

깃허브 (GIT HUB) 사용법



소프트웨어 소스 형상관리 툴의 대명사

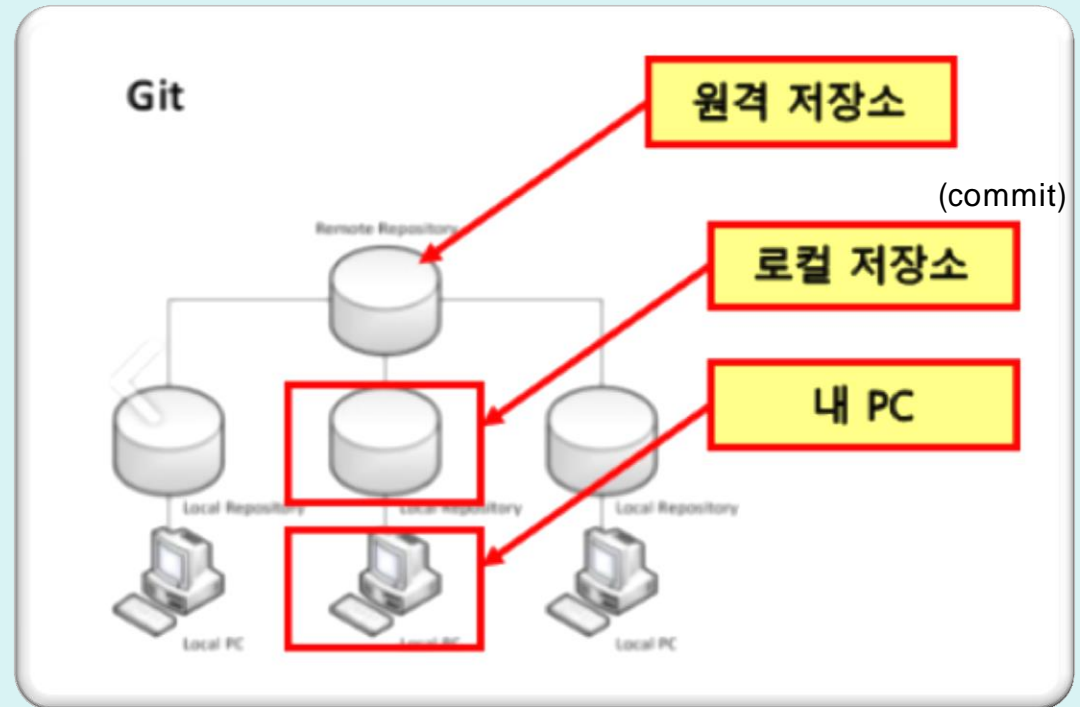
2020.08.19

TOM BRACE PARKER

깃허브의 관리 구조



원격 저장소에 공유 소스를 올려 놓고 각 팀원들의 깃허브 계정에 연결을 허용하여 로컬 저장소에 저장하고 다시 원격 저장소로 소스를 업데이트하여 하나의 소스로 합치는 작업을 버전관리를 하며 효과적으로 관리 할 수 있도록 만들어진 버전관리 시스템이다.



1. 가입하기



- 깃허브 사이트에 접속하여 오른쪽 위 sign in 버튼을 클릭하여 회원가입을 한다.

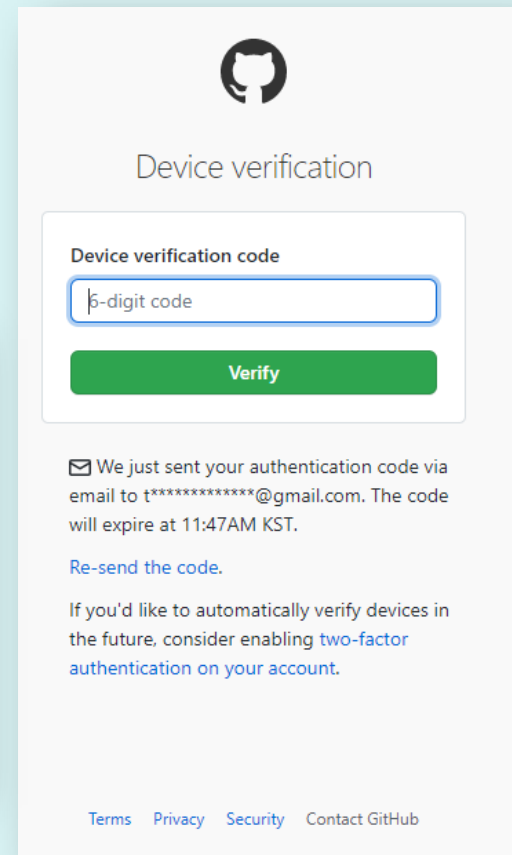
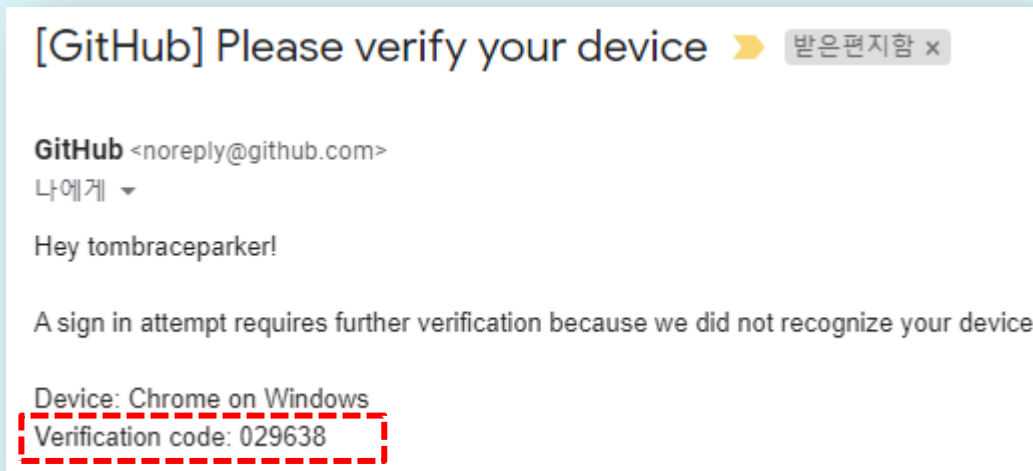
The screenshot shows the GitHub 'Create your account' page. At the top is a dark navigation bar with the GitHub logo, links for 'Why GitHub?', 'Team', 'Enterprise', 'Explore', 'Marketplace', and 'Pricing', a search bar, and a 'Sign in' button. Below the navigation bar, the text 'Join GitHub' is centered above the main heading 'Create your account'. The form contains three input fields: 'Username', 'Email address', and 'Password', each with an asterisk indicating it is required. Below the password field is a note: 'Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)'. There is a section for 'Email preferences' with a checkbox labeled 'Send me occasional product updates, announcements, and offers.' which is currently unchecked. At the bottom, there is a 'Verify your account' section with a partially visible input field.

2. 가입 후 메일 인증하기



- 회원가입 후 개인 메일로 아이디 인증 메일의 인증코드를 넣어서 가입을 완료한다.

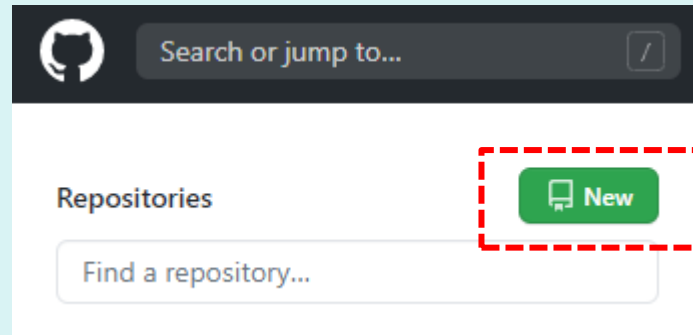
아이디로 입력한 이메일 계정으로 온 깃허브 인증코드를 확인하여 오른쪽 인증 페이지에 넣고 Verify 버튼을 클릭한다!



3. 레파지토리(Repositories) 만들기



- 가입 후 첫화면에서 나오는 간단한 질문지에 답을 한 후(질문지는 직업군이나 목적을 물어봄) 첫 페이지에서 레파지토리(Repository) 만들기를 한다.
- 레파지토리란 저장소의 역할을 하는 관리폴더 개념이라 생각할 수 있다.
- New 버튼을 클릭한다.



4. 레파지토리 입력하기



- 레파지토리 New버튼을 클릭하면 간단히 레파지토리 이름을 쓴 후 생성버튼을 클릭한다.

The screenshot shows the 'Create a new repository' form. It includes fields for 'Owner' (tombraceparker), 'Repository name' (project202008), and 'Description' (optional). The 'Repository name' field is highlighted with a red dashed box and a green checkmark. The 'Public' option is selected under the 'Description' section, also highlighted with a red dashed box. At the bottom, the 'Create repository' button is highlighted with a red dashed box. Red arrows point from the Korean text annotations to these specific elements.

Create a new repository

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

Owner *
tombraceparker

Repository name *
project202008

Great repository names are short and unique. project202008 is available. Inspiration? How about scaling-engine?

Description (optional)

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

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

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

☐ Initialize this repository with a README
This will let you immediately clone the repository to your computer.

Add .gitignore: None Add a license: None ⓘ

Create repository

적절한 이름을 영어로 씀

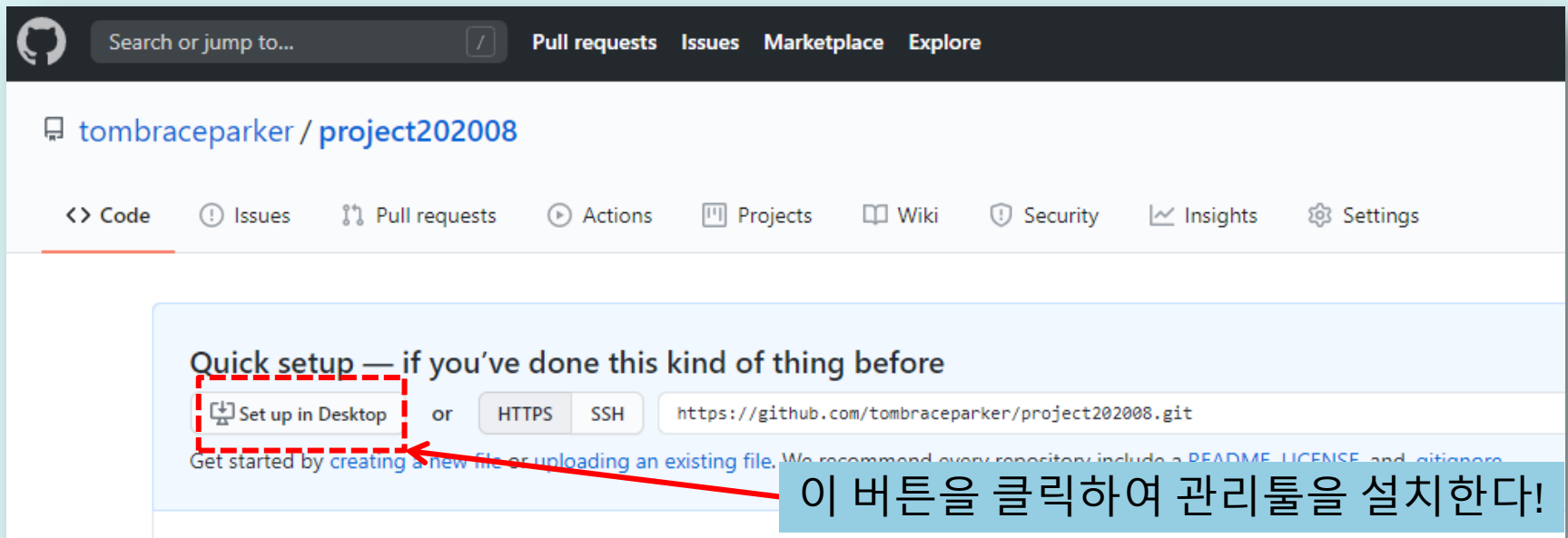
보통 공개(public)으로 설정

생성하기 버튼클릭!

5. 레파지토리 생성 후 화면



- 레파지토리 생성후 첫 화면은 소스를 보여주는 화면입니다. 그런데 아직 파일업로드 전이므로 화면에 파일이 안보입니다.
- 우선 데스크탑용 툴을 꼭 설치하세요!!!

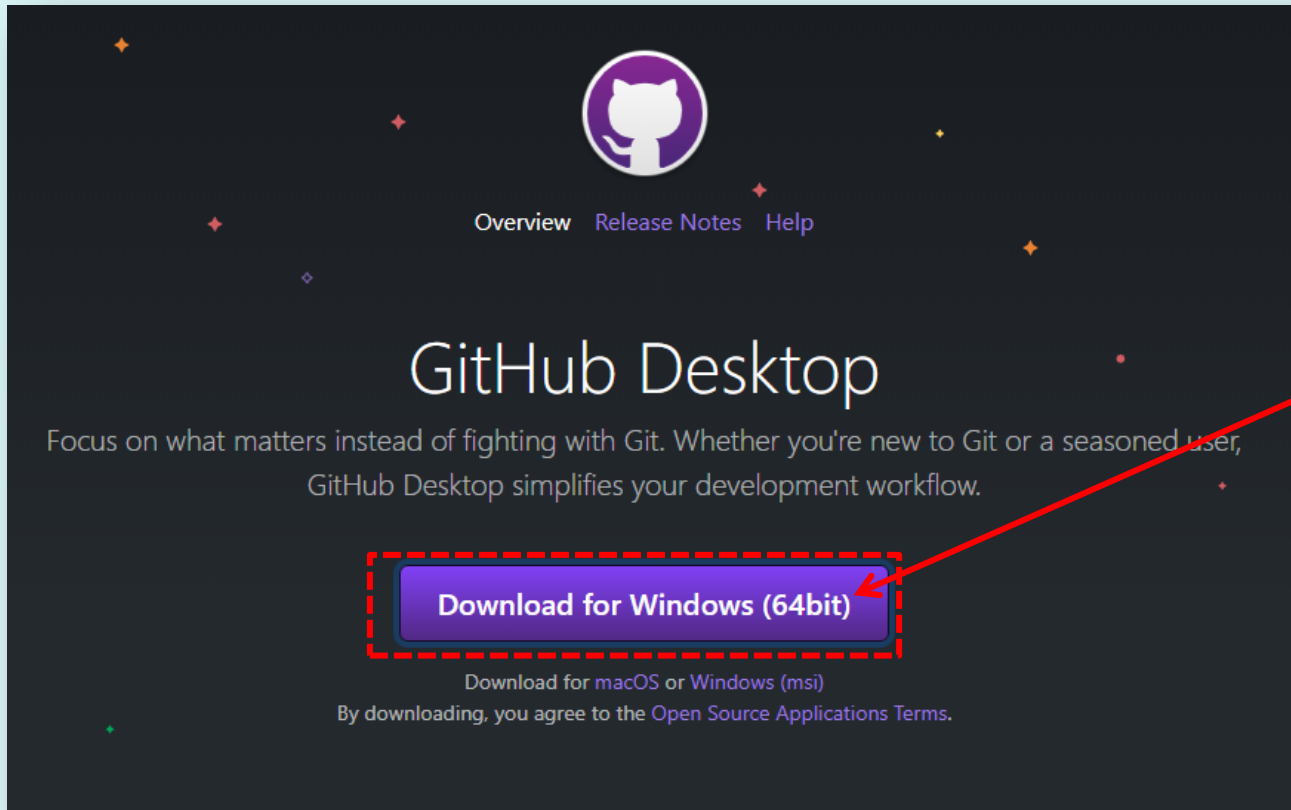


만일 버튼클릭시 반응이 없으면 <https://desktop.github.com/> 로 접속하여 설치한다.

6. 깃허브 데스크탑 툴 다운로드하기



- 정규 다운로드 사이트 주소는 아래와 같다
<https://desktop.github.com/>



이 버튼을 클릭
하여 관리툴을
설치한다!

7. 깃허브 데스크탑 설치화면



- 설치 후 아래와 같은 화면에서 로그인을 진행한다.

Make GitHub Desktop better!

Would you like to help us improve GitHub Desktop by submitting usage stats?

☒ Yes, submit periodic usage stats

Finish Cancel

로그인 마무리

Welcome to GitHub Desktop

GitHub Desktop is a seamless way to contribute to projects on GitHub and GitHub Enterprise Server. Sign in below to get started with your existing projects.

New to GitHub? [Create your free account.](#)

Sign in to GitHub.com

Sign in to GitHub Enterprise Server

Sign in to GitHub.com using your username and password

Skip this step

클릭!

8. 깃허브 데스크탑 로그인 후 서버의 레파지토리 클론으로 복제하기



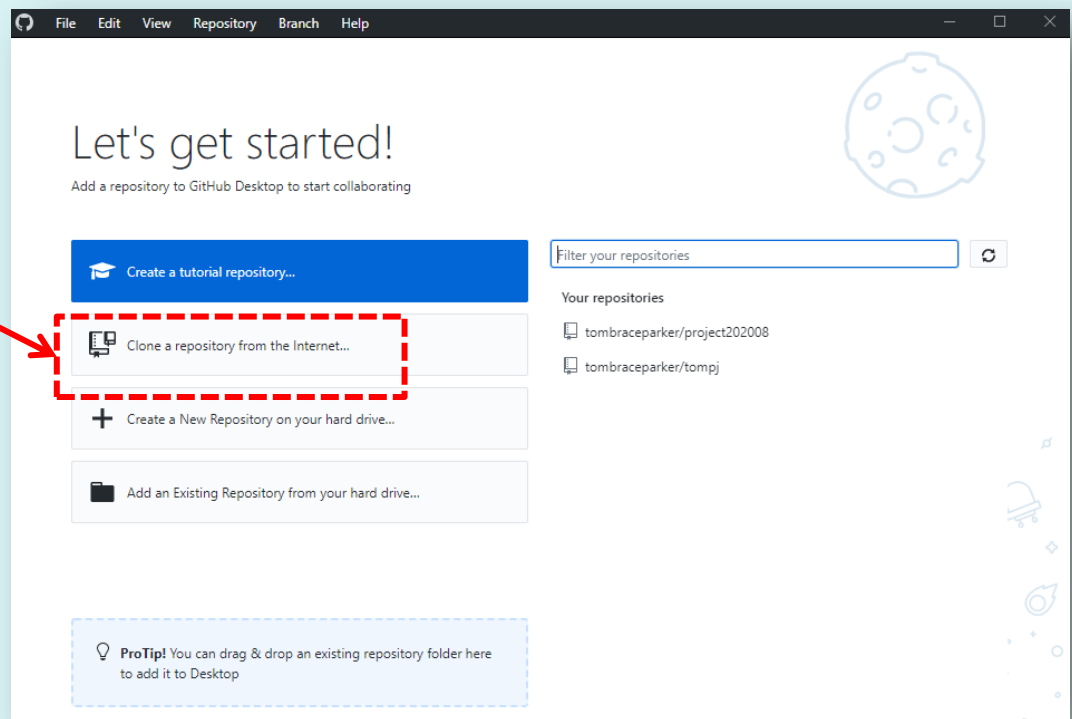
- 깃허브 데스크탑 로그인 완료 후 첫화면은 아래와 같다. 여기서 두번째 버튼이 인터넷을 통한 레파지토리 복제하기 버튼을 클릭한다.

new repository

Clone a repository from
the internet...
버튼을 클릭!

file 가

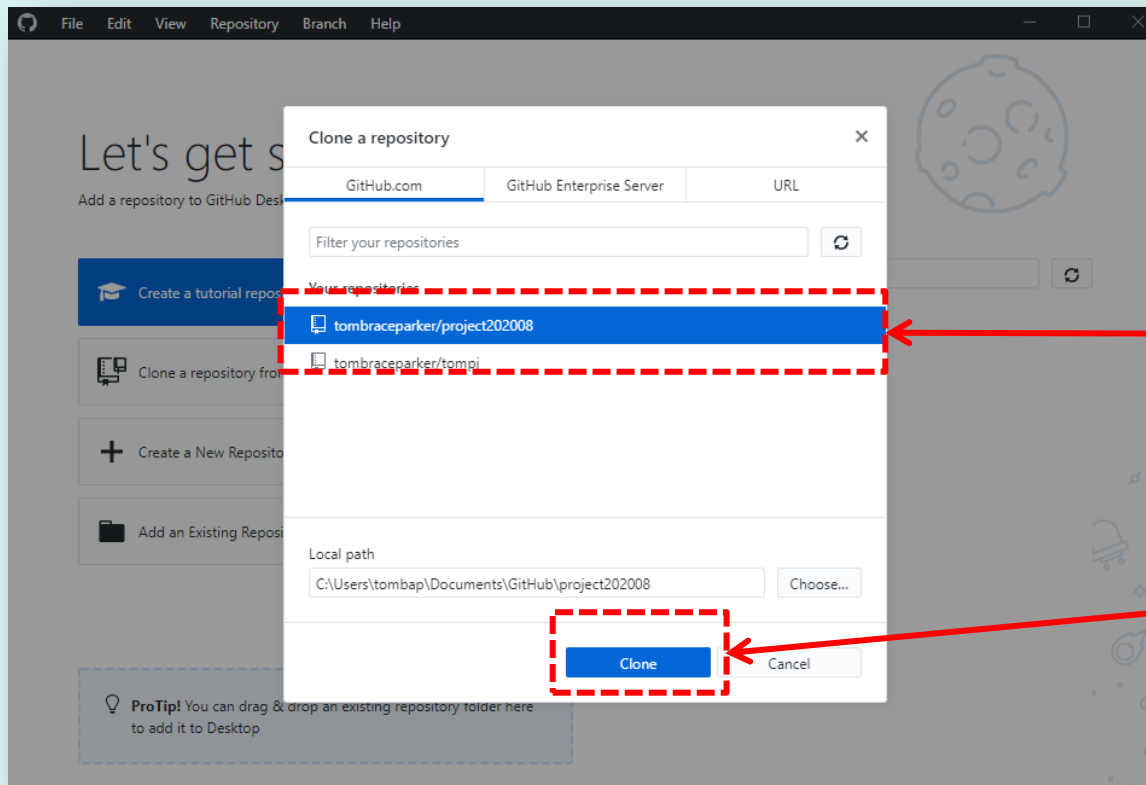
clone repository



9. 레파지토리 복제버튼 클릭 시 선택



- 레파지토리를 인터넷으로 연결하여 복제시 이미 만들어진 레파지토리 리스트에서 복제할 대상을 선택한다.



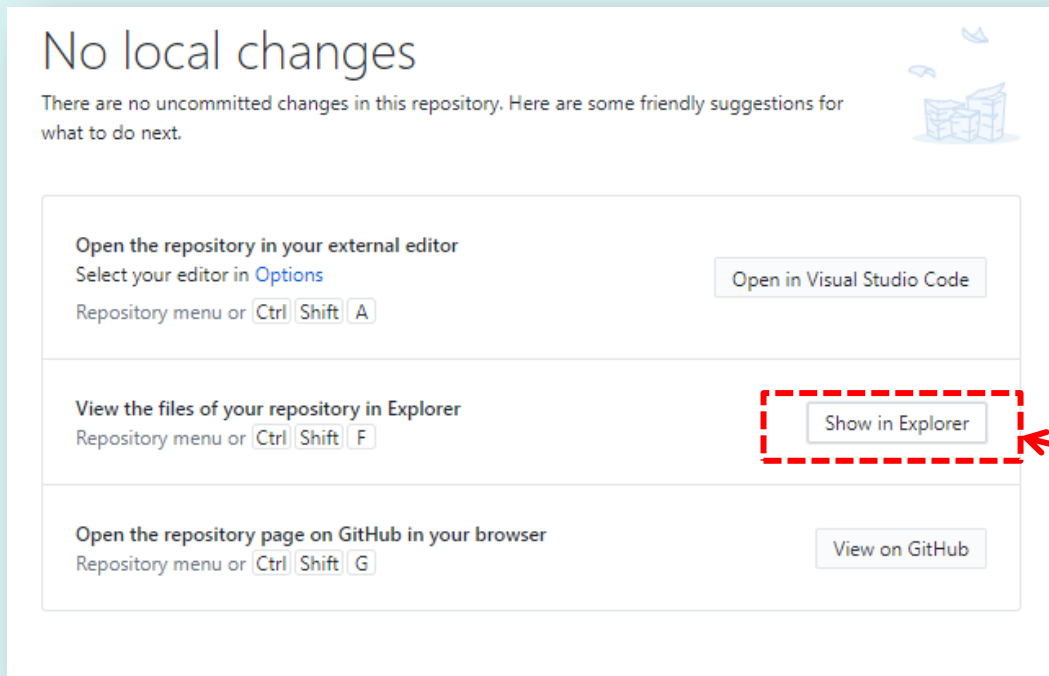
이미 만들어진 레파지토리 중 복제대상을 선택하여 클릭!

복제(Clone)버튼 클릭!

10. 복제완료 후 화면



- 복제 후 파일올리기를 위해 익스플로러 화면 열기를 선택한다. 그러면 탐색기를 통해 파일 위치를 보여준다.

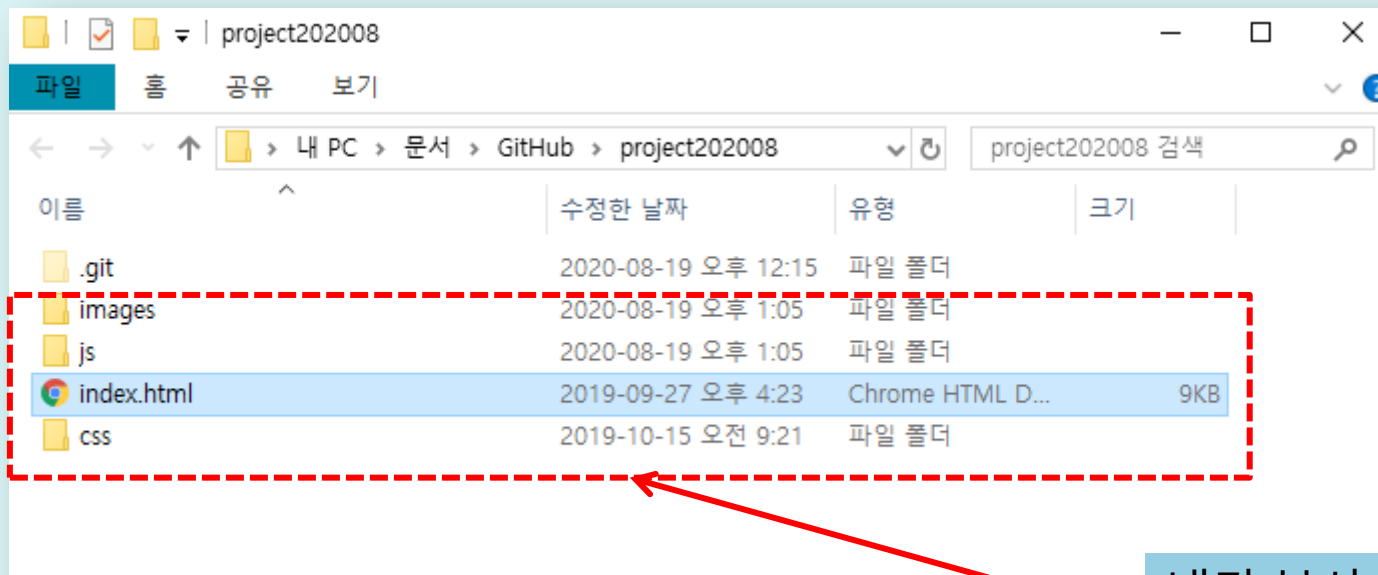


Show in Explorer 버튼을 클릭한다.
(탐색기가 뜨지만 현재 올린파일이 없으므로 git이라는 관리파일 외엔 아무것도 안보인다.)

11. 로컬에 복제된 폴더에 파일올리기



- 이전 슬라이드에서 클릭하여 탐색기를 연후 여기에 올리고자 하는 프로젝트 파일을 복사하여 넣는다.

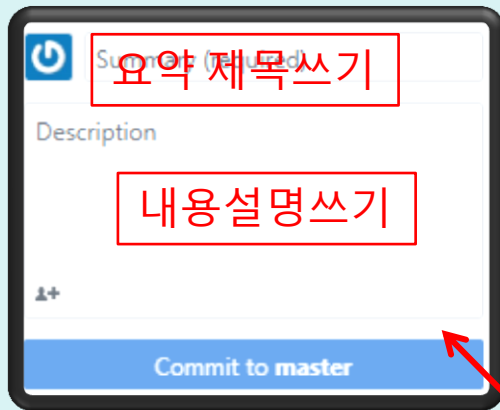


내가 복사하여 넣은
프로젝트 파일

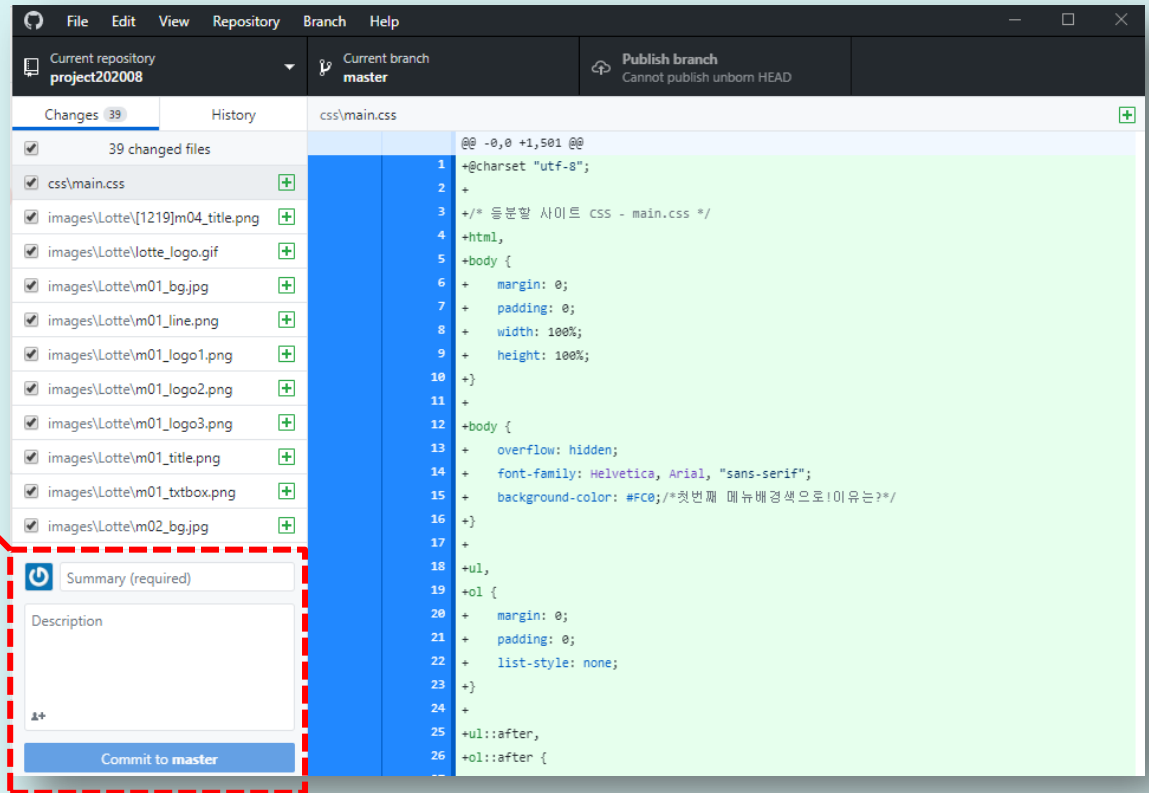
12. 복제하여 넣은 파일을 깃허브 데스크탑에서 확인



- 좀 전에 복사하여 넣은 로컬 복제 폴더가 그대로 깃허브 데스크탑 프로그램에서 볼 수 있다.



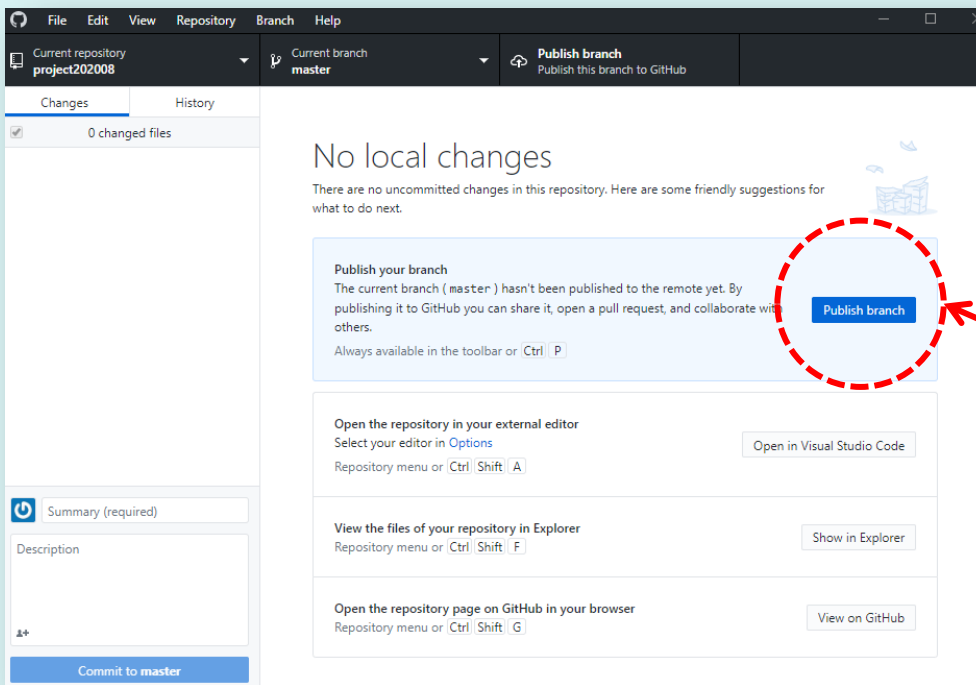
왼쪽 아래에서 변경된 파일들에 대한 간단한 제목과 내용을 쓰면 아래쪽 Commit to master 버튼이 활성화 된다. 이때 클릭! (제목만 써도 활성화됨)



13. Commit to master 클릭 후 서버에 반영하기



- Commit to master 버튼을 클릭하면 마스터 쪽 로컬 반영분을 서버에 실제로 반영시키는 버튼이 보이게된다.



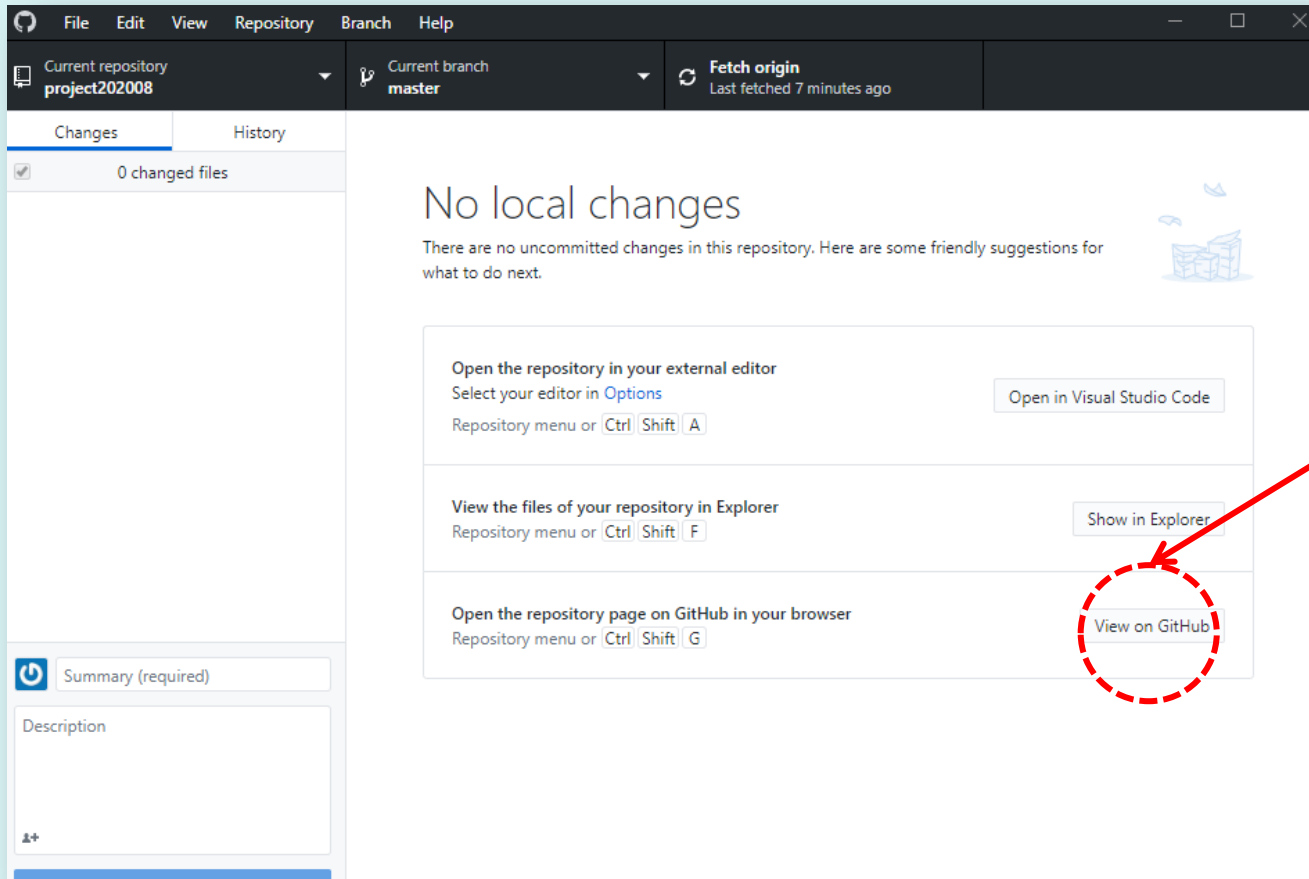
Publish branch 버튼을 클릭

- 이것은 실제로 서버에 배포하여 적용한다는 의미다.
- 이것을 클릭하면 실제 서버에 본 내용이 반영되어 나타난다.

14. 서버 반영 후 깃 허브 데스크탑에서 이동함



- 서버 반영 후에 깃허브 열기 버튼을 클릭하여 확인한다.



View on GitHub 버튼을 클릭하여 본 깃허브 사이트로 이동한다.

15. 반영분을 깃허브 사이트에서 확인



- 로컬 변경분이 서버에 반영되었음을 확인하다.

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

tombraceparker / project202008

<> Code ! Issues 🔗 Pull requests 🔄 Actions 📁 Projects 📖 Wiki 🛡 Security 📊 Insights ⚙ Settings

🔑 master 1 branch 0 tags Go to file Add file Code

tombraceparker 처음 업로드 89bf9d3 1 hour ago 1 commits

css	처음 업로드	1 hour ago
images	처음 업로드	1 hour ago
js	처음 업로드	1 hour ago
index.html	처음 업로드	1 hour ago

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

16. 깃허브 사이트에 올린 index페이지 노출하기



- 깃허브 사이트 Settings 메뉴에서 GitHub Pages 부분에서 Source 에 마스터를 설정한다.

The screenshot shows the GitHub Pages settings interface. The 'Source' section is highlighted with a red dashed box. It contains a dropdown menu for 'Branch' set to 'master', a folder icon for '/ (root)', and a 'Save' button. A red arrow points from the 'Save' button to a smaller inset window on the right. The inset window shows the 'GitHub Pages' status as 'Published' with a green checkmark and a URL: 'https://tombraceparker.github.io/project202008/'.

None 에서 master로
변경 선택한다

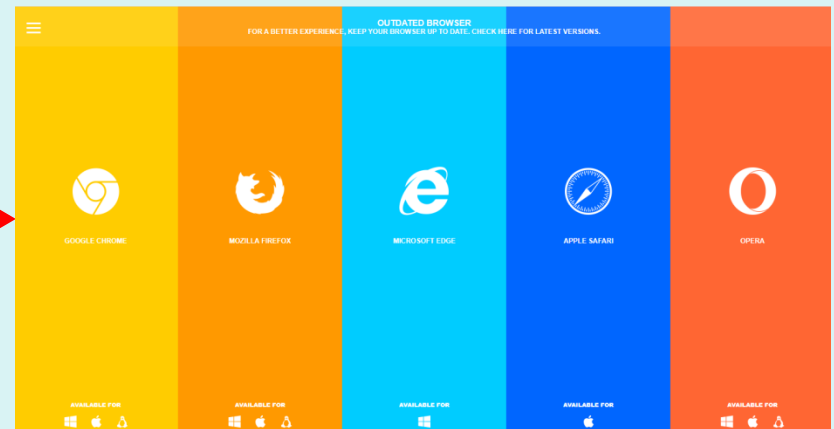
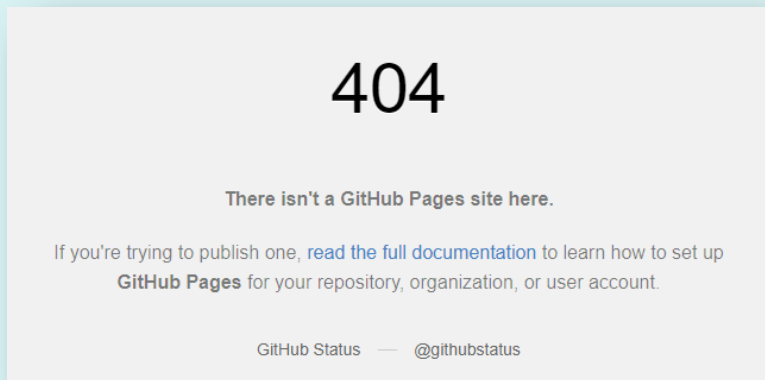
그 후 Save버튼을 클릭한다.

저장 후엔 위와 같이 사
이에 링크가 생김. 이때
이 주소를 클릭하면 404
페이지 없음이 나옴

17. 404 페이지 없음이 나오면 어떻게 하나?



- 이때 브라우저 창 상단에 슬래쉬 뒤에 실제로 존재하는 페이지 이름을 구체적으로 적어서 다시 엔터를 친다.
- 이때 이 사이트가 나온 주소를 복사한다.

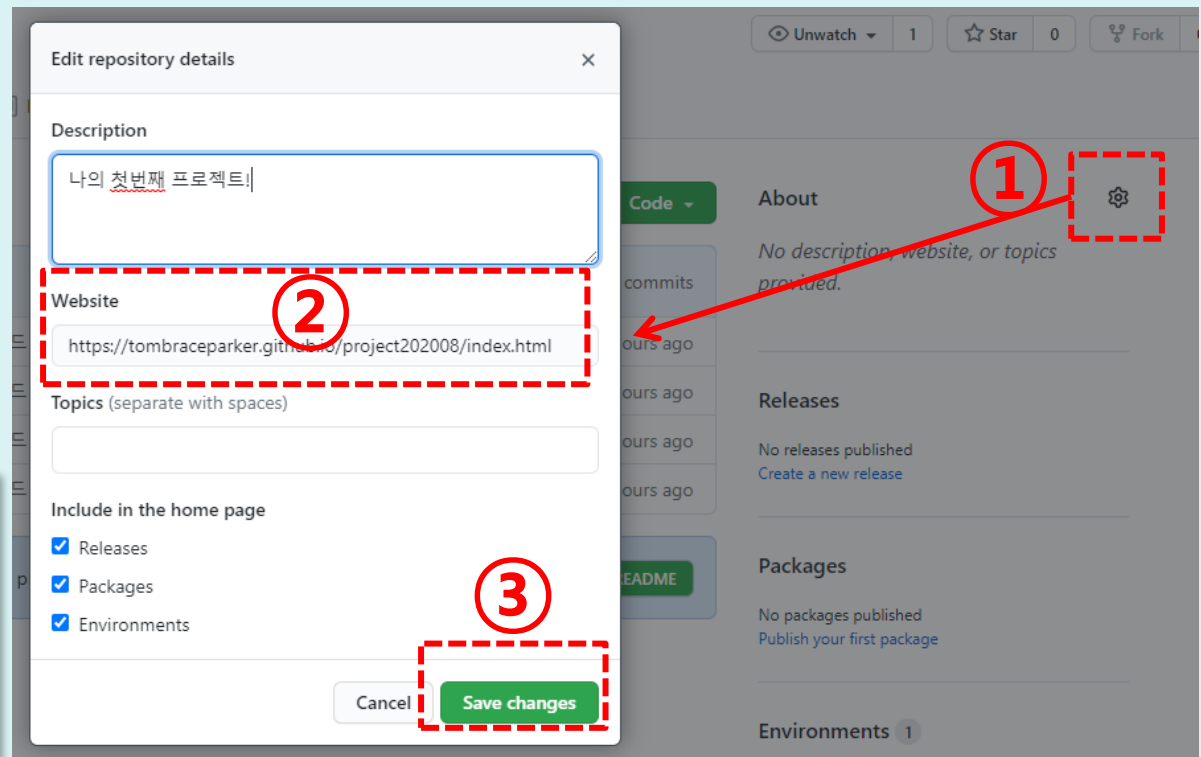


18. 깃허브에서 생성된 메인페이지 주소를 깃허브 첫 페이지 Code 메뉴 오른쪽에서 처리



- 깃허브에서 생성된 메인페이지 주소를 깃허브 첫 페이지 Code 메뉴 오른쪽의 About 옆의 톱니바퀴 모양 설정을 클릭한다.

- ① 설정 버튼 클릭
- ② 웹 사이트 주소 넣기
- ③ 변경 저장하기 클릭
- ④ 변경 후 오른쪽 About 설명 아래에 링크가 노출되어서 언제든지 열어서 볼 수 있음



19. 깃허브의 기본 개념

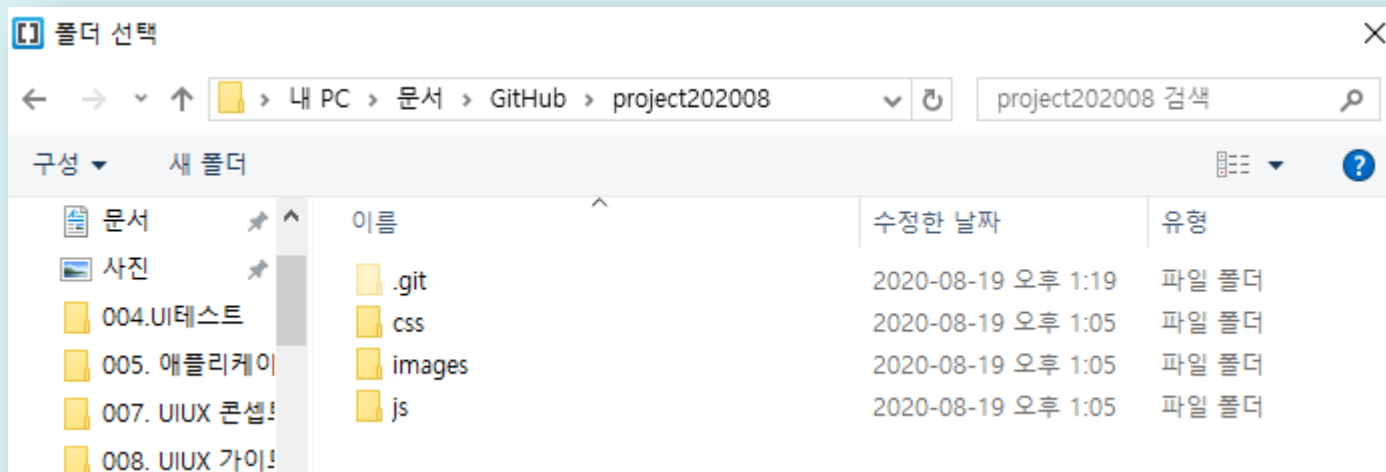


- 깃허브는 소스의 형상관리와 공동작업(co-work)를 위해서 탄생한 사이트다.
- 네트워크로 연결되어서 기본 자기의 저장 공간을 만들고 여기에 액세스 하여 소스를 업데이트 할 때 이것에 대한 버전이 남아서 언제든지 되돌리거나 할 수 있다.
- 관리를 위해 소스편집툴(예:브라켓, 코드, 서브라임텍스트 등)에서 복제한 사이트를 편집Root로 잡고서 이를 변경하고 테스트한 후 서버에 반영하는 순서로 작업을 진행하면 된다.

20. 브라켓과 깃허브 소스 연결하기



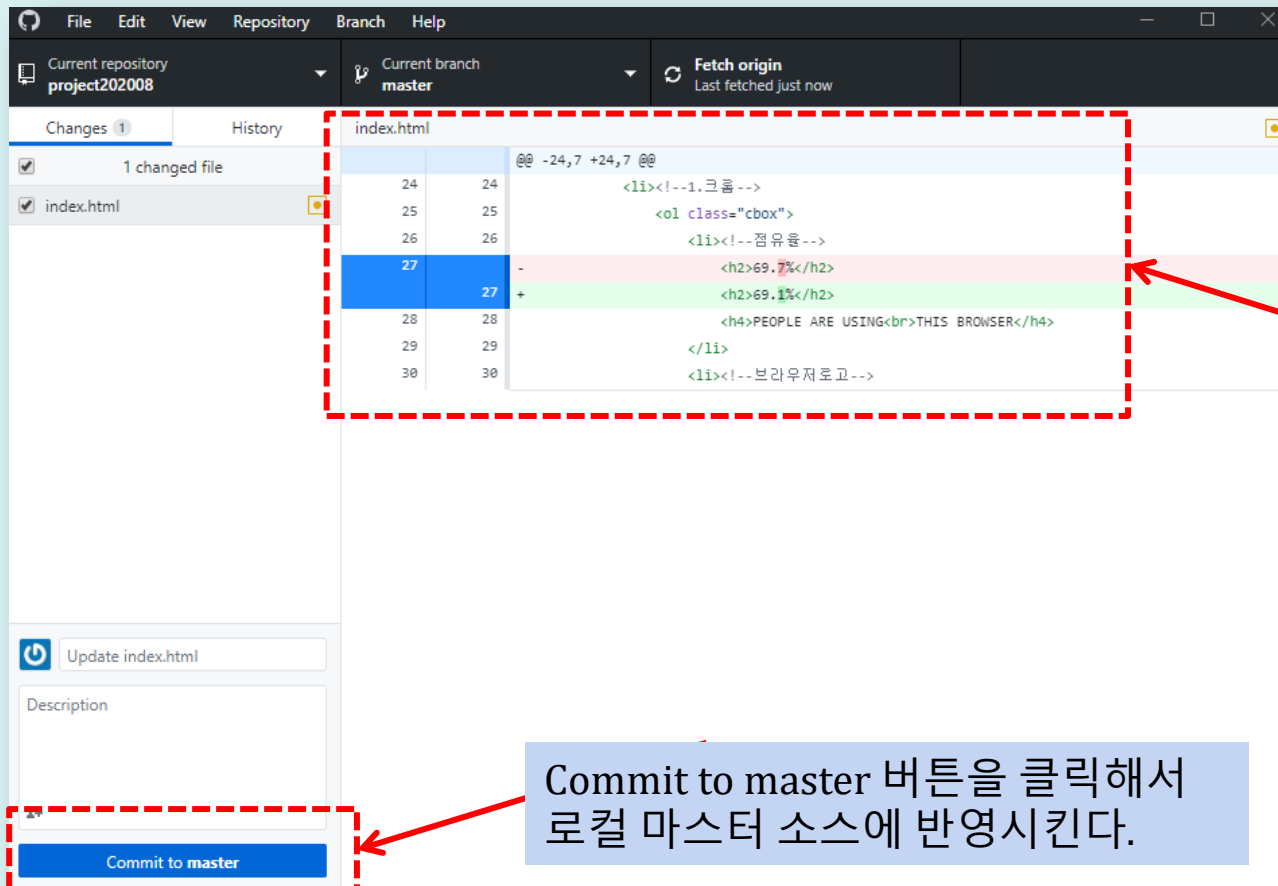
- 우선 브라켓을 열고 폴더설정을 깃허브에서 복제한 곳으로 설정을 한다.



21. 브라켓에서 본 소스를 수정한다.



- 브라켓에서 복제폴더의 파일을 열고 수정 후 저장한다.



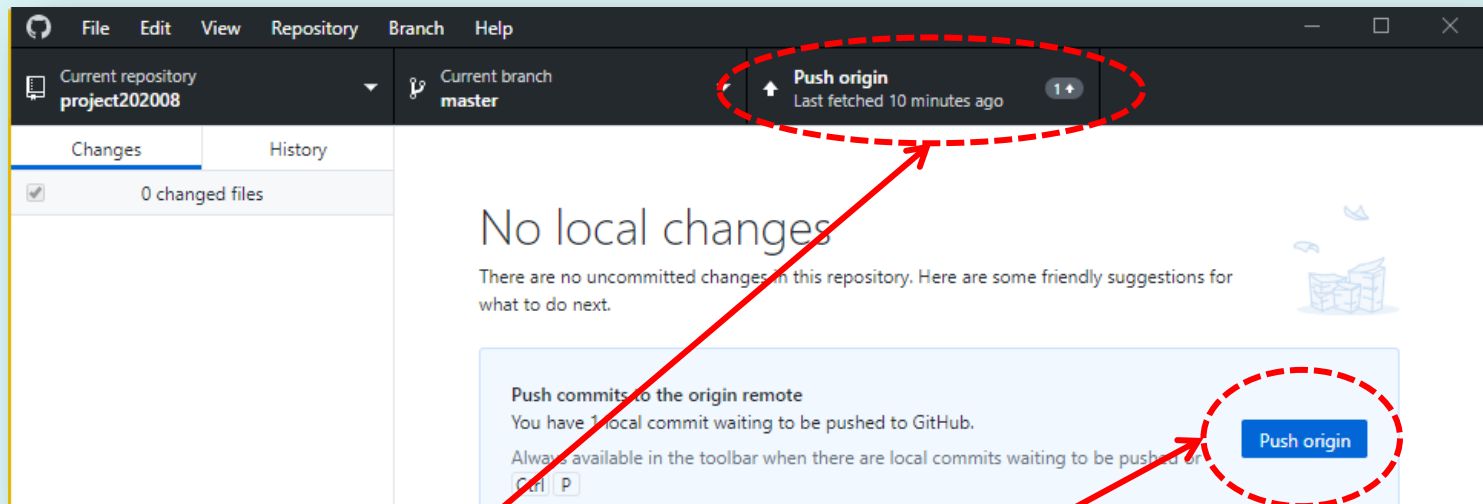
변경된 소스가 깃허브
데스크탑에서 보이며
이전 소스와 병행하여
표시된다.

Commit to master 버튼을 클릭해서
로컬 마스터 소스에 반영시킨다.

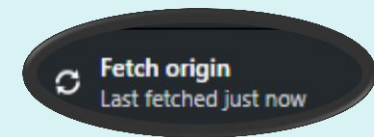
22. 변경된 소스를 서버 원본에 반영하기



- Commit to master 버튼으로 로컬 마스터에 반영하면 깃허브 데스크탑에서 이를 본 서버에 반영할 수 있도록 버튼이 생성된다.



두 군데 있는 Push origin 버튼을 클릭하여 서버에 있는 원본에 변경 파일분을 반영할 수 있다.



반영된 후엔 본 페이지에서는 사라지고 상단 버튼은 Fetch origin 으로 변경된다.

23.깃허브 서버의 원본에서 변경사항을 확인



- 깃허브 사이트에 가서 본 소스를 확인해 본다.

tombraceparker / project202008

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

master 1 branch 0 tags Go to file Add file Code

tombraceparker Update index.html ✓ 5e6888f 4 hours ago 3 commits

css	처음 업로드	7 hours ago
images	처음 업로드	7 hours ago
js	처음 업로드	7 hours ago
index.html	Update index.html	4 hours ago

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

24. 소스 되돌리기(Reset, Revert)

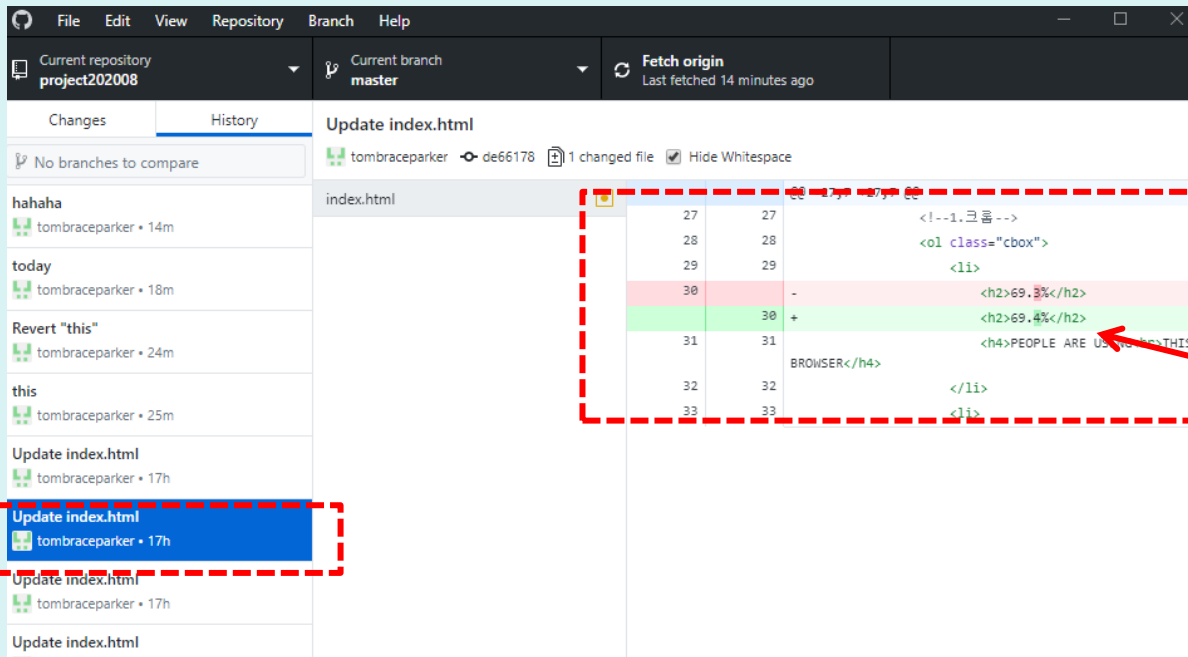


- 소스 되돌리기는 다음 두 가지 방법이 있다.
 - Reset : 돌아 가려는 커밋으로 리파지토리는 재설정되고, 해당 커밋 이후의 이력은 사라집니다.
 - Revert : Revert는 상태를 되돌린다고 볼 수 있습니다. 되돌린 이력이 새로 추가되어 보인다.
 - 주로 명령어로 이루어 지는 방식이라서 쉽게 접근하려면 깃허브 데스크탑에서 할 수 있는 소스 되돌리기 방식을 소개한다.(다음 25슬라이드)

25. 깃허브 소스 되돌리기 실제적용1



- 우선 MS Visual Studio Code 편집툴을 설치한다.
 - 이 툴은 깃허브와 자동연동되도록 등록되어 있다.
- 여러버전을 수정하여 깃허브 데스크탑에 보이게한다.

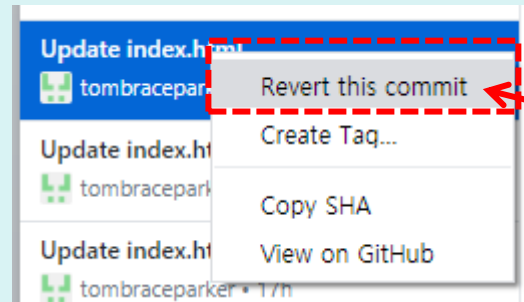


예컨데 수정이력 중 하나를 클릭하여 수정 이전 내용이 무엇인지 확인한다.

25. 깃허브 소스 되돌리기 실제적용2

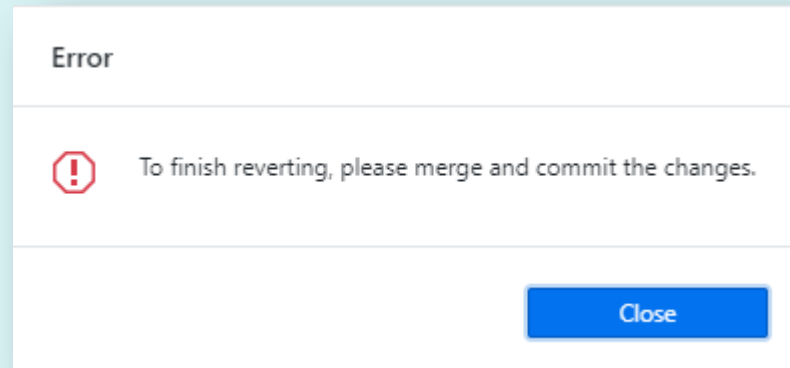


- 히스토리 리스트에서 선택한 되돌릴 버전에 오른쪽 클릭한다.



Revert this commit 버튼을 클릭한다.

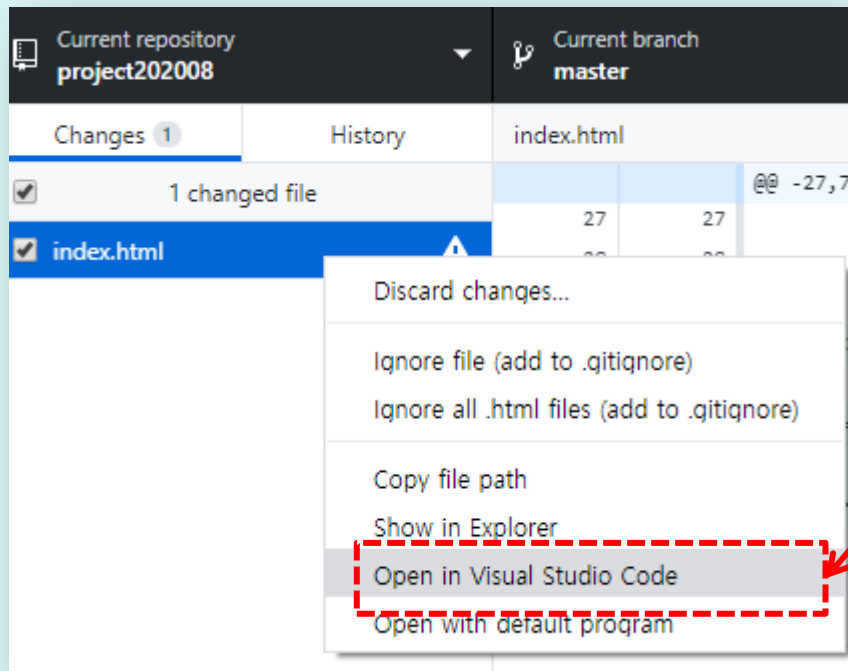
- 이때 에러 메시지가 나오지만 그대로 close버튼으로 닫는다.



25. 깃허브 소스 되돌리기 실제적용3



- Changes 탭으로 가서 방금 등록한 변경리스트를 확인 후 오른쪽 버튼을 클릭한다.



이 버전을 지우고자 할 때는 Discard changes... 버튼을 클릭하면 된다. 그러나 수락하려면 아래쪽 비주얼 스튜디오 코드 툴로 열어서 수락하면 된다.

25. 깃허브 소스 되돌리기 실제적용4



- VS Code툴을 통해 열린 소스는 수락옵션을 보여주는데 이 들 중 “수신변경사항수락”을 선택하면 소스가 수정된다.

```
<li>
현재 변경 사항 수락 | 수신 변경 사항 수락 | 모든 변경 사항 모두 수락 | 변경 사항 비교
<<<<<<< HEAD (현재 변경 사항)
<h2>69.1%</h2>
=====
>>>>>>> parent of de66178... Update index.html (수신 변경 사항)
<h2>69.3%</h2>
<h4>PEOPLE ARE USING<br>THIS BROWSER</h4>
</li>
```

수신변경사항이란
비교되고 있는 소스 중
아래쪽의 소스로 변경하
도록 수락하는 것이다.

- 수락후 저장을 해야만 깃허브 데스크탑 툴에서 변경분이 반영되어 체크표시가 생긴다.

25. 깃허브 소스 되돌리기 실제적용5



- 깃허브 테스트 탭에서 간략한 내용만 쓰고 커밋버튼을 클릭한다.

Current repository
project202008

Changes 1 History

1 changed file

index.html ✓

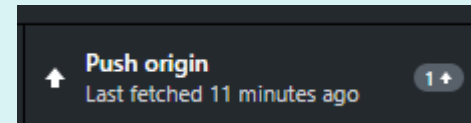
지금업데이트

Description

Commit to master

입력

클릭



Push origin 버튼을 클릭하여
실제 서버에 반영하여 확인하
면 이전 버전의 소스로 변경되
었음을 확인할 수 있다