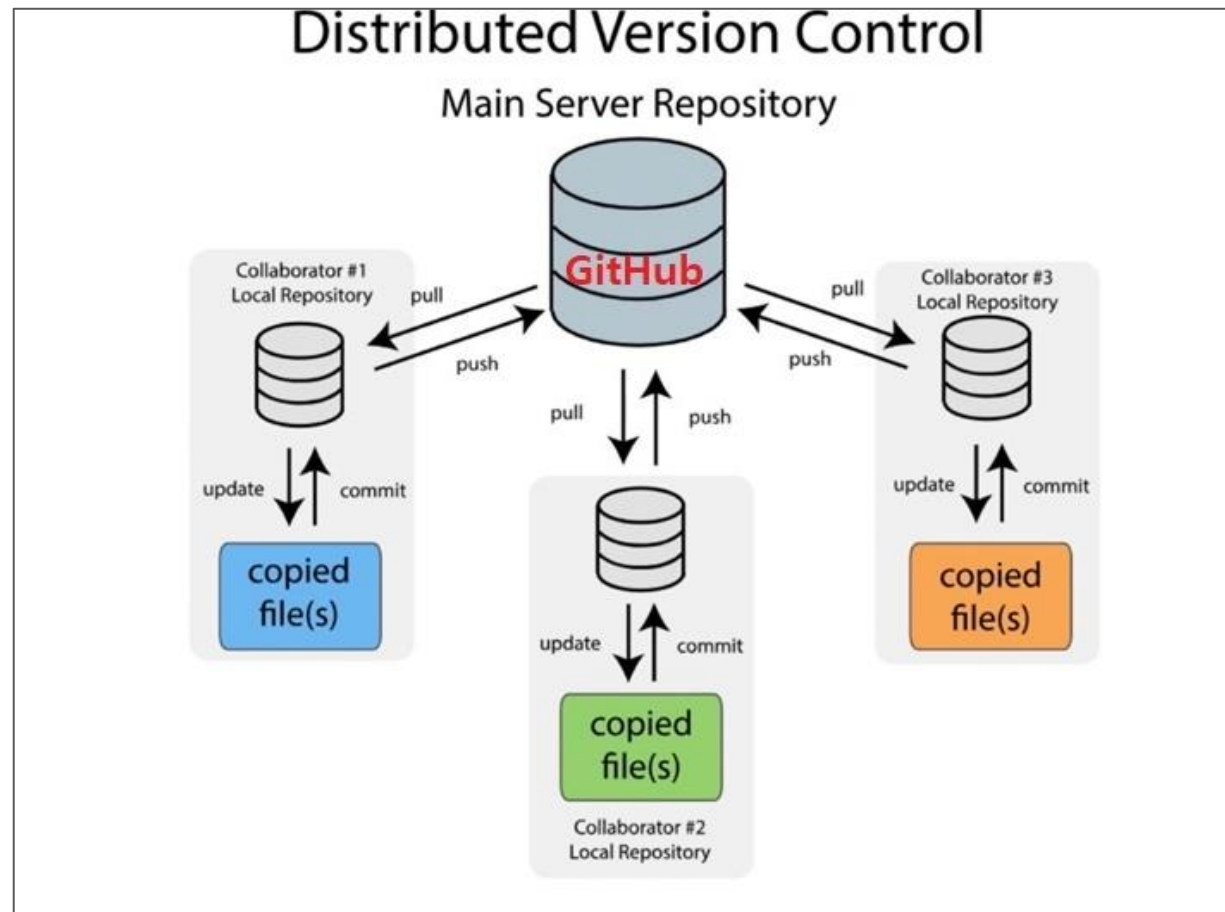


# 원격 저장소(Github) 생성 및 활용

동의과학대학교 컴퓨터정보과

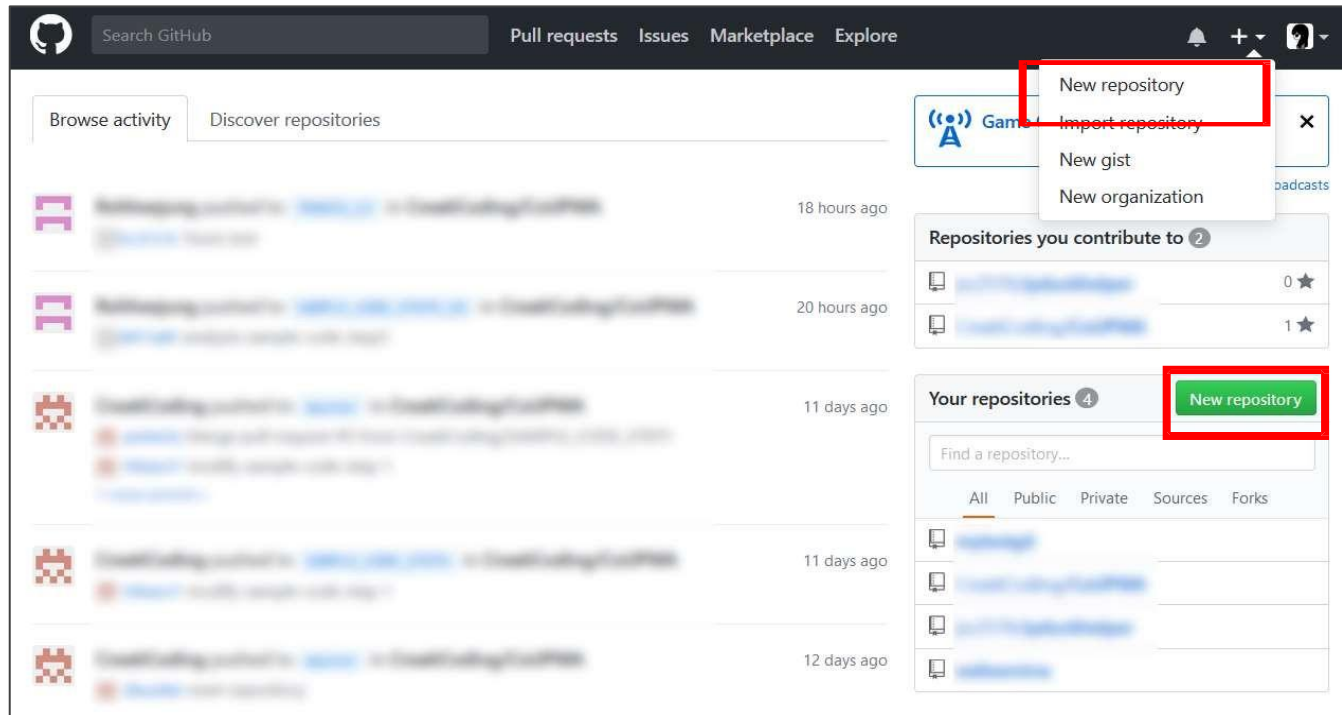
# 원격 저장소(GitHub)

- Github는 가장 큰 Git 저장소 호스트
- 많은 오픈 소스 프로젝트는 Github를 이용하여 Git 호스팅, 코드 리뷰 등의 작업 진행

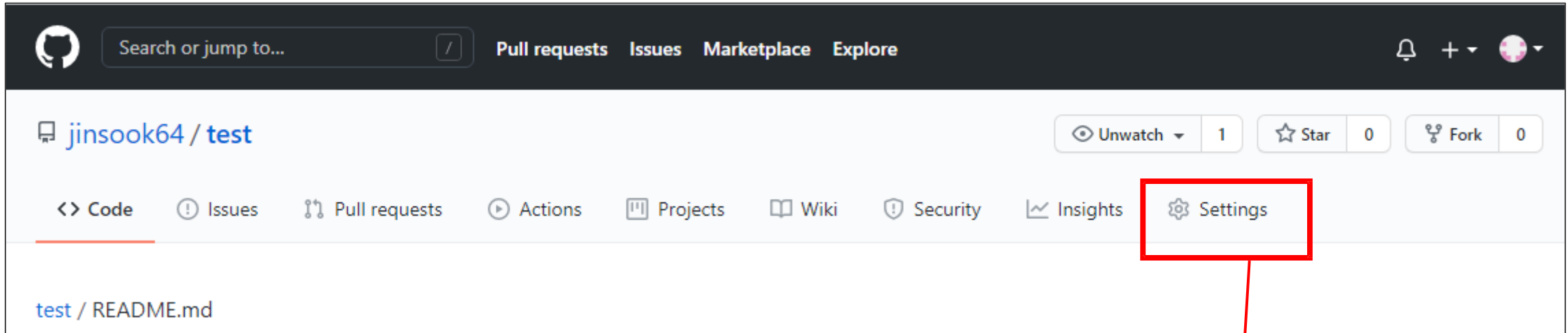


# 원격저장소(GitHub) Repository 생성

1. <https://github.com/> 에 접속하기
2. 계정 만들고(Sign up) 로그인하기(Sign in)
3. 저장소 생성하기 (repository name : test)
  - 상단의 +를 누르고 New repository를 누르거나 Your repository의 New repository 클릭



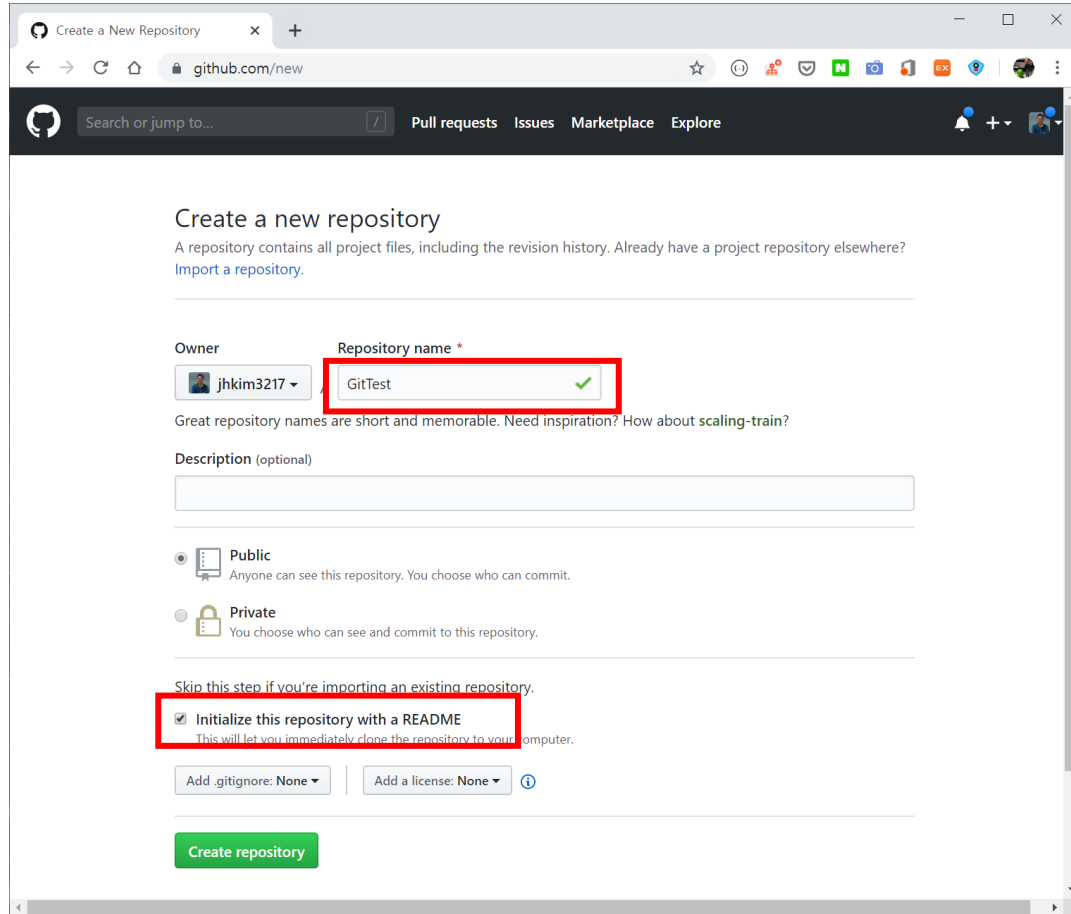
# 원격저장소(GitHub) Repository 삭제



## Danger Zone

<b>Change repository visibility</b> This repository is currently public.	<a href="#">Change visibility</a>
<b>Transfer ownership</b> Transfer this repository to another user or to an organization where you have the ability to create repositories.	<a href="#">Transfer</a>
<b>Archive this repository</b> Mark this repository as archived and read-only.	<a href="#">Archive this repository</a>
<b>Delete this repository</b> Once you delete a repository, there is no going back. Please be certain.	<a href="#">Delete this repository</a>

# 원격저장소(GitHub) Repository 생성



Create a new repository

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

Owner: jhkim3217 Repository name: GitTest ✓

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

Description (optional)

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

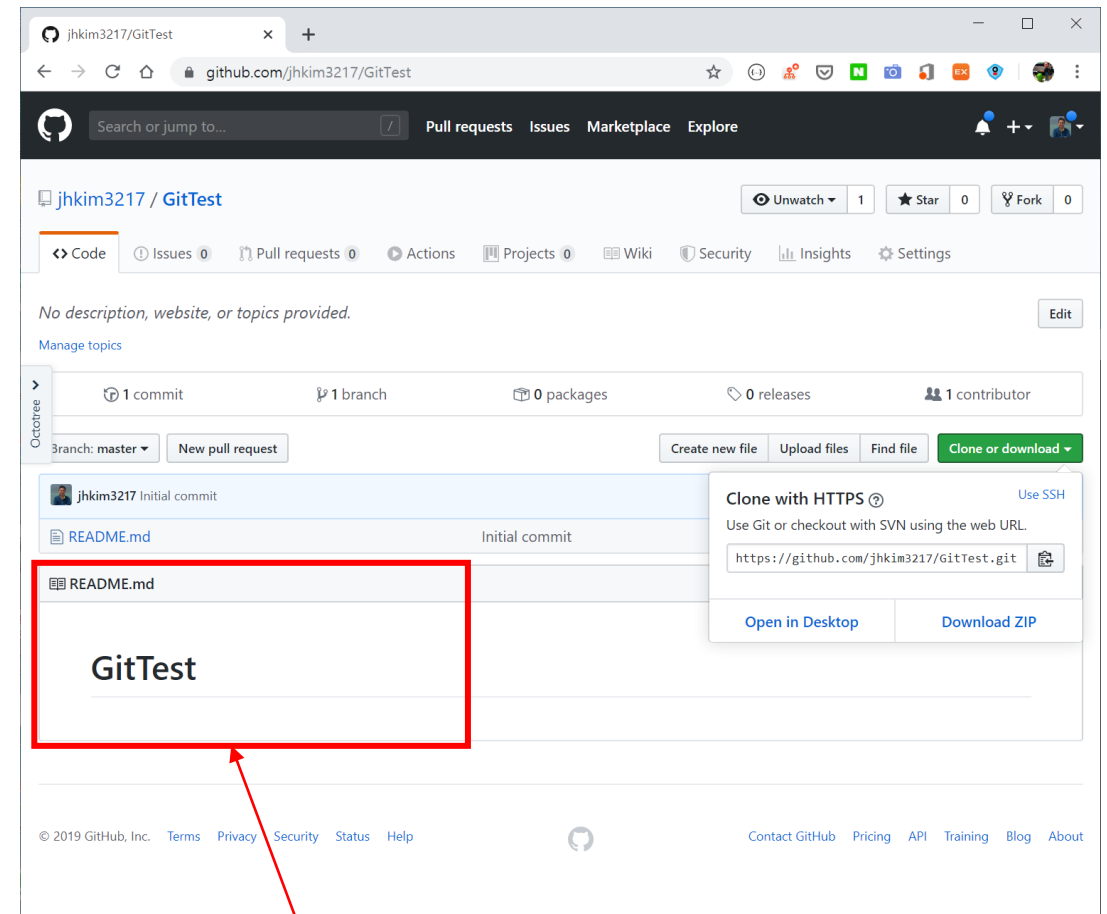
☐ Private  
You choose who can see and commit to this repository.

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



jhkim3217 / GitTest

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security Insights Settings

No description, website, or topics provided. [Edit](#)

Manage topics

1 commit 1 branch 0 packages 0 releases 1 contributor

branch: master New pull request

Create new file Upload files Find file Clone or download

Initial commit

README.md Initial commit

Clone with HTTPS ⓘ Use SSH

Use Git or checkout with SVN using the web URL.

<https://github.com/jhkim3217/GitTest.git>

Open in Desktop Download ZIP

© 2019 GitHub, Inc. Terms Privacy Security Status Help Contact GitHub Pricing API Training Blog About

Mark Down 언어로 주석 작성(후에 연습할 것)

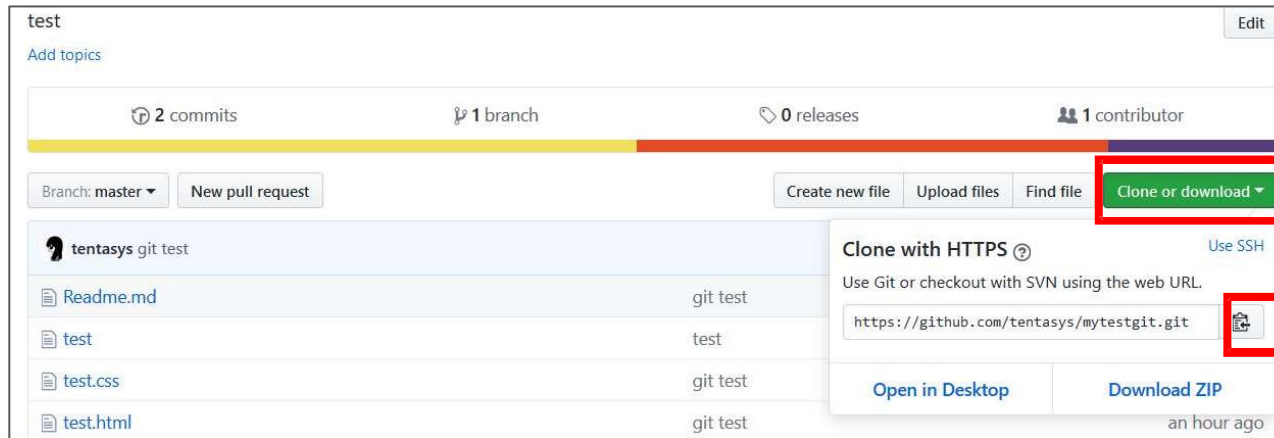
# 저장소 URL 가져오는 법

비어 있는  
repository  
의 경우



클릭하여 저장소 주소  
복사

내용이 있  
는 repository  
의 경우



클릭하여 저장소 주소  
복사

# 관련 명령어

- 참고 사이트 : <https://git-scm.com/book/ko/v2>

명령어	설명
<b>git clone</b>	<ul style="list-style-type: none"><li>클라이언트 상에 아무것도 없을 때 서버의 프로젝트를 내려 받는 명령어</li></ul>
<b>git remote</b> <b>git remote -v</b>	<ul style="list-style-type: none"><li>현재 프로젝트에 등록된 리모트 저장소의 이름과 url 확인</li><li>Clone 하면 `origin`이라는 리모트 저장소가 자동으로 등록</li></ul>
<b>git remote add &lt;name&gt; &lt;url&gt;</b>	<ul style="list-style-type: none"><li>기존 워킹 디렉토리에 새 리모트 저장소를 추가</li></ul>
<b>git remote show</b>	<ul style="list-style-type: none"><li>리모트 저장소의 이름 확인</li></ul>
<b>git remote rename &lt;old&gt; &lt;new&gt;</b>	<ul style="list-style-type: none"><li>리모트 저장소의 이름을 변경</li></ul>
<b>git remote remove &lt;name&gt;</b>	<ul style="list-style-type: none"><li>리모트 저장소를 삭제</li></ul>
<b>git push origin master</b>	<ul style="list-style-type: none"><li>원격 저장소 master branch에 commit을 저장</li><li>리모트 저장소에 쓰기 권한이 있고, Clone 하고 난 이후 아무도 원격 저장소에 Push 하지 않았을 때만 사용 가능</li><li>Clone 한 사람이 여러 명 있을 때, 다른 사람이 Push 한 후에 Push 하려고 하면 Push 할 수 없다.</li></ul>
<b>git fetch &lt;remote&gt;</b>	<ul style="list-style-type: none"><li>이 명령은 로컬에는 없지만, 리모트 저장소에는 있는 데이터를 모두 가져옴</li><li>자동으로 merge 하지는 않음</li></ul>

# 원격저장소(GitHub) Repository clone

- \$ git clone <https://github.com/jinsook64/gitHubTest.git>

```
keonhyo@BaekKeonHyo-Main MINGW64 ~/project2 (master)
$ git clone http://github.com/ghbaek64/tutorial.git
Cloning into 'tutorial'...
warning: redirecting to https://github.com/ghbaek64/tutorial.git/
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (6/6), done.
```

```
keonhyo@BaekKeonHyo-Main MINGW64 ~/project2 (master)
$ ls -l
total 2
-rw-r--r-- 1 keonhyo 197121 71 11월 13 17:52 sample01.txt
-rw-r--r-- 1 keonhyo 197121 47 11월 13 17:29 sample02.txt
drwxr-xr-x 1 keonhyo 197121  0 11월 13 20:29 tutorial/
```

각자 github에 생  
repository 경로 지



- clone 후 자동적으로 git 관리 설정이 됨(git init한 것과 같음)

```
dit@Melon-HP-Elite: ~/GitTest
dit@Melon-HP-Elite:~/GitTest$ ls -al
total 0
drwxrwxrwx 1 dit dit 4096 Nov 29 17:09 .
drwxr-xr-x 1 dit dit 4096 Nov 29 17:09 ..
drwxrwxrwx 1 dit dit 4096 Nov 29 17:09 .git
-rw-rw-rw- 1 dit dit   9 Nov 29 17:09 README.md
```

# 로컬 저장소(Local Repository) 관리

- \$ vi README.md // 기존 파일 수정
- md 파일은 mark down 형식의 파일임
  - 참고 : <https://gist.github.com/ihoneymon/652be052a0727ad59601>

```
1  ## GIT 공부를 위한 폴더
2
3  #### 즐거운 git 공부를 해봅시다^^
4
5  __개발자 목록__
6  -----
7  * 홍길동 팀
8      - 홍길동
9      - 임격정
10 * 춘향전 팀
11     - 이몽룡
12     - 월매
13     - 변학도
```

markdown 포맷으로 작성

## GIT 공부를 위한 폴더

즐거운 git 공부를 해봅시다^^

### 개발자 목록

- 홍길동 팀
  - 홍길동
  - 임격정
- 춘향전 팀
  - 이몽룡
  - 월매
  - 변학도

실행

- \$ git status // 아직 staging area(INDEX)에 들어가지 않음

```
MINGW64:/c/Users/keonhyo/project01/tutorial
keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/tutorial (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   ../myfolder/a1.txt
    new file:   sample01.txt
    new file:   sample02.txt

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   sample02.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    ../project2

keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/tutorial (master)
$
```

```
MINGW64:/c/Users/keonhyo/project01/tutorial
keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/tutorial (master)
$ git add *
warning: LF will be replaced by CRLF in tutorial/sample02.txt.
The file will have its original line endings in your working directory

keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/tutorial (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   ../myfolder/a1.txt
    new file:   sample01.txt
    new file:   sample02.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    ../project2

keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/tutorial (master)
$
```

- \$ git add \* // working directory의 모든 파일을 stage에 add

- 로컬 저장소(Local Repository) Commit
  - \$ git commit -m "baek이 수정후 v1.03을 올림 "

```
keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/tutorial (master)
$ git commit -m "baek이 수정 후 v1.03 을 올림 "
[master (root-commit) 71256a0] baek이 수정 후 v1.03 을 올림
3 files changed, 3 insertions(+)
create mode 100644 myfolder/a1.txt
create mode 100644 tutorial/sample01.txt
create mode 100644 tutorial/sample02.txt

keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/tutorial (master)
$
```

- 리모트 저장소(GitHub) Commit
  - \$ git push -u origin master 혹은
  - \$ git push --set-upstream origin master
  - > 그다음 commit 부터는
  - \$ git push

# [참고] Markdown

- Markdown이란?
  - 일반 텍스트 문서의 양식을 편집하는 태그 형식의 문법
  - 2004년에 존 그루버와 에런 스위츠가 개발함
  - README 파일이나 온라인 문서, 혹은 일반 텍스트 편집기로 문서 양식을 편집할 때 사용
  - 텍스트로 저장하기 때문에 Git을 통한 버전관리가 가능하고 용량이 적어 보관 용이
  - 핵심 문법을 제외하고는 표준이 없는 것이 단점

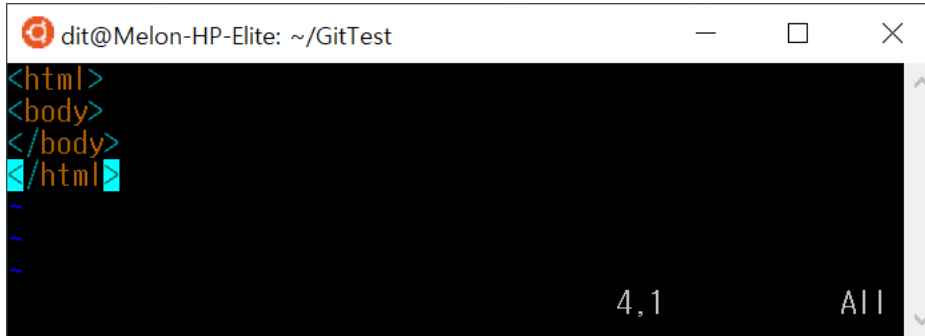
<https://dev-youngjun.tistory.com/51>

# [참조] Markdown

구분	설명	예제
제목	<h1>~ <h6>까지 #으로 제목 표현	# 제목1 ## 제목2 ...
강조	이탤릭 : 앞 뒤로 _ 또는 * 굵게 : 앞 뒤로 __ 또는 ** 취소선 : 앞 뒤로 ~	_Hello_ 또는 *Hello* __Hello__ 또는 **Hello** ~Hello~
목록	순서가 필요(번호를 같이 써준다) 들여쓰기로 서브목록 표현	1. 순서가 필요한 목록 - 서브목록1 - 서브목록2
	순서가 필요하기 않은 목록에 사용가능 기호: -, *, +	2. 순서가 필요한 목록 * 서브목록
링크	[...](...)	[google]( <a href="http://google.com">http://google.com</a> ) → google
수평선	각 기호(-, *, _)를 3개 이상 입력	--- *** ___

# 로컬 저장소(Local Repository) 관리

- \$ vi hello.html // 새로운 파일 hello.html 생성



```
dit@Melon-HP-Elite: ~/GitTest
<html>
<body>
</body>
</html>
~
~
4,1 All
```

HTML 파일은 각자가 나름대로 작성한다.

- \$ git status // 아직 staging area(INDEX)에 들어가지 않음

```
dit@Melon-HP-Elite: ~/GitTest
dit@Melon-HP-Elite:~/GitTest$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   README.md

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

        hello.html

no changes added to commit (use "git add" and/or "git commit -a")
```

- \$ git add \* // working directory의 모든 파일을 stage에 add

```
dit@Melon-HP-Elite: ~/GitTest
dit@Melon-HP-Elite:~/GitTest$ git add *
dit@Melon-HP-Elite:~/GitTest$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        modified:   README.md
        new file:   hello.html
```



- 로컬 저장소(Local Repository) Commit
  - \$ git commit -m "hello.html 파일이 추가되었습니다 "

```
dit@Melon-HP-Elite: ~/GitTest
dit@Melon-HP-Elite:~/GitTest$ git commit -m "hello.html 파일이 추가되었습니다."
[master 06cd737] hello.html 파일이 추가되었습니다.
2 files changed, 7 insertions(+), 1 deletion(-)
create mode 100644 hello.html
dit@Melon-HP-Elite:~/GitTest$
```

- 리모트 저장소(GitHub) Commit
  - \$ git push
  - 이때 자신의 GitHub ID, Password 물어본다.

```
dit@Melon-HP-Elite: ~/GitTest
dit@Melon-HP-Elite:~/GitTest$ git push
Username for 'https://github.com': jhkim3217
Password for 'https://jhkim3217@github.com':
Counting objects: 4, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 418 bytes | 418.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0)
To https://github.com/jhkim3217/GitTest.git
8cbb393..06cd737 master -> master
dit@Melon-HP-Elite:~/GitTest$
```

# GitHub Repository 확인

The screenshot displays the GitHub interface for the repository `jhkim3217/GitTest`. The browser address bar shows `github.com/jhkim3217/GitTest`. The repository page includes a search bar, navigation links (Pull requests, Issues, Marketplace, Explore), and repository statistics (Unwatch 1, Star 0, Fork 0). The repository description states "No description, website, or topics provided." Below this, a summary bar shows 2 commits, 1 branch, 0 packages, 0 releases, and 1 contributor. The file list on the left shows `README.md` and `hello.html`, both with the commit message "hello.html 파일이 추가되었습니다." and a timestamp of "6 minutes ago". The README content is displayed below the file list, showing the title "GitTest" and the text "GitTest 원격 저장소를 활용해 봅시다."

jhkim3217 / GitTest

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security Insights Settings

No description, website, or topics provided. Edit

Manage topics

2 commits 1 branch 0 packages 0 releases 1 contributor

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

jhkim3217 hello.html 파일이 추가되었습니다. Latest commit 06cd737 6 minutes ago

File	Commit Message	Time
README.md	hello.html 파일이 추가되었습니다.	6 minutes ago
hello.html	hello.html 파일이 추가되었습니다.	6 minutes ago

README.md

## GitTest

GitTest 원격 저장소를 활용해 봅시다.

© 2019 GitHub, Inc. Terms Privacy Security Status Help Contact GitHub Pricing API Training Blog About

GitTest/hello.html at master · jhkim3217 · jhkim3217

github.com/jhkim3217/GitTest/blob/master/hello.html

Search or jump to...

Pull requestsIssuesMarketplaceExplore

CodeIssues 0Pull requests 0Projects 0WikiSecurityPulseCommunity

Branch: master ▾GitTest / hello.html

Find fileCopy path

> Octotree

jhkim3217 hello.html 파일이 추가되었습니다.06cd737 29 minutes ago

1 contributor

4 lines (4 sloc) | 30 Bytes

RawBlameHistory

1<html>

2<body>

3</body>

4</html>

© 2019 GitHub, Inc. TermsPrivacySecurityStatusHelp

Contact GitHubPricingAPITrainingBlogAbout

# Commit message 확인

The screenshot shows a web browser window displaying the GitHub repository page for `jhkim3217/GitTest`. The browser's address bar shows the URL `github.com/jhkim3217/GitTest/commits/master`. The GitHub navigation bar at the top includes a search bar, links for Pull requests, Issues, Marketplace, and Explore, and a user profile icon. Below the navigation bar, the repository name `jhkim3217 / GitTest` is displayed, along with buttons for Unwatch (1), Star (0), and Fork (0). A secondary navigation bar contains links for Code, Issues (0), Pull requests (0), Actions, Projects (0), Wiki, Security, Insights, and Settings. The main content area shows the commit history for the `master` branch. It features a sidebar with a calendar view for October and a list of commits. The most recent commit, dated December 1, 2019, has the message "hello.html 파일이 추가되었습니다." (The file hello.html was added.) and was committed by `jhkim3217` 30 minutes ago. The commit hash is `06cd737`. Below it, the "Initial commit" from November 29, 2019, is shown, committed by `jhkim3217` 2 days ago, with a verified status and hash `8cbb393`. At the bottom of the commit list are "Newer" and "Older" buttons. The footer of the page contains copyright information for GitHub, Inc. and various links like Terms, Privacy, Security, Status, Help, Contact GitHub, Pricing, API, Training, Blog, and About.

Commits · jhkim3217/GitTest

github.com/jhkim3217/GitTest/commits/master

Search or jump to... Pull requests Issues Marketplace Explore

jhkim3217 / GitTest

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security Insights Settings

Branch: master

Octobree

Commits on Dec 1, 2019

hello.html 파일이 추가되었습니다.  
jhkim3217 committed 30 minutes ago

Commits on Nov 29, 2019

Initial commit  
jhkim3217 committed 2 days ago

Verified 8cbb393

Newer Older

© 2019 GitHub, Inc. Terms Privacy Security Status Help

Contact GitHub Pricing API Training Blog About

# Commit hash 코드(06cd737) 변경사항 확인

The screenshot displays a GitHub commit page for the repository `jhkim3217 / GitTest`. The commit hash is `06cd737d2acf9ed1e805549b2d3550321590df44`, made 31 minutes ago. The commit message is "hello.html 파일이 추가되었습니다." (hello.html file added).

The commit shows two changed files:

- `README.md`: A diff view showing the addition of a new section titled `# GitTest` with a description in Korean: `## GitTest 원격 저장소를 활용해 봅시다.`
- `hello.html`: A diff view showing the addition of a new HTML file with the following content:

```
<html>
<body>
</body>
</html>
```

The page also includes a comment section at the bottom with the text "0 comments on commit 06cd737" and a "Lock conversation" button.

# 실습 - github 을 이용한 clone 기능

1. github(<http://github.com>) 에 자신의 계정 생성.
2. 원격저장소 Repository(이름: web2) 생성.
3. README.md에 웹개발기초II에 대한 설명 입력
4. clone 으로 가져오기

```
keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/tutorial (master)
$ pwd
/c/Users/keonhyo/project01/tutorial

keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/tutorial (master)
$ cd ..

keonhyo@BaekKeonHyo-Main MINGW64 ~/project01 (master)
$ git clone http://github.com/ghbaek64/travel.git
Cloning into 'travel'...
warning: redirecting to https://github.com/ghbaek64/travel.git/
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (6/6), done.

keonhyo@BaekKeonHyo-Main MINGW64 ~/project01 (master)
$ ls
myfolder/  project2  travel/   tutorial/

keonhyo@BaekKeonHyo-Main MINGW64 ~/project01 (master)
$ cd travel

keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/travel (main)
$ ls
test01.txt  test02.txt

keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/travel (main)
$ cat test01.txt
this file name is test01.txt
created 20/11/15 day, by baek

keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/travel (main)
$
```

# 실습 - github 을 이용한 push 기능

1. clone으로 가져온 git repository 에 웹개발II (이름: web2) 에서 실습한 코드를 2개 이상 가져다 둔다.
2. 파일들을 staging area에 올린다.
3. 파일들을 버전으로 만든다.(commit)
4. remote repository에 push 한다.
5. github에서 확인한다.

```
keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/travel02 (master)
$ git add *.*

keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/travel02 (master)
$ git commit -m "15 baek이 수정함"
[master 170117f] 15 baek이 수정함
2 files changed, 4 insertions(+)
create mode 100644 travel02/test01.txt
create mode 100644 travel02/test02.txt

keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/travel02 (master)
$ git push --set-upstream http://github.com/ghbaek64/travel02 master
warning: redirecting to https://github.com/ghbaek64/travel02/
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 525 bytes | 262.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To http://github.com/ghbaek64/travel02
  71256a0..170117f  master -> master
Branch 'master' set up to track remote branch 'master' from 'http://gi
hbaek64/travel02'.

keonhyo@BaekKeonHyo-Main MINGW64 ~/project01/travel02 (master)
$ .....
```

# clone 하지 않고 GitHub 활용하기

- GitHub 원격 저장소(repository) 생성 : githib.com/사용자\_아이디/GitTest2

Create a new repository

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

Owner: jhkim3217 / Repository name: GitTest2 ✓

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

Description (optional)

☒ Public  
Anyone 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

© 2019 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Help](#) [Contact GitHub](#) [Pricing](#) [API](#) [Train](#)

jhkim3217 / GitTest2

Unwatch 1 Star

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security Insights Settings

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

Set up in Desktop or HTTPS SSH <https://github.com/jhkim3217/GitTest2.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 "# GitTest2" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/jhkim3217/GitTest2.git
git push -u origin master
```

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

```
git remote add origin https://github.com/jhkim3217/GitTest2.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

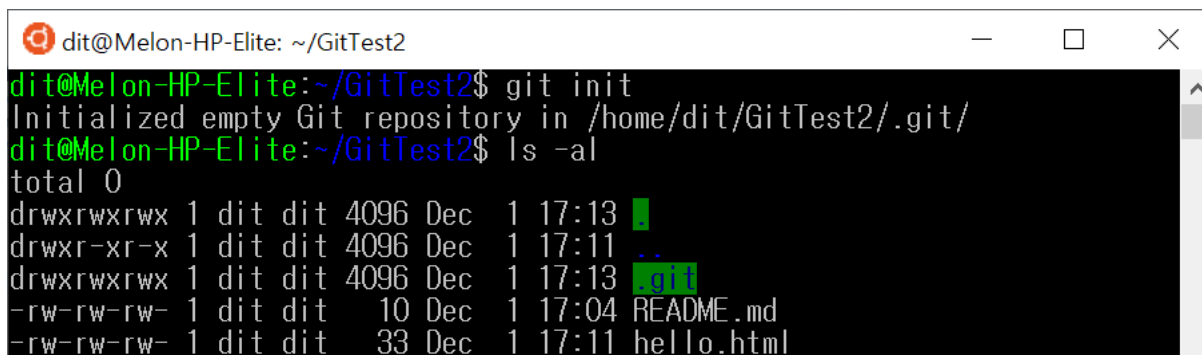


# 로컬 저장소(local repository) 생성 및 관리

- 로컬 저장소 워킹 디렉토리 생성 및 이동
  - \$ mkdir GitTest2
  - \$ cd GitTest2
- 워킹 디렉터리에 파일 작성
  - \$ echo "# GitTest2" >> README.md
  - \$ vi hello.html
- Git 초기화
  - \$ git init

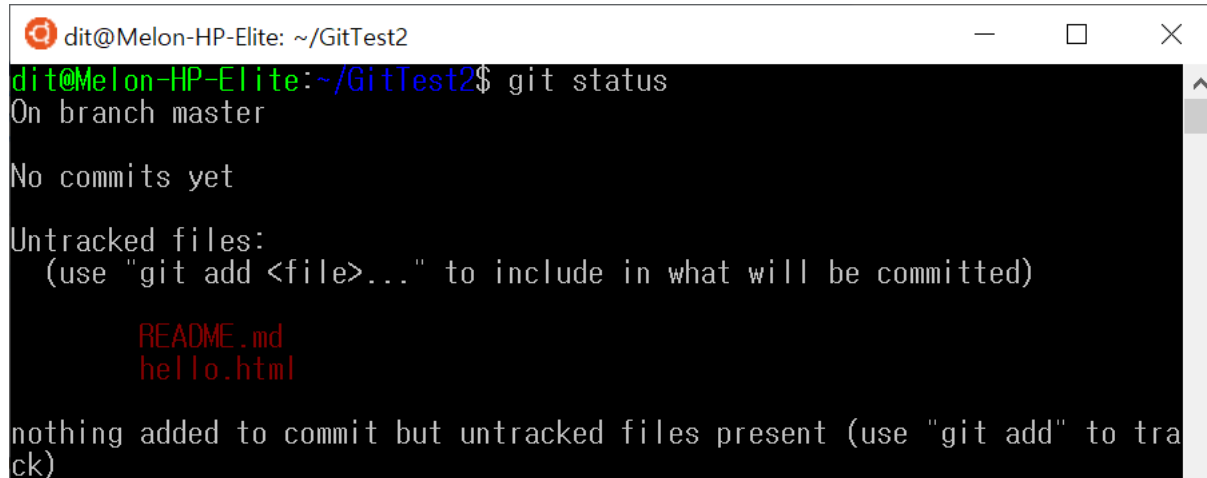


```
<html>
  <body>
</body>
</html>
```



```
dit@Melon-HP-Elite: ~/GitTest2
dit@Melon-HP-Elite:~/GitTest2$ git init
Initialized empty Git repository in /home/dit/GitTest2/.git/
dit@Melon-HP-Elite:~/GitTest2$ ls -al
total 0
drwxrwxrwx 1 dit dit 4096 Dec  1 17:13 .
drwxr-xr-x 1 dit dit 4096 Dec  1 17:11 ..
drwxrwxrwx 1 dit dit 4096 Dec  1 17:13 .git
-rw-rw-rw- 1 dit dit  10 Dec  1 17:04 README.md
-rw-rw-rw- 1 dit dit  33 Dec  1 17:11 hello.html
```

- \$ git status

A terminal window titled 'dit@Melon-HP-Elite: ~/GitTest2' showing the output of the 'git status' command. The output indicates the user is on the 'master' branch, there are no commits yet, and there are untracked files named 'README.md' and 'hello.html'. It also provides instructions on how to use 'git add' to track these files.

```
dit@Melon-HP-Elite: ~/GitTest2$ git status
On branch master

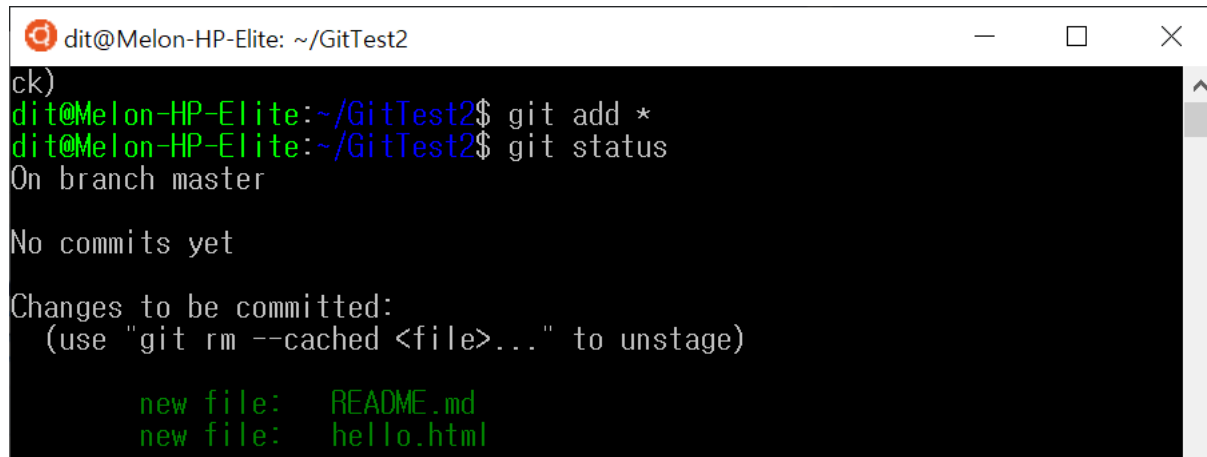
No commits yet

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

        README.md
        hello.html

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

- Staging area에 넣기 : 추적을 위한 INDEX(스냅샷)이 생성됨
  - \$ git add \*

A terminal window titled 'dit@Melon-HP-Elite: ~/GitTest2' showing the output of 'git add \*' followed by 'git status'. The output shows that the files 'README.md' and 'hello.html' have been staged for commit, appearing as 'new file:'.

```
dit@Melon-HP-Elite: ~/GitTest2$ git add *
dit@Melon-HP-Elite: ~/GitTest2$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)


        new file:   README.md
        new file:   hello.html
```

- 로컬 저장소에 commit 하기
  - \$ git commit -m "hello.html 파일 추가 "


```
dit@Melon-HP-Elite: ~/GitTest2
dit@Melon-HP-Elite:~/GitTest2$ git commit -m "hello.html 파일 추가"
[master (root-commit) 26e8342] hello.html 파일 추가
2 files changed, 6 insertions(+)
create mode 100644 README.md
create mode 100644 hello.html
dit@Melon-HP-Elite:~/GitTest2$ git status
On branch master
nothing to commit, working tree clean
dit@Melon-HP-Elite:~/GitTest2$
```

- 원격 저장소(GitHub) 연결
  - origin 이란 이름(alias)으로 GitHub 원격저장소 연결(다른 이름도 가능)
  - \$ git remote add origin https://github.com/jhkim3217/GitTest2.git

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

 Set up in Desktop or **HTTPS** **SSH** <https://github.com/jhkim3217/GitTest2.git>

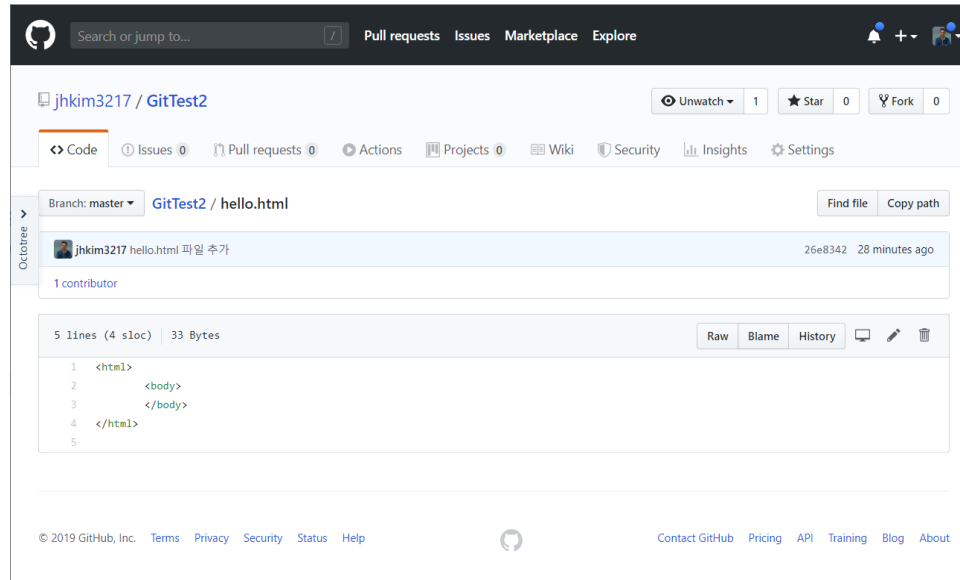
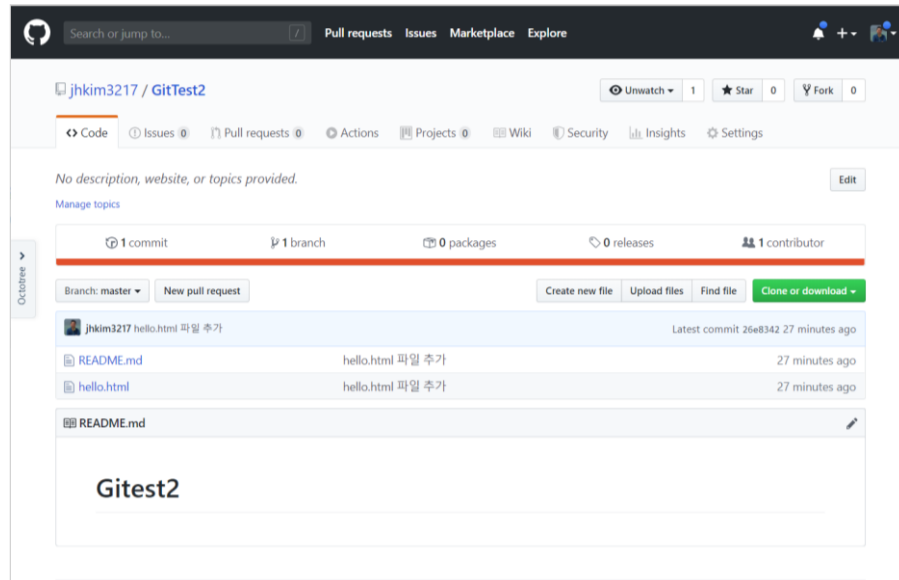
Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

  
크리  
21

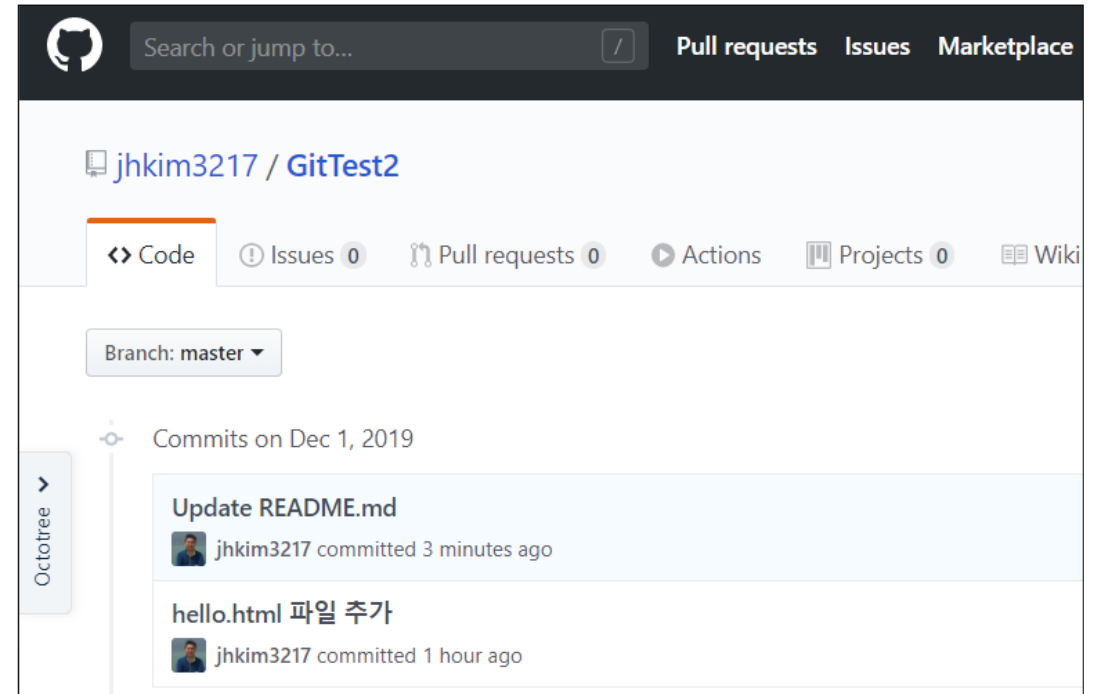
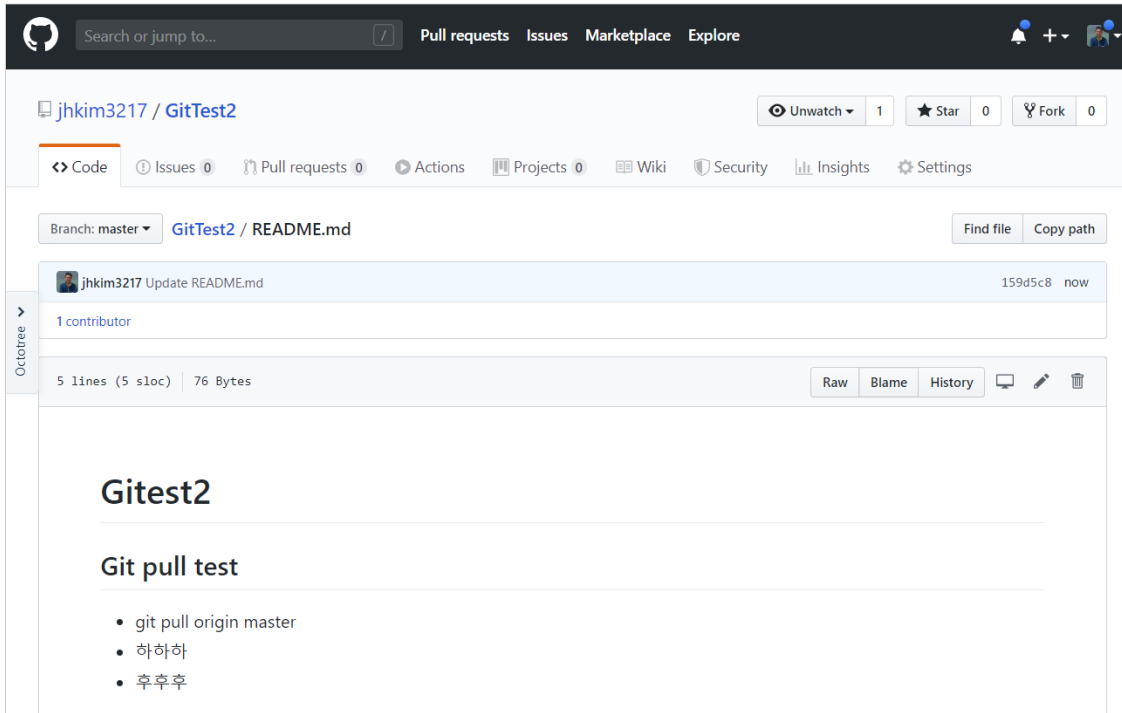
- 원격 저장소(GitHub)에 push 하기
  - master는 로컬 저장소의 이름
  - \$ git push -u origin master

```
Username for 'https://github.com': jhkim3217
Password for 'https://jhkim3217@github.com':
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

- 원격 저장소(GitHub) 확인



- 원격 저장소(GitHub) pull 하기
  - GitHub 원격 저장소 README.md 파일 수정




- 원격 저장소(GitHub) pull 하기

- \$ git remote add origin https://github.com/jhkim3217/GitTest2.git
- \$ git pull origin master

앞에서 remote 연결을 한  
상태면 이 명령을 사용할 필  
요가 없음

```
dit@Melon-HP-Elite:~/03GitTest02$ git pull origin master
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/jhkim3217/03GitTest2
 * branch                master      -> FETCH_HEAD
    233d384..619ca91      master      -> origin/master
Updating 233d384..619ca91
Fast-forward
 README.md | 2 ++
 1 file changed, 2 insertions(+)
dit@Melon-HP-Elite:~/03GitTest02$ cat README.md
# GitTest2
## Git pull test
- git pull origin master
```

 선택 dit@Melon-HP-Elite: ~/GitTest3

```
dit@Melon-HP-Elite:~/GitTest3$  
dit@Melon-HP-Elite:~/GitTest3$  
dit@Melon-HP-Elite:~/GitTest3$ ls -al  
total 0  
drwxrwxrwx 1 dit dit 4096 Dec  1 18:29 .  
drwxr-xr-x 1 dit dit 4096 Dec  1 18:23 ..  
drwxrwxrwx 1 dit dit 4096 Dec  1 18:29 .git  
-rw-rw-rw- 1 dit dit   76 Dec  1 18:29 README.md  
-rw-rw-rw- 1 dit dit   33 Dec  1 18:29 hello.html  
dit@Melon-HP-Elite:~/GitTest3$ cat README.md  
# Gittest2  
## Git pull test  
- git pull origin master  
- 하하하  
- 후후후
```

# Git/GitHub 공부 좋은 동영상 추천

- <https://www.youtube.com/watch?v=tQa7DoaFaxM>