



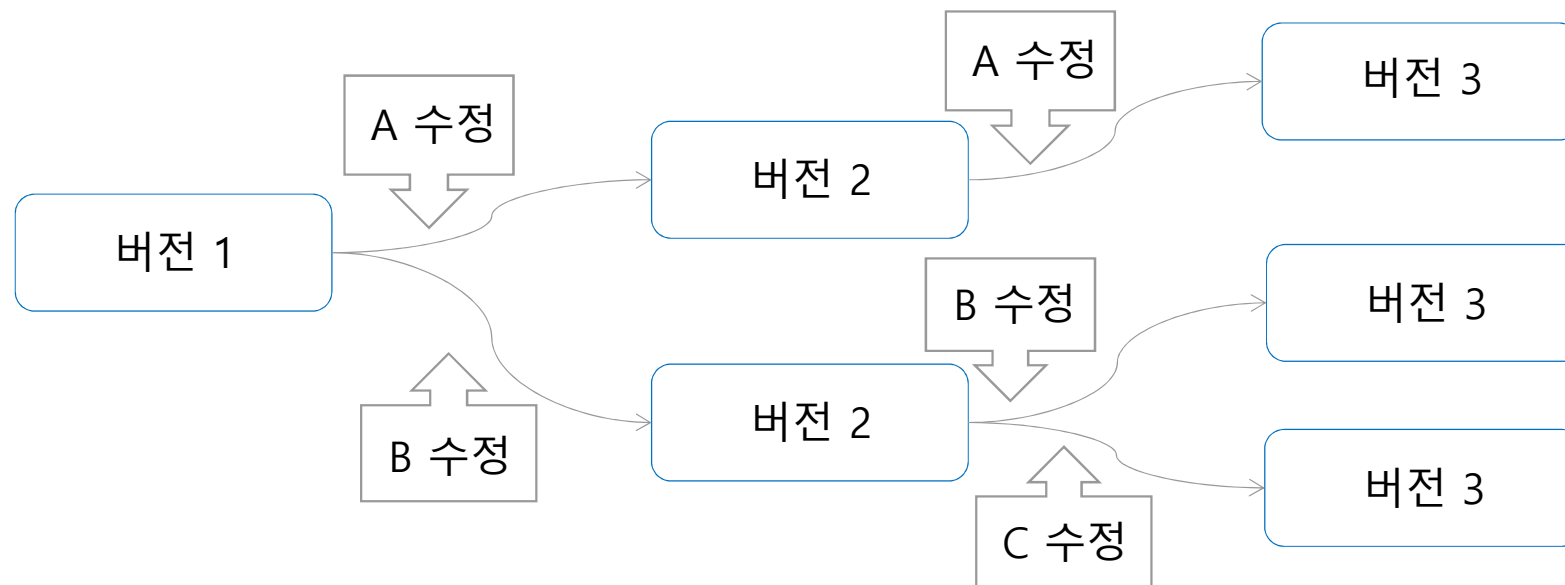
**Git / GitHub**

## ▶ 버전 관리란?

혼자서 작업한다면 버전 관리 시스템이 없어도 괜찮지만

팀 프로젝트의 소스 코드는 버전 1에 대해서 팀원 A와 B가 동시에 수정을 하여 각각의 버전 2를 만든 경우 어떤 것이 최종 업데이트 파일인지 확인할 수 없음.

따라서 협업 프로젝트는 버전 관리가 필요함.



## ▶ Git

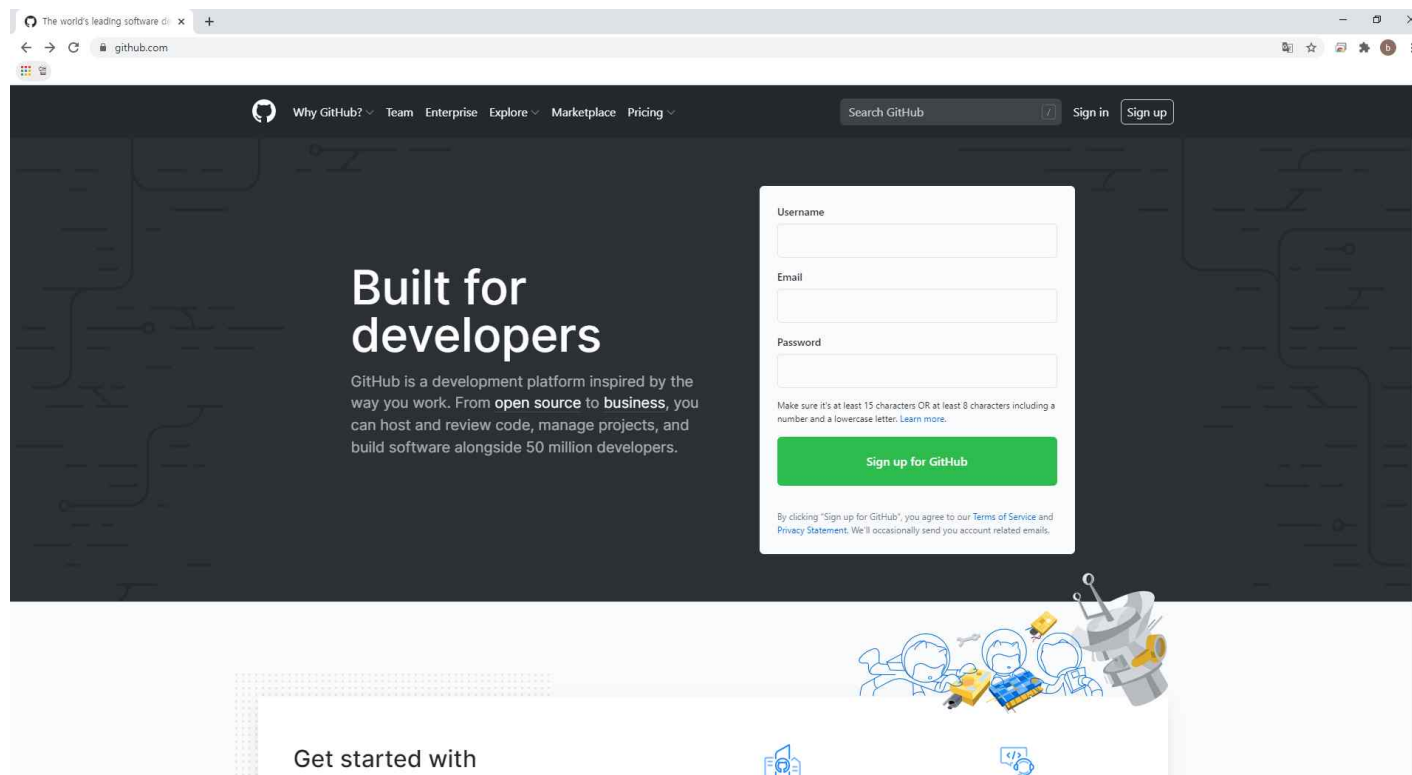
원하는 시점마다 깃발을 꽂고 깃발이 꽂힌 시점으로 자유롭게 이동 가능한 소스코드 버전 관리 시스템.  
개인 컴퓨터에 저장한다면 혼자 사용. usb에 저장하면 휴대하고 사용.  
구글 드라이브와 같은 서버에 올리면 팀원들과 함께 인터넷을 통한 버전 관리 가능.

## ▶ GitHub

git 호스팅 사이트 중 하나. 시간과 공간의 제약 없이 협업 가능. 공개 저장소로 만들 시 모르는 사람과도 협업 가능. 누구든지 기여할 수 있는 공개저장소 프로젝트 => 오픈소스  
GitLab, BitButcket등 다양한 호스팅 사이트가 있음

## ▶ GitHub 가입하기

github.com에 접속하여 sign up 진행



# ▶ GitHub 가입하기

필요한 정보 입력 후 create account, join a free plan, complete set up 클릭

## Create your account

Username \*

kh-bl ✓

Email address \*

khblgit@gmail.com ✓

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.

Verify your account

이미지가 올바르게 되었을 때 완료!를 누르세요.

완료 ?

Create account

By creating an account, you agree to the [Terms of Service](#). For more information about GitHub's

Join GitHub

## Create your account

Verify your account

✓

?

Email preferences

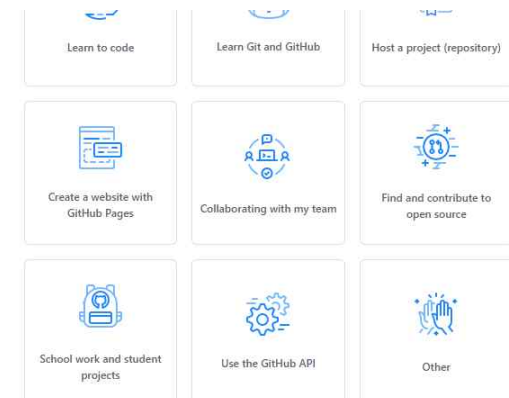
☐ Send me occasional product updates, announcements, and offers.

Join a free plan

Product  
Features

Platform  
Developer API

Support  
Help



I am interested in:

languages, frameworks, industries

We'll connect you with communities and projects that fit your interests.  
For example: windows deployment covid-19

Complete setup

Security

Status

Help

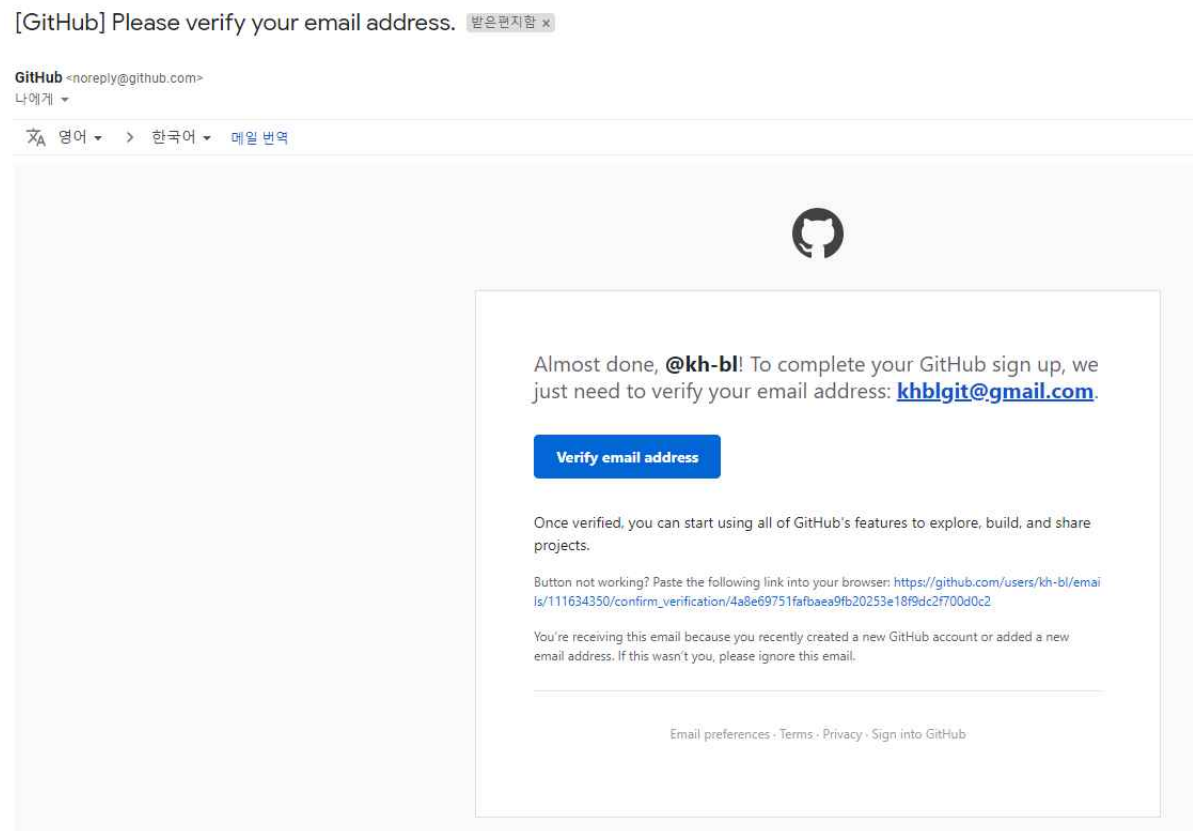
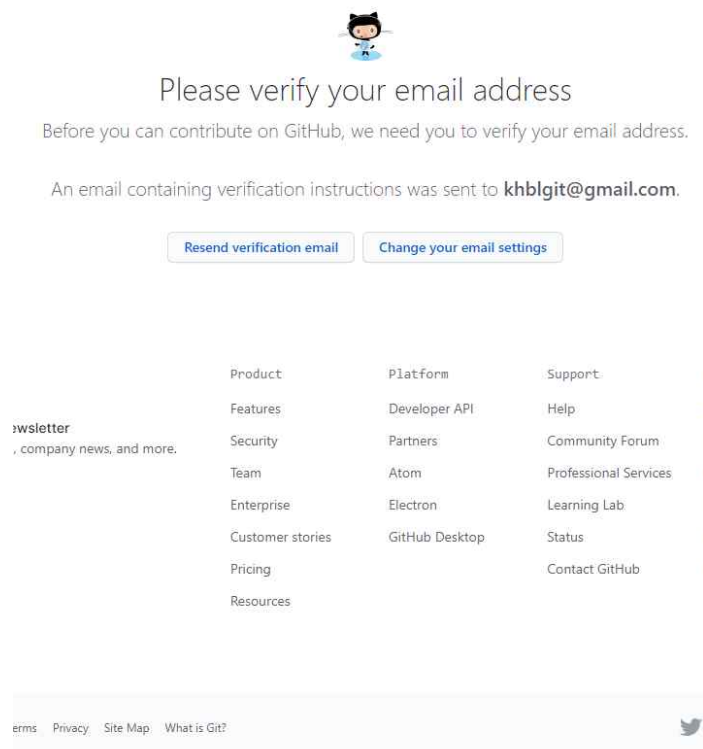


Contact GitHub

Pricing

# ▶ GitHub 가입하기

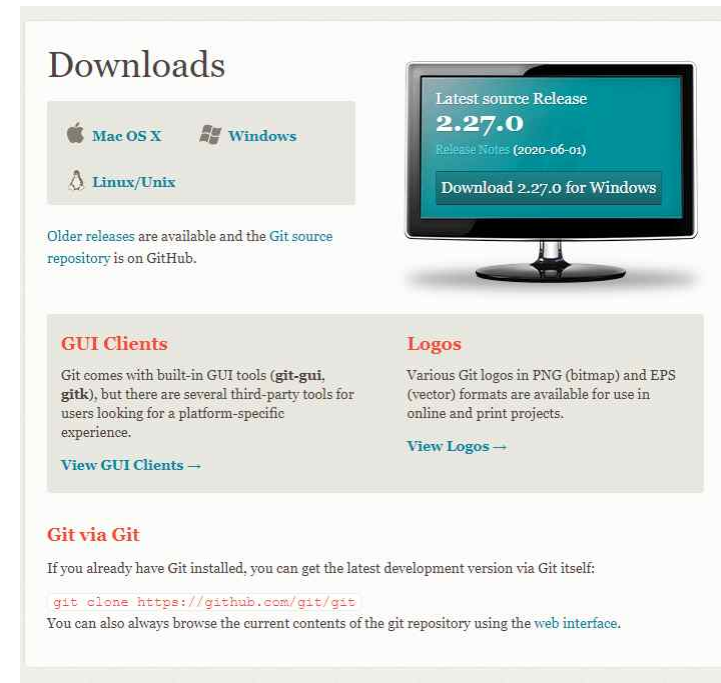
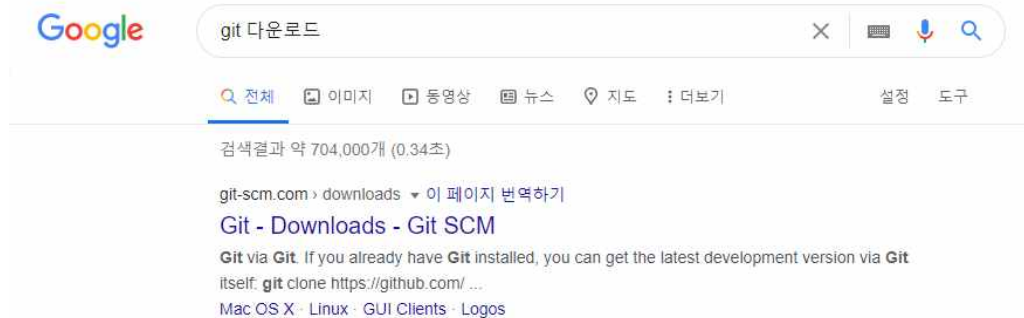
가입한 email 주소로 발송 된 메일 확인하여 verify email address 클릭하면 가입 완료



# **Git - local repository**

## ▶ Git 설치

git 다운로드 검색하여 git-scm.com/download/에 접속  
OS에 맞는 최신 버전의 git 다운로드

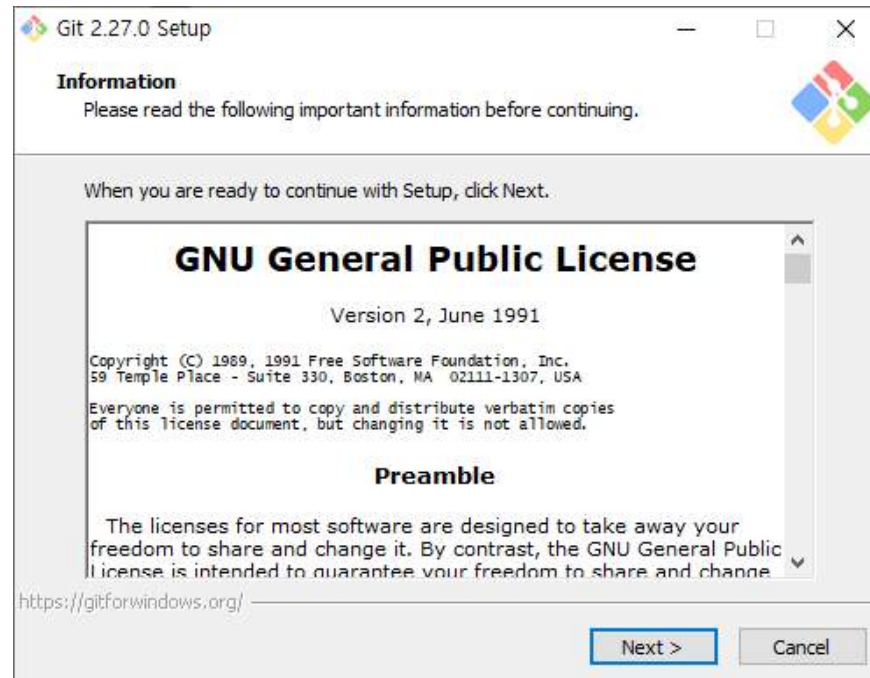




## ▶ Git 설치

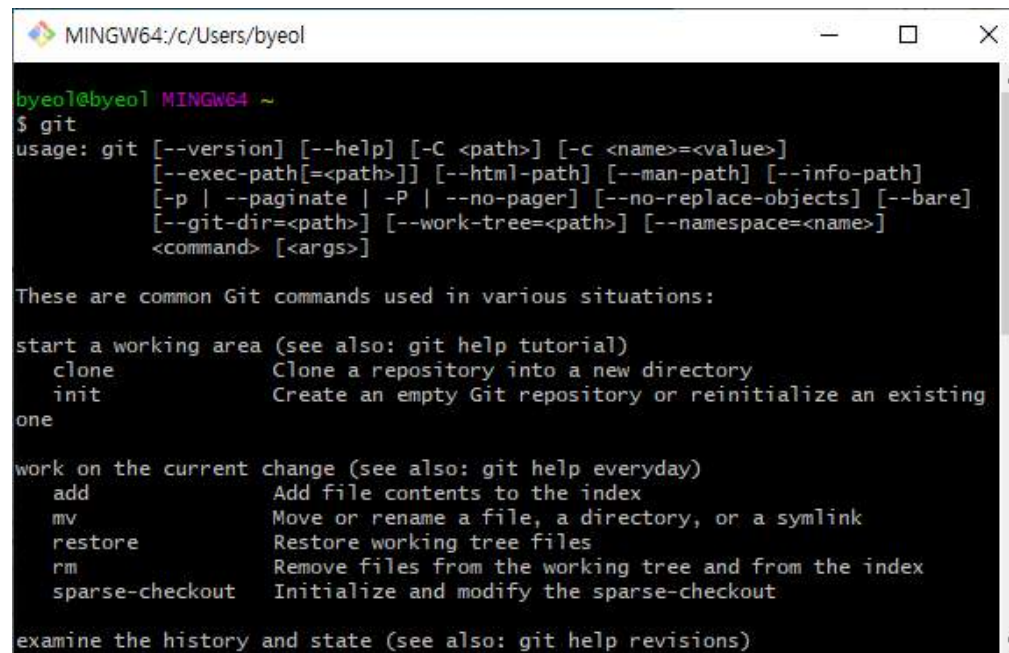
다운로드 된 파일 실행하면 아래와 같은 창이 뜬

기본 설정으로 next 및 install 버튼을 클릭해서 설치 진행 후 finish 버튼을 클릭하면 설치 완료



## ▶ Git 설치

설치 확인을 위해 검색을 통해 Git Bash 실행 후 \$ 옆에 git 입력  
아래와 같이 기본 명령어에 대한 안내가 나오면 설치 완료 확인



```
MINGW64:/c/Users/byeol
byeol@byeol MINGW64 ~
$ git
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
        [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
        [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
        [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
        <command> [<args>]

These are common Git commands used in various situations:


start a working area (see also: git help tutorial)
    clone             Clone a repository into a new directory
    init              Create an empty Git repository or reinitialize an existing
one

work on the current change (see also: git help everyday)
    add              Add file contents to the index
    mv               Move or rename a file, a directory, or a symlink
    restore           Restore working tree files
    rm               Remove files from the working tree and from the index
    sparse-checkout   Initialize and modify the sparse-checkout

examine the history and state (see also: git help revisions)
```

## ▶ 로컬 저장소(local repository) 만들기

바탕 화면 > gitLocalTest > test 경로에 README.txt 파일 만들기  
txt 파일 열어 아래와 같이 적고 저장 후 닫기



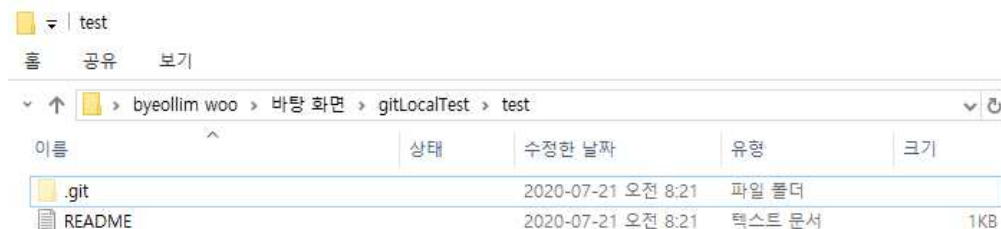
## ▶ 로컬 저장소(local repository) 만들기

해당 경로에서 우클릭하여 Git Bash Here 클릭

\$ 옆에 git init 입력하여 'Initialized empty Git repository' 나오면 초기화 성공

.git 폴더 자동 생성 (Git으로 생성한 버전 정보, 원격 저장소 주소 등이 들어있는 로컬 저장소)

```
MINGW64:/c/Users/byeol/OneDrive/바탕 화면/gitLocalTest/test
byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test
$ git init
Initialized empty Git repository in C:/Users/byeol/OneDrive/바탕 화면/gitLocalTest/test/.git/
byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ |
```



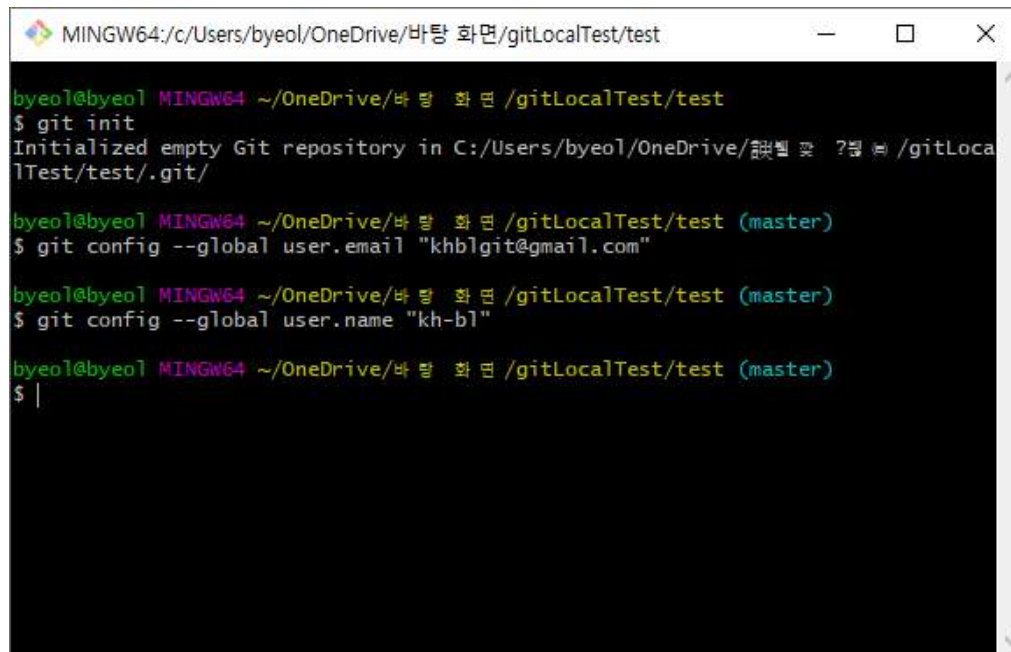
이름	상태	수정한 날짜	유형	크기
.git		2020-07-21 오전 8:21	파일 폴더	
README		2020-07-21 오전 8:21	텍스트 문서	1KB

## ▶ 로컬 저장소(local repository)에 commit 만들기

GitHub에 등록한 email, username을 사용하여 아래와 같은 명령어 입력

```
$git config --global user.email "GitHub이메일"
```

```
$git config --global user.name "GitHub이름"
```

A screenshot of a terminal window titled "MINGW64:/c:/Users/byeol/OneDrive/바탕 화면/gitLocalTest/test". The terminal shows the following commands and output:

```
byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test
$ git init
Initialized empty Git repository in C:/Users/byeol/OneDrive/바탕 화면 /gitLocalTest/test/.git/

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ git config --global user.email "khlgit@gmail.com"

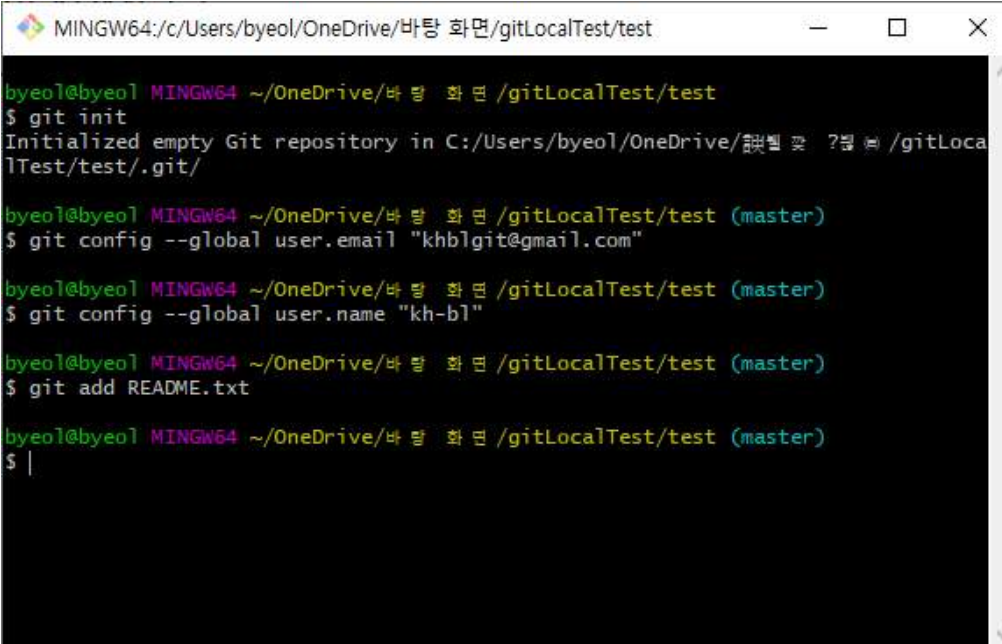
byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ git config --global user.name "kh-bl"

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ |
```

## ▶ 로컬 저장소(local repository)에 commit 만들기

commit에 추가할 파일 선택

```
$ git add README.txt
```

A screenshot of a Windows terminal window with a black background and white text. The window title is 'MINGW64:/c:/Users/byeol/OneDrive/바탕 화면/gitLocalTest/test'. The terminal shows the following commands and output:

```
byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test
$ git init
Initialized empty Git repository in C:/Users/byeol/OneDrive/바탕 화면 /gitLocalTest/test/.git/

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ git config --global user.email "khlgit@gmail.com"

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ git config --global user.name "kh-bl"

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ git add README.txt

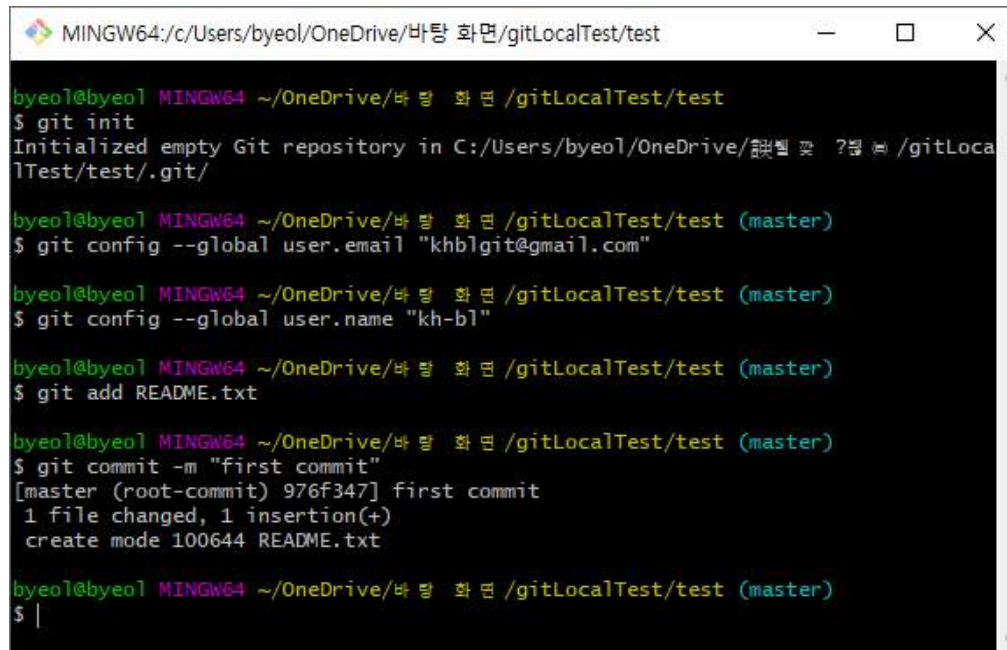
byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ |
```

## ▶ 로컬 저장소(local repository)에 commit 만들기

커밋에 "first commit"이라는 상세 설명을 적어 첫번째 커밋하기 (-m : message)

```
$ git commit -m "first commit"
```

1 file changed, 1 insertion(+) 확인되면 첫번째 버전 생성 완료



```
MINGW64:/c/Users/byeol/OneDrive/바탕 화면/gitLocalTest/test

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test
$ git init
Initialized empty Git repository in C:/Users/byeol/OneDrive/바탕 화면/gitLocalTest/test/.git/

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ git config --global user.email "khlgit@gmail.com"

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ git config --global user.name "kh-bl"

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ git add README.txt

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ git commit -m "first commit"
[master (root-commit) 976f347] first commit
1 file changed, 1 insertion(+)
create mode 100644 README.txt

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ |
```

## ▶ 로컬 저장소(local repository)에 commit 만들기

README.txt 파일 수정 후

```
$ git add README.txt
```

```
$ git commit -m "updated"
```

\*README - Windows 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

1 git local repository

2 updated

```
MINGW64:/c/Users/byeol/OneDrive/바탕 화면/gitLocalTest/test
byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test (master)
$ git add README.txt

byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test (master)
$ git commit -m "updated"
[master d3dad41] updated
1 file changed, 2 insertions(+), 1 deletion(-)

byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test (master)
$ |
```



## ▶ 로컬 저장소(local repository)의 다른 커밋으로 이동하기

\$ git log 명령어로 지금까지 만든 두 개의 커밋 확인

first commit의 앞 7자리 커밋 아이디를 복사하여 checkout 명령어로 해당 커밋으로 코드를 되돌림

\$ git checkout 976f347

"HEAD is now at 976f347 first commit" 확인

```
MINGW64:/c/Users/byeol/OneDrive/바탕 화면/gitLocalTest/test
byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test (master)
$ git add README.txt

byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test (master)
$ git commit -m "updated"
[master d3dad41] updated
1 file changed, 2 insertions(+), 1 deletion(-)

byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test (master)
$ git log
commit d3dad412012cc303e56cc65cd005c5d4ad7e5ca8 (HEAD -> master)
Author: kh-bl <khblgit@gmail.com>
Date: Tue Jul 21 13:27:34 2020 +0900

    updated

commit 976f347ba0a398480ba40c61519b1bd92271cfe6
Author: kh-bl <khblgit@gmail.com>
Date: Tue Jul 21 08:40:06 2020 +0900

    first commit

byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test (master)
$
```

```
MINGW64:/c/Users/byeol/OneDrive/바탕 화면/gitLocalTest/test
byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test (master)
$ git checkout 976f347
Note: switching to '976f347'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 976f347 first commit

byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test ((976f347...))
$
```

## ▶ 로컬 저장소(local repository)의 다른 커밋으로 이동하기

README.txt 파일 확인 하면 first commit 상태로 돌아가 있음

\$ git checkout - 를 입력하여 다시 최신 커밋인 두 번째 커밋으로 돌아가기

README.txt 파일 확인하면 updated 상태로 돌아가 있음

README - Windows 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

1 git local repository

```
MINGW64:/c/Users/byeol/OneDrive/바탕 화면/gitLocalTest/test
You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 976f347 first commit

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test ((976f347...))
$ git checkout -
Previous HEAD position was 976f347 first commit
Switched to branch 'master'

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ |
```

README - Windows 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

1 git local repository  
2 updated

# **Git - remote repository**

## ▶ 원격 저장소(remote repository)에 커밋 올리기

### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?

[Import a repository.](#)

Owner \*

kh-bl

Repository name \*

gitRemoteTest

Great repository names are short and memorable. Need inspiration? How about [musical-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.

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

☐ Initialize this repository with a README

This will let you immediately clone the repository to your computer.

Add .gitignore: None

Add a license: None



Create repository

GitHub에서 new repository 클릭

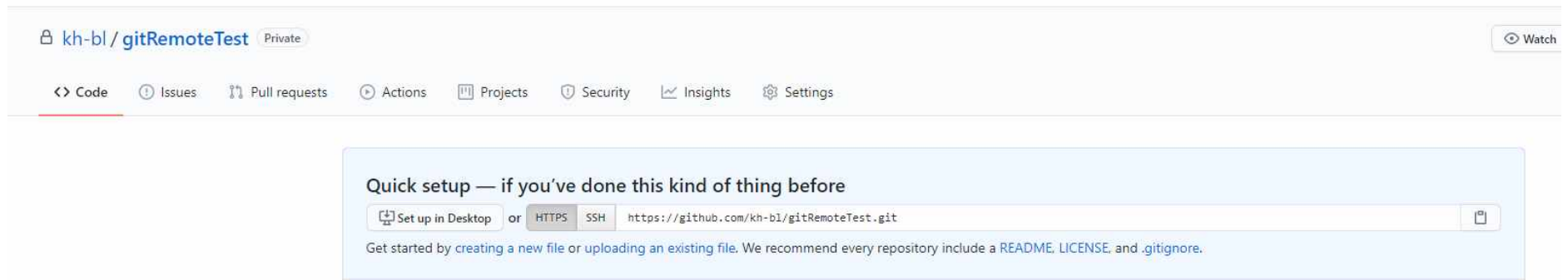
repository 이름 입력 후 옵션 선택한 뒤  
create repository 클릭



## ▶ 원격 저장소(remote repository)에 커밋 올리기

<https://github.com/kh-bl/gitRemoteTest.git>

이 주소를 통해 원격 저장소에 접속 가능함

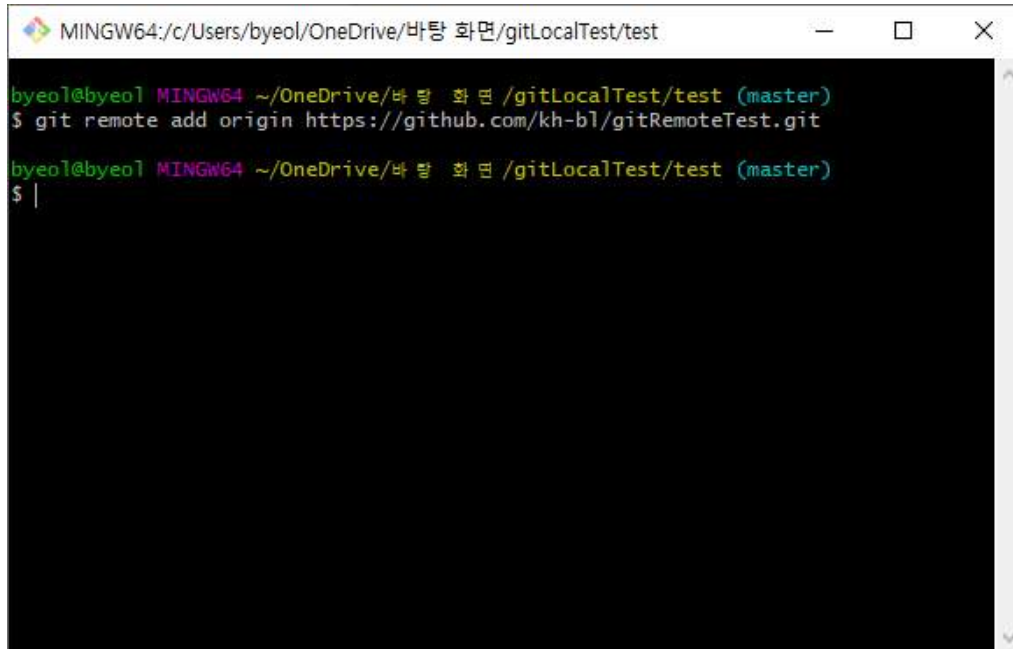


## ▶ 원격 저장소(remote repository)에 커밋 올리기

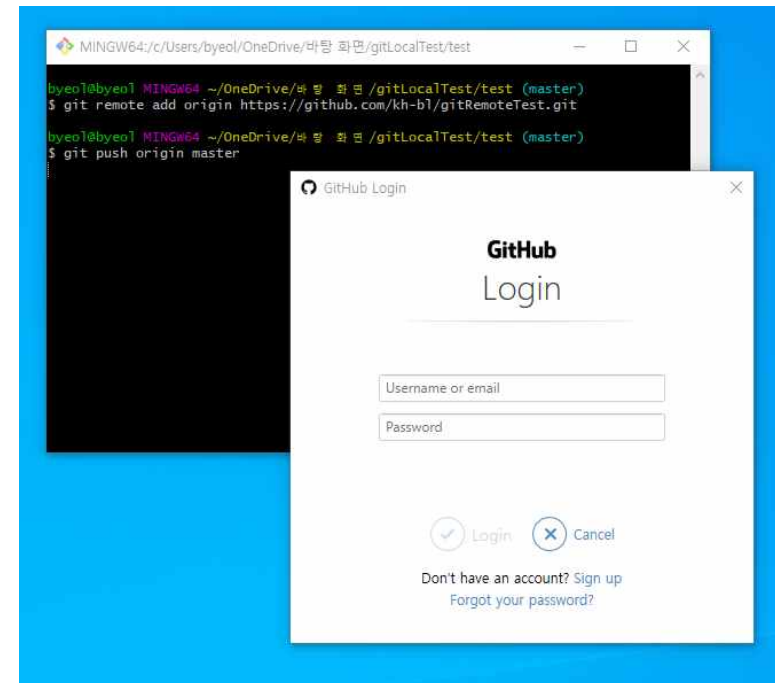
```
$ git remote add origin https://github.com/kh-bl/gitRemoteTest.git
```

로컬 저장소에 원격 저장소 주소 입력

```
$ git push origin master
```

 입력 시 GitHub 로그인 창이 뜨므로 email과 password 입력하여 로그인

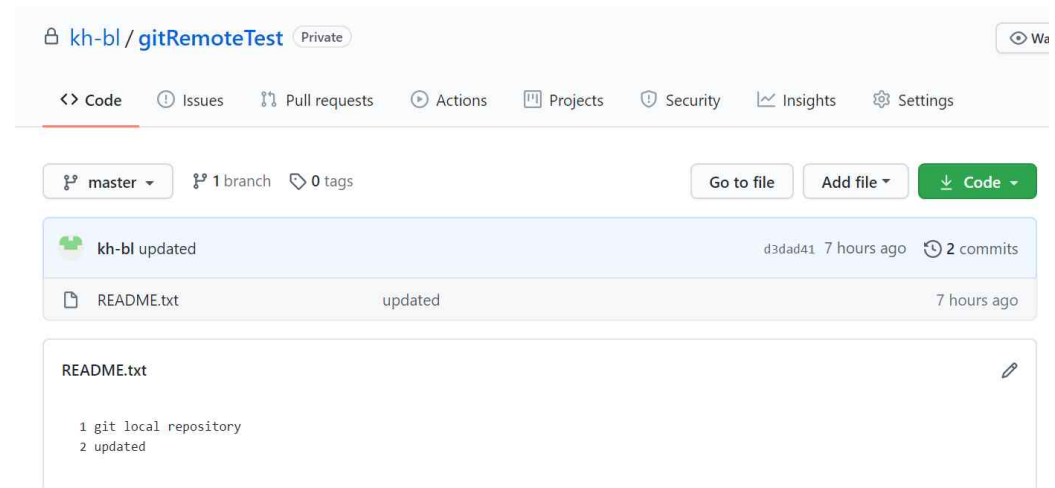
```
MINGW64:/c/Users/byeol/OneDrive/바탕 화면/gitLocalTest/test
byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ git remote add origin https://github.com/kh-bl/gitRemoteTest.git
byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test (master)
$ |
```



## ▶ 원격 저장소(remote repository)에 커밋 올리기

GitBash에서 아래와 같은 화면 확인 후 GitHub에서 저장소 확인

```
MINGW64:/c:/Users/byeol/OneDrive/바탕 화면/gitLocalTest/test
byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test (master)
$ git remote add origin https://github.com/kh-bl/gitRemoteTest.git
byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test (master)
$ git push origin master
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (6/6), 464 bytes | 92.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/kh-bl/gitRemoteTest.git
 * [new branch]      master -> master
byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test (master)
$
```



## ▶ 원격 저장소의 커밋을 로컬 저장소에 내려받기

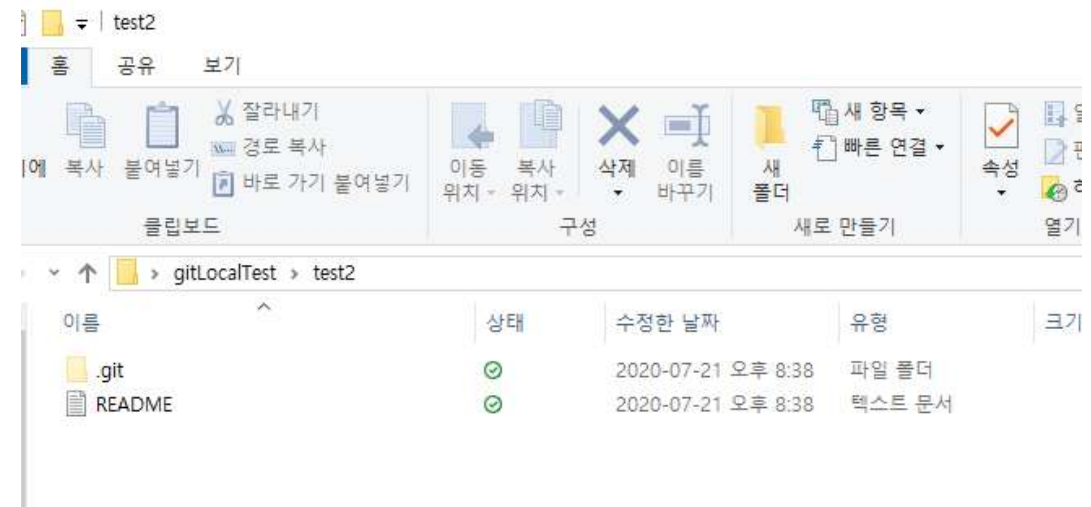
바탕 화면 > gitLocalTest > test2 경로 만들고 우클릭하여 Git Bash Here 클릭

```
$ git clone https://github.com/kh-bl/gitRemoteTest.git .
```

원격 저장소 주소 뒤의 마침표의 의미는 현재 폴더를 뜻함

README.txt 파일과 .git 폴더가 보이면 clone 성공

```
MINGW64:/c/Users/byeol/OneDrive/바탕 화면/gitLocalTest/test2
byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test2
$ git clone https://github.com/kh-bl/gitRemoteTest.git .
Cloning into '.'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 6 (delta 0), reused 6 (delta 0), pack-reused 0
Unpacking objects: 100% (6/6), 444 bytes | 19.00 KiB/s, done.
byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test2 (master)
$ |
```





## ▶ 원격 저장소의 커밋을 로컬 저장소에 내려받기

README.txt 파일을 수정하고 저장

\$ git add README.txt (README.txt 파일 커밋)

\$ git commit -m "test2 first commit" (커밋에 메시지로 설명 추가)

\$ git push origin master (원격 저장소에 push)



```
MINGW64:/c:/Users/byeol/OneDrive/바탕 화면/gitLocalTest/test2
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 6 (delta 0), reused 6 (delta 0), pack-reused 0
Unpacking objects: 100% (6/6), 444 bytes | 19.00 KiB/s, done.

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test2 (master)
$ git add README.txt

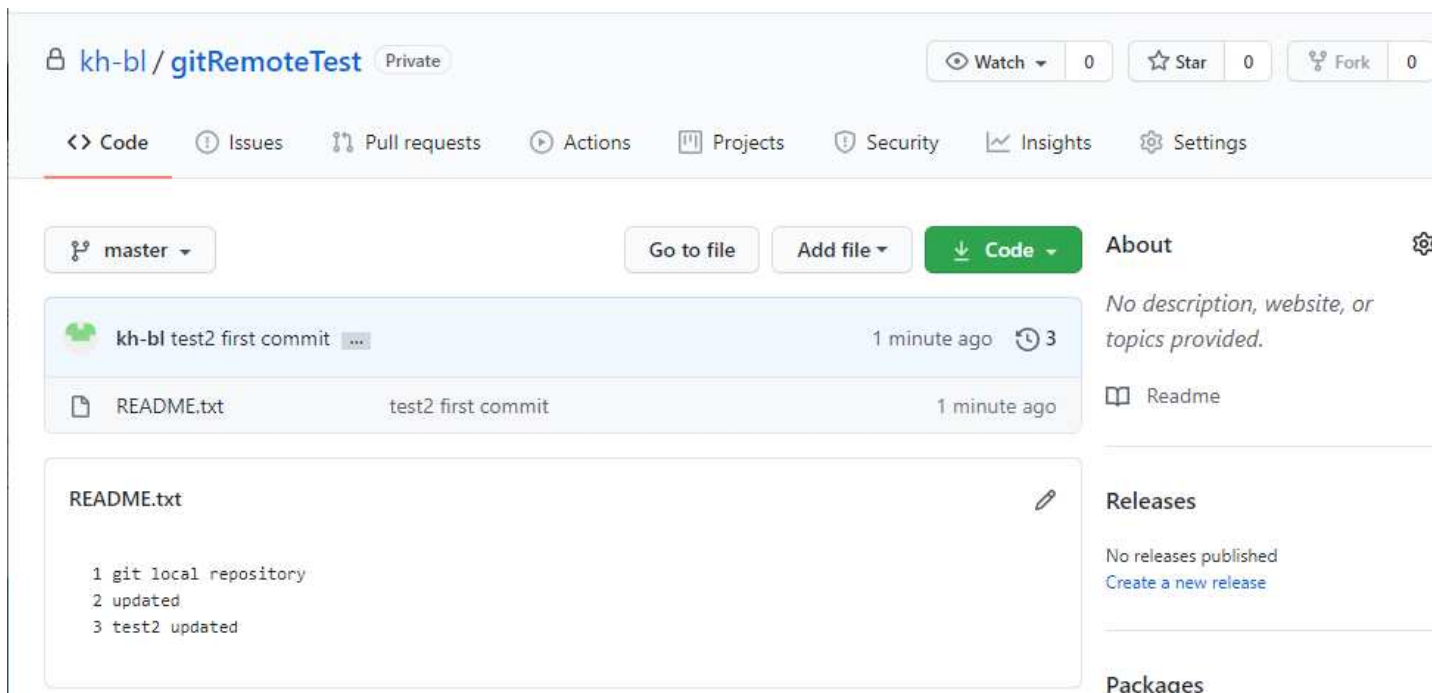
byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test2 (master)
$ git commit -m "test2 first commit"
[master 08d631a] test2 first commit
1 file changed, 2 insertions(+), 1 deletion(-)

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test2 (master)
$ git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 281 bytes | 70.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/kh-b1/gitRemoteTest.git
d3dad41..08d631a master -> master

byeol@byeol MINGW64 ~/OneDrive/바탕 화면 /gitLocalTest/test2 (master)
$
```

## ▶ 원격 저장소의 커밋을 로컬 저장소에 내려받기

GitHub의 원격 저장소에 들어가 "test2 first commit" 올라와 있는 것 확인

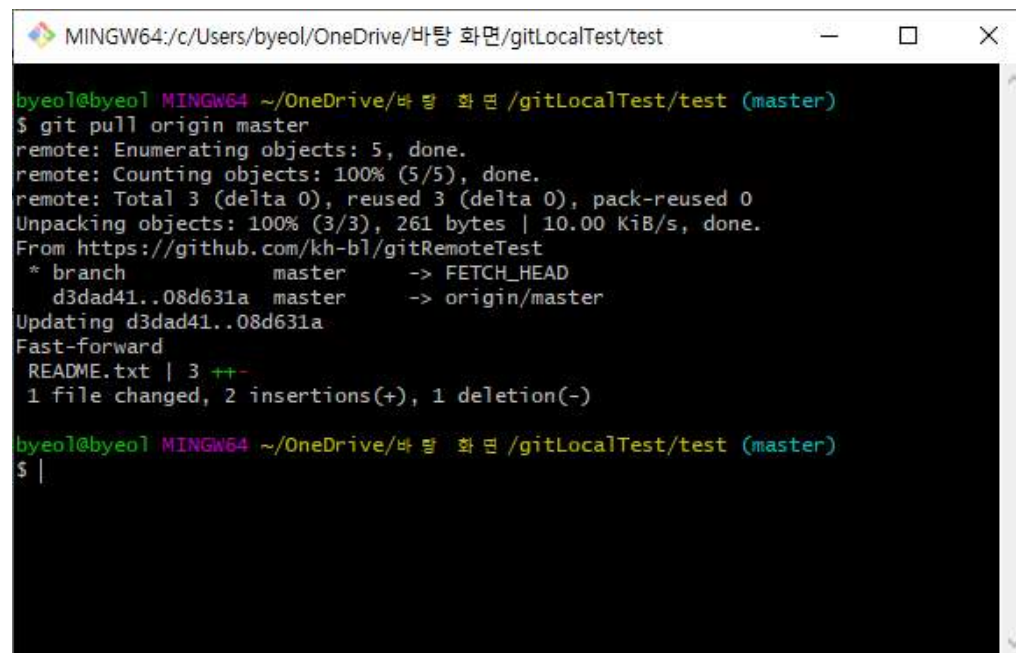


## ▶ 원격 저장소의 커밋을 로컬 저장소에 내려받기

바탕 화면 > gitLocalTest > test 경로의 README.txt 파일은 "test2 first commit" 변경 사항이 반영되어 있지 않으므로 원격 저장소에서부터 변경사항을 반영함.

해당 경로에서 우클릭 Git Bash Here 클릭하여

\$ git pull origin master (원격 저장소의 새로운 커밋을 로컬 저장소로 받아옴)




```
MINGW64:/c/Users/byeol/OneDrive/바탕 화면/gitLocalTest/test
byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test (master)
$ git pull origin master
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 261 bytes | 10.00 KiB/s, done.
From https://github.com/kh-b1/gitRemoteTest
* branch      master      -> FETCH_HEAD
  d3dad41..08d631a master  -> origin/master
Updating d3dad41..08d631a
Fast-forward
 README.txt | 3 ++-
 1 file changed, 2 insertions(+), 1 deletion(-)

byeol@byeol MINGW64 ~/OneDrive/바탕 화면/gitLocalTest/test (master)
$ |
```

## ▶ 원격 저장소의 커밋을 로컬 저장소에 내려받기

바탕 화면 > gitLocalTest > test 경로의 README.txt 파일을 열어 변경 사항 확인



```
README - Windows 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
1 git local repository
2 updated
3 test2 updated
```

## ▶ 용어 정리

Git : 버전 관리 시스템

GitHub : Git으로 관리하는 프로젝트를 올려둘 수 있는 사이트

Git Bash : CLI 방식으로 Git을 사용할 수 있는 환경

CLI(Command Line Interface) - 명령어를 입력하는 방식. GUI(Graphic User Interface) - 마우스로 클릭하는 방식.

Commit : 버전 관리를 통해 생성된 파일, 혹은 그 행위.

Log : 지금까지 만든 커밋을 모두 확인하는 명령어.

Checkout : 원하는 지점으로 파일을 되돌림.

Local repository : 로컬 저장소. Git으로 버전 관리하는 내 컴퓨터 안의 폴더를 의미함.

Remote repository : 원격 저장소. GitHub에서 협업하는 공간을 의미함.

Push : 로컬 저장소의 커밋(버전 관리한 파일)을 원격 저장소에 올리는 것.

Pull : 원격 저장소의 커밋을 로컬 저장소에 내려 받는 것.