

Week 4

Javascript

[Basic & Data Fetching]

01. What's Javascript?

02. Javascript Basic

03. DOM 조작

04. JSON

05. Server Request

06. Data Fetching

07. GitHub Hosting

01

What's Javascript?



Javascript

웹 페이지를 Interactive하게 동작하도록 도와주는 언어

Javascript 엔진



Internet Explorer
Chakra



Chrome
V8



Firefox
SpiderMonkey

ECMAScript

ECMAScript는 ECMA International이라는 비영리 단체에서 만든 표준화된 프로그래밍 언어

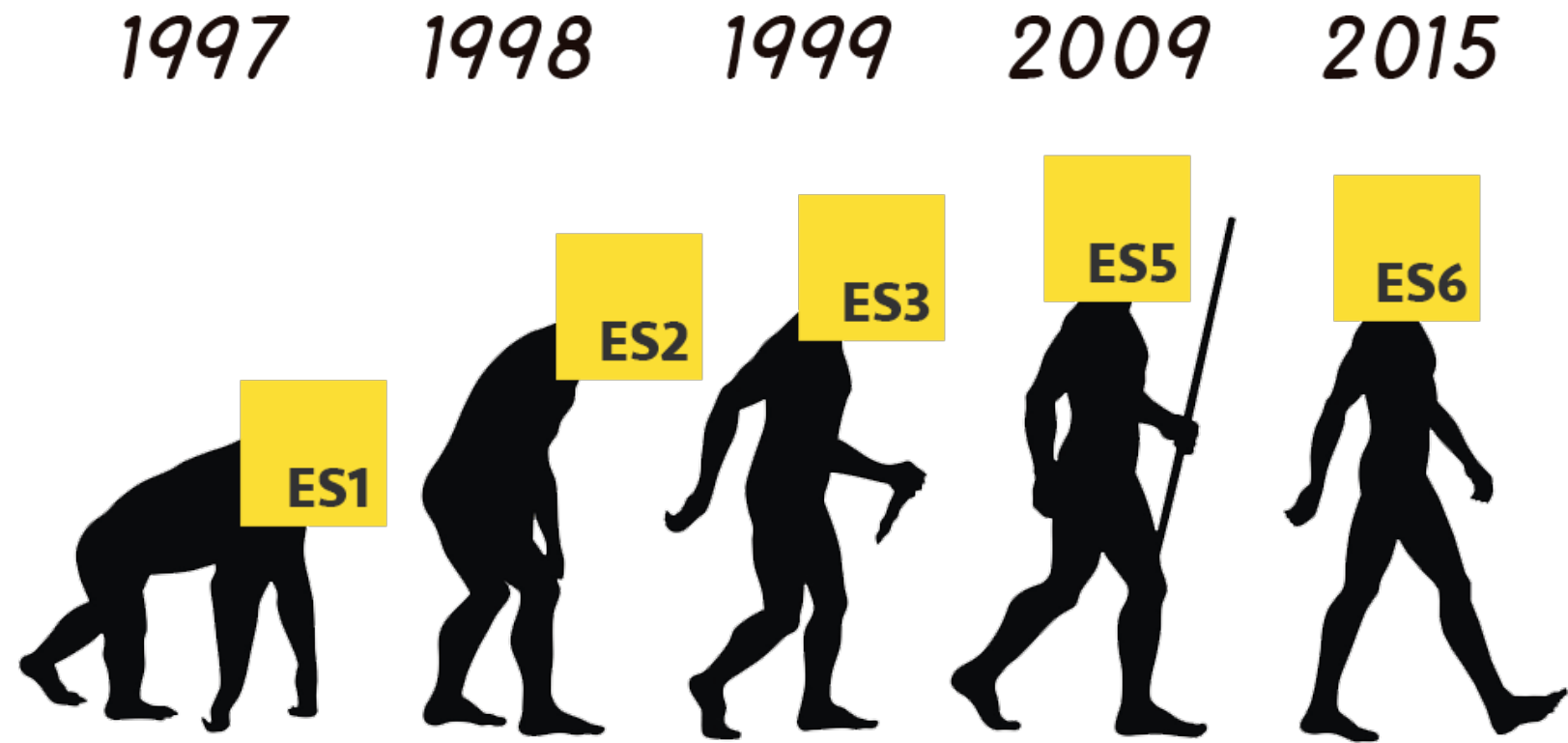
Google Chrome이 등장하기 전에는 Internet Explorer가 웹 브라우저 시장의 90% 이상을 가지고 있었고, ECMAScript의 표준을 무시하고 자신들이 표준이라며 Javascript의 생태계를 파괴하였습니다.

하지만 Chrome의 강력한 Javascript 엔진인 V8이 등장하며 Javascript 생태계는 안정되었습니다.



ECMAScript





1997년부터 시작된 ECMAScript는 2015년에 가장 큰 변화를 겪습니다.

Javascript를 사용하면서 정말 많이 들어보셨을 ES6(ES2015)가
2015년의 ECMAScript 표준입니다.



Javascript 실습 시작!!

VSCode를 열어주세요 ~

02

Javascript Basic

변수 선언

var, let, const를 사용하여 javascript 변수를 선언합니다.



```
let name = "Youngkwon Kim";  
let from = "경기도 부천시";  
let age = 26;  
  
const bloodType = "B";
```

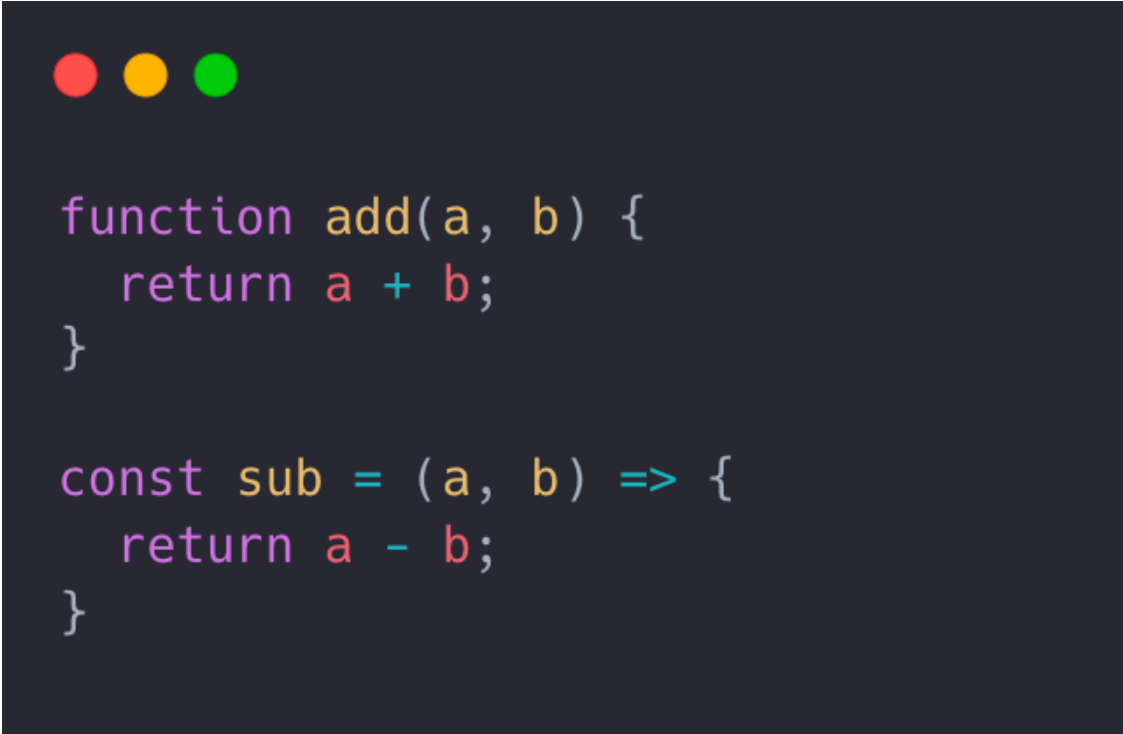
var: 사용하지 않는다

let: 일반적인 변수 선언에 사용한다

const: 한번 할당된 값이 절대 변하지 않을 경우 사용한다

함수 선언

function, arrow-function을 사용하여 javascript 함수를 선언합니다.



```
function add(a, b) {  
  return a + b;  
}  
  
const sub = (a, b) => {  
  return a - b;  
}
```

기능적으로는 완전히 동일합니다 :)

배열 선언


여러개의 값을 하나의 변수에 보관하고 싶을때 배열을 사용합니다.



```
let groupMember = ['김영권', '윤선영', '김민규', '김진우', '이상민', '이수은', '변주현', '유하린'];  
const weekend = ['SAT', 'SUN'];
```

객체 선언

‘구분되는’ 여러개의 값을 하나의 변수에 보관하고 싶을때 배열을 사용합니다.



```
const youngkwon = {  
  name: 'Youngkwon Kim',  
  age: 26,  
  lion: ['9th', '10th']  
}
```

출력 및 추가기능



```
console.log("출력!!!");
```

```
alert("경고 메시지");
```

```
const isConfirmed = confirm("잘 따라오고 계신가요~?");
```

변수 재할당 & 조건문과 반복문



```
let age = 26;

if (age > 19) {
  console.log("어른입니다");
}

for (i = 1; i < 6; i++) {
  age = age + i
  console.log(`${i}년 후 나이는 ${age}입니다.`);
}
```

쉬는시간 !

03

DOM 조작



Document Object Model

HTML, CSS, JS 코드의 결과물로서 브라우저에 나타나는 화면

Web Browser Rendering



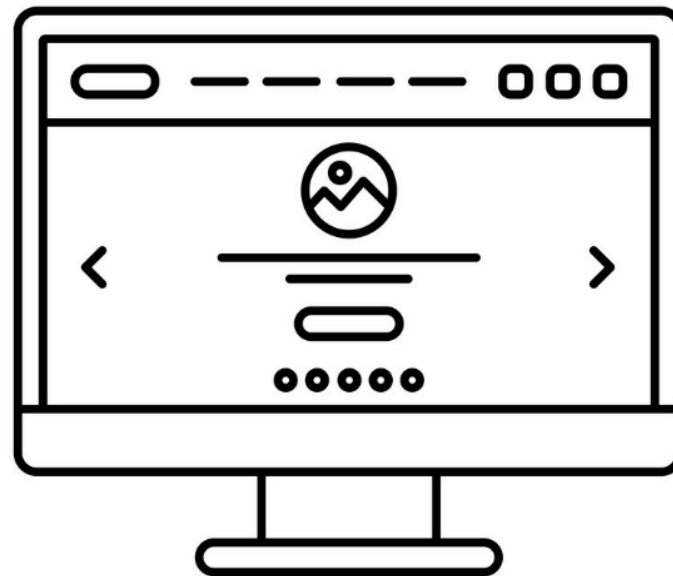
HTML + CSS + JS



WEB Page

Interactive Web

Javascript

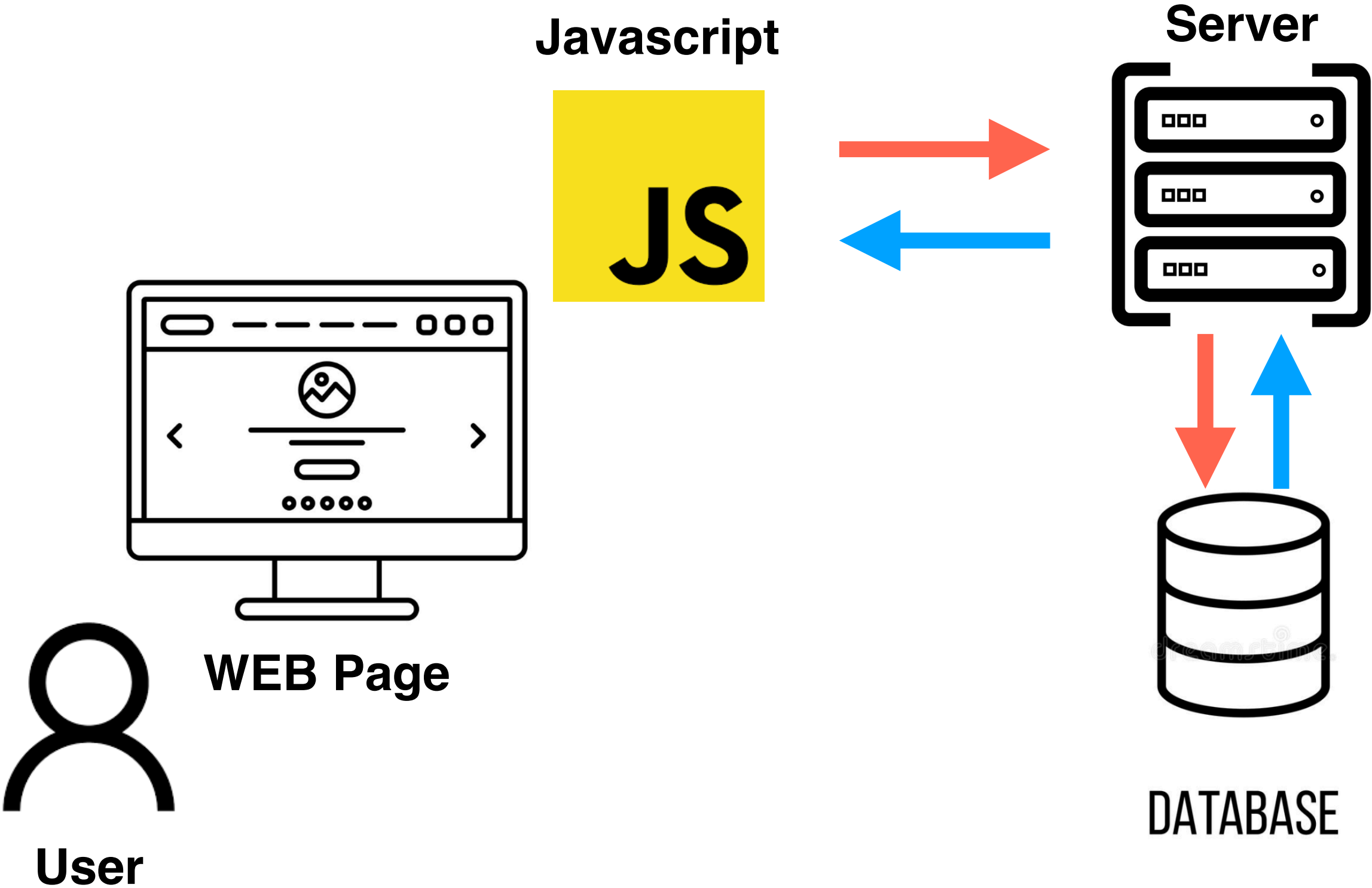


WEB Page



User

Dynamic Web



DOM Element



// tag name으로 접근

```
document.querySelector( 'nav' );
```

// id로 접근

```
document.querySelector( '#nav-logo' );
```

// class name으로 접근


```
document.querySelector( '.thumbnail' );
```

DOM Element Style



```
document.querySelector('#name').style.color = 'black';  
document.querySelector('#name').style.fontSize = '20px';  
document.querySelector('#name').style.backgroundColor = 'blue';
```

DOM Element Content



```
document.querySelector('#introduction').innerHTML = "안녕하세요! 반갑습니다";  
  
const age = 26;  
  
document.querySelector('#age').innerHTML = `내 나이 ${age},, 조금 많나..?`;  
  
document.querySelector('#empty-space').innerHTML = "";
```


Javascript 불러오기



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="stylesheet" href="./asset/css/font.css" />
    <link rel="stylesheet" href="./asset/css/style.css" />

    <script src="./asset/js/main.js"></script>
    <title>4th-Session</title>
  </head>
```



```
<script src="./asset/js/main.js"></script>
```

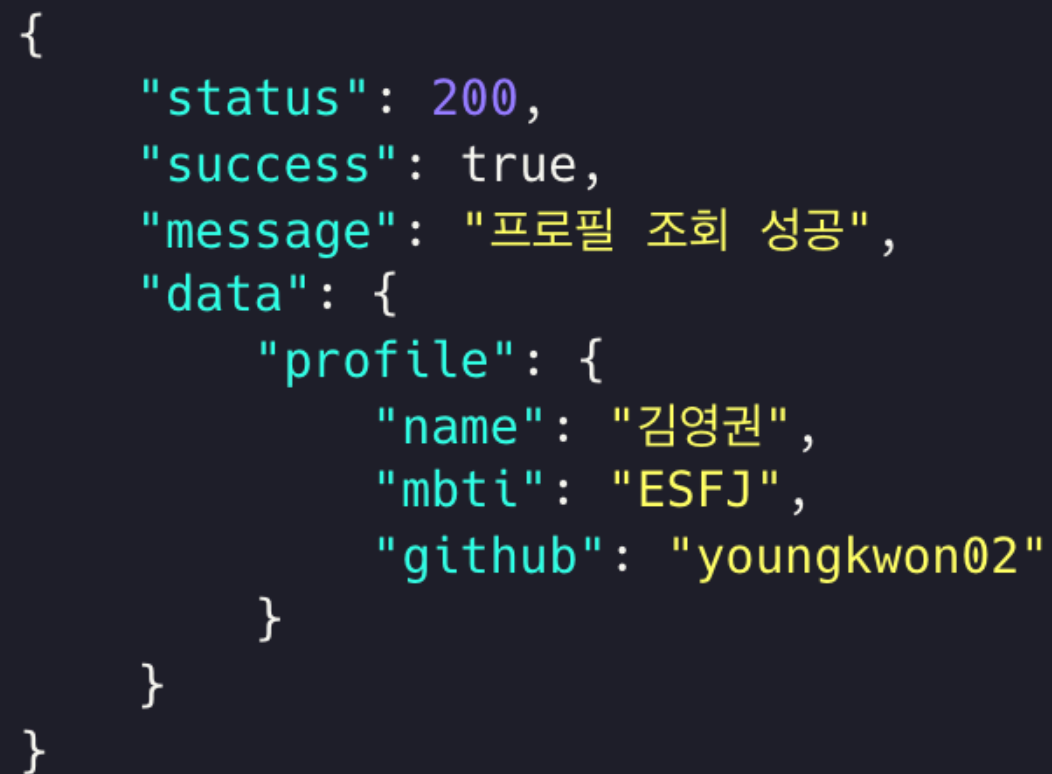
실습 + 쉬는시간 !

04

JSON

(JavaScript Object Notation)

JSON



```
{
  "status": 200,
  "success": true,
  "message": "프로필 조회 성공",
  "data": {
    "profile": {
      "name": "김영권",
      "mbti": "ESFJ",
      "github": "youngkwon02"
    }
  }
}
```

05

Server Request

Server Status Code

200 (OK, 성공)

400(Bad Request, 잘못된 요청)

401(Unauthorized, 권한 없음)

403(Forbidden, 금지됨)

404(Not Found, 찾을 수 없음)

500(Internal Server Error)

<https://github.com/KakaoFarm/Youngkwon-TIL/blob/main/Article/HTTP-Status-Code.md>

HTTP Request Method

GET : 서버에 특정 데이터를 요청하여 원하는 데이터를 얻습니다

POST : 서버에 특정 데이터를 추가 (새로 생성)할 것을 요청합니다

PATCH : 특정 데이터의 특정 값을 변경할 것을 요청합니다

PUT : 특정 데이터를 전체적으로 변경할 것을 요청합니다

GET

- Request Body가 존재하지 않습니다
- Path parameter, Query string을 통해 데이터를 전달합니다.



```
https://asia-northeast3-likelion-js-practice.cloudfunctions.net/api/photo/1
```

Path Parameter



```
https://asia-northeast3-likelion-js-practice.cloudfunctions.net/api/member?name=김영권
```

Query String

POST

- Request Body가 존재하지 않습니다
- JSON 형태의 Request Body를 서버에 전달합니다



```
{  
  "content": "안녕 영권아~",  
  "receiverName": "김영권",  
}
```

Request Body

06

Data Fetching

**[https://github.com/LikeLion-at-CAU-10th/
SESSION/tree/main/session-04/full-source](https://github.com/LikeLion-at-CAU-10th/SESSION/tree/main/session-04/full-source)**

07

GitHub Hosting


GitHub Hosting (Deploy)

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 *

 youngkwon02 ▾


 /

youngkwon ✓


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

Description (optional)

Youngkwon's Portfolio Web Page

☒  **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.](#)

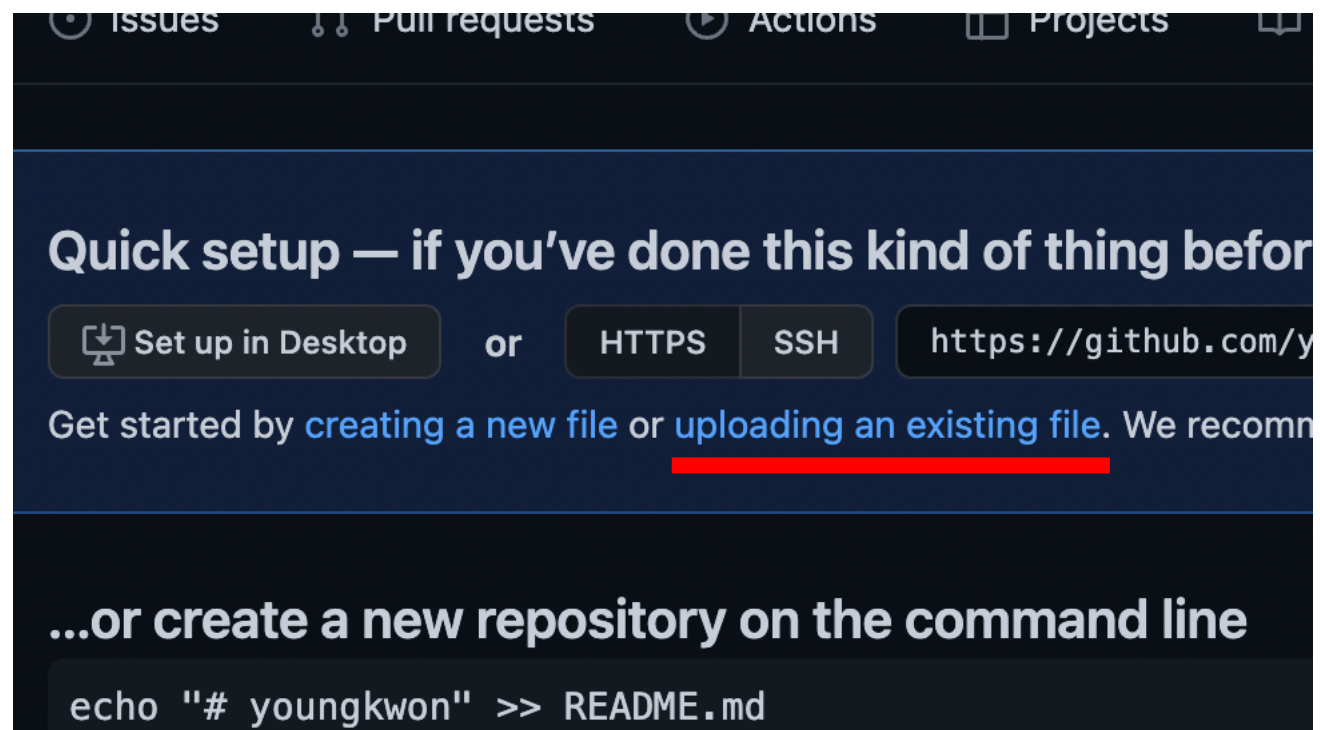
Add .gitignore

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

.gitignore template: None ▾

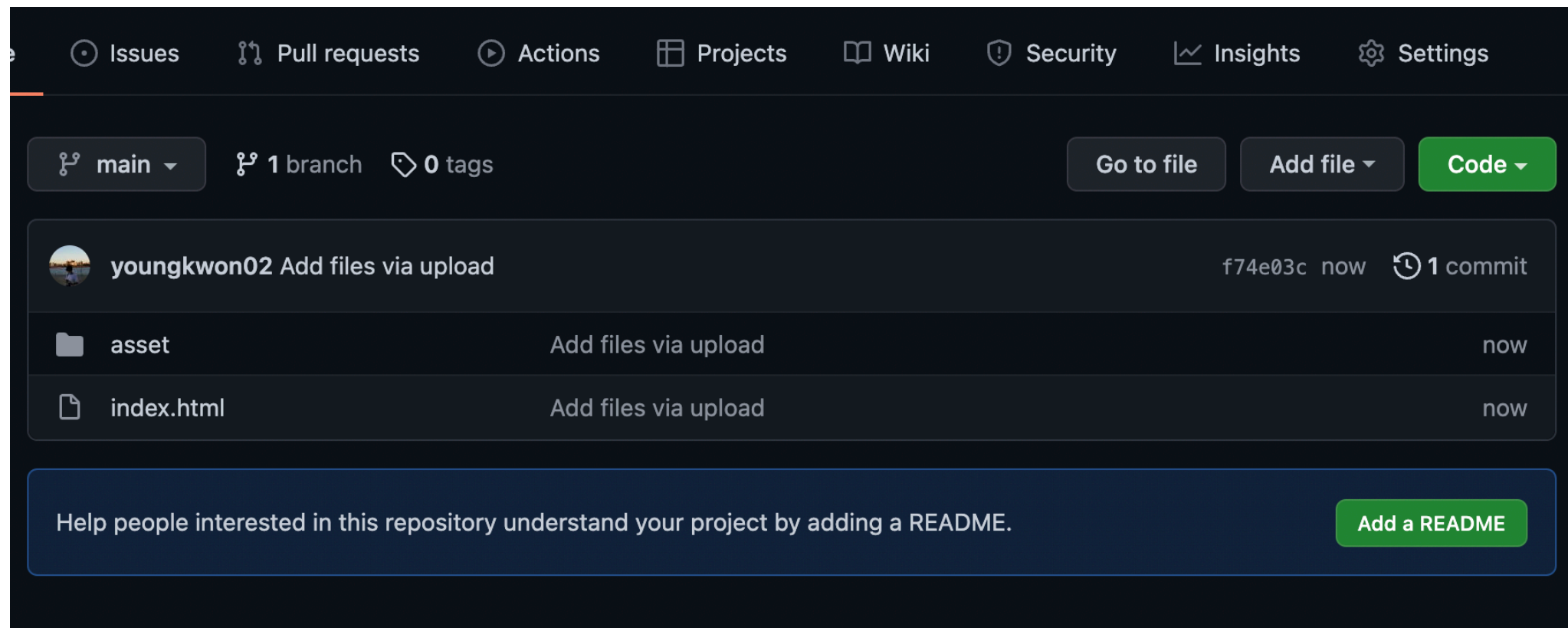
레포지토리 이름은 추후
URL에 포함됩니다!!

GitHub Hosting (Deploy)



**Bash shell 등의 CLI 환경에서
파일을 업로드 하셔도 좋습니다 :)**

GitHub Hosting (Deploy)

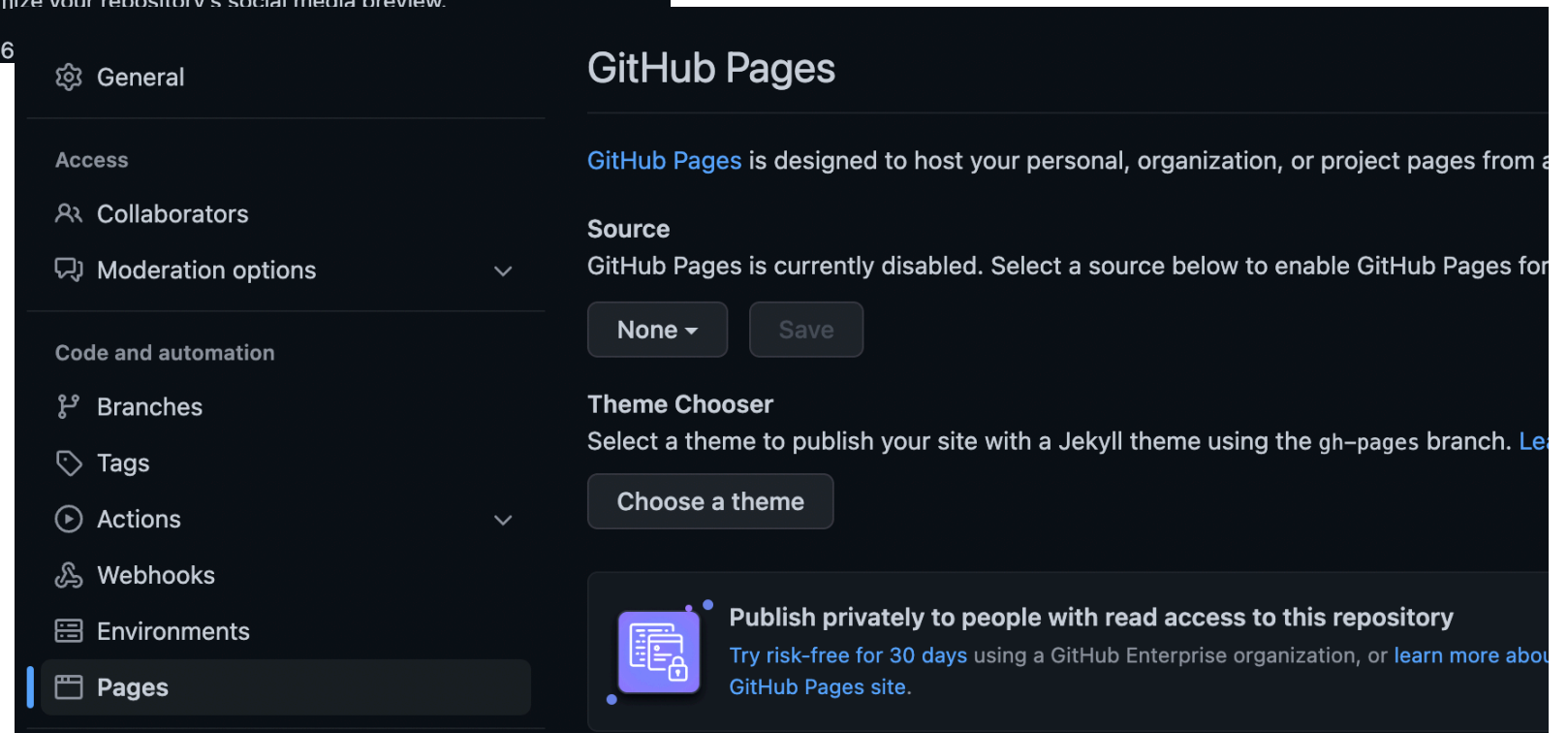
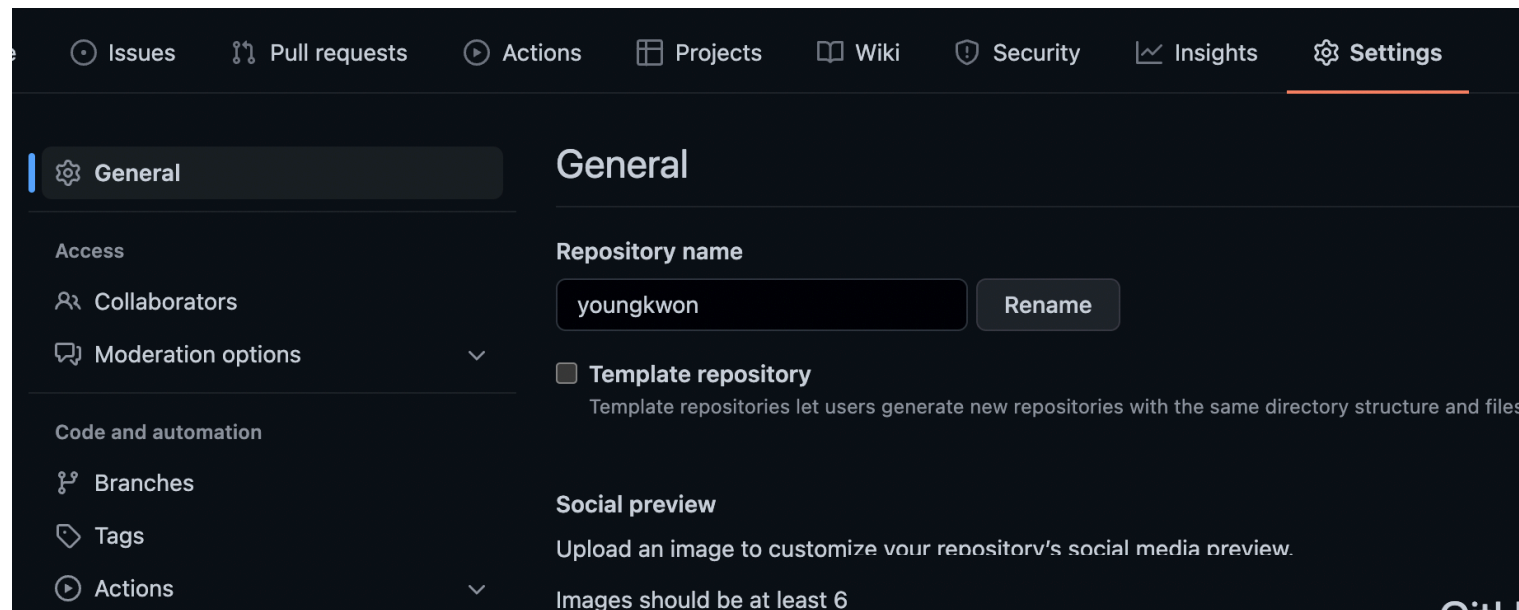


성공적으로 파일이 업로드 된 모습
(index.html 파일은 최상단에 위치해야 합니다!!)

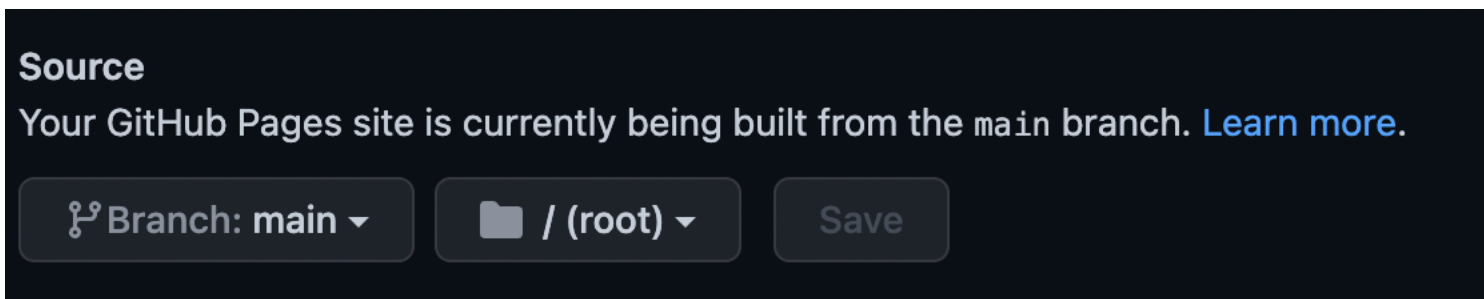
GitHub Hosting (Deploy)

Settings Tab에 들어가서

Pages 선택

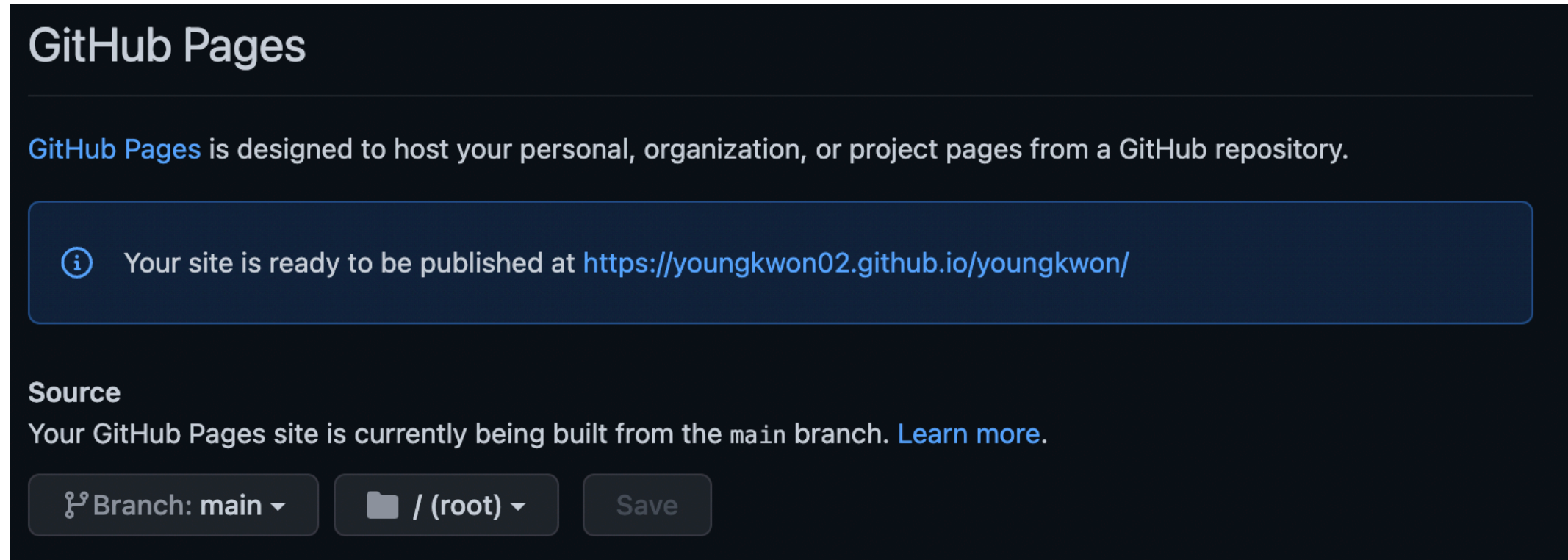


GitHub Hosting (Deploy)



Branch: main
Directory: /(root) 선택하고
Save!

GitHub Hosting (Deploy)



배포 완료!
반영되는데 시간이 조금 걸릴 수 있어요~~

과제 설명

Thank You