LG전자 BS팀

『1일차』: 오후

◈ 훈련과정명 : [BS] Git 형상관리에 대한 이해 및 실습

◈ 훈 련 기 간 : 2023.06.07~2023.06.09

HS글로벌

목차

『1과목』 Git, GitHub 기본 사용

5-6교시 :
커밋 내역 수정

1. 인프라 설계 준비

학습목표

- 로컬저장소를 소스트리에 불러올 수 있다.
- 소스트리로 커밋을 만들고 푸시를 할 수 있다.

눈높이 체크

- SourceTree를 알고 계신가요?

1. Pre-Requisites

1. new repository 생성

- 앞선 만든, git_study_1st_project 이용

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV
$ git config --global user.name
looker2zip

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV
$ git config --global user.email
looker2zip@gmail.com

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV
$ cd gitworkspaces/

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces
$ pwd
/c/DEV/gitworkspaces

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces
$ ll -a
total 4
drwxr-xr-x 1 apro621 197121 0 5월 8 09:14 ./
drwxr-xr-x 1 apro621 197121 0 5월 8 09:14 ../

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces
$ git clone https://github.com/looker2zip/git_study_1st_project.git
Cloning into 'git_study_1st_project'...
warning: You appear to have cloned an empty repository.

1. Pre-Requisites

1. new repository 생성

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces
$ ll -a
total 4
drwxr-xr-x 1 apro621 197121 0 5월 8 09:14 ./
drwxr-xr-x 1 apro621 197121 0 5월 8 09:14 ../
drwxr-xr-x 1 apro621 197121 0 5월 8 09:14 git_study_1st_project/

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces
$ cd git_study_1st_project/

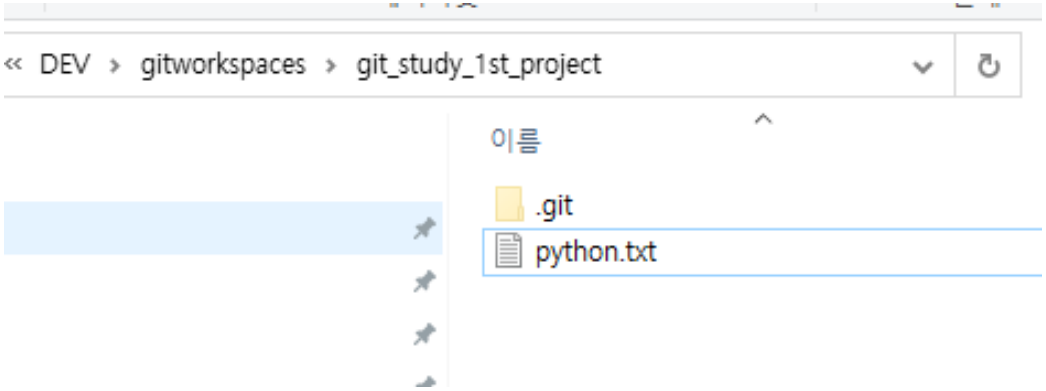
apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ ll -a
total 5
drwxr-xr-x 1 apro621 197121 0 5월 8 09:20 ./
drwxr-xr-x 1 apro621 197121 0 5월 8 09:14 ../
drwxr-xr-x 1 apro621 197121 0 5월 8 09:14 .git/

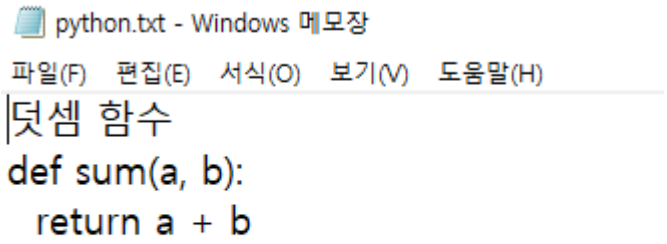
apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$

1. Pre-Requisites

2. 파일 생성

- 파이썬 함수 작성 순서.txt 파일 생성





1. Pre-Requisites

3. Push

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ ll -a
total 5
drwxr-xr-x 1 apro621 197121 0 5월 8 09:30 ./
drwxr-xr-x 1 apro621 197121 0 5월 8 09:14 ../
drwxr-xr-x 1 apro621 197121 0 5월 8 09:30 .git/
-rw-r--r-- 1 apro621 197121 45 5월 8 09:22 python.txt

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git status
On branch main

No commits yet

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

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

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git add python.txt

3. Push

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git status
On branch main

No commits yet

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

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git commit -m "Add python.txt"
[main (root-commit) 2c6c8e3] Add python.txt
 1 file changed, 3 insertions(+)
 create mode 100644 python.txt

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git status
On branch main
Your branch is based on 'origin/main', but the upstream is gone.
  (use "git branch --unset-upstream" to fixup)

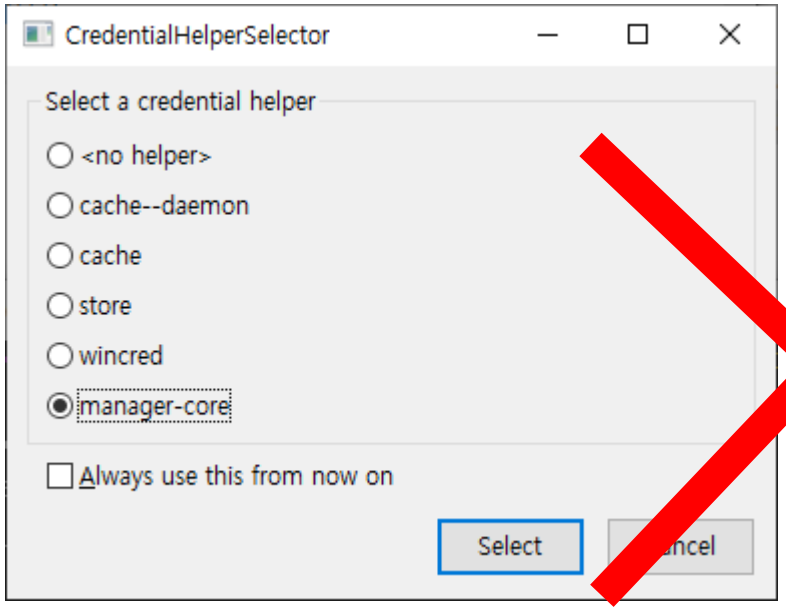
nothing to commit, working tree clean

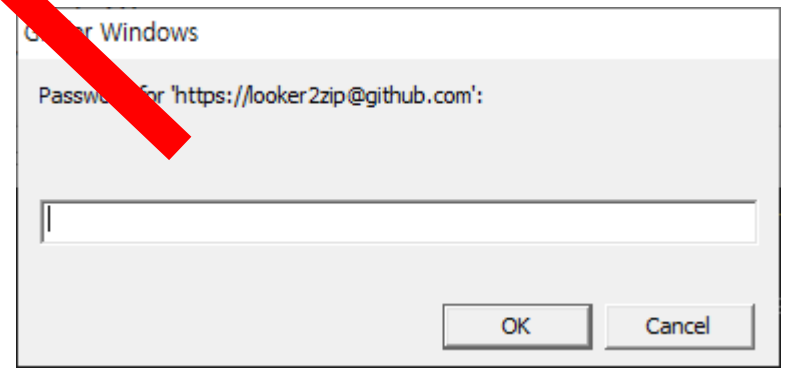
apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)

1. Pre-Requisites

3. Push

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git push



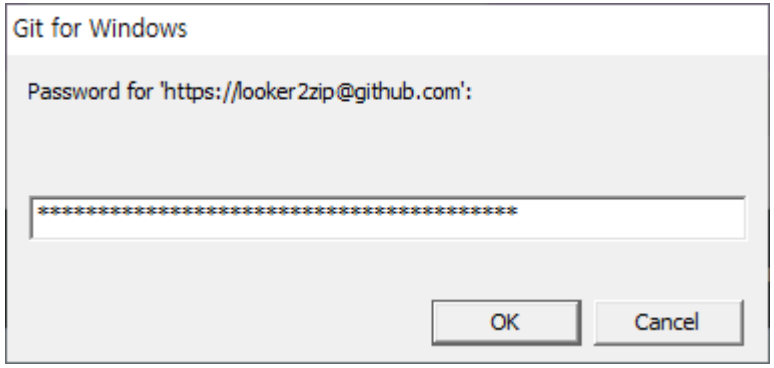


1. Pre-Requisites

3. Push

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git push

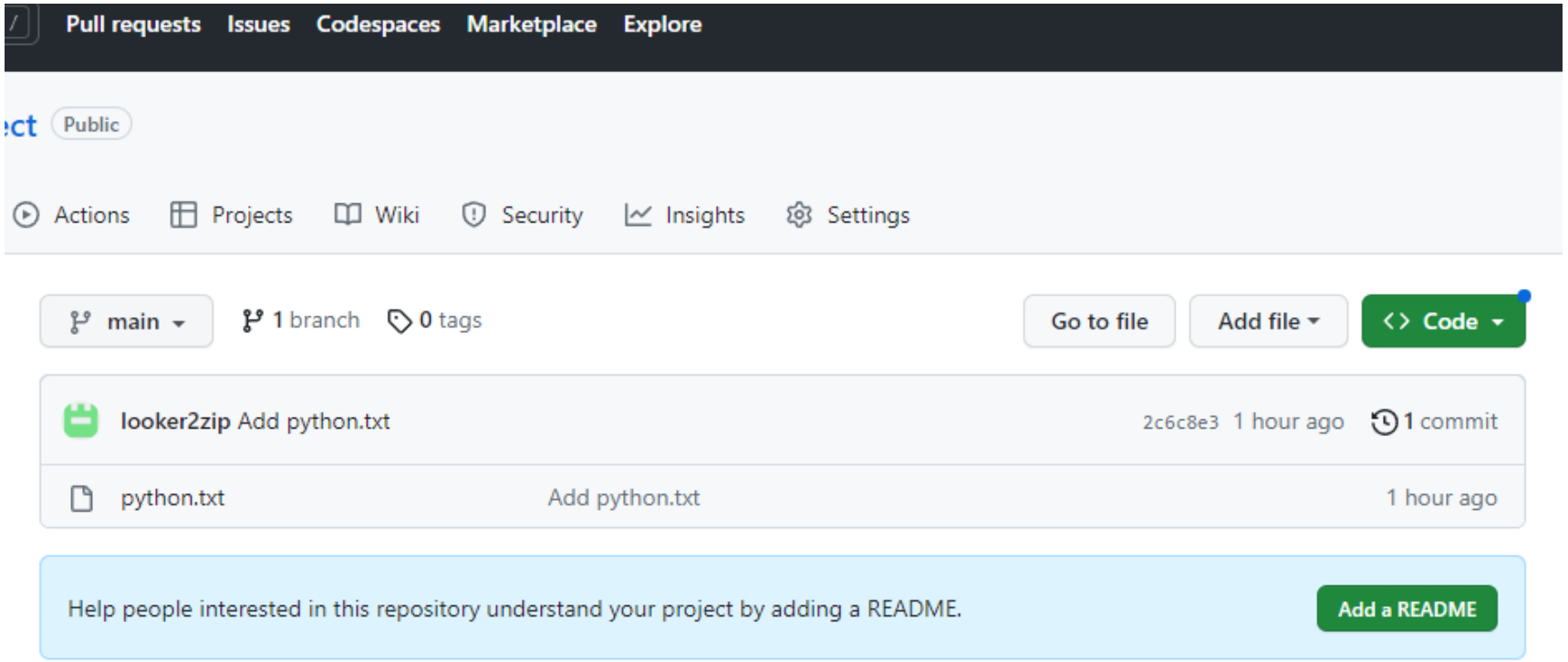
- **GitHub에 push 등의 작업을 수행할 때,PW를 입력하라고 뜨면, 방금 생성한 Personal Access Token을 입력**



1. Pre-Requisites

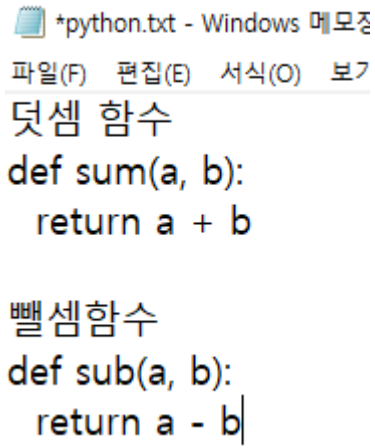
3. Push

- **Push 완료**



1. Pre-Requisites

4. 파일 갱신



apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

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: python.txt

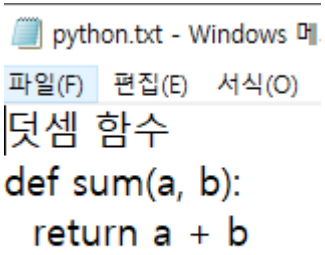
no changes added to commit (use "git add" and/or "git commit -a")

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$

5. 되돌리기

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git checkout -- python.txt

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$



$ git checkout -- python.txt

1. Pre-Requisites

6.파일 재 생성 후 Push

*python.txt - Windows

파일(F) 편집(E) 서식(O)

```
덧셈 함수
def sum(a, b):
  return a + b

뺄셈 함수
def sub(a, b):
  return a - b
```

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
~~$ git checkout -- python.txt~~

$ git add python.txt

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git commit -m "Add python.txt [Sub]"
[main 36e9c10] Add python.txt [Sub]
 1 file changed, 5 insertions(+), 1 deletion(-)

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$

커밋 명령을 변경하고자 할 경우, $ git commit --amend 을 이용할 수 있다.

6.파일 재 생성 후 Push

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

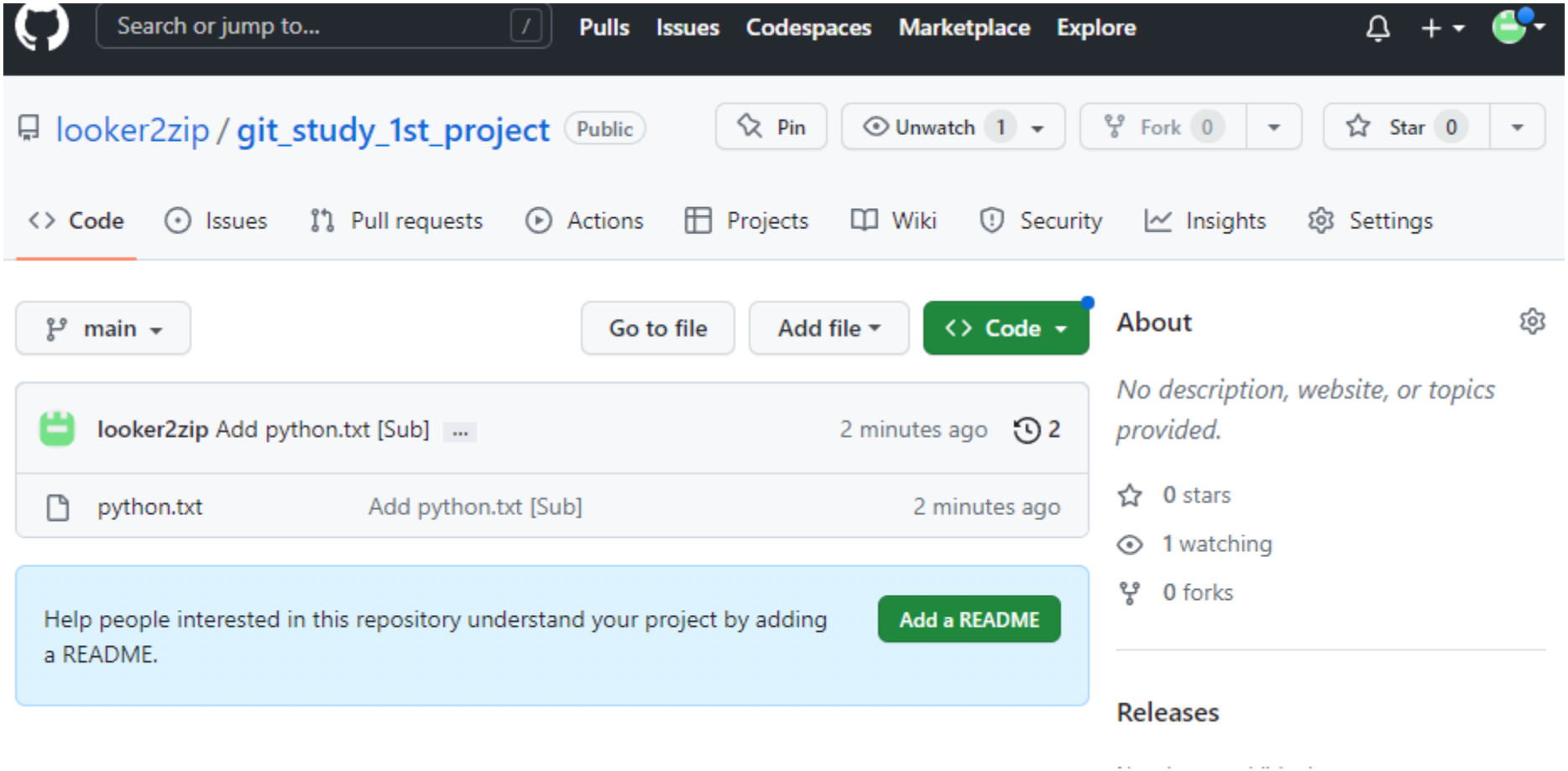
nothing to commit, working tree clean

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git push 전부 로컬저장소에 push되는 중
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 313 bytes | 313.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/looker2zip/git_study_1st_project.git
   2c6c8e3..8dc5a9c main -> main

apro621@DESKT

1. Pre-Requisites

7.파일 재 생성 후 Push 완료



2. 커밋(Commit) 내역 수정

1. pull

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git pull
Already up to date.

- **파일 변경**

python.txt - Windows 메모장

파일(F) 편집(E) 서식(O) 보기(V

```
덧셈 함수
def sum(a, b):
  return a + b

뺄셈 함수
def sub(a, b):
  return a - b

곱셈 함수
def mul(a, b):
  return a * b
```

2. 커밋(Commit) 내역 수정

2. push

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git add .

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git commit -m "Add python.txt [Mul]"
[main 09cc239] Add python.txt [Mul]
 1 file changed, 5 insertions(+), 1 deletion(-)

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 324 bytes | 324.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/looker2zip/git_study_1st_project.git
   8dc5a9c..09cc239 main -> main

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$

2. 커밋(Commit) 내역 수정

3. Git log

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git log
commit 09cc2394273fb2283de0eb71b6cfe18b91025ca4 (HEAD -> main, origin/main)
Author: looker2zip <looker2zip@gmail.com>
Date: Mon May 8 11:05:38 2023 +0900

    Add python.txt [Mul]

commit 8dc5a9c0672bedb12fc8989928f61c9e69b02ccf
Author: looker2zip <looker2zip@gmail.com>
Date: Mon May 8 10:51:54 2023 +0900

    Add python.txt [Sub]

commit 2c6c8e3d686e89c3f7b79ed69e6ba7b836ad37f3
Author: looker2zip <looker2zip@gmail.com>
Date: Mon May 8 09:32:20 2023 +0900

    Add python.txt

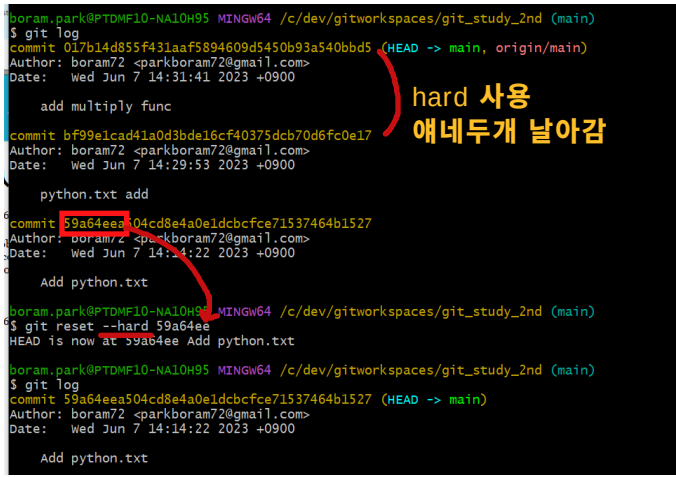
apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$

로그 창 빠져 나올 때 Q 입력

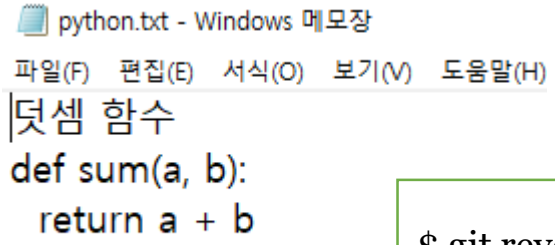
3. 커밋 전으로 되돌아가기

4. 되돌아가기

```
apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git reset --hard 2c6c8e3d686e89c3f7b79ed69e6ba7b836ad37f3
HEAD is now at 2c6c8e3 Add python.txt

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git log
commit 2c6c8e3d686e89c3f7b79ed69e6ba7b836ad37f3 (HEAD -> main)
Author: looker2zip <looker2zip@gmail.com>
Date:   Mon May 8 09:32:20 2023 +0900

    Add python.txt

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$
```



--hard 옵션은 선택한 것 이후에 것은 모두 삭제함




```
$ git revert e1d2c50a478510f70fed6a5857aed9b43d7ffc46
$ git revert --no-commit e1d2c50a478510f70fed6a5857aed9b43d7ffc46
```

3. 커밋 전으로 되돌아가기

5. Push 할 경우

$ git push

To https://github.com/looker2zip/git_study_1st_project.git
 ! [rejected] main -> main (non-fast-forward)
error: failed to push some refs to 'https://github.com/looker2zip/git_study_1st_project.git'
hint: Updates were rejected because the tip of your current branch is behind
hint: its remote counterpart. Integrate the remote changes (e.g.
hint: 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

apr0621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$

에러 남

```
boram.park@PTDMF10-NA10H95 MINGW64 /c/dev/gitworkspaces/git_study_2nd (main)
$ git push
To https://github.com/boram72/git_study_2nd.git
 ! [rejected]        main -> main (non-fast-forward)
error: failed to push some refs to 'https://github.com/boram72/git_study_2nd.git'
hint: Updates were rejected because the tip of your current branch is behind
hint: its remote counterpart. Integrate the remote changes (e.g.
hint: 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
```

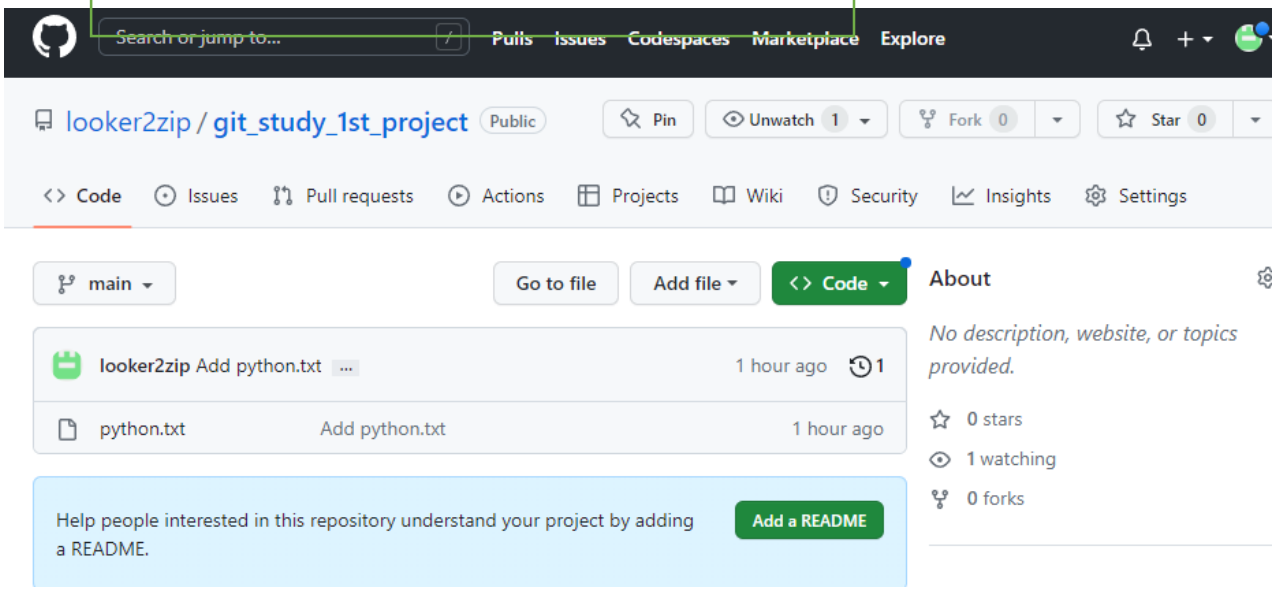
3. 커밋 전으로 되돌아가기

5. Push 할 경우

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git push -f
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/looker2zip/git_study_1st_project.git
 + 09cc239...2c6c8e3 main -> main (forced update)

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$

강제로 Push



4. 커밋 메시지 변경

1. 기존 커밋 메시지

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git log
commit 2c6c8e3d686e89c3f7b79ed69e6ba7b836ad37f3 (HEAD -> main, origin/main)
Author: looker2zip <looker2zip@gmail.com>
Date: Mon May 8 09:32:20 2023 +0900

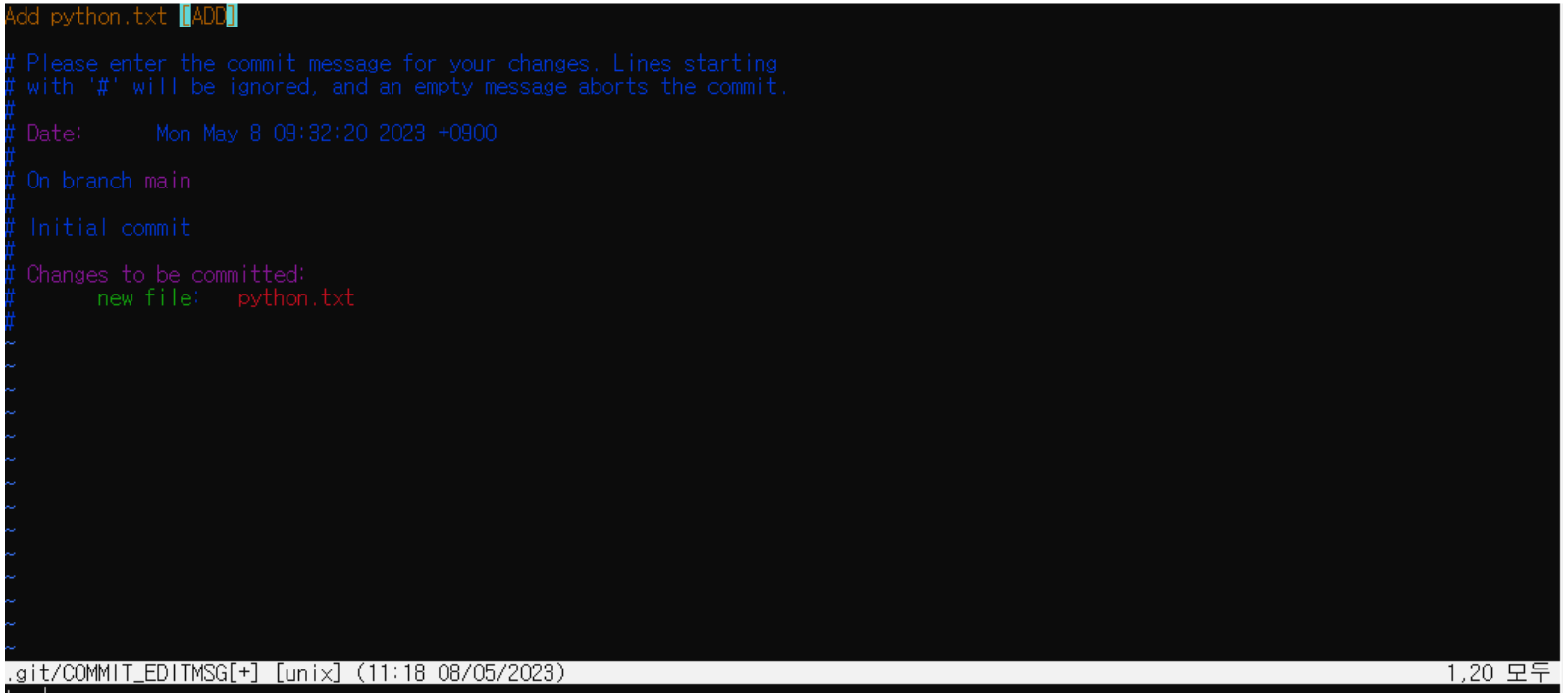
    Add python.txt

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$

4. 커밋 메시지 변경

2. 새 커밋 메시지

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git commit --amend



4. 커밋 메시지 변경

2. 새 커밋 메시지

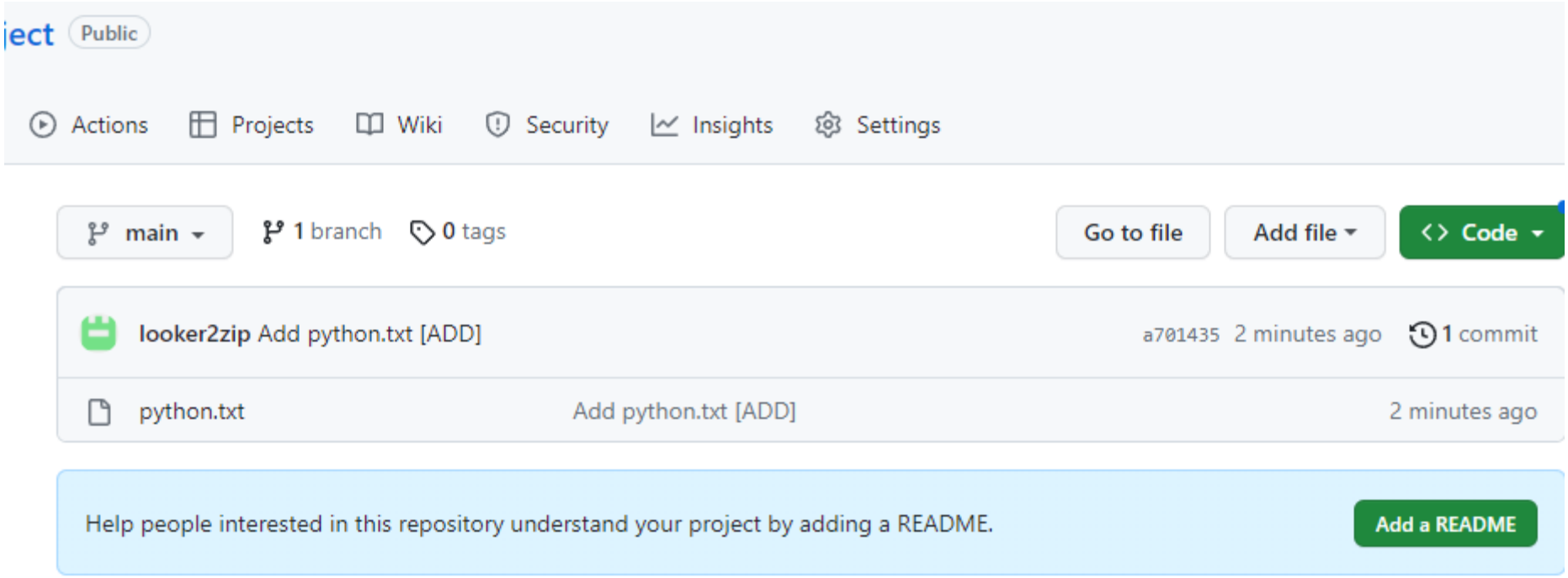
apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git log
commit a701435eb269f1609e7d0de393fd92943f4812f0 (HEAD -> main)
Author: looker2zip <looker2zip@gmail.com>
Date: Mon May 8 09:32:20 2023 +0900

    Add python.txt [ADD]
apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$ git push -f
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 272 bytes | 272.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/looker2zip/git_study_1st_project.git
 + 2c6c8e3...a701435 main -> main (forced update)

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_1st_project (main)
$

4. 커밋 메시지 변경

3. Push 완료



『2과목』 GUI 환경에서 Git 사용

7-8교시 : GUI 환경에서 버전 관리 시작하기

1. 인프라 설계 준비

학습목표

- GUI 버전 관리 환경 구축을 할 수 있다.
- SourceTree 로 커밋 만들고 푸시를 할 수 있다.

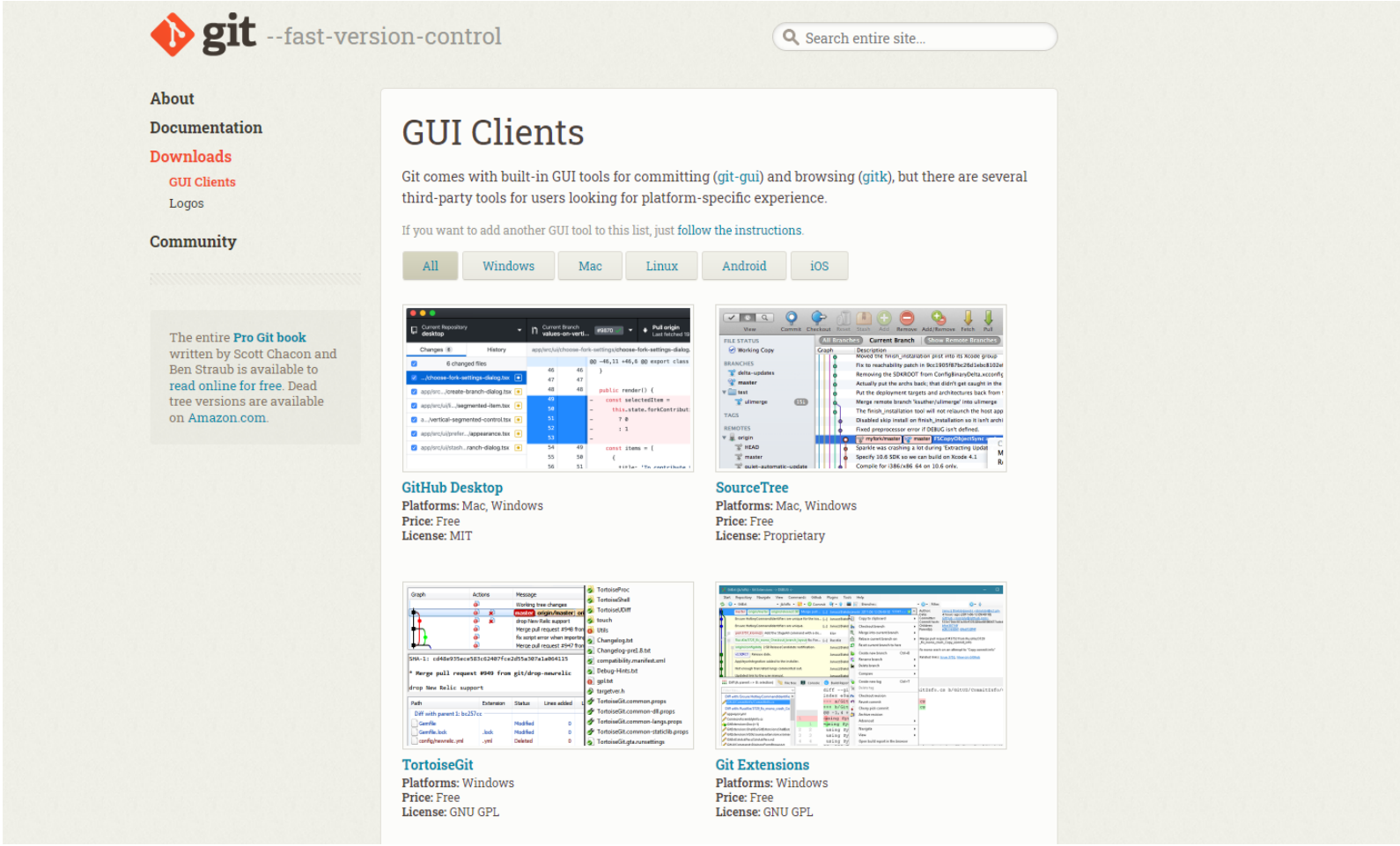
눈높이 체크

- SourceTree를 알고 계신가요?

1. 내 컴퓨터에 소스트리 설치하기

GUI Clients

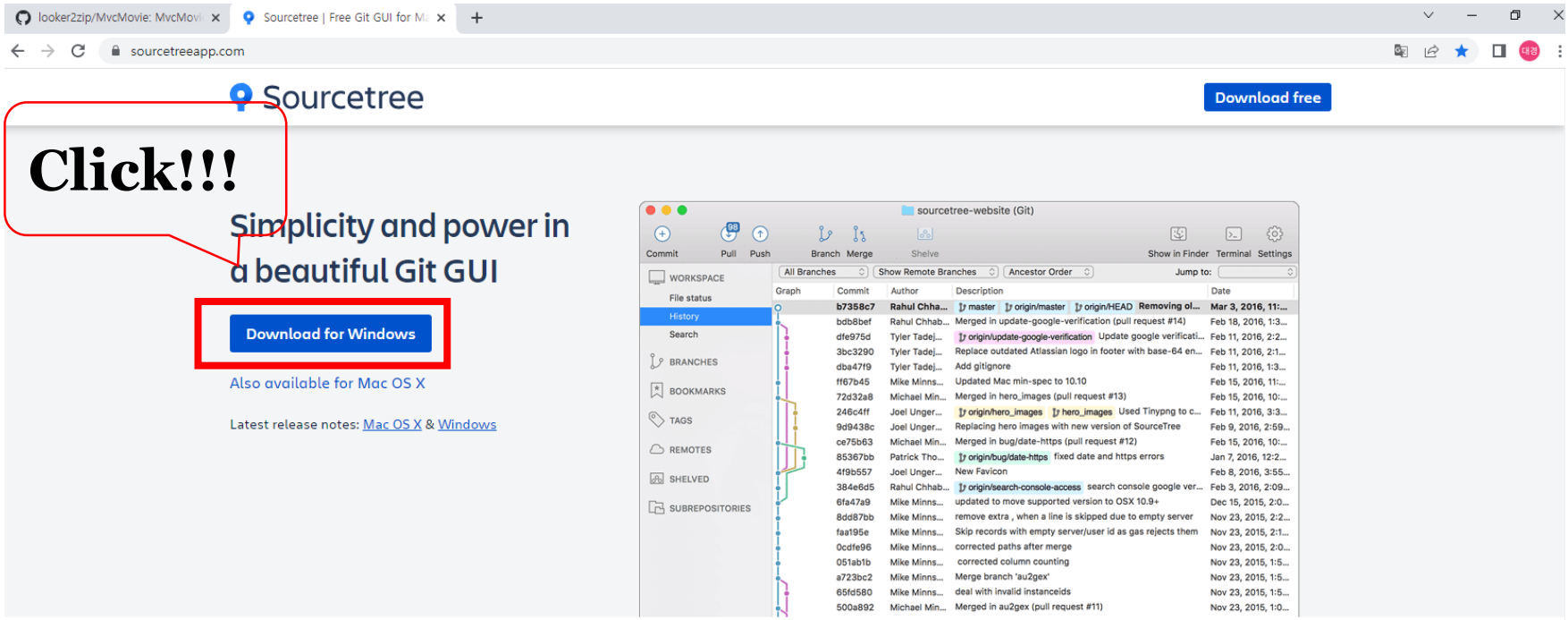
- **https://git-scm.com/downloads/guis**



1. 내 컴퓨터에 소스트리 설치하기

소스트리 다운로드 하기

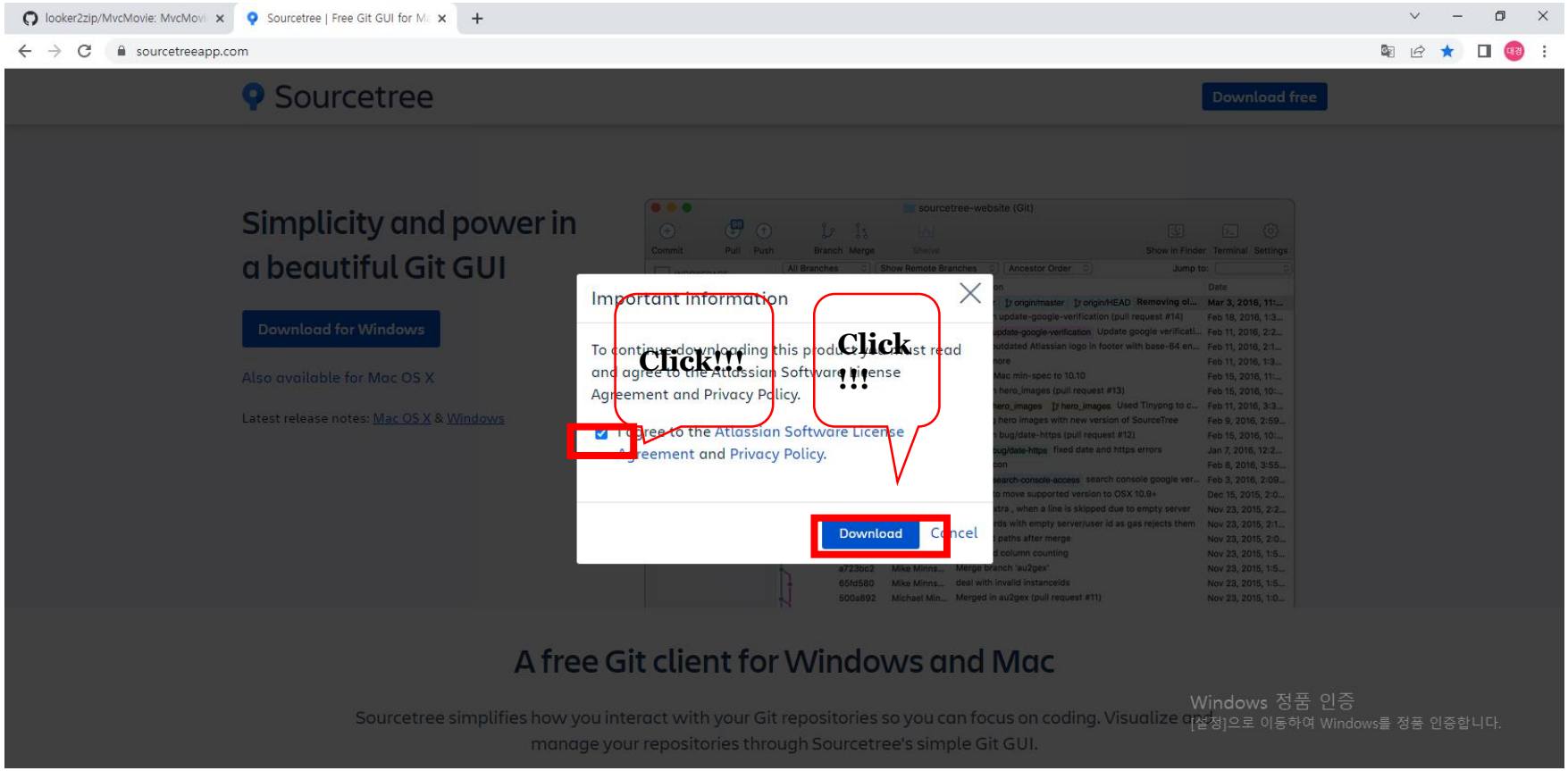
- **https://www.sourcetreeapp.com/**



1. 내 컴퓨터에 소스트리 설치하기

소스트리 다운로드 하기

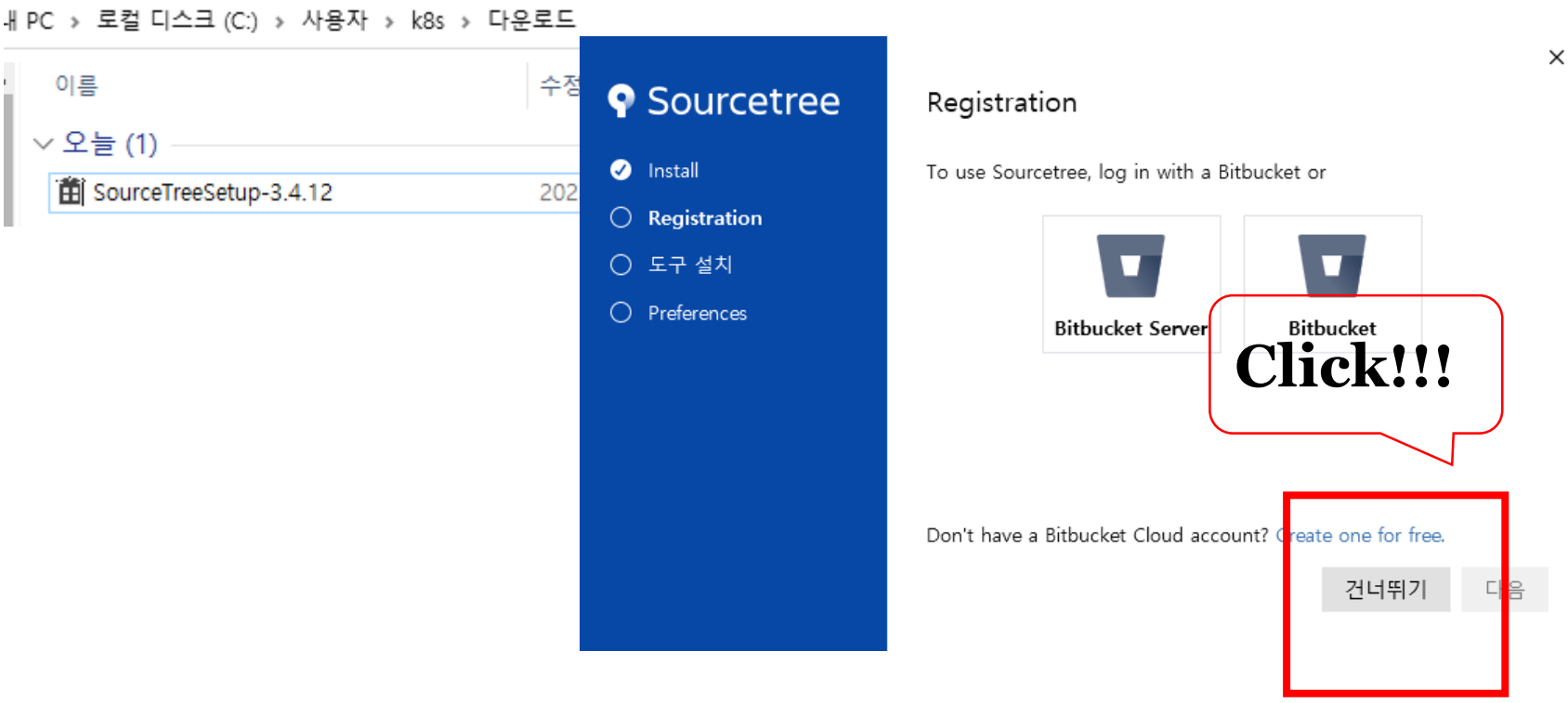
- **Atlassian Software License Agreement and Privacy Policy 에 체크하고 다운로드 버튼 클릭**



1. 내 컴퓨터에 소스트리 설치하기

소스트리 설치하기

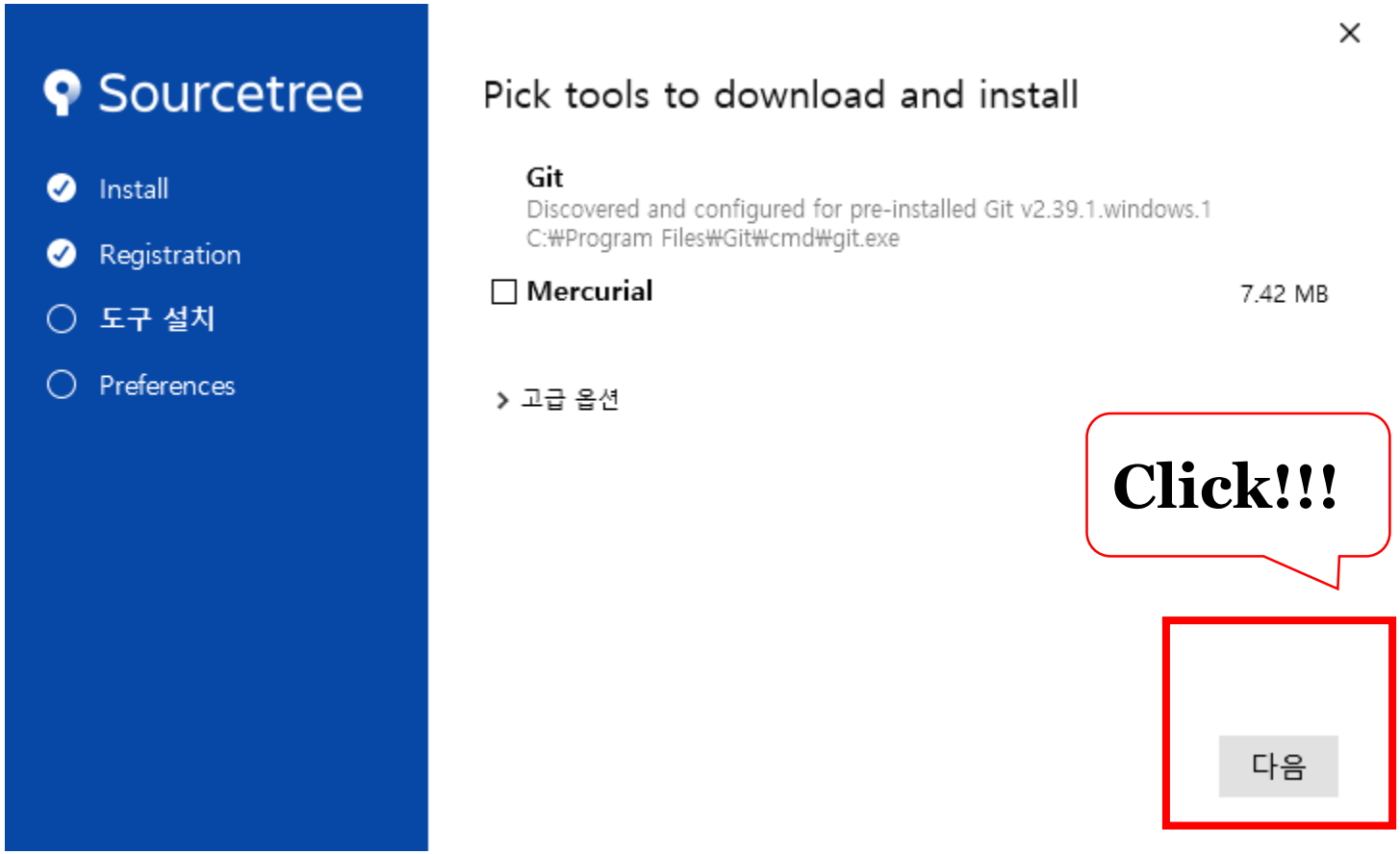
- **SourceTreeSetup-3.4.12.exe 더블 클릭**
- 건너뛰기 클릭



1. 내 컴퓨터에 소스트리 설치하기

소스트리 설치하기

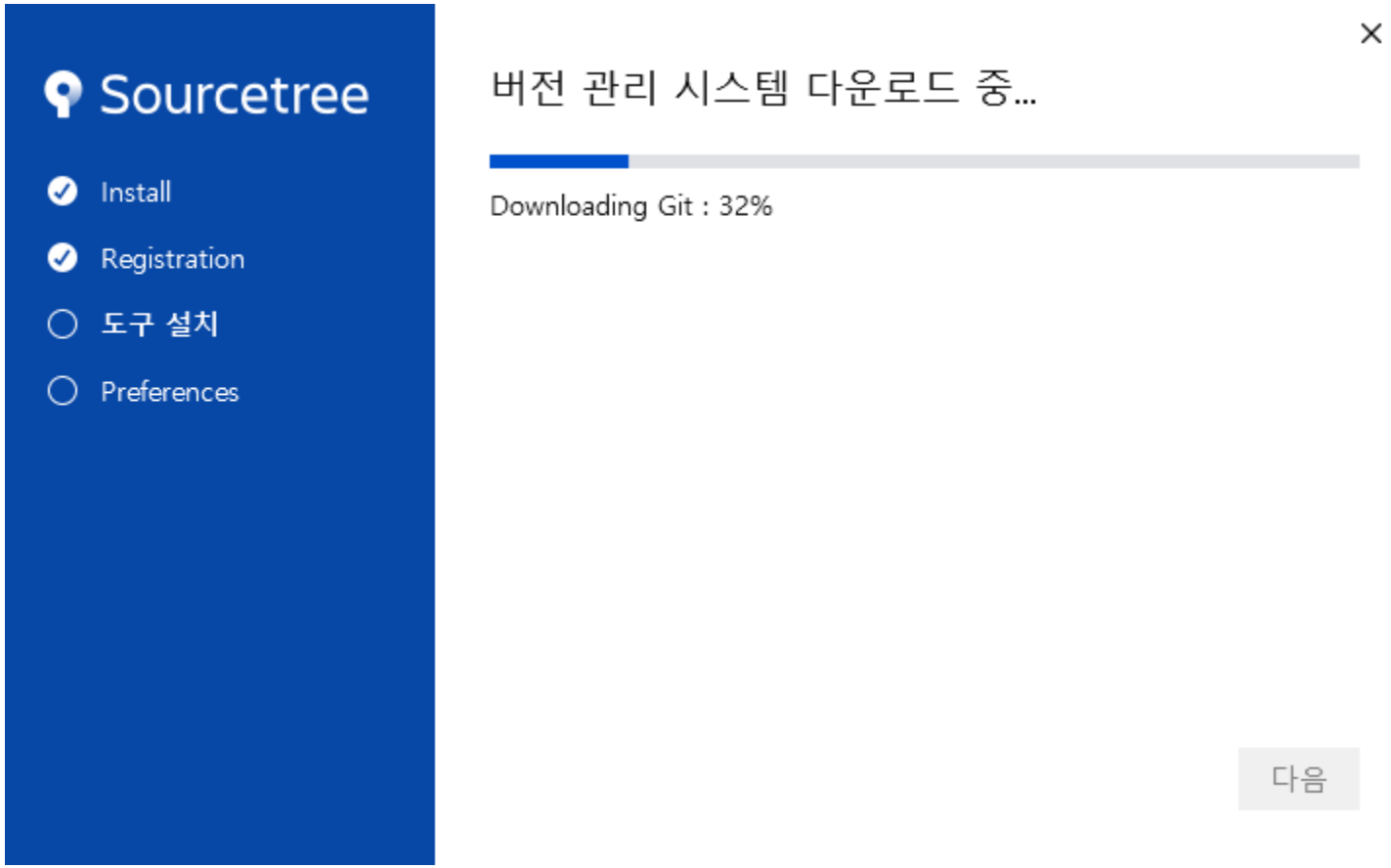
- **Mercurial 체크 해제 후, 다음**



1. 내 컴퓨터에 소스트리 설치하기

소스트리 설치하기

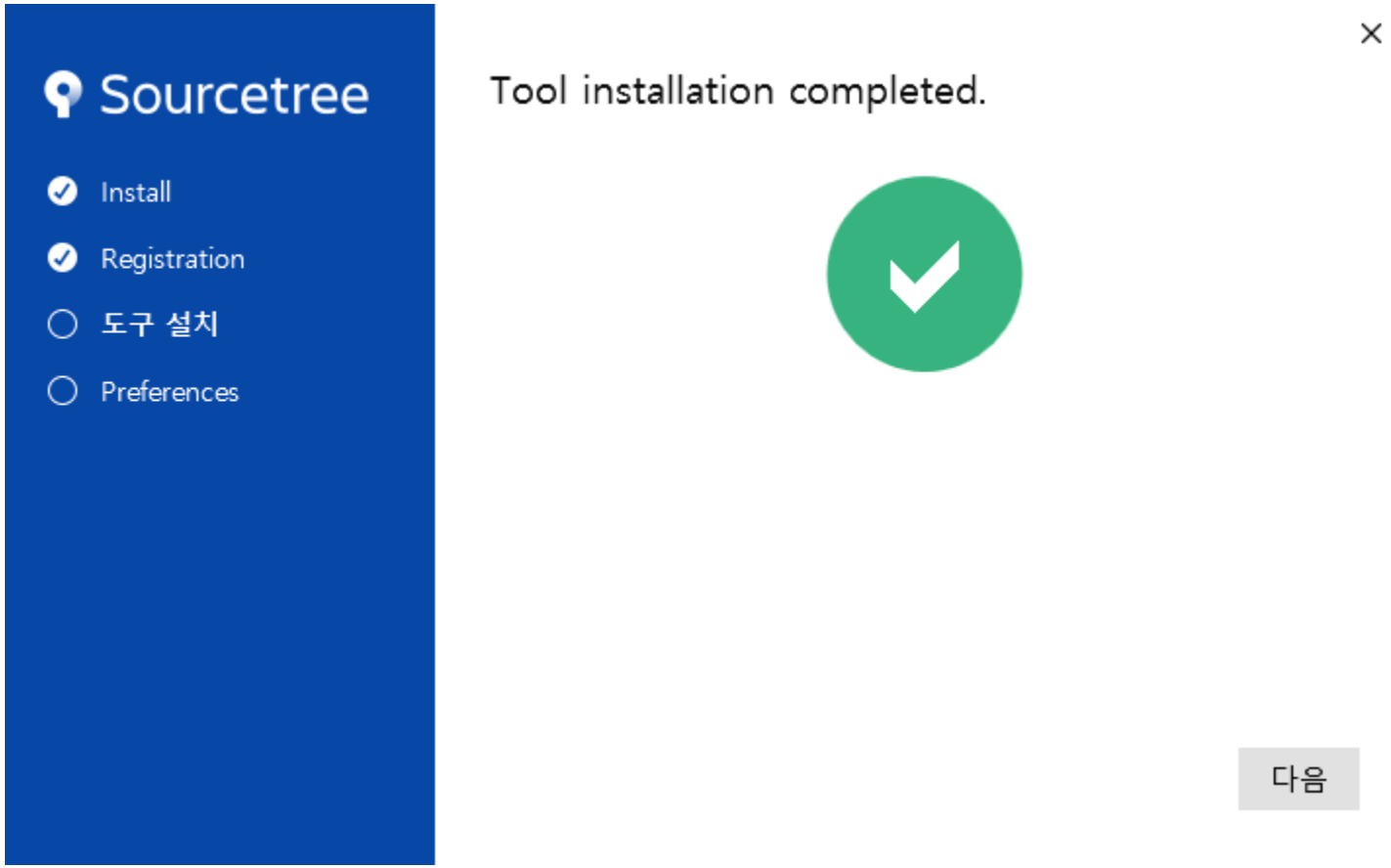
- 설치 진행



1. 내 컴퓨터에 소스트리 설치하기

소스트리 설치하기

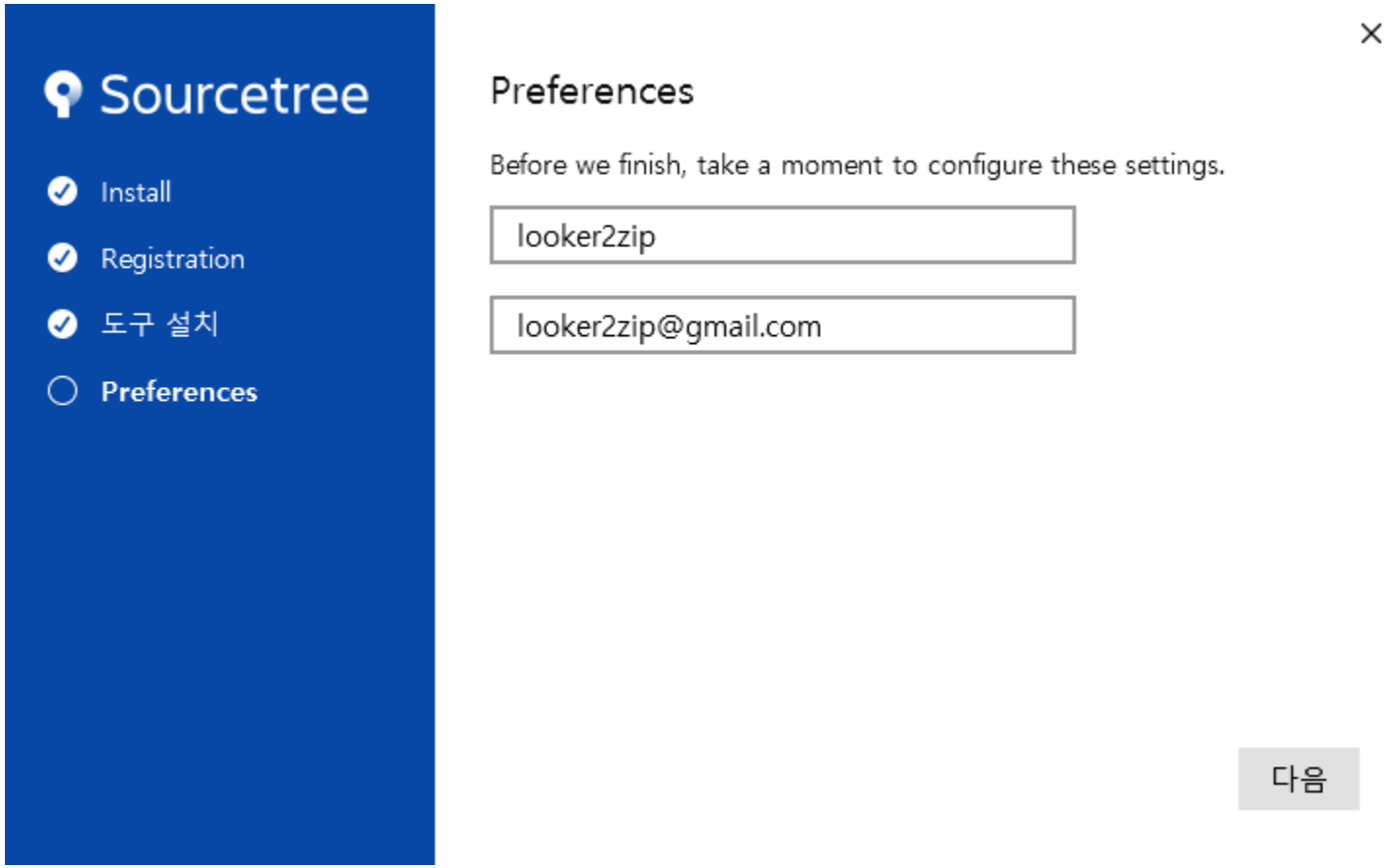
- 설치 완료



1. 내 컴퓨터에 소스트리 설치하기

소스트리 설치하기

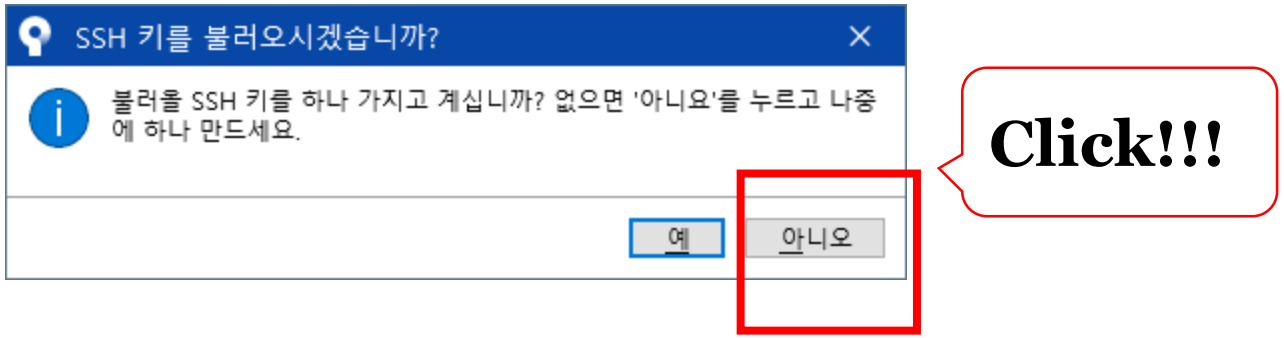
- 깃 계정



1. 내 컴퓨터에 소스트리 설치하기

소스트리 설치하기

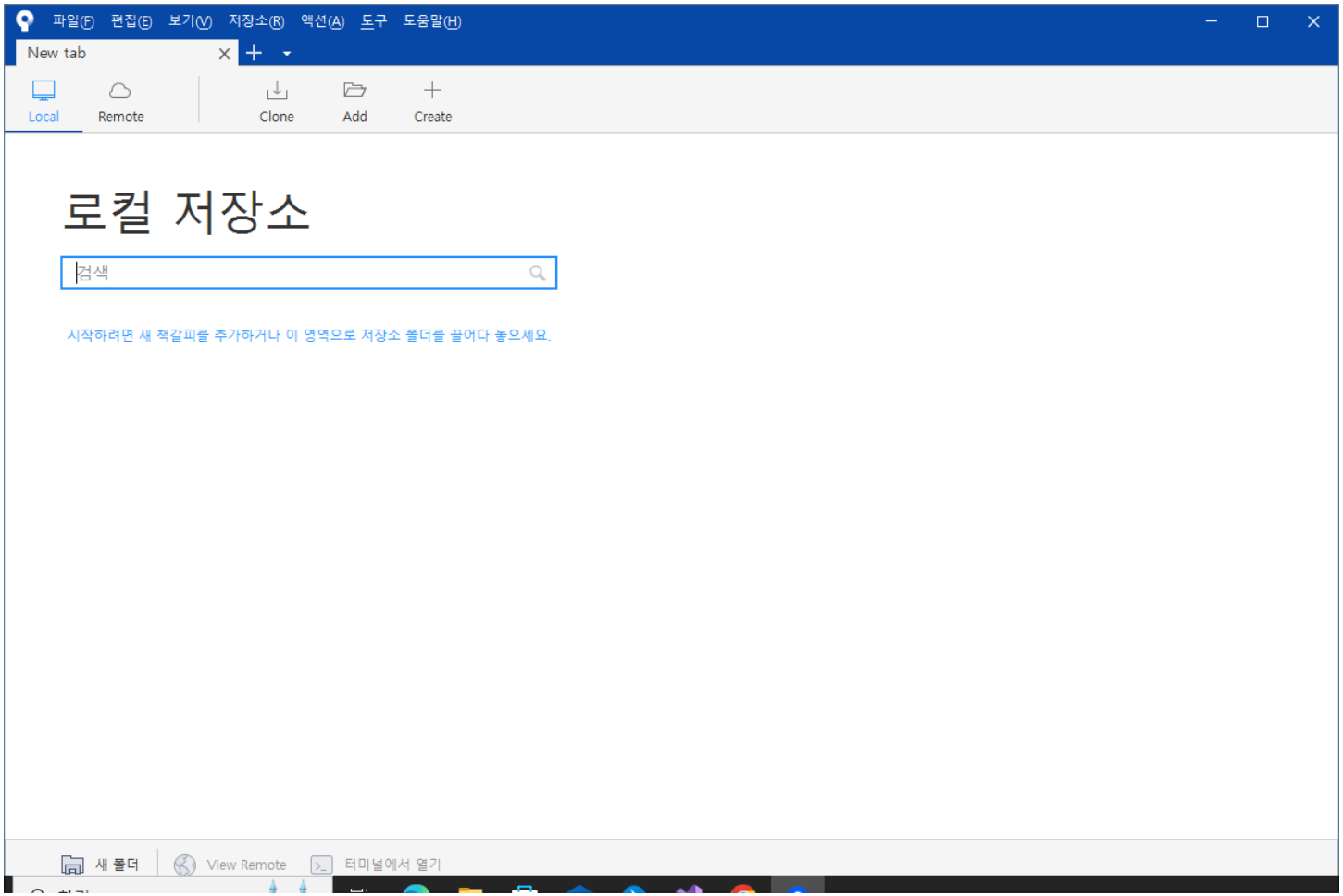
- 아니오 클릭



1. 내 컴퓨터에 소스트리 설치하기

소스트리 설치하기

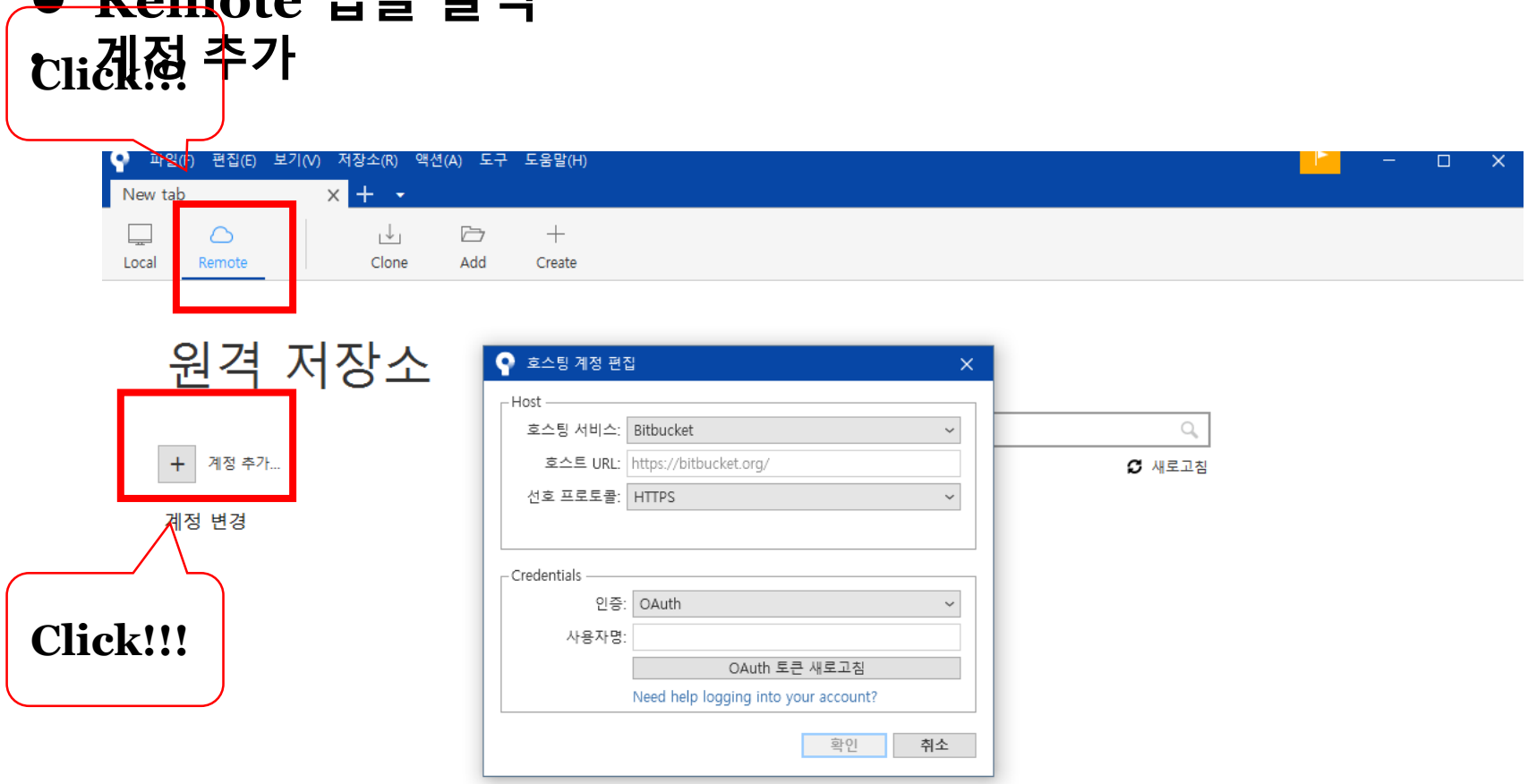
- 소스트리 메인 화면



1. 내 컴퓨터에 소스트리 설치하기

소스트리 소스트리에서 GitHub 로그인하기

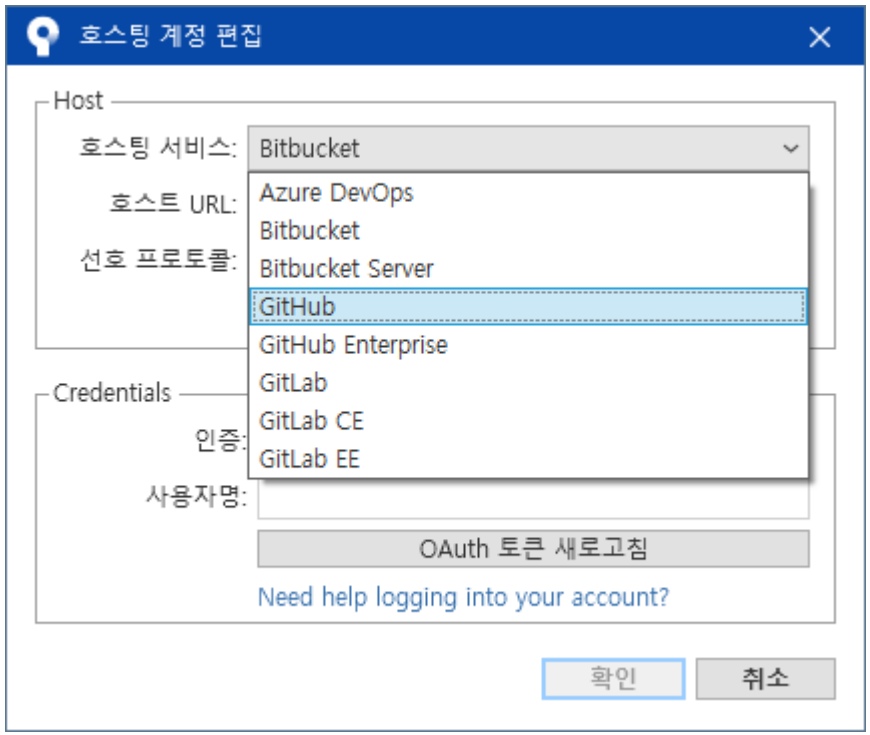
- **Remote 탭을 클릭**
  계정 추가

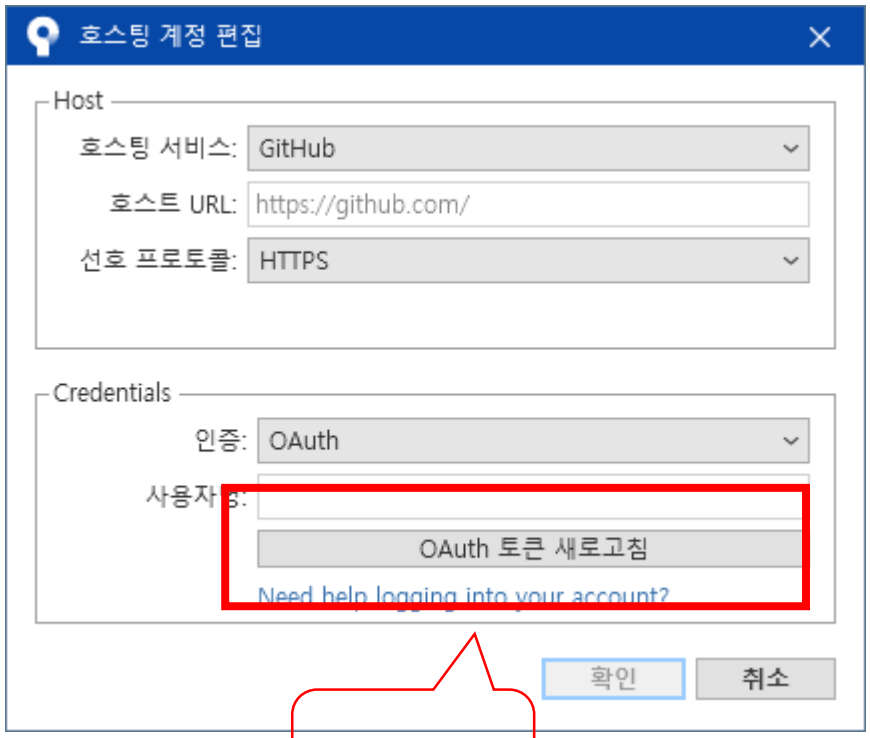


1. 내 컴퓨터에 소스트리 설치하기

소스트리 소스트리에서 GitHub 로그인하기

- **Remote 탭을 클릭**
- 계정 추가

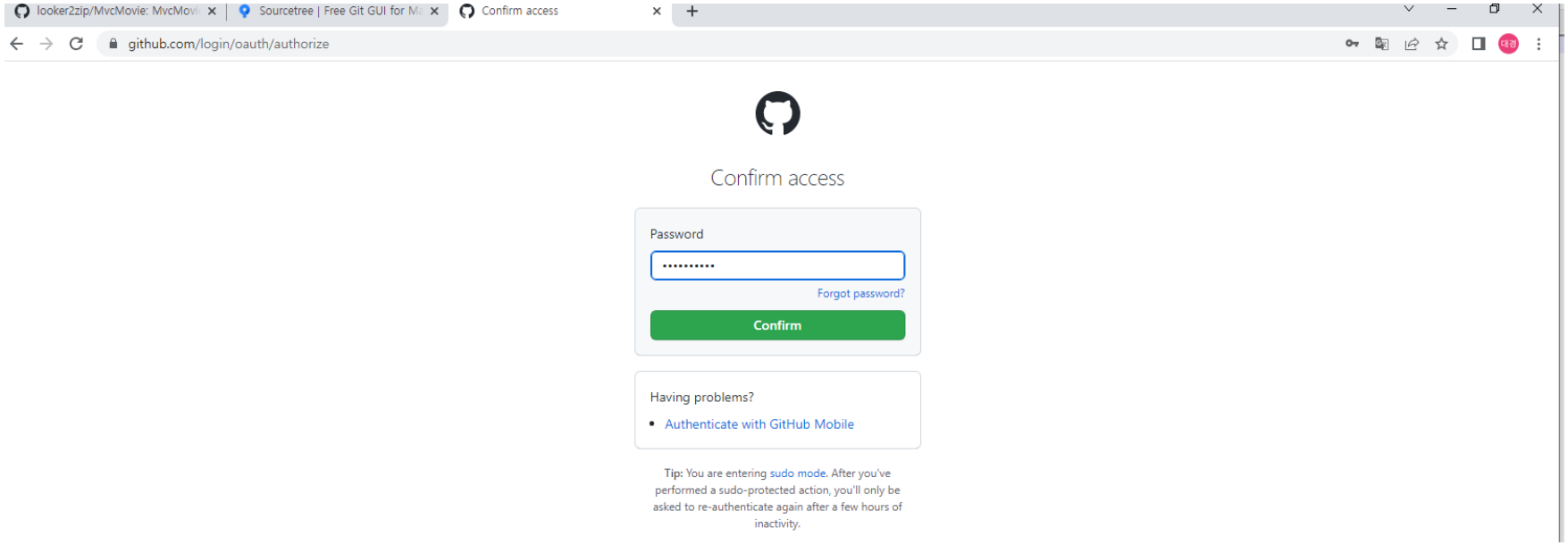




1. 내 컴퓨터에 소스트리 설치하기

소스트리 소스트리에서 GitHub 로그인하기

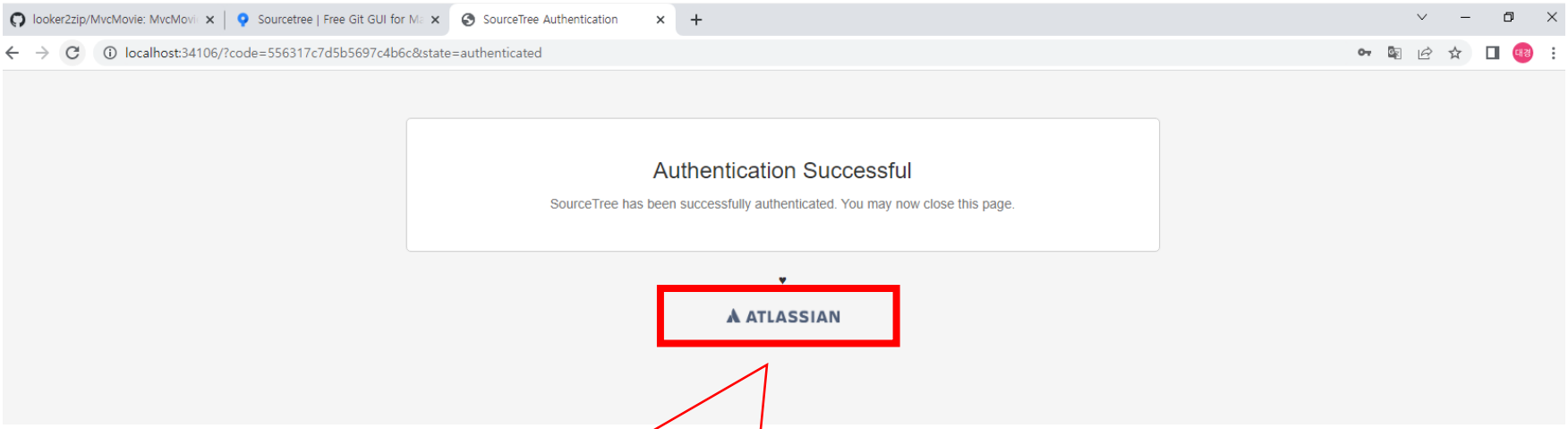
- **Use your password 클릭**



1. 내 컴퓨터에 소스트리 설치하기

소스트리 소스트리에서 GitHub 로그인하기

- A ATLASSIAN 클릭을 클릭해야 함

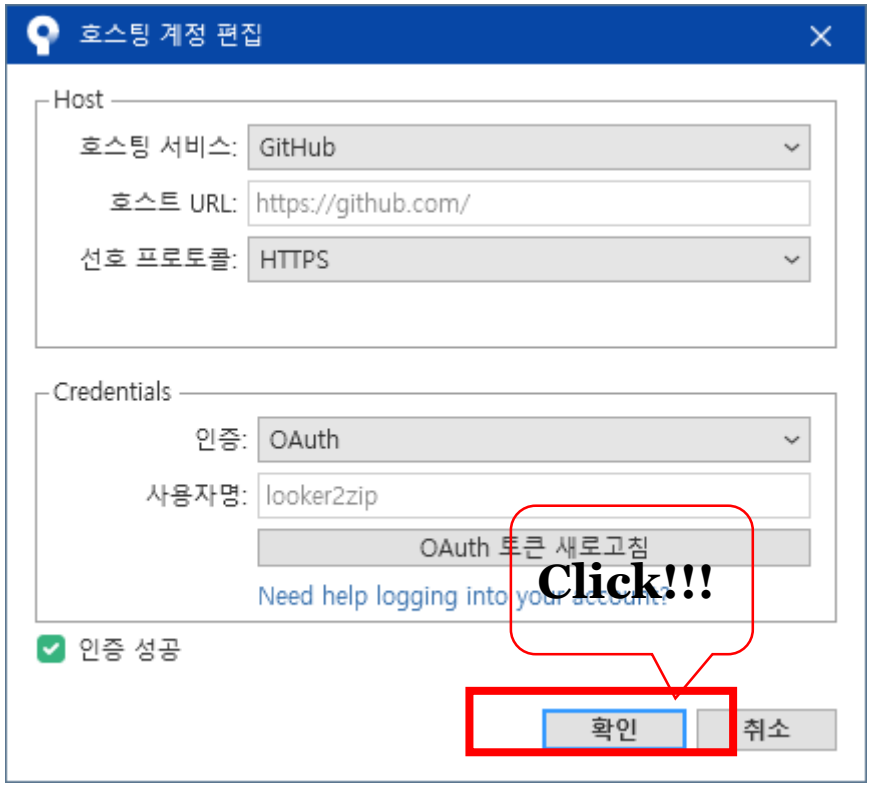


반반드시 Click해야 함!!!

1. 내 컴퓨터에 소스트리 설치하기

소스트리 소스트리에서 GitHub 로그인하기

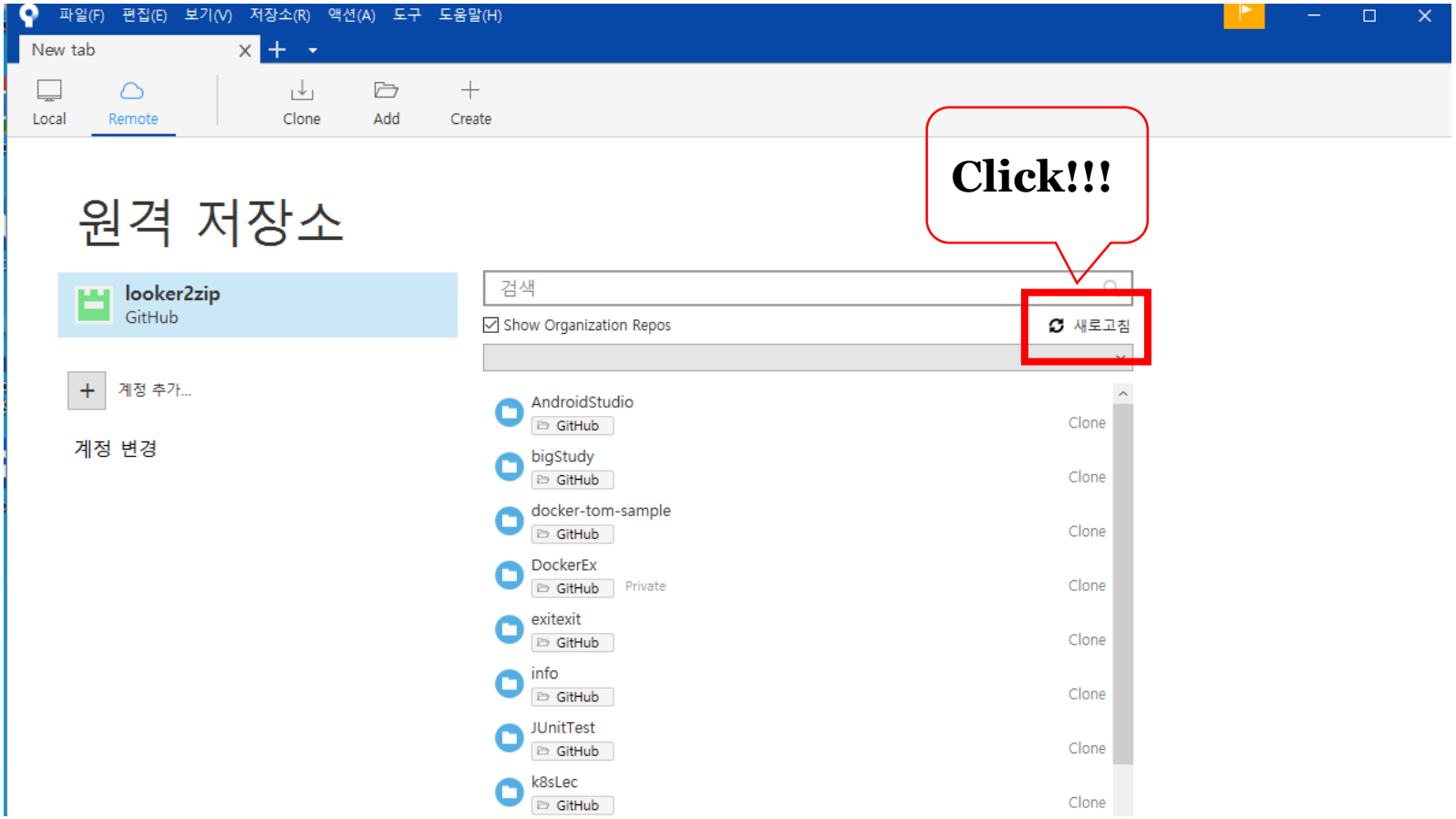
- 인증 성공



1. 내 컴퓨터에 소스트리 설치하기

소스트리 소스트리에서 GitHub 로그인하기

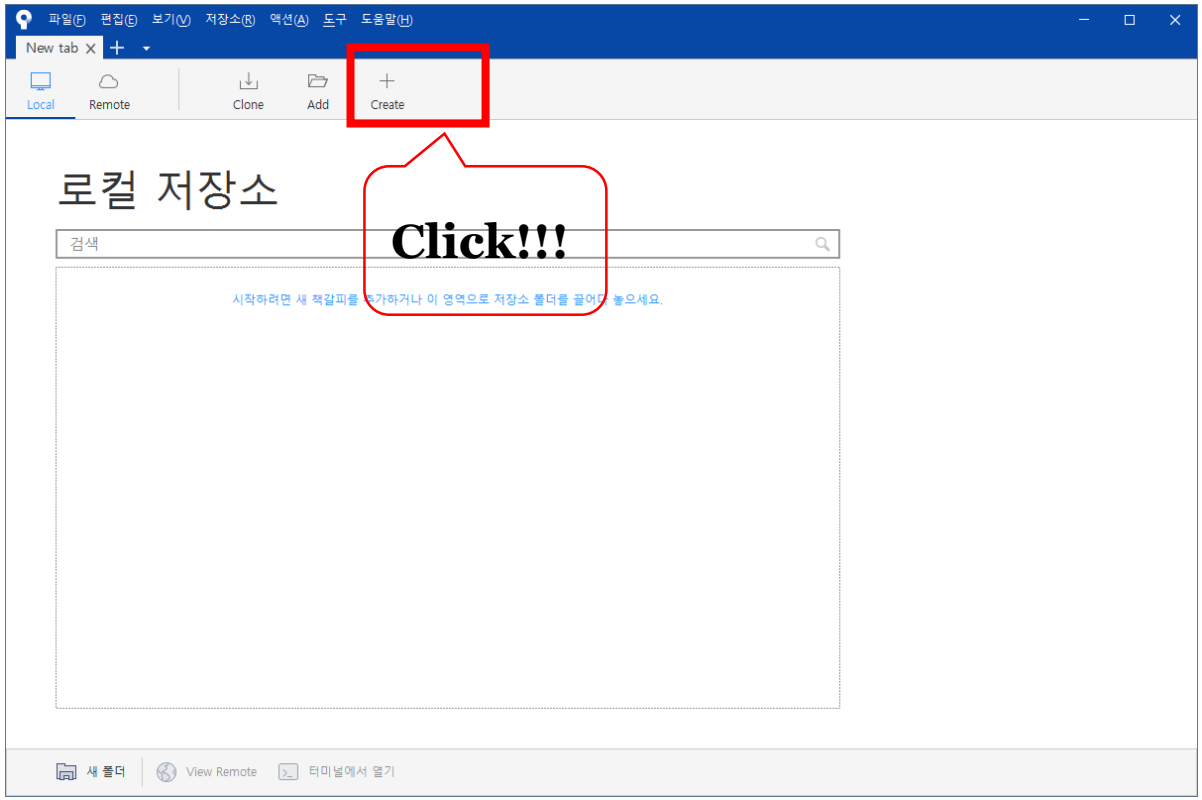
- 새로 고침을 클릭하면 원격 저장소 조회가 가능



1. 내 컴퓨터에 소스트리 설치하기

소스트리에서 깃 저장소 생성

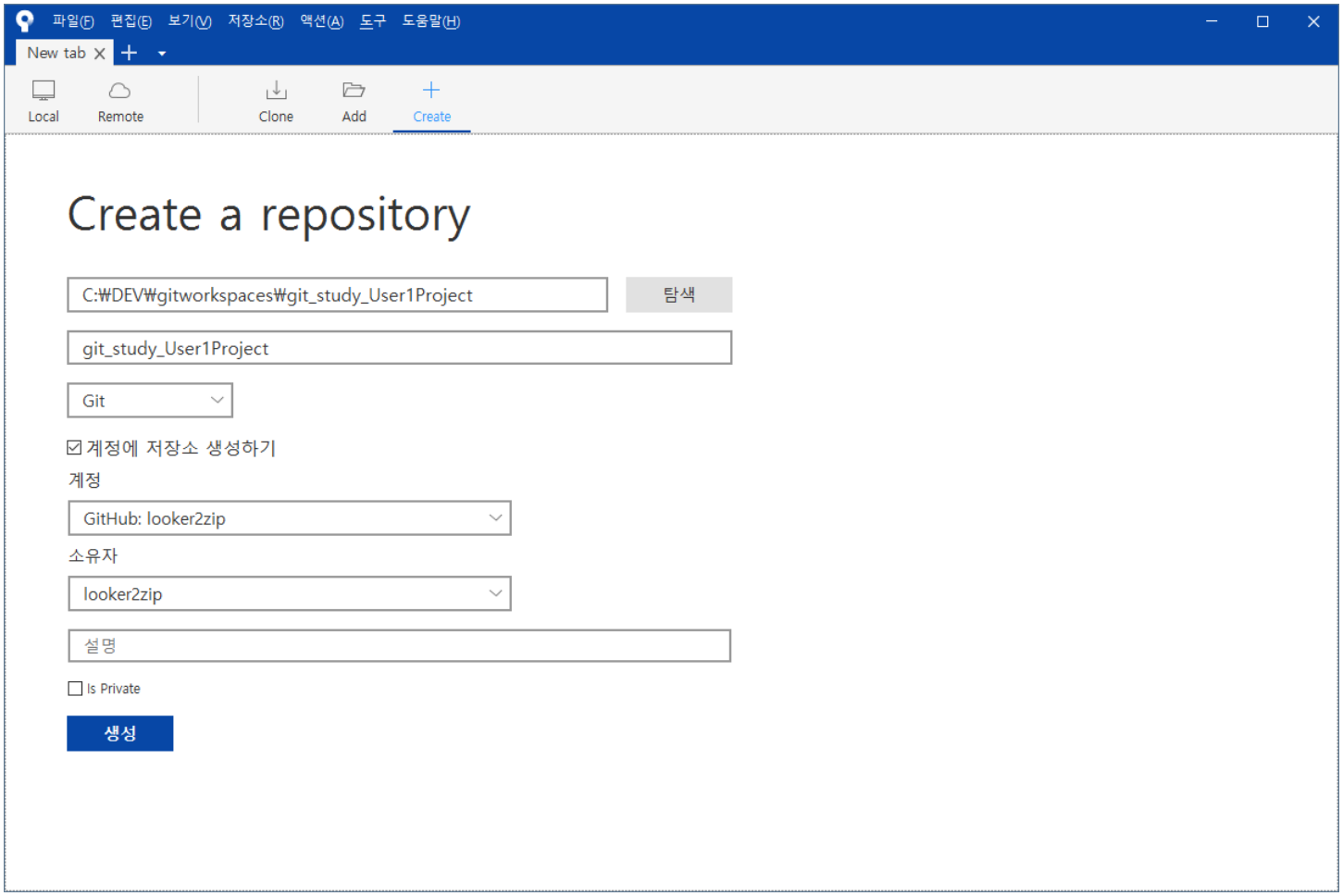
- 새로 고침을 클릭하면 원격 저장소 조회가 가능



1. 내 컴퓨터에 소스트리 설치하기

소스트리에서 깃 저장소 생성

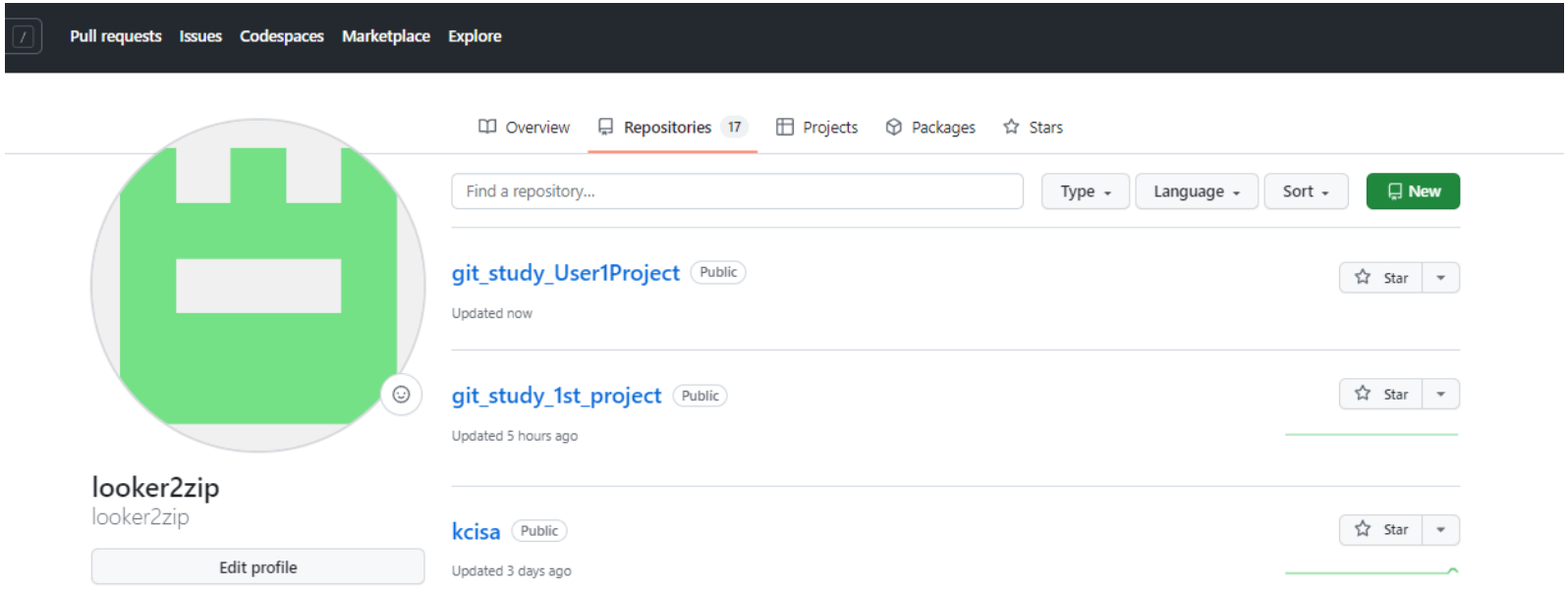
- 새로 고침을 클릭하면 원격 저장소 조회가 가능



1. 내 컴퓨터에 소스트리 설치하기

소스트리에서 깃 저장소 생성

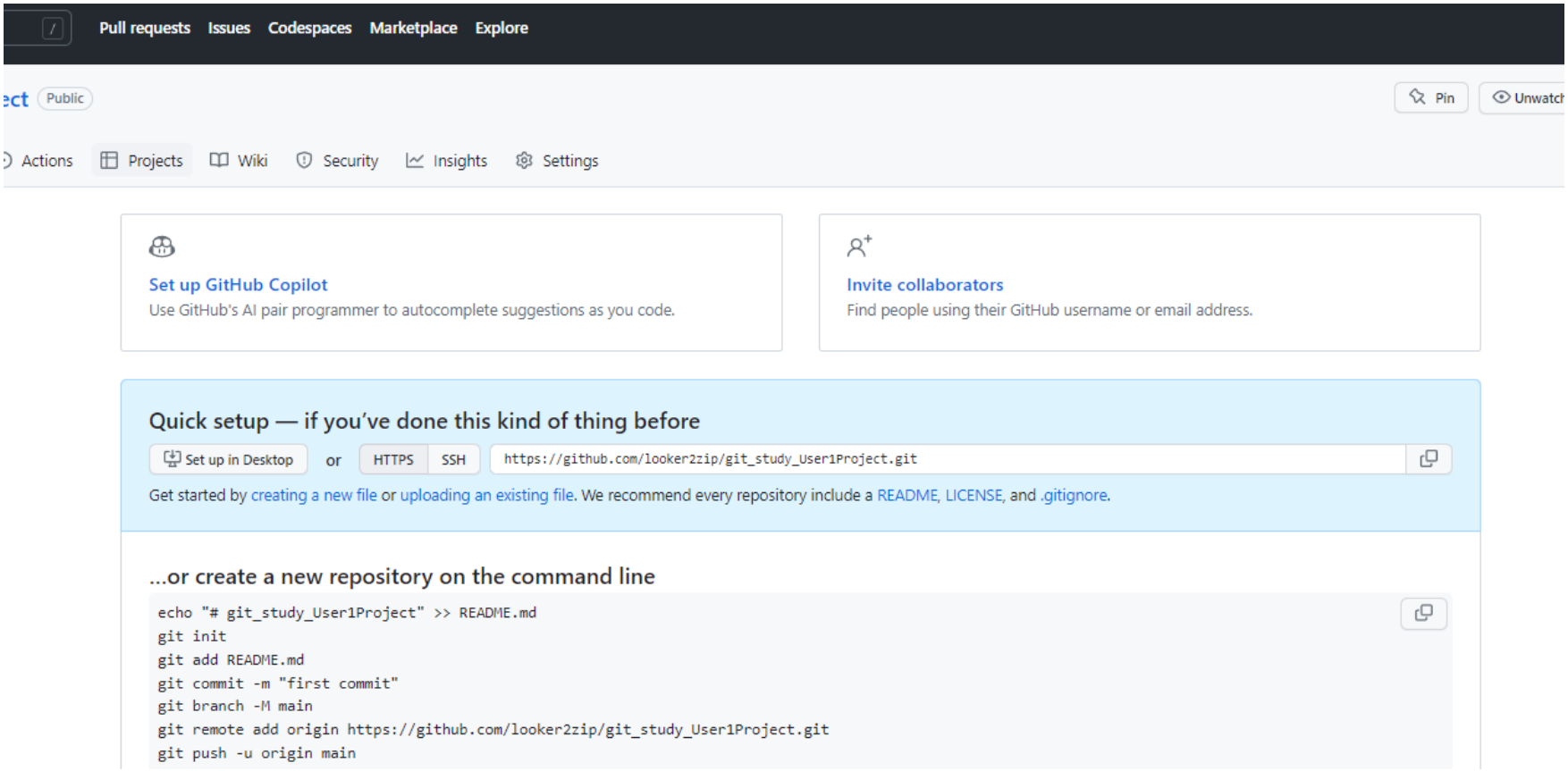
- 새로 고침을 클릭하면 원격 저장소 조회가 가능



1. 내 컴퓨터에 소스트리 설치하기

소스트리에서 깃 저장소 생성

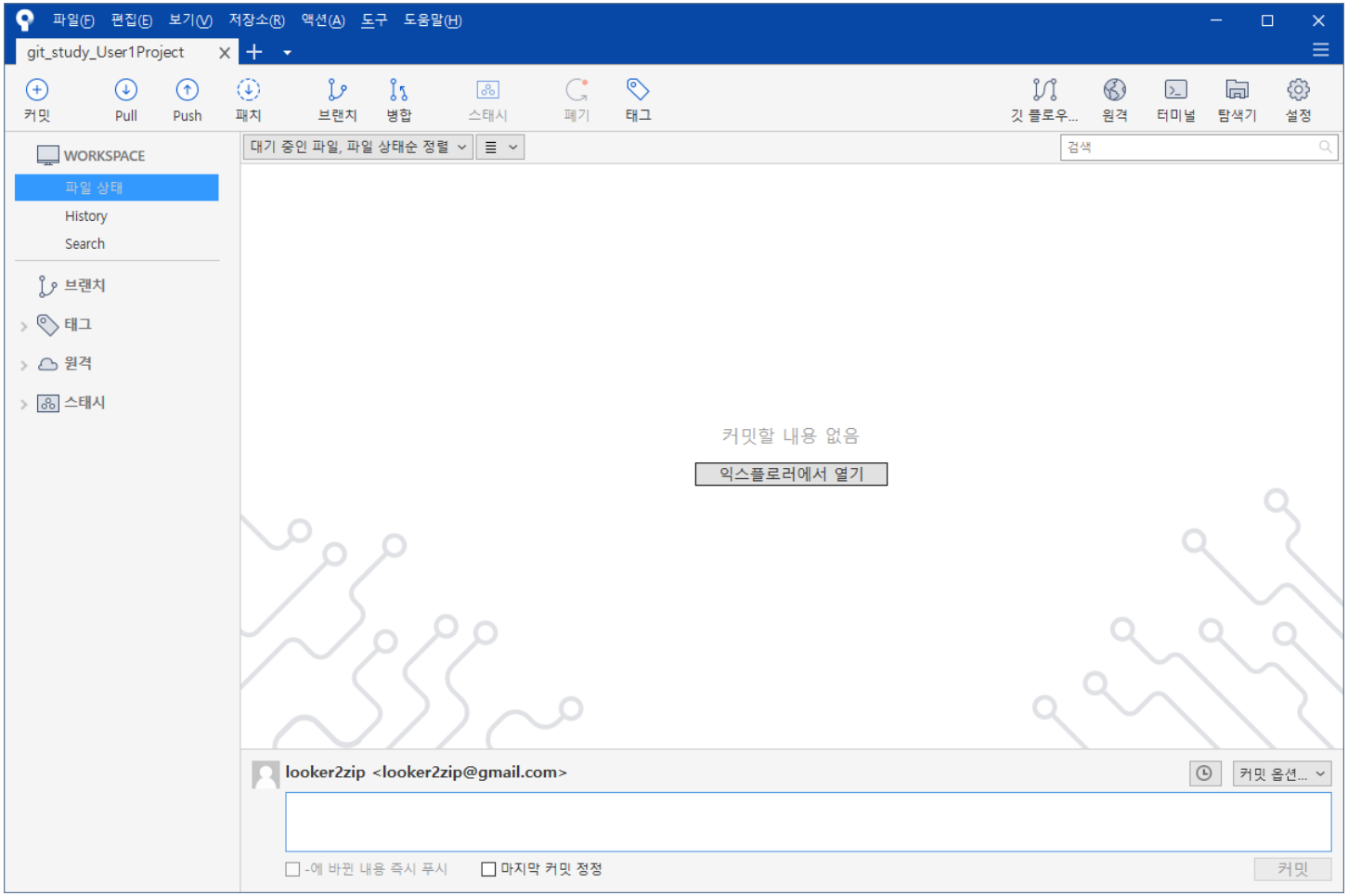
- 새로 고침을 클릭하면 원격 저장소 조회가 가능



1. 내 컴퓨터에 소스트리 설치하기

소스트리에서 깃 저장소 생성

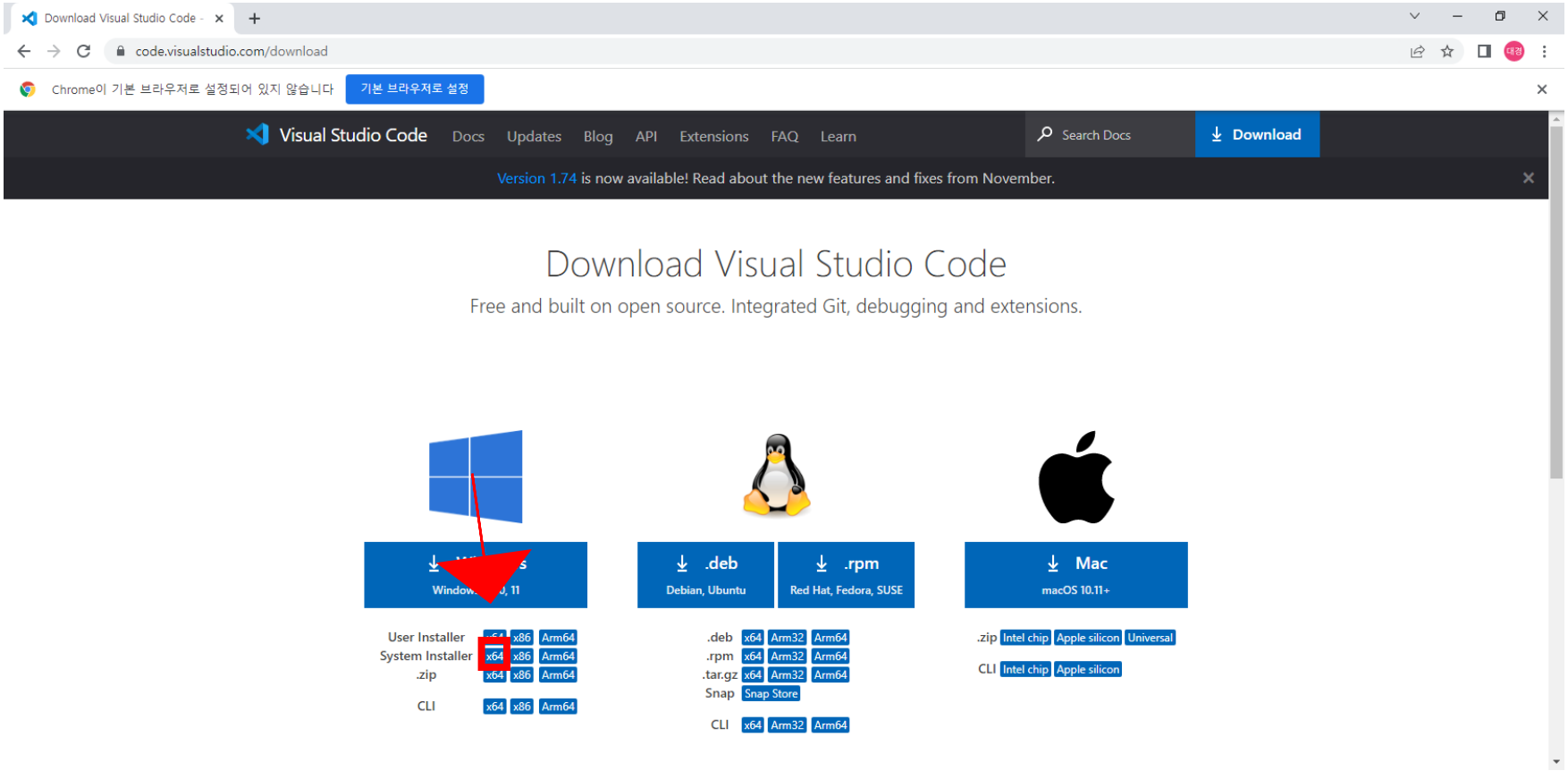
- 새로 고침을 클릭하면 원격 저장소 조회가 가능



2.Visual Studio Code 설치하기

Visual Studio Code 설치

https://code.visualstudio.com/download



2.Visual Studio Code 설치하기

Visual Studio Code 설치

User Installer, System Installer, zip 압축파일이 배포되고 있습니다.

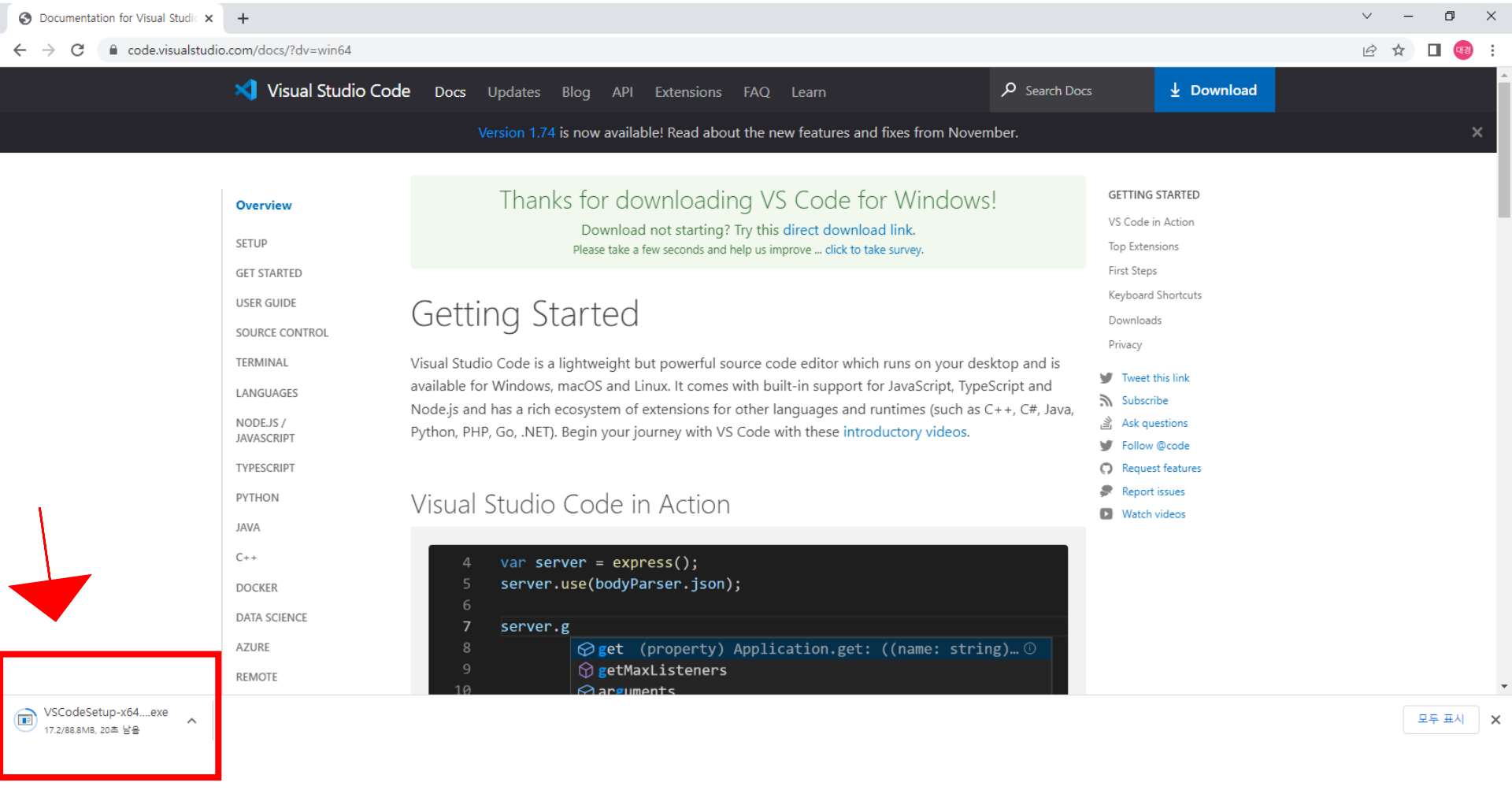
- **User Installer의 경우 다음 위치에 설치되며 유저 인터페이스의 디폴트 언어가 영어가 됩니다.**
- **C:\Users\사용자이름\AppData\Local\Programs\Microsoft VS Code**
- **System Installer의 경우에는 다음 위치에 설치되며 유저 인터페이스의 디폴트 언어가 영어가 됩니다.**

C:\Program Files\Microsoft VS Code

2.Visual Studio Code 설치하기

Visual Studio Code 설치

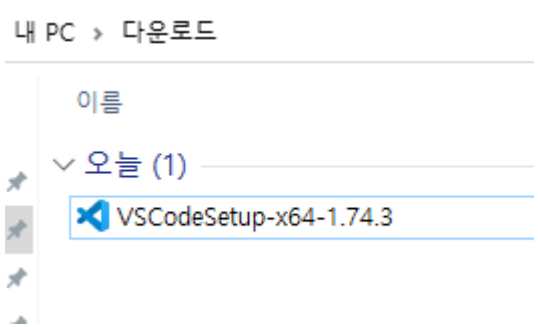
- **System Installer x64를 설치**

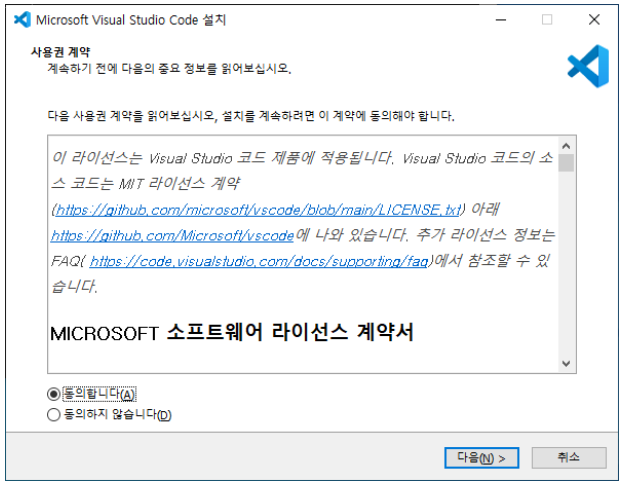


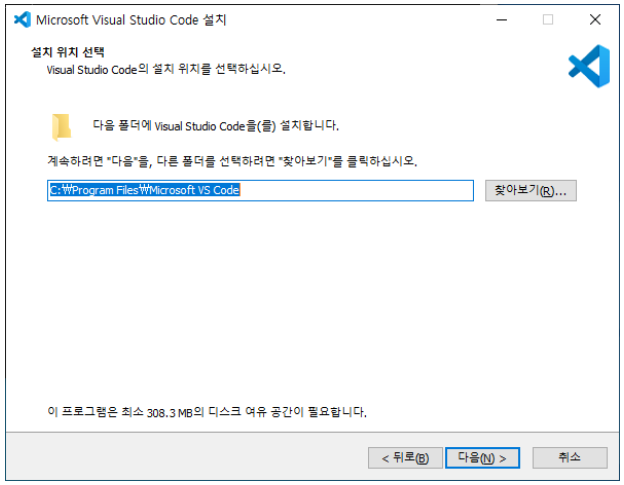
2.Visual Studio Code 설치하기

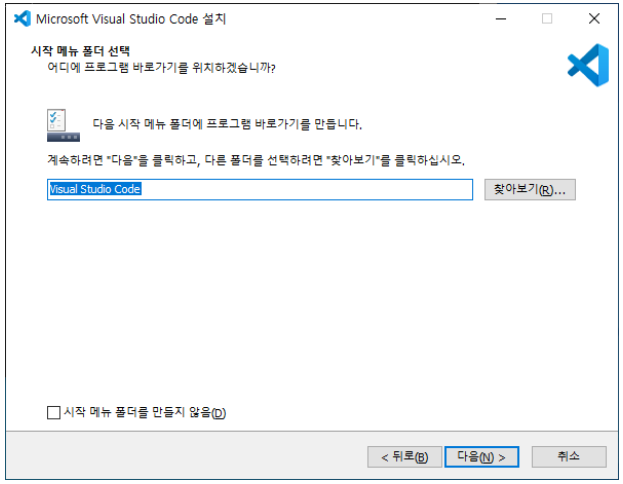
Visual Studio Code 설치

Visual Studio Code 설치





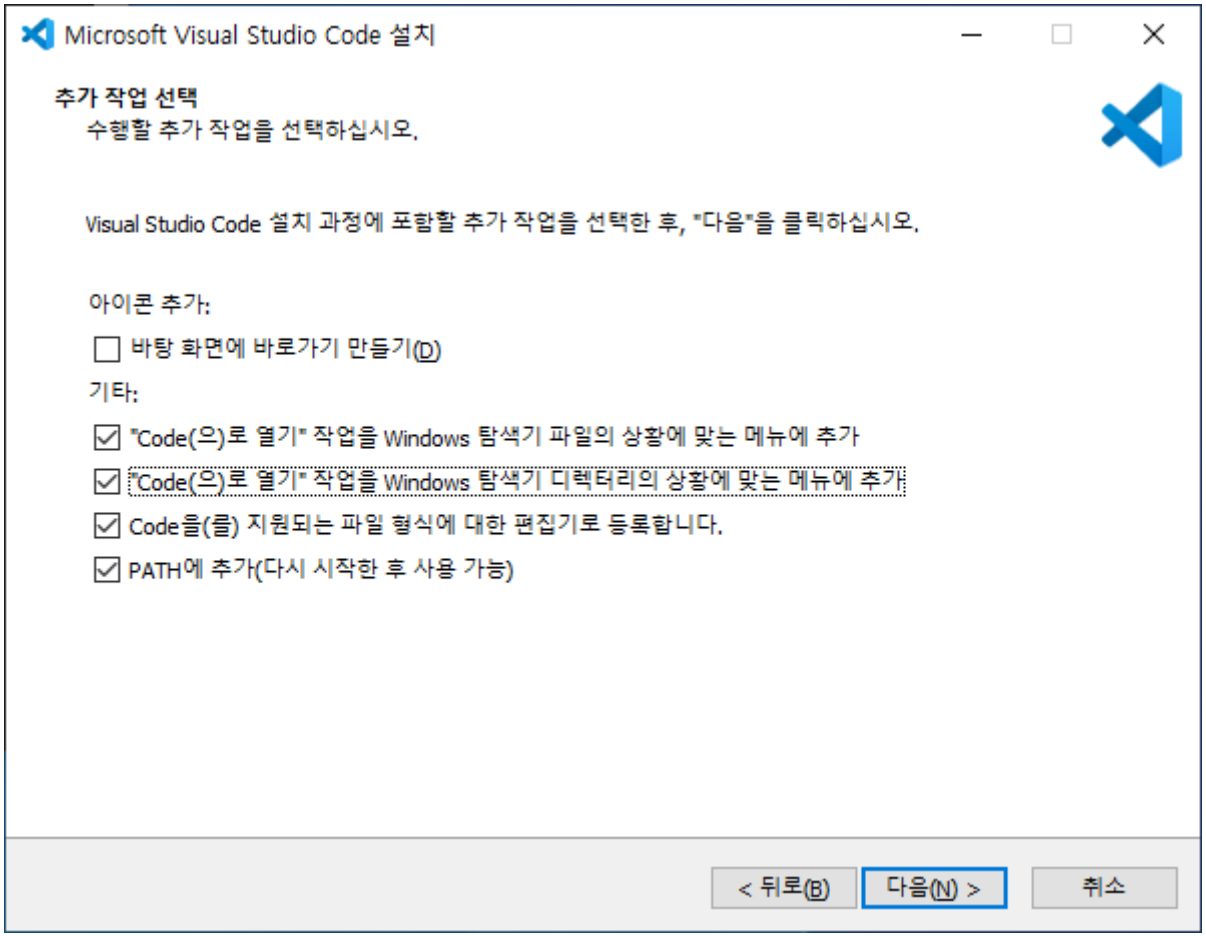




2.Visual Studio Code 설치하기

Visual Studio Code 설치

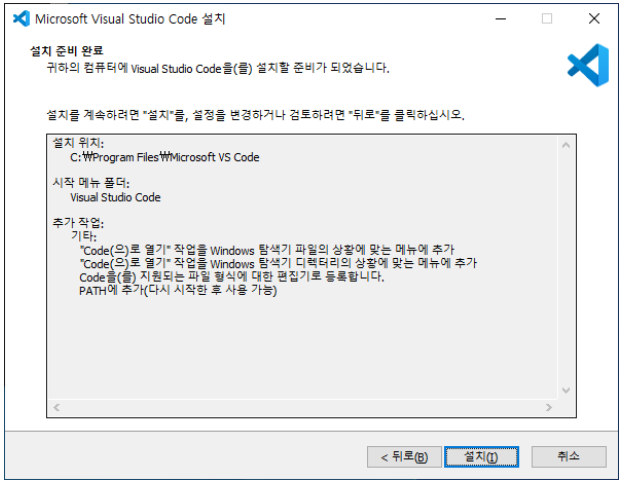
Visual Studio Code 설치

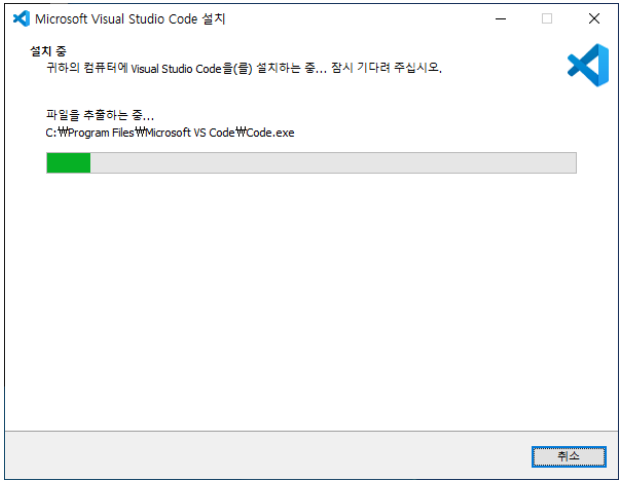


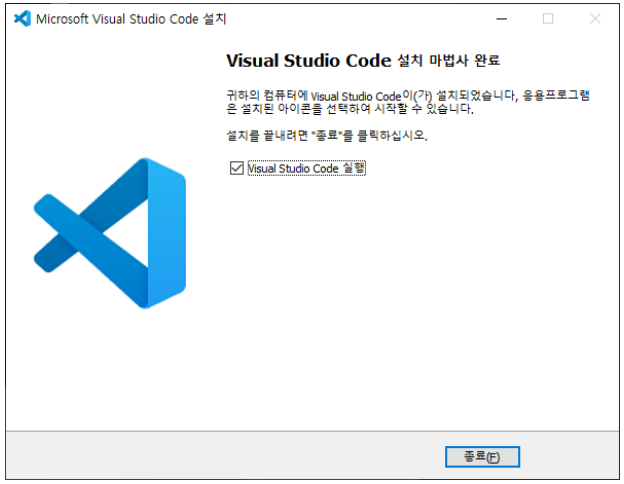
2.Visual Studio Code 설치하기

Visual Studio Code 설치

Visual Studio Code 설치



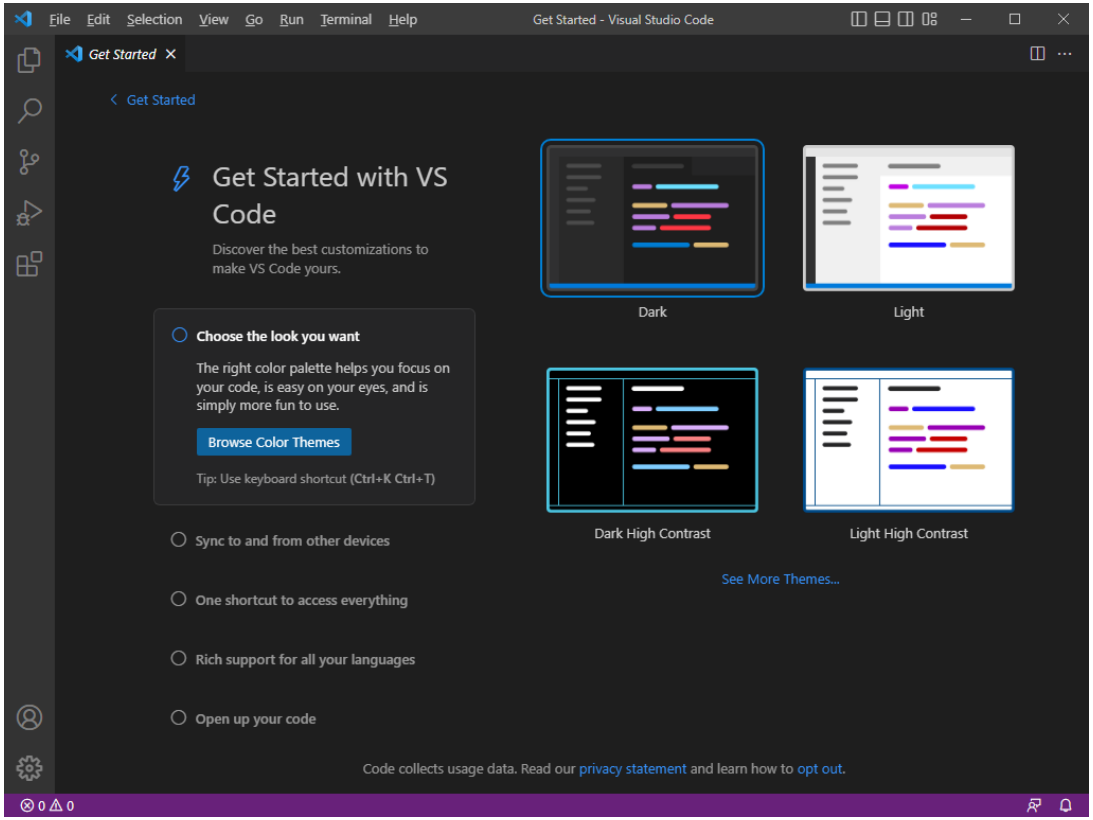




2.Visual Studio Code 설치하기

Visual Studio Code 설치

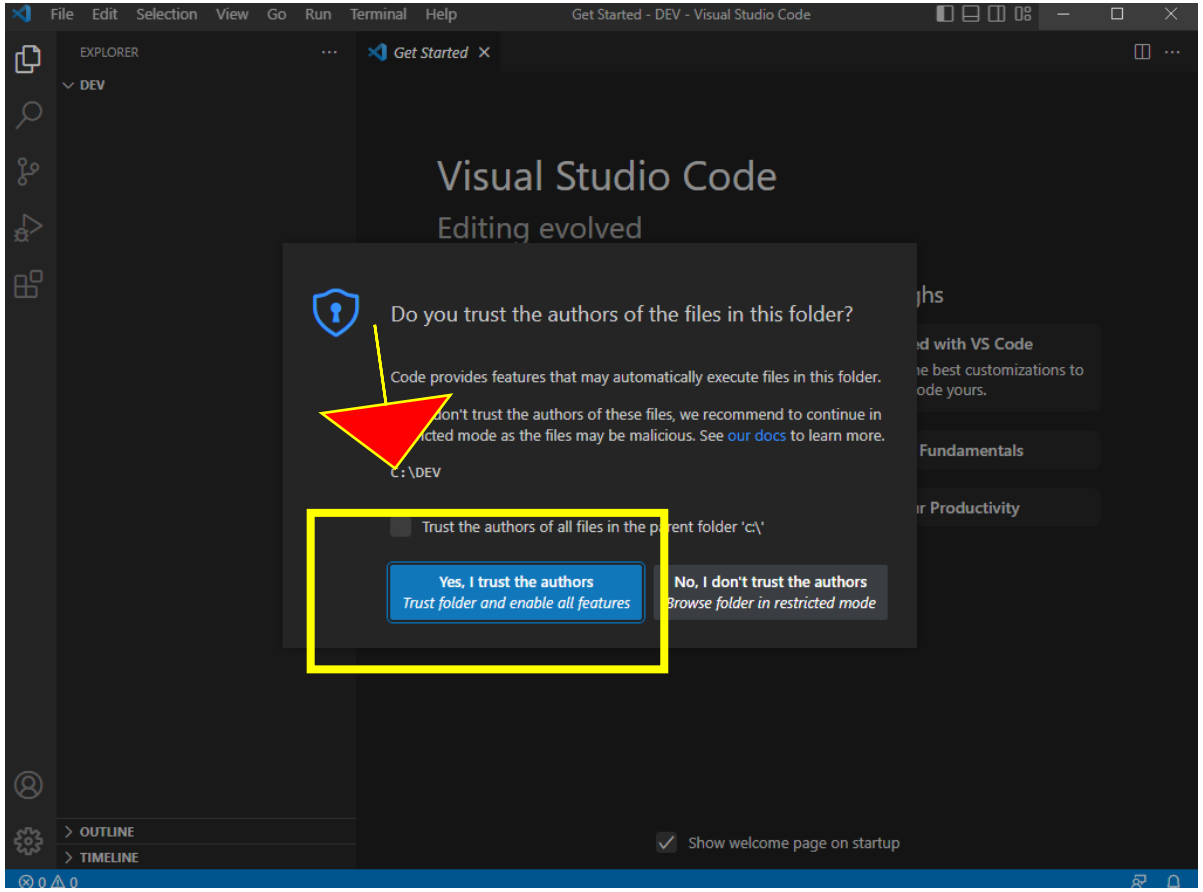
Visual Studio Code 설치



2.Visual Studio Code 설치하기

Visual Studio Code 설치

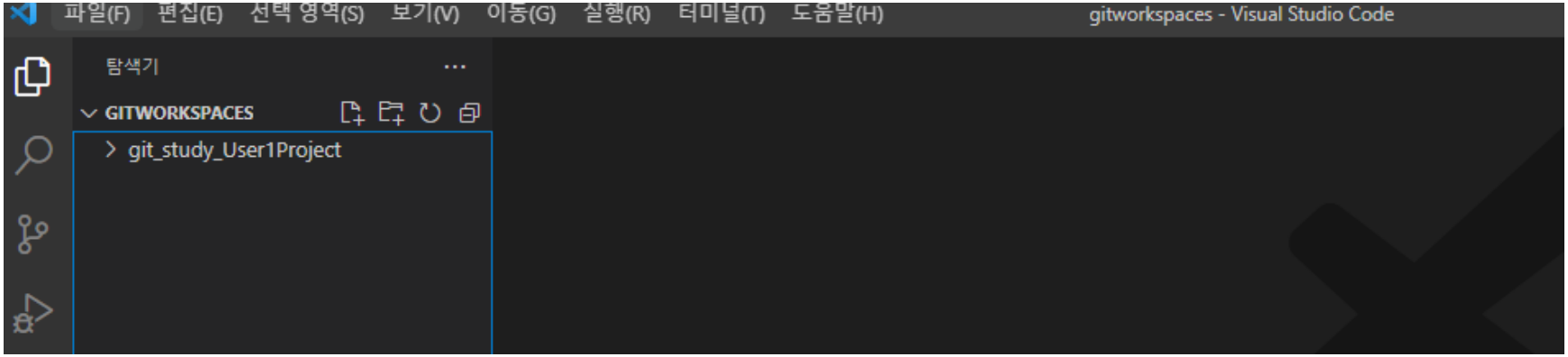
❖**Visual Studio Code를 실행하여 메뉴에서 File > Open Folder를 선택**



2.Visual Studio Code 설치하기

Visual Studio Code 설치

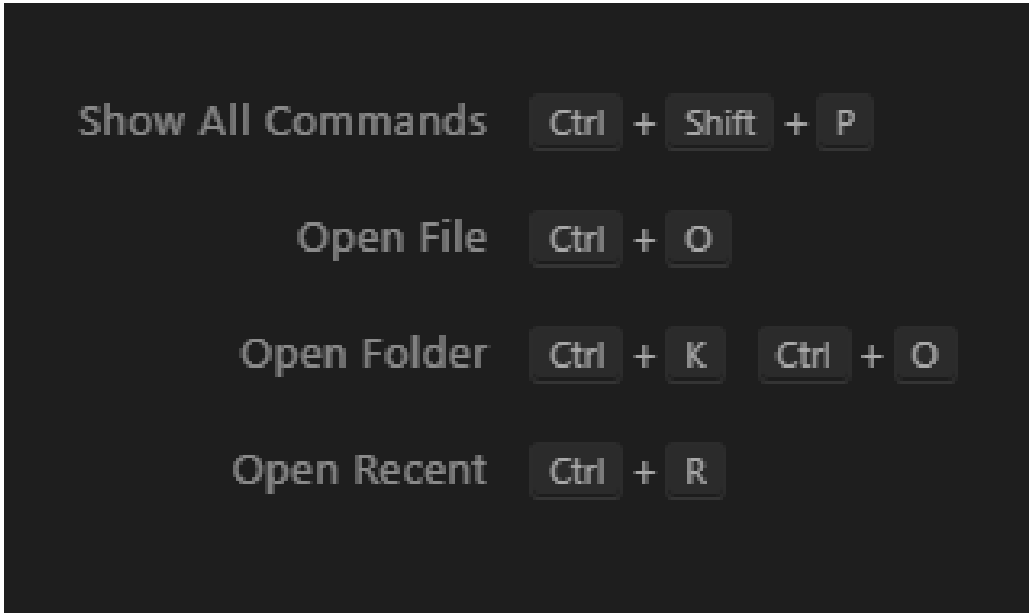
❖Visual Studio Code를 실행하여 메뉴에서 File > Open Folder를 선택



3. 소스트리 기본 수행

VSCode로 파일 만들고 수정하기

- **VSCode 기본 명령어**



3. 소스트리 기본 수행

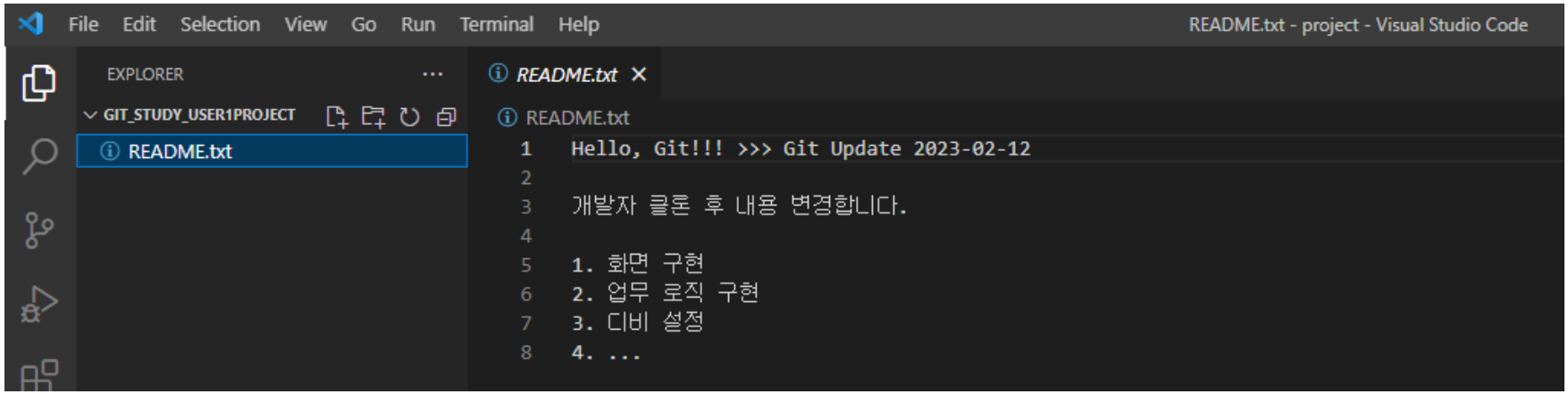
VSCode로 파일 만들고 수정하기

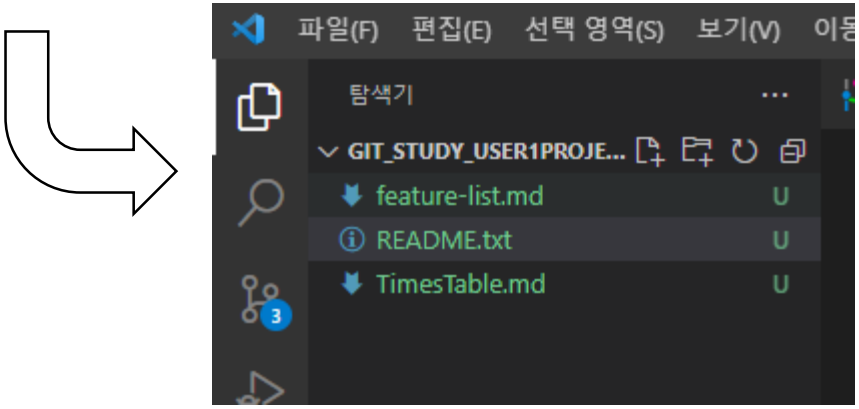
```
pro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces
$ cd git_study_User1Project/

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_User1Project (main)
$ ls -al
total 8
drwxr-xr-x 1 apro621 197121 0  6월  4 20:37 ./
drwxr-xr-x 1 apro621 197121 0  6월  4 20:35 ../
drwxr-xr-x 1 apro621 197121 0  6월  4 20:37 .git/

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_User1Project (main)
$ echo >> TimesTable.md

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_User1Project (main)
$ echo >> README.txt

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_User1Project (main)
$ echo >> feature-list.md

apro621@DESKTOP-CL1JPPC MINGW64 /c/DEV/gitworkspaces/git_study_User1Project (main)
$
```

3. 소스트리 기본 수행

VSCode로 파일 만들고 수정하기

- VSCode 새 파일 클릭 > TimesTable.md 입력



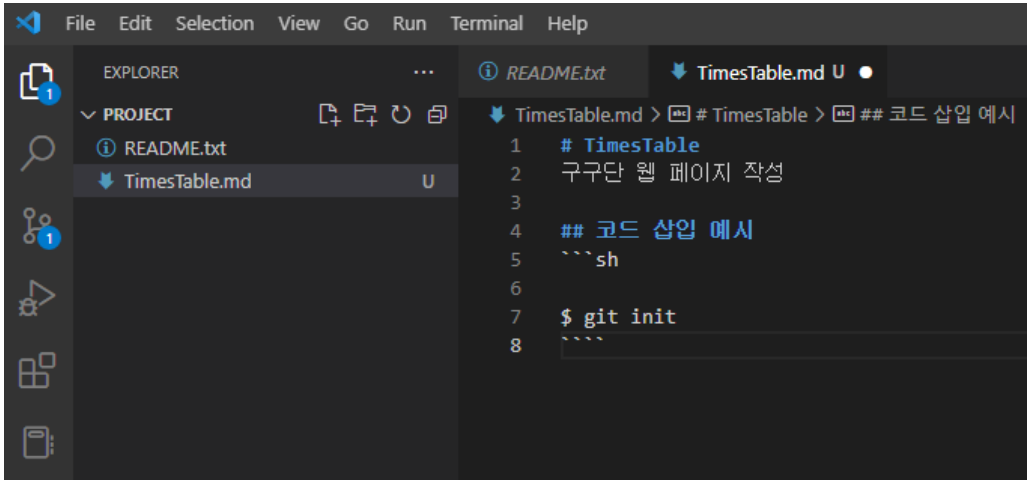
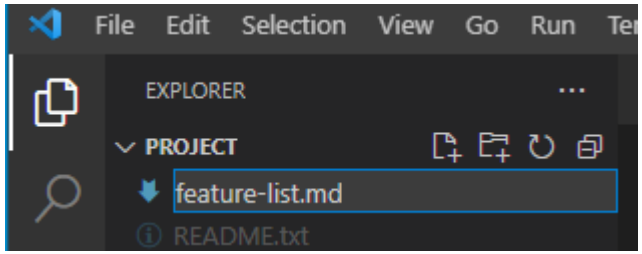
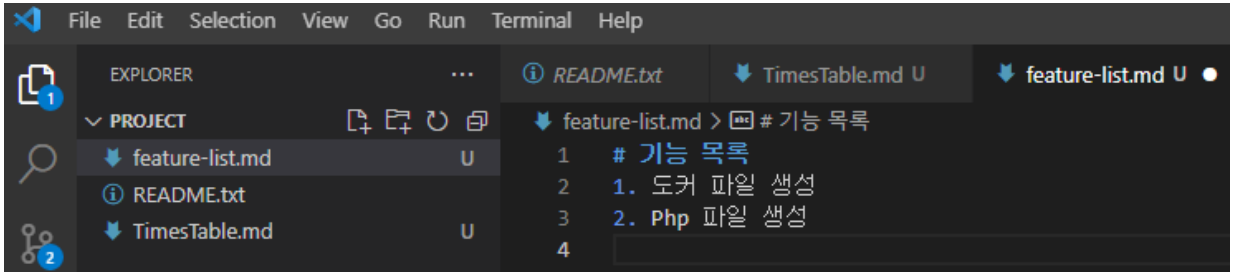


3. 소스트리 기본 수행

VSCode로 파일 만들고 수정하기

- **VSCode 새 파일 클릭 > TimesTable.md 입력**

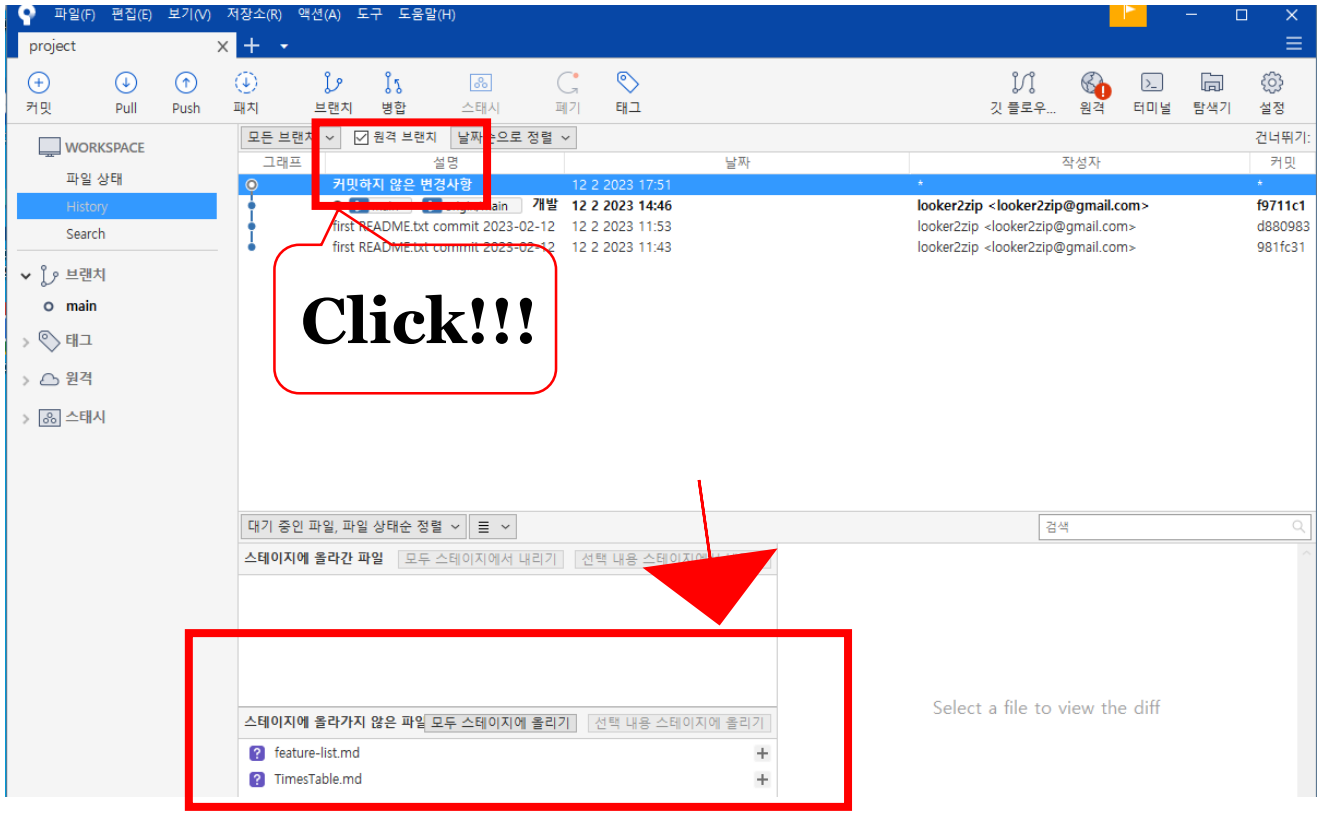
````
# TimesTable
구구단 웹 페이지 작성

## 코드 삽입 예시
```sh

$ git init
````


````

3. 소스트리 기본 수행

VSCode로 파일 만들고 수정하기

- VSCode 새 파일 클릭 > feature-list.md 입력




```
# 기능 목록
1. 도커 파일 생성
2. Php 파일 생성
````
```


```

3. 소스트리 기본 수행

소스트리에서 파일을 선택하고 커밋하기

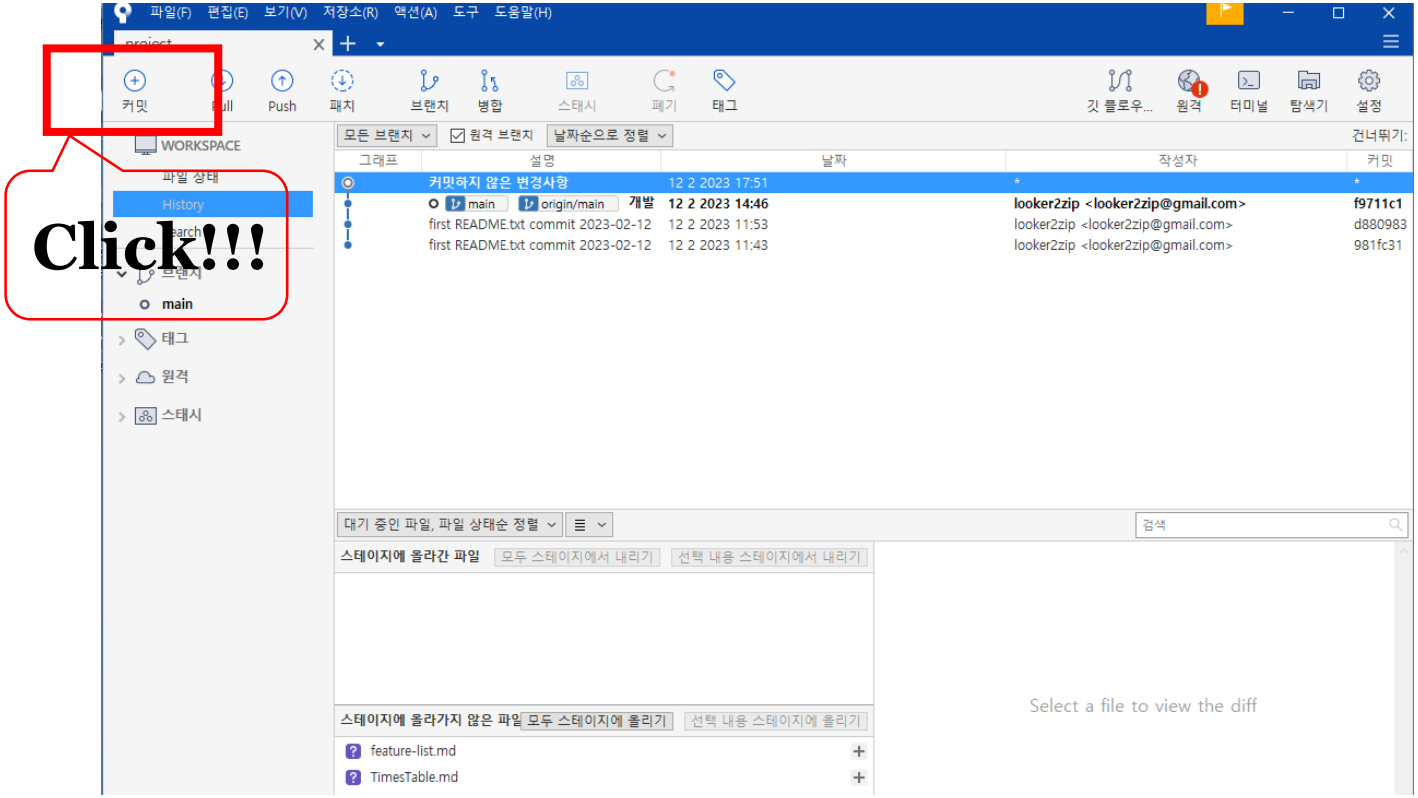
- 소스트리로 돌아모면 그래프 최상단에 '커밋하지 않은 변경사항'이 보임
- 이 텍스트를 클릭하면 소스트리 하다 [스테이지에 올라가지 않은 파일]섹션에 방금 만든 파일 2개가 보임



3. 소스트리 기본 수행

커밋

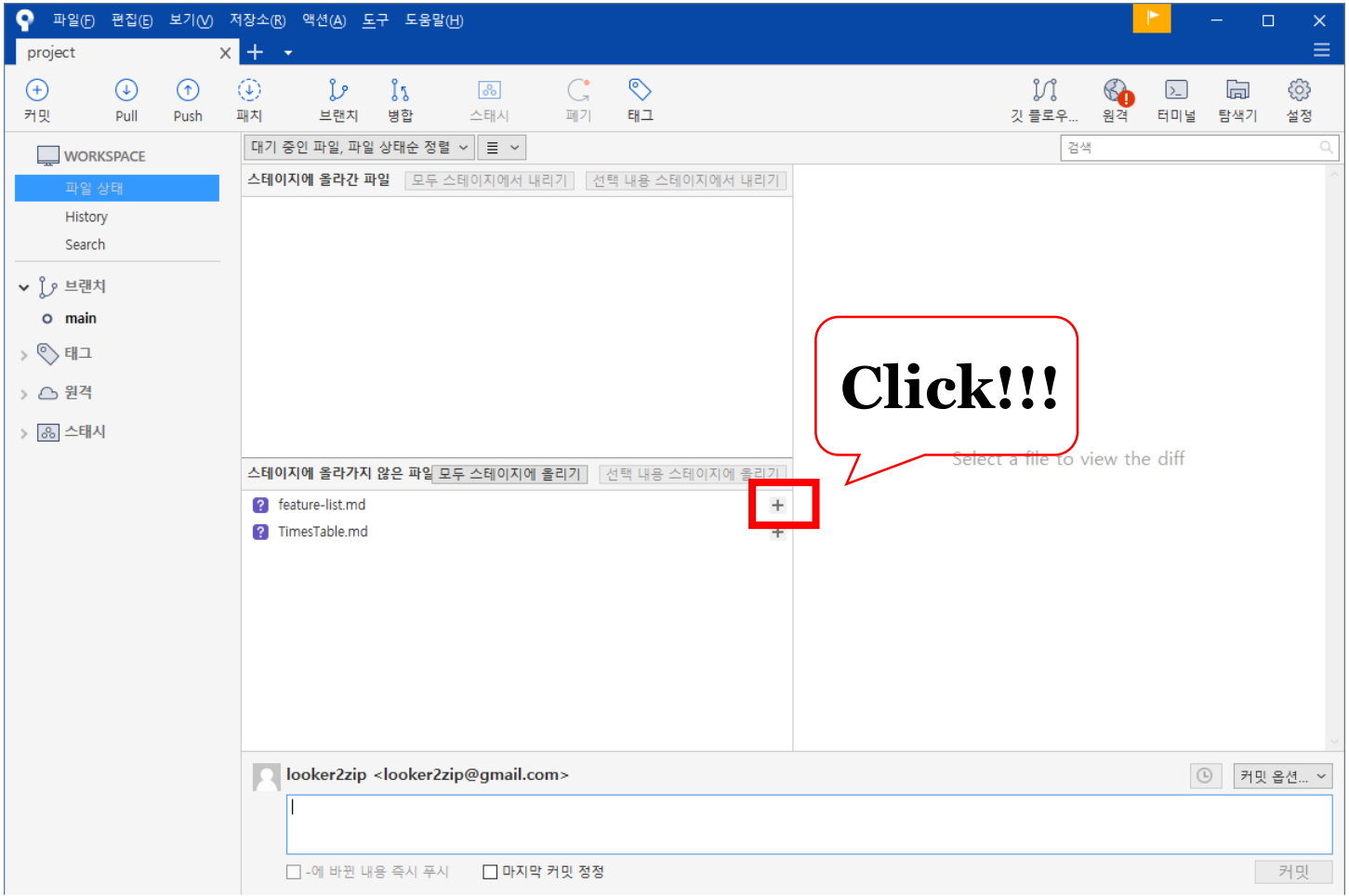
- 우상단에 커밋 아이콘을 클릭



3. 소스트리 기본 수행

커밋

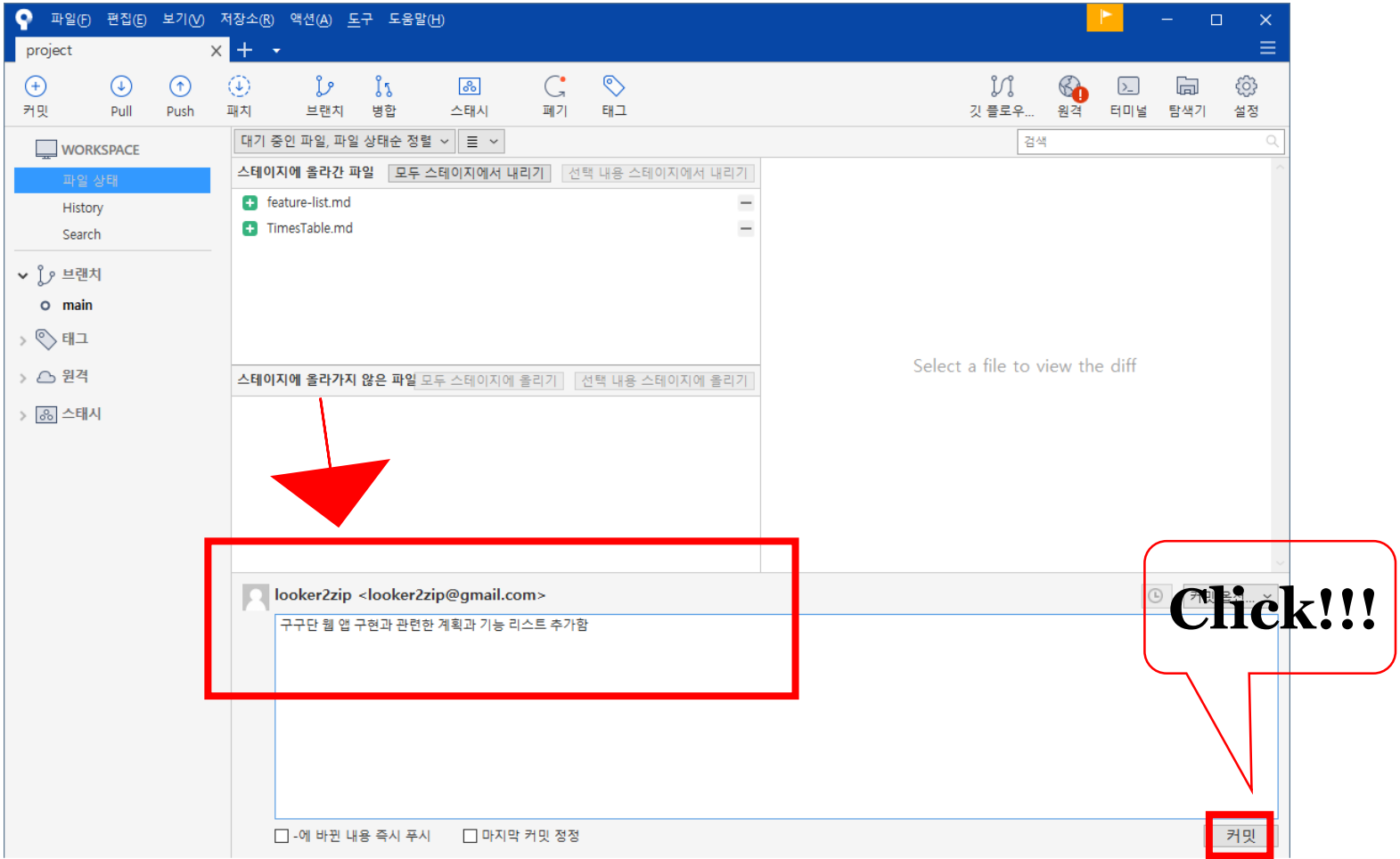
- 나누어진 스테이지 뷰에서 feature-list.md와 TimesTable.md [+] 아이콘을 클릭



3. 소스트리 기본 수행

커밋

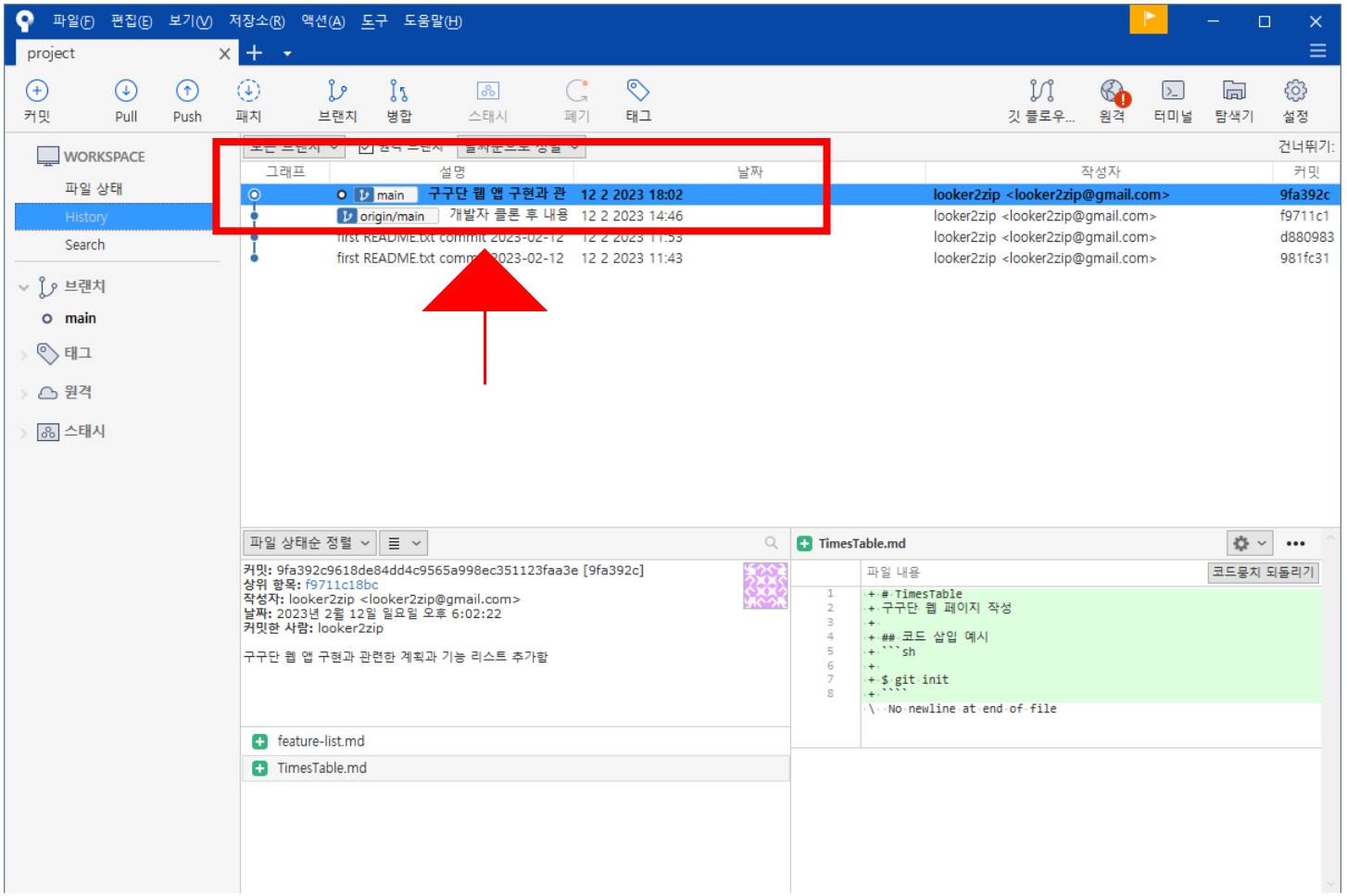
- 스테이지에 올라간 feature-list.md와 TimesTable.md 확인 후, 커밋 메시지 적음. 그리고 커밋 버튼 클릭



3. 소스트리 기본 수행

커밋

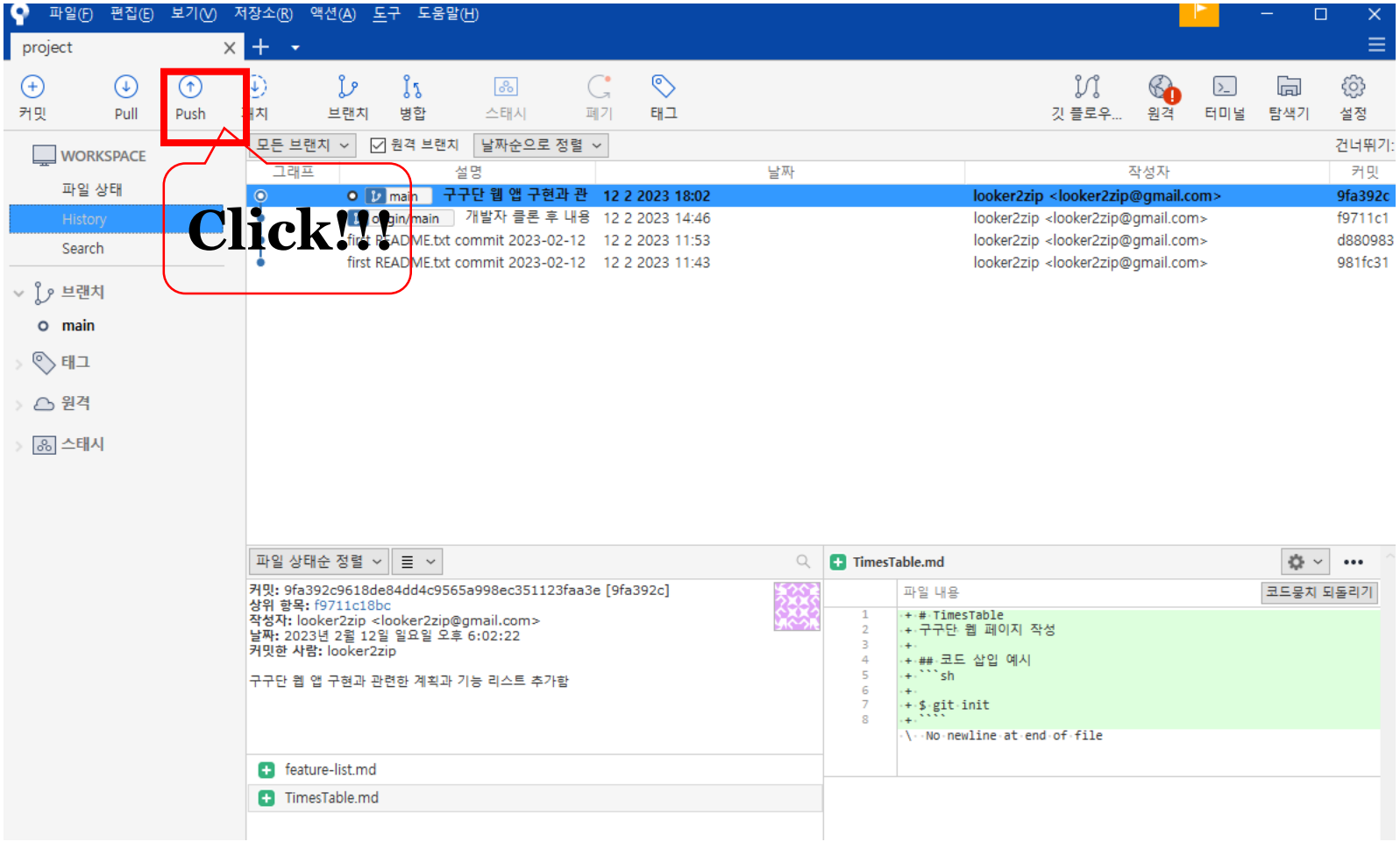
- **History 탭 > 커밋 그래프에서 커밋 내용 확인**



3. 소스트리 기본 수행

커밋을 원격저장소에 푸시하기

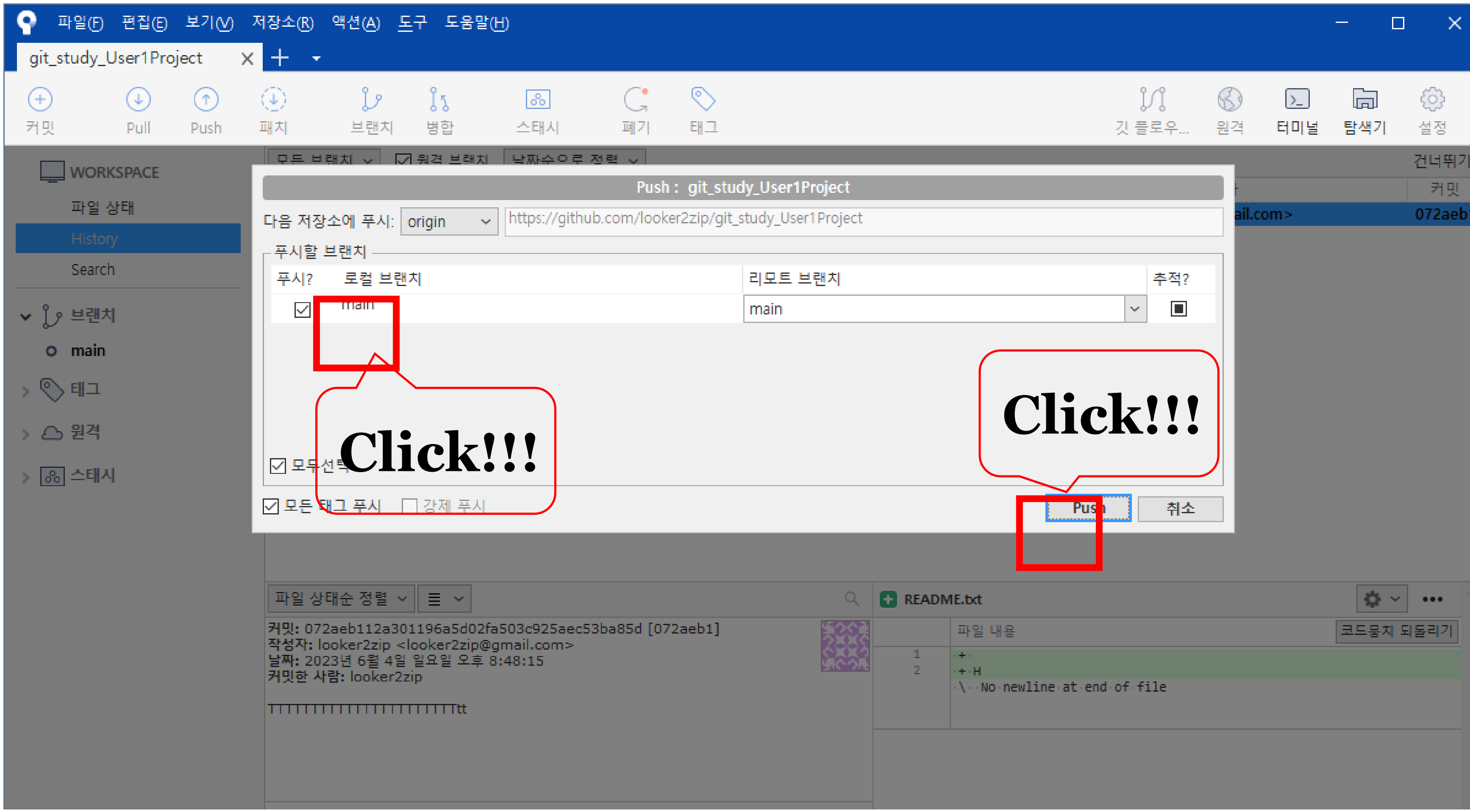
- **Push 버튼 클릭**



3. 소스트리 기본 수행

커밋을 원격저장소에 푸시하기

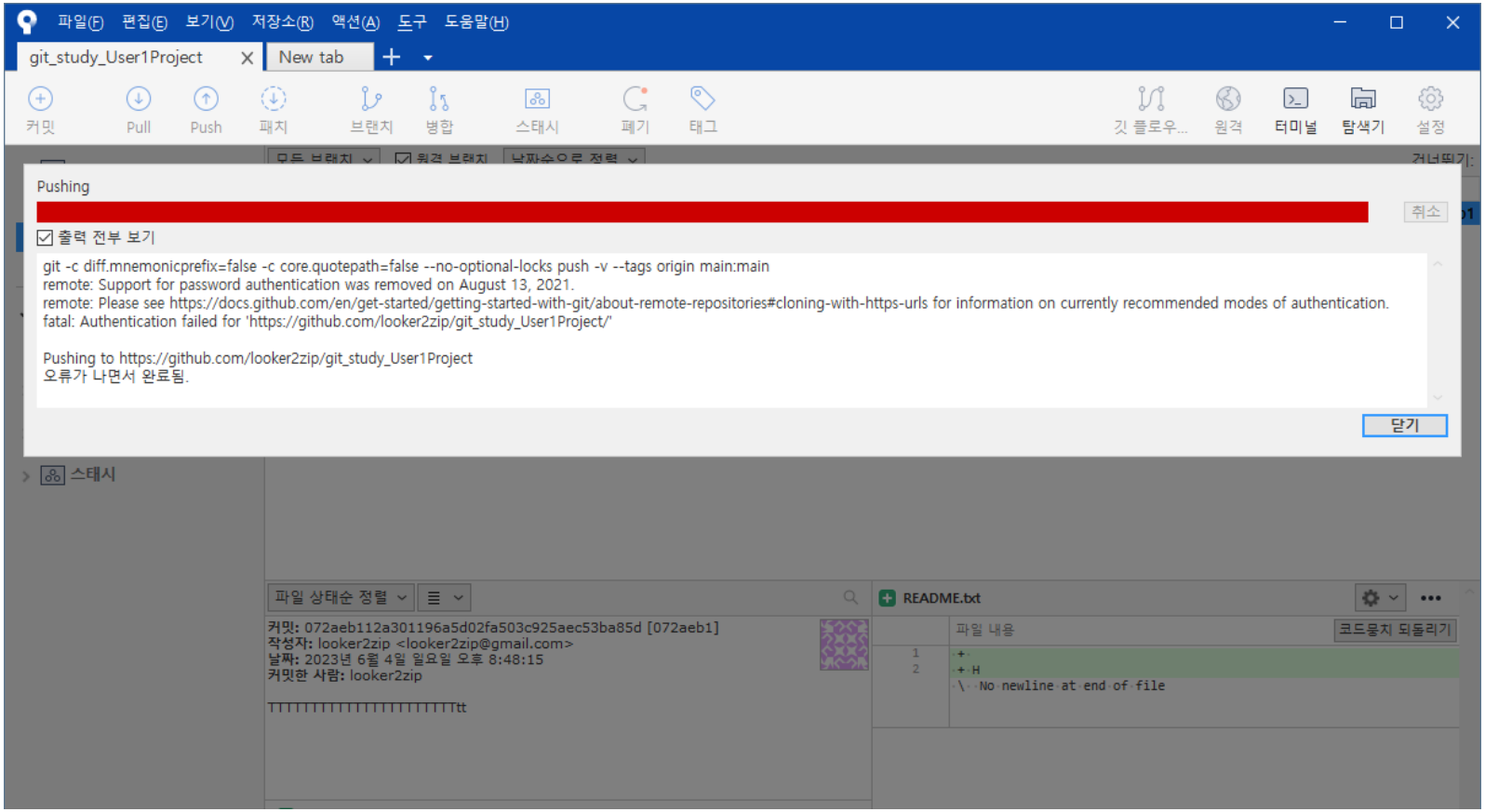
- **Main 열 체크 후, Push 버튼 클릭**



3. 소스트리 기본 수행

커밋을 원격저장소에 푸시하기

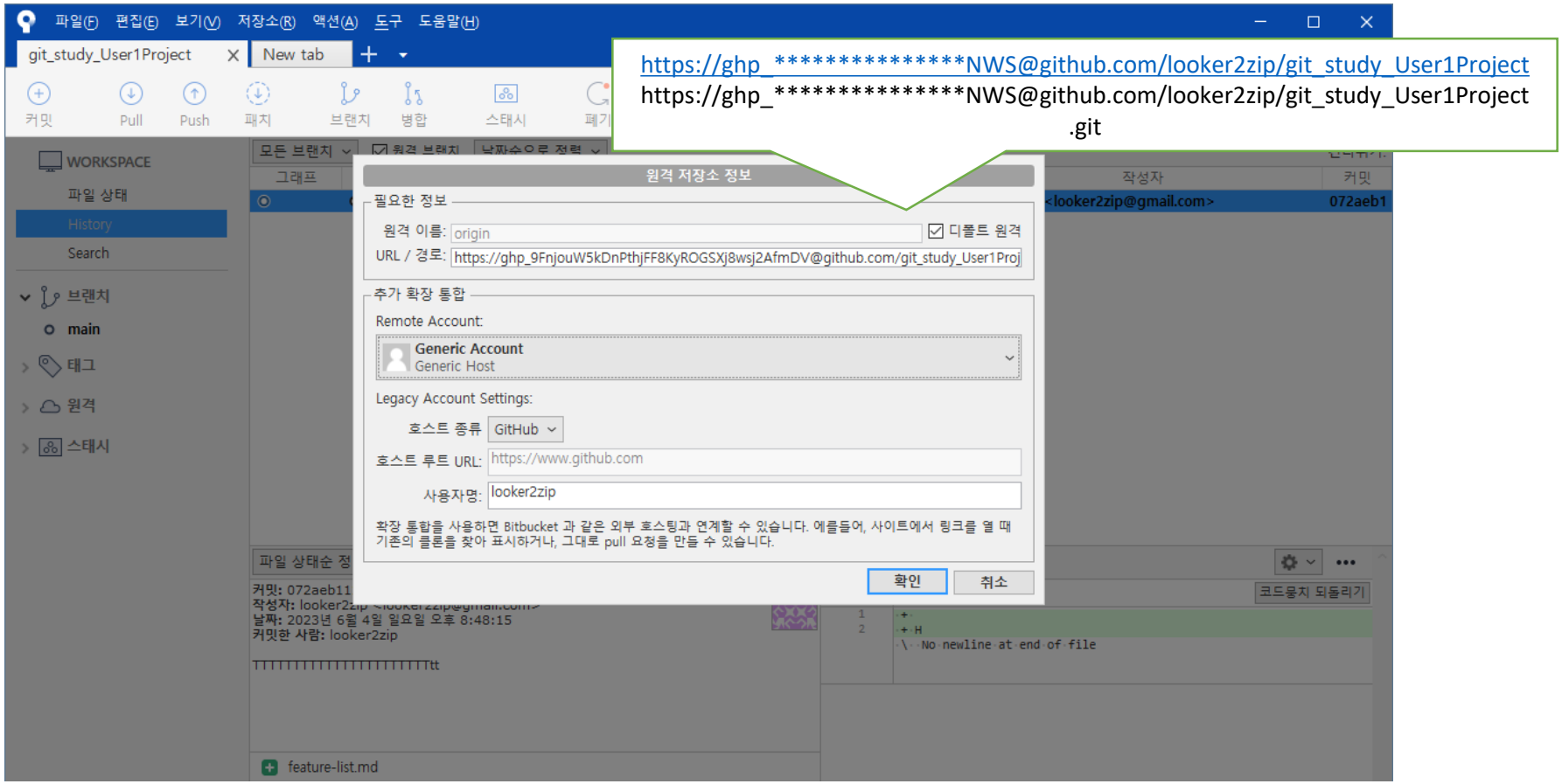
- **Main 옆 체크 후, Push 버튼 클릭**



3. 소스트리 기본 수행

커밋을 원격저장소에 푸시하기

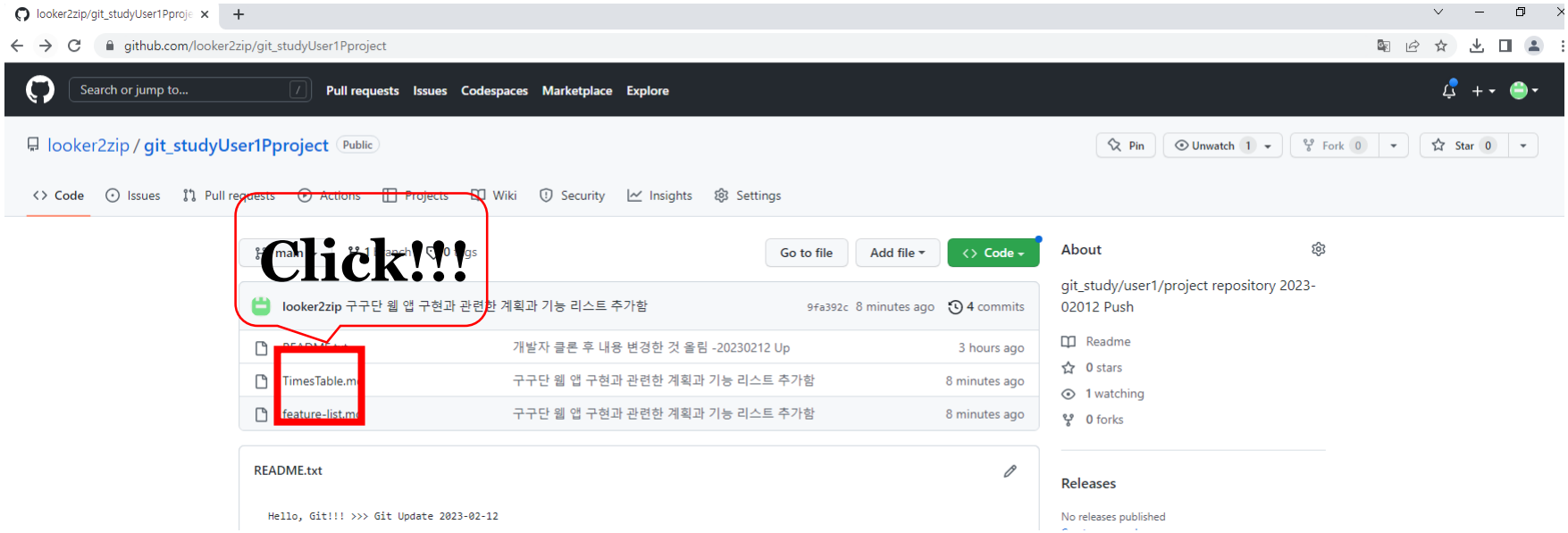
- **Main 옆 체크 후, Push 버튼 클릭**



3. 소스트리 기본 수행

원격저장소 확인

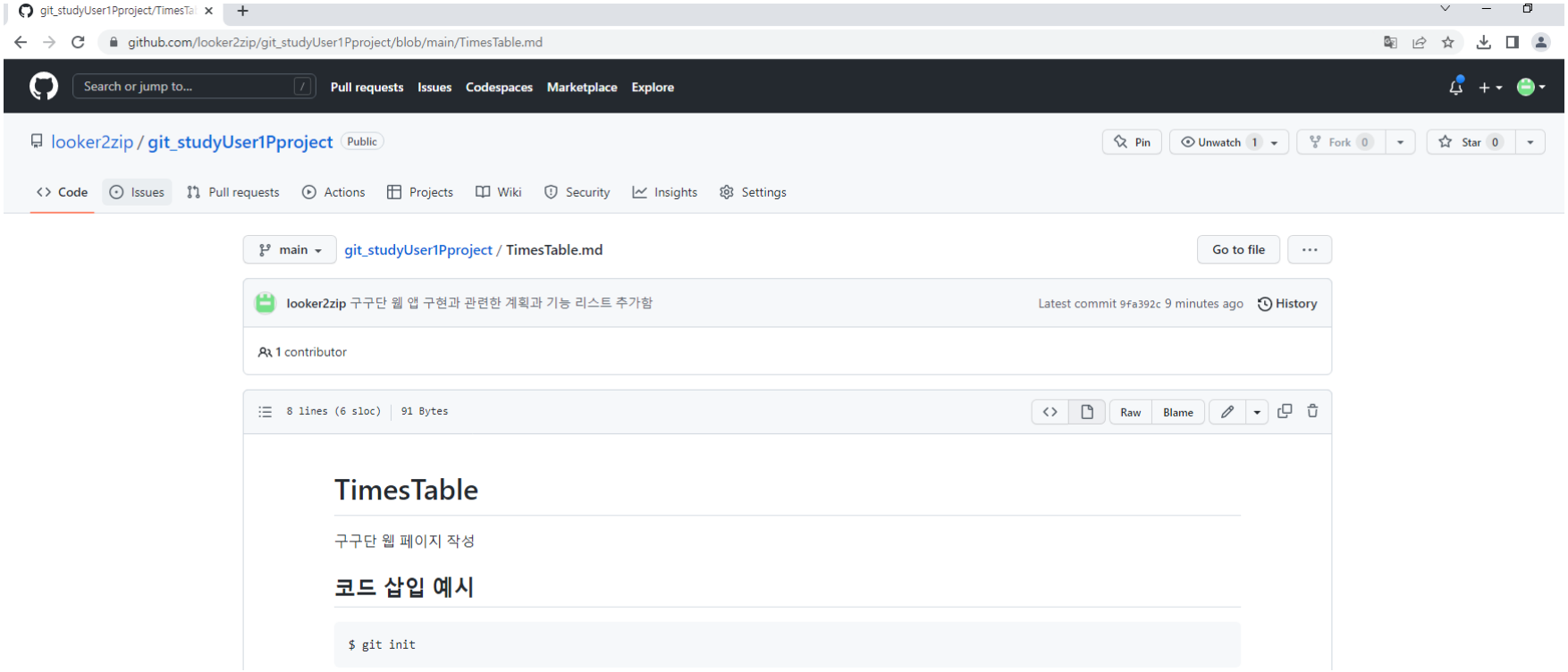
- 원격 저장소에서 Push 된 내용을 확인



3. 소스트리 기본 수행

원격저장소 확인

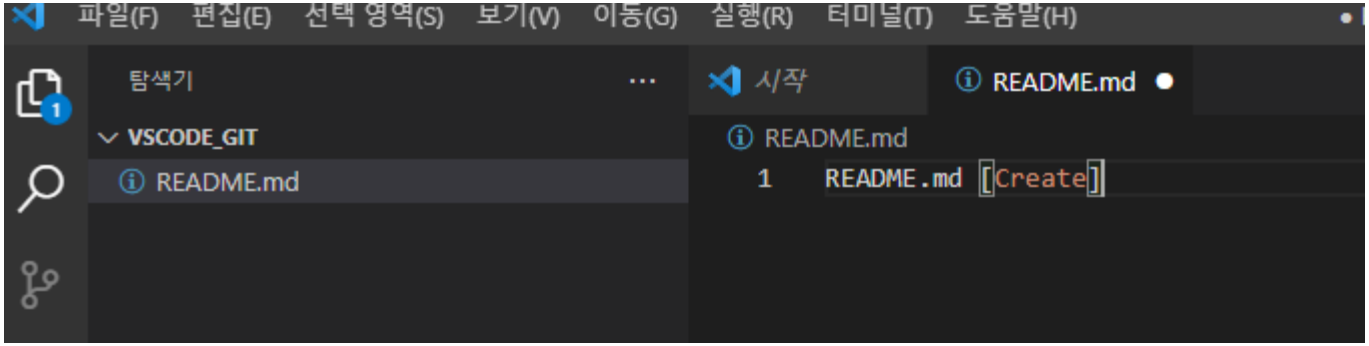
- 원격 저장소에서 Push 된 내용을 확인



4. VSCode Git 연동

1. 새 프로젝트 생성 후 실습

- **README.md 파일이 1개 포함된 폴더를 한개 생성**

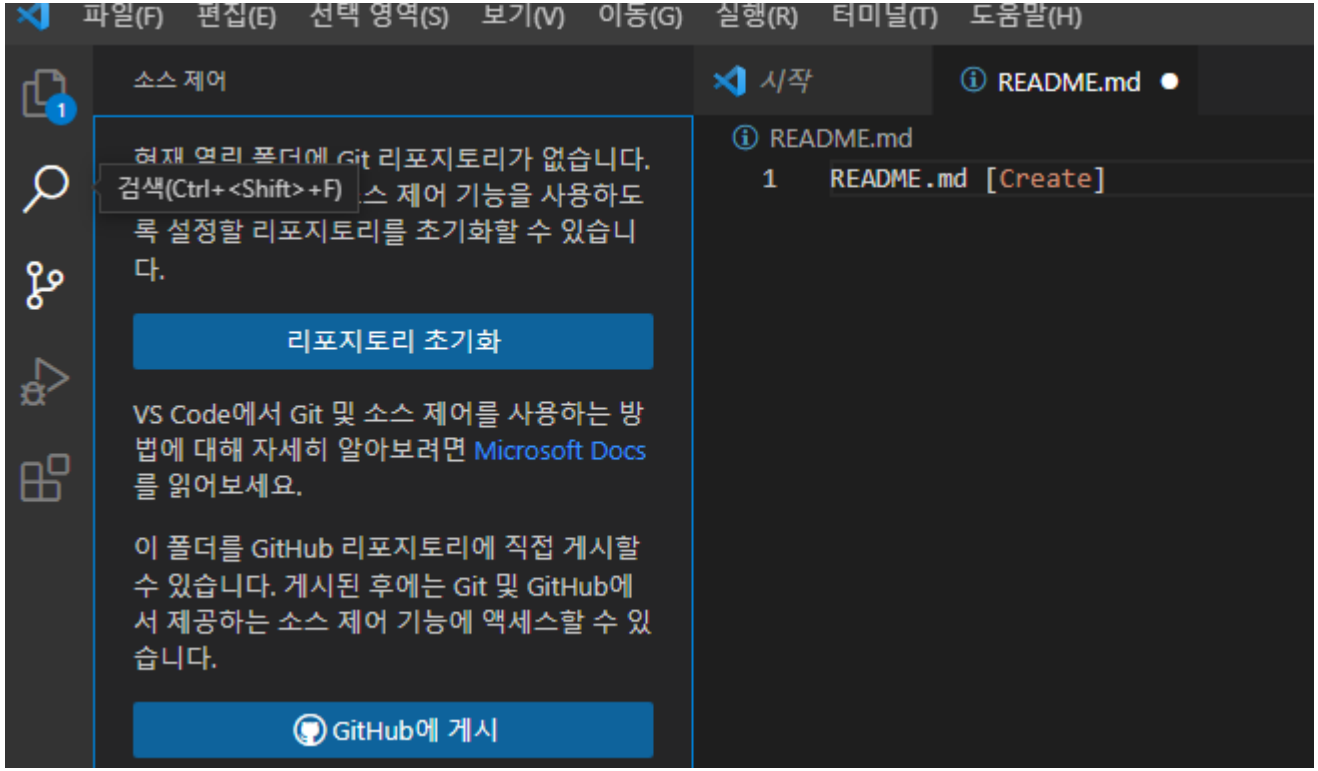



```
C:\DEV\gitworkspaces\vscode_git>git --version
git version 2.40.1.windows.1


C:\DEV\gitworkspaces\vscode_git>dir/w/p
 C 드라이브의 볼륨에는 이름이 없습니다.
 볼륨 일련 번호: BED0-C858

 C:\DEV\gitworkspaces\vscode_git 디렉터리

[.]         [..]        README.md
               1개 파일                  18 바이트
               2개 디렉터리  8,597,274,624 바이트 남음

C:\DEV\gitworkspaces\vscode_git>
```

4. VSCode Git 연동

1. 프로젝트 생성

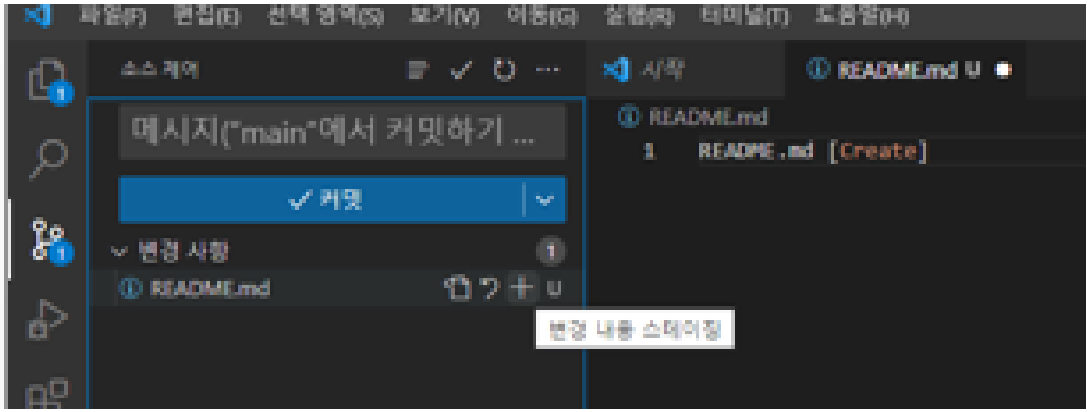
- VS Code에는 기본적으로 좌측 사이드 메뉴에, 따로 Extension을 설치하지 않아도 Git 메뉴가 있다. 이 메뉴를 클릭하여, "Initialize Repository"를 클릭. git init 커맨드와 동일한 역할을 하는 버튼.



4. VSCode Git 연동

1. 프로젝트 생성

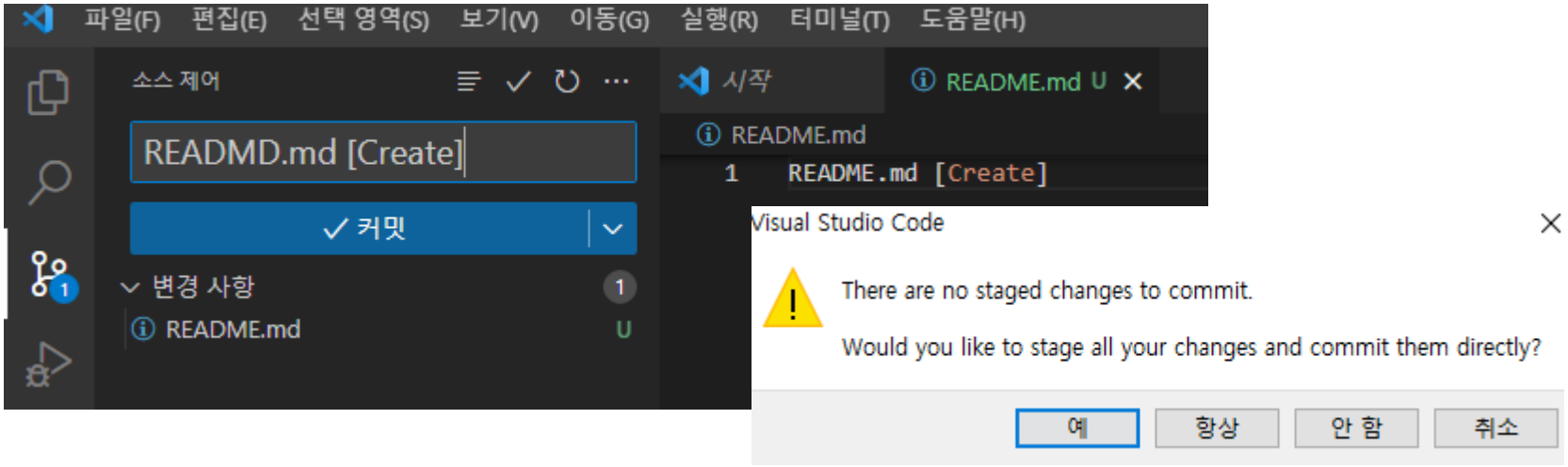
- Git init이 완료 되면, 아래와 같이 Source Control 메뉴에서 내가 생성하거나 수정한 파일이 Changes에 노출되는 것을 확인할 수 있습. 그러면 파일 하나하나 우측 + 버튼을 클릭하여 Stage 상태로 변경을 하거나 혹은 Chages 우측의 +(Stage All Changes) 버튼을 통해 모든 파일을 Stage 상태로 변경. 이 것은 git add 커맨드와 동일한 작업.



4. VSCode Git 연동

1. 프로젝트 생성

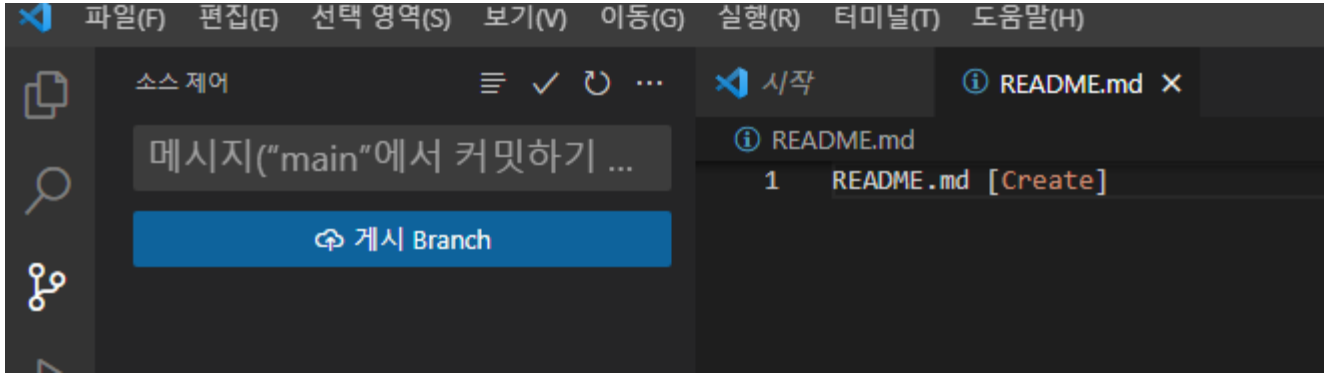
- Git add가 완료되면, 아래와 같이 해당 파일이 Staged Changes로 이동을 한 것을 확인할 수 있다. Source Control 우측에 브이 모양의 Commit 버튼을 클릭하면 Commit Message를 입력할 수 있는 창이 오픈이 된다. 여기에 메시지를 입력하고 엔터를 누르면 정상적으로 커밋이 완료된다.



4. VSCode Git 연동

1. 프로젝트 생성

- Git add가 완료되면, 아래와 같이 해당 파일이 Staged Changes로 이동을 한 것을 확인할 수 있다. Source Control 우측에 브이 모양의 Commit 버튼을 클릭하면 Commit Message를 입력할 수 있는 창이 오픈이 된다. 여기에 메시지를 입력하고 엔터를 누르면 정상적으로 커밋이 완료된다.

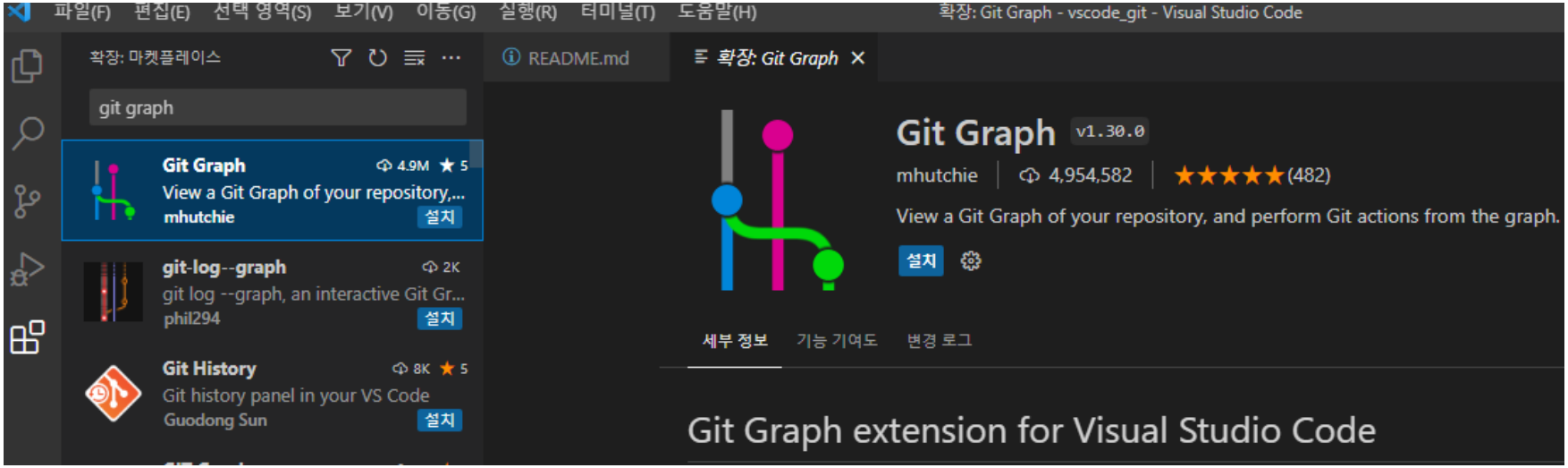


C:\DEV\gitworkspaces\vscode_git>git log --oneline --all --graph
* 3131588 (HEAD -> main) READMD.md [Create]

4. VSCode Git 연동

1. 프로젝트 생성

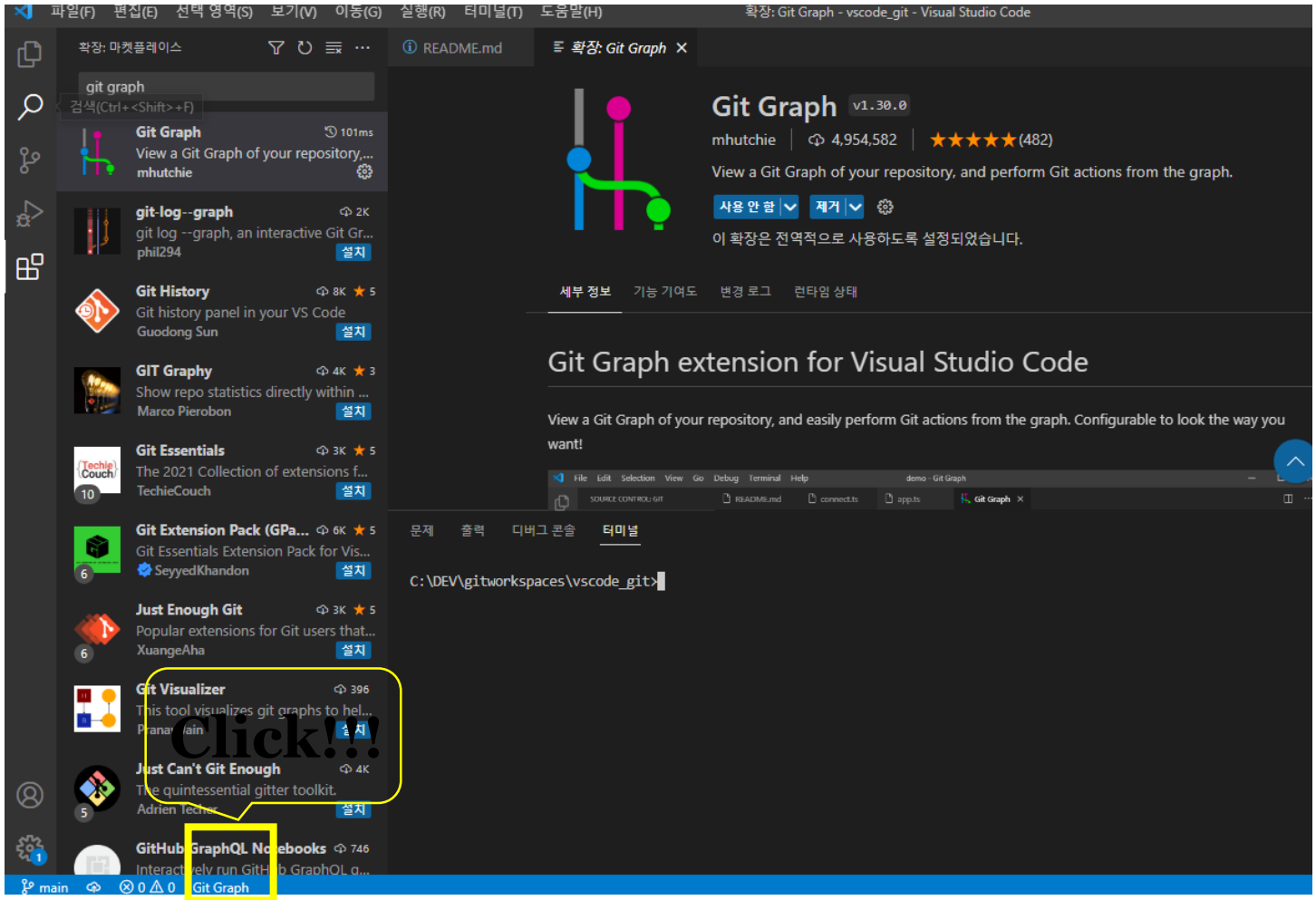
- Git Graph 설치



4. VSCode Git 연동

1. 프로젝트 생성

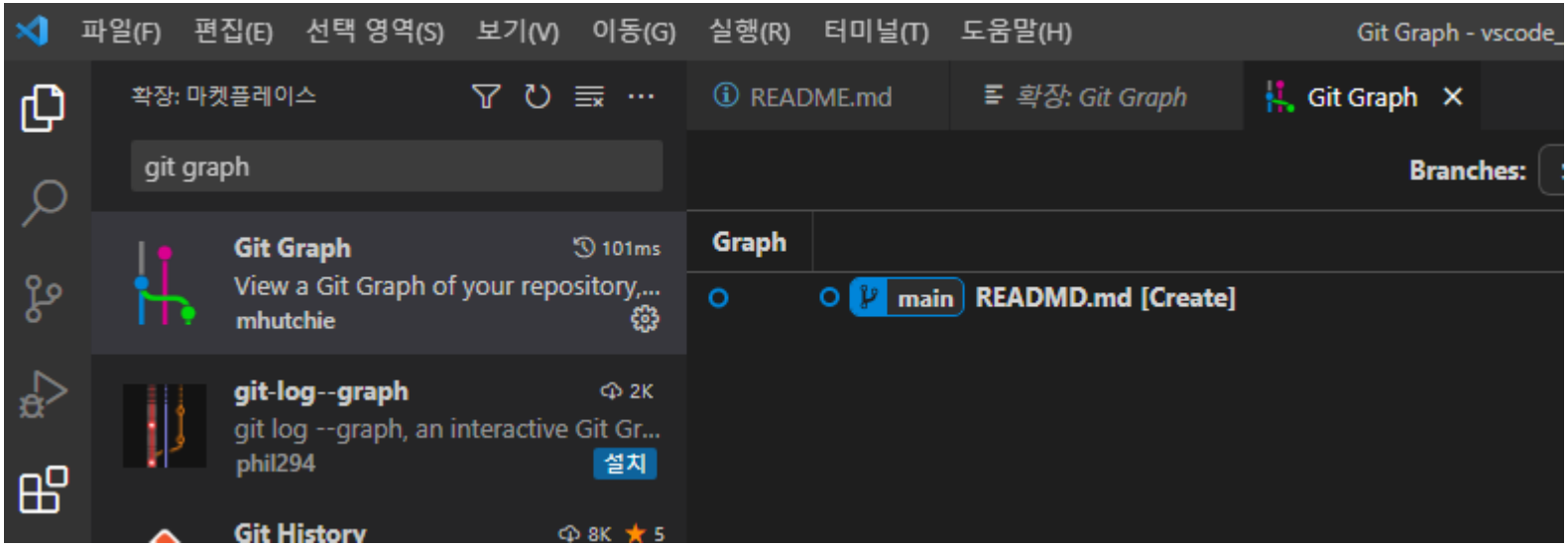
- Git Graph 클릭

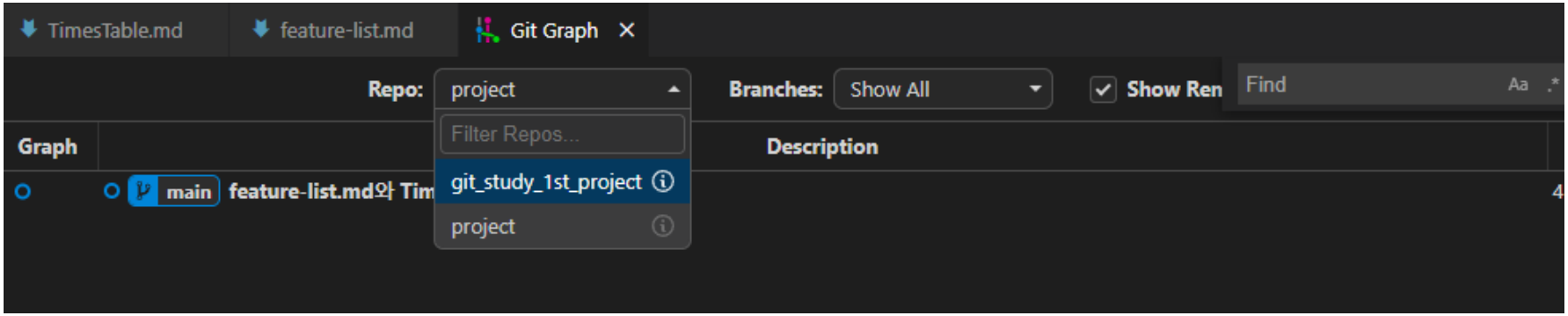


4. VSCode Git 연동

1. 프로젝트 생성

- Git Graph 클릭

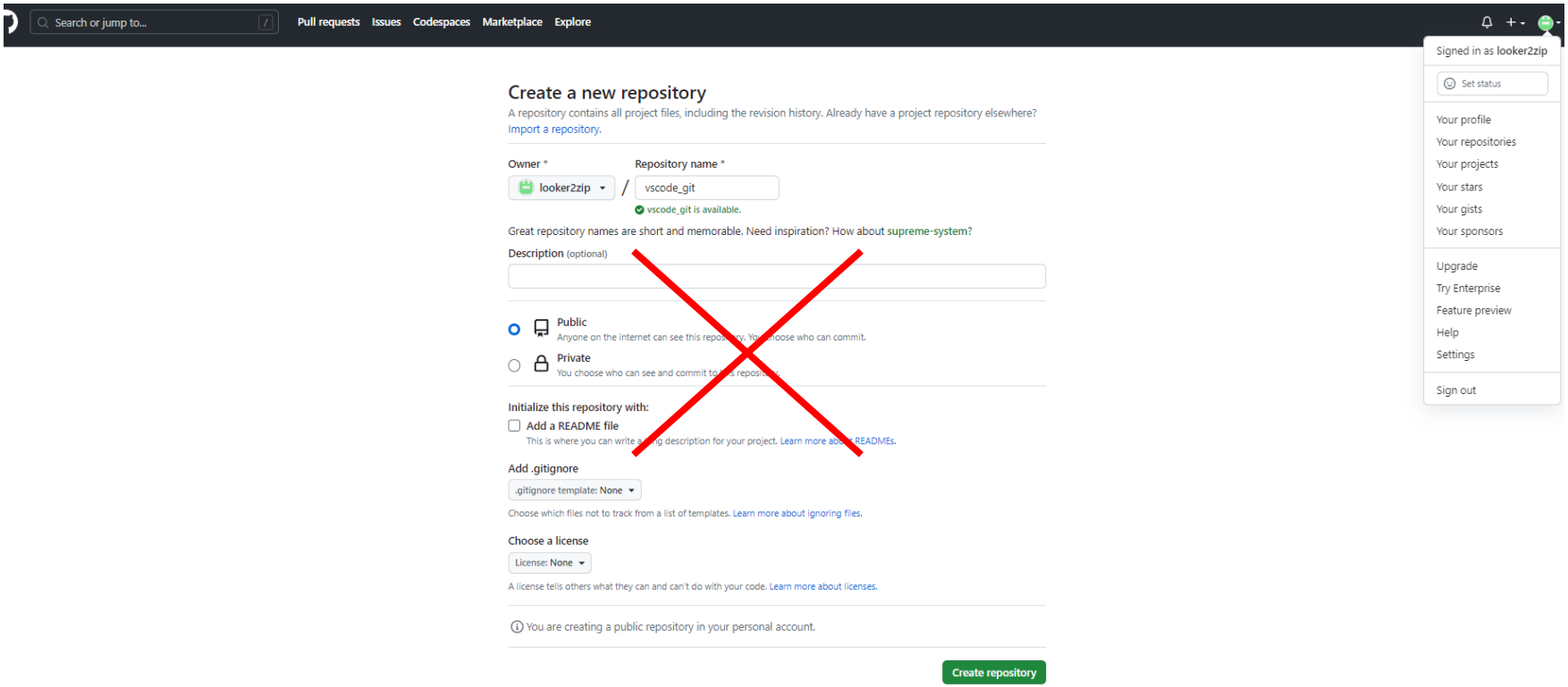




4. VSCode Git 연동

2. Push

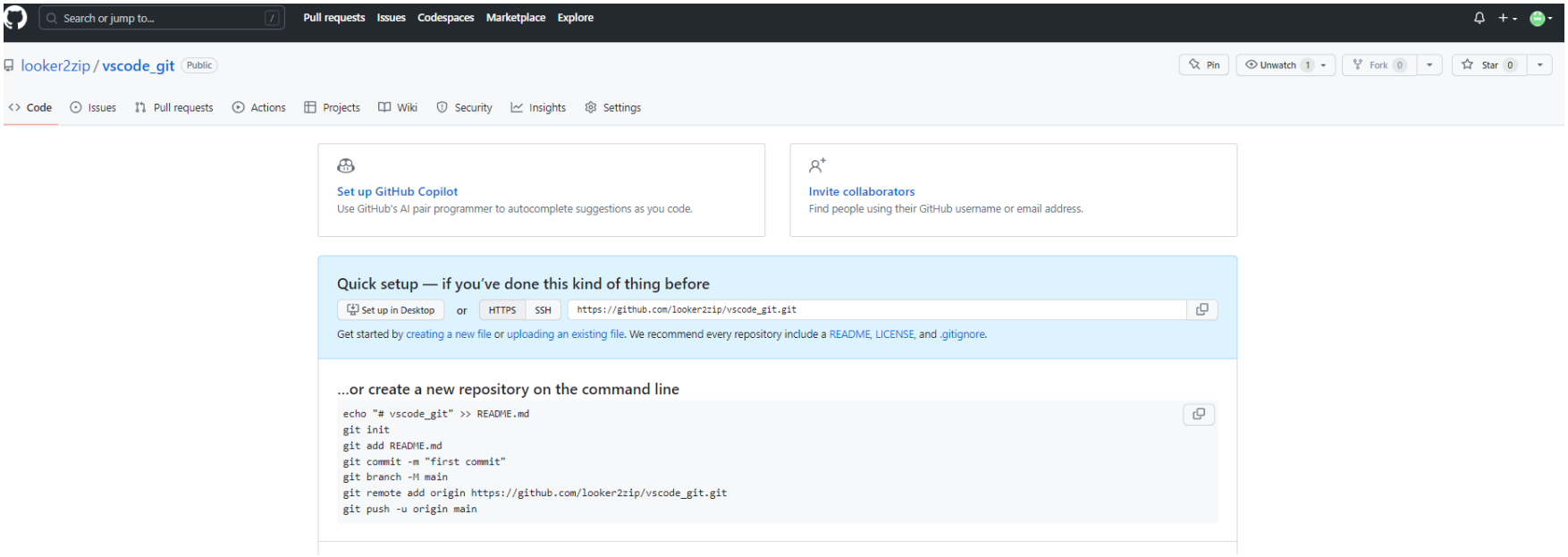
- 원격 저장소 생성



4. VSCode Git 연동

2. Push

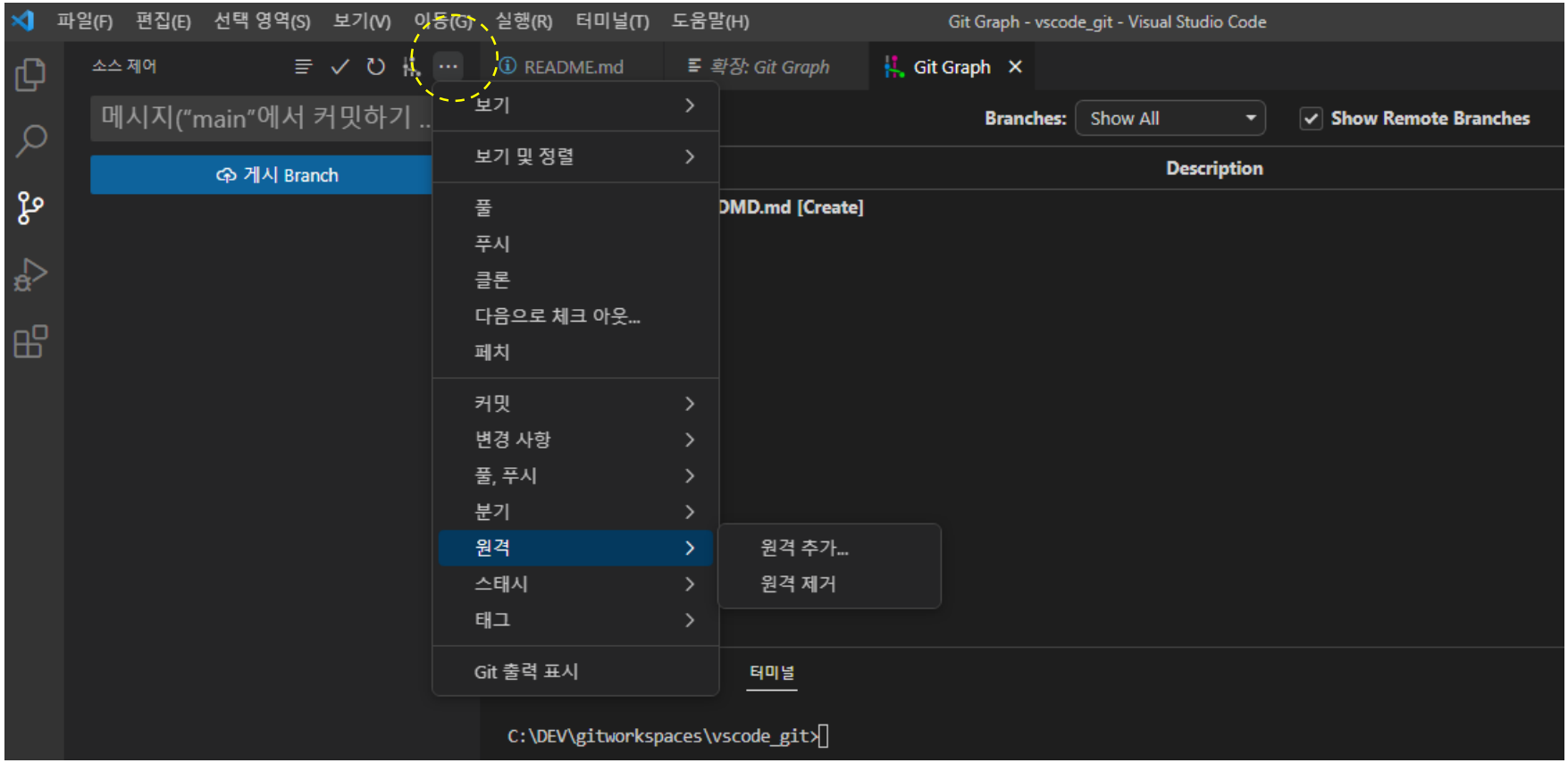
- 원격 저장소 생성



4. VSCode Git 연동

2. Push

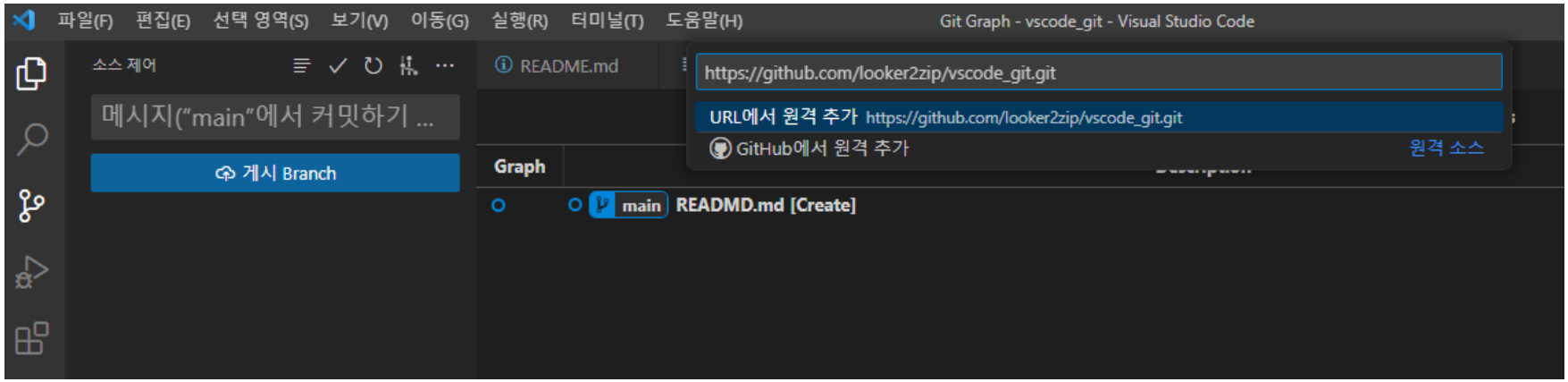
- 원격 저장소 등록 > 소스 제어 > 원격 > 원격 추가



4. VSCode Git 연동

2. Push

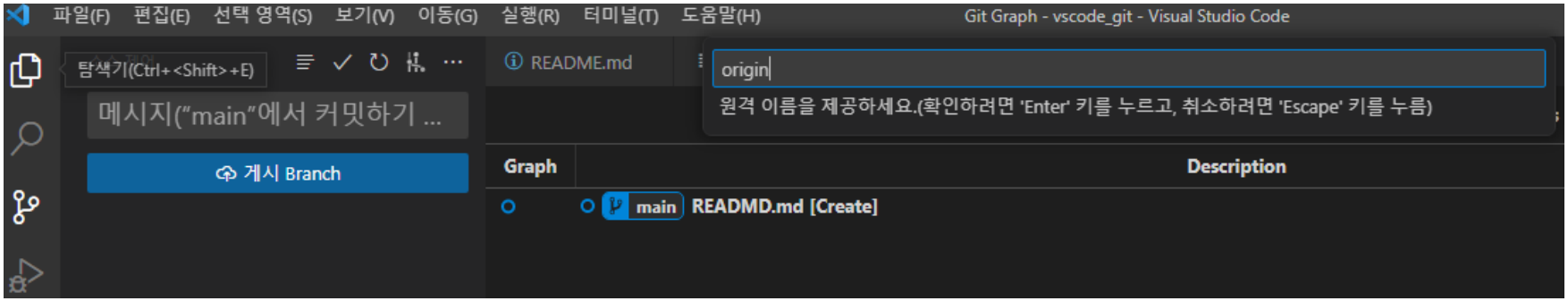
- 원격 저장소 등록
- 주소 입력



4. VSCode Git 연동

2. Push

- 원격 저장소 등록
- 주소 별칭 등록



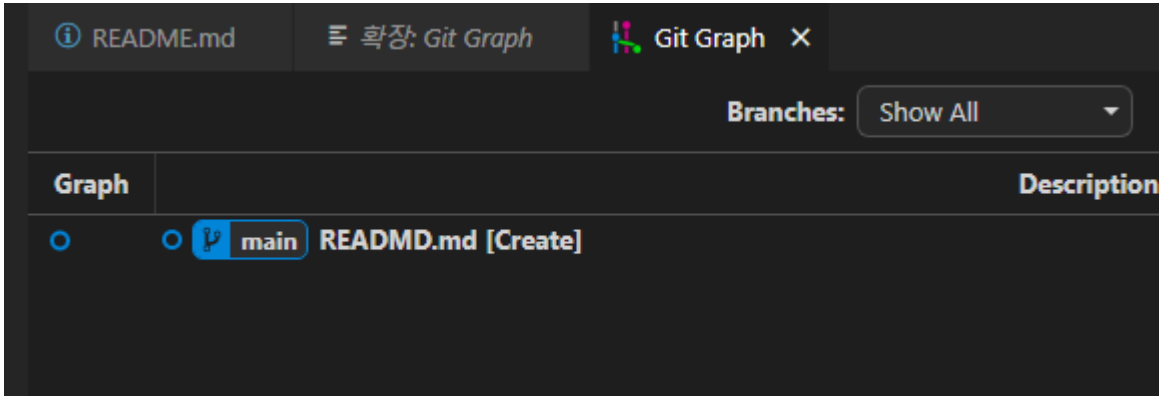
4. VSCode Git 연동

2. Push

- 로그 확인

```
C:\DEV\gitworkspaces\vscode_git>git log --oneline --all --graph
* 3131588 (HEAD -> main) READMD.md [Create]

C:\DEV\gitworkspaces\vscode_git>
```

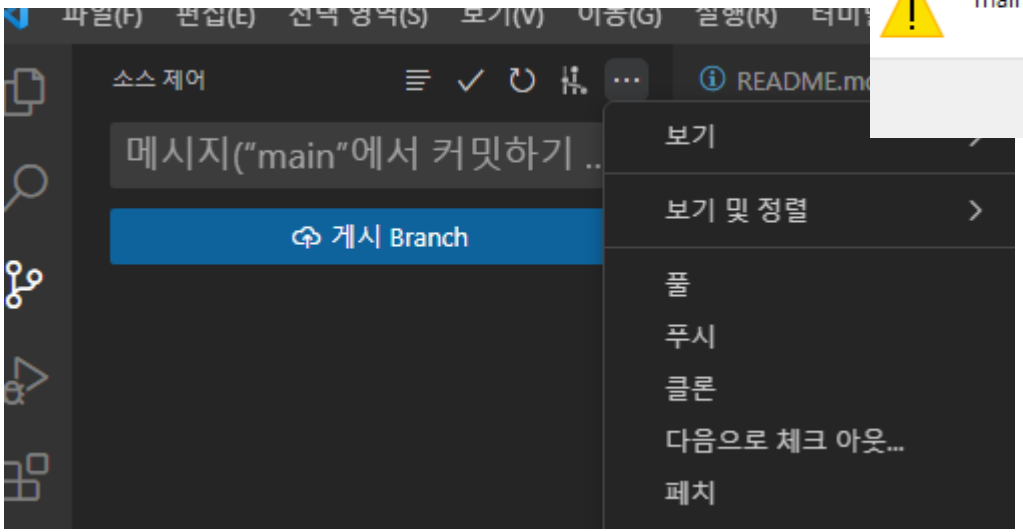
HEAD가 main 을 가리키고 있음

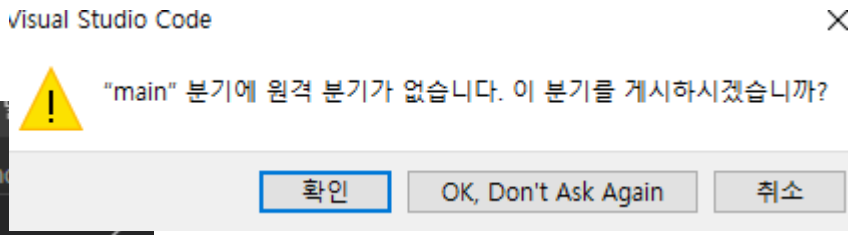


4. VSCode Git 연동

2. Push

- 로그 확인





게시 Branch 혹은 푸시를 클릭

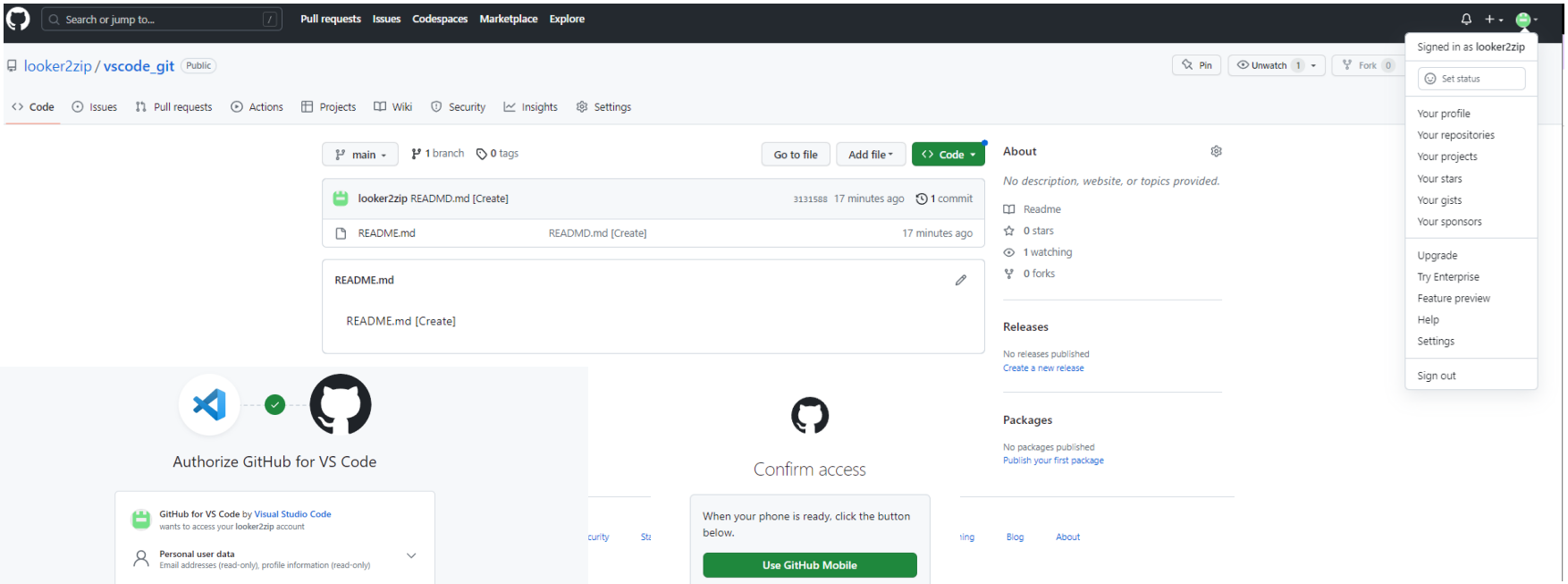
C:\DEV\gitworkspaces\vscode_git>git log --oneline --all --graph
* 3131588 (HEAD -> main, origin/main) READMD.md [Create]

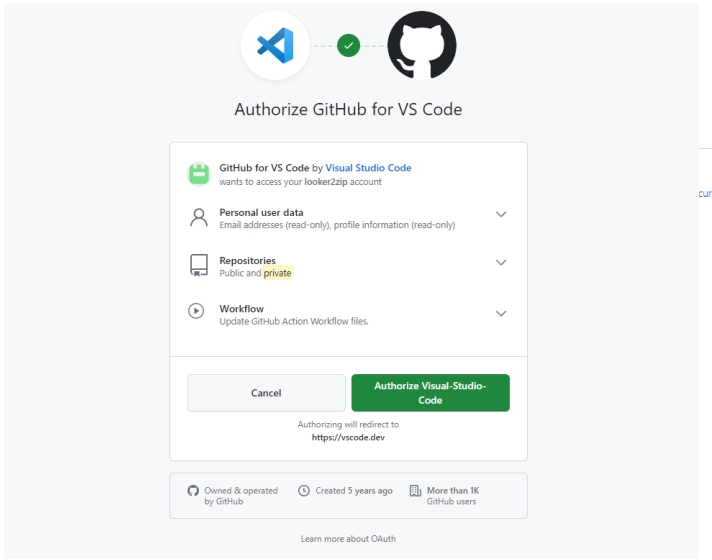
C:\DEV\gitworkspaces\vscode_git>

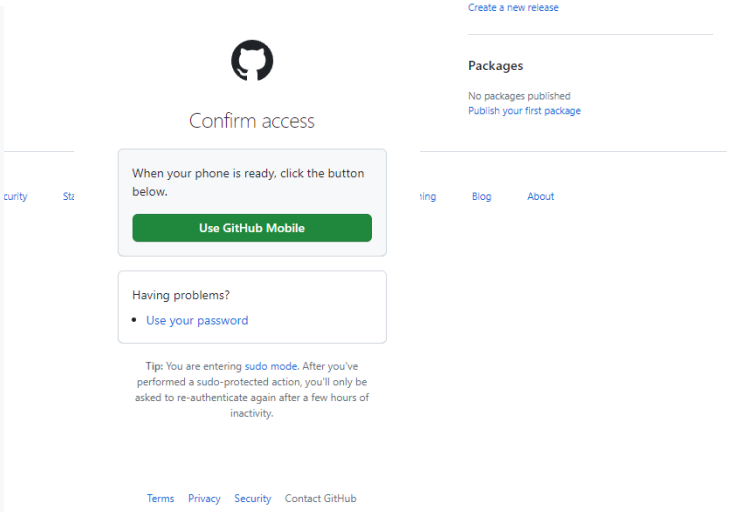
4. VSCode Git 연동

2. Push

- Git Hub 확인





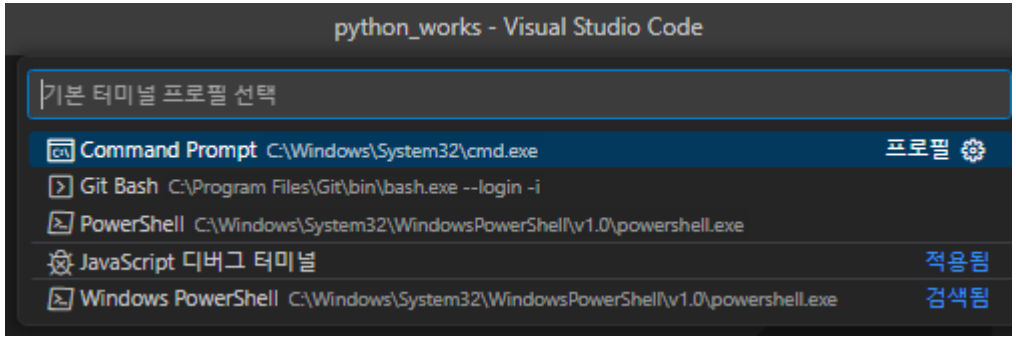


4. VSCode Git 연동

3. Git Bash 터미널 열기

보기 > 명령 팔레트

- Select Default Profile



4. VSCode Git 연동

3. Git Bash 터미널 열기

- 터미널 창에서 Git Bash 선택