



경성대학교  
KYUNGSUNG UNIVERSITY

# 깃과 친해지기

이름 : 전상훈

학번 : 2019775054

# 목차

---

001	깃 설치
002	깃으로 파일관리 하기
003	VS코드와 깃 연동
004	Branch 사용해보기
004	깃 소스트리 사용해보기

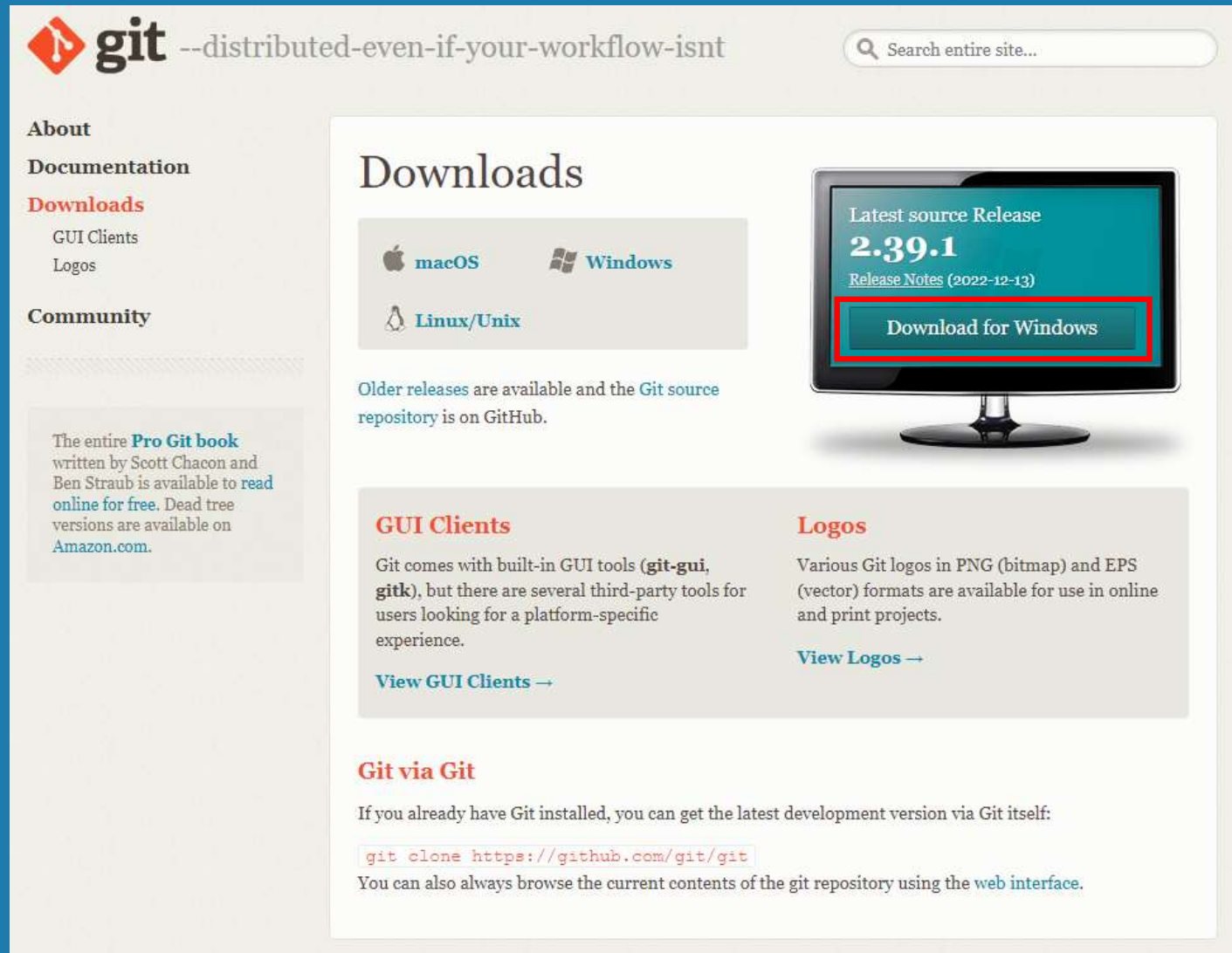
# Part 1,

---

## 깃 설치

# 깃 설치

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



The screenshot shows the Git Downloads page. The header includes the Git logo and the tagline "--distributed-even-if-your-workflow-isnt", along with a search bar. The left sidebar contains links for "About", "Documentation", "Downloads" (highlighted in red), "GUI Clients", "Logos", and "Community". Below the sidebar, there is a text block about the "Pro Git book". The main content area is titled "Downloads" and features a large monitor graphic displaying the "Latest source Release 2.39.1" with a red box around the "Download for Windows" button. Below the monitor, there are sections for "GUI Clients" and "Logos". At the bottom, there is a section titled "Git via Git" with a code block for cloning the repository and a link to the web interface.

**git** --distributed-even-if-your-workflow-isnt

Search entire site...

**About**

**Documentation**

**Downloads**

- GUI Clients
- Logos

**Community**

The entire **Pro Git book** written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

## Downloads

Older releases are available and the Git source repository is on GitHub.

Latest source Release  
**2.39.1**  
[Release Notes \(2022-12-13\)](#)  
**Download for Windows**

**GUI Clients**

Git comes with built-in GUI tools (**git-gui**, **gitk**), but there are several third-party tools for users looking for a platform-specific experience.

[View GUI Clients →](#)

**Logos**

Various Git logos in PNG (bitmap) and EPS (vector) formats are available for use in online and print projects.

[View Logos →](#)

### Git via Git

If you already have Git installed, you can get the latest development version via Git itself:

```
git clone https://github.com/git/git
```

You can also always browse the current contents of the git repository using the [web interface](#).

# 깃 설치



 **git** --distributed-even-if-your-workflow-isnt

Search entire site...

**About**

**Documentation**

**Downloads**

- GUI Clients
- Logos

**Community**

The entire [Pro Git book](#) written by Scott Chacon and Ben Straub is available to read [online for free](#). Dead tree versions are available on [Amazon.com](#).

## Download for Windows

[Click here to download](#) the latest (2.39.1) 64-bit version of **Git for Windows**. This is the most recent [maintained build](#). It was released **1 day ago**, on 2023-01-17.

### Other Git for Windows downloads

**Standalone Installer**

- [32-bit Git for Windows Setup.](#)
- [64-bit Git for Windows Setup.](#)

**Portable ("thumbdrive edition")**

- [32-bit Git for Windows Portable.](#)
- [64-bit Git for Windows Portable.](#)

**Using winget tool**

# 깃 설치 확인

```
C:\Users\User>git
usage: git [-v | --version] [-h | --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>]
          [--super-prefix=<path>] [--config-env=<name>=<envvar>]
          <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

가장 정확



Git Bash

앱

학교 및 웹 검색

git bash - 학교 및 웹 결과 보기 >

git bash **download** >

git bash **설치** >

git bash **사용법** >

git bash **명령어** >

git bash **windows** >

git bash **다운로드** >

git bash **commands** >

명령

git bash >

git bash

# Part 2,

---

## 깃으로 파일 관리하기

# Git hub 접속

The screenshot shows a GitHub profile page for user `jeonsanghoon711`. The **Repositories** tab is selected and highlighted with a red box. The profile includes a yellow GitHub logo avatar, an `jeonsanghoon711` username, and an `Edit profile` button.

**Popular repositories**

- `SlidingPuzzle-java` (Public, Java)
- `test` (Public)

**11 contributions in the last year**

Contribution calendar showing activity for the last year (Jan 2022 to Jan 2023). Contributions are marked with green squares on specific days.

**Contribution activity**

January 2023

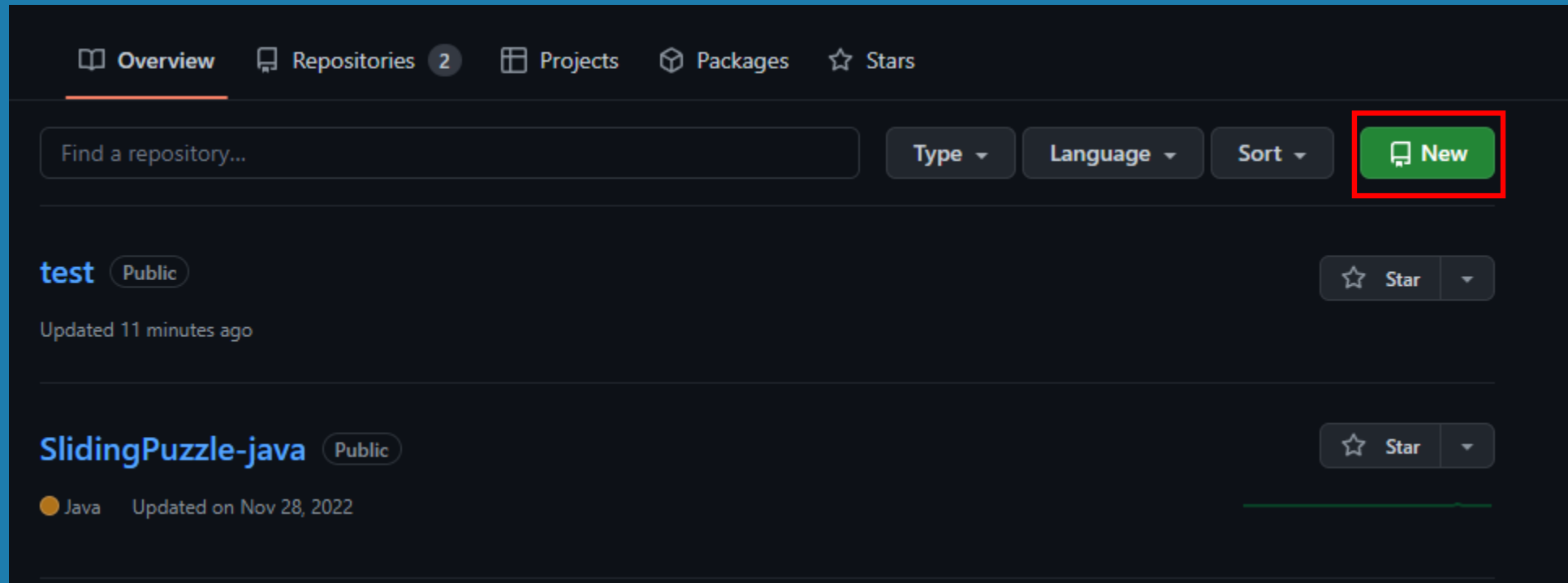
- Created 1 repository
  - `jeonsanghoon711/test` (Jan 19)

[Show more activity](#)

Seeing something unexpected? Take a look at the [GitHub profile guide](#).



# Git hub 접속




# Git hub 접속

## Create a new repository

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

Owner \*

Repository name \*


 jeonsanghoon711 ▾

 / 


test1 ✓

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

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.

**Initialize this repository with:**

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

☐ **Add a README file**

This is where you can write a long description for your project. [Learn more.](#)

**Add .gitignore**


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

**.gitignore template: None ▾**

**Choose a license**

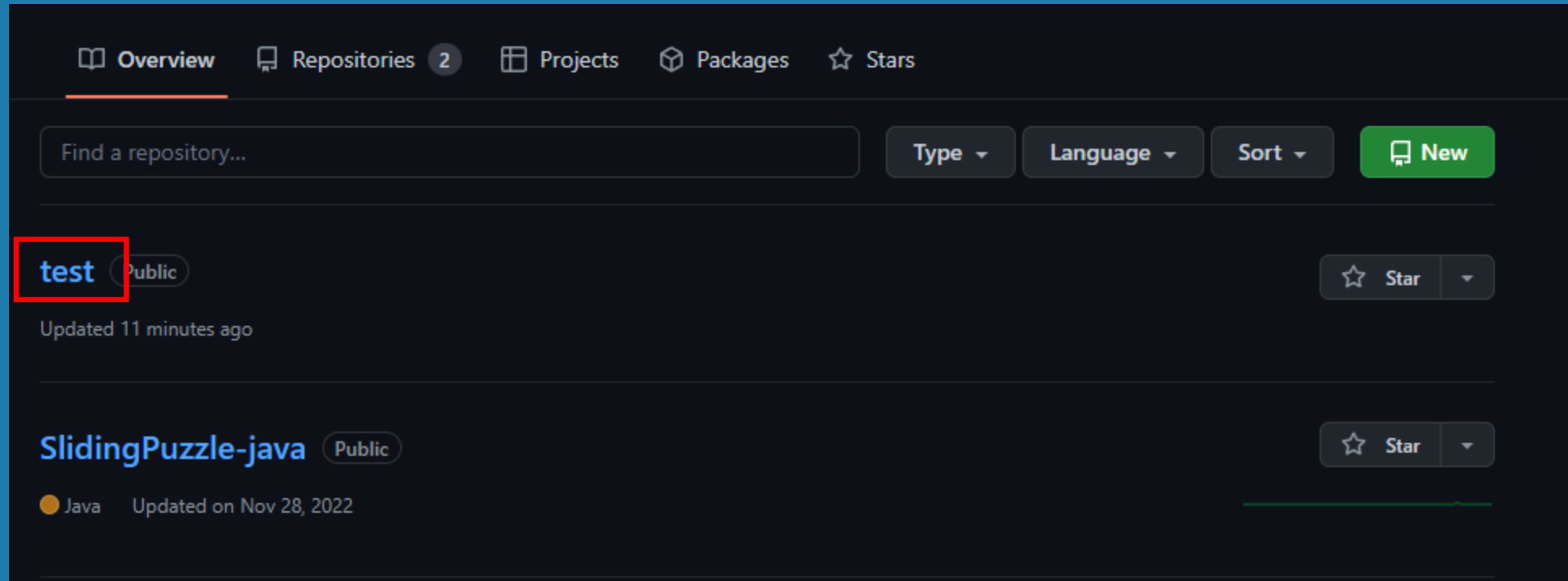
A license tells others what they can and can't do with your code. [Learn more.](#)

**License: None ▾**

 You are creating a public repository in your personal account.


Create repository

# Git hub 접속



# Git hub 접속

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

 Set up in Desktop

or

**HTTPS**

SSH

git@github.com:jeonsanghoon711/test.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 "# test" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin git@github.com:jeonsanghoon711/test.git
git push -u origin main
```



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

```
git remote add origin git@github.com:jeonsanghoon711/test.git
git branch -M main
git push -u origin main
```



## ...or import code from another repository

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

[Import code](#)

# Git bash 실행

깃 복사 단축키 : Ctrl + Ins  
깃 붙여넣기 : Shift + Ins

```
git config -global init.defaultbranch main
```

```
git config --global user.name "사용자이름"
```

```
User@DESKTOP-M960RTM MINGW64 ~  
$ git config --global user.name "사 용 자 이 름"
```

```
git config --global user.email "사용자이메일@...com"
```

```
User@DESKTOP-M960RTM MINGW64 ~  
$ git config --global user.email "사 용 자 이 메 일 @...com"
```

```
git config --list
```

```
User@DESKTOP-M960RTM MINGW64 ~  
$ git config --list  
diff.astextplain.textconv=astextplain  
filter.lfs.clean=git-lfs clean -- %f  
filter.lfs.smudge=git-lfs smudge -- %f  
filter.lfs.process=git-lfs filter-process  
filter.lfs.required=true  
http.sslbackend=openssl  
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt  
core.autocrlf=true  
core.fscache=true  
core.symlinks=false  
pull.rebase=false  
credential.helper=manager-core  
credential.https://dev.azure.com.usehttppath=true  
init.defaultbranch=master  
user.email=wjstkdgns711@naver.com  
user.name=JeonSangHoon711
```

# Git bash 실행

git init

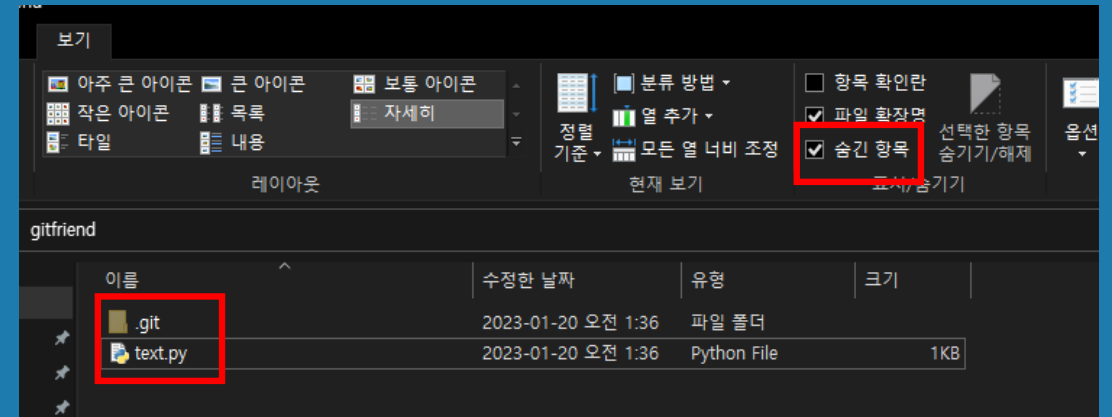
```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/gitfriend
$ git init
Initialized empty Git repository in C:/Users/User/Desktop/gitfriend/.git/
```

git remote add <원격저장소명> <url>

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/gitfriend (master)
$ git remote add test https://github.com/jeonsanghoon711/test.git
```

git remote -v

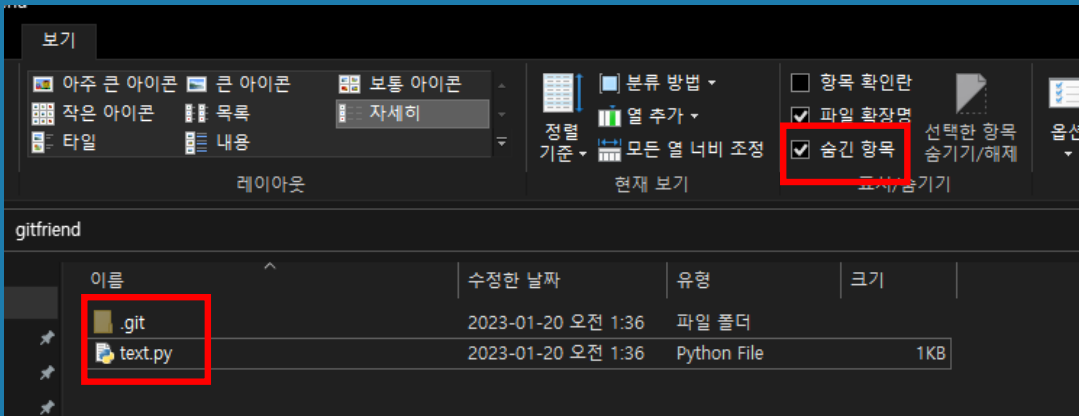
```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/gitfriend (master)
$ git remote -v
test    https://github.com/jeonsanghoon711/test.git (fetch)
test    https://github.com/jeonsanghoon711/test.git (push)
```



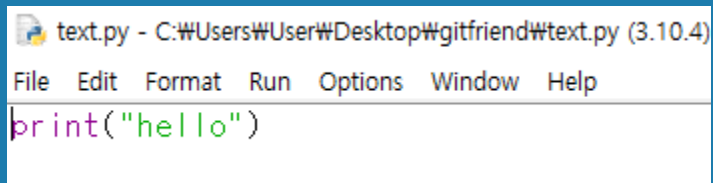
테스트용 파이썬 파일 생성

# Git Push

보기 - 숨김 항목 체크 - .git 파일 확인



테스트용 파이썬 파일 생성



git add “파일이름”

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/gitfriend (master)
$ git add text.py
```

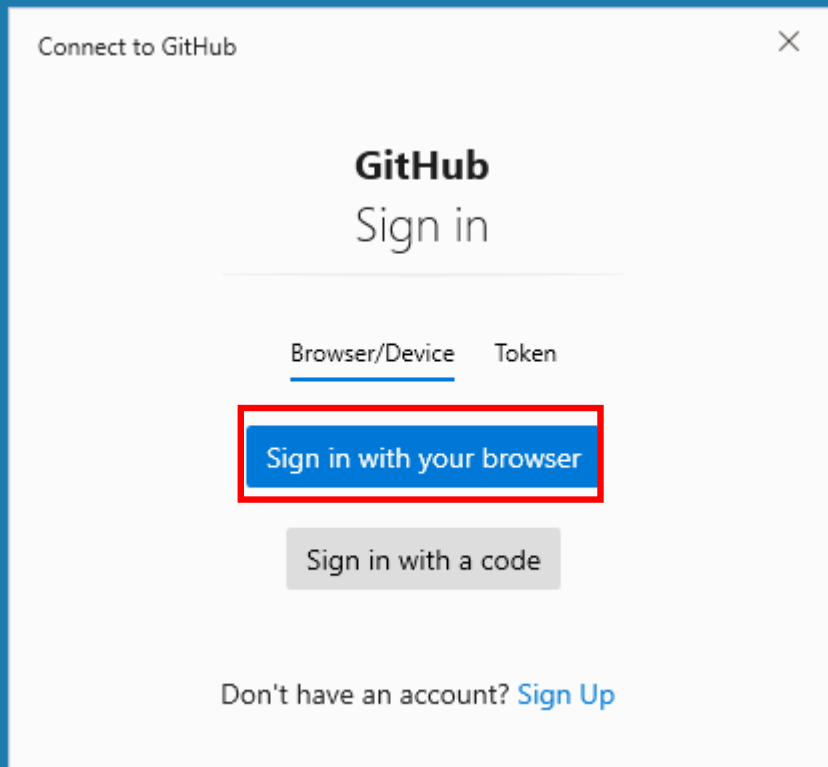
git commit -m “내용”

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/gitfriend (master)
$ git commit -m "test"
[master (root-commit) 05f3acd] test
1 file changed, 1 insertion(+)
create mode 100644 text.py
```

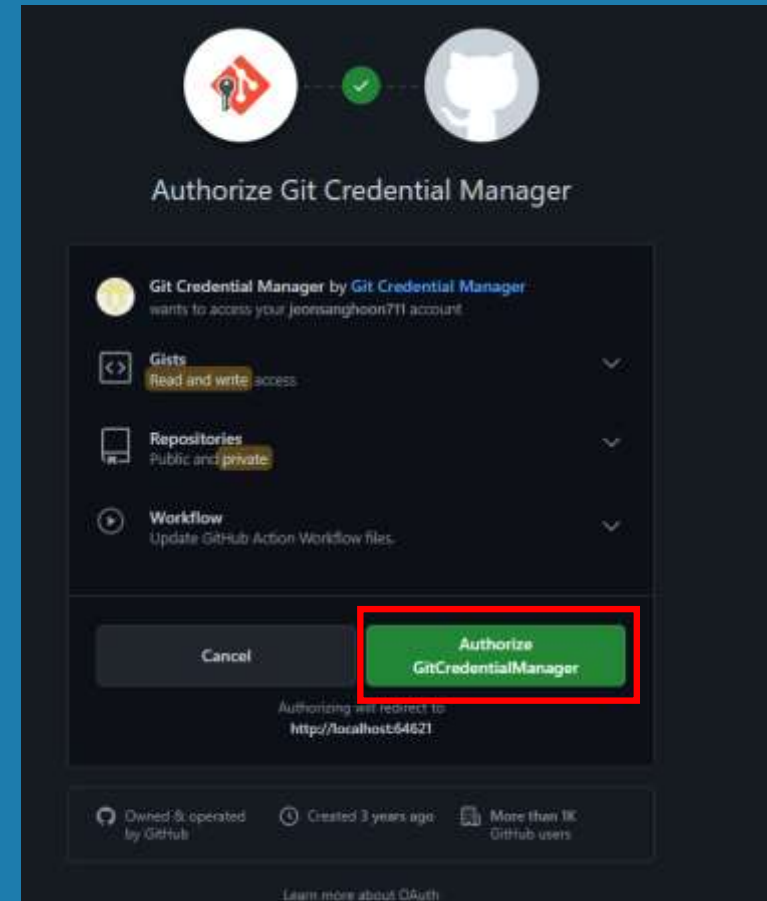
# Git Push

`git push <원격저장소명> <브랜치명>`

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/friendgit (main)
$ git push test main
```



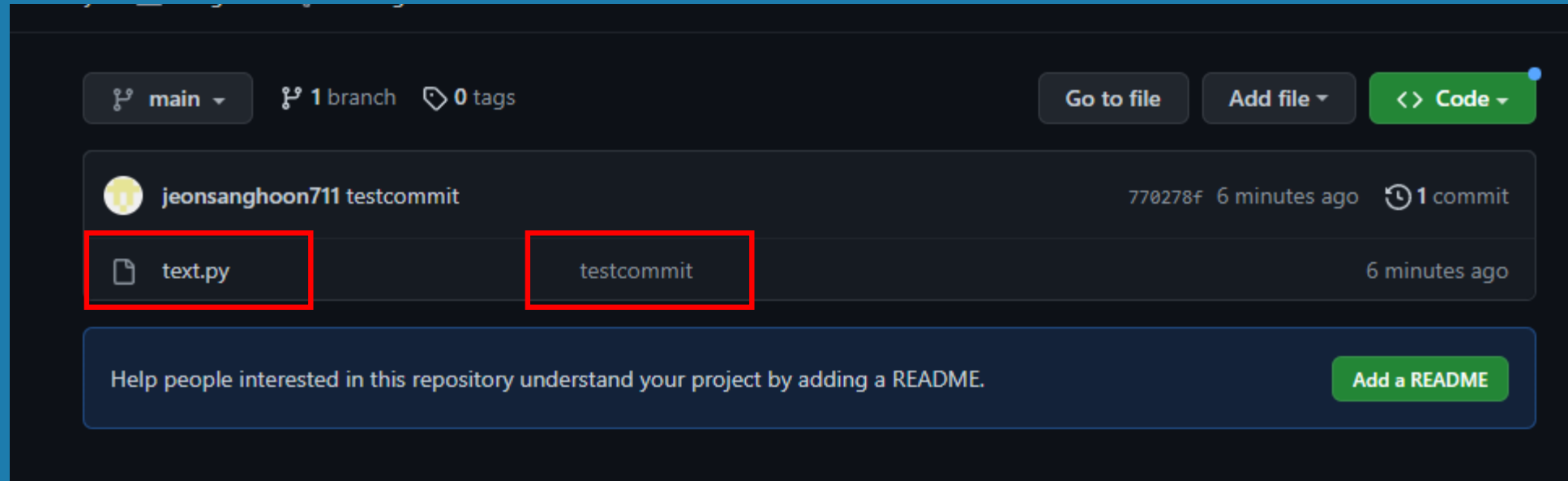
로그인





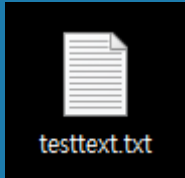
# Git Push

## 파일 업로드 확인

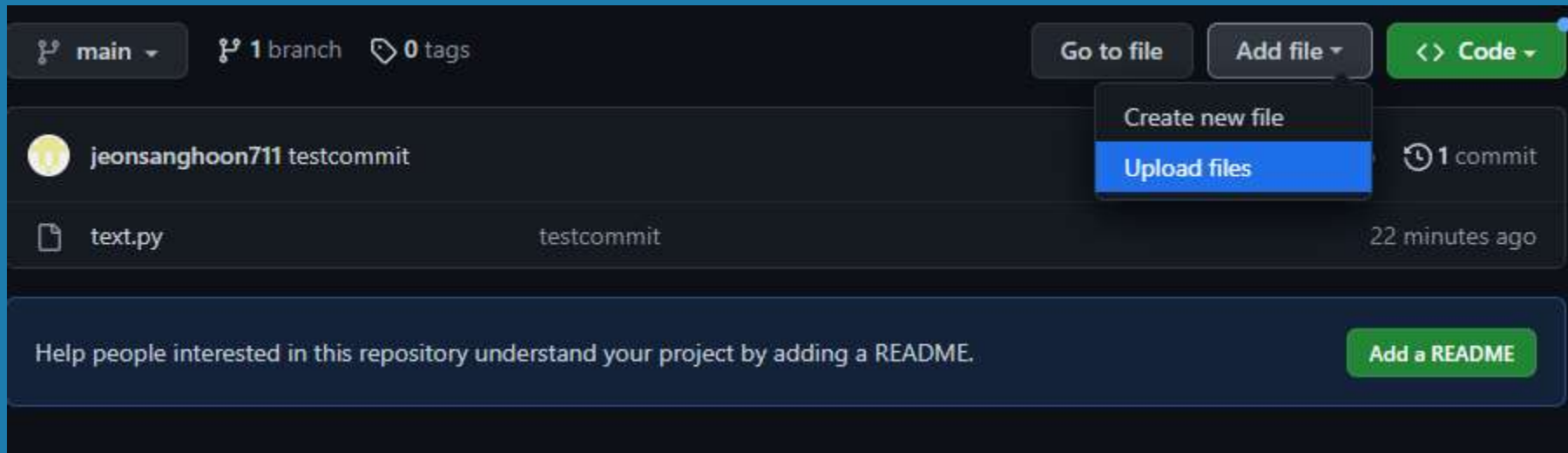


# Git Pull

바탕화면에 테스트용 파일 생성



깃허브 업로드

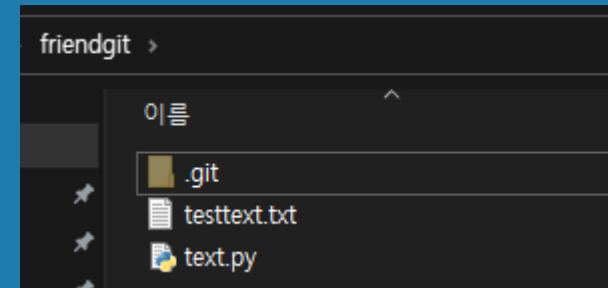


# Git Pull

`git pull <원격저장소명> <브랜치명>`

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/friendgit (main)
$ git pull test main
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 676 bytes | 135.00 KiB/s, done.
From https://github.com/jeonsanghoon711/test
* branch          main      -> FETCH_HEAD
   770278f..4d736ec main      -> test/main
Updating 770278f..4d736ec
Fast-forward
 testtext.txt | 1 +
1 file changed, 1 insertion(+)
create mode 100644 testtext.txt
```

디렉토리 파일 확인



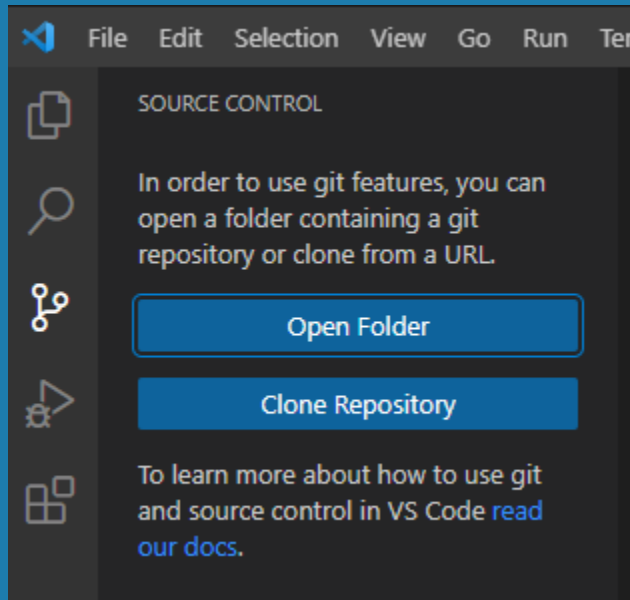
# Part 3,

---

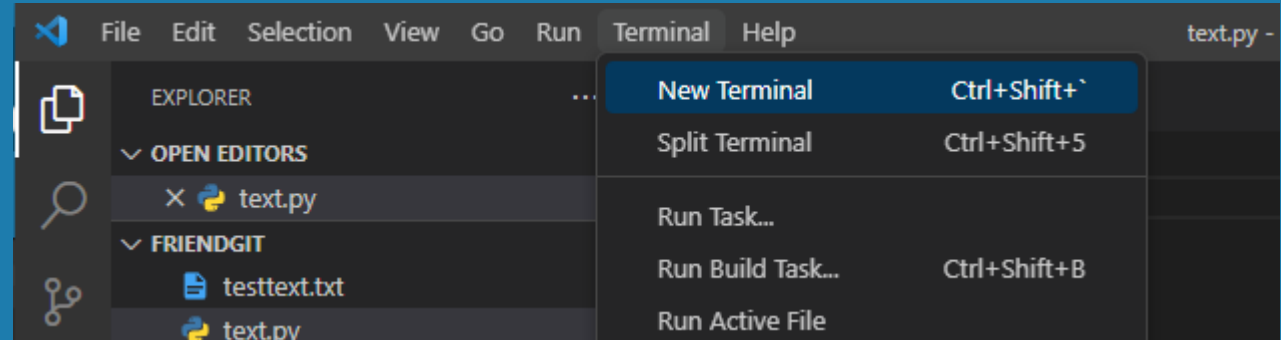
## VS코드와 깃 연동

# Vs코드와 깃

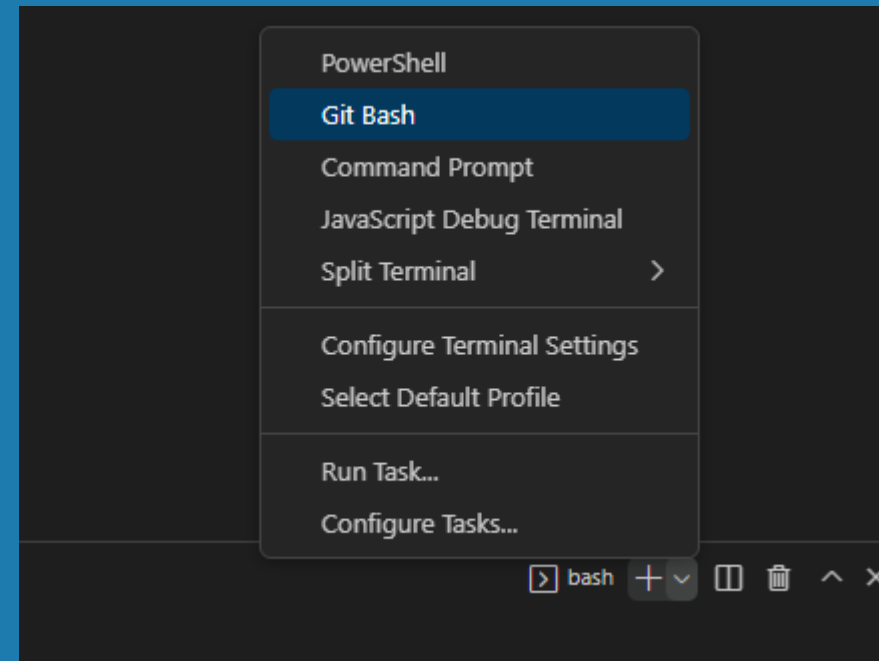
## Open Folder



## Terminal – New Terminal

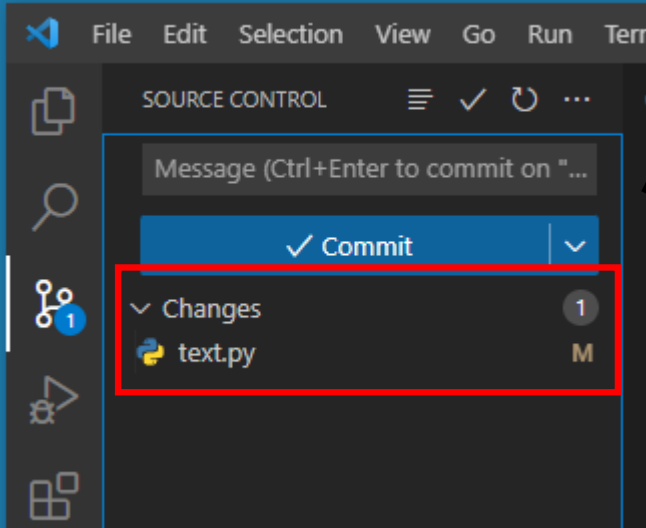


## (우측하단) + -> Git Bash



# Vs코드 Push

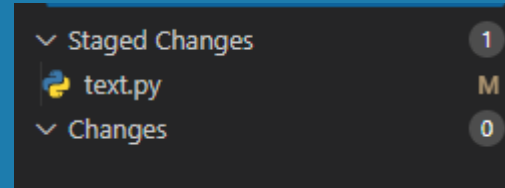
테스트 파일에 `print("이름")` 추가  
-> Changes 에 파일 있는지 확인



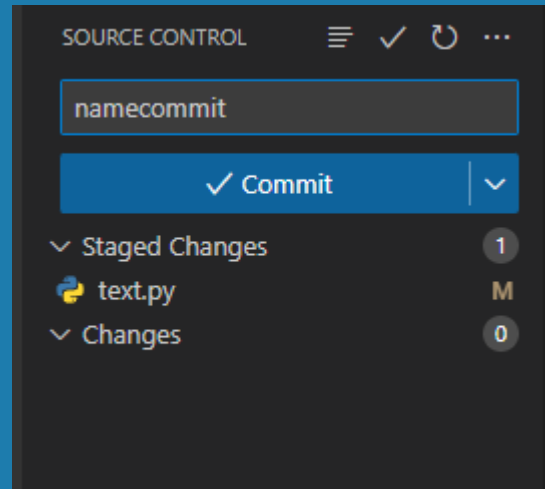
git add "파일명"

-> Staged Changes 이동 확인

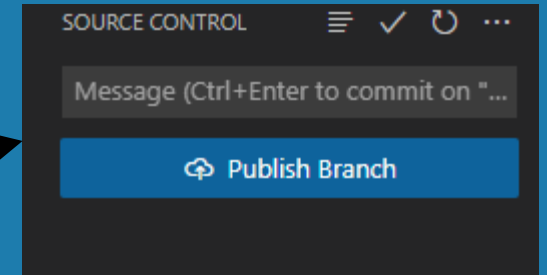
-> git push 할 때 Staged Changes에 있는 파일들이 Push 됨



Commit 입력 후 Commit 버튼 클릭

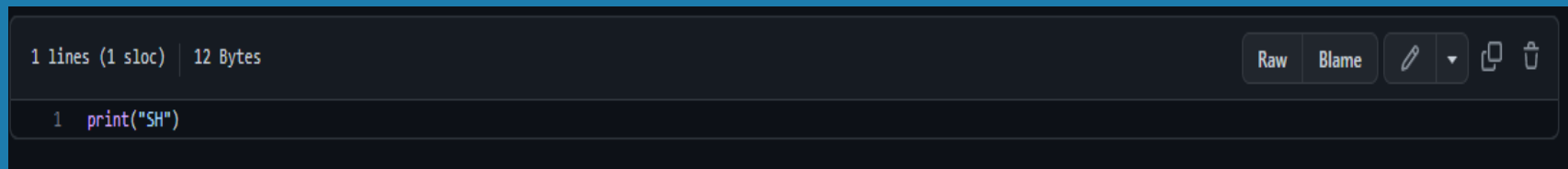
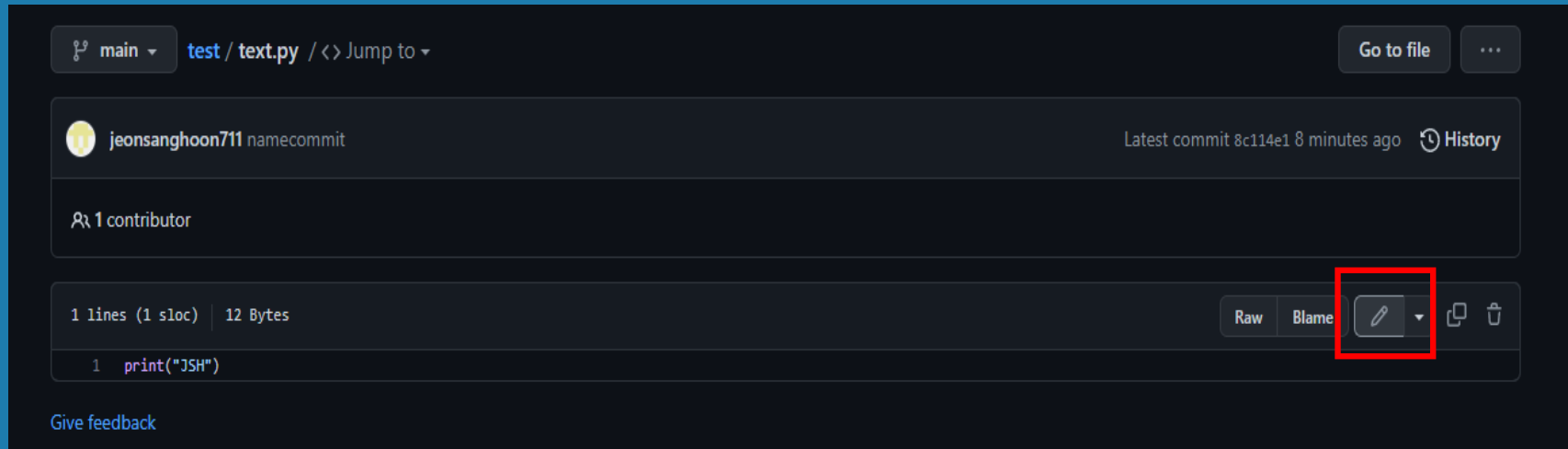


Publish Branch 클릭



# Vs코드 Pull

## 깃 허브 에서 파일 수정



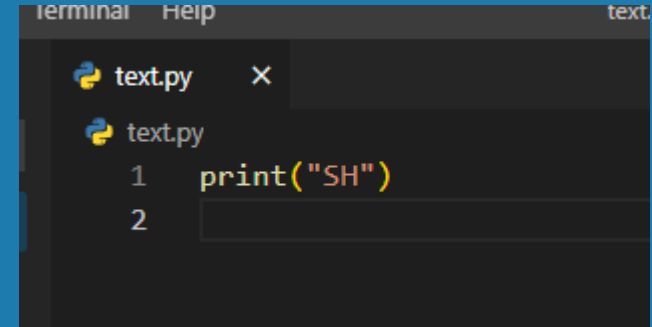
# Vs코드 Pull

Vs코드에서

`git pull <원격저장소명> <브랜치명>`

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/friendgit (main)
$ git pull test main
```

파일 확인

A screenshot of a VS Code editor window. The title bar shows 'terminal Help text'. The editor has a tab for 'text.py' with a Python icon and a close button. The file content is displayed as follows:

```
1 print("SH")
2
```



# Part 4,

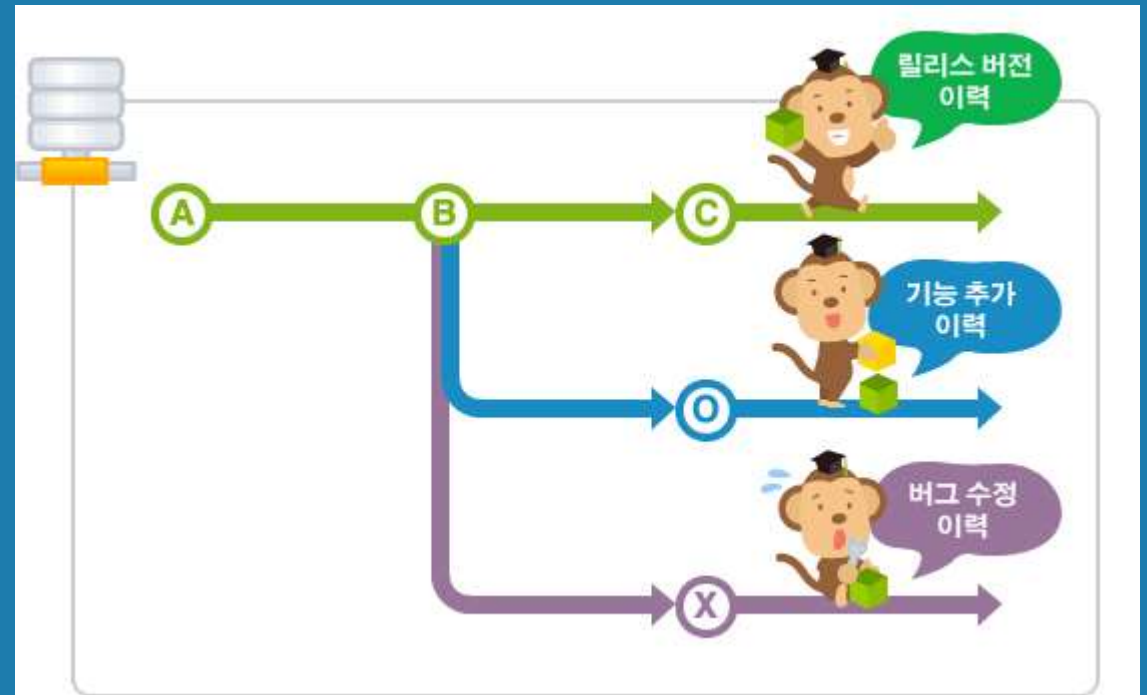
---

## Branch 사용해보기

# Branch

## 브랜치(branch)란?

브랜치란 독립적으로 어떤 작업을 진행하기 위한 개념입니다.  
필요에 의해 만들어지는 각각의 브랜치는 다른 브랜치의 영향을 받지 않기 때문에,  
여러 작업을 동시에 진행할 수 있습니다.



# Branch

git branch <브랜치명>

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/friendgit  
$ git branch JSH
```

Branch 이동 : git checkout <브랜치명>

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/friendgit (JSH)  
$ git checkout JSH
```

git add “파일명”

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/friendgit (JSH)  
$ git add text.py
```

git commit -m “commit내용”

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/friendgit (JSH)  
$ git commit -m "새로운브랜치"  
On branch JSH
```

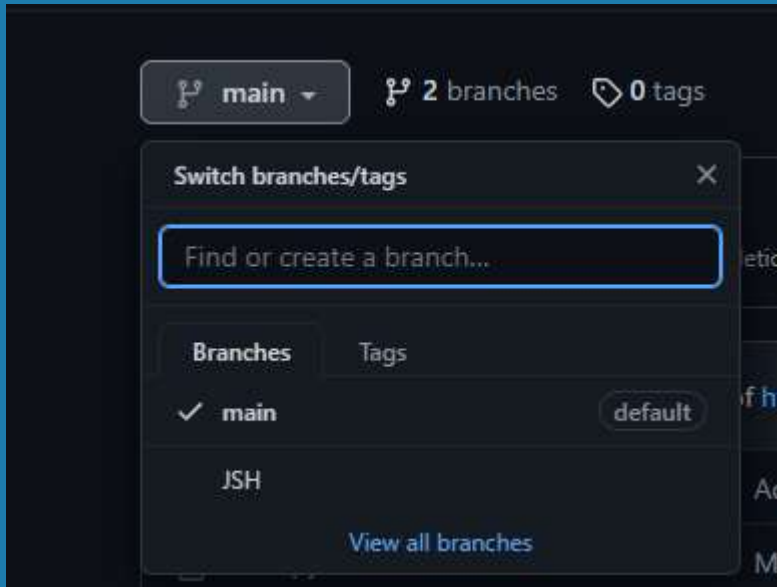
git push <원격저장소명> <브랜치명>

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/friendgit (JSH)  
$ git push test JSH  
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0  
remote:  
remote: Create a pull request for 'JSH' on GitHub by visiting:  
remote:      https://github.com/jeonsanghoon711/test/pull/new/JSH  
remote:  
To https://github.com/jeonsanghoon711/test.git  
* [new branch]      JSH -> JSH
```

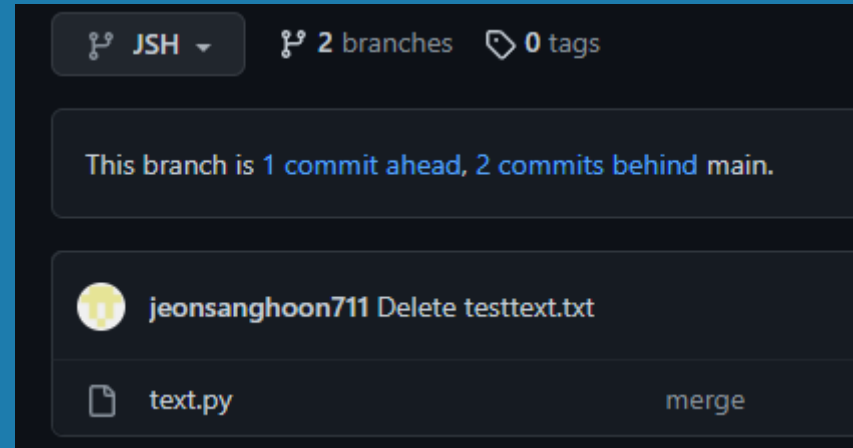
# Branch

Git hub에 새로운 Branch 확인

Branch 이동

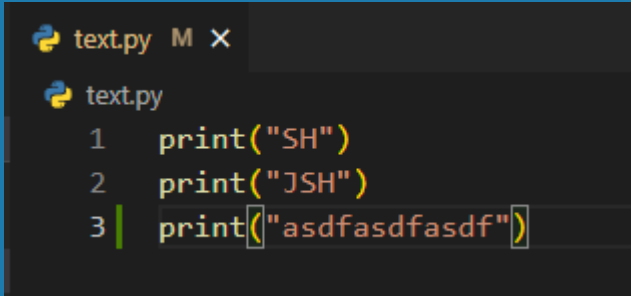


Branch 확인



# Branch

## 파일 수정



```
text.py M X
text.py
1 print("SH")
2 print("JSH")
3 print("asdfasdfasdf")
```

git add “파일명”

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/friendgit (JSH)
$ git add text.py
```

git commit -m “commit내용”

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/friendgit (JSH)
$ git commit -m "새로운브랜치"
[master 75H]
```

git push <원격저장소명> <브랜치명>

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/friendgit (JSH)
$ git push test JSH
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'JSH' on GitHub by visiting:
remote:   https://github.com/jeonsanghoon711/test/pull/new/JSH
remote:
To https://github.com/jeonsanghoon711/test.git
 * [new branch]      JSH -> JSH
```

# Merge

## 파일 확인



git checkout <브랜치명>

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/friendgit (JSH)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'test/main'.
```

git merge <브랜치명>

```
User@DESKTOP-M960RTM MINGW64 ~/Desktop/friendgit (main)
$ git merge JSH
Auto-merging text.py
CONFLICT (content): Merge conflict in text.py
Automatic merge failed; fix conflicts and then commit the result.
```

# Merge

The screenshot shows the Visual Studio Code Merge Editor interface. The title bar indicates the file is 'text.py' and the editor is 'InfernoGit - Visual Studio Code'. The editor window displays a conflict between the 'HEAD (Current Change)' and 'JSH (Incoming Change)' branches. The conflict is highlighted with a green background for the 'HEAD' change and a blue background for the 'JSH' change. The code in the editor is as follows:

```
1 print("SH")
2 print("JSH")
3 <<<<<< HEAD (Current Change)
4 print("JSH")
5 print("JSH")
6
7 print("JSH")
8 =====
9 print("asdfasdfasdf")
10 >>>>>> JSH (Incoming Change)
11
```

At the bottom right of the editor, there is a button labeled 'Resolve in Merge Editor' which is highlighted with a red rectangle.

# Merge

The image shows a code editor interface with three main sections: Incoming, Current, and Result.

**Incoming (4b0551a - test/JSH, JSH):**

```
1 print("SH")
2 print("JSH")
3 print("asdfasdfasdf")
```

A red box highlights the conflict resolution options for line 1: "Accept Incoming | Accept Combination | Ignore".

**Current (ac8aac2 - main, test/main):**

```
1 print("SH")
2 print("JSH")
3 print("JSH")
4 print("JSH")
5
6 print("JSH")
```

A red box highlights the conflict resolution options for line 1: "Accept Current | Accept Combination | Ignore".

**Result (text.py):**

```
1 print("SH")
2 print("JSH")
3 print("JSH")
4 print("JSH")
5
6 print("JSH")
7 print("asdfasdfasdf")
```

A yellow dot is on line 2. A red box highlights the conflict resolution options for line 2: "No Changes Accepted", "Manual Resolution", and "Reset to base".

In the bottom right corner, a blue button labeled "Complete Merge" is highlighted with a red box.




# Part 4,

---

## 깃 소스트리 사용해보기

# 깃소스트리

<https://www.sourcetreeapp.com/>

 **Sourcetree**

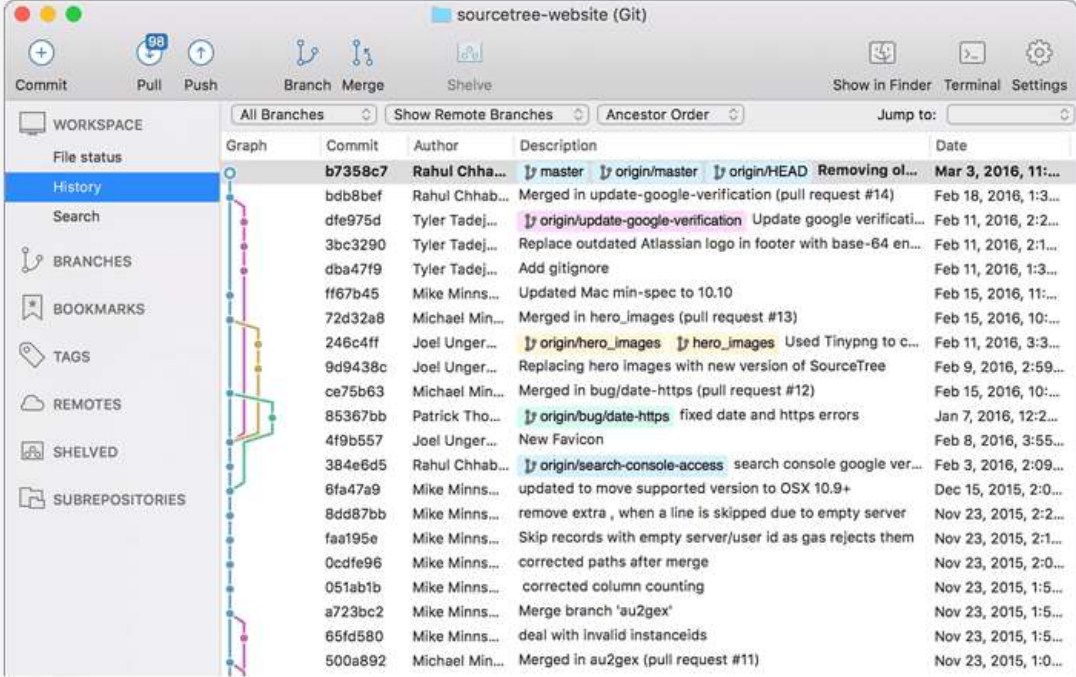
[Download free](#)

## Simplicity and power in a beautiful Git GUI

[Download for Windows](#)

Also available for Mac OS X


Latest release notes: [Mac OS X](#) & [Windows](#)



Commit	Author	Description	Date
b7358c7	Rahul Chha...	master origin/master origin/HEAD Removing ol...	Mar 3, 2016, 11:...
bdb8bef	Rahul Chhab...	Merged in update-google-verification (pull request #14)	Feb 18, 2016, 1:3...
dfe975d	Tyler Tadej...	origin/update-google-verification Update google verificati...	Feb 11, 2016, 2:2...
3bc3290	Tyler Tadej...	Replace outdated Atlassian logo in footer with base-64 en...	Feb 11, 2016, 2:1...
dba47f9	Tyler Tadej...	Add gitignore	Feb 11, 2016, 1:3...
ff67b45	Mike Minns...	Updated Mac min-spec to 10.10	Feb 15, 2016, 11:...
72d32a8	Michael Min...	Merged in hero_images (pull request #13)	Feb 15, 2016, 10:...
246c4ff	Joel Unger...	origin/hero_images hero_images Used Tinypng to c...	Feb 11, 2016, 3:3...
9d9438c	Joel Unger...	Replacing hero Images with new version of SourceTree	Feb 9, 2016, 2:59...
ce75b63	Michael Min...	Merged in bug/date-https (pull request #12)	Feb 15, 2016, 10:...
85367bb	Patrick Tho...	origin/bug/date-https fixed date and https errors	Jan 7, 2016, 12:2...
4f9b557	Joel Unger...	New Favicon	Feb 8, 2016, 3:55...
384e6d5	Rahul Chhab...	origin/search-console-access search console google ver...	Feb 3, 2016, 2:09...
8fa47a9	Mike Minns...	updated to move supported version to OSX 10.9+	Dec 15, 2015, 2:0...
8dd87bb	Mike Minns...	remove extra , when a line is skipped due to empty server	Nov 23, 2015, 2:2...
faa195e	Mike Minns...	Skip records with empty server/user id as gas rejects them	Nov 23, 2015, 2:1...
0cdf96	Mike Minns...	corrected paths after merge	Nov 23, 2015, 2:0...
051ab1b	Mike Minns...	corrected column counting	Nov 23, 2015, 1:5...
a723bc2	Mike Minns...	Merge branch 'au2gex'	Nov 23, 2015, 1:5...
65fd580	Mike Minns...	deal with invalid instanceids	Nov 23, 2015, 1:5...
500a892	Michael Min...	Merged in au2gex (pull request #11)	Nov 23, 2015, 1:0...

# 깃소스트리

<https://www.sourcetreeapp.com/>

 **Sourcetree**

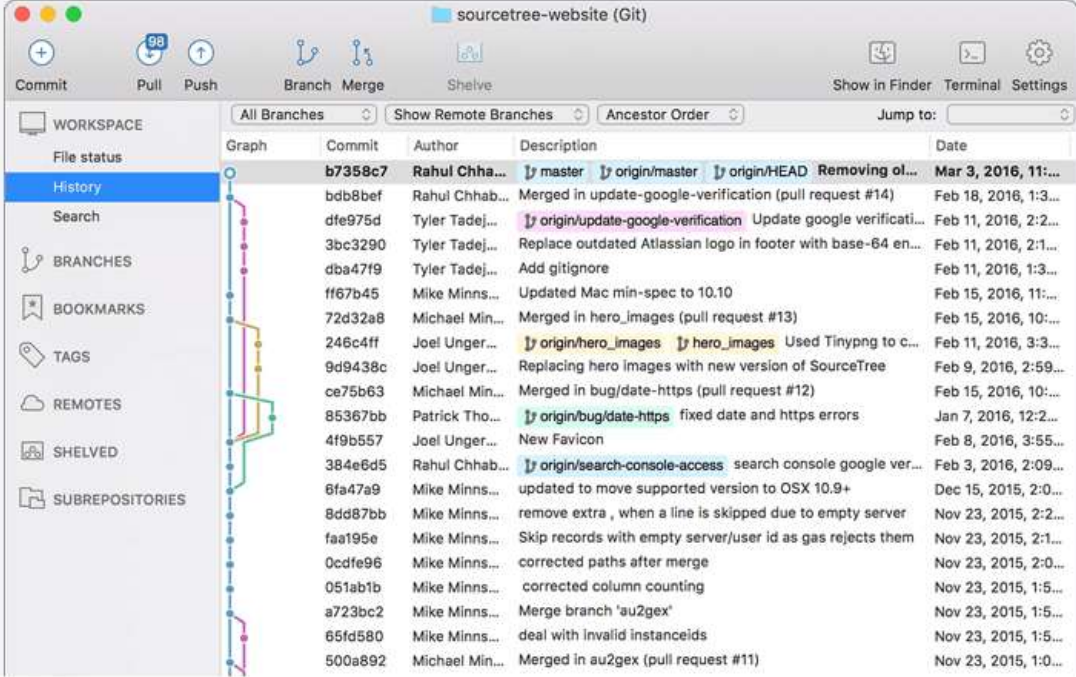
[Download free](#)

## Simplicity and power in a beautiful Git GUI

[Download for Windows](#)

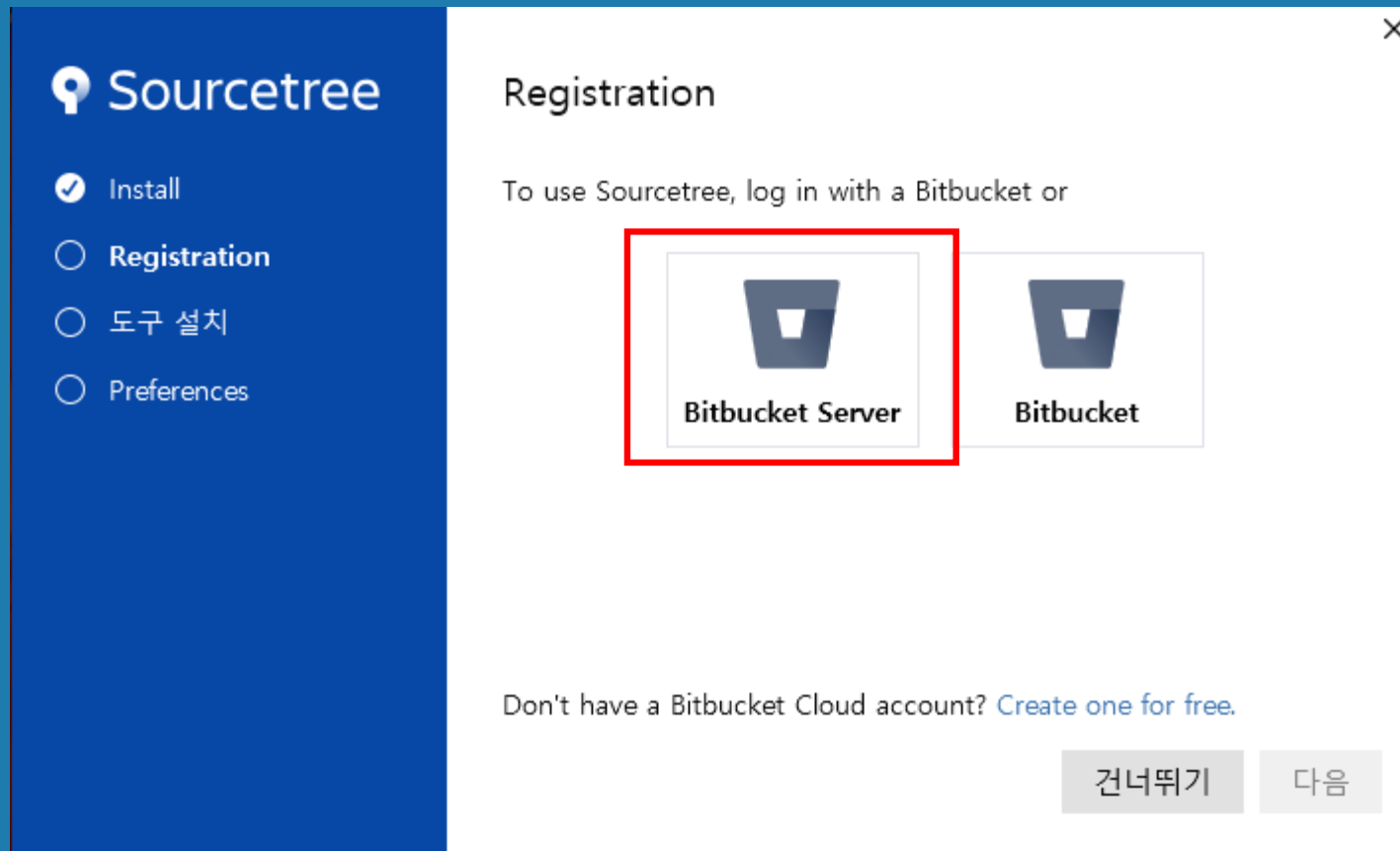
Also available for Mac OS X


Latest release notes: [Mac OS X](#) & [Windows](#)



Commit	Author	Description	Date
b7358c7	Rahul Chha...	master origin/master origin/HEAD Removing ol...	Mar 3, 2016, 11:...
bdb8bef	Rahul Chhab...	Merged in update-google-verification (pull request #14)	Feb 18, 2016, 1:3...
dfe975d	Tyler Tadej...	origin/update-google-verification Update google verificati...	Feb 11, 2016, 2:2...
3bc3290	Tyler Tadej...	Replace outdated Atlassian logo in footer with base-64 en...	Feb 11, 2016, 2:1...
dba47f9	Tyler Tadej...	Add gitignore	Feb 11, 2016, 1:3...
ff67b45	Mike Minns...	Updated Mac min-spec to 10.10	Feb 15, 2016, 11:...
72d32a8	Michael Min...	Merged in hero_images (pull request #13)	Feb 15, 2016, 10:...
246c4ff	Joel Unger...	origin/hero_images hero_images Used Tinypng to c...	Feb 11, 2016, 3:3...
9d9438c	Joel Unger...	Replacing hero Images with new version of SourceTree	Feb 9, 2016, 2:59...
ce75b63	Michael Min...	Merged in bug/date-https (pull request #12)	Feb 15, 2016, 10:...
85367bb	Patrick Tho...	origin/bug/date-https fixed date and https errors	Jan 7, 2016, 12:2...
4f9b557	Joel Unger...	New Favicon	Feb 8, 2016, 3:55...
384e6d5	Rahul Chhab...	origin/search-console-access search console google ver...	Feb 3, 2016, 2:09...
8fa47a9	Mike Minns...	updated to move supported version to OSX 10.9+	Dec 15, 2015, 2:0...
8dd87bb	Mike Minns...	remove extra , when a line is skipped due to empty server	Nov 23, 2015, 2:2...
faa195e	Mike Minns...	Skip records with empty server/user id as gas rejects them	Nov 23, 2015, 2:1...
0cdf96	Mike Minns...	corrected paths after merge	Nov 23, 2015, 2:0...
051ab1b	Mike Minns...	corrected column counting	Nov 23, 2015, 1:5...
a723bc2	Mike Minns...	Merge branch 'au2gex'	Nov 23, 2015, 1:5...
65fd580	Mike Minns...	deal with invalid instanceids	Nov 23, 2015, 1:5...
500a892	Michael Min...	Merged in au2gex (pull request #11)	Nov 23, 2015, 1:0...

# 깃소스트리



 **Sourcetree**

☒ Install  
☐ **Registration**  
☐ 도구 설치  
☐ Preferences

Login to Bitbucket Server

You'll need to enter a bit more details before finishing

루트 URL:

사용자명

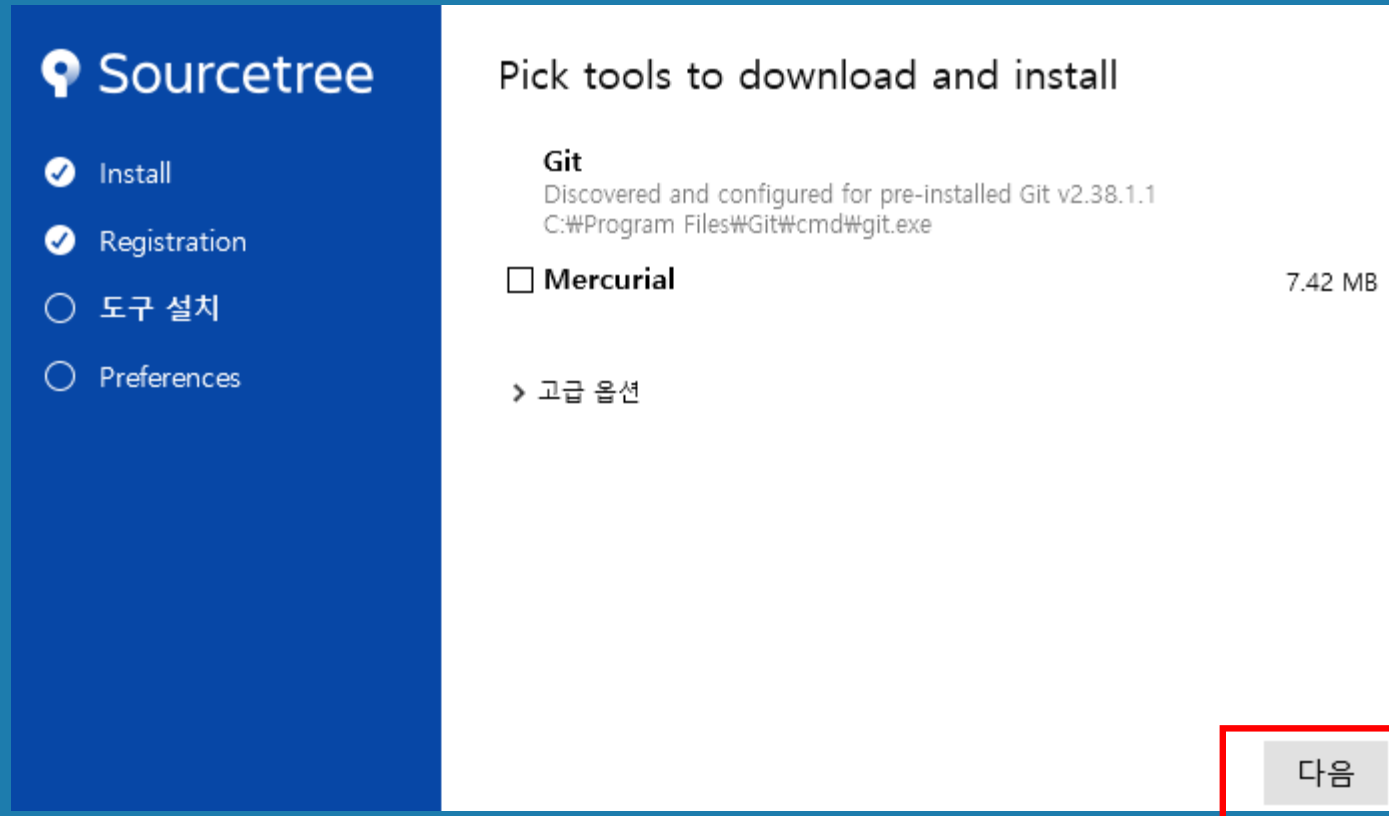
뒤로

로그인

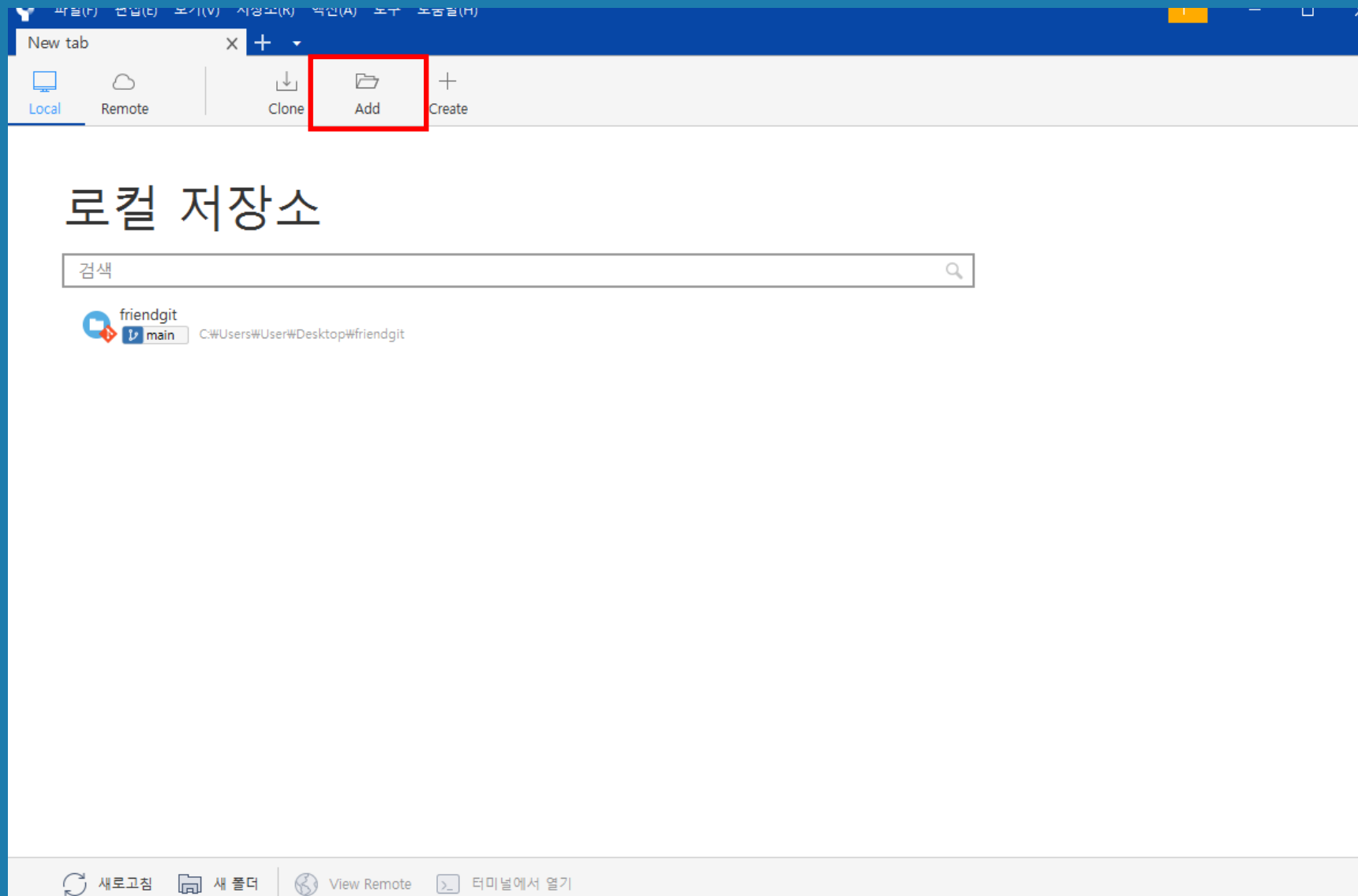
건너뛰기

다음

# 깃소스트리



# 깃소스트리



# 깃소스트리

The screenshot shows the 'friendgit' application interface. The top toolbar includes icons for commit, pull, push, patch, branch, merge, stash, discard, and tag. The left sidebar shows the 'WORKSPACE' with options for '파일 상태' (File Status), 'History', and 'Search'. Below this, there are sections for '브랜치' (Branches) showing 'JSH' and 'main', '태그' (Tags), '원격' (Remote), and '스태시' (Stash).

The main area displays a commit history table. The current commit is 'merage' (likely a typo for merge) with hash '088f507'. The table lists the following commits:

날짜	작성자	커밋
20 1 2023 4:45	JeonSangHoon71	088f507
20 1 2023 4:34	JeonSangHoon71	4b0551a
20 1 2023 4:33	JeonSangHoon71	c2bf060
20 1 2023 4:31	jeonsanghoon711	03484e7
20 1 2023 4:14	JeonSangHoon71	ac8aac2
20 1 2023 3:57	JeonSangHoon71	52facfd
20 1 2023 3:56	jeonsanghoon711	faccbe7
20 1 2023 3:50	jeonsanghoon711	f321e62
20 1 2023 3:39	JeonSangHoon71	8c114e1
20 1 2023 3:14	JeonSangHoon71	e2b97a4
20 1 2023 3:02	jeonsanghoon711	4d736ec
20 1 2023 2:33	JeonSangHoon71	770278f

The bottom section shows a file diff for 'testtext.txt'. The diff indicates a change in the file's content, specifically mentioning '만들하세요' (Create this file) and 'No newline at end of file'.



# 출처

---

[https://backlog.com/git-tutorial/kr/stepup/stepup1\\_1.html](https://backlog.com/git-tutorial/kr/stepup/stepup1_1.html)

감사합니다