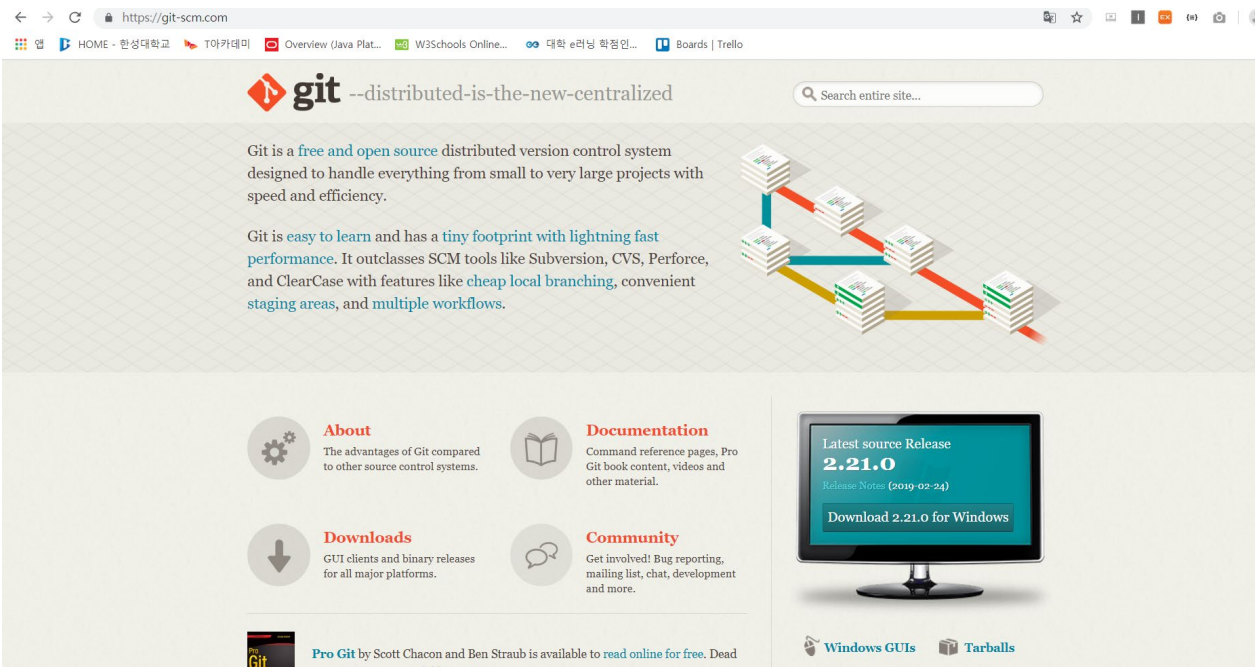
01

사전준비

Git 설치 -> 아래 url 들어가서 Download하면 됨

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

설치 후 Git Bash 라는 프로그램이 생성되면 제대로 설치한 것!!!!



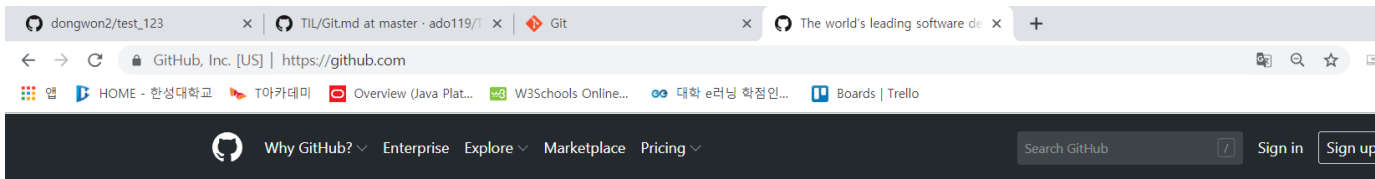
01

사전준비

Github 아이디 가입

<https://github.com/>

추후 username, email, password 다 필요하니 까먹지 않을 것!!!



Build better together

From [open source](#) to [business](#), you can host and review code, manage projects, and build software alongside 31 million developers.

For Individuals & Teams

Username

Email

Password

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

Sign up for free

For Enterprises

Get started with Enterprise

Take collaboration to the next level with security and administrative features built for businesses.

Start a free trial of Enterprise Server →

Get started with Enterprise Cloud →

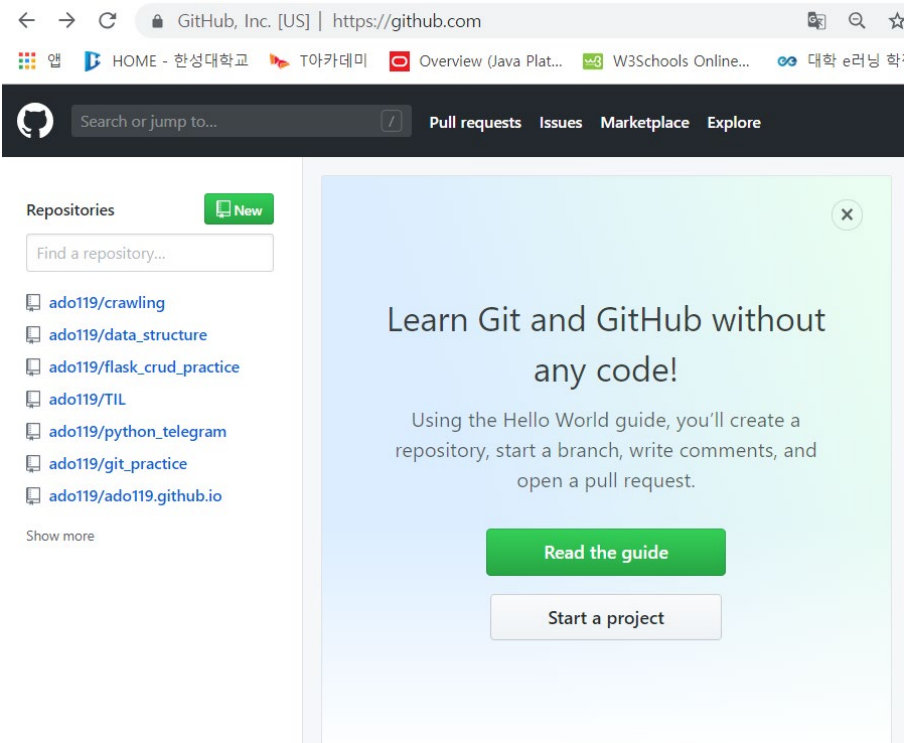
Request a demo

Made by DSUAL

02

Git 기본 사용법

Start a project -> 클릭 후 Repository name에 원하는 name 작성 후 Create repository 클릭 하면 생성됨



The screenshot shows the GitHub homepage. At the top, there's a navigation bar with the GitHub logo and links for Pull requests, Issues, Marketplace, and Explore. Below this, there's a search bar and a list of repositories. A large blue box in the center contains the text "Learn Git and GitHub without any code!" and a green button labeled "Read the guide". Below this, there's a button labeled "Start a project".



Create a new repository

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

Owner: ado119 / Repository name:

Great repository names are short and memorable. Need inspiration? How about **cuddly-enigma**?

Description (optional):

☒ 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** ⓘ

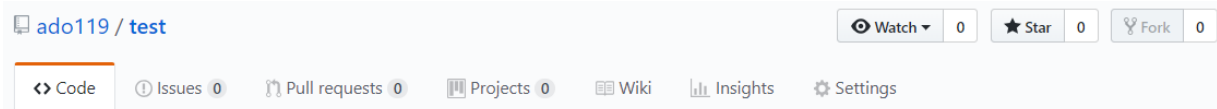
02

Git 기본 사용법


밑의 사진과 같은 화면이 나옵니다.

...or push an existing repository from the command line 밑에 있는 2줄이 중요합니다.

저 화면이 나왔다면 github에 올리고 싶은 파일들이 있는 폴더로 갑니다.



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

 Set up in Desktop or ☐ HTTPS ☐ SSH

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 "# test" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/ado119/test.git
git push -u origin master
```

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

```
git remote add origin https://github.com/ado119/test.git
git push -u origin master
```

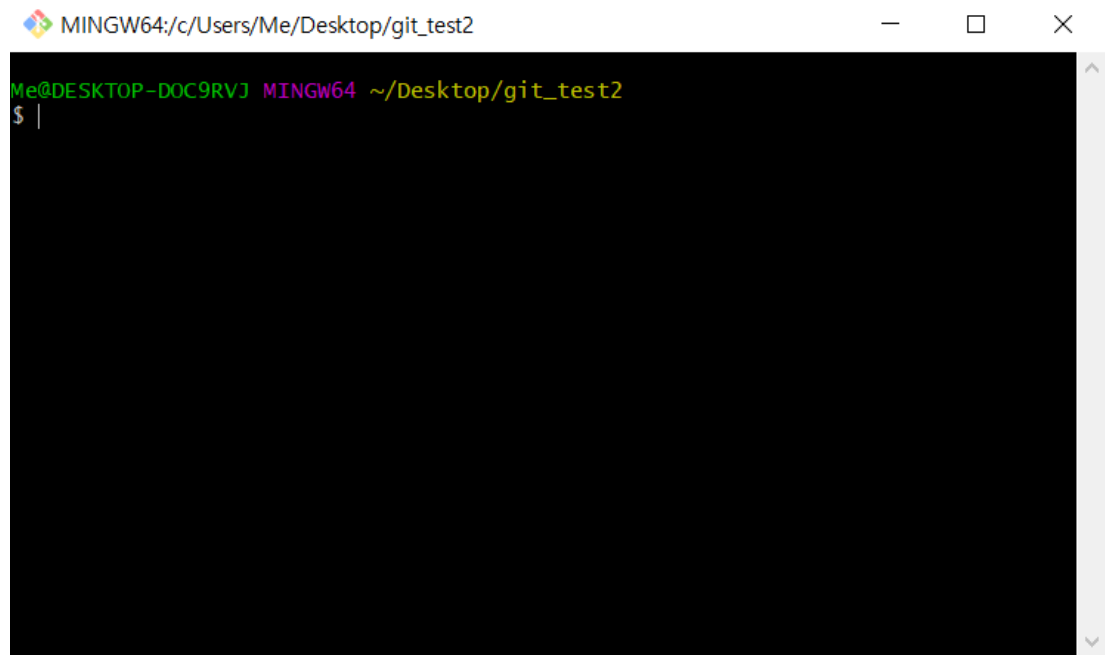
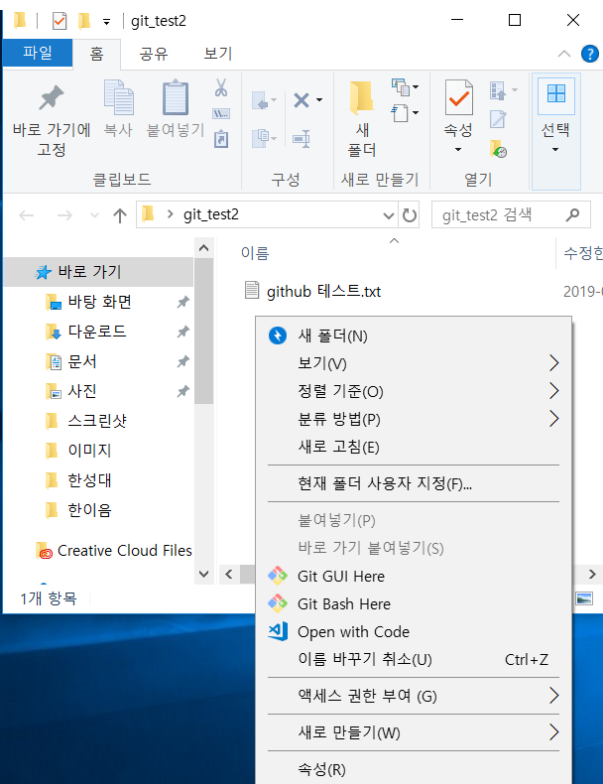
...or import code from another repository

02

Git 기본 사용법

원하는 폴더로 이동 후 우 클릭 후 Git Bash Here 클릭 -> 아래 이미지와 같은 CLI가 나옵니다.

CLI를 통해 폴더에 있는 데이터들을 github에 올리고 내려 받을 것입니다.

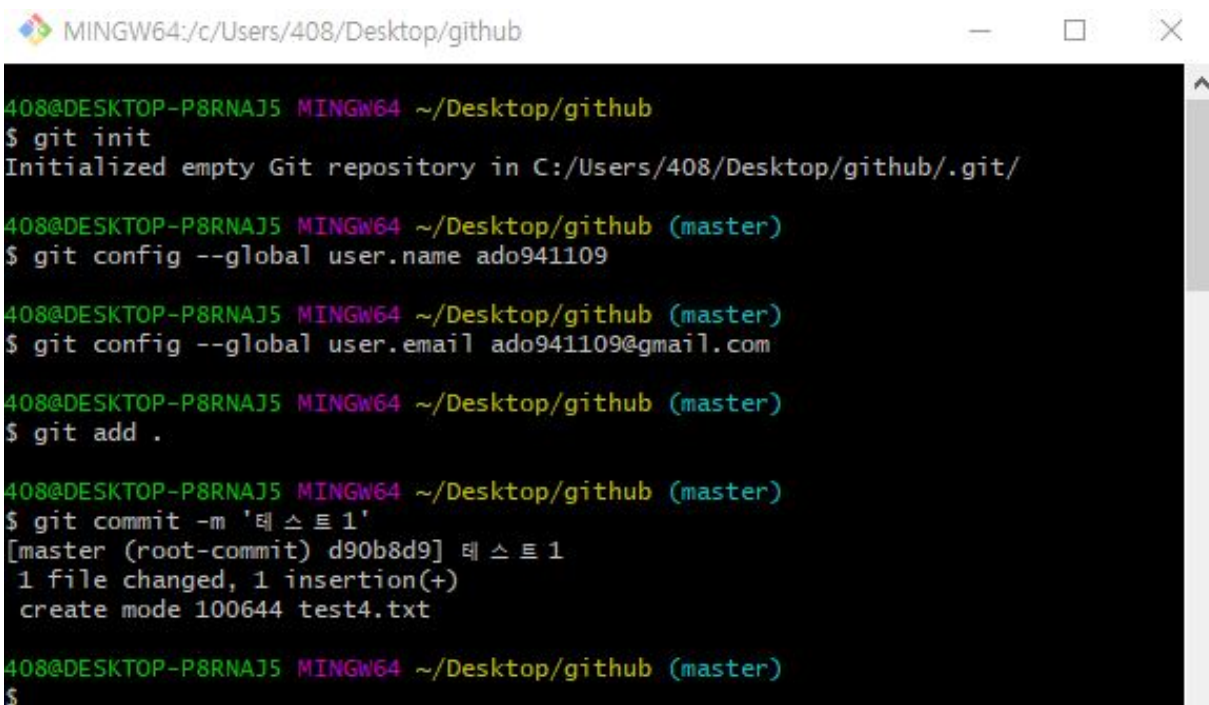


02

Git 기본 사용법

기본 명령어 소개

1. `git init` : git 저장소를 생성하는 것입니다. (숨김항목 볼 수 있도록 체크해주면 `.git` 이라는 폴더가 생기는것을 확인 가능)
2. `git config --global user.name '본인 username'`
`git config --global user.email '본인 user email'`
위의 명령어를 각각 한번 씩 작성합니다. 본인 소유의 OS를 사용하면 한번만 하면 됩니다.



```
MINGW64:/c/Users/408/Desktop/github
408@DESKTOP-P8RNAJ5 MINGW64 ~/Desktop/github
$ git init
Initialized empty Git repository in C:/Users/408/Desktop/github/.git/
408@DESKTOP-P8RNAJ5 MINGW64 ~/Desktop/github (master)
$ git config --global user.name ado941109
408@DESKTOP-P8RNAJ5 MINGW64 ~/Desktop/github (master)
$ git config --global user.email ado941109@gmail.com
408@DESKTOP-P8RNAJ5 MINGW64 ~/Desktop/github (master)
$ git add .
408@DESKTOP-P8RNAJ5 MINGW64 ~/Desktop/github (master)
$ git commit -m '테스트 1'
[master (root-commit) d90b8d9] 테스트 1
1 file changed, 1 insertion(+)
create mode 100644 test4.txt
408@DESKTOP-P8RNAJ5 MINGW64 ~/Desktop/github (master)
$
```


02

Git 기본 사용법

기본 명령어 소개

3. `git add .` : `git add`는 현재 `working tree`에서 `commit`할 목록을 담아 놓는 것입니다.

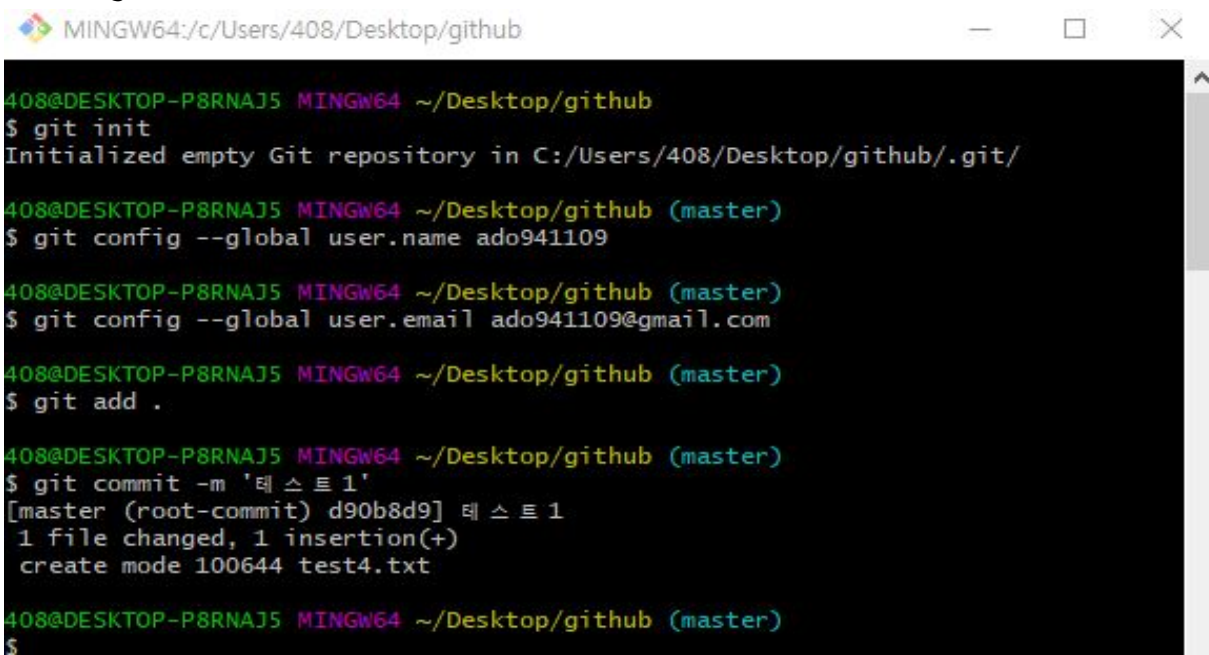
현재 디렉토리의 파일 모두 추가할 것이면 위와 같이 `git add .` 을 치면 됩니다.

특정 파일을 추가하고 싶으면 `git add '특정 파일명'` ex) `git add test.txt`

4. `git commit`은 현재 소스코드의 이력을 남기는 것으로, 현재 커밋할 목록에 담겨 있는 파일을 기록하는 것입니다.

일종의 현재 시점의 스냅샷을 찍는 것

`git commit -m '커밋메세지 입력'`



```
MINGW64:/c/Users/408/Desktop/github
408@DESKTOP-P8RNAJ5 MINGW64 ~/Desktop/github
$ git init
Initialized empty Git repository in C:/Users/408/Desktop/github/.git/

408@DESKTOP-P8RNAJ5 MINGW64 ~/Desktop/github (master)
$ git config --global user.name ado941109

408@DESKTOP-P8RNAJ5 MINGW64 ~/Desktop/github (master)
$ git config --global user.email ado941109@gmail.com

408@DESKTOP-P8RNAJ5 MINGW64 ~/Desktop/github (master)
$ git add .

408@DESKTOP-P8RNAJ5 MINGW64 ~/Desktop/github (master)
$ git commit -m '테스트 1'
[master (root-commit) d90b8d9] 테스트 1
1 file changed, 1 insertion(+)
create mode 100644 test4.txt

408@DESKTOP-P8RNAJ5 MINGW64 ~/Desktop/github (master)
$
```

02

Git 기본 사용법

기본 명령어 소개

5. Commit 후 중요하다고 했던 2줄을 각각 입력해줍니다.

git bash에 붙여넣기는 shift + insert 를 하면 됩니다.

6. 그 후 github 가서 F5 눌러주면 현재 디렉토리에 있는 파일들이 업로드 된 것을 확인할 수 있습니다.

7. 그 후 디렉토리의 변경 사항이 있으면 git add -> git commit -m '메시지' -> git push origin master
(동일 디렉토리일 경우입니다.)

8. git push origin master 설정해둔 원격 저장소로 commit한 것을 보내는 명령어

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

```
git remote add origin https://github.com/ado119/test.git  
git push -u origin master
```




03

Github 협업

협업을 하기 위해서는 Collaborators 추가가 필요합니다.

프로젝트를 생성한 관리자가 협력자를 추가할 수 있습니다.

추가를 하고 추가를 받은 인원이 승인을 해야 pull과 push를 할 수 있는 권한이 생깁니다.

 dongwon2 / test_123

Watch 0

Star 0

Fork 0

<> Code

! Issues 0

Pull requests 0

Projects 0

Wiki

Insights

Settings

Options

Collaborators

Branches

Webhooks

Notifications

Integrations & services


Deploy keys

Moderation

Interaction limits

Collaborators

Push access to the repository

 DoYongJu

X

Search by username, full name or email address

You'll only be able to find a GitHub user by their email address if they've chosen to list it publicly. Otherwise, use their username instead.

Add collaborator

03

Github 협업

Github 협업하기

받아올 github으로 이동 후 clone or download 클릭 후 나온 url 복사
그 후 내려받을 디렉터리로 이동 후 git bash 키기

ado119 / project

Watch 0 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

No description, website, or topics provided. Edit

Manage topics

1 commit 1 branch 0 releases 1 contributor

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

ado119 프로젝트 정리

Notepad(2018년 2학기 웹앱)

프로젝트

android_app(2018년 2학기 과제)

프로젝트

project_multicam

프로젝트

Clone with HTTPS ? Use SSH

Use Git or checkout with SVN using the web URL.

https://github.com/ado119/project.git

Open in Desktop Download ZIP

Help people interested in this repository understand your project by adding a README. Add a README

03

Github 협업

0. git init

git init으로 저장소 생성을 하는 것이 우선

1. git clone :

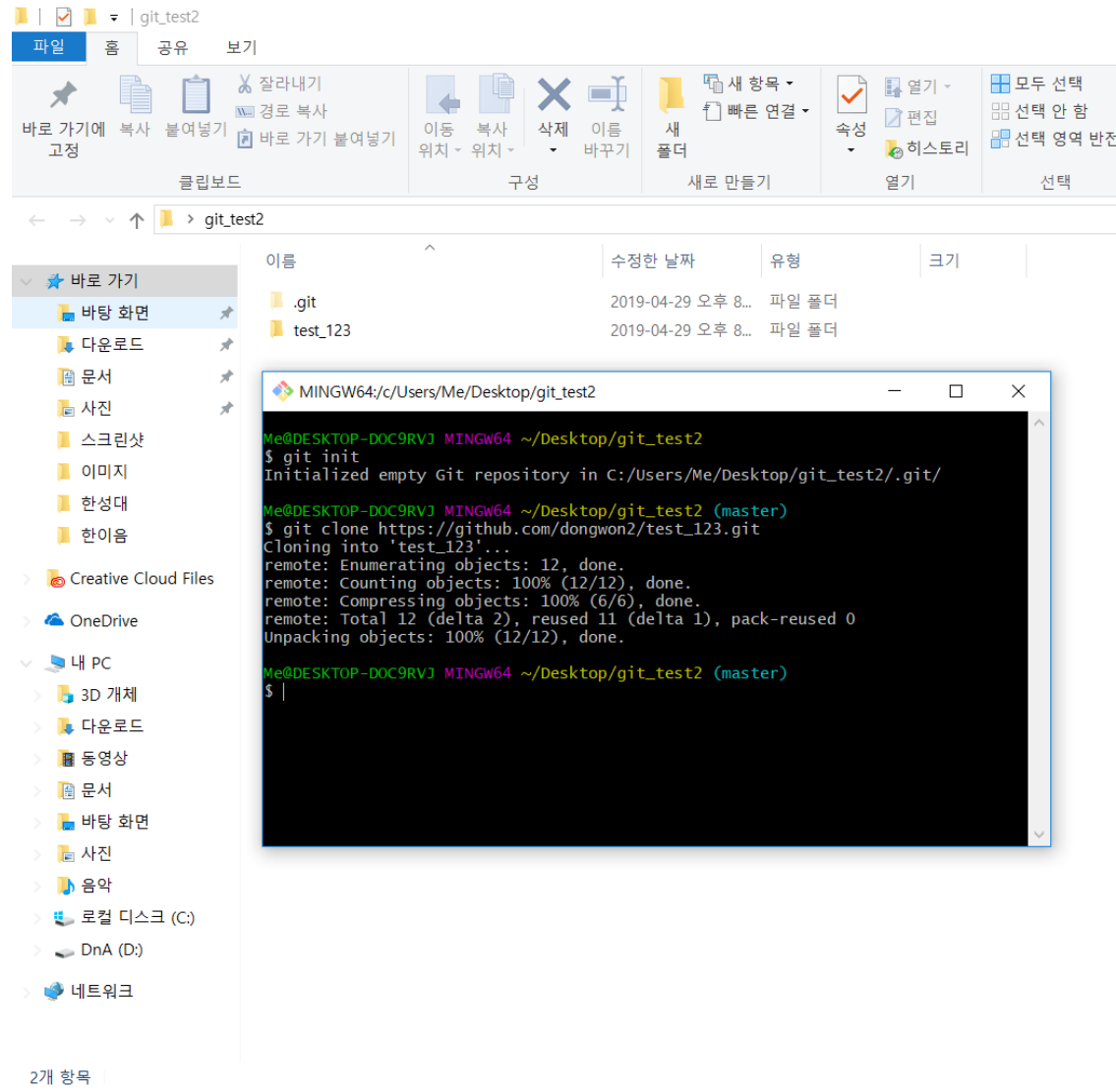
git clone + 복사한 주소

위 명령어를 실행하면 옆의 이미지와 같이

git 저장소를 폴더로 받아 옴

2. 추후 작성을 위해서는 받아온 디렉토리로 이동 후

add, push, pull 등이 사용 가능



03

Github 협업

git저장소에 내용을 추가할 시

3. git add

가장 최근의 commit과 변경이 있을 경우 working tree에 추가 가능

4. git commit -m '커밋메세지'

working tree에 추가가 된 내용을 커밋함

5. git push origin master

commit된 내용들을 git 저장소에 올림

MINGW64:/c/Users/Me/Desktop/git_test2/test_123

```
Me@DESKTOP-DOC9RVJ MINGW64 ~/Desktop/git_test2/test_123 (master)
$ git add .

Me@DESKTOP-DOC9RVJ MINGW64 ~/Desktop/git_test2/test_123 (master)
$ git commit -m '마 지 막 이 였 으 면 좋 겠 다 .'
[master 3b0884c] 마 지 막 이 였 으 면 좋 겠 다 .
1 file changed, 3 insertions(+), 1 deletion(-)

Me@DESKTOP-DOC9RVJ MINGW64 ~/Desktop/git_test2/test_123 (master)
$ git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 327 bytes | 163.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/dongwon2/test_123.git
   b414d52..3b0884c  master -> master

Me@DESKTOP-DOC9RVJ MINGW64 ~/Desktop/git_test2/test_123 (master)
$ |
```

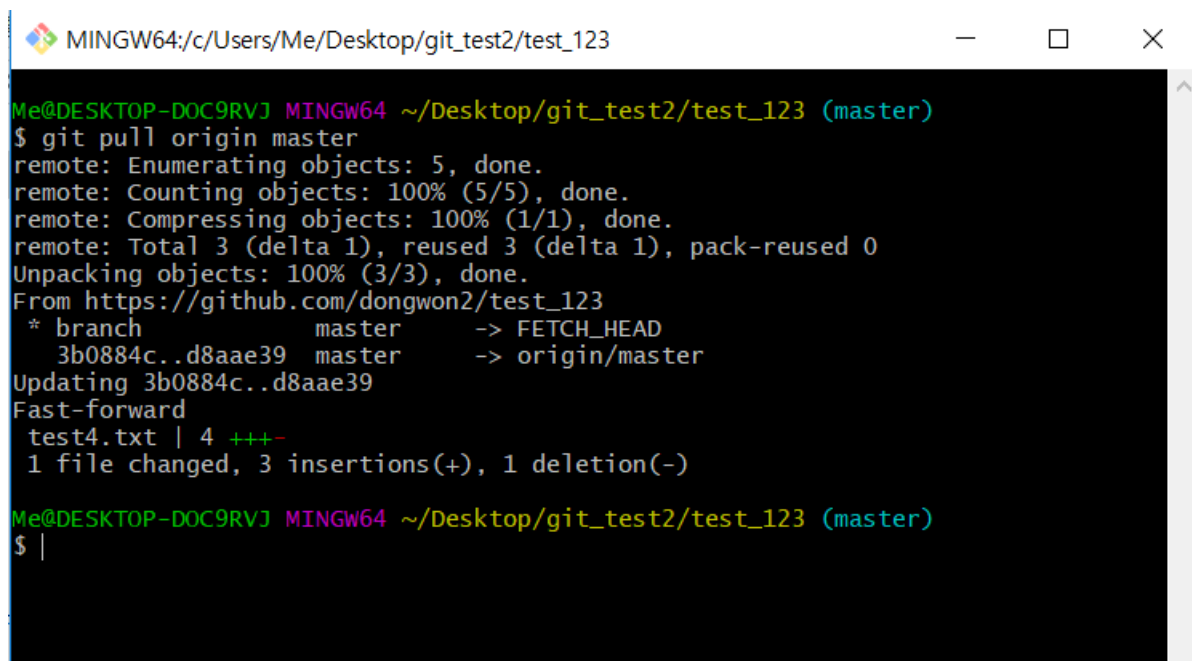
03

Github 협업

팀원이 git저장소에 추가한 내용을 받아올 때

이미 디렉토리에 프로젝트를 받아온 경우이며, 프로젝트 commit이 추가된 경우 가장 최신의 commit을 가져옴

명령어 : git pull origin master

A screenshot of a Windows terminal window with a black background and white text. The window title is 'MINGW64:/c/Users/Me/Desktop/git_test2/test_123'. The prompt is 'Me@DESKTOP-DOC9RVJ MINGW64 ~/Desktop/git_test2/test_123 (master)'. The user enters '\$ git pull origin master'. The output shows the process of pulling from the remote repository: 'remote: Enumerating objects: 5, done.', 'remote: Counting objects: 100% (5/5), done.', 'remote: Compressing objects: 100% (1/1), done.', 'remote: Total 3 (delta 1), reused 3 (delta 1), pack-reused 0', 'Unpacking objects: 100% (3/3), done.', 'From https://github.com/dongwon2/test_123', a table mapping the remote branch 'master' to 'FETCH_HEAD' and 'origin/master', 'Updating 3b0884c..d8aae39', 'Fast-forward', and a diff for 'test4.txt' showing 4 additions and 1 deletion. The final prompt is '\$ |'.

```
MINGW64:/c/Users/Me/Desktop/git_test2/test_123
Me@DESKTOP-DOC9RVJ MINGW64 ~/Desktop/git_test2/test_123 (master)
$ git pull origin master
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (1/1), done.
remote: Total 3 (delta 1), reused 3 (delta 1), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/dongwon2/test_123
 * branch                master      -> FETCH_HEAD
  3b0884c..d8aae39 master      -> origin/master
Updating 3b0884c..d8aae39
Fast-forward
 test4.txt | 4 +++-
 1 file changed, 3 insertions(+), 1 deletion(-)
Me@DESKTOP-DOC9RVJ MINGW64 ~/Desktop/git_test2/test_123 (master)
$ |
```

04

주의할 점

이 ppt에 있는 내용은 매우매우 기초적인 내용입니다.

log, status, branch 관련으로 필요한 요소도 없는

급하게 만든 ppt입니다.

그러므로 너무 망신하지 말고 각자 따로 더 공부할 필요가

있습니다.

그래도 프로젝트 시작을 위한 내용은 있으니

다들 github사용은 가능할 것입니다.

헛소리 하지마 임마!





Q&A

감사합니다