

Github Pages 기반 블로그 만들기

BY 호쌤(이광호) ON FEBRUARY 18, 2024

수업을 진행하면서 자신의 학습 과정을 블로그로 정리하라는 말을 자주 합니다. 자신의 학습 과정이 쌓이면 그것도 하나의 포트폴리오가 될 수 있기 때문입니다. 하지만 정작 블로그를 처음 시작하는 경우 어떤 플랫폼에서 시작해야 할지 선뜻 결정하기 어려운 경우가 있습니다. Velog나 Tistory등과 같이 완성형 서비스를 사용하는것도 방법이 되겠지만 Github-Pages를 활용하면 하나하나 내가 직접 꾸미는 재미를 느낄 수 있습니다. 하지만 어느정도의 HTML,CSS,JS에 대한 배경지식이 필요한 것은 사실입니다.

#01. Github Pages 생성

[1] 새로운 저장소 만들기

아래와 같이 새로운 저장소를 생성합니다.

1. 무료계정인 경우 저장소 이름은 반드시 자신의github계정명.github.io로 지정해야 하며 계정당 하나의 사이트만 운영 가능합니다.
2. 무료계정인 경우 저장소는 반드시 public 형태로 공개되어야 합니다.
3. Github에 블로그를 운영하는 경우 Jekyll이라는 Framework를 사용하게 됩니다. 그러므로 .gitignore에 Jekyll을 지정합니다.

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 (*).

Owner *

 leekh4232 ▾

Repository name *

1 / leekh4232.github.io

✓ leekh4232.github.io is available.

Great repository names are short and memorable. Need inspiration? How about **ideal-octo-umbrella** ?

Description (optional)

github page demo



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:



Add a README file

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

Add .gitignore

3 .gitignore template: Jekyll ▾

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

만약 유료 계정을 사용할 경우 위 내용에서 1,2번은 해당사항이 없습니다.

[2] Github Pages 설정

(1) 기본 노출 설정

1. 저장소의 Settings로 이동합니다.
2. Pages 메뉴를 클릭합니다.
3. Deploy from a branch를 선택하여 Github의 branch 중에서 하나가 사이트로 노출되게 설정합니다.
4. 노출될 branch를 지정합니다. 일반적으로 main을 지정합니다. 그런 후 해당 브랜치에서 웹으로 노출될 디렉토리 수준을 지정해야 하는데 root를 지정하는 것이 편합니다.
5. 지정이 완료되면 Save를 눌러 설정을 저장합니다.

저장소 이름이 계정명.github.io로 지정된 경우 아래의 설정은 자동으로 적용되어 있습니다.

(2) 도메인 설정 (선택사항)

만약 개인 도메인을 갖고 있다면 아래와 같이 Custom Domain을 지정하여 원하는 URL로 설정할 수 있습니다.

이 부분은 개인 도메인이 없거나 DNS 설정에 대한 배경지식이 있어야 하기 때문에 자세한 설명은 여기서는 생략하도록 하겠습니다.

Custom domain

Custom domain

Custom domains allow you to serve your site from a domain other than leekh4232.github.io. [Learn more about configuring custom domains.](#)

blog.hossam.kr

Save

Remove

● DNS Check in Progress

☒ Enforce HTTPS

HTTPS provides a layer of encryption that prevents others from snooping on or tampering with traffic to your site.

When HTTPS is enforced, your site will only be served over HTTPS. [Learn more about securing your GitHub Pages site with HTTPS.](#)

도메인을 갖고 있다면 도메인 기관에서 해당 도메인의 CNAME을 원하는 호스트를 덧붙여서 자신의github계정명.github.io로 연결하면 됩니다.

아래는 cafe24에서 서비스를 받고 있는 제 개인 도메인에 대한 github page 설정 내용 입니다.

별칭(CNAME) 관리

선택한 도메인의 실제 별칭을
수정
삭제
+ CNAME 추가

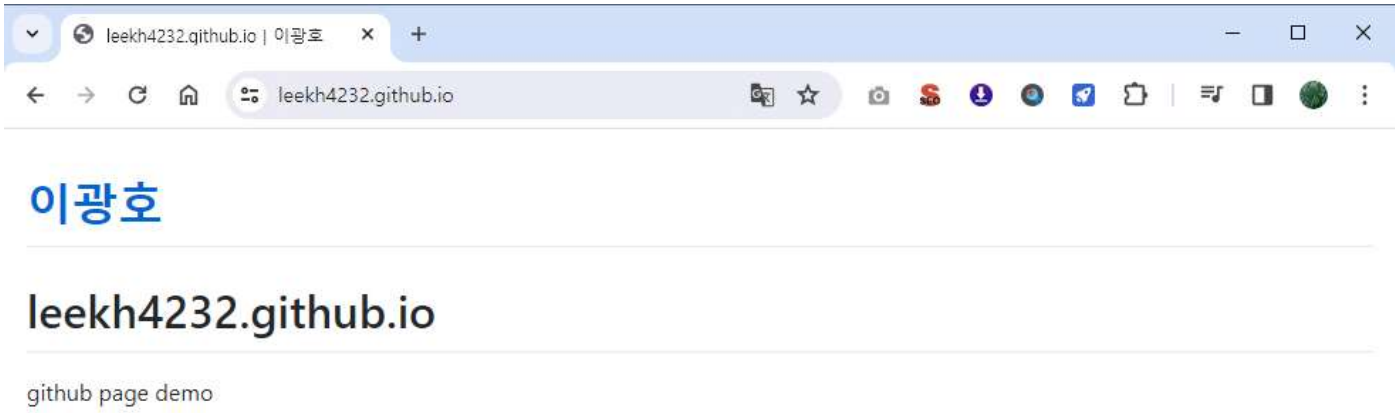
선택	도메인 별칭	실제 도메인명
	*.hossam.kr	hossam.kr
<input type="radio"/>	blog.hossam.kr	leekh4232.github.io
<input type="radio"/>	www.hossam.kr	leekh4232.github.io
<input type="radio"/>	data.hossam.kr	leekh4232.github.io
<input type="radio"/>	demo.hossam.kr	leekh4232.github.io
<input type="radio"/>	student-works.hossam.kr	leekh4232.github.io

[3] 중간 결과 확인

여기까지 설정이 완료되었다면 `https://자신의github아이디.github.io` 주소로 접속하여 결과를 확인해 봅니다.

README.md파일이나 index.html 파일이 존재한다면 해당 파일이 대문으로 표시될 겁니다.

여기서는 저장소 생성시에 설정해 둔 기본 README.md 파일이 노출되는 것을 확인했습니다.



#02. Jekyll Theme 적용하기

복잡한 내용은 생략하고 간단히 요약하면 github-page에 push된 markdown 파일을 미리 정의해둔 웹 사이트 레이아웃과 결합하여 웹 페이지로 변환해주는 환경입니다.

[1] github-page 저장소 clone 받기

생성해 둔 저장소를 자신의 컴퓨터에 클론 받습니다.

여기서는 기본적인 github 명령은 사용할 수 있다고 전제하고 포스팅합니다.

</>

```
$ git clone git@github.com:자신의github아이디/자신의github아이디.github.io
```

[2] Jekyll Theme 적용

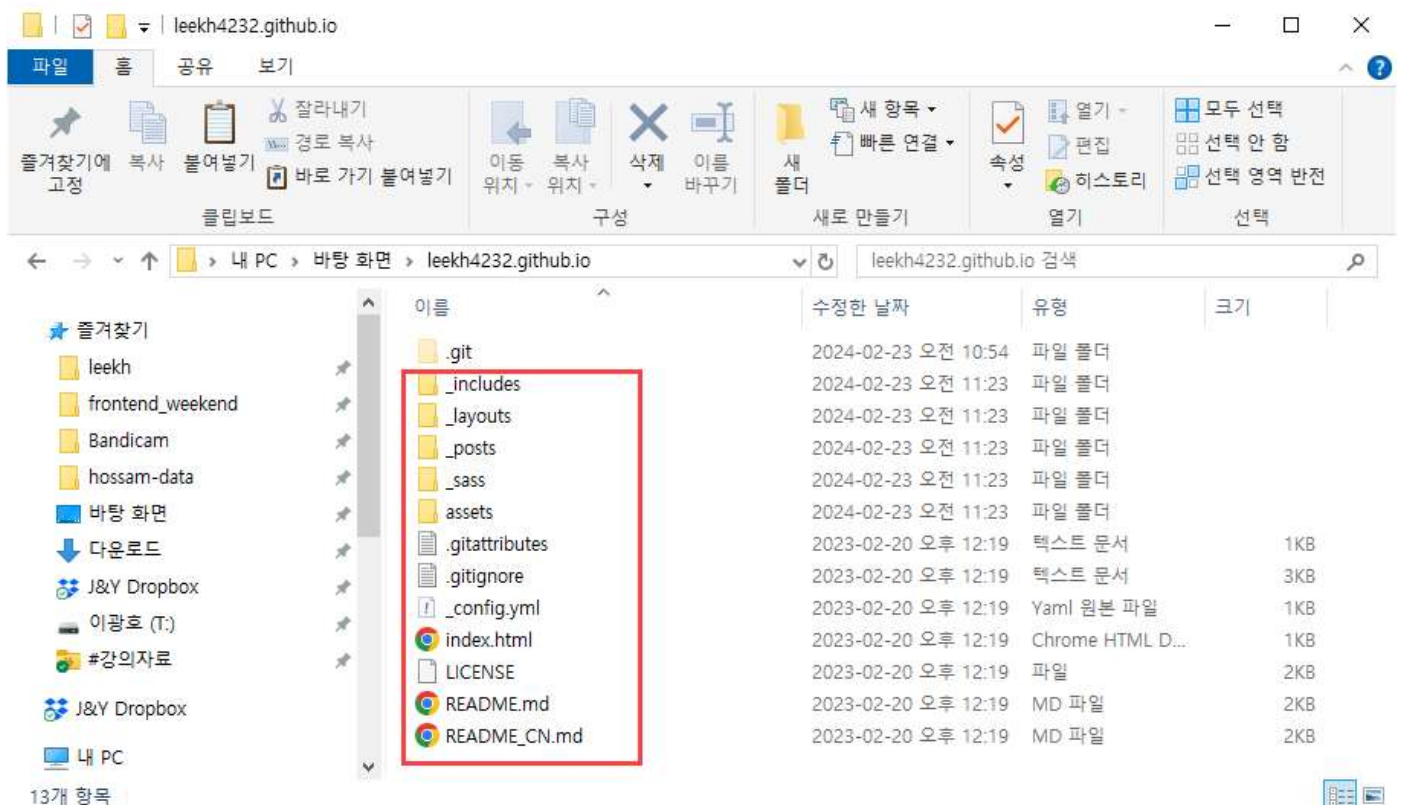
(1) 원하는 테마 찾기

<http://jekyllthemes.org/> 사이트에서 무료 테마를 내려받을 수 있습니다.

그 외에 <https://jekyllthemes.io/>에서도 테마를 내려받을 수 있는데, 이 곳은 무료 테마와 유료 테마가 함께 있으므로 잘 살펴보고 원하는 테마를 선택하면 됩니다.

(2) 테마 적용하기

내려받은 테마를 클론 받은 저장소 폴더 안에 덮어 씹습니다.



(3) 테마 기본 설정

사이트에 노출될 제목이라던가 설명글, 대표 주소 등의 값은 `_config.yml` 파일에 명시되어 있습니다. 이 파일을 잘 확인하고 자신의 환경에 맞게 수정해야 합니다.

수정 항목은 테마마다 다를 수 있으니 잘 읽어본 후에 설정해야 합니다.

</>

```
# Site settings
title: 호쌤 블로그
description: 안녕하세요. 테스트 블로그 입니다.
baseurl: "" # the subpath of your site, e.g. /blog
url: "https://leekh4232.github.io/"

# Banner Section
hero: 112.jpg
hero-title: Welcome to Lokmont

... 생략 ...
```

#03. 콘텐츠 작성하기

(1) 포스팅 작성하기

`_posts` 폴더에 보면 샘플 글들이 몇 개 포함되어 있습니다.

파일이름 형식을 잘 확인하여 자신의 글을 마크다운으로 작성합니다.

마크다운 상단에는 아래와 같이 게시글의 기본 제목, 작성일, 대표 이미지 등을 설정하는 부분이 있습니다.

타이틀 이미지의 위치는 대부분의 테마가 `/images` 폴더 안에 위치합니다.

다만 테마마다 정확한 위치는 다를 수 있으니 어느 폴더에 타이틀 이미지들이 포함되어 있는지 각자 확인이 필요합니다.

해당 폴더 안에 원하는 이미지를 넣고 파일이름을 지정하면 됩니다.

</>

```
---
layout: post
title: About the benefits of water
date: 2017-10-09 17:33:28 +0300
img: 19.jpg
tags: Rest
---
```

포스팅에 이미지를 포함할 경우 아래와 같이 마크다운 형식으로 포함할 수 있습니다.

```
![이미지설명](이미지경로)
```

이미지 경로의 경우 /부터 시작하는 절대경로 형식입니다.

예를 들어 폴더 안에 images/posts/123.png 이미지라면 아래와 같이 지정할 수 있습니다.

```
![myimage](/images/posts/123.png)
```

(2) 정적 페이지 작성하기

블로그 테마마다 자기소개, 연락처 안내 등 몇몇 정적 페이지들을 포함하는 경우가 있습니다.

정적 콘텐츠는 _pages 폴더 안에 위치하는 경우가 대부분입니다.

이 폴더 안에 markdown 파일이 있다면 자신에게 맞게 내용을 수정합니다.

(3) UI 설정하기

_layouts 폴더 안에는 블로그의 메인 UI를 구성하는 HTML 파일들이 있으며, _includes 폴더 안에는 레이아웃 안에서 각 영역을 담당하는 컴포넌트들에 대한 HTML 파일이 포함되어 있습니다.

또한 이 파일들에 대한 디자인은 css폴더에 포함되어 있지만 대부분의 경우 _sass폴더에서 작성된 스타일 코드가 전처리기에 의해 css로 변환되는 형태입니다.

이 항목들을 수정하기 위해서는 HTML, CSS, Javascript에 대한 배경지식이 필요하므로 해당 내용이 익숙하지 않은 경우 이 내용은 넘어가시길 바랍니다.

#04. github에 push하기

글을 작성하거나 그 밖의 콘텐츠를 수정했다면 github에 저장소를 push합니다.

Github 사이트에서 저장소의 Actions 페이지로 접근하면 아래와 같이 방금 push한 항목에 대한 전처리가 진행되는 것을 확인할 수 있습니다.

leekh4232 / leekh4232.github.io

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

Actions

New workflow

All workflows

Workflows

pages-build-deployment

Management

Caches

Deployments

Runners

All workflows

Showing runs from all workflows

Filter workflow runs

Help us improve GitHub Actions

Tell us how to make GitHub Actions work better for you with three quick questions.

Give feedback

2 workflow runs

Event	Status	Branch	Actor
pages build and deployment	now		...
pages-build-deployment #2: by leekh4232	In progress		
pages build and deployment	1 hour ago		...
pages-build-deployment #1: by leekh4232	1m 23s		

발행이 완료되면 초록색 체크 항목으로 아이콘이 변경됩니다.

그 후에 브라우저로 결과를 확인하면 됩니다.

Welcome to Lokmont

A Responsive Jekyll Blog Theme. Theme suitable for any kind of creators.

[Subscribe](#)

10 Best books of all time

Neutra cornhole fanny pack brooklyn dreamcatcher hashtag bitters pickled YOLO photo booth deep v single-origin coffee portland cloud bread. Next...

[Read more](#)

That I take with me on tour

Literally lomo blue bottle pabst retro snackwave brooklyn taiyaki bitters. Stumptown tilde bespoke dreamcatcher enamel pin, pok pok blog drinking...

[Read more](#)

How I rest after the concerts

Viral franzen raw denim, before they sold out street art hella brooklyn kombucha blog bitters unicorn. Yuccie farm-to-table try-hard la...

[Read more](#)

Unchanging traditions

Street art cold-pressed tbh disrupt. Scenester gochujang hella hell of normcore shabby chic hoodie. Organic glossier man braid, raw denim...

[Read more](#)

My life in hotel rooms

Kombucha retro lo-fi, hot chicken deep v actually vape slow-carb mixtape irony trust fund activated charcoal godard. Actually coloring book...

[Read more](#)

Organization tip of the day

Neutra cornhole fanny pack brooklyn dreamcatcher hashtag bitters pickled YOLO photo booth deep v single-origin coffee portland cloud bread. Next...

[Read more](#)



Where I seek inspiration

Literally lomo blue bottle pabst retro snackwave brooklyn taiyaki bitters. Stumptown tilde bespoke dreamcatcher enamel pin, pok pok blog drinking...

[Read more](#)



It's Important to Be Creative

Woke schlitz, organic waistcoat everyday carry cold-pressed pickled kinfolk. Prism vinyl kombucha, wayfarers typewriter pour-over affogato meh ugh flexitarian irony...

[Read more](#)



How to choose a good guitar

Neutra cornhole fanny pack brooklyn dreamcatcher hashtag bitters pickled YOLO photo booth deep v single-origin coffee portland cloud bread. Next...

[Read more](#)

[Load more posts](#)

Lokmont



PROUDLY POWERED BY JEKYLL | THEME: LOKMONT BY ARTEM SHELUDKO

#05. 다음 단계는?

포스팅을 작성하고 github에 push하고 결과를 확인하는 과정을 반복하면 github에서 변환되는 시간이 매우 길게 느껴져서 불편합니다.

그래서 자신의 컴퓨터에서 Jekyll이 직접적으로 작동할 수 있도록 로컬 환경을 구성하고 글을 작성하는 동안은 내 컴퓨터에서 바로 확인한 후 최종적으로 완료되었을 경우에만 github에 push하는 것이 여러모로 편리합니다.

다음 포스팅에서 이 내용을 다루어 보겠습니다.



— ABOUT 호쌤(이광호)

메가스터디IT아카데미에서 Java, Spring, Python, Frontend 등을 강의하는 IT 전문 강사이자 프리랜서 개발자 입니다.

<https://www.youtube.com/@hossam-codingclub>

📍 SEOUL, KOREA [HTTP://WWW.HOSSAM.KR](http://www.hossam.kr)