



Manual of Github Homepage

Kyungmin Kwon

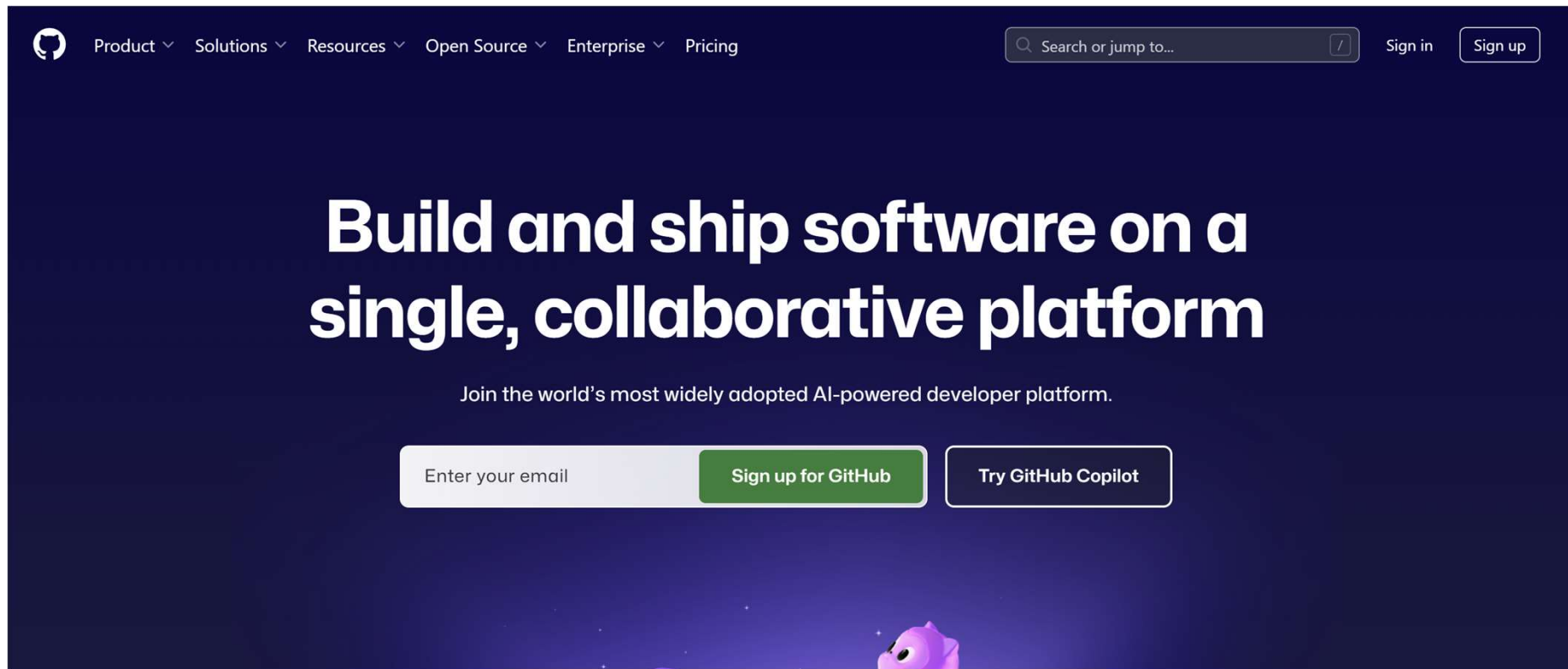


준비물 (Window 기준 설명)

Github 회원가입

- github에 회원가입하기

<https://github.com/>



Git 설치 (Window 기준 설명)

- Git 설치

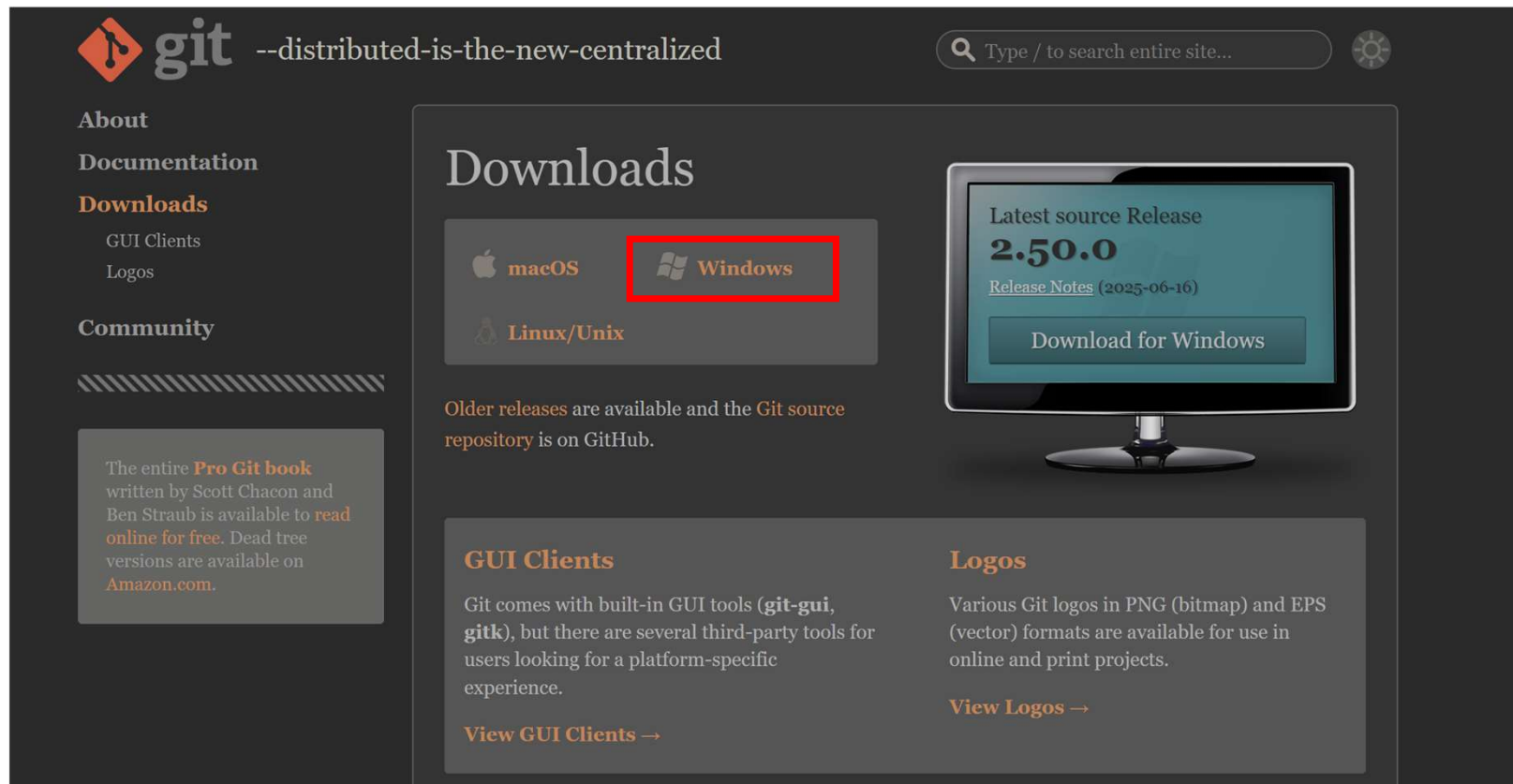
코드의 변경 내역을 효율적으로 추적하고 협업할 수 있도록 해주는 도구 (없어도 되지만 있으면 편함.)

만약 git을 안 쓴다면 수동으로 github에 파일들을 올려야 함.

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

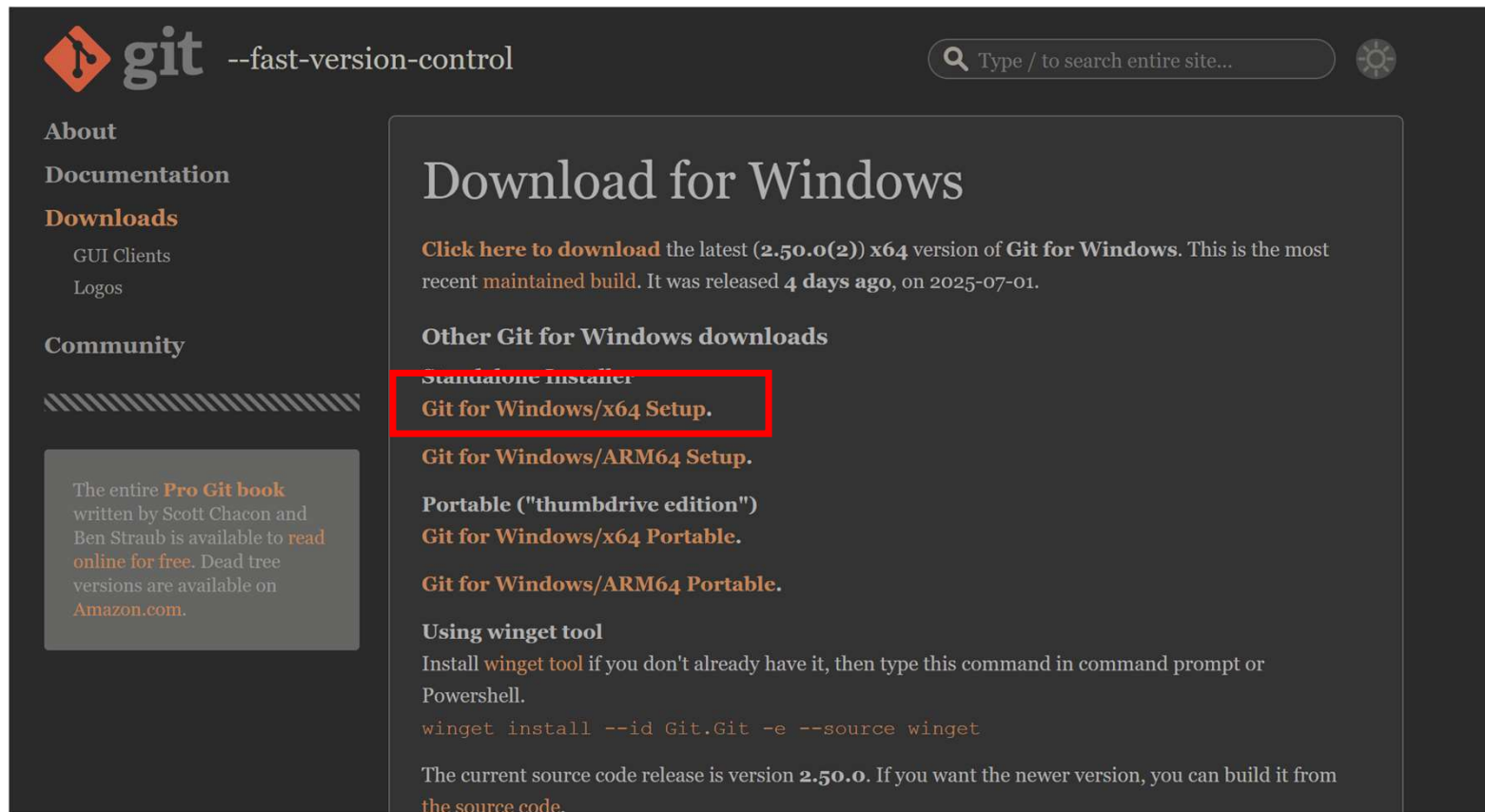
Git 설치

- Git 설치



Git 설치

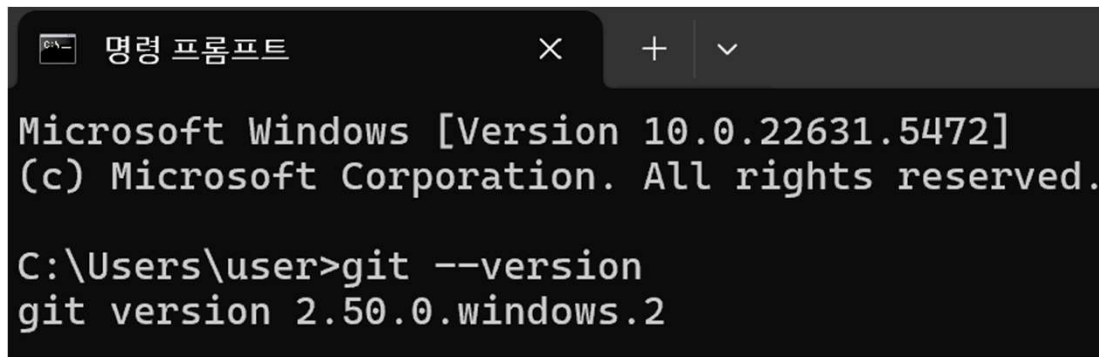
- Git 설치



Git 설치 완료 확인

- Git 설치가 완료되었다면,,
- 윈도우 검색에 cmd(명령 프롬프트) 검색 후 클릭

git --version



```
Microsoft Windows [Version 10.0.22631.5472]
(c) Microsoft Corporation. All rights reserved.

C:\Users\user>git --version
git version 2.50.0.windows.2
```

설치가 정상적으로 완료되었다면 위의 사진처럼 버전이 나옴.

Ruby 설치 (Window 기준)

- GitHub Pages는 내부적으로 Ruby 기반의 Jekyll을 사용하여 마크다운 파일을 HTML로 변환하고 사이트를 생성.
- <https://rubyinstaller.org/> 접속



[About](#) [Download](#) [Help](#)

The easy way to install Ruby on Windows

This is a **self-contained Windows-based installer** that includes the **Ruby language**, an execution environment, important **documentation**, and more.

Download

Add-ons



Ruby 설치



About **Download** Help

Downloads

RubyInstallers

Archives»

Not sure what version to download? Please read the right-hand column for recommendations.

WITH DEVKIT

- Ruby+Devkit 3.4.4-2 (arm)
- => **Ruby+Devkit 3.4.4-2 (x64)**
- Ruby+Devkit 3.4.4-2 (x86)
- Ruby+Devkit 3.3.8-1 (x64)
- Ruby+Devkit 3.3.8-1 (x86)
- Ruby+Devkit 3.2.8-1 (x64)
- Ruby+Devkit 3.2.8-1 (x86)
- Ruby+Devkit 3.1.7-1 (x64)
- Ruby+Devkit 3.1.7-1 (x86)

WITHOUT DEVKIT

- Ruby 3.4.4-2 (arm)
- Ruby 3.4.4-2 (x64)
- Ruby 3.4.4-2 (x86)
- Ruby 3.3.8-1 (x64)
- Ruby 3.3.8-1 (x86)
- Ruby 3.2.8-1 (x64)
- Ruby 3.2.8-1 (x86)
- Ruby 3.1.7-1 (x64)
- Ruby 3.1.7-1 (x86)

WHICH VERSION TO DOWNLOAD?

If you don't know what version to install and you're getting started with Ruby, we recommend that you use the **Ruby+Devkit 3.4.X (x64)** installer. It provides the biggest number of compatible gems and installs the MSYS2 Devkit alongside Ruby, so gems with C-extensions can be compiled immediately. The 32 bit (x86) version is not recommended, unless custom 32 bit native DLLs or COM objects have to be used.

HOW TO UPDATE?

Ruby can be updated to the latest patch version (e.g. from 3.3.0 to 3.3.3) by running the new installer version. Installed gems are not overwritten and will work with the new version without re-installation. It's sufficient to use the RubyInstaller without Devkit for these update installations. The Devkit can be updated separately using the `ridk install` command.

If the new Ruby version is from a different stable branch, then please use a new target directory for installation. That is to say, a previous RubyInstaller-3.3.x installation **should not** be updated by installing RubyInstaller-3.4.x into the same directory. This is because gems with C extensions are not compatible between ruby-3.3 and 3.4. Find out more in the [FAQ](#).

RUBYINSTALLER-HEAD

RubyInstaller-head is a daily-updated version of the **Ruby development branch**. It can be used for continuous integration tests (CI) on your gems or applications, so you're prepared for upcoming changes to the Ruby core.

다운로드!

Ruby 설치 완료

- 다운 완료하였다면, 아래처럼 화면이 나옴. (But 따로 할 작업은 X)



```

  _____
 /         \
| Ruby      |
| Installed |
|           |
|  For      |
| Windows   |
|
| 1 - MSYS2 base installation
| 2 - MSYS2 system update (optional)
| 3 - MSYS2 and MINGW development toolchain
|
| Which components shall be installed? If unsure press ENTER [1,3]
|
|_____

```

Ruby 설치 완료 확인

- Git 설치가 완료되었다면,,
- 윈도우 검색에 cmd(명령 프롬프트) 검색 후 클릭

ruby -v

gem -v

```
C:\Users\user>ruby -v
ruby 3.4.4 (2025-05-14 revision a38531fd3f) +PRISM [x64-mingw-ucrt]

C:\Users\user>gem -v
3.6.7

C:\Users\user>
```

- 추후 에러 방지를 위해 최신버전으로 업데이트

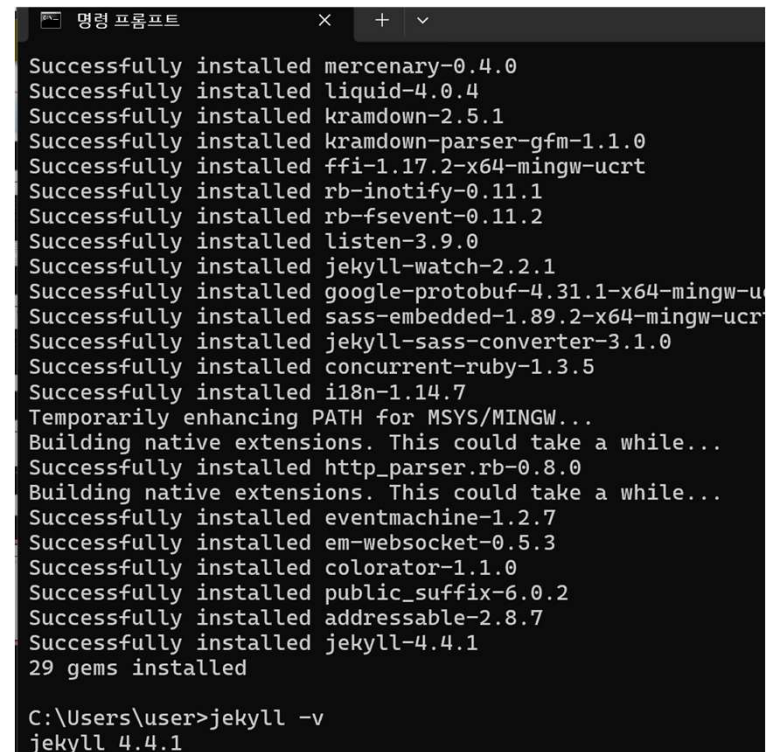
gem update --system 3.6.9

Jekyll 설치

- Jekyll은 마크다운 문서를 자동으로 HTML로 변환해주는 정적 사이트 생성기이기 때문에, 블로그나 문서 사이트를 만들기 쉬움

```
gem install bundler Jekyll
```

```
jekyll -v
```



```
명령 프롬프트
Successfully installed mercenary-0.4.0
Successfully installed liquid-4.0.4
Successfully installed kramdown-2.5.1
Successfully installed kramdown-parser-gfm-1.1.0
Successfully installed ffi-1.17.2-x64-mingw-ucrt
Successfully installed rb-inotify-0.11.1
Successfully installed rb-fsevent-0.11.2
Successfully installed listen-3.9.0
Successfully installed jekyll-watch-2.2.1
Successfully installed google-protobuf-4.31.1-x64-mingw-ucrt
Successfully installed sass-embedded-1.89.2-x64-mingw-ucrt
Successfully installed jekyll-sass-converter-3.1.0
Successfully installed concurrent-ruby-1.3.5
Successfully installed i18n-1.14.7
Temporarily enhancing PATH for MSYS/MINGW...
Building native extensions. This could take a while...
Successfully installed http_parser.rb-0.8.0
Building native extensions. This could take a while...
Successfully installed eventmachine-1.2.7
Successfully installed em-websocket-0.5.3
Successfully installed colorator-1.1.0
Successfully installed public_suffix-6.0.2
Successfully installed addressable-2.8.7
Successfully installed jekyll-4.4.1
29 gems installed

C:\Users\user>jekyll -v
jekyll 4.4.1
```



웹사이트 만들기

웹사이트를 위한 폴더 생성

- 원하는 폴더 이름 생성 (아래의 예시는 mysite가 폴더 이름)

`jeekyll new mysite`

```
Bundler: Resolving dependencies...
Bundler: Fetching rake 13.3.0
Bundler: Installing rake 13.3.0
Bundler: Fetching base64 0.3.0
Bundler: Fetching bigdecimal 3.2.2
Bundler: Fetching csv 3.3.5
Bundler: Fetching json 2.12.2
Bundler: Fetching rexml 3.4.1
Bundler: Fetching wdm 0.2.0
Bundler: Fetching tzinfo 2.0.6
Bundler: Installing base64 0.3.0
Bundler: Installing rexml 3.4.1
Bundler: Installing json 2.12.2 with native extensions
Bundler: Installing csv 3.3.5
Bundler: Installing wdm 0.2.0 with native extensions
Bundler: Installing bigdecimal 3.2.2 with native extensions
Bundler: Installing tzinfo 2.0.6
Bundler: Fetching tzinfo-data 1.2025.2
Bundler: Installing tzinfo-data 1.2025.2
Bundler: Fetching jeekyll-feed 0.17.0
Bundler: Fetching jeekyll-seo-tag 2.8.0
Bundler: Installing jeekyll-feed 0.17.0
Bundler: Installing jeekyll-seo-tag 2.8.0
Bundler: Fetching minima 2.5.2
Bundler: Installing minima 2.5.2
Bundler: Bundle complete! 7 Gemfile dependencies, 41 gems now installed.
Bundler: Use 'bundle info [gemname]' to see where a bundled gem is installed.
New jeekyll site installed in C:/Users/user/mysite.

C:\Users\user>
```

폴더 내에 웹사이트 파일들 생성

- 생성한 폴더 열기

`cd mysite`

- Ruby 프로젝트에서 필요한 모든 gem(패키지) 설치하기

`bundle install`

```
C:\Users\user\mysite>bundle install
Bundle complete! 7 Gemfile dependencies, 41 gems now installed.
Use `bundle info [gemname]` to see where a bundled gem is installed.
```

- 로컬 테스트 (사이트가 잘 만들어졌는지 test)

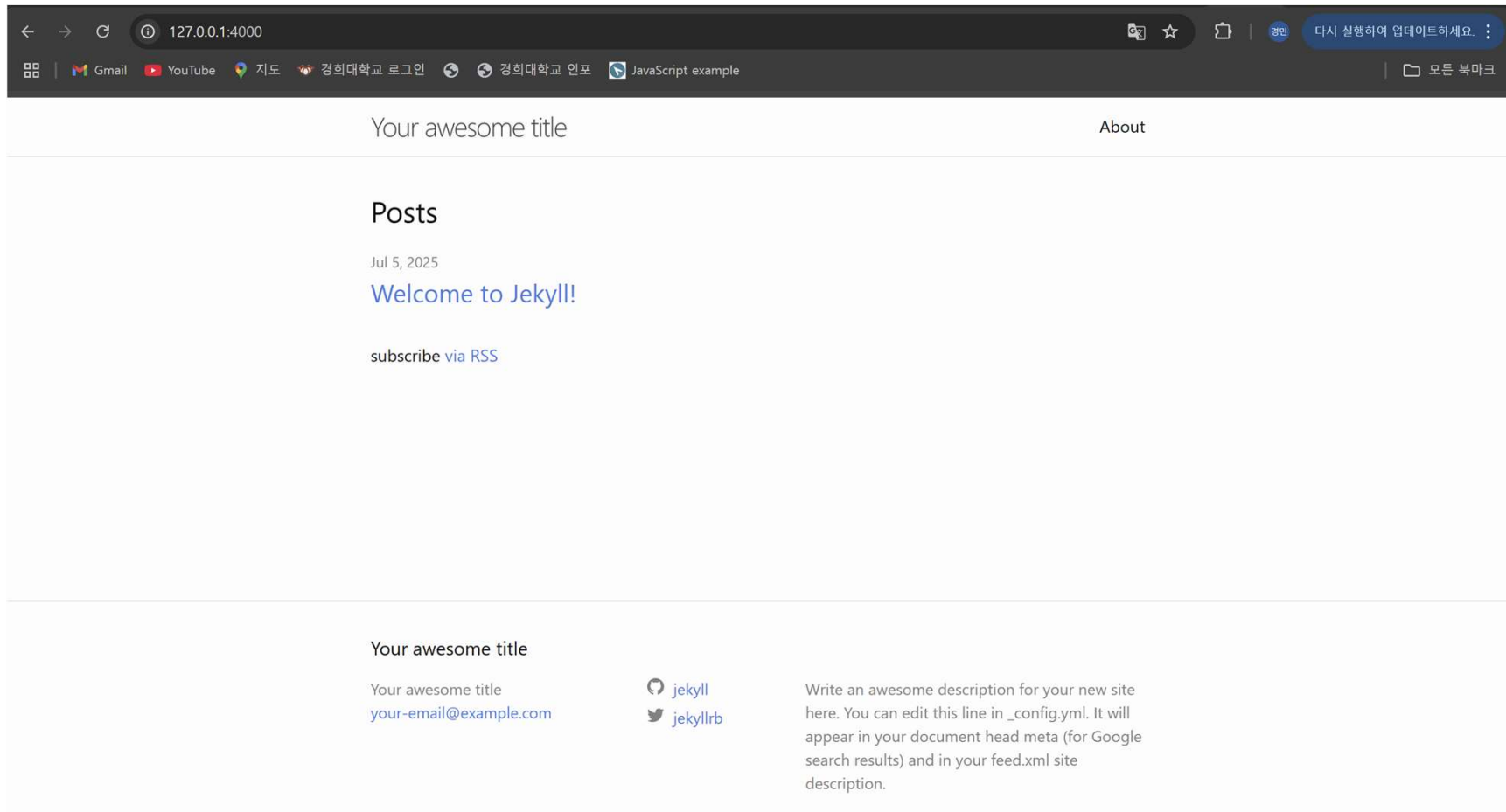
`bundle exec jekyll serve`

```
..\..\..\..\minima-2.5.2\_sass\minima\_base.scss 240:25 @import
minima.scss 48:3 @import
C:\Users\user\mysite\assets\main.scss 1:9 root stylesheet
Warning: 6 repetitive deprecation warnings omitted.
Run in verbose mode to see all warnings.
done in 1.244 seconds.
Auto-regeneration: enabled for 'C:/Users/user/mysite'
Server address http://127.0.0.1:4000/
Server running... press ctrl-c to stop.
```

복사 후 인터넷에 붙여넣기

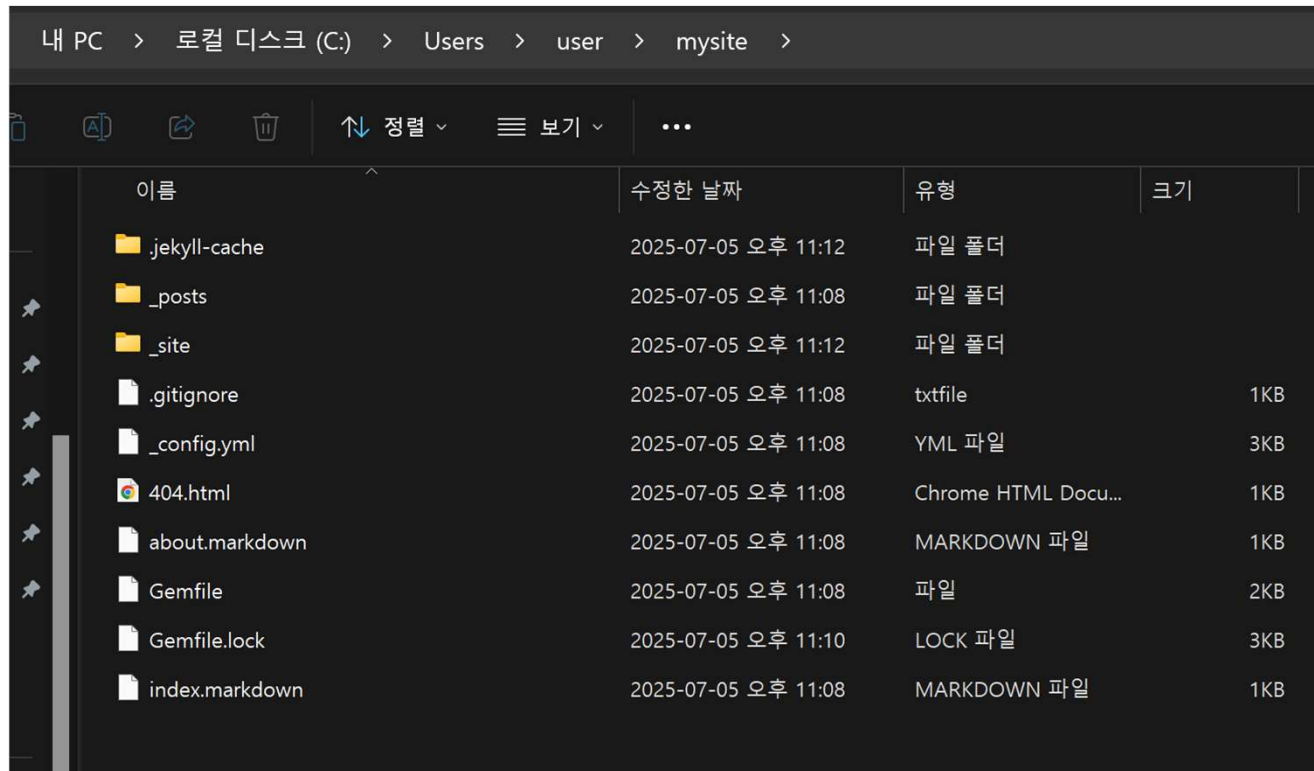
웹사이트 테스트

- 아래처럼 나온다면 성공!



웹사이트 테스트

- 윈도우 검색에서 mysite를 검색하면, 폴더 내 파일들이 생성된 것도 알 수 있음



이름	수정한 날짜	유형	크기
.jeekyll-cache	2025-07-05 오후 11:12	파일 폴더	
_posts	2025-07-05 오후 11:08	파일 폴더	
_site	2025-07-05 오후 11:12	파일 폴더	
.gitignore	2025-07-05 오후 11:08	txtfile	1KB
_config.yml	2025-07-05 오후 11:08	YML 파일	3KB
404.html	2025-07-05 오후 11:08	Chrome HTML Docu...	1KB
about.markdown	2025-07-05 오후 11:08	MARKDOWN 파일	1KB
Gemfile	2025-07-05 오후 11:08	파일	2KB
Gemfile.lock	2025-07-05 오후 11:10	LOCK 파일	3KB
index.markdown	2025-07-05 오후 11:08	MARKDOWN 파일	1KB



웹사이트 꾸미기

웹사이트 꾸미기

- ChatGPT나 gemini(원하는 ai)에게 연구자용 mysite 구조 및 기본 코드 알려달라고 말하기

Gpt한테 코드를 부탁할 때, 모바일도 호환되도록 부탁하면 좋은 것 같습니다.

원하는 헤드 베너, 내용, 웹사이트 스타일, 옵션 넣어달라고 하기.

- Mysite Structure **이건 제 구조입니다. 아래를 수정하여 이런 경로로 파일을 만들어달라고 해도 괜찮습니다.**

mysite/	
—— _layouts/	# 페이지 전체 구조를 정의하는 템플릿 디렉토리
—— default.html	# 모든 페이지에서 공통으로 사용하는 기본 레이아웃
—— _includes/	# 여러 레이아웃에서 재사용되는 HTML 조각들
—— head.html	# <head> 태그에 들어갈 공통 요소들 (title, meta 등)
—— header.html	# 상단 네비게이션 바 등 공통 헤더 구성
—— footer.html	# 하단 영역 HTML
—— _sass/	# SCSS 코드 조각들을 저장하는 디렉토리 (스타일 모듈화)
—— _variables.scss	# 색상, 폰트 등 스타일 변수 정의
—— _utilities.scss	# 공통 유틸리티 클래스 정의 (예: margin, flex 등)
—— assets/	# 정적 자산(스타일, 이미지 등)을 담는 디렉토리
—— css/	
—— style.scss	# 사이트 전체 스타일 정의 파일 (여기서 _sass 파일들을 import 함)
—— images/	
—— cat-logo.svg	# 원하는 홈페이지 로고 이미지 (저는 고양이로 했습니다)
—— banner.jpg	# 배너 이미지 (선택 사항)
—— index.md	# 홈페이지 콘텐츠 (Jekyll이 자동으로 HTML로 변환함)
—— research.md	# 연구에 대해 올리는 페이지
—— projects.md	# 프로젝트 소개 페이지
—— publications.md	# 논문/출판물 페이지
—— _config.yml	# 사이트 설정 파일 (제목, 테마 등 지정)
—— Gemfile	# 이 사이트에서 필요한 Ruby gem 목록 (Jekyll 포함)

웹사이트 꾸미기

- GPT에게 작성시킨 코드들을 아래의 경로에 맞게 **visual studio**를 사용하여 파일을 하나씩 만들어주기.
- ex) header.html 파일 만들기

```
mysite/
├── _layouts/
│   └── default.html
├── _includes/
│   ├── head.html
│   ├── header.html
│   └── footer.html
├── _sass/
│   ├── _variables.scss
│   └── _utilities.scss
└── assets/

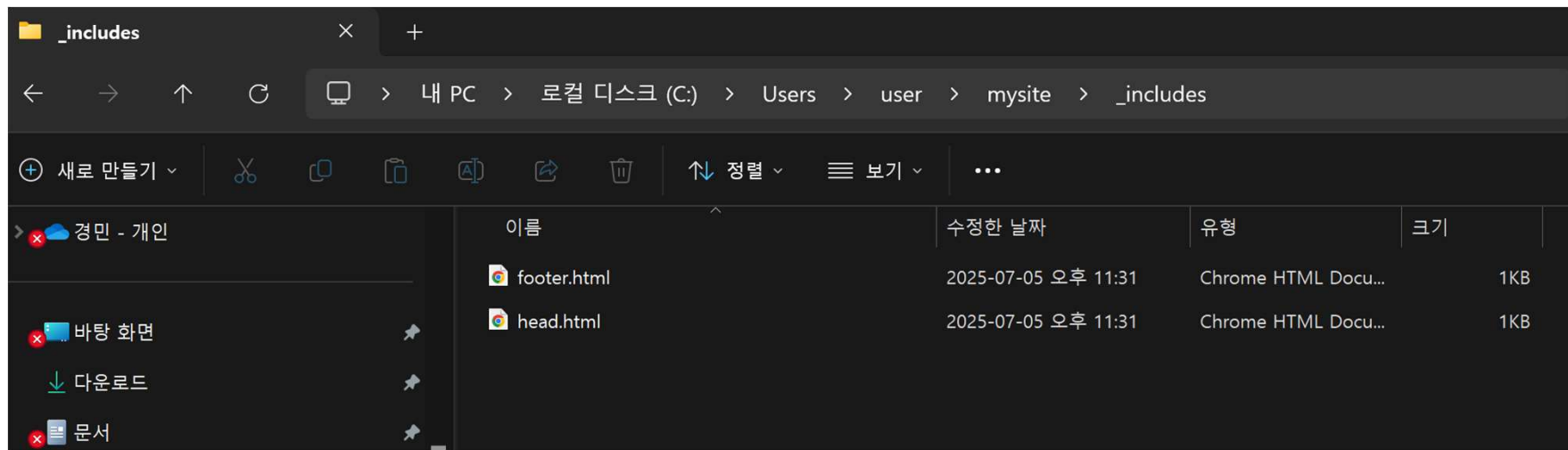
├── css/
│   └── style.scss
├── images/
│   ├── cat-logo.svg
│   └── banner.jpg
├── index.md
├── research.md
├── projects.md
├── publications.md
├── _config.yml
└── Gemfile
```

웹사이트 꾸미기 (예제)

- ex) header.html 파일 만들기

(1) 알맞은 경로에 들어가기

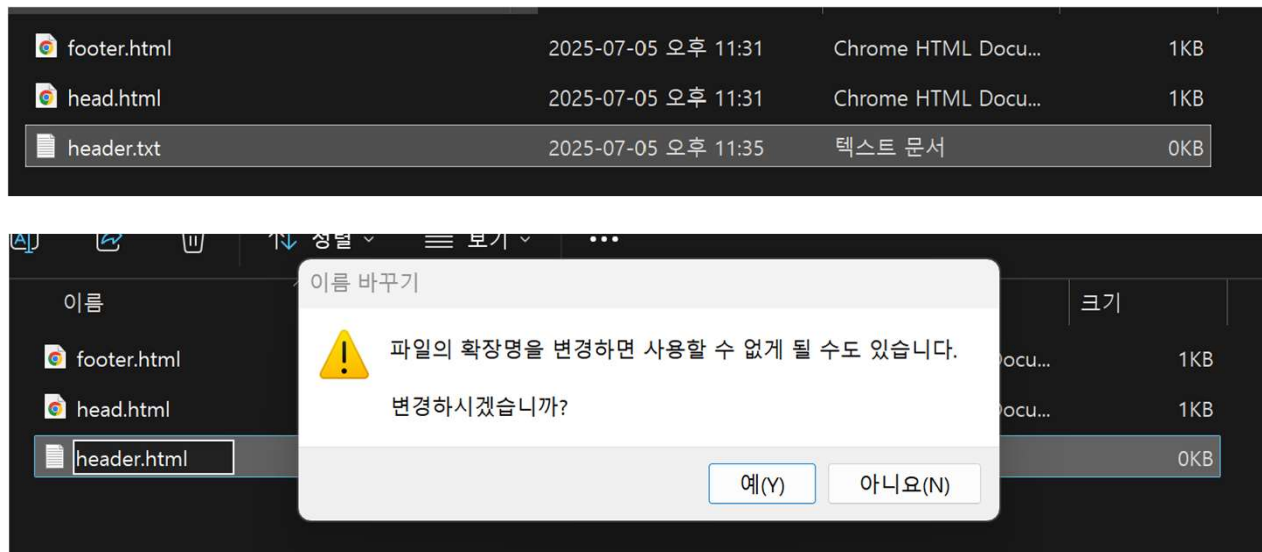
```
mysite/  
├── _config.yml  
├── _includes/  
│   └── header.html
```



웹사이트 꾸미기 (예제)

- ex) header.html 파일 만들기

(2) 텍스트 파일을 만들고, 확장자 이름을 파일에 맞는 확장자로 바꾸기. (이번 예시에서는 html)

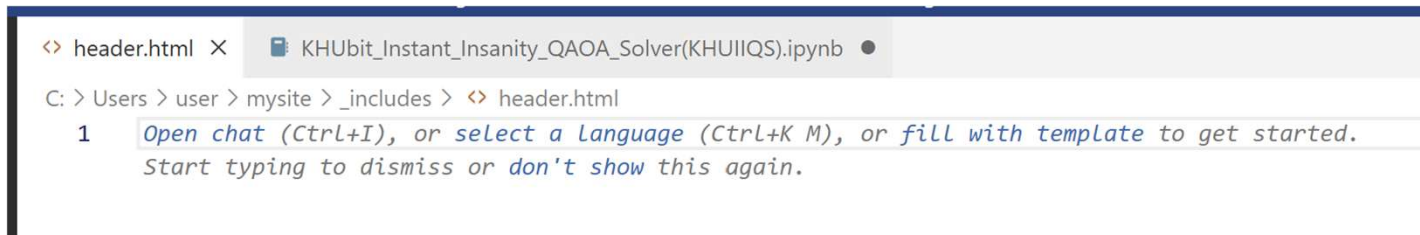


'예'를 누르고, 우클릭 -> 연결프로그램 -> visual studio code

웹사이트 꾸미기 (예제)

- ex) header.html 파일 만들기

(2) Visual studio를 연결하였으면, 아래에 코드 입력 (gpt가 알려준 기본 코드 or 자기가 짠 코드)



```
<> header.html X KHUbit_Instant_Insanity_QAOA_Solver(KHUIIQS).ipynb ●
C: > Users > user > mysite > _includes > <> header.html
1 Open chat (Ctrl+I), or select a language (Ctrl+K M), or fill with template to get started.
  Start typing to dismiss or don't show this again.
```

코드 입력 완료하였으면, 'ctrl + s'로 저장한 뒤, 같은 방법으로 모든 파일 생성

이때 그림 파일은 넣고 싶은 사진의 알맞은 확장자 (png, jpg 등등..)에 맞게 코드 쓰기

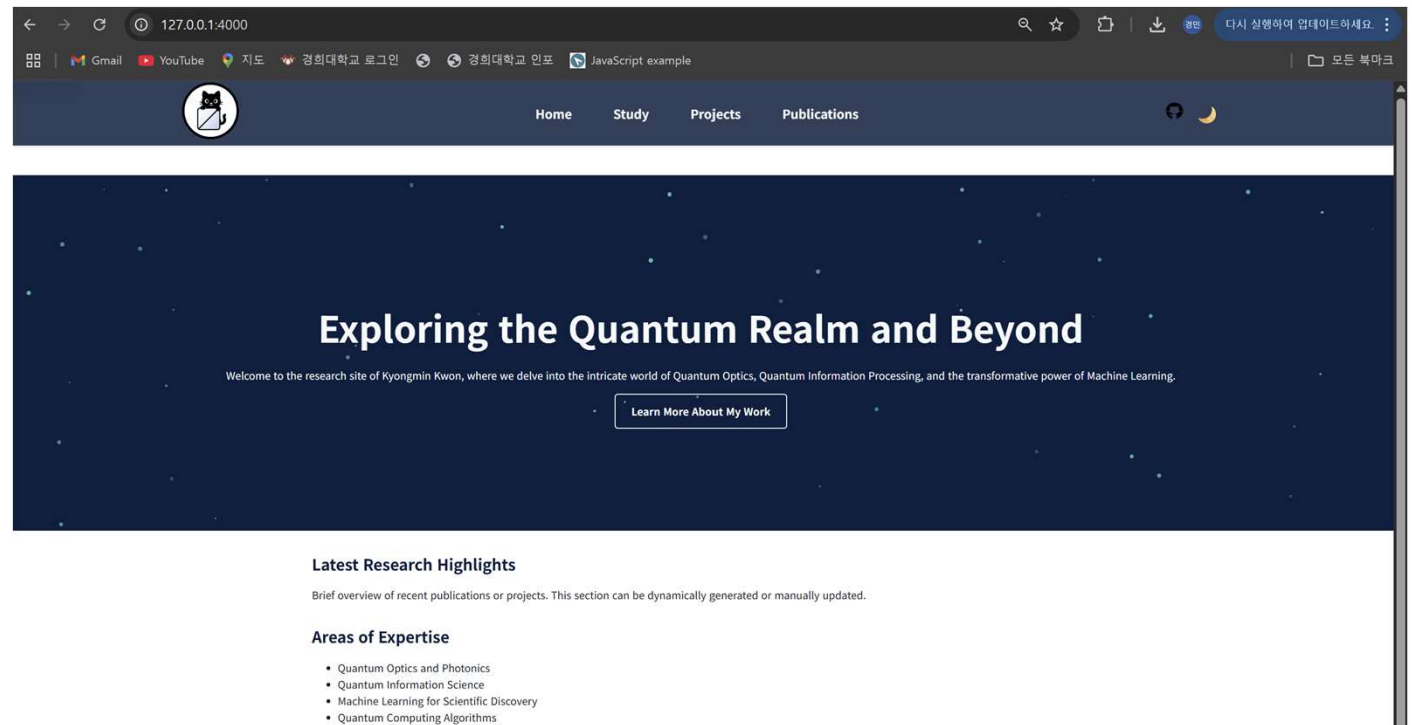
웹사이트 꾸미기 (예제)

- 모든 파일을 입력하였다면,, 아까 웹사이트 새로그침 or cmd에 코드 다시 작성

jeekyll new mysite

bundle exec jeekyll serve

아래처럼 수정한
website로
나온다면 성공!



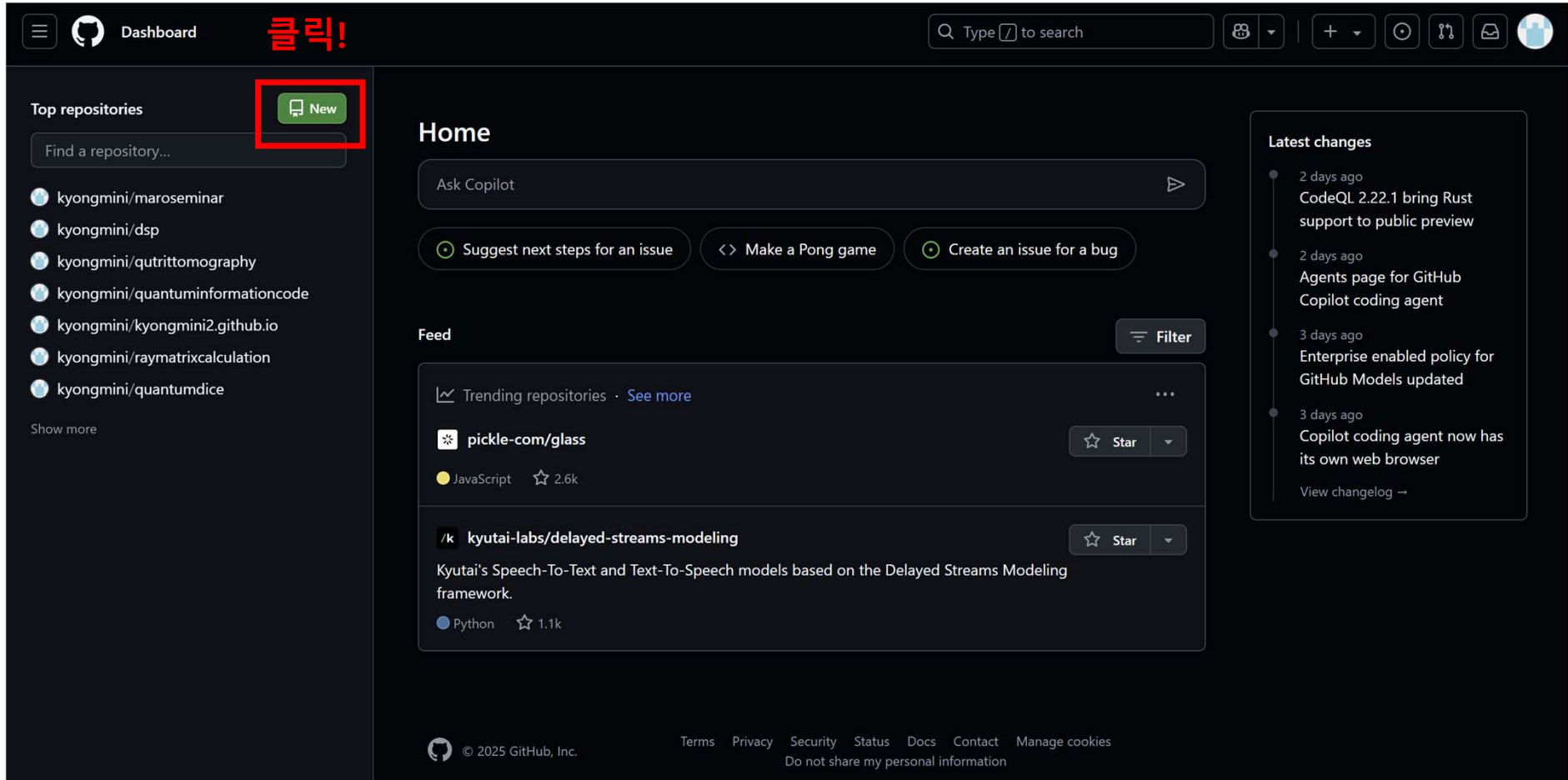


Github에 Push하기

Github에 Push 해야 하는 이유

- 아까 만든 웹사이트는 로컬 주소로만 가능하기 때문에, 타인은 접속할 수 없음. 모두가 접속할 수 있도록 github에 push해야 함.

Github에 Push 방법



Github에 Push 방법

Create a new repository

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

Required fields are marked with an asterisk (*).

Repository template

No template ▾

Start your repository with a template repository's contents.

Owner *

kyongmini ▾

Repository name *

kyongmini.github.io

✔ kyongmini.github.io is available.

반드시 github 닉네임으로 만들 것

Great repository names are short and memorable. Need inspiration? How about [super-duper-telegram](#) ?

Description (optional)

My research webiste



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.

Git 연결 및 푸시

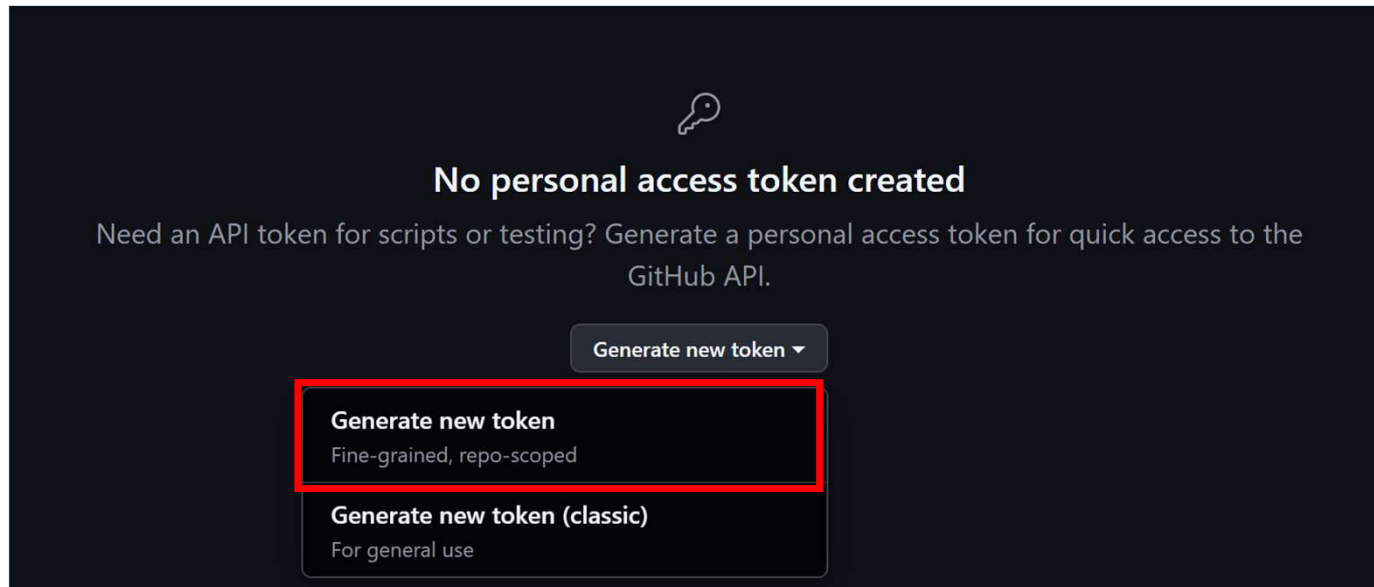
- github의 저장소 준비가 완료되었다면,,
- Git 연결 및 push 하기위해 cmd 들어가기! (파란 글씨는 자신에 맞게 수정)

```
git config --global user.name "Gildong Hong"
```

```
git config --global user.email "gildong@example.com"
```

Git 연결 및 푸시

- git에 연결하여 푸시하기 위해, <https://github.com/settings/tokens> 접속
- GitHub는 보안상의 이유로, 아이디+비밀번호 대신 토큰을 사용하는 것으로 정책이 바뀜



Git 연결 및 푸시

- token 이름은 원하는 대로

New fine-grained personal access token


Create a fine-grained, repository-scoped token suitable for personal API use and for using Git over HTTPS.

Token name *

✓ 'my-research-website-2025' is available.


A unique name for this token. May be visible to resource owners or users with possession of the token.

Description
Resource owner

 **kyongmini** ▼

The token will only be able to make changes to resources owned by the selected resource owner. Tokens can always read all public repositories.

Expiration

 **30 days (Aug 05, 2025)** ▼

Git 연결 및 푸시

- Only select repositories에서 아까 만든 저장소로 클릭!

Repository access

☐ **Public repositories**
Read-only access to public repositories.

☐ **All repositories**
This applies to all current and future repositories you own. Also includes public repositories (read-only).

☒ **Only select repositories**
Select at least one repository. Max 50 repositories. Also includes public repositories (read-only).

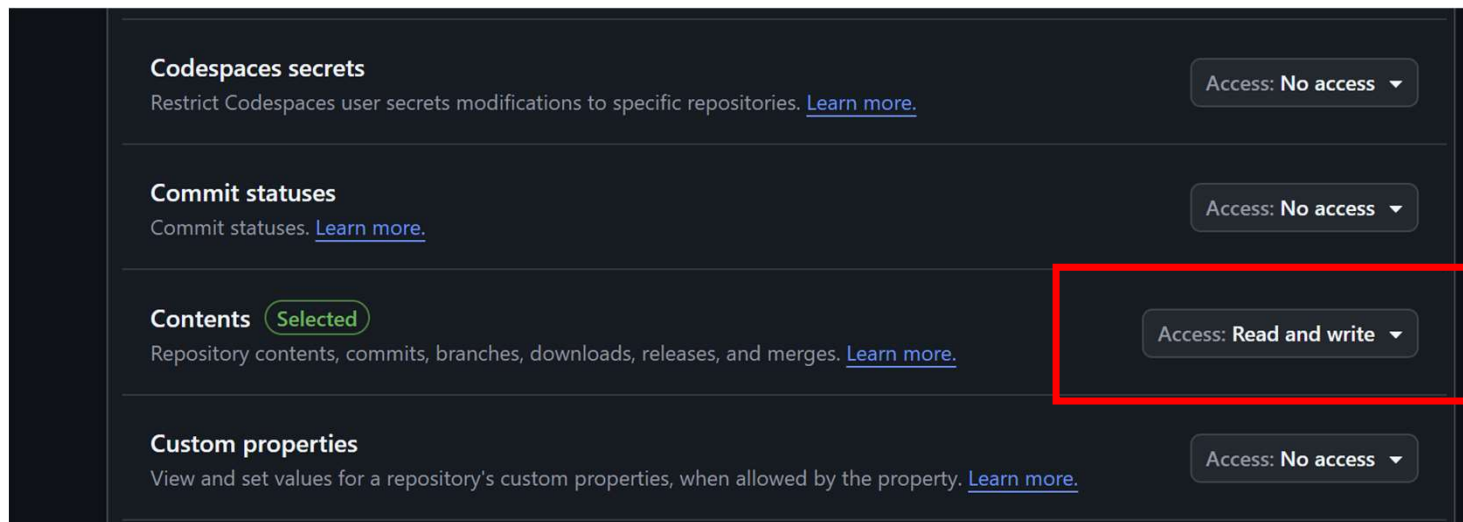
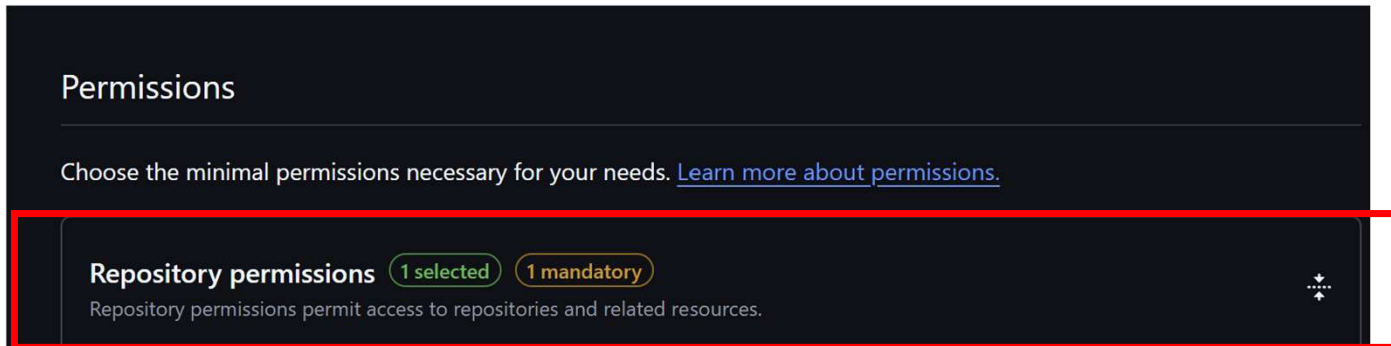
Select repositories 1 ▾

Selected 1 repository.

kyongmini/kyongmini.github.io ×

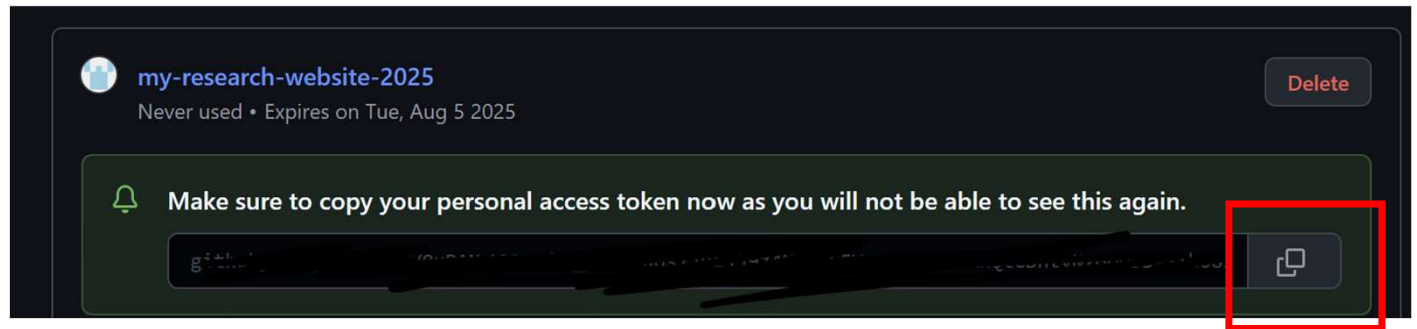
Git 연결 및 푸시

- Permissions -> Repository permissions 클릭!
- Contents -> Read and Write으로 수정 후 Token 생성



Git 연결 및 푸시

- Token 생성 완료하면, key를 얻을 수 있음. 복사하기!
- 다시 cmd로 돌아오기



```
cd mysite
```

```
git init
```

```
git remote add origin https://github.com/kyongmini/kyongmini.github.io
```

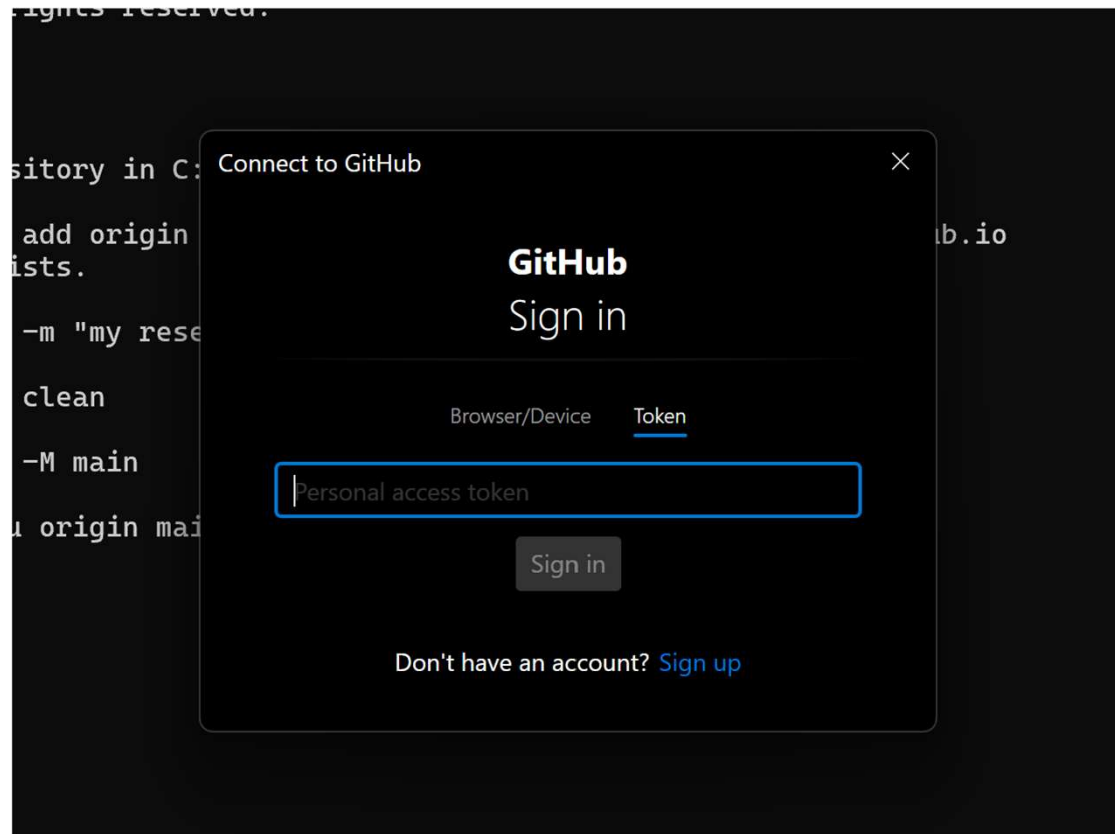
```
git commit -m "my research website"
```

```
git branch -M main
```

```
git push -u origin main
```

Git 연결 및 푸시

- 복사한 Token 입력



Git 연결 및 푸시

- 아래처럼 나온다면 Push 성공!


```
C:\Users\user\mysite>git commit -m "my research website"
On branch main
nothing to commit, working tree clean

C:\Users\user\mysite>git branch -M main

C:\Users\user\mysite>git push -u origin main
Enumerating objects: 31, done.
Counting objects: 100% (31/31), done.
Delta compression using up to 16 threads
Compressing objects: 100% (26/26), done.
Writing objects: 100% (31/31), 797.14 KiB | 21.54 MiB/s, done.
Total 31 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/kyongmini/kyongmini.github.io
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```

Git 연결 및 푸시

- Github 저장소에서 확인 가능

 **Kyungmin Kwon** my research website ✓

19a207e · 29 minutes ago 1 Commit

📁 _includes	my research website	29 minutes ago
📁 _layouts	my research website	29 minutes ago
📁 _sass	my research website	29 minutes ago
📁 assets	my research website	29 minutes ago
📄 .gitignore	my research website	29 minutes ago
📄 404.html	my research website	29 minutes ago
📄 Gemfile	my research website	29 minutes ago
📄 Gemfile.lock	my research website	29 minutes ago
📄 _config.yml	my research website	29 minutes ago
📄 about.markdown	my research website	29 minutes ago
📄 contact.md	my research website	29 minutes ago
📄 index.markdown	my research website	29 minutes ago
📄 index.md	my research website	29 minutes ago

My research webiste

Activity

0 stars

0 watching

0 forks

Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

Deployments 1

✓ **github-pages** now

Languages

SCSS 52.5%

JavaScript 25.2%

HTML 20.7%

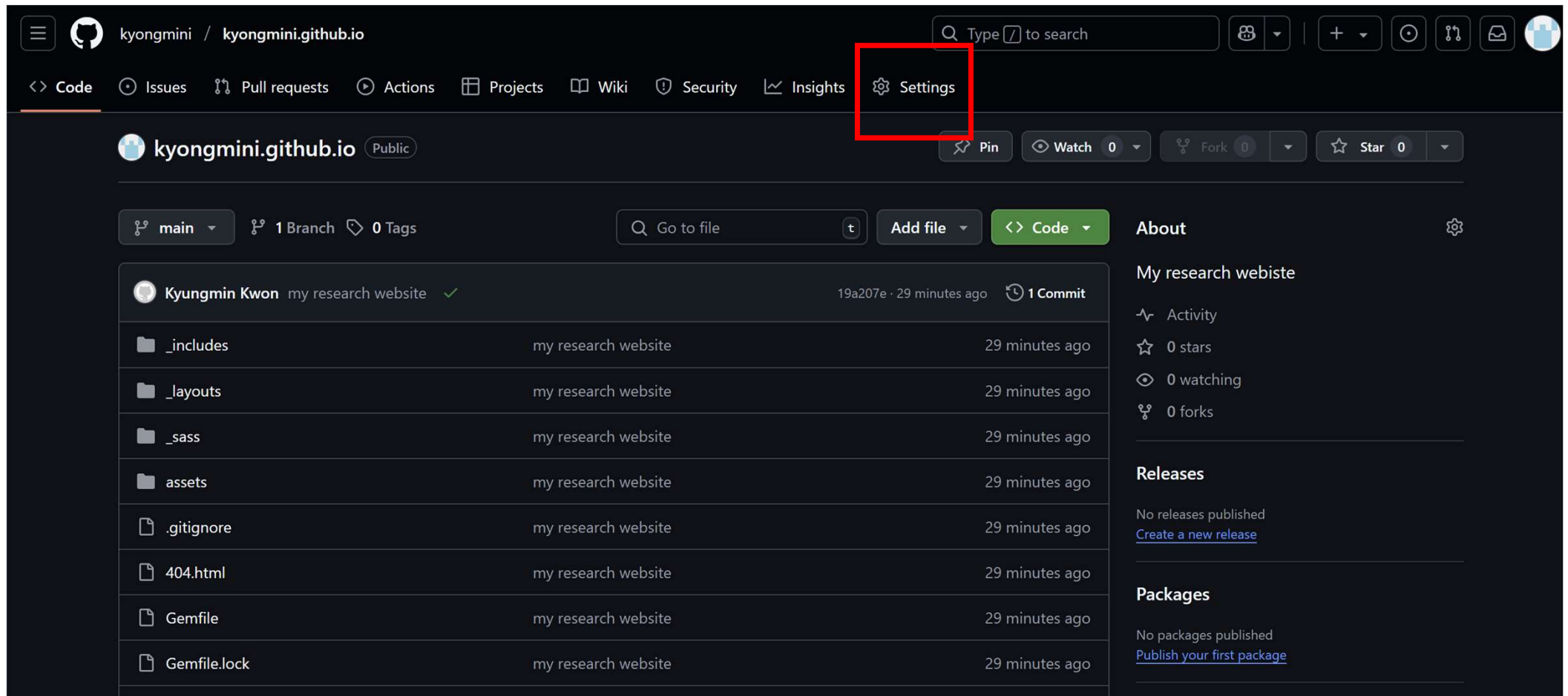
Ruby 1.6%



웹사이트 연동하기

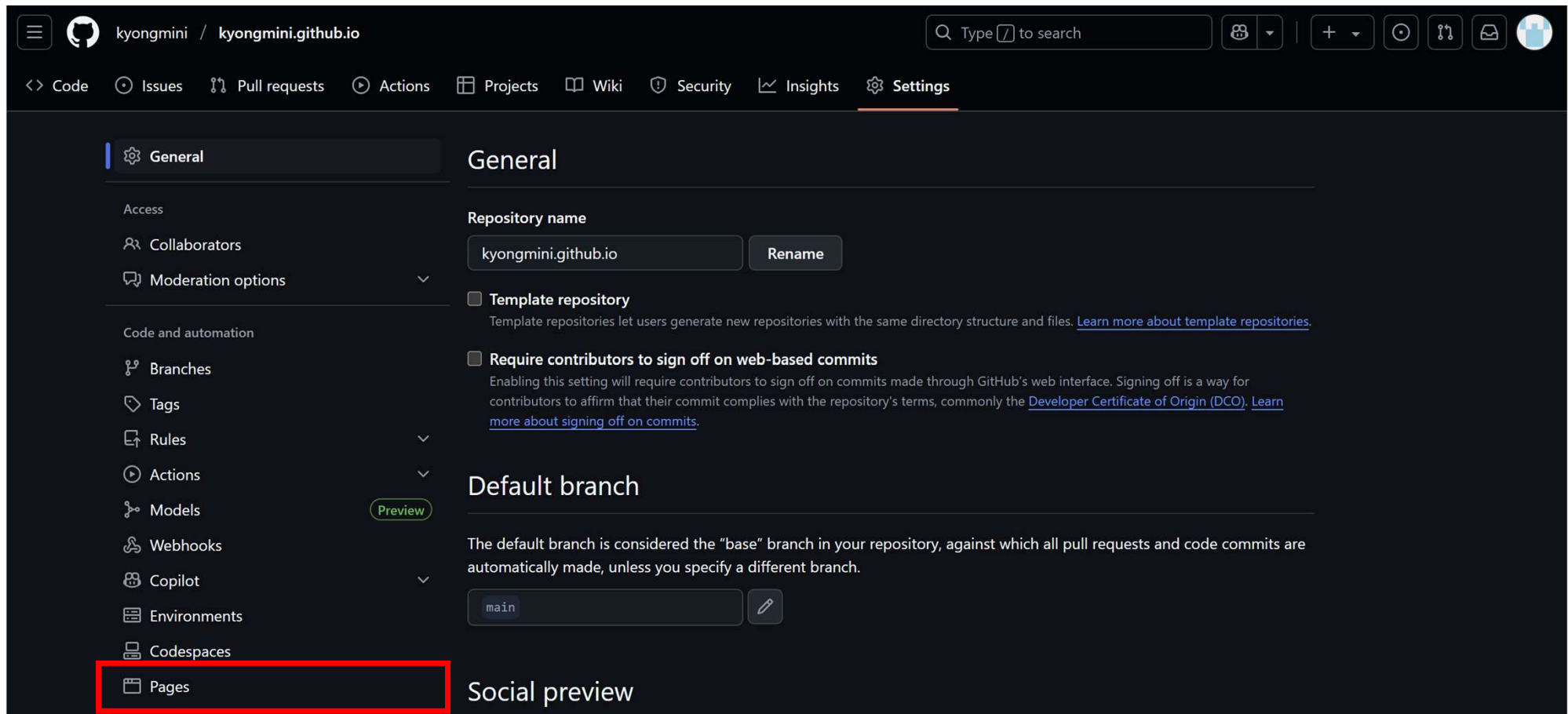
Github page 만들기

- 다시 github 저장소에 들어가서 Setting 클릭



Github page 만들기

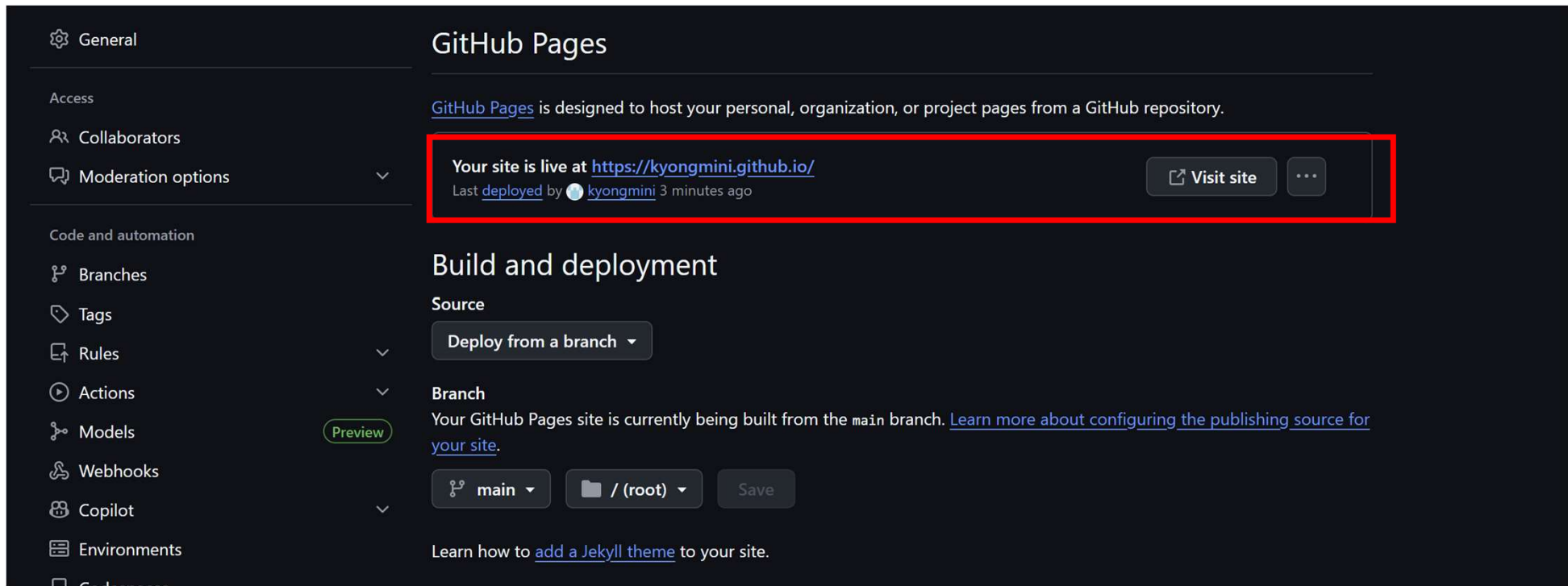
- Pages 클릭



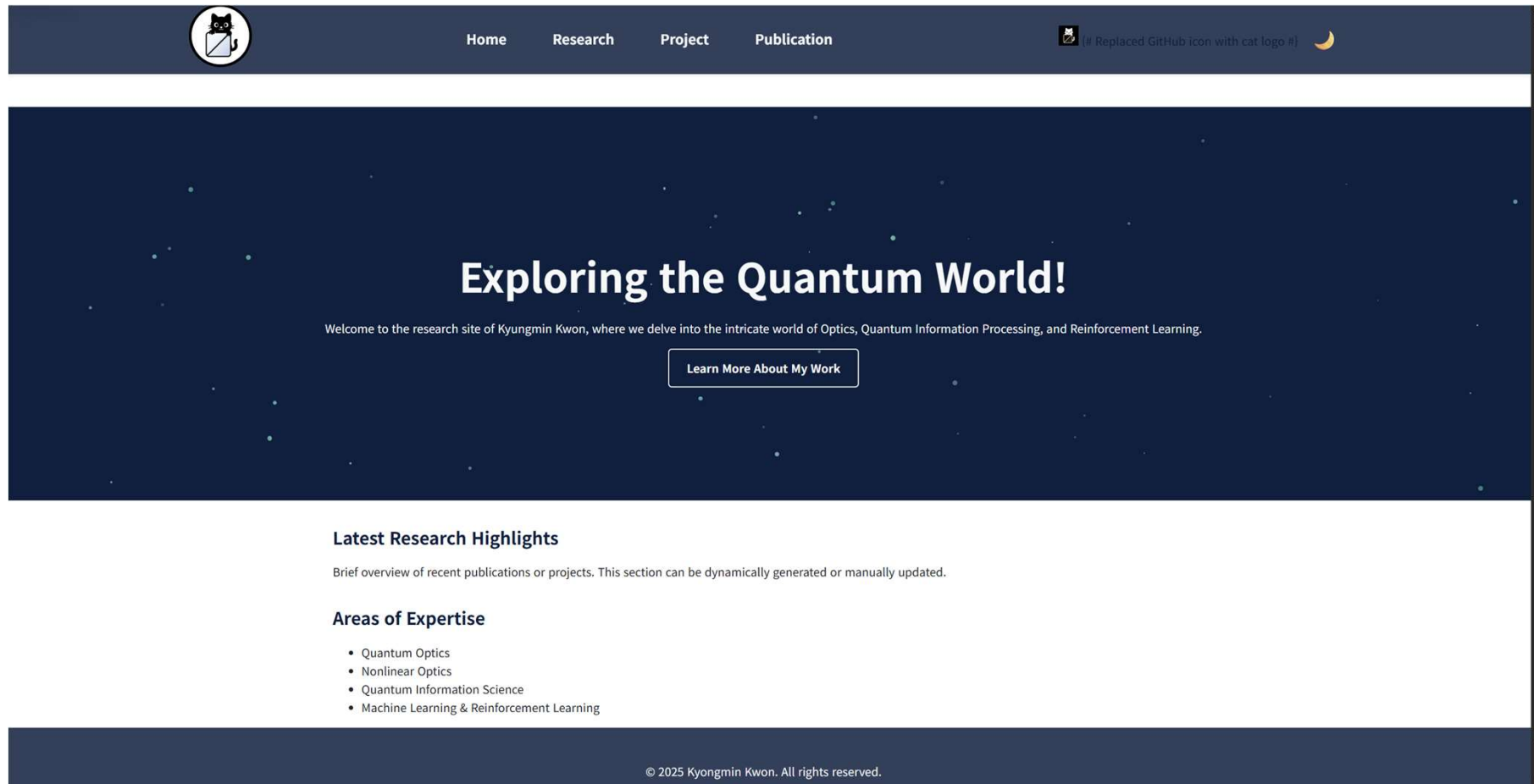
Github page 만들기

- 웹사이트 생성 완료 확인. Visit site 누르거나,

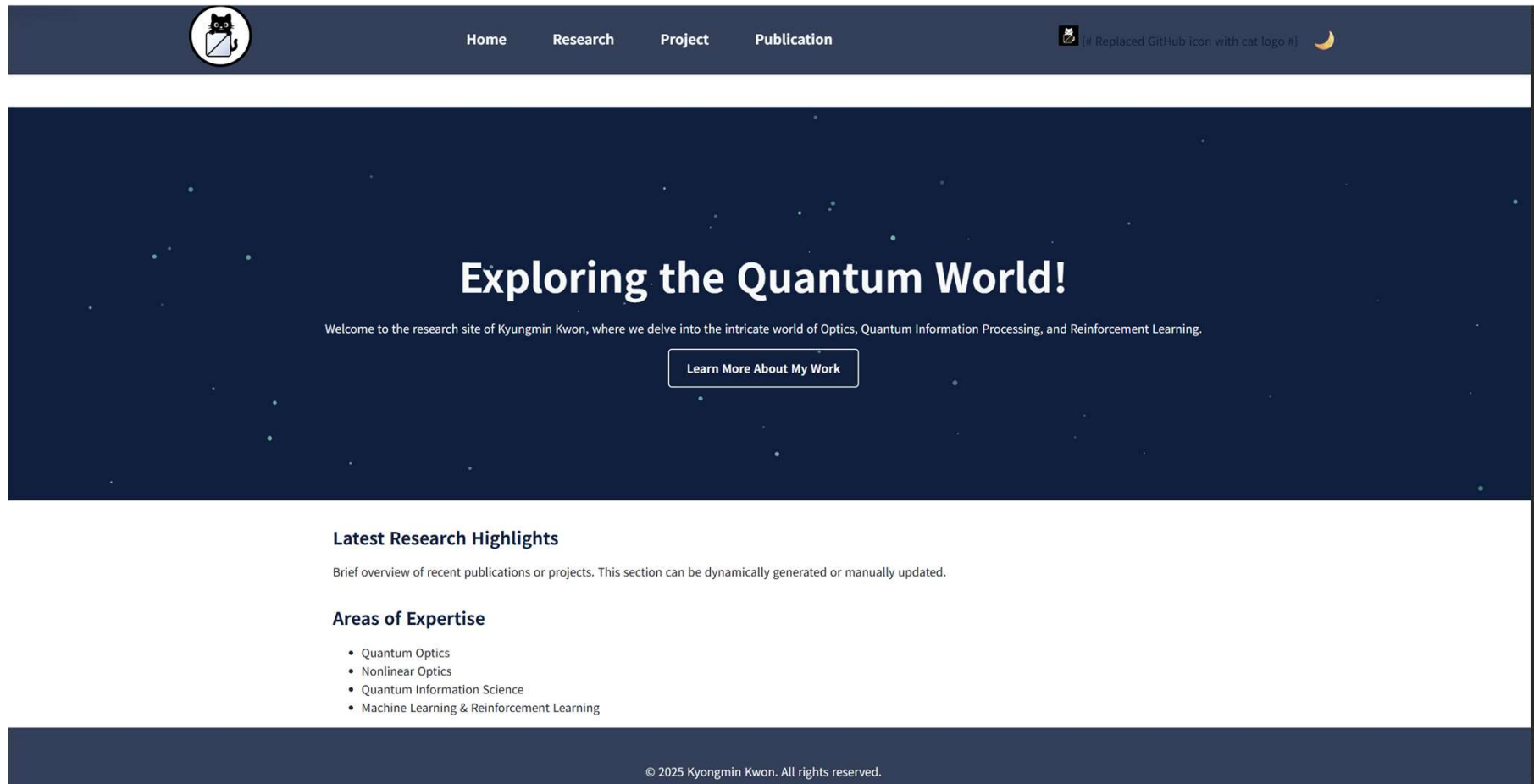
<https://자기닉네임.github.io>



Github Page



Github Page



Github page 만들기

- 만약 수정사항이 있다면 local 웹사이트를 컨 상태로 파일 수정 후,

```
cd mysite
```

```
bundle exec jekyll serve
```

```
cd mysite
```

```
git add .
```

```
git commit -m "Fix routing and permalink structure"
```

```
git push
```



Thank you