

# Session 1

개발환경 설정 & HTML,CSS 기초

**NEXT X LIKELION** 강동혁

# 목차

0. 개발 세션 개괄
1. 개발 환경 설정
2. 왜 웹개발인가?
3. 웹의 구조
4. HTML,CSS 기초
5. Github 사용해보기

# | 12기 NEXT 첫 세션!



와 개발이당-!

# | 세션 예고



부디 10시 반엔 끝낼 수 있도록 협조 부탁드립니다...



# 개발 세션 개발

고려대학교 스타트업 스테이션

매주 월/목 19시 30분 ~ 22시 30분

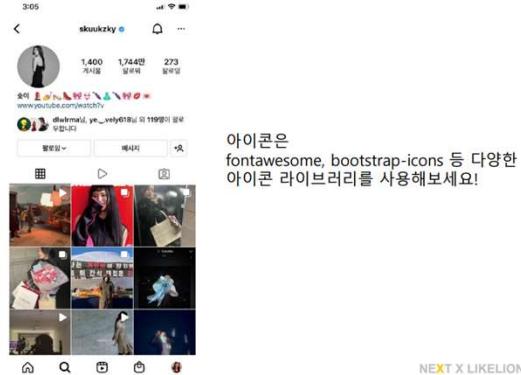


# 과제

## 1. 세션 복습 과제

### 과제: 인스타그램 클론코딩

실습3 끝더



아이콘은  
fontawesome, bootstrap-icons 등 다양한  
아이콘 라이브러리를 사용해보세요!

### 과제

할 수 있다!

#### 제출 방법

##### 1. 글자수 세기 프로젝트

- 본인 레포지토리에 git push한 링크
- runserver하여 단어수 세는 기능 화면 녹화

##### 2. ★개인 포트폴리오 사이트★

- 본인 레포지토리에 git push한 링크
- runserver하여 화면 녹화

제출 기한 다음 세션(2023/03/30/목)

기한 안에 본인 Repository에 Push한 후 댓글로 링크 남기기

# | 과제

## 2. 개발 환경 세팅

Ex. 파이썬 깔아오기, Django 프로젝트 세팅해오기 등등



# 과제

## 3. 추가 영상 학습

아주아주 가끔

The screenshot shows the homepage of codecademy. At the top, there is a navigation bar with links for Catalog, Resources, Community, Pricing, Career Center (new), For Teams, a search icon, Log In, and Sign Up. Below the navigation bar, a large card for the 'Learn HTML' course is displayed. The card features a 'Free Course' badge, the title 'Learn HTML' in large bold letters, a description stating 'Start at the beginning by learning HTML basics – an important foundation for building and editing web pages.', a rating of '4.6 ★★★★☆ 33,375 ratings', a 'Start' button, and a statistic showing '3,003,620 learners enrolled'. Below this card, there is a summary section with four items: 'Skill level Beginner', 'Time to complete 7 hours', 'Certificate of completion Included with paid plans', and 'Prerequisites None'.

## | 학회원 1학기 룰

**개발 세션 22회, 창업 세션 6회, 총 28회**

→ 지각 0.5점 결석 2점 (특별한 사유 있을 시, 임원진 연락 필수)

→ 과제 지각 제출 0.5점, 과제 미제출 or 표절 1점

→ 총 5점일 경우 수료 불가

**주요 행사 필참**

| 당부 말씀 1..



ChatGPT

쓰셔도 됩니다 !!

## | 당부 말씀 1..

자자 GPT야.

우리 이번 NEXT의 과제는  
~~~~  
이거야

코드 짜줘 ㅎ ㅎ!



**HTML 코드를 짜는 중이야**

**Button 객체를 가로로 돌리는  
코드가 뭔지 궁금한데**

**알려줄래?**

# 당부 말씀 1..

A screenshot of a web-based AI interface showing generated code. The code consists of a CSS style block and an HTML div:

```
.rotate {  
    transform: rotate(90deg);  
}  
</style>  
  
<div class="rotate">  
    이 div는 90도 회전합니다.  
</div>
```

Below the code, a message says "AI가 생성한 코드입니다. 신중하게 검토하고 사용하세요. FAQ의 자세한 정보." (This is AI-generated code. Please review it carefully and use it. See detailed information in the FAQ.)

The main text area contains a message: "위의 코드에서 `rotate` 클래스가 적용된 div 요소는 90도 회전합니다. 각도를 조정하여 원하는 만큼 회전시킬 수 있습니다. 이 방법은 HTML 요소를 가로로 돌리는 데 사용할 수 있습니다. 이 코드를 자신의 HTML에 적용해 보세요. 이것이 도움이 되었으면 좋겠습니다! 😊" (In the above code, the `rotate` class is applied to the div element, which rotates it 90 degrees. You can adjust the angle to rotate it as much as you want. This method can be used to rotate HTML elements horizontally. Try applying this code to your own HTML. If it was helpful to you, please let us know! 😊)

Below the message are standard social sharing icons and a page number indicator "1 / 30".

At the bottom, there are three buttons with questions: "CSS에서 transform 속성이란 무엇인가요?", "어떻게 요소를 세로로 돌릴 수 있나요?", and "transform: rotate() 외에 다른 함수도 있나요?".

Finally, there is a comment input field with placeholder text "무엇이든 물어보세요" and a character limit of "0/4000".

## Bing AI를 조금 더 추천합니다

NEXT X LIKELION

## | 당부 말씀 2..



세션 담당자를 적극 활용해주세요!

# 당부 말씀 3..

## 2024-NEXT-Assignment

NEXT 12기



### Sessions

- 03/04 - session1 개발 세션 개괄 + HTML/CSS 기초
- 03/07 - session2 CSS 심화
- 03/11 - session3 협업을 위한 git
- 03/14 - session4 프로그래밍 언어 & 파이썬 기초 문법
- 03/18 - session5 파이썬 심화 문법
- 03/21 - session6 캐시링 크롤링

- [NEXT 11th Assignment] Session 9 - 장고(django) CRUD  
#9 opened on Apr 4, 2023 by jinsim □ 26
- [NEXT 11th Assignment] Session 8 - 장고(django) CR + DB개념  
#8 opened on Mar 31, 2023 by JHLvJz □ 27
- [NEXT 11th Assignment] Session 5 - Selenium&Crawling  
#7 opened on Mar 29, 2023 by ssoty0514 □ 27
- [NEXT 11th Assignment] Session 7 - 장고(django) 기초  
#6 opened on Mar 27, 2023 by nymvmt □ 27
- [NEXT 11th Assignment] Session 4 - 파이썬 응용 1  
#5 opened on Mar 23, 2023 by ziweek □ 28
- [NEXT 11th Assignment] Session 3 - 파이썬 익숙해지기  
#4 opened on Mar 16, 2023 by Mo1oM □ 28
- [NEXT 11th Assignment] Session 2 - 프로그래밍 언어 및 Python 기초  
#3 opened on Mar 16, 2023 by ziweek □ 28
- [NEXT 11th Assignment] Session 1 - CSS 심화  
#2 opened on Mar 13, 2023 by Kim-Jiseong □ 28
- [NEXT 11th Assignment] Session 0 - 개발환경 세팅 & HTML, CSS 기초  
#1 opened on Mar 7, 2023 by Mo1oM □ 28

과제는 최대한 본인 힘으로!

## 당부 말씀 3..



개발은 과칠세삼

NEXT X LIKELION

## | 당부 말씀 3..

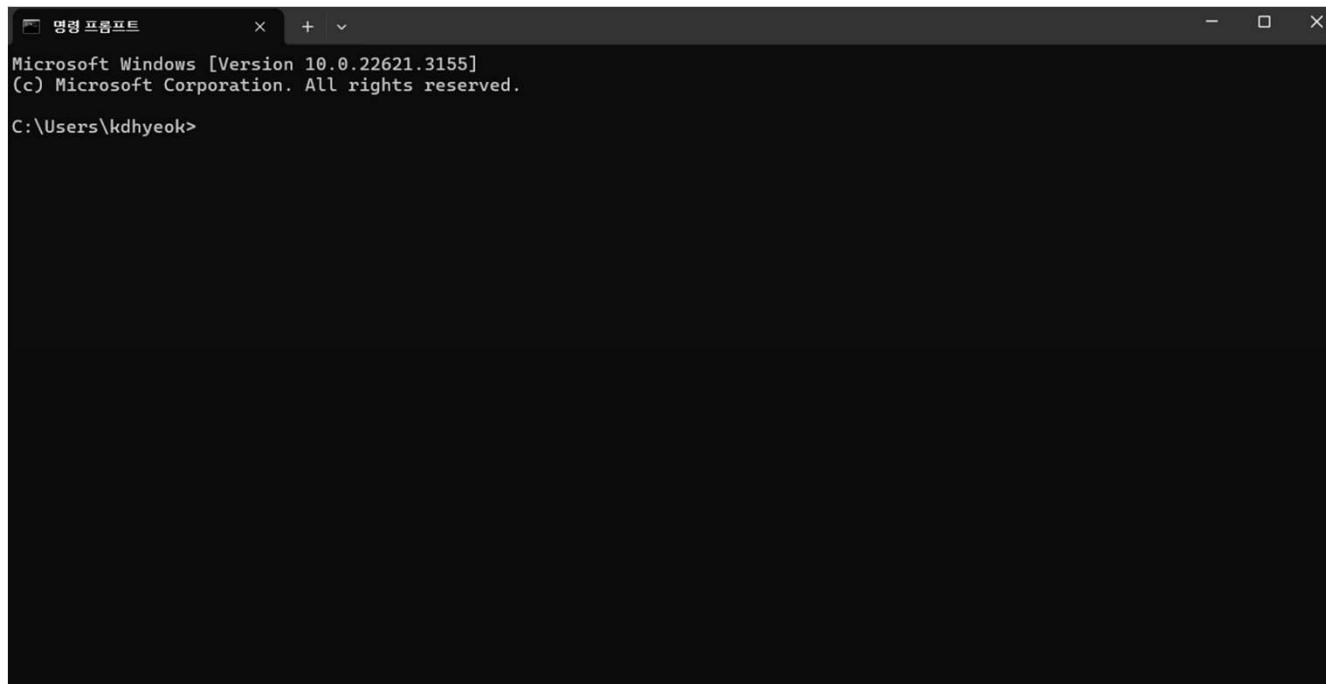


# 개발 환경 세팅

# | 계정명 영문 변경-윈도우

시작 메뉴에서 cmd를 검색해서 명령 프롬프트를 구동해줍니다.

C:\Users\ 뒤에 오는 이름이 영어면 앞으로 큰 문제 없습니다!



## | NEXT 폴더 생성 - 윈도우

1. 바탕화면에서 우클릭 후 '터미널에서 열기' 클릭  
+ 해당 버튼 없을 경우 검색 창에 cmd 입력 후 명령 프롬프트 구동  
이후 cd Desktop 명령어 입력

2. 아래 명령어 한 줄씩 입력

```
mkdir NEXT
```

```
cd NEXT
```

```
mkdir Session HW
```

```
for /L %i in (1,1,22) do mkdir Session\NEXT_Session_%i
```

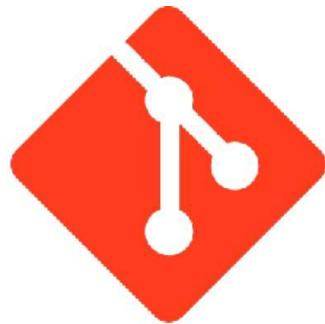
```
for /L %i in (1,1,22) do mkdir HW\NEXT_HW_%i
```

## | NEXT 폴더 생성 - 맥

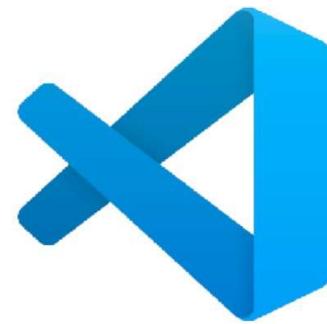
1. Spotlight에서 Terminal 검색 후 구동
2. “cd Desktop” 명령어로 바탕화면으로 이동
3. 아래 명령어 한 줄씩 입력

```
mkdir -p NEXT/{Session,HW}  
for i in {1..22}; do mkdir -p NEXT/Session/NEXT_Session_$i; done  
for i in {1..22}; do mkdir -p NEXT/HW/NEXT_HW_$i; done
```

# | 개발환경 설정



Git  
(버전 관리 시스템)

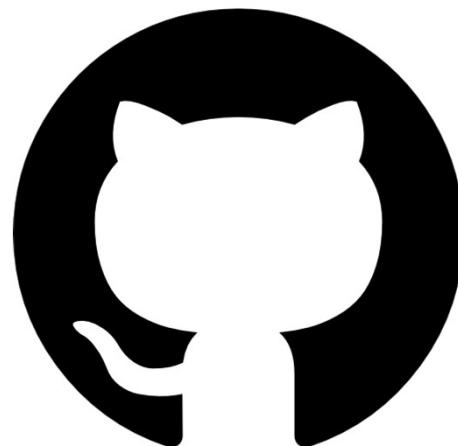


Visual Studio Code  
(텍스트 에디터)

## | GitHub 가입

**<https://github.com>**

가입하고 NEXT\_HW 레포지토리를 만들어봅시다!



# | Git 설치

macOS 유저는 **Terminal**에서 `git --version`으로 설치 유무 검색.  
Window 유저는 왼쪽 하단 검색창에서 **Git Bash** 프로그램 유무 검색.

이미 있으면 설치 x, 없으면 설치 ↗ ↘

ONLY Window!  
(Mac은 터미널에 이미 내장되어 있음)

# | Git 설치

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



Window 유저

## Download for Windows

[Click here to download](#) the latest (2.35.1) 32-bit version of Git for Windows. This is the most recent maintained build. It was released [about 1 month ago](#), on 2022-02-01.

### Other Git for Windows downloads

[Standalone Installer](#)

[32-bit Git for Windows Setup.](#)

[64-bit Git for Windows Setup.](#)

[Portable \("thumbdrive edition"\)](#)

[32-bit Git for Windows Portable.](#)

[64-bit Git for Windows Portable.](#)

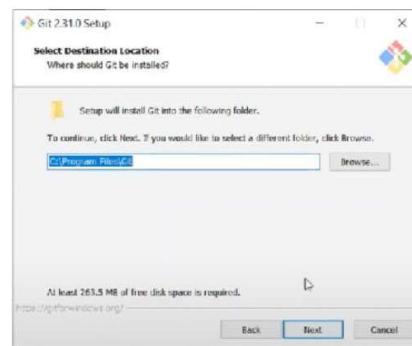
### Using winget tool

Install [winget tool](#) if you don't already have it, then type this command in command prompt or Powershell.

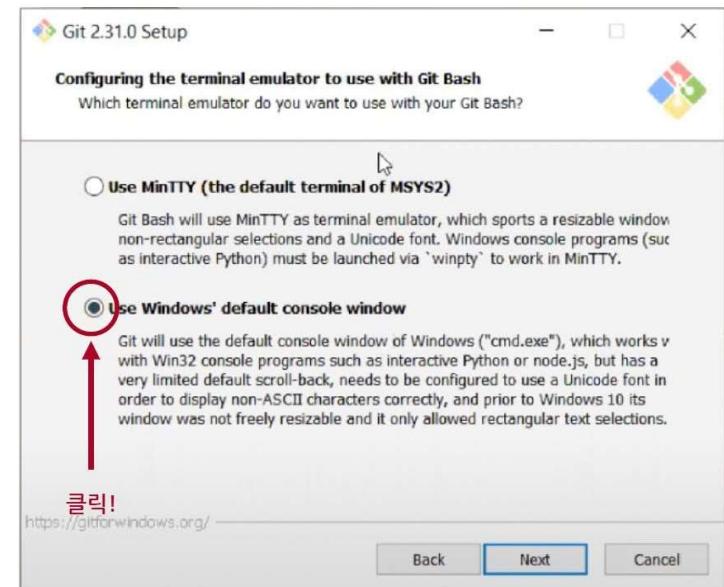
```
winget install --id Git.Git -e --source winget
```



# Git 설치

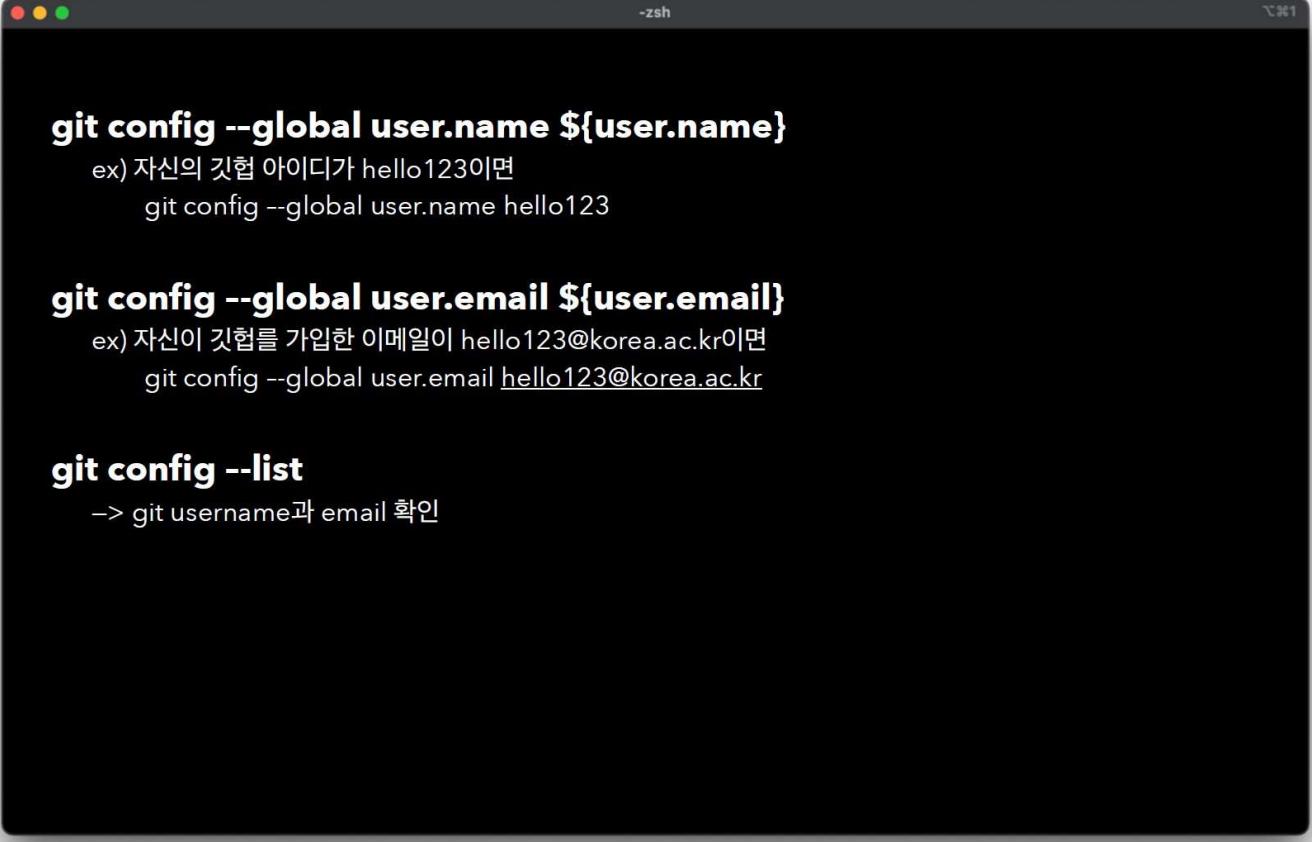


계속 next  
누르다가..



# | Git 초기 설정

맥은 Terminal,  
윈도우는 Git Bash에서



```
-zsh

git config --global user.name ${user.name}
ex) 자신의 깃헙 아이디가 hello123이면
    git config --global user.name hello123

git config --global user.email ${user.email}
ex) 자신이 깃헙을 가입한 이메일이 hello123@korea.ac.kr이면
    git config --global user.email hello123@korea.ac.kr

git config --list
-> git username과 email 확인
```

# | VS Code 설치

<https://code.visualstudio.com>



Window 유저

설치 후 git bash에서  
code . 쳐서 vscode 뜨면 설치 성공

\* code . 의미: vscode를 현재 경로에서 열겠다.  
(‘.’은 현재 경로를 의미하고, code는 (vs)code를 가리킴.)



MacOS 유저

다운로드 후 code . 따로 설정 필요!

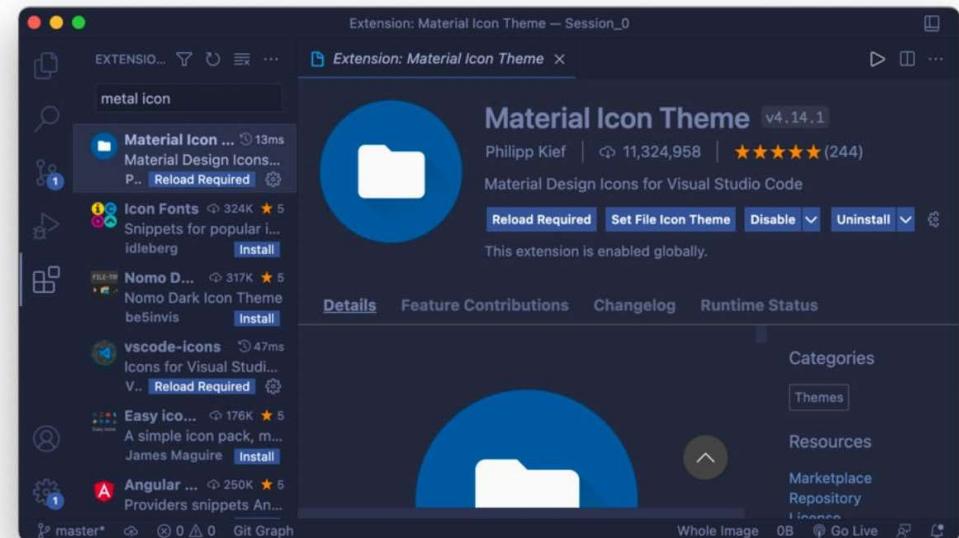
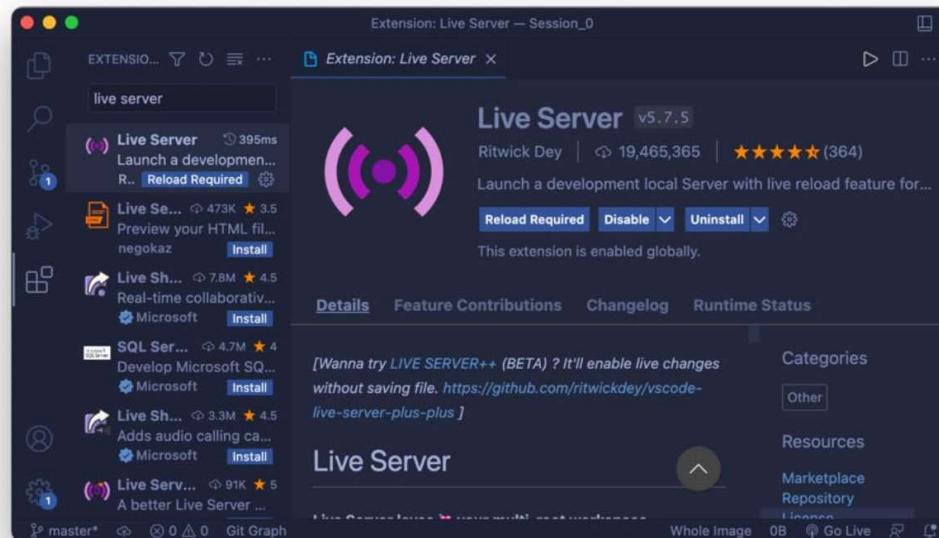


```
>shell command
Shell Command: Install 'code' command in PATH
Shell Command: Uninstall 'code' command from PATH
```

Command Palette ( $\uparrow \text{⌘} \text{P}$ ) 연 다음 shell command 검색 & 설치

# VS Code 설치

Live Server & material icon theme 설치하기



# VS Code 설치

## Prettier 설치

The screenshot shows the details page for the "Prettier - Code formatter" extension in the VS Code Marketplace. The extension is version v10.1.0, has over 41,931,488 downloads, and a rating of 4.5 stars from 446 reviews. It is described as a "Code formatter using prettier". The status bar indicates it is enabled globally. The main content area is titled "Prettier Formatter for Visual Studio Code" and explains that Prettier is an opinionated code formatter that enforces a consistent style by parsing your code and re-printing it with its own rules. It supports various languages including JavaScript, TypeScript, Flow, JSX, JSON, CSS, SCSS, Less, HTML, Vue, Angular, Handlebars, Ember, Glimmer, GraphQL, Markdown, and YAML. A "Your favorite language?" link is also present. Below this, there's an "Installation" section with instructions to search for "Prettier - Code formatter" in the VS Code extensions market place or to run the command `code style prettier` in the terminal. The right sidebar contains sections for "Categories" (Formatters), "Resources" (Marketplace, Issues, Repository, License, Prettier), and "More Info" (published date, last released, last updated, identifier).

Prettier - Code formatter v10.1.0

Prettier prettier.io | 41,931,488 | ★★★★☆(446) | Sponsor

Code formatter using prettier

Disable Uninstall ⚙️

This extension is enabled globally.

DETAILS FEATURES CHANGELOG

### Prettier Formatter for Visual Studio Code

Prettier is an opinionated code formatter. It enforces a consistent style by parsing your code and re-printing it with its own rules that take the maximum line length into account, wrapping code when necessary.

JavaScript · TypeScript · Flow · JSX · JSON  
CSS · SCSS · Less  
HTML · Vue · Angular · HANDLEBARS · Ember · Glimmer  
GraphQL · Markdown · YAML  
Your favorite language?

Main passing downloads 212M installs 42M code style prettier follow prettier

#### Installation

Install through VS Code extensions. Search for [Prettier - Code formatter](#)

Visual Studio Code Market Place: [Prettier - Code formatter](#)

Can also be installed in VS Code: Launch VS Code Quick Open (Ctrl+P), paste the following command, and press enter.

Categories

Formatters

Resources

Marketplace Issues Repository License Prettier

More Info

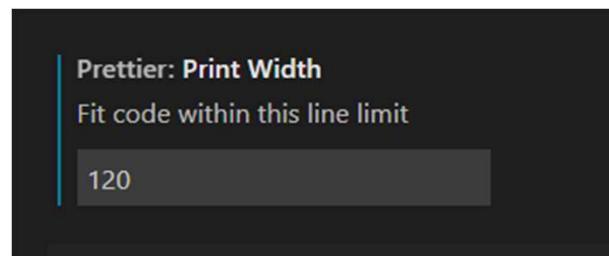
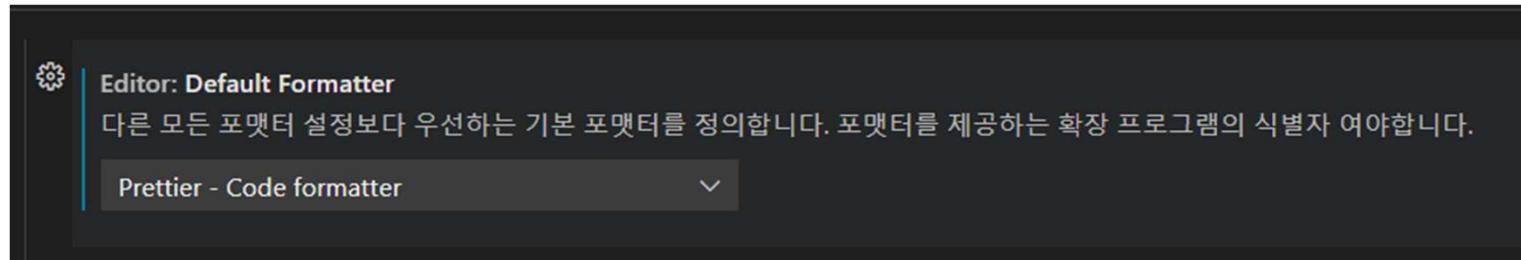
|               |                        |
|---------------|------------------------|
| Published     | 2017-01-11, 04:52:02   |
| Last released | 2023-08-01, 01:08:00   |
| Last updated  | 2023-08-01, 14:23:27   |
| Identifier    | esbenp.prettier-vscode |

# VS Code 설치

## Prettier 세부 설정

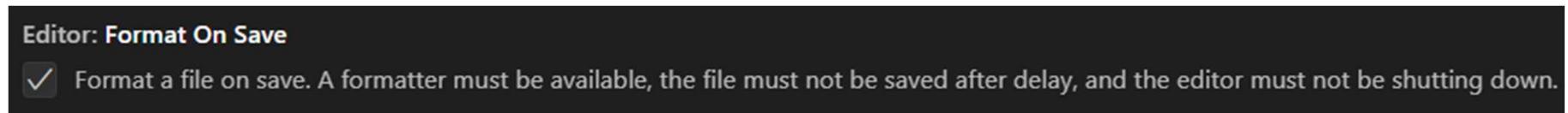
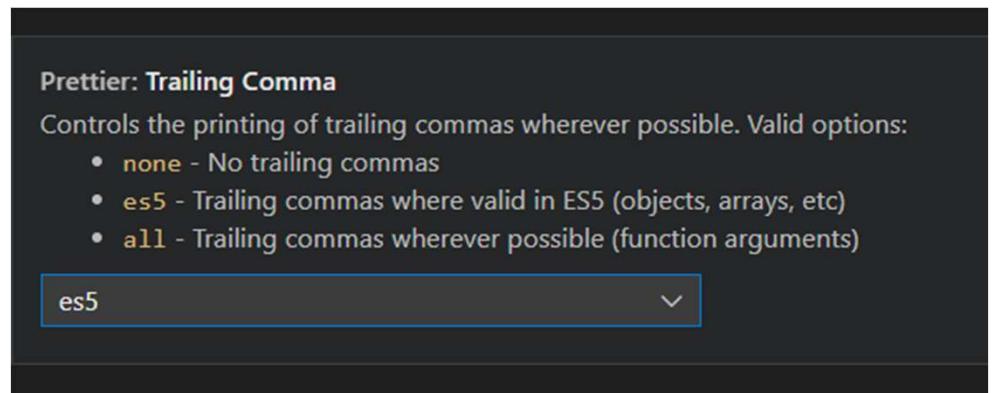
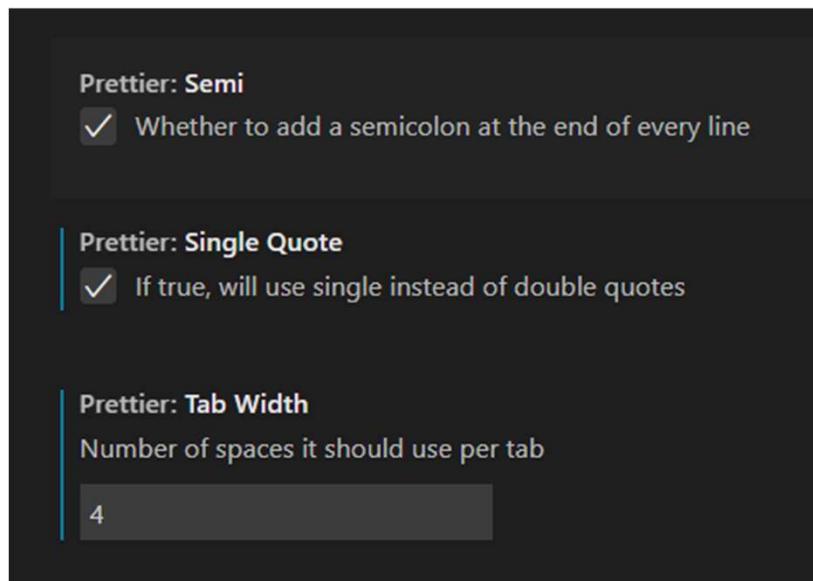
Window는 Ctrl + 쉼표, Mac은 Commnad+쉼표 눌러서 VScode 환경 설정 켜기

검색하면서 하나씩 세부 설정 변경



# VS Code 설치

## Prettier 세부 설정



웹 개발?

# | 왜 웹개발인가?

웹보다는 앱개발이 대세 아닌가요?



# | 왜 웹개발인가?

## 1. 낮은 획득 비용

# | 왜 웹개발인가?

APP

CAC(획득 비용)

\$4.18

WEB

\$2.69

Retention Rate(재방문률)

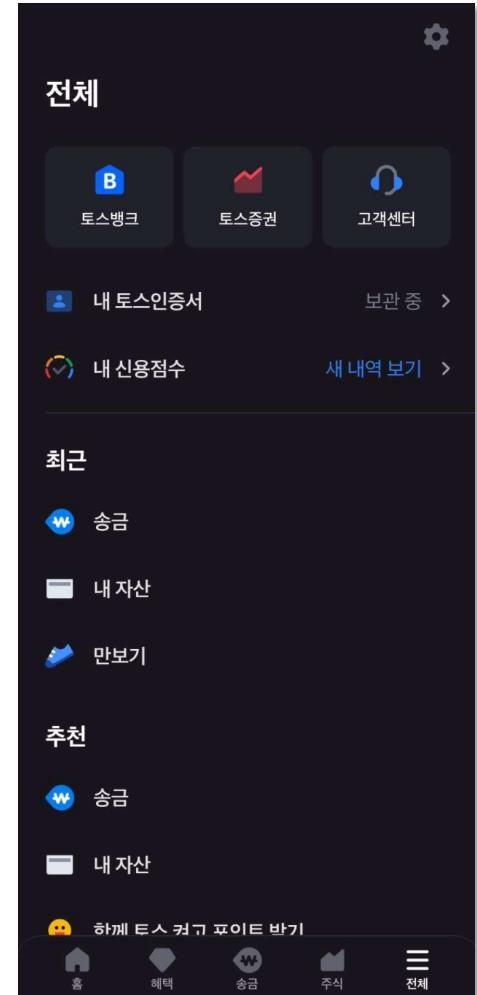
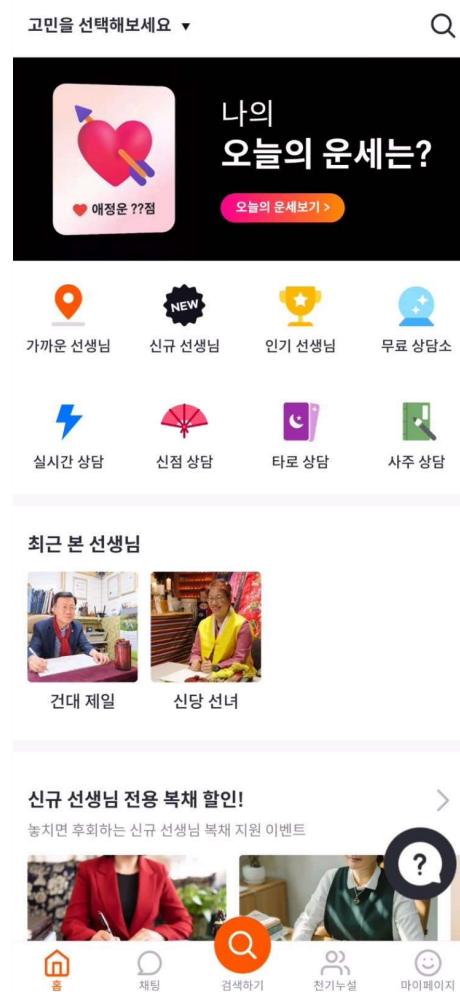


MVP용으로 가장 좋음...!

# | 왜 웹개발인가?

## 2. 뛰어난 확장성

# 왜 웹개발인가?



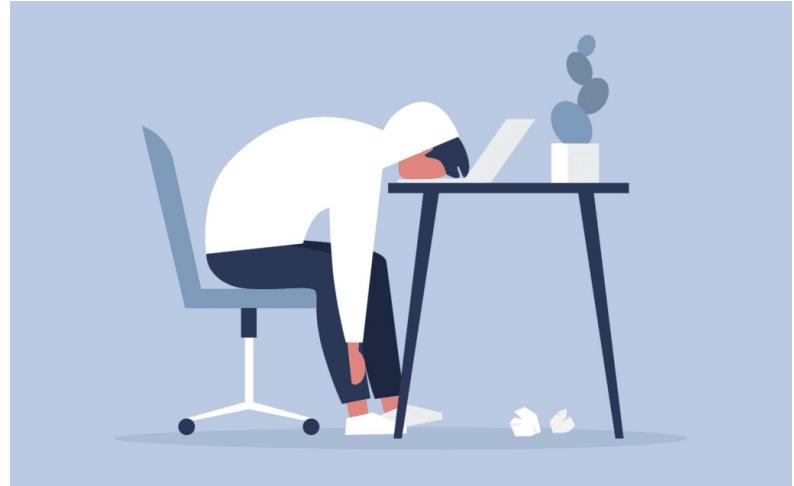
# | 왜 웹개발인가?

## 3. 재미

# | 왜 웹개발인가?

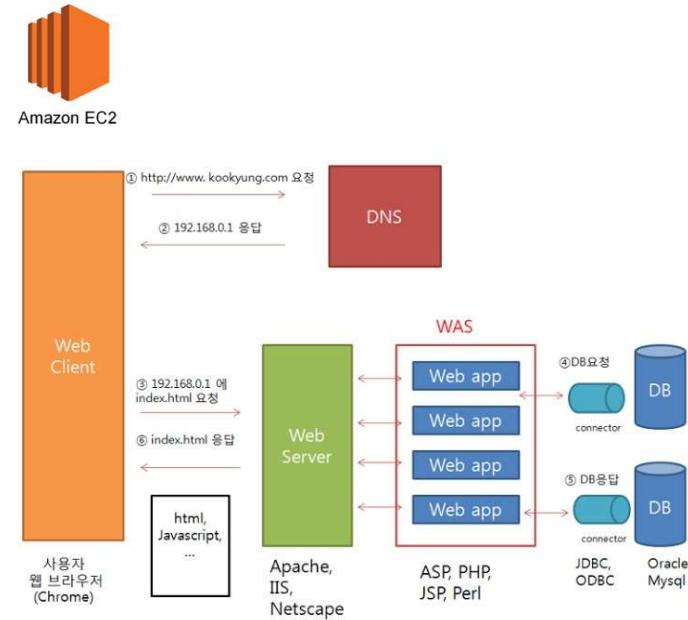
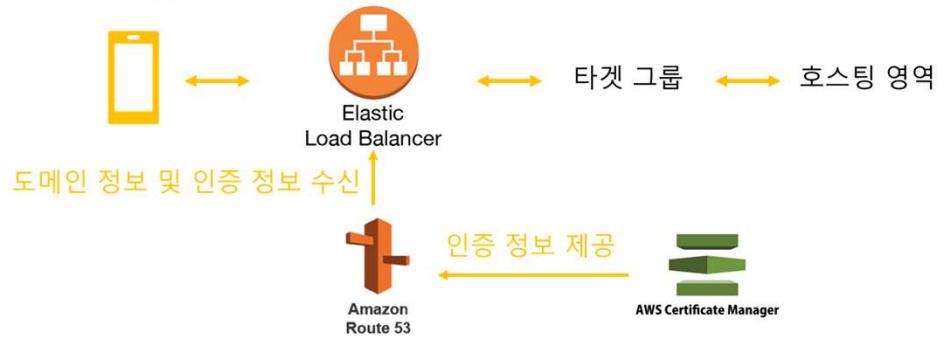


VS

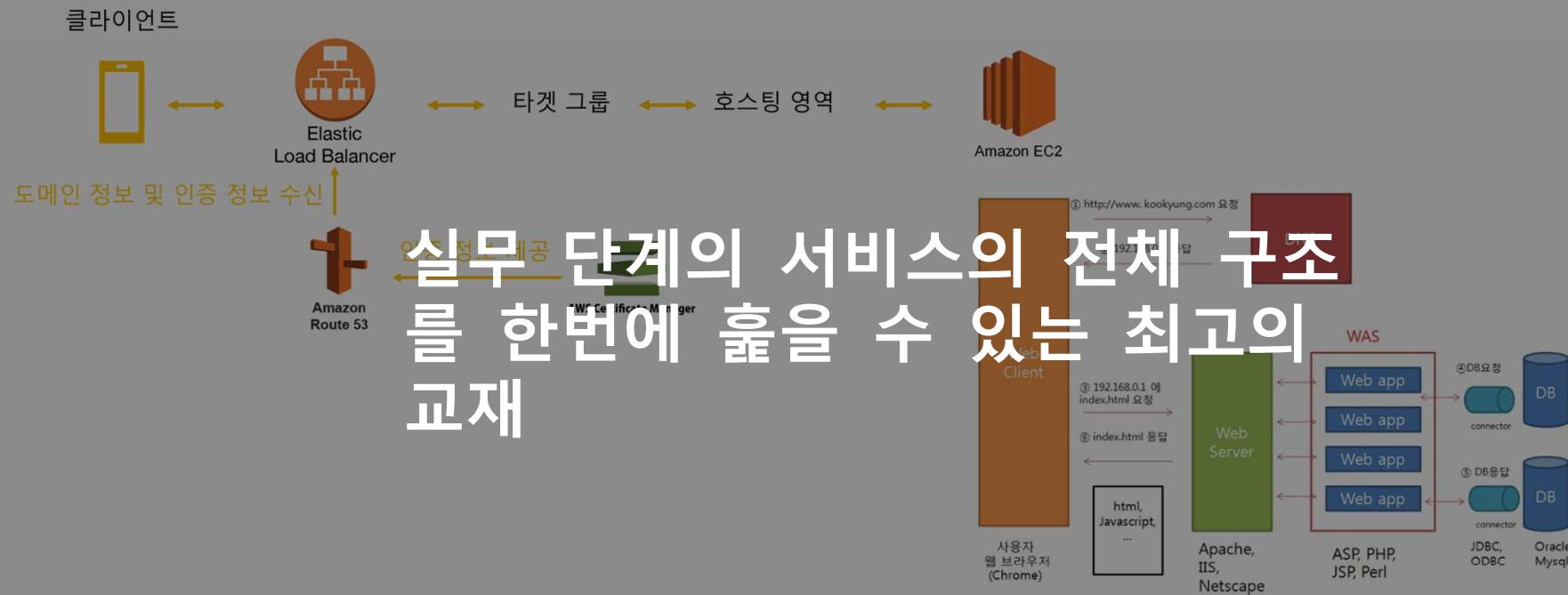


# 왜 웹개발인가?

클라이언트

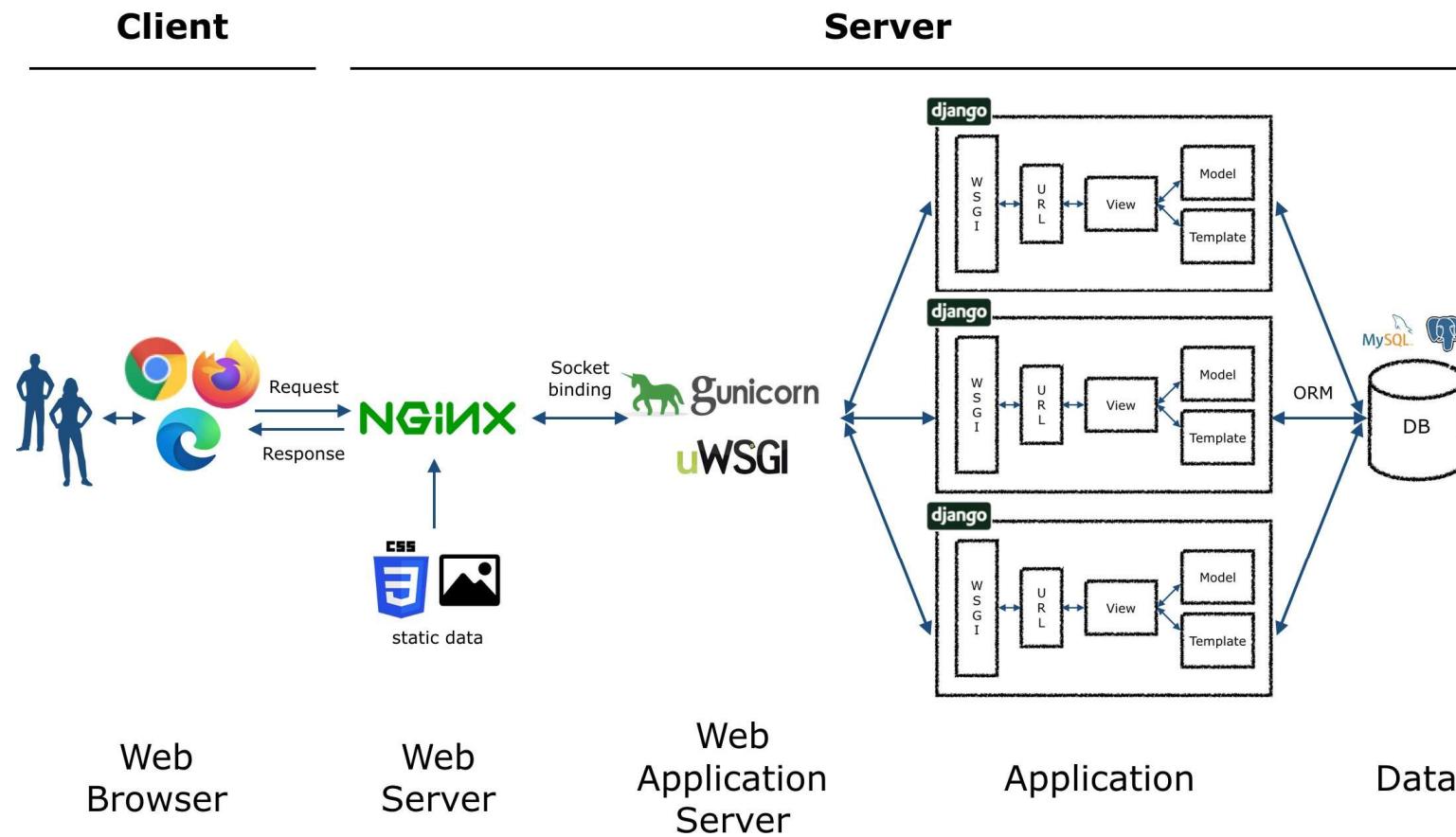


## 왜 웹개발인가?



NEXT X LIKELION

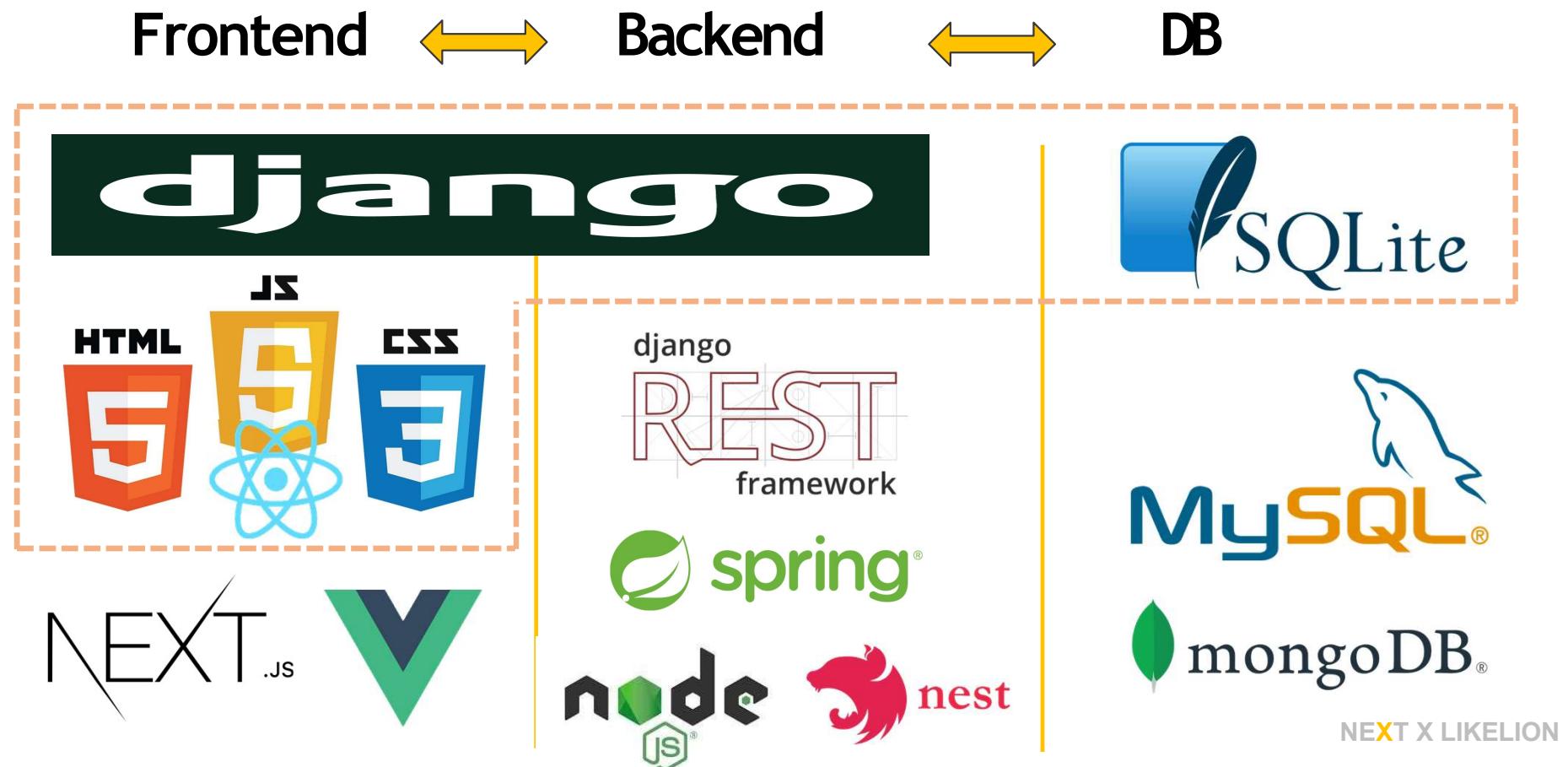
# 웹 서비스 구조



# | 웹 서비스 구조



# | 웹 서비스 구조



NEXT X LIKELION

# | 웹 서비스 구조-FE



뼈



피부



근육



# HTML 기초

# | HTML이란?

## Hyper Text Mark-up Language

한 문서에서 다른 문서로 즉시 접근할 수 있는,  
태그 및 텍스트 기반으로 구조화된 문서

웹 페이지 구조 혹은 데이터 작성용 마크업 언어.

# HTML 문서 구조

F12 누르고 네이버 손민수하기

```
<html>
<head>
</head>
<body>
<script>           </script>
</body>
</html>
```

```
<!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">
<title>Document</title>
</head>
<body>
</body>
</html>
```

# | HTML 문서 구조

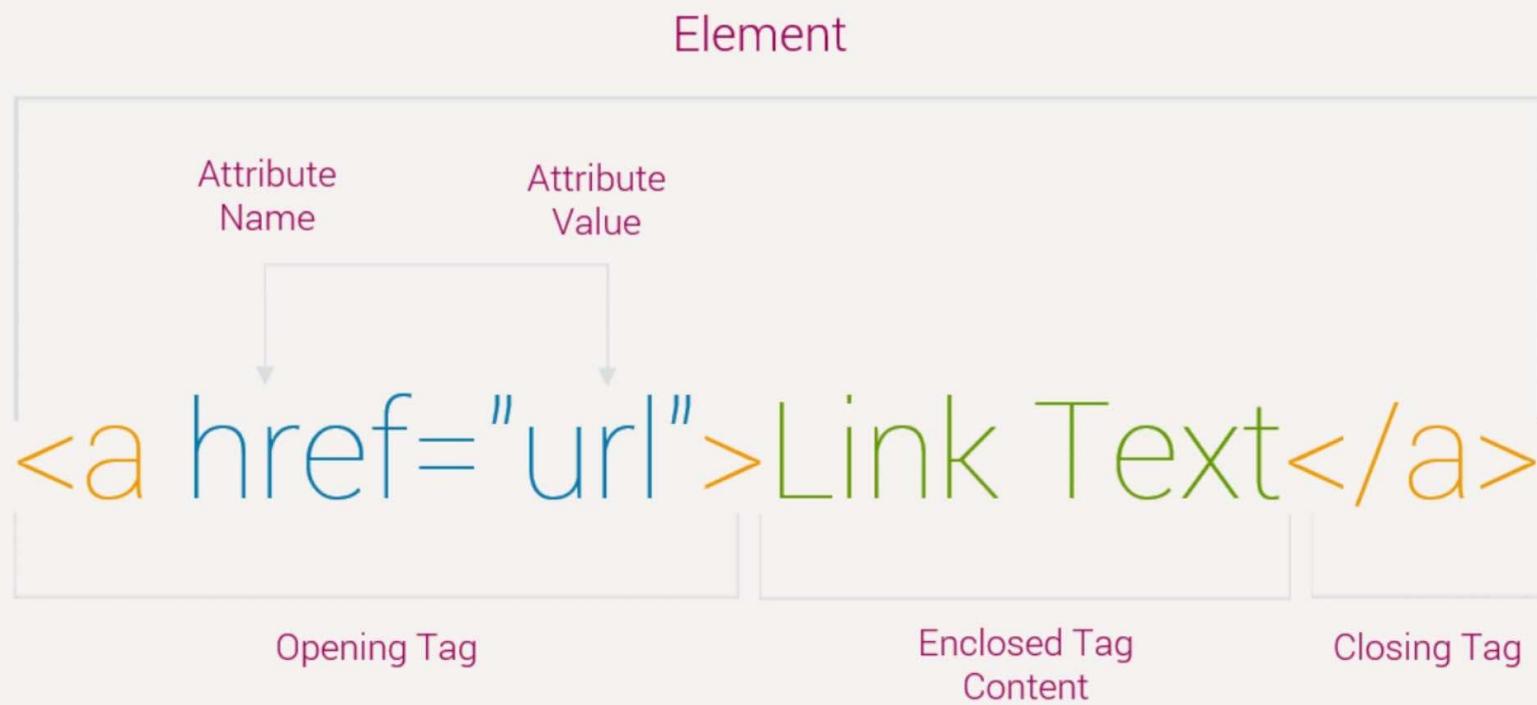
F12 누르고 네이버 손민수하기

네이버 노잼;;

세계 최초 웹사이트 정도는 봐야지;;;

<http://info.cern.ch/>

# | HTML 태그



# HTML 태그 종류

<h1> ~ <h6>

<form>

<p>

<script>

<a>

<ul>

<li>

<img/>

<div>

:

HTML태그는 절대로 다 외울 수 없습니다! 자주쓰는것만 외우고 다른건 그때그때 검색!

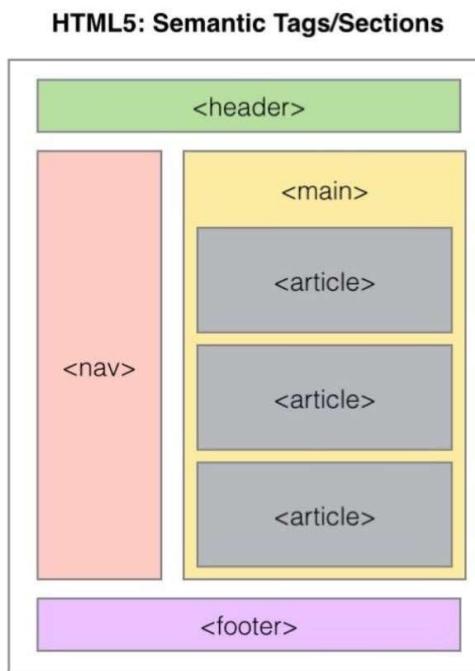
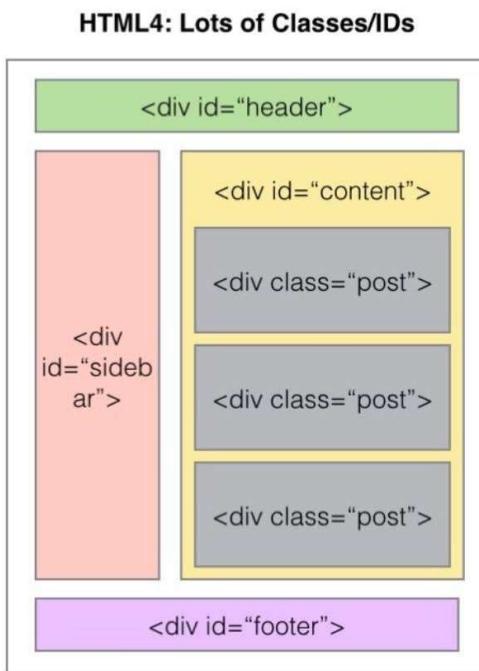
<https://developer.mozilla.org/ko/docs/Web/HTML/Element>

<https://www.w3schools.com/tags/default.asp>

# HTML5- Semantic tag

의미를 담은 태그 - 왕위를 계승하는 중입니다, HTML4!

## HTML4 vs HTML5 Page Structure on a Blog



<nav>  
<main>  
<header>  
<footer>  
<section>  
<article>  
<details>  
<aside>  
:  
:

# Tags: <img/>

사진도 넣어봅시다

```
< img src="../image.png" alt = "description" />
```



첨부하려는  
이미지 경로 (필수)



오류시 이미지  
대체할 수 있는 설명

절대경로: 페이지나 파일의 고유 주소(root기준)

상대경로: 현재 위치 기준으로 한 상대적 주소

# Tags: <form></form>

모든 입력의 부모 태그

```
< form action="server.html" method = "GET" />
```



폼 전송할 서버쪽  
스크립트 파일 지정



폼을 서버에 전송할  
http 메소드 지정

# Tags: <input></input>

뭐라도 입력은 받아야죠

```
< input type="type" name ="username" value ="name" />
```

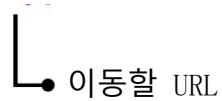
The diagram shows the HTML code for an input tag with three annotations:

- A bracket under "type" points to the text "입력 받을 data type".
- A bracket under "name" points to the text "Form data 이름".
- A bracket under "value" points to the text "Form data 내용".

## | Tags: <a></a>

링크 걸고 싶으면 a를 기억하세요

< a href=https://naver.com>네이버로 이동</a>



• 이동할 URL

# HTML 실습- 자기소개 페이지 만들기

HTML 맨보기

자기 소개 페이지를 만들어보자.

Browser title,

Text

List

Link

:

# CSS 기초

# CSS

CSS란...

## Cascading Style Sheets

스타일, 레이아웃 등 사용자에게 문서를 표시하는 방법을 지정하는 언어.

\* cascading : 위에서 아래로 흐르는, 상속 또는 종속하는.

## CSS 기본 문법

선택자 {속성: 값;}

```
p {  
    font-size : 1.2em;  
    color : gray;  
}
```

# 선택자

HTML 요소 선택하기

\*

모두 선택

```
* {  
    width : 100%;  
    box-sizing : border-box;  
}
```

div, p, body  
Tag 직접 선택

```
div {  
    font-size: 12px;  
    color : black;  
}
```

#id  
id로 선택

```
#id {  
    font-size: 24px;  
}
```

.class  
class로 선택

```
.class {  
    font-size: 24px;  
}
```

## **Selector Hierarchy(선택자 계층) -> Cascading**

**\* > tag > .class > #id > Inline Style**

## ID와 CLASS

id : 고유한 요소로 한 문서에 하나만 존재할 수 있음.  
class : 공통 요소로 다른 요소에 같은 class 반복 사용 가능

## | id, class

개발자님께서 멋진 이름을 주셨어요!

```
<a id="naverBtn" class="btn testBtn"  
href=https://naver.com
```

## CSS 실습 - CSS 적용하기

파일을 따로 관리합시다

```
<head>  
  ...  
  <link rel = "stylesheet" href=".style.css"/>  
</ head>
```

# 요소의 크기관리 : box-sizing

요소가 만들어지는 구조. 얘 외워야 웹개발 인생 편해집니다~~



# CSS 실습 : 자기소개 페이지 꾸미기

오늘은 CSS 맛만 보자구요



김지성입니다.

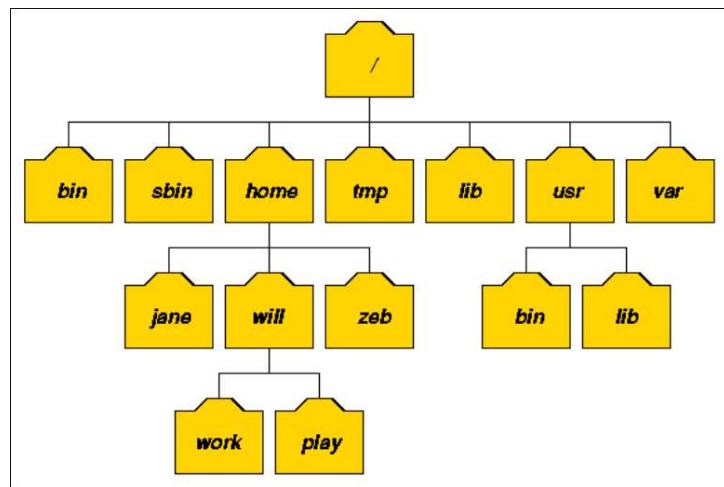
- 고려대학교 영어영문학과였다가
- 고려대학교 산업경영공학부가 되었습니다

사실 이것도 리스트인데  
점이 없어요

# Git 입문

# File System 알아보기

Git 입문 1



디렉토리 구조: 계층 트리 구조

하위 폴더에 접근하려면 현재 경로에서부터 차례대로

ex1) /home/will/play

ex2) /usr/bin

# CLI(CommandLine Interface) 배우기

Git 입문 2



## CLI (Command Line Interface)

텍스트 기반의 명령어 입력으로 시스템 조작  
<→ GUI (Graphic User Interface) 버튼, 아이콘, 이미지  
등과 같은 운영 체제와 상호 작용할 수 있는 비주얼을 제공

# CLI(CommandLine Interface) 배우기

## Basic Command

[경로 탐색] **pwd**: print (current) working directory

[경로 이동] **cd**: change directory

[파일 확인] **ls**: list segments

**ls -a**: list segments of all

**ls -l**: list directory with long listing

**ls -al**: list directory of all with long listing

[파일 생성] **touch**: create a file

**mkdir**: make a directory

[파일 제거] **rm**: remove a file

**rm -rf**: remove forcibly including directory and the files in it

# CLI(CommandLine Interface) 배우기

Basic Command

pwd

현재 절대 경로를 확인합니다.

```
chunhyerin@iMac Session_0 % pwd  
/Users/chunhyerin/Desktop/Session_0
```

# CLI(CommandLine Interface) 배우기

## Basic Command

cd 원하는\_경로

원하는 절대 경로, 상대 경로 주소값으로  
이동합니다.

```
chunhyerin@iMac Session_0 % cd ..  
chunhyerin@iMac Desktop % cd Session_0  
chunhyerin@iMac Session_0 %
```

상위 폴더인 Desktop 으로 이동  
Session\_0 으로 이동

절대경로 - root directory에서 해당 directory까지의 경로  
(Pwd 를 사용했을 때 나오는 경로)

상대경로 - 현재 directory에서 해당 directory까지의 경로

“.” - 현재 경로를 의미  
“..” - 위의 경로를 의미

# CLI(CommandLine Interface) 배우기

## Basic Command

**ls** 현재 디렉토리 위치에 내용 나열

**ls -a** 숨김 파일 (.., .git, .gitignore 등)  
포함해서 모두 표시

**ls -l** 좀 더 자세한 정보 (수정 날짜 등)  
포함해서 출력

**ls -al** ls -a, ls -l 둘 다 같이 섞어서  
출력

```
chunhyerin@iMac Session_0 % ls
README.md      Session0.png      Session0.txt
chunhyerin@iMac Session_0 % ls -a
.              ..              .git          README.md      Session0.png      Session0.txt
chunhyerin@iMac Session_0 % ls -l
total 8
-rw-r--r--  1 chunhyerin  staff  103  3 10 10:20 README.md
-rw-r--r--  1 chunhyerin  staff    0  3 10 09:04 Session0.png
-rw-r--r--  1 chunhyerin  staff    0  3 10 09:04 Session0.txt
chunhyerin@iMac Session_0 % ls -al
total 8
drwxr-xr-x  6 chunhyerin  staff  192  3 10 09:32 .
drwx-----+ 7 chunhyerin  staff  224  3 10 09:03 ..
drwxr-xr-x 12 chunhyerin  staff  384  3 10 09:35 .git
-rw-r--r--  1 chunhyerin  staff  103  3 10 10:20 README.md
-rw-r--r--  1 chunhyerin  staff    0  3 10 09:04 Session0.png
-rw-r--r--  1 chunhyerin  staff    0  3 10 09:04 Session0.txt
```

# CLI(CommandLine Interface) 배우기

Basic Command

**touch 파일\_이름**

현재 디렉터리에서 “파일\_이름”을  
가진 파일을 생성합니다.

```
chunhyerin@iMac Session_0 % touch hello.txt
chunhyerin@iMac Session_0 % ls
README.md          Session0.png      Session0.txt      hello.txt
```

# CLI(CommandLine Interface) 배우기

## Basic Command

### rm 파일\_이름

현재 디렉터리에서 “파일\_이름”을 가진 파일을 삭제합니다.

### rm - rf 폴더\_이름

현재 디렉터리에서 “폴더\_이름”을 가진 파일을 삭제합니다.

```
chunhyerin@iMac Session_0 % ls
README.md      Session0.png    Session0.txt    hello.txt      newFolder
chunhyerin@iMac Session_0 % rm hello.txt
chunhyerin@iMac Session_0 % ls
README.md      Session0.png    Session0.txt    newFolder
chunhyerin@iMac Session_0 % rm -rf newFolder
chunhyerin@iMac Session_0 % ls
README.md      Session0.png    Session0.txt
```

# Git 기초 버전관리

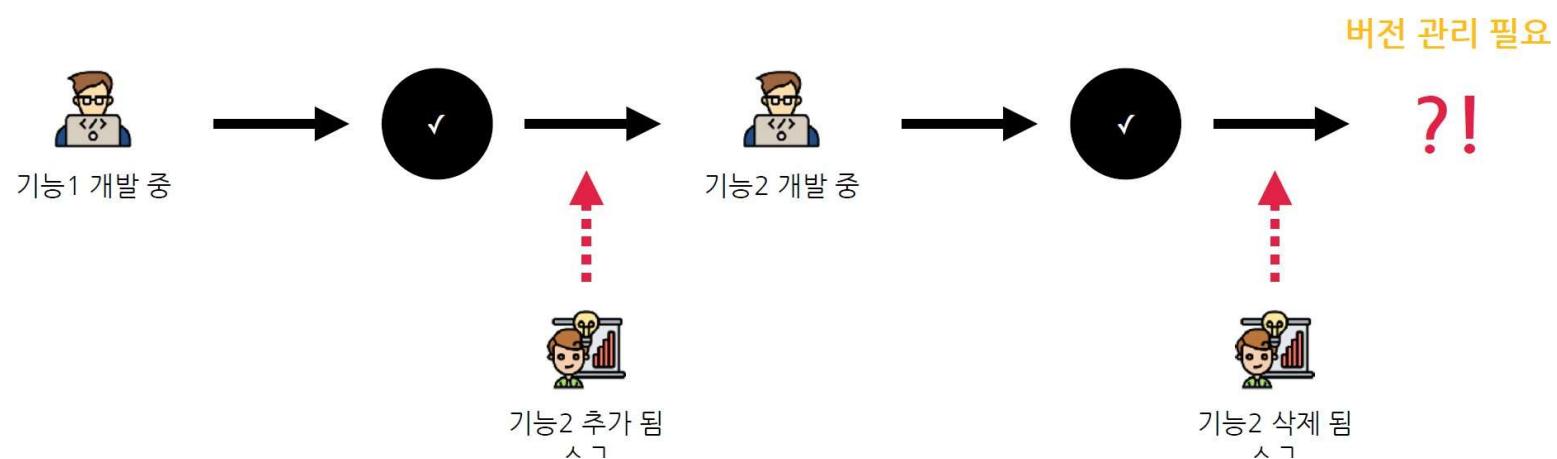
깃이 뭔데. 그거 어떻게 하는건데.

커밋이요? 머지요? 뭐요?



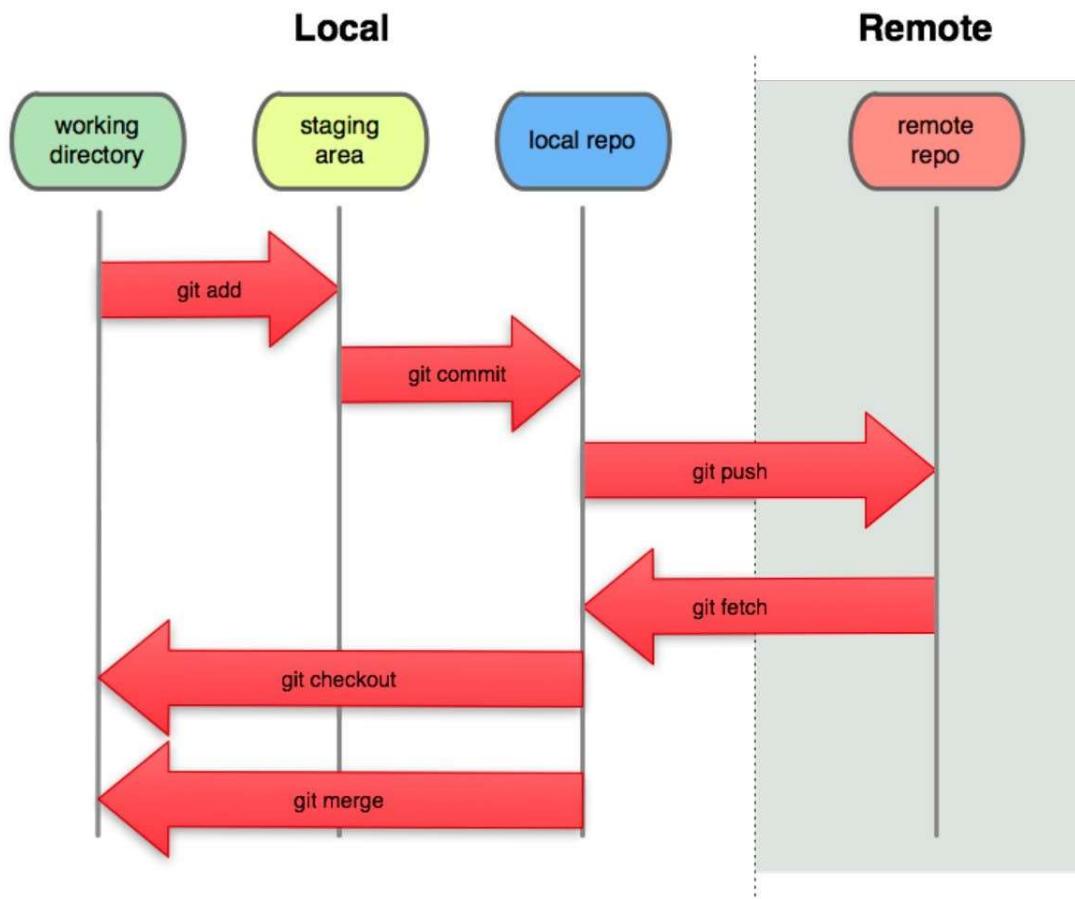
# Git 기초 버전관리

그거 왜쓰는데



# Git 기초 버전관리

## Git 기능 정리

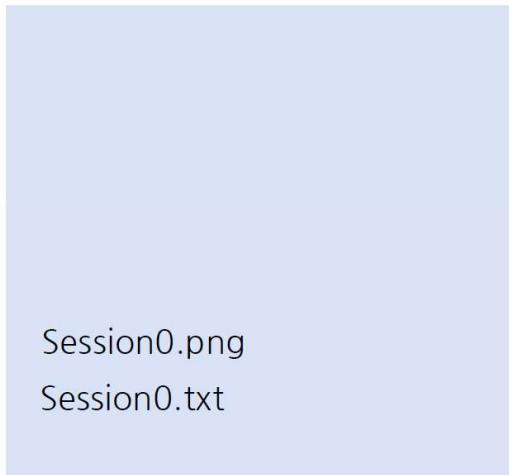


# Git 기초 버전관리

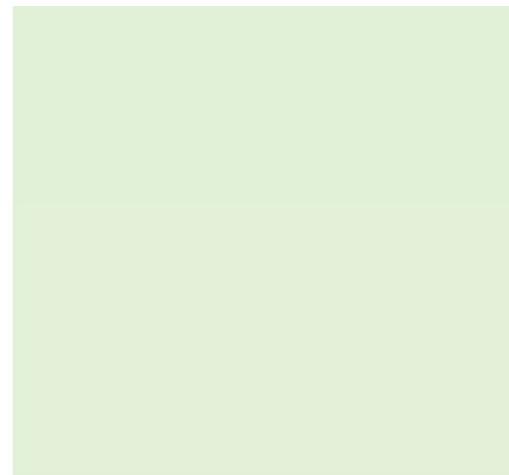
아무것도 안했을 때 초기 상태

Local

Working Directory



Remote



# Git 기초 버전관리

스테이징 영역 활성화 및 추적 시작



`git init`

# Git 기초 버전관리

스테이징 영역에 파일 추가

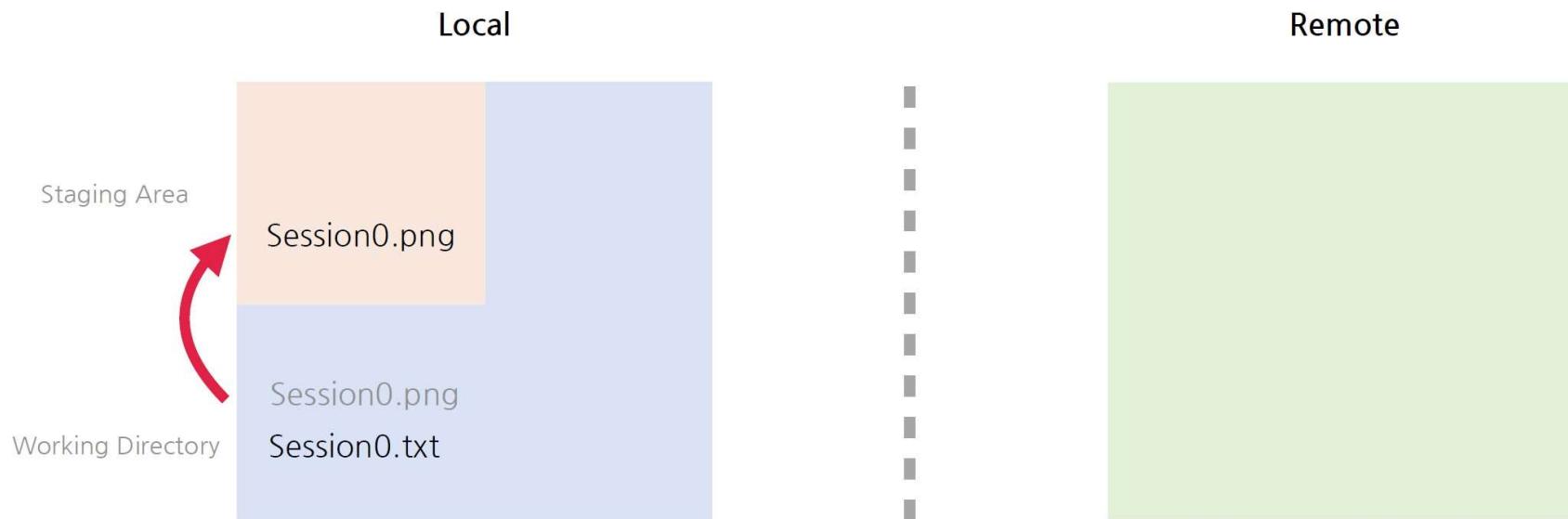


**git add \${file\_name}**

git add index.html      git add .

# Git 기초 버전관리

Local Repository에 확정



**git commit -m “\${커밋메세지}”**

git commit -m “first commit”

# Git 기초 버전관리

## Git 기록 관리

```
chunhyerin@iMac Session_0 % git log
commit c3f15a64ed342864b782469bfb2ebae2525a3eba (HEAD -> master)
Author: deborah91298 <deborah91298@korea.ac.kr>
Date:   Thu Mar 10 09:34:09 2022 +0900
```

첫 세션 푸시

git log

# Git 기초 버전관리

## Git 기록 관리



```
git add .  
git commit -m "second commit"
```

“.” - 현재 경로를 의미.  
즉, 현재 경로에 있는 모든  
파일 및 폴더를 포함함

## Github에 푸시하기

Github 그게 뭔데. 그거 어떻게 하는건데.



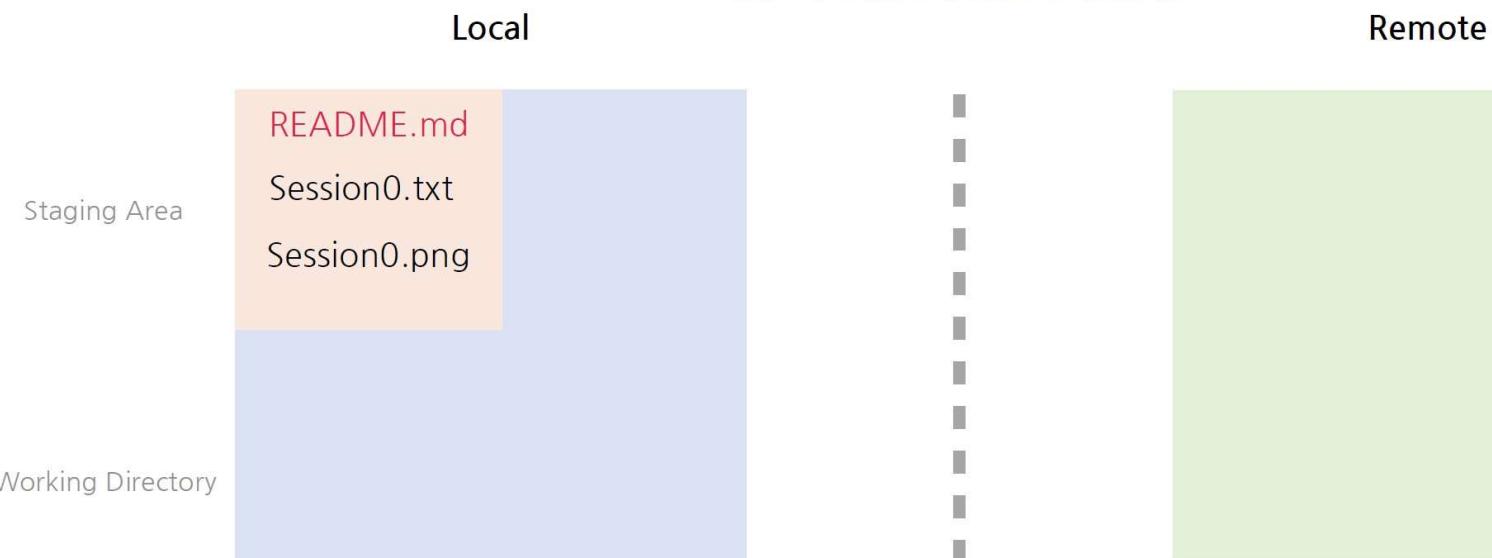
**Github**

협업 지원, 오픈 소스의 성지, 코드 관리 페이지

# Github에 푸시하기

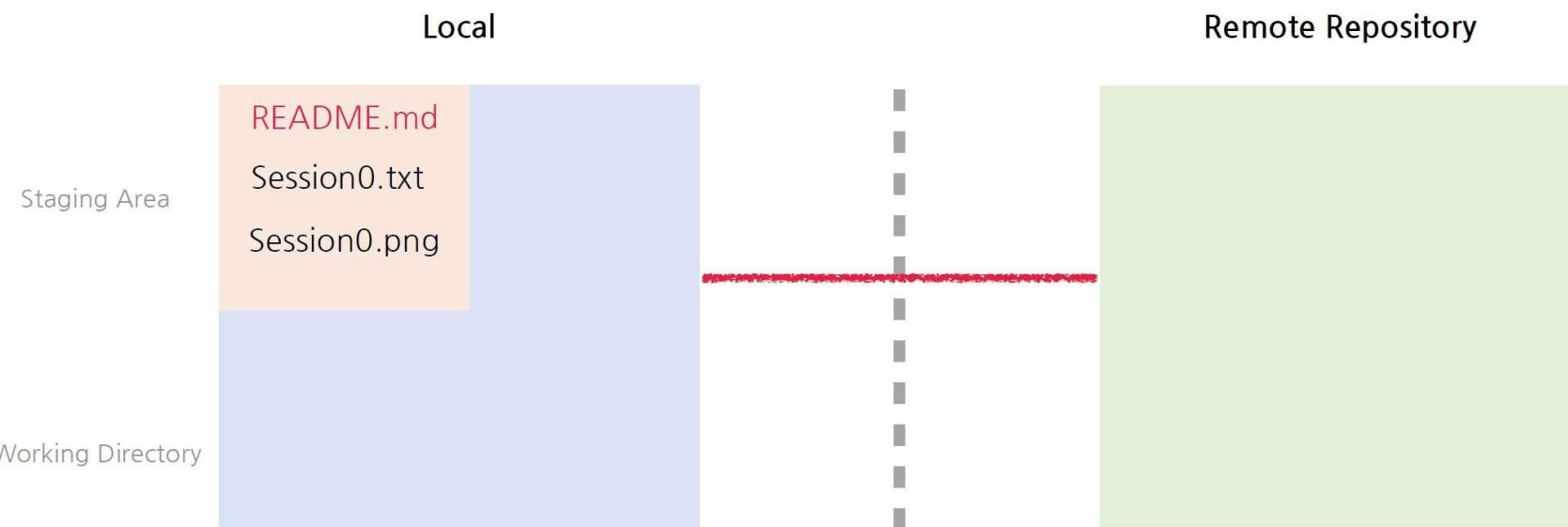
Remote Repository 현재 상태

만들어져 있는데 연결이 안 됐네?



# Github에 푸시하기

Local Repository와 Remote Repository 연결하기



`git remote add origin ${repository_url}`

`git remote add origin https://github.com/${github_id}/${repository_name}.git`

# Github에 푸시하기

Local Repository와 Remote Repository 연결 확인

```
chunhyerin@iMac Session_0 % git remote -v
origin  https://github.com/deborah91298/Next_Likelion.git (fetch)
origin  https://github.com/deborah91298/Next_Likelion.git (push)
```

**git remote -v**

# Github에 푸시하기

Github Repository에 접근 가능한 Personal Access Token 생성

My profile > Settings > Developer Settings > Personal access tokens > Generate new token

The screenshot shows the GitHub developer settings page under the 'Personal access tokens' tab. It displays a single token entry:

| repo — admin:repo_hook, delete_repo, gist, repo, user | Last used within the last week | Delete |
|-------------------------------------------------------|--------------------------------|--------|
| ⚠️ This token has no expiration date.                 |                                |        |

Below the table, a note states: "Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication."

At the bottom, there is a footer with links: © 2022 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About

# Github에 푸시하기

## Github Repository에 접근 가능한 Personal Access Token 생성

Note  
repo

What's this token for?  
Expiration  
This token has no expiration date. To set a new expiration date, you must [regenerate the token](#).

Select scopes  
Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

|                                                     |                                                                     |
|-----------------------------------------------------|---------------------------------------------------------------------|
| <input checked="" type="checkbox"/> repo            | Full control of private repositories                                |
| <input checked="" type="checkbox"/> repo:status     | Access commit status                                                |
| <input checked="" type="checkbox"/> repo:deployment | Access deployment status                                            |
| <input checked="" type="checkbox"/> public_repo     | Access public repositories                                          |
| <input checked="" type="checkbox"/> repo:invite     | Access repository invitations                                       |
| <input checked="" type="checkbox"/> security_events | Read and write security events                                      |
| <input type="checkbox"/> workflow                   | Update GitHub Action workflows                                      |
| <input type="checkbox"/> write:packages             | Upload packages to GitHub Package Registry                          |
| <input type="checkbox"/> read:packages              | Download packages from GitHub Package Registry                      |
| <input type="checkbox"/> delete:packages            | Delete packages from GitHub Package Registry                        |
| <input type="checkbox"/> admin:org                  | Full control of orgs and teams, read and write org projects         |
| <input type="checkbox"/> write:org                  | Read and write org and team membership, read and write org projects |
| <input type="checkbox"/> read:org                   | Read org and team membership, read org projects                     |
| <input type="checkbox"/> admin:public_key           | Full control of user public keys                                    |
| <input type="checkbox"/> write:public_key           | Write user public keys                                              |
| <input type="checkbox"/> read:public_key            | Read user public keys                                               |
| <input checked="" type="checkbox"/> admin:repo_hook | Full control of repository hooks                                    |
| <input checked="" type="checkbox"/> write:repo_hook | Write repository hooks                                              |
| <input checked="" type="checkbox"/> read:repo_hook  | Read repository hooks                                               |
| <input type="checkbox"/> admin:org_hook             | Full control of organization hooks                                  |
| <input checked="" type="checkbox"/> gist            | Create gists                                                        |
| <input type="checkbox"/> notifications              | Access notifications                                                |

|                                                    |                                                                         |
|----------------------------------------------------|-------------------------------------------------------------------------|
| <input checked="" type="checkbox"/> user           | Update ALL user data                                                    |
| <input checked="" type="checkbox"/> read:user      | Read ALL user profile data                                              |
| <input checked="" type="checkbox"/> user:email     | Access user email addresses (read-only)                                 |
| <input checked="" type="checkbox"/> user:follow    | Follow and unfollow users                                               |
| <input checked="" type="checkbox"/> delete_repo    | Delete repositories                                                     |
| <input type="checkbox"/> write:discussion          | Read and write team discussions                                         |
| <input type="checkbox"/> read:discussion           | Read team discussions                                                   |
| <input type="checkbox"/> admin:enterprise          | Full control of enterprises                                             |
| <input type="checkbox"/> manage_runners:enterprise | Manage enterprise runners and runner-groups                             |
| <input type="checkbox"/> manage_billing:enterprise | Read and write enterprise billing data                                  |
| <input type="checkbox"/> read:enterprise           | Read enterprise profile data                                            |
| <input type="checkbox"/> admin:gpg_key             | Full control of public user GPG keys <small>(Developer Preview)</small> |
| <input type="checkbox"/> write:gpg_key             | Write public user GPG keys                                              |
| <input type="checkbox"/> read:gpg_key              | Read public user GPG keys                                               |

[Update token](#) [Cancel](#)

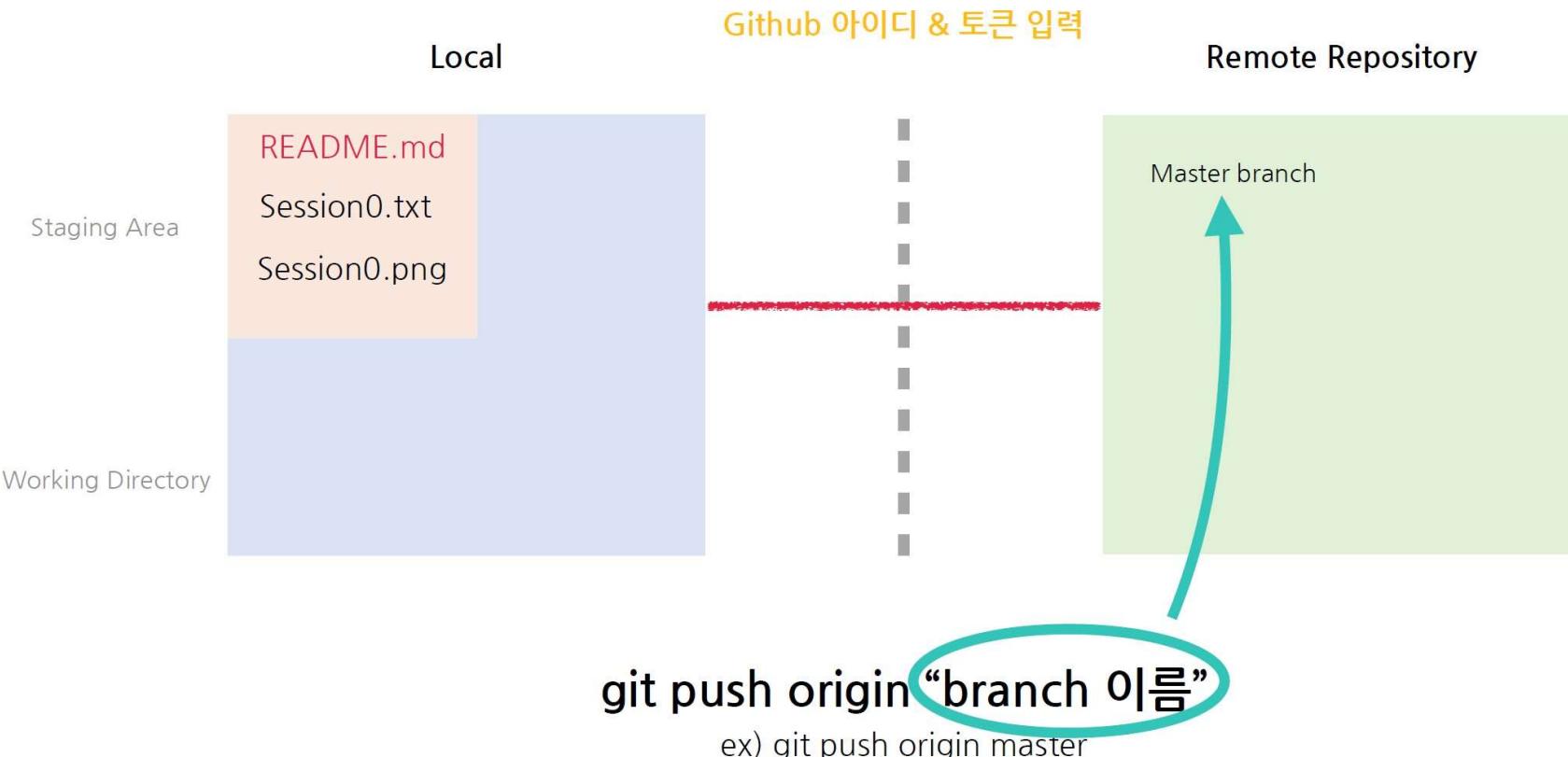
### Delete personal access token

Any applications or scripts using this token will no longer be able to access the GitHub API. You cannot undo this action.

[Delete this token](#)

# Github에 푸시하기

Github Repository에 코드 올리기



# Git 명령어 정리

## <git 초기 설정>

[전역 사용자명/이메일 구성하기] git config --global user.name \${user.name}  
git config --global user.email \${user.email}  
git config --list

[새로운 저장소 초기화] git init

[저장소 복제] git clone \${깃헙 주소}

[Local & Remote 저장소 연결] git remote add origin \${본인 레포 주소}  
git remote -v

## <git 파일 스테이징 & 푸시>

[git 스테이징 영역에 새로운 파일 추가] git add \${file\_name}  
git status

[git 스테이징 영역에 존재하는 파일 커밋하기] git commit -m "\${commit message}"  
git log

[Github 레포에 코드 올리기] git push origin \${branch name}

**복습 과제 (3월 7일 세션 전까지)**

교수님 살려주세요

오늘 실습하면서 만든 자기소개 페이지를  
예쁘게 꾸며서  
과제용 레포지토리에 푸시하고  
NEXT Github에 댓글을 남겨주세요!

**영상 과제 (3월 7일 세션 전까지)**

교수님 살려주세요

**코드카데미 강의를 수강하고 (\*pro 제외)**

**위 복습 과제 Github 댓글과 함께**

**수강 인증 캡처 사진을 남겨주세요!**

Learn HTML

<https://www.codecademy.com/learn/learn-html>

Learn CSS

<https://www.codecademy.com/learn/learn-css>