

WEB

• • •



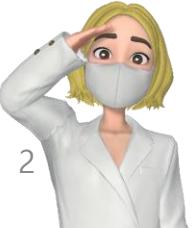
한성호 강사
(ALOHA)



OT

수강을 환영합니다!

- 강사 소개
- 수업 안내
 - 커리 쿨럼
 - 수업 시간
 - 학습 도구
 - 수업 준비
 - Q & A





한성호

강사



ALOHA CLASS



강의 과목

- HTML CSS JavaScript jQuery
- 프로그래밍 언어
- 데이터베이스
- 웹 디자인 기능사
- 정보처리 기사





주요 이력

- 정보통신공학 전공
- 2018~2021
 - 온/오프라인 IT 레슨
- 2019~2020
 - e-HR 인사시스템 개발
 - 신세계그룹 인사시스템 재구축 프로젝트
 - 페퍼저축은행 e-HR 암호설정 조건변경 고도화
 - SK엠앤서비스 Biz in 시스템 취약점개선조치 프로젝트





주요 이력

- 2018~2017
 - 창업아이디어 공모전 우수상
 - SK청년비상 창업경진대회 은상
 - 캡스톤디자인 프로젝트 경진대회 대상
 - 캡스톤디자인 프로젝트 경진대회 동상
 - K-ICT 디바이스랩 용인 사물인터넷 메이커톤 우수상
 - 창의적 종합설계 경진대회 장려상



Soomgo



5.0 리뷰 평점 | 41 리뷰수 | 115 고용수

.|| 활동분석



리뷰

5.0 ★★★★★
41개 리뷰

이** ★★★★★ 2021.05.09

혼자 학습하다 고수님을 만나 부족한부분을 코칭받고 시험에 응했고 결국 합격!!!
수업을 들으면서 정말 많은 사람들을 가르쳐보셨구나 하는 느낌을 많이 받았으며, 눈높이에 맞추어 강의
해주셔서 너무 감사했고 고수님 같은 개발자가 되어야겠다고 다짐하는 계기가 되었습니다.

[더보기](#)

안** ★★★★★ 2021.04.12

자바가 어려워 고민하던 중 선생님을 만났고, 쉬운 예제와 눈높이에 맞춰주신 설명으로 자바에 흥미가 불었습니다.

주변에 자바가 어려워 고민하는 지인이 있다면 추천의사 100%!!

이** ★★★★★ 2021.04.07

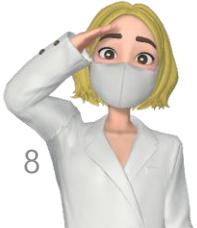
Web 과정은 처음이라 걱정을 많이 했는데 친절하게 하나씩 레슨 해주셔서 감사히 수강하고 있습니다. 개인 상황에 맞게 커리큘럼 짜주시고 알려 주셔서 많은 도움이 되고 있습니다. 앞으로도 잘 부탁드립니다. ^
^ 감사합니다~

김** ★★★★★ 2021.04.05

매우 친절하시고 꼼꼼하게 가르켜주세요 그리고 처음 봄 분한테도 안심할 수 있게 다가와 주시고요 매우 만족입니다.

수업안내

무엇을 배우나요?





교육 커리큘럼

차시	수업내용	차시	수업내용
1	OT, <u>프로그램</u> 설치	12	위치 속성
2	HTML 개념과 구조	13	변형 속성
3	웹 표준, 기본 태그	14	변환 속성
4	멀티미디어 태그, 입력 양식 태그	15	로그인 페이지 만들기
5	레이아웃 태그	16	애니메이션 속성
6	CSS 기초, 선택자와 적용 방법	17	CSS 레이아웃 이해하기
7	텍스트 태그 및 타이포그래피 속성	18	CSS 기본 레이아웃 만들기
8	박스모델	19	레이아웃 모듈 - 헤더, 메뉴
9	배치 속성	20	레이아웃 모듈 - 섹션, 푸터
10	표현 속성, 리스트 속성	21	Flex 레이아웃
11	배경과 색상 표현	22	Grid 레이아웃



Q & A

궁금하세요?



프로그램 설치

#1

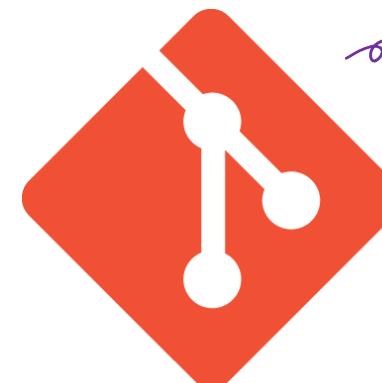


프로그램 설치

설치할 프로그램을 알려드릴게요😊



코드 에디터



형상관리 툴

VS Code



Git





VS Code

설치하기

- 다운로드
 - <https://code.visualstudio.com/download>
- 설정 하기
 - 폴더 설정
 - 폰트 설정
- 확장 설치
 - Korean Language Pack
 - Live Sever
 - ...



Download Visual Studio Code - +

code.visualstudio.com/download

Visual Studio Code Docs Updates Blog API Extensions FAQ Learn

Search Docs Download

Version 1.58 is now available! Read about the new features and fixes from June.

영어 한국어 Google Translate

Download Visual Studio Code

Free and built on open source. Integrated Git, debugging and extensions.



↓ Windows
Windows 7, 8, 10

User Installer: 64 bit 32 bit ARM
System Installer: 64 bit 32 bit ARM
.zip 64 bit 32 bit ARM



↓ .deb
Debian, Ubuntu

↓ .rpm
Red Hat, Fedora, SUSE

.deb 64 bit ARM ARM 64
.rpm 64 bit ARM ARM 64
.tar.gz 64 bit ARM ARM 64
[Snap Store](#)



↓ Mac
macOS 10.10+

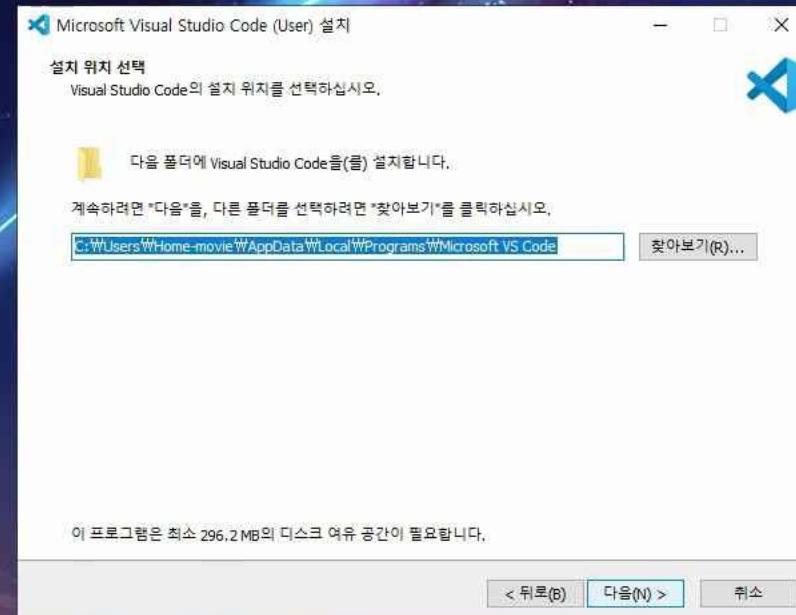
.zip Universal Intel Chip Apple Silicon

By downloading and using Visual Studio Code, you agree to the [license terms](#) and [privacy statement](#).

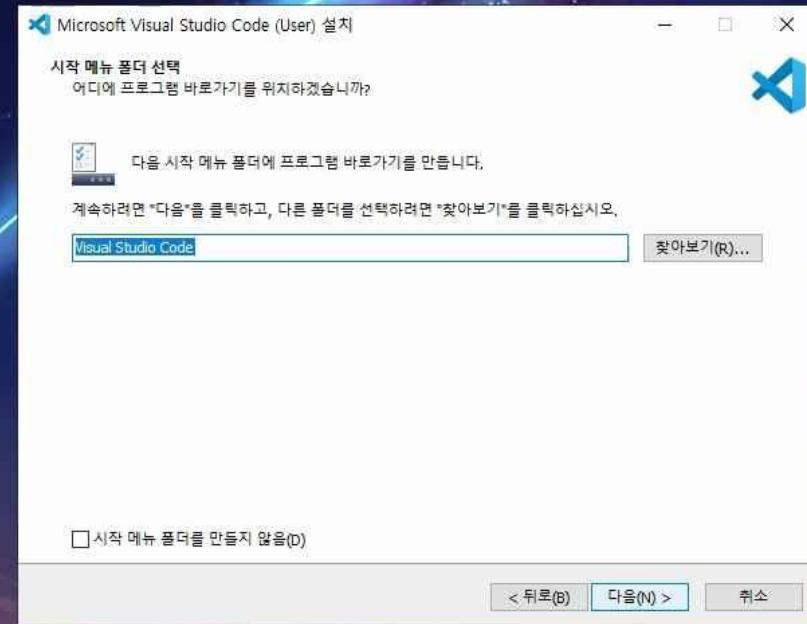
Want new features sooner?

Get the [Insiders build](#) instead.

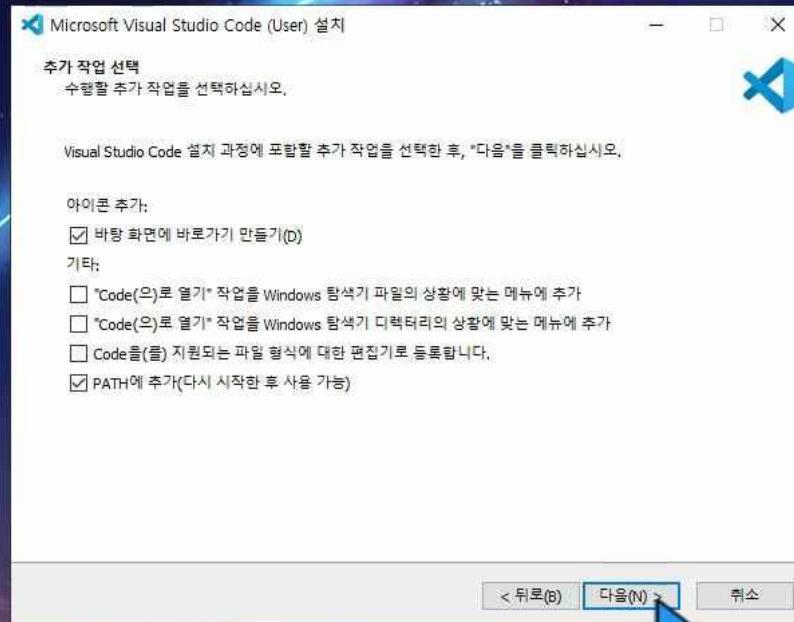




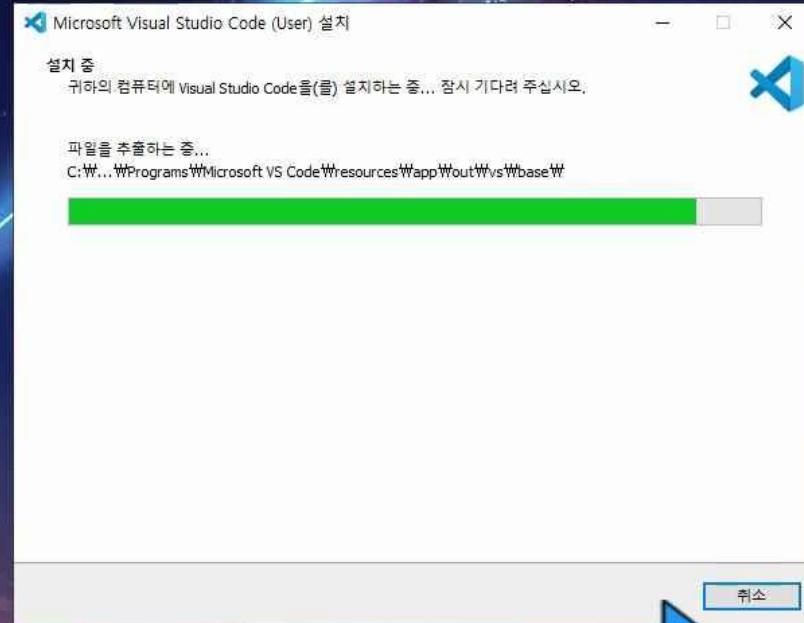
모두 기본값으로 두고 [다음]을 클릭하세요!

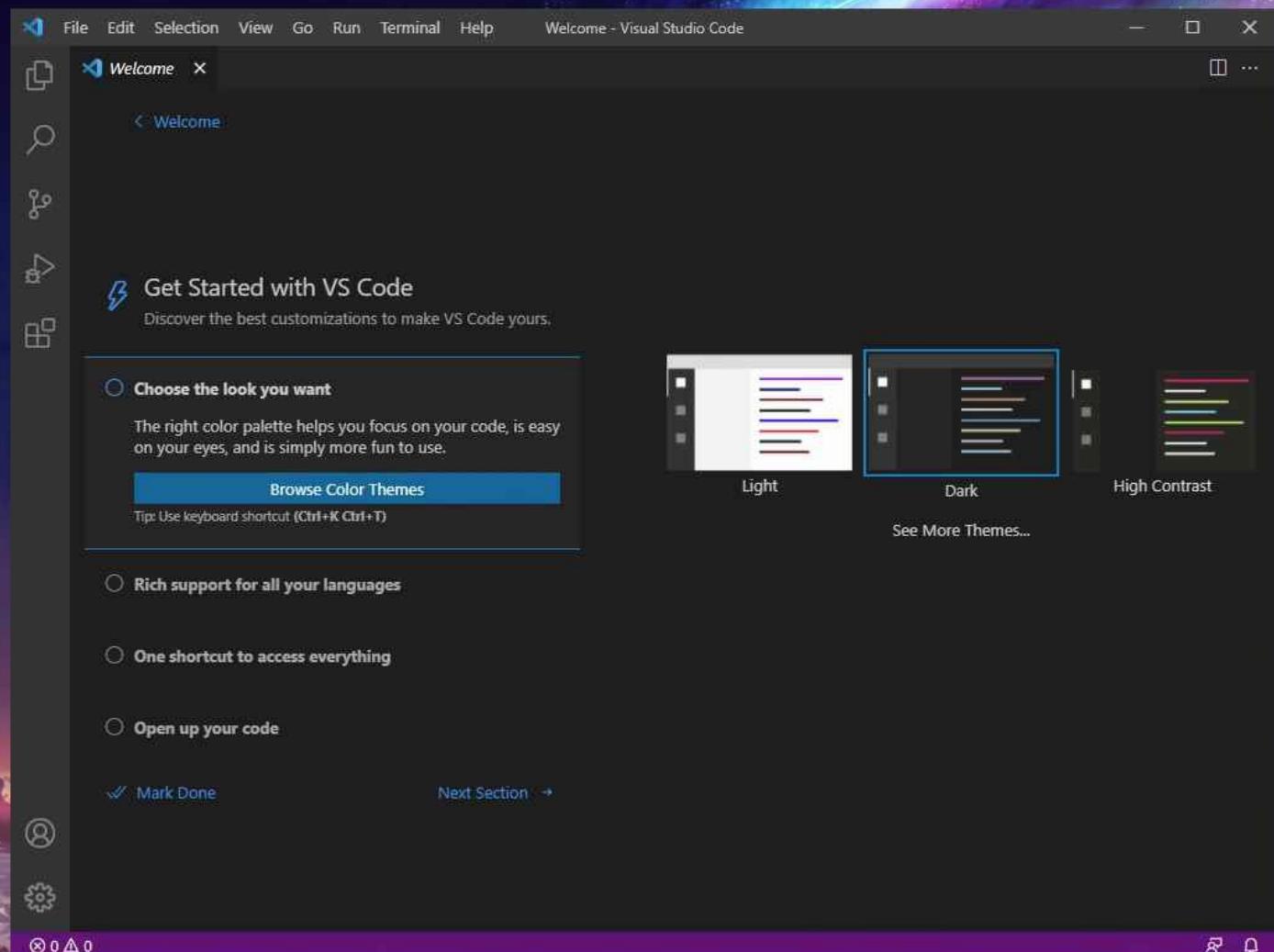


모두 기본값으로 두고 [다음]을 클릭하세요!



[다음]







VS Code

설정하기

- 폴더생성



C:\개인폴더\WEB

- 폴더열기



[파일] > [폴더 열기]





VS Code

설정하기

ctrl + , 단축키

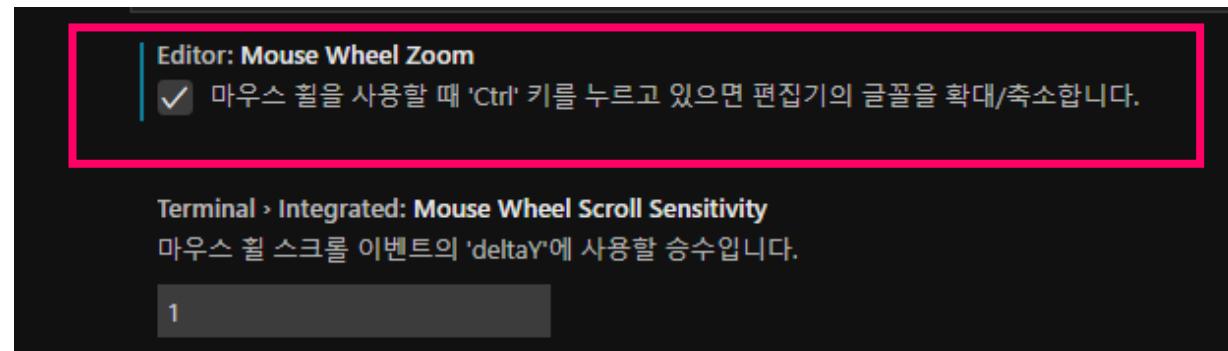
■ 폰트설정

☰ [파일] > [기본 설정] > [설정]

: mouse 검색

Mouse Wheel Zoom 체크

ctrl + 텍스트 크기를
쉽게 조절!





VS Code

확장 설치



- Korean Langunage Pack 한글 언어 패치
- Live Server 실시간 미리보기 확장프로그램
- Auto Close Tag 종료 태그 자동완성 기능
- Material Icon Theme 직관적인 아이콘 테마





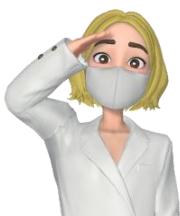
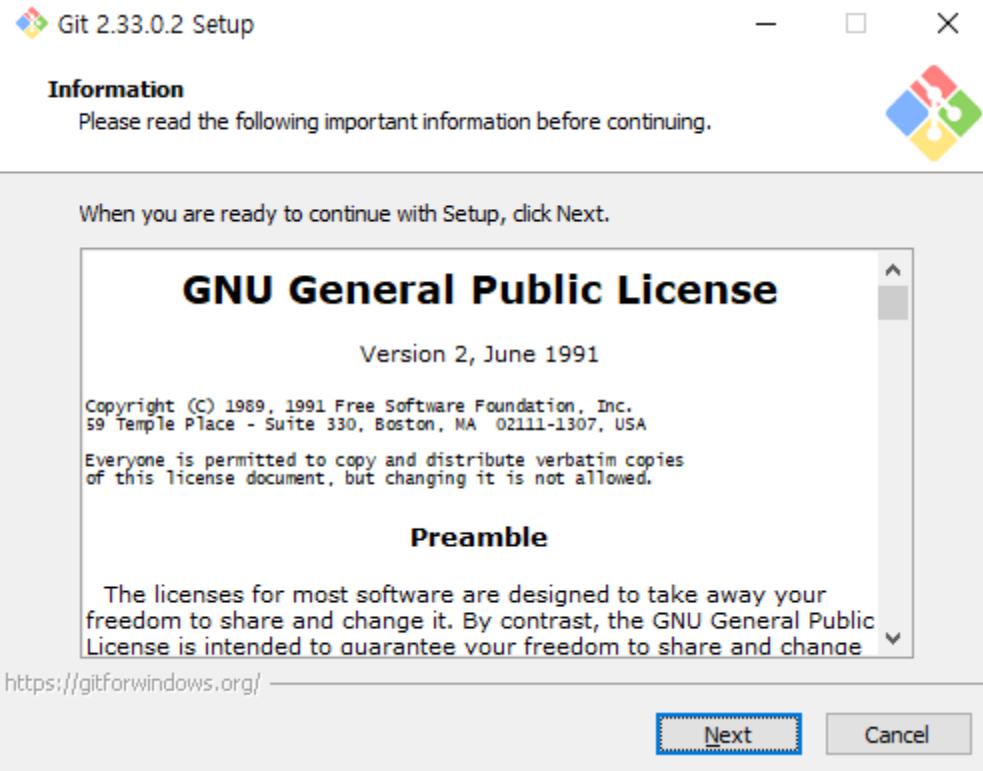
Git

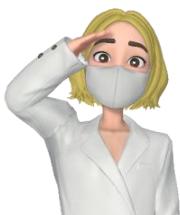
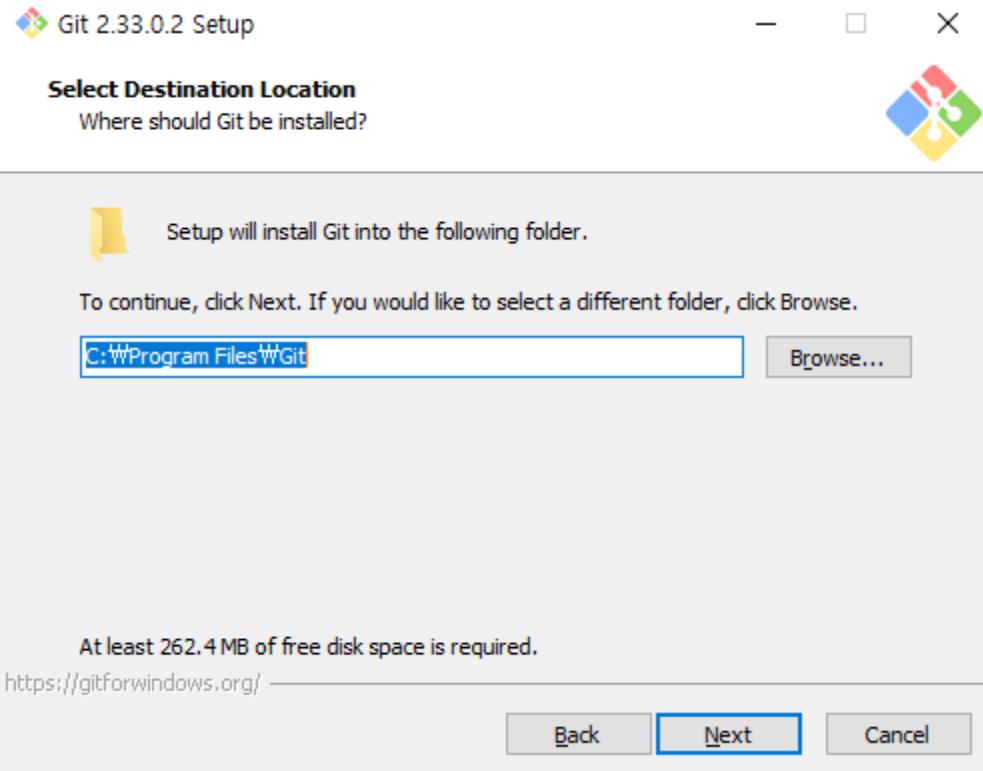
설치하기

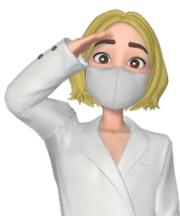
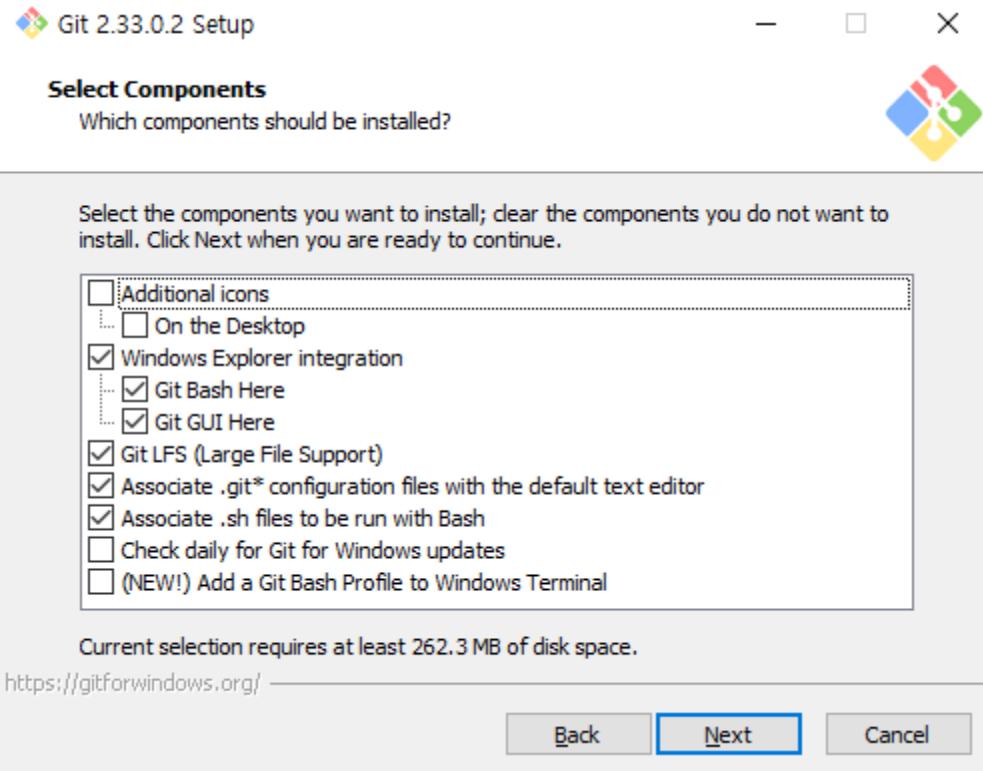
- 다운로드
 - <https://git-scm.com/downloads>
- GitHub 회원가입
 - <https://github.com/>
- 레포지토리(저장소) 만들기
- VS Code에서 Git 연동하기

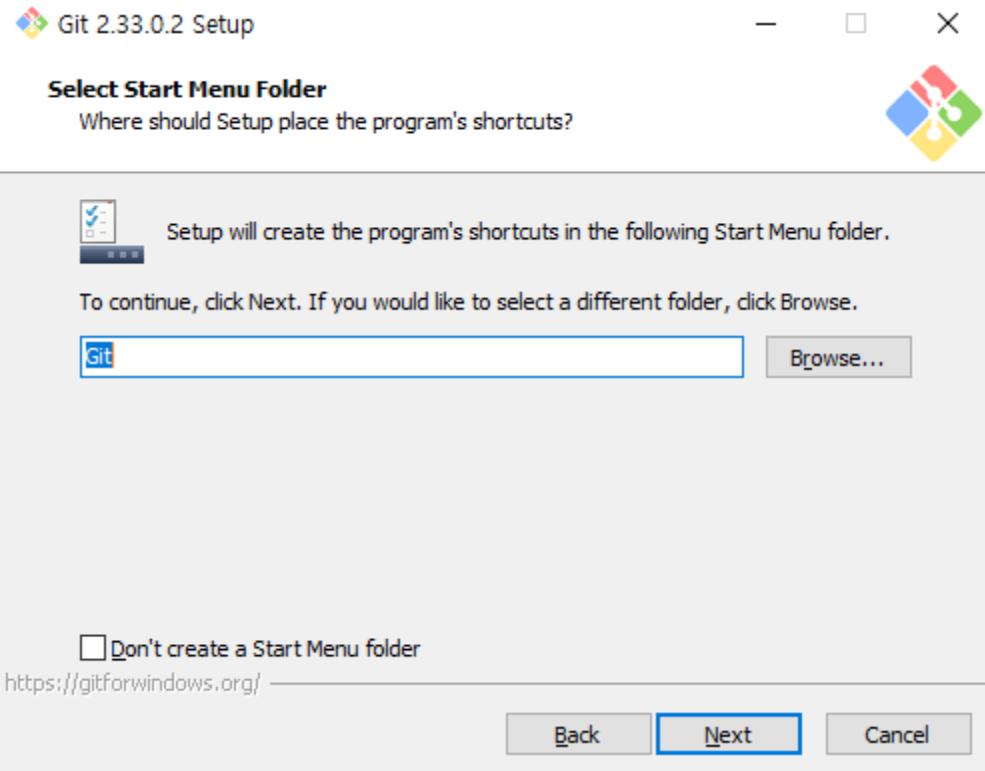


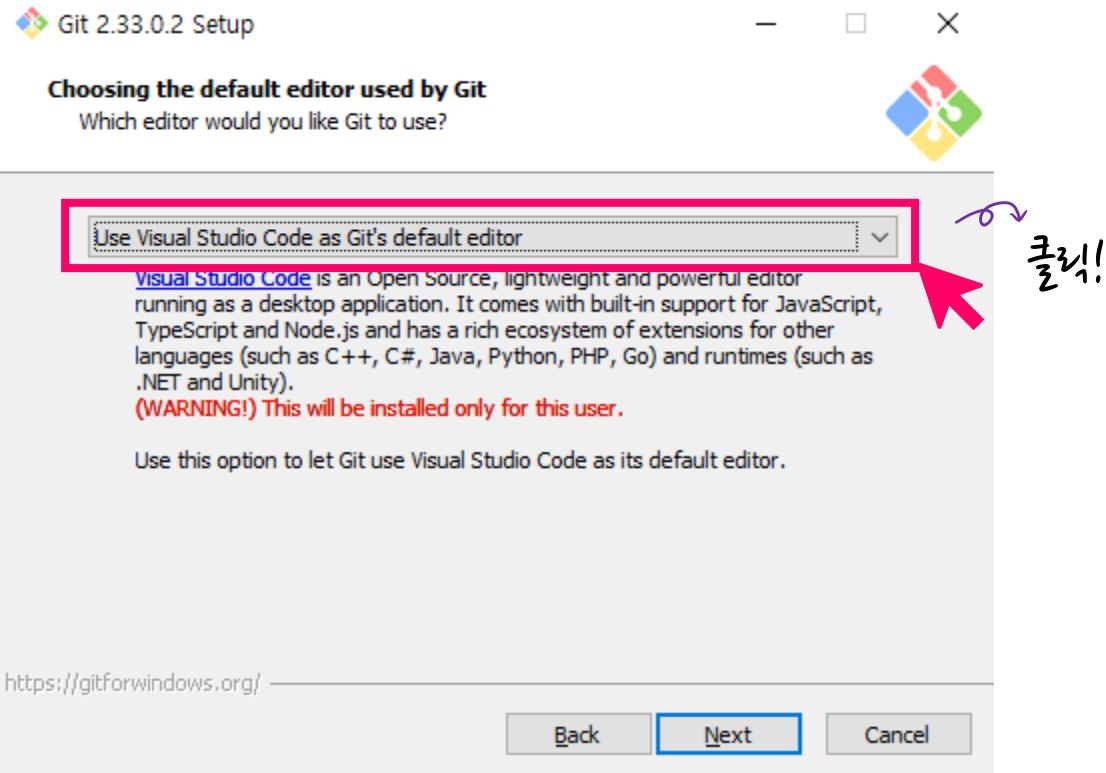


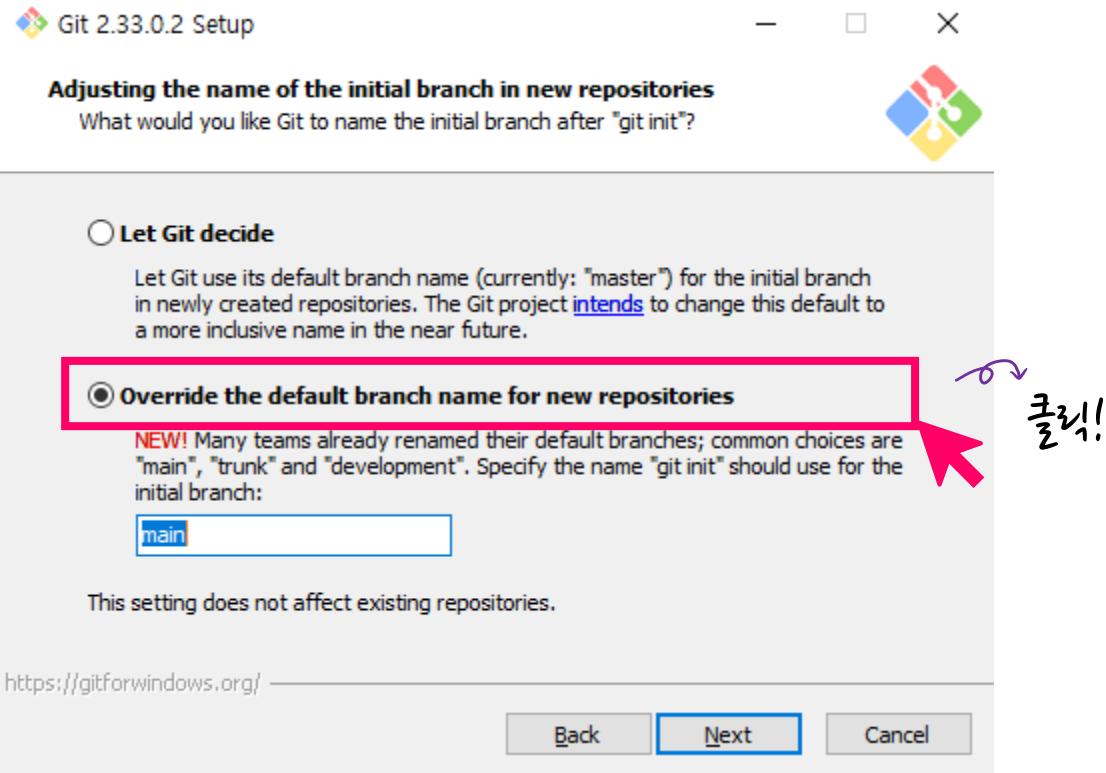


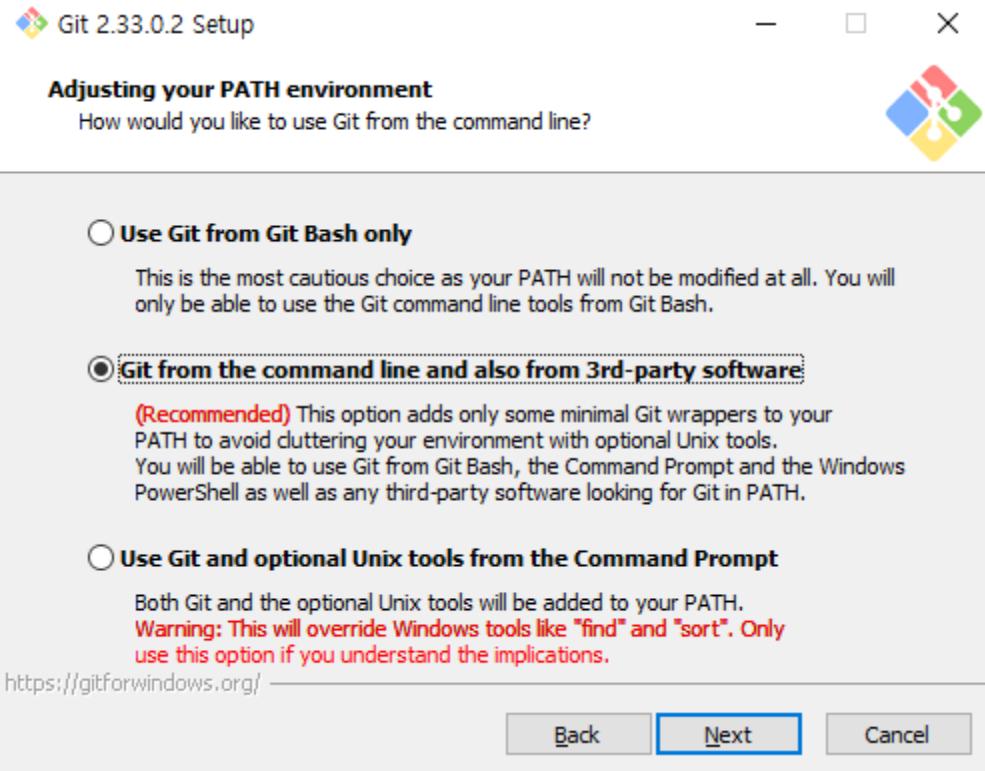


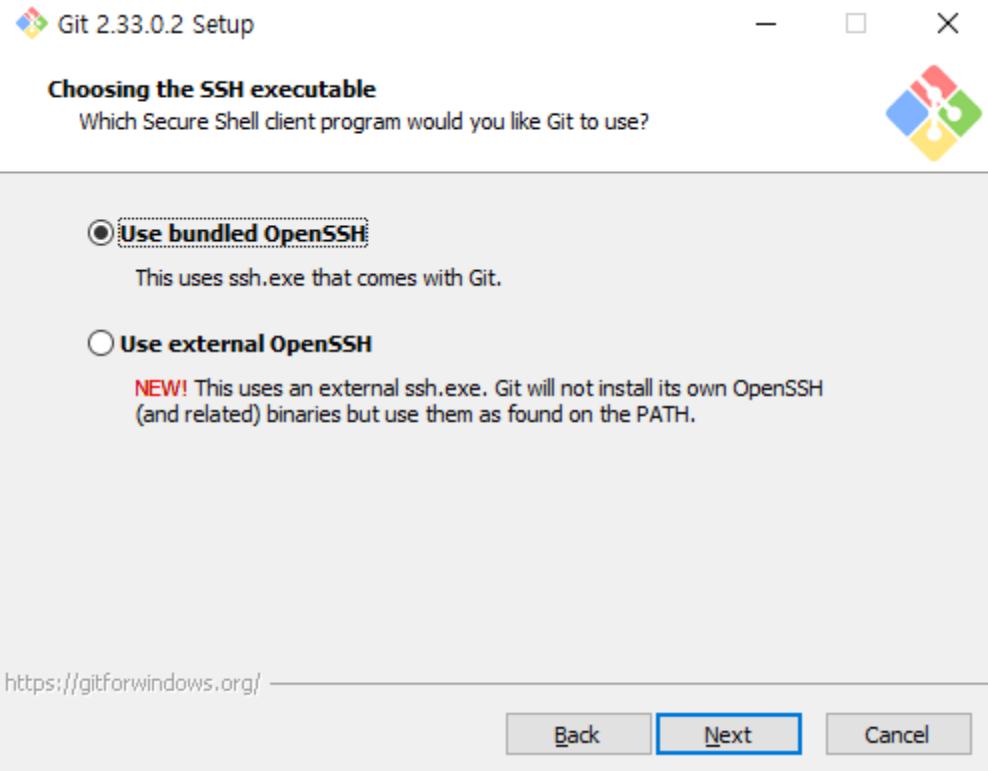


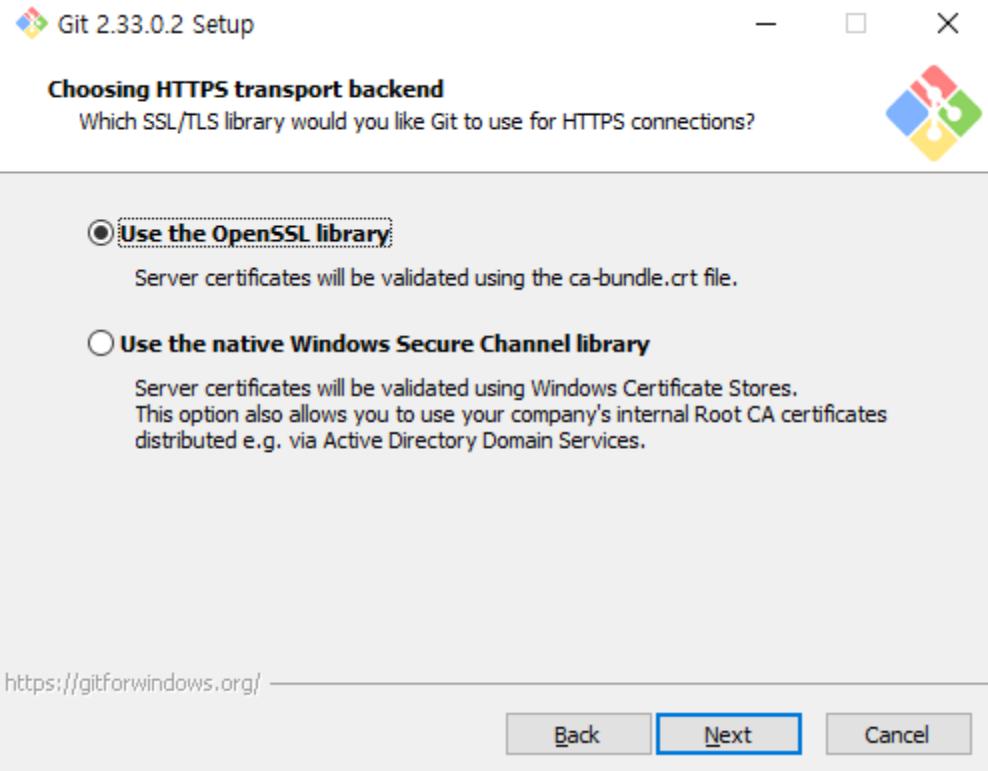


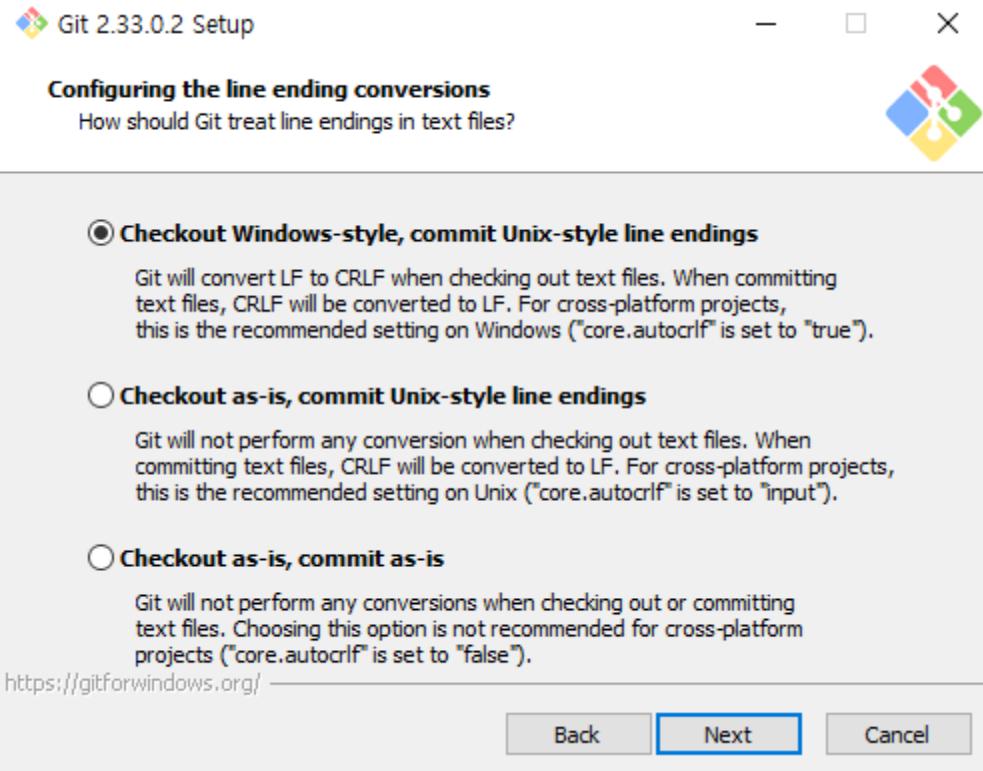


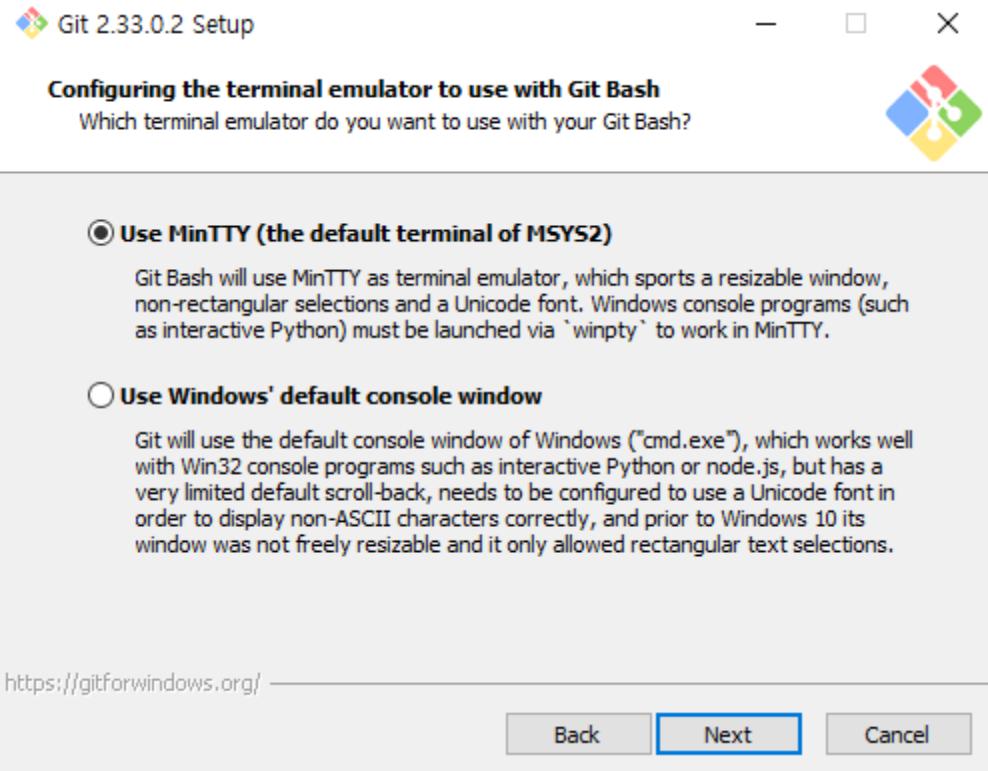


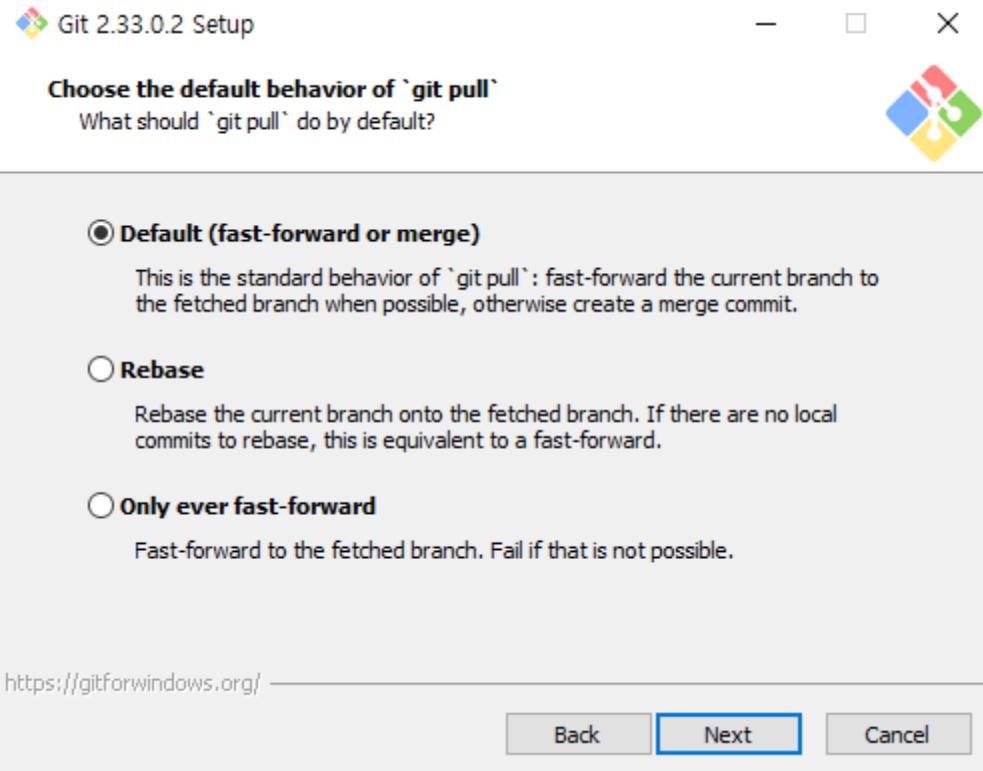


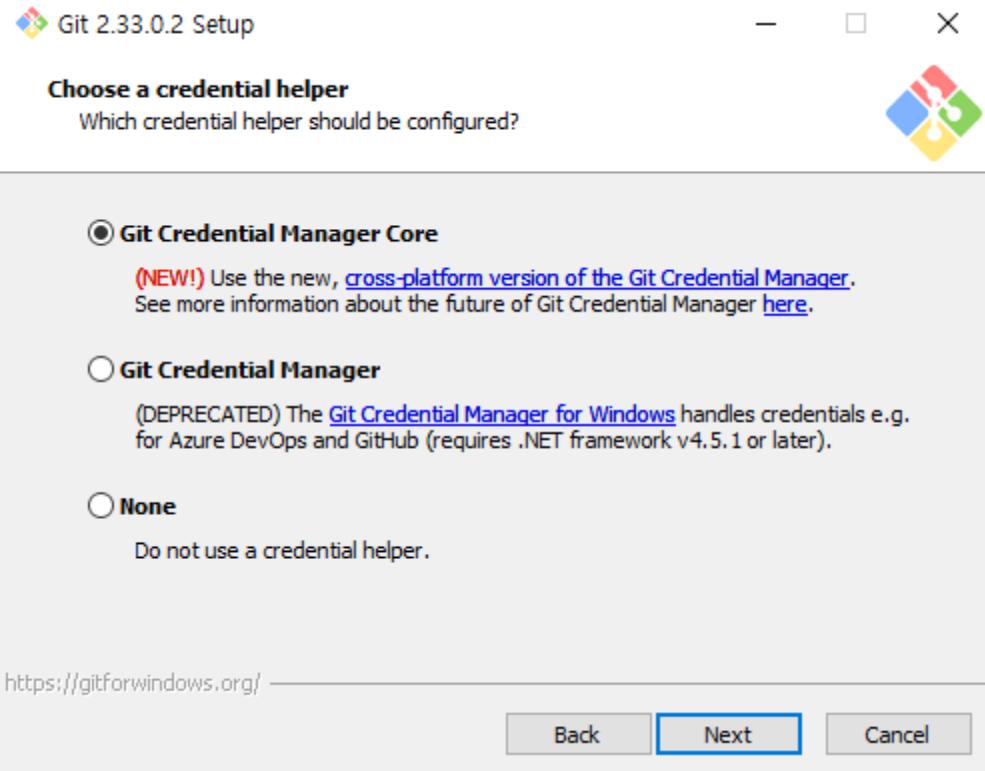


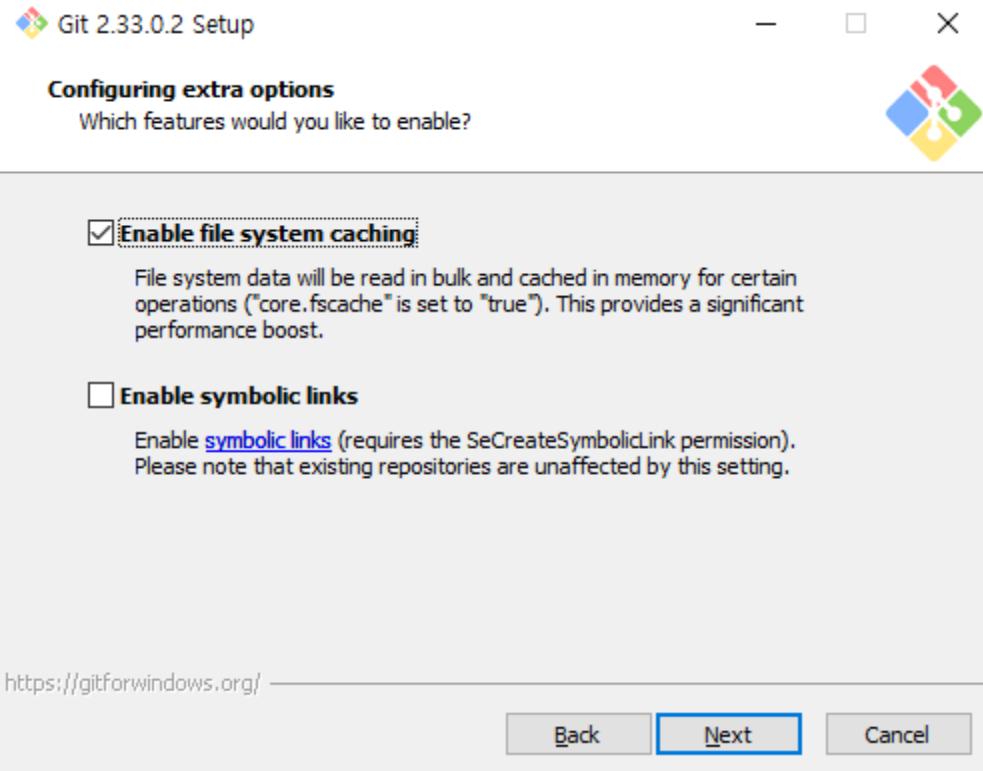


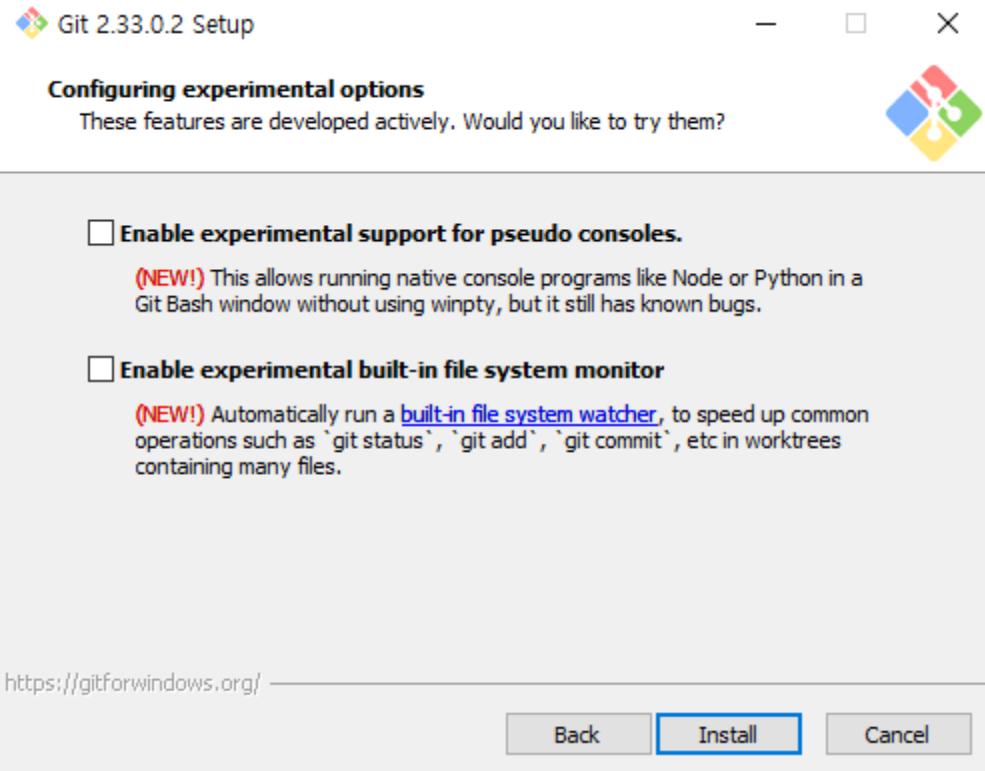


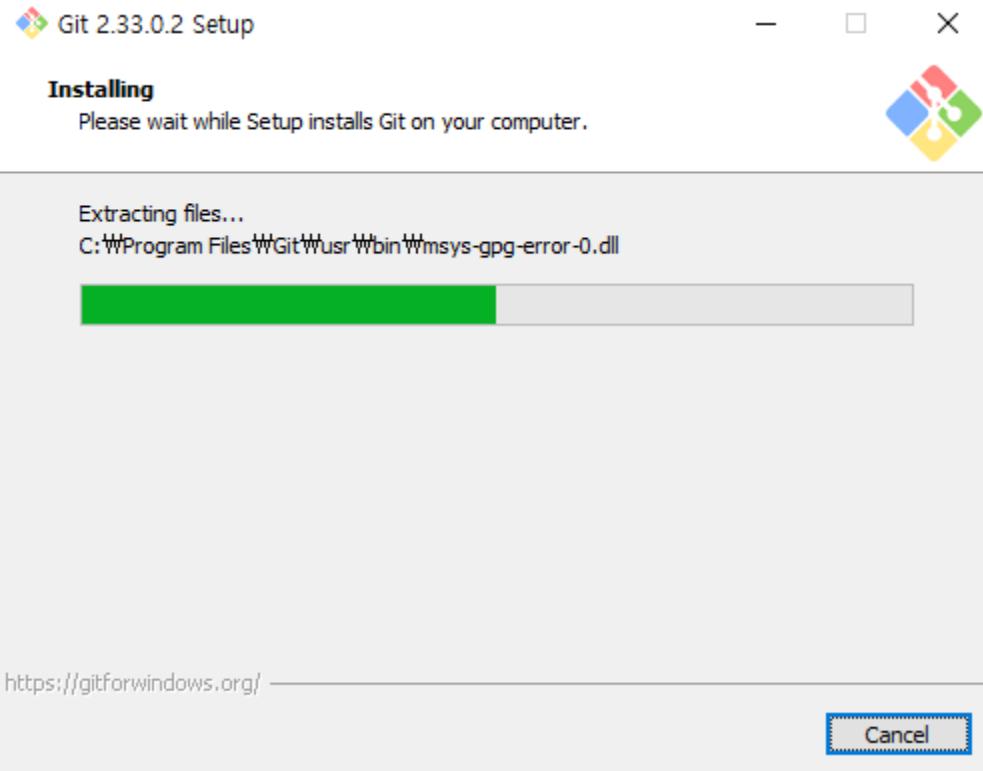














Git 2.33.0.2 Setup



Completing the Git Setup Wizard

Setup has finished installing Git on your computer. The application may be launched by selecting the installed shortcuts.



Click Finish to exit Setup.

- [Launch Git Bash](#)
- [View Release Notes](#)

[Finish](#)





Git

레포지토리(저장소) 만들기

- new
 - <https://github.com/new>
- 입력
 - 레포지토리 이름
 - 설명
 - 공개여부 (Public / Private)
 - 체크 : Add a README file





Owner *  wwwalohacampus *

Repository name *

Great repository names are short and memorable. Need inspiration? How about [probable-octo-adventure](#)?

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.

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](#).

Choose a license
A license tells others what they can and can't do with your code. [Learn more](#).

This will set  main as the default branch. Change the default name in your [settings](#).

Create repository





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

wwwwalohacampus / Public Unwatch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags Go to file Add file Code

wwwwalohacampus Initial commit 35b2e6e now 1 commit

README.md Initial commit now

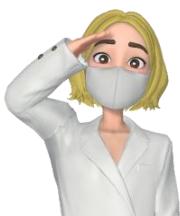
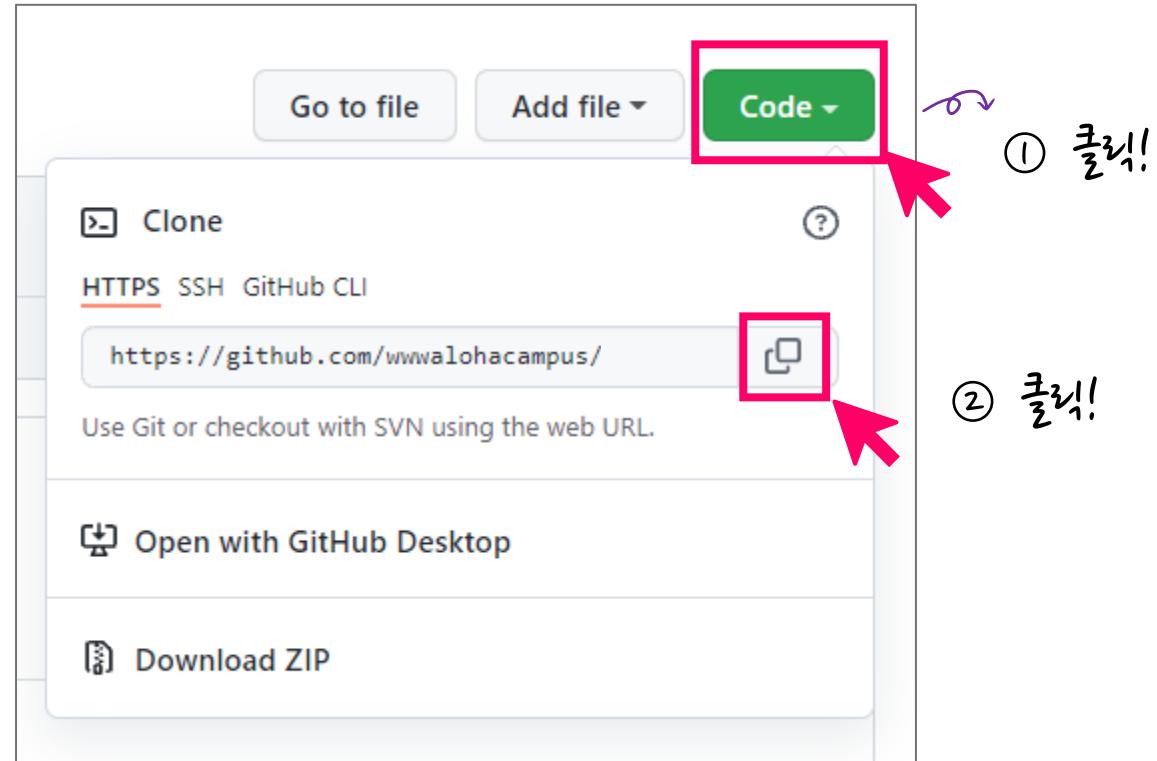
README.md

About Readme

Releases No releases published Create a new release

Packages No packages published Publish your first package







Git

원격 저장소 복제하기

☰ [보기] > [명령 팔레트] > git clone

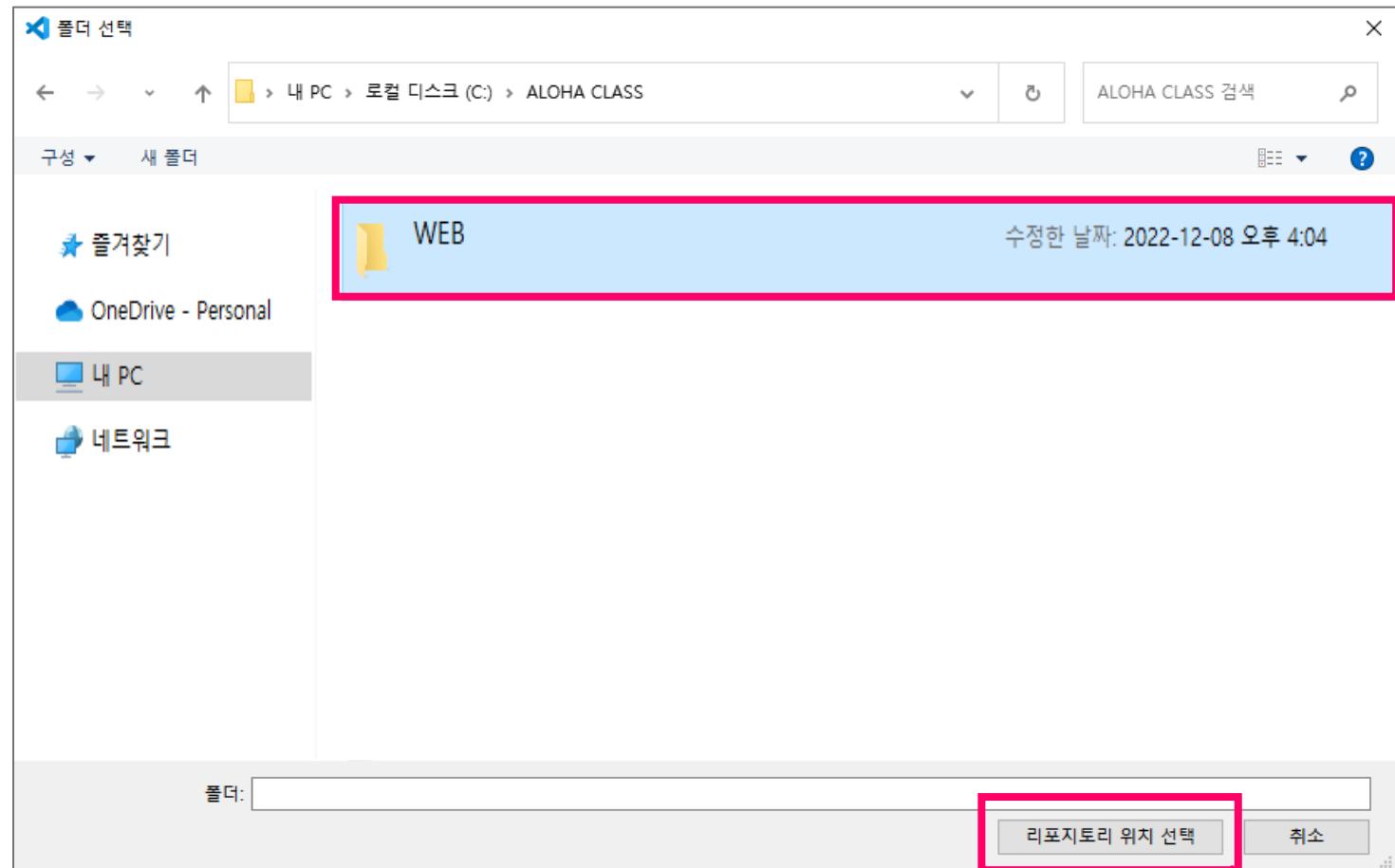
- 입력

📁 [레포지토리 URL]

https://github.com/wwwalohacampus/.git

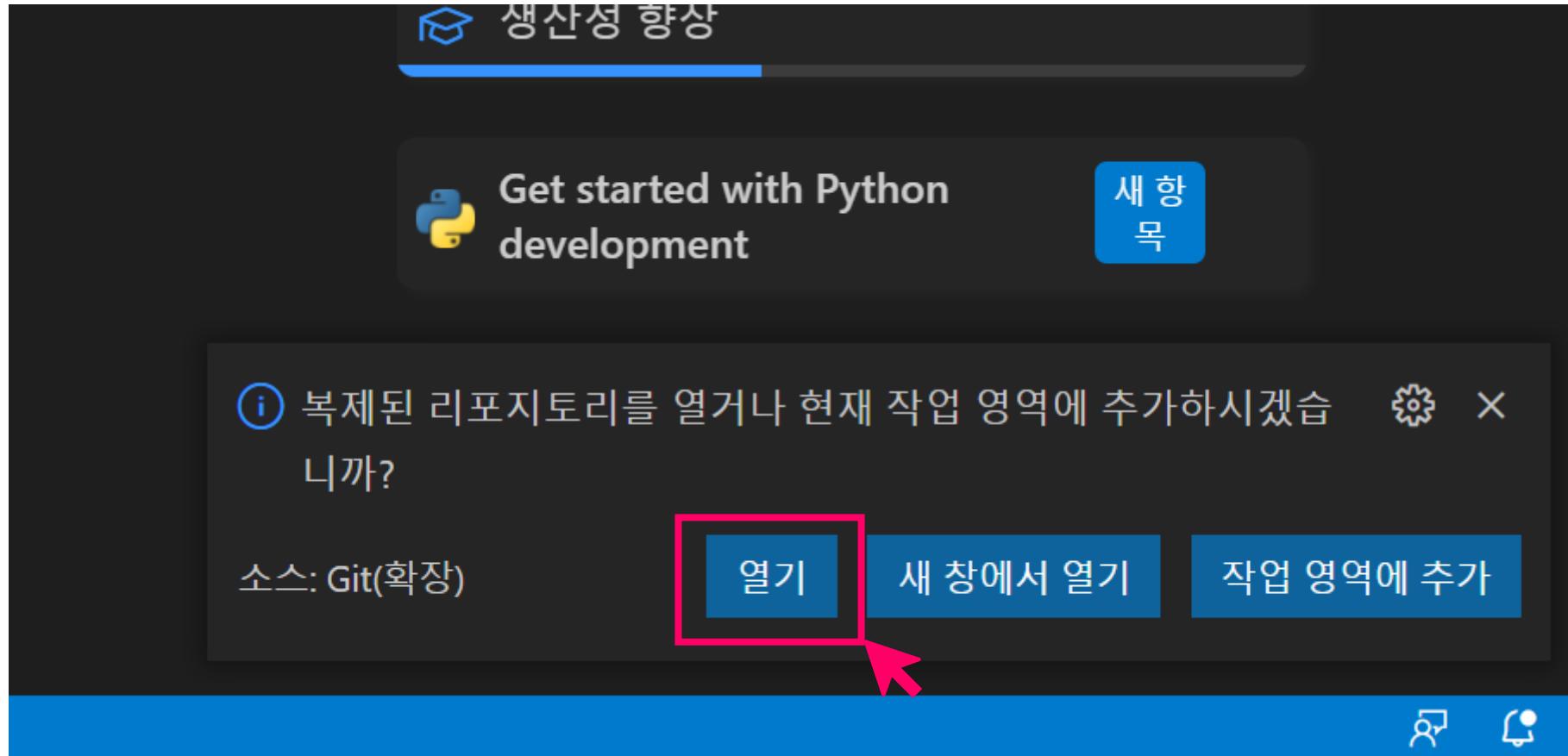
리포지토리 URL https://github.com/.git





클릭!







파일(F) 편집(E) 선택 영역(S) 보기(V) 이동(G) 실행(R) 터미널(T) 도움말(H) 시작하기 - sbs_web2 - Visual Studio Code [관리자] - □ ×

탐색기 ... 시작하기 x

열려 있는 편집기 x 시작하기

SBS_WEB2 README.md

새 파일...

파일 열기...

폴더 열기...

Git 리포지토리 복제...

시작

연습

VS Code 시작
VS Code를 자신의 것으로 만들기 위한 가장 적합한 사용자 지정을 발견하세요.

기본 사항 알아보기
VS Code 오른쪽으로 이동하여 필수 기능에 대한 개요를 확인하세요.

생산성 향상

Get started with Python development 새 항목

Dive into Java Development

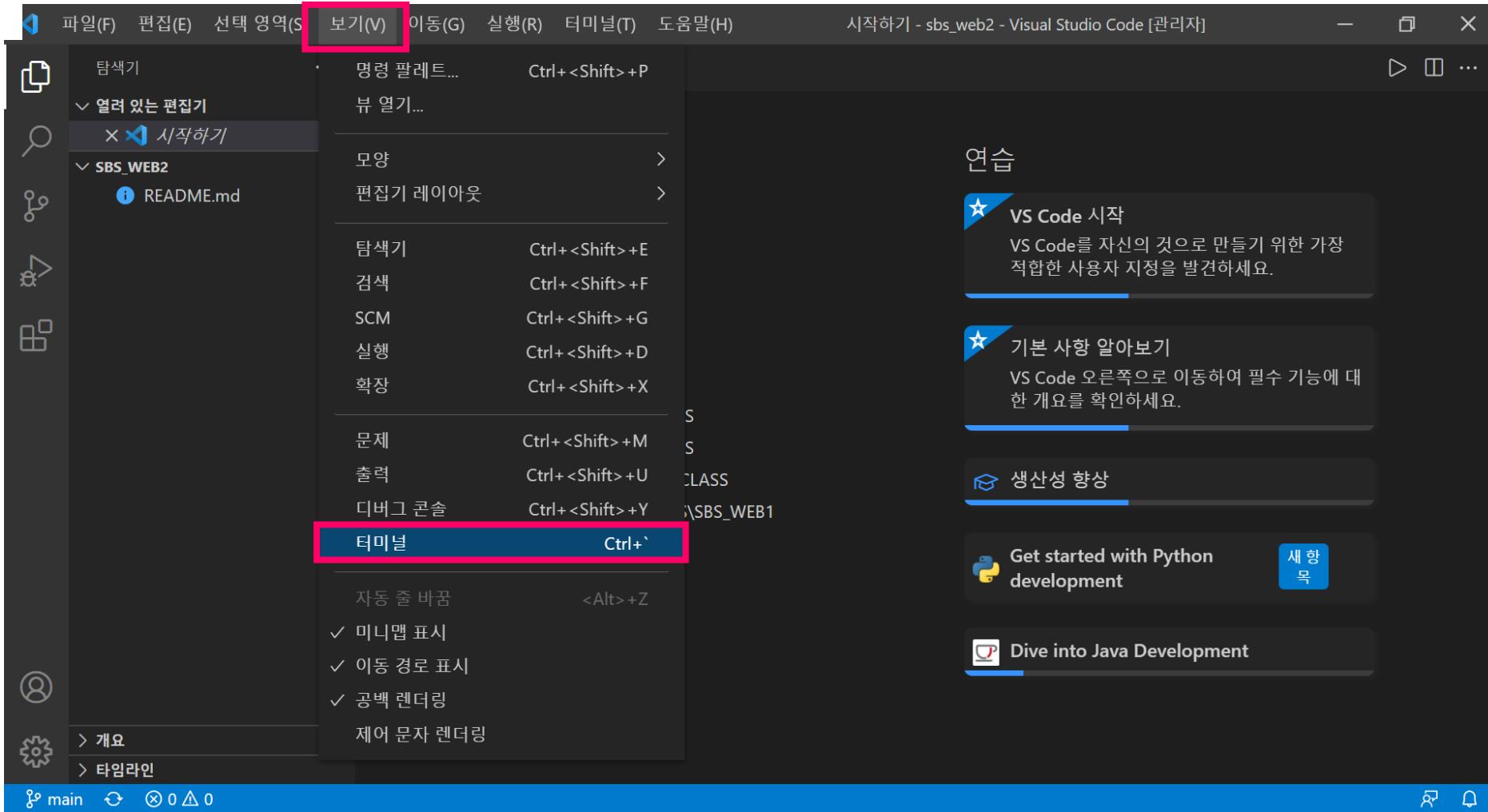
최근 항목

SBS_WEB2 C:\ALOHA CLASS
SBS_WEB1 C:\ALOHA CLASS
Vue.js프로젝트 C:\ALOHA CLASS
sbs_web1 C:\ALOHA CLASS\SBS_WEB1
Home C:\ALOHA CLASS
More...

개요 타임라인

main 0 0 0 0



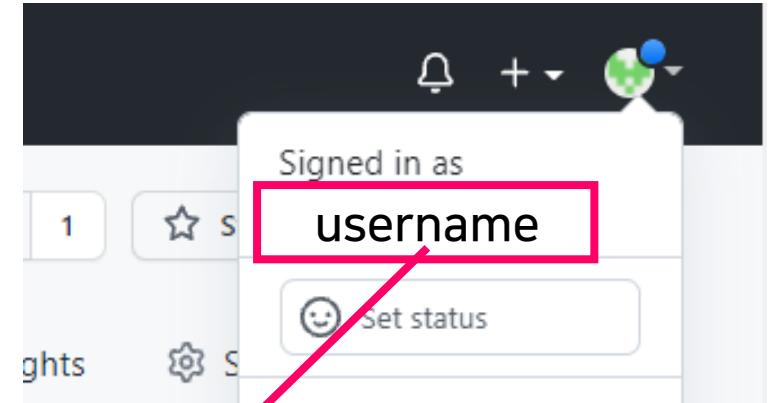




깃허브!

깃허브 사용 방법, 이메일 설정을 넣어주세요😊

```
s_WEB2\sbs_web2> git config --global user.name username  
s_WEB2\sbs_web2> git config --global user.email email  
s_WEB2\sbs_web2>
```



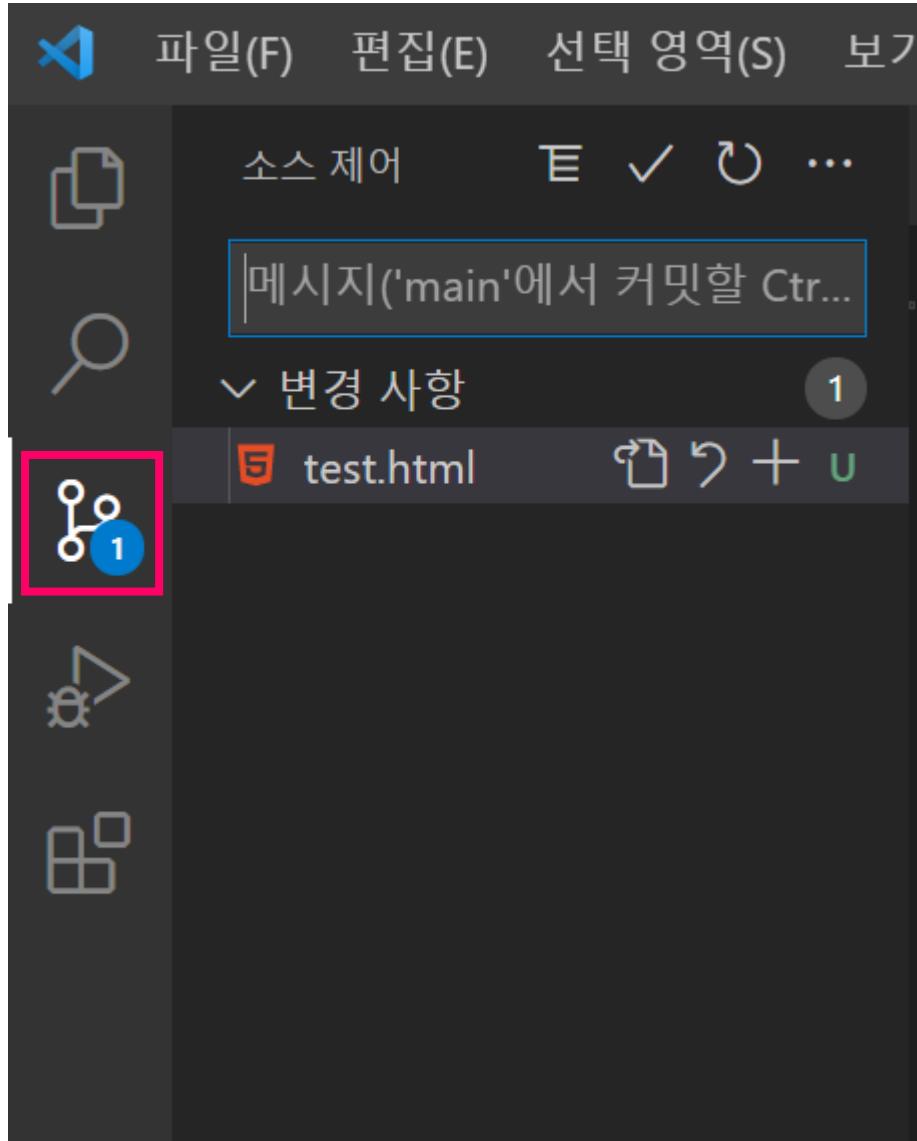


테스트 파일 생성 : test.html

The screenshot shows the Microsoft Edge DevTools interface. The title bar says "시작하기" and "test.html". The left sidebar shows the file tree with "열려 있는 편집기" containing "시작하기" and "test.html", and a folder "SBS_WEB2" containing "README.md" and "test.html". The main pane displays the DOM structure of "test.html":

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <h1>테스트</h1>
```



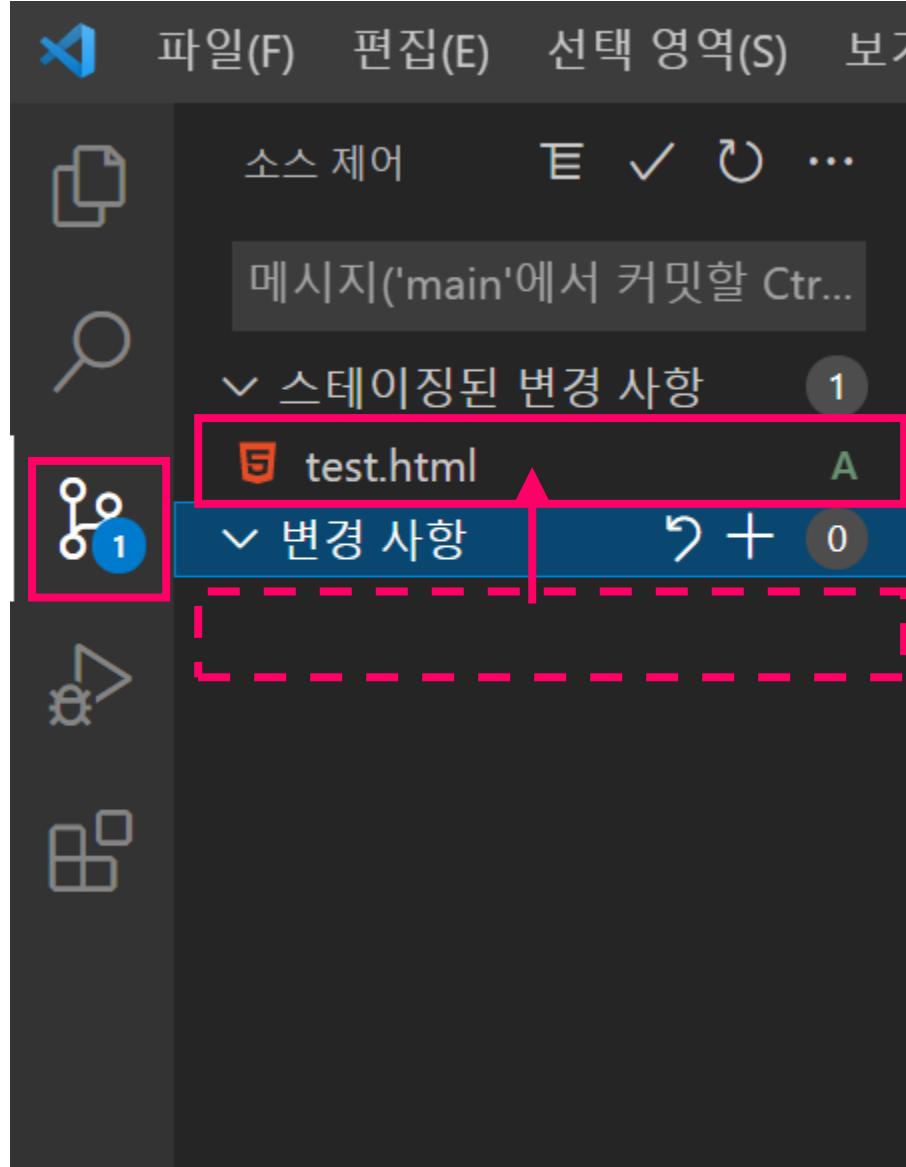


변경 사항이 있으면,

[소스 제어]

메뉴에 변경된 파일 숫자가 붙어요😊





변경 사항을 반영하고 싶은 파일을
스테이지에 올려주세요!

변경된 파일에 마우스를 올려 +
또는

변경 사항 메뉴에 마우스를 올려 + (전체)

변경된 파일이 스테이지에 올라갔습니다!
(스테이지에 올라간 파일만 commit 합니다.)





커밋 버튼 클릭!

A screenshot of a code editor interface. At the top, there is a menu bar with Korean labels: 파일(F), 편집(E), 선택 영역(S), 보기(V), 이동(G), 실행(R), 터미널(T), and 도움말(H). Below the menu is a toolbar with icons for file operations like '소스 제어' (Source Control) and a commit button with a checkmark. A dropdown menu is open, showing the text '테스트' (Test) which is highlighted with a pink rectangle. Below the dropdown, a message says '커밋 메시지를 제공하세요.(확인하려면 'Enter' 키를 누르고,' followed by a list of four lines of HTML code:

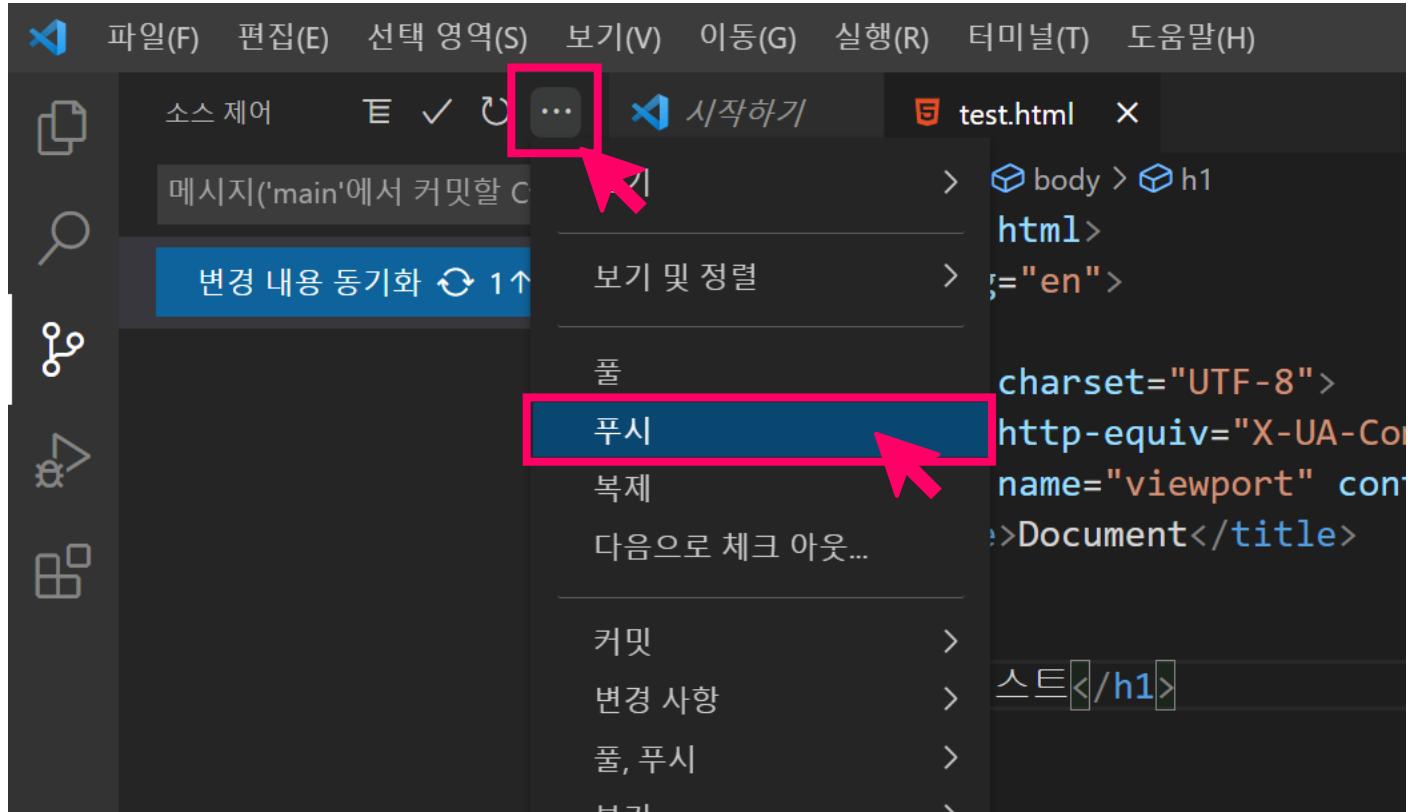
```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 |   <meta charset="UTF-8">
```

반드시, 커밋 메시지를 작성해주세요!



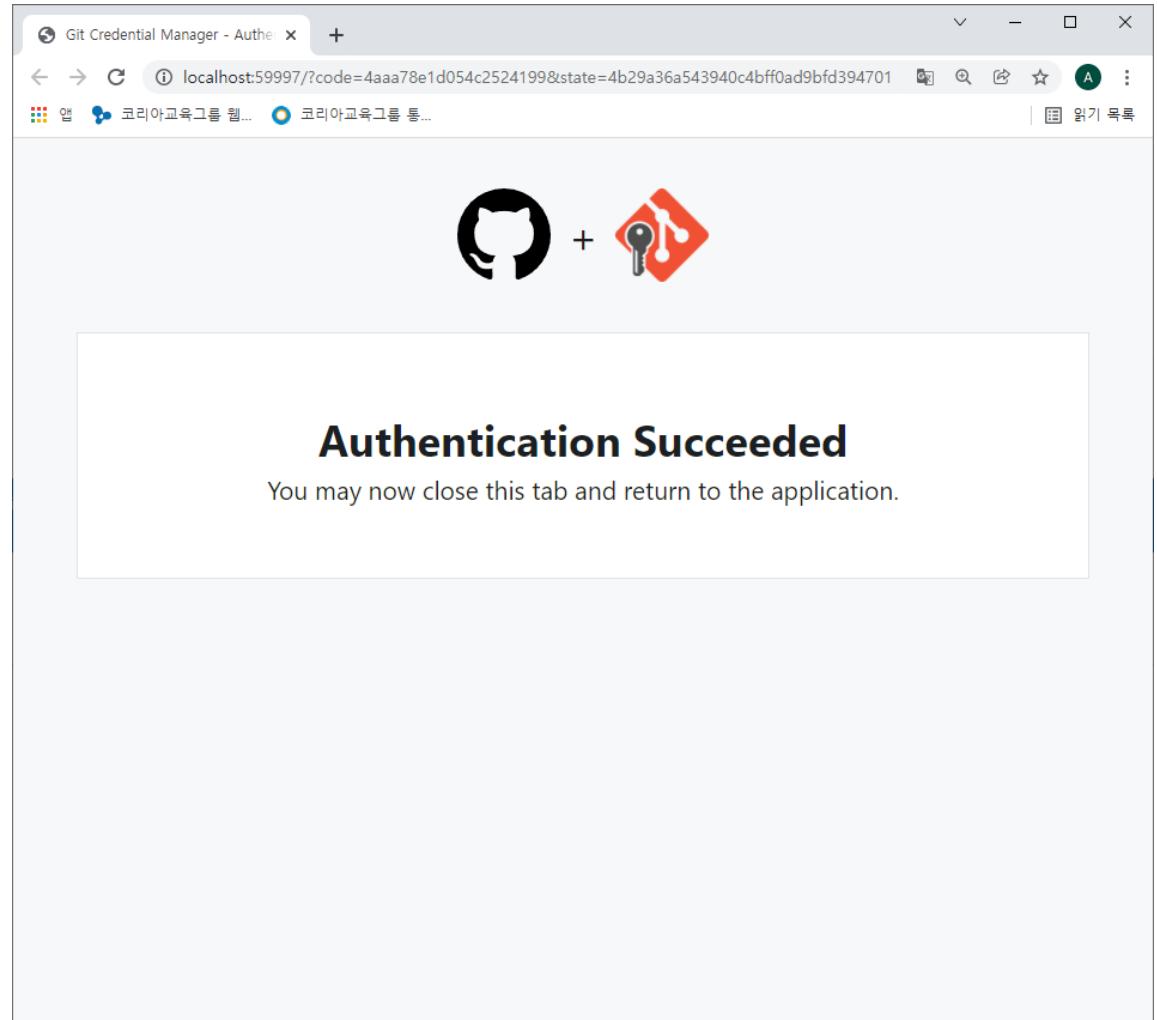
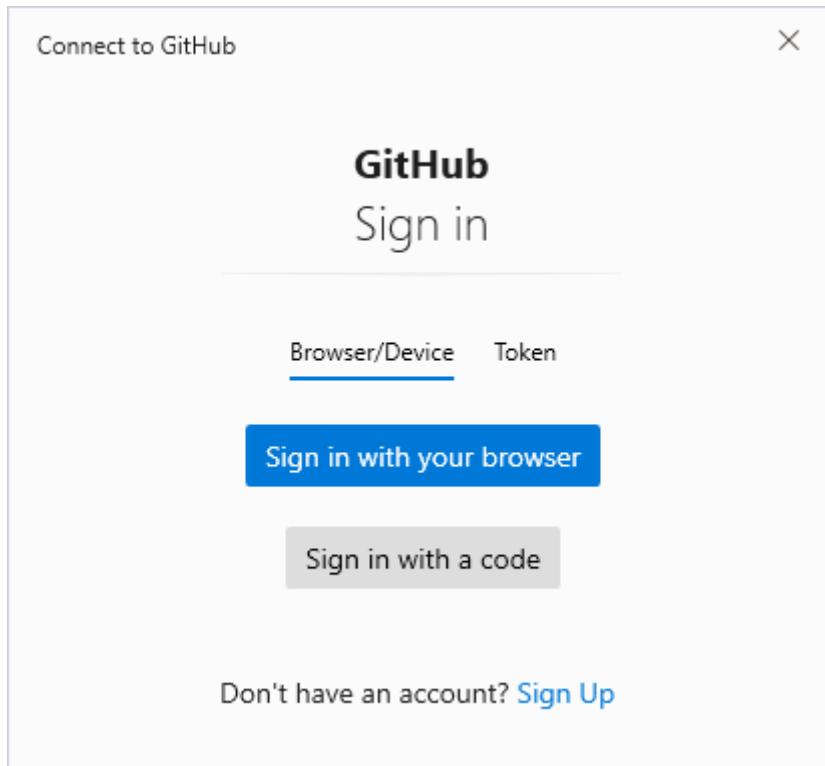


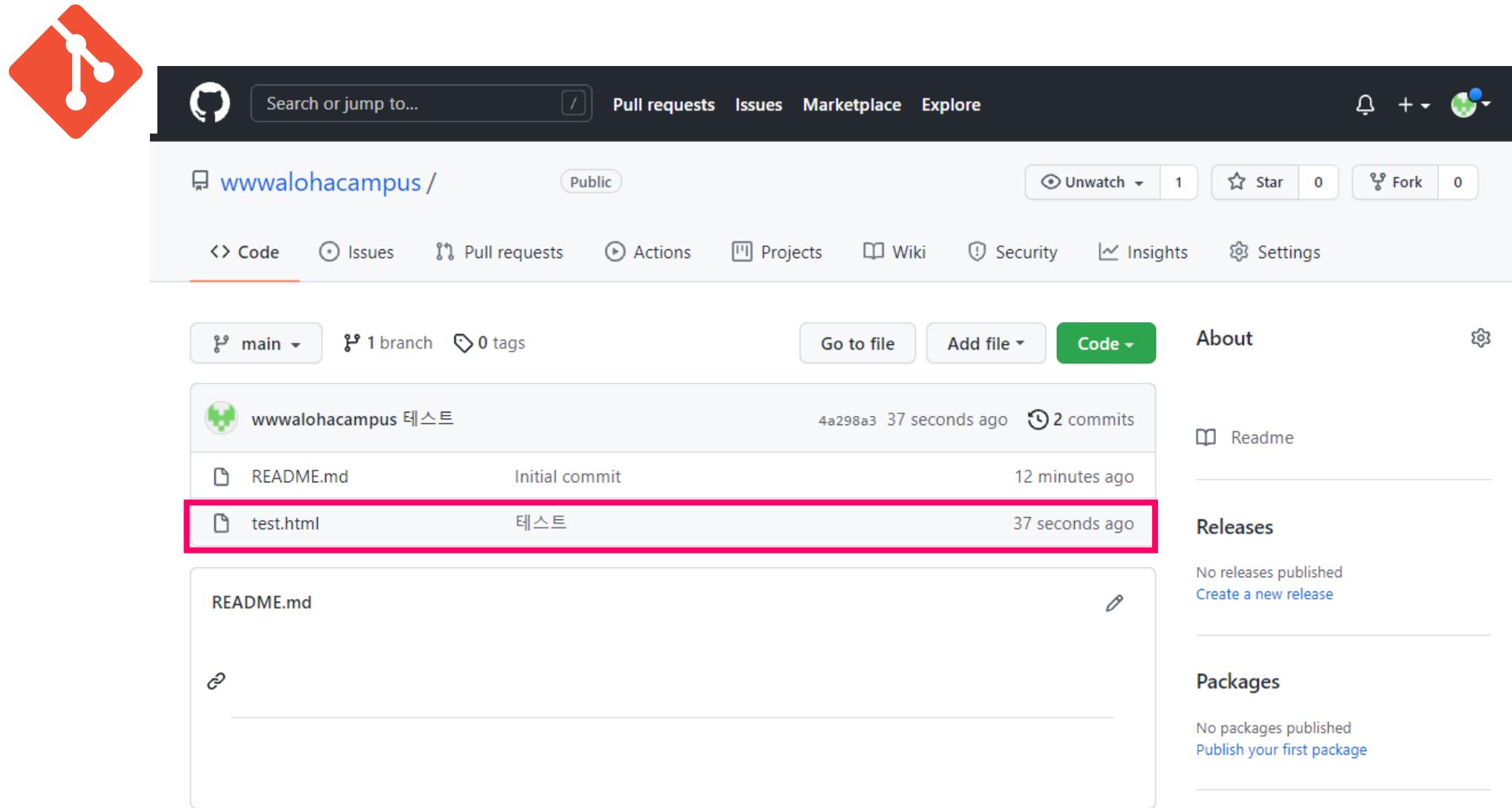
푸시 클릭!



이제 Github 원격 저장소에 소스파일을 업로드 됩니다!







wwwwalohacampus / Public

Unwatch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags Go to file Add file Code

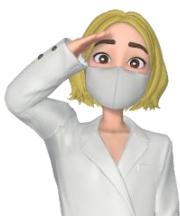
wwwwalohacampus 테스트 4a298a3 37 seconds ago 2 commits

README.md Initial commit 12 minutes ago

test.html 테스트 37 seconds ago

README.md

업로드된 소스파일을 확인할 수 있습니다.





Git

VS Code 에서 Git 연동하기

- 소스제어

- git clone - 원격 저장소의 파일들을 로컬 저장소에 복제
- Commit - 코드 추가/변경과 메시지를 저장소에 기록
- Push - 추가/변경 파일을 원격 저장소에 업로드
- Pull - 추가/변경 파일을 원격 저장소에서 다운로드





Git

원격 저장소 복제하기

☰ [보기] > [명령 팔레트] > git clone

- 입력

📁 [레포지토리 URL]

https://github.com/wwwalohacampus/.git

리포지토리 URL https://github.com/.git





Git

커밋 하기 (Commit)



- 소스제어 > 변경 사항 > + (변경 내용 스테이징)
- ✓ (커밋) > 메시지 입력

클릭!





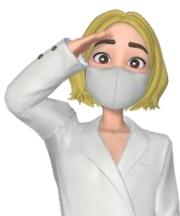
Git

푸시 하기 (Push)



- 소스제어 > … (더보기) > 푸시

클릭!





Git

풀하기 (Pull)



- 소스제어 > … (더보기) > 풀



클릭!





Git

학원 & 집에서 이어서 코드 공부하기



- 수업 끝날 때
 - Commit & Push

- 집에서 복습할 때

- 최초로 1번만, Clone
- 다음부터는 Pull 만
- 집에서 과제나 복습하고 나서 Commit & Push

클릭!



HTML 개념과 구조

#2



HTML 개념과 구조

Hypertext Markup Language

- 웹 페이지를 만들기 위한 표준 **마크업 언어**
- 웹 페이지의 구조를 설명합니다.
- 일련의 **element(요소)**로 구성됩니다.
- element(요소)는 브라우저에 콘텐츠를 표시하는 방법을 알려줍니다.





HTML

Hypertext Markup Language

웹페이지가 어떻게 구조화되어 있는지
브라우저로 하여금 알 수 있도록 하는
마크업 언어



```
<!DOCTYPE html>
<html>
<head>
    <title>Page Title</title>
</head>
<body>

    <h1>My First Heading</h1>
    <p>My first paragraph.</p>

</body>
</html>
```



HTML 기본 구조



<!DOCTYPE html>

HTML5 문서임을 정의합니다

<html>

HTML 페이지의 루트 요소입니다

<head>

HTML 페이지에 대한 메타 정보를 포함합니다

<title>

페이지의 제목을 지정합니다

<body>

문서의 본문을 정의합니다



- 검색 엔진 키워드
- 웹 페이지 설명
- 문서의 저작
- 뷰포트 ...



```
<head>

<meta charset="UTF-8">      인코딩 방식

<meta name="keyword" content="HTML, CSS">      검색 엔진 키워드

<meta name="description" content="WEB 수업 페이지">      웹 페이지 설명

<meta name="author" content="SBS Academy">      문서의 저자

<meta name="viewport" content="width=device-width, initial-scale=1.0">      뷰포트

<title>HTML 개념과 구조</title>      페이지 제목

</head>
```

charset	해당 문서의 문자 인코딩 방식 을 명시함.
name	메타데이터 를 위한 이름을 명시함.
content	관련된 값 을 명시함.





인코딩 방식

- **UTF-8**

- 전 세계의 모든 문자를 표현할 수 있는 인코딩
- 사용 권장



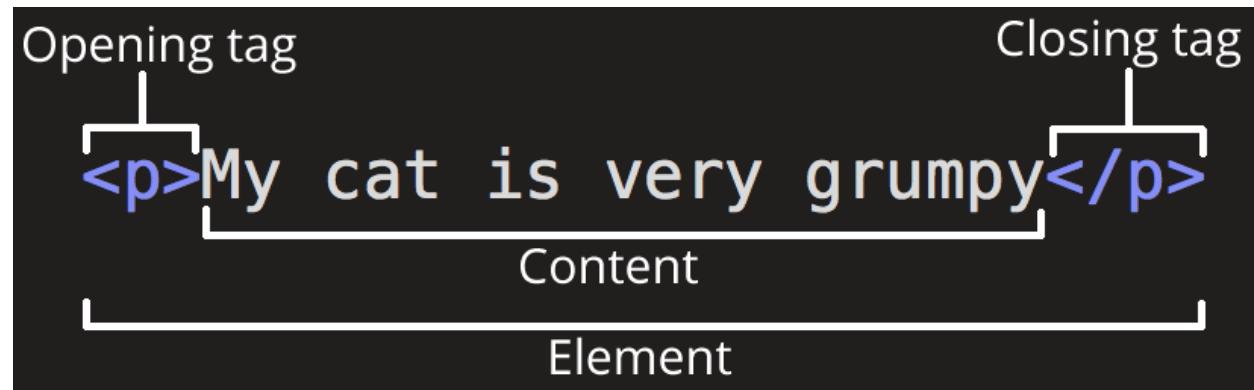
주로 사용

- **EUC-KR**

- 한글, 한자, 영문 표현 가능
- 일본식 한자, 중국어 간체 표현 불가



HTML 요소(Element)



요소 = 시작태그 + 내용 + 종료태그



빈 요소

empty element

종료태그가 없고 content 를 가지고 있지 않은 요소

```
area hr param base br col command embed  
img source input keygen link meta track wbr
```





HTML 작성 주의사항

- 태그 이름은 대소문자 구별하지 않는다.
- 하나의 요소가 다른 요소에 포함 가능하다.
- 시작태그와 종료태그 사이의 연속된 공백은 하나의 공백으로 취급한다.



속성(Attribute)

```
<tag 속성="속성 값">요소 내용</tag>
```

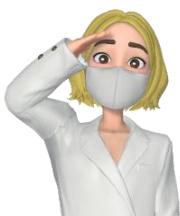
속성은 요소의 추가적인 정보를 제공한다.



공통 속성

모든 HTML 요소에서 지원되는 속성

속성	설명
id	하나의 요소를 지정하는 고유 식별 속성
class	여러 요소를 지정하는 그룹 식별 속성
style	스타일을 지정하는 속성
title	요소에 대한 정보를 나타내는 속성 (툴팁)



주석

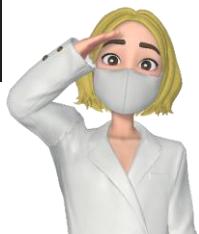
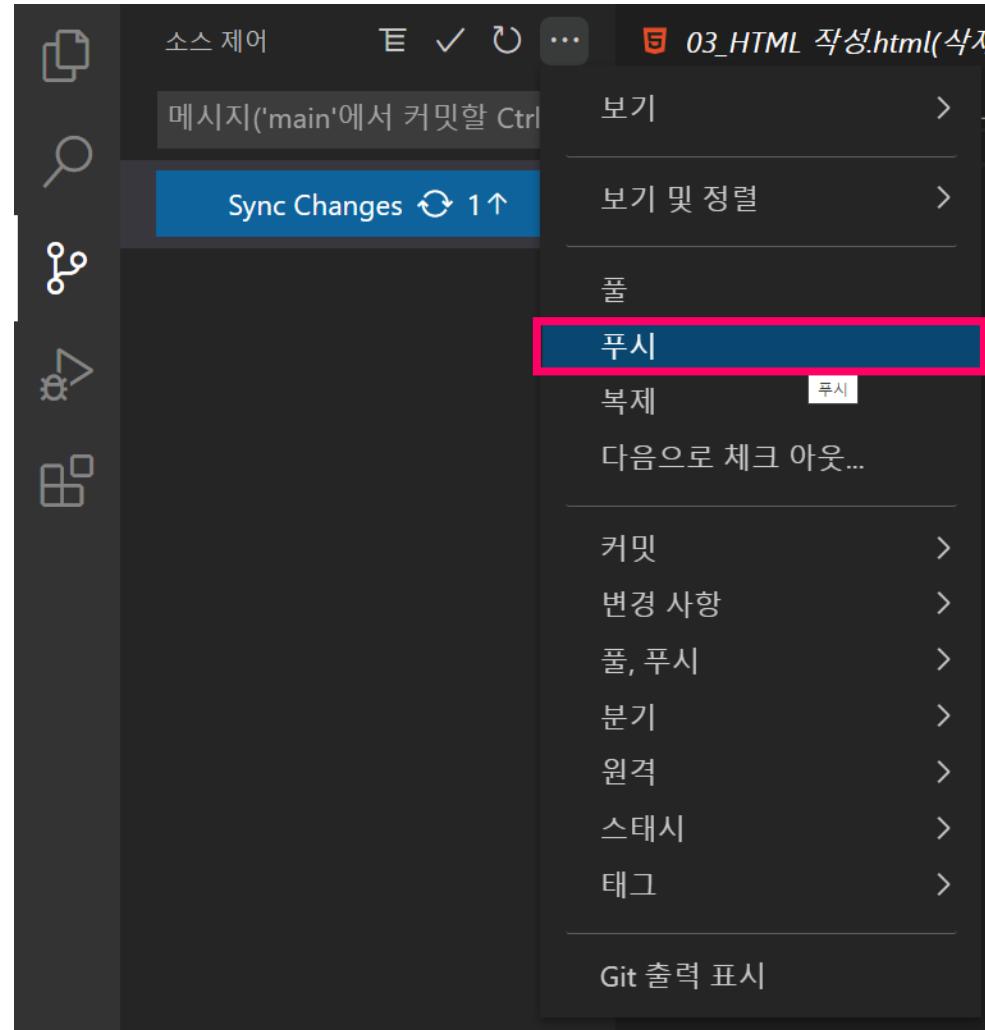
<!-- -->

브라우저에 표시되지 않고 HTML 코드를 설명하는 내용이다.



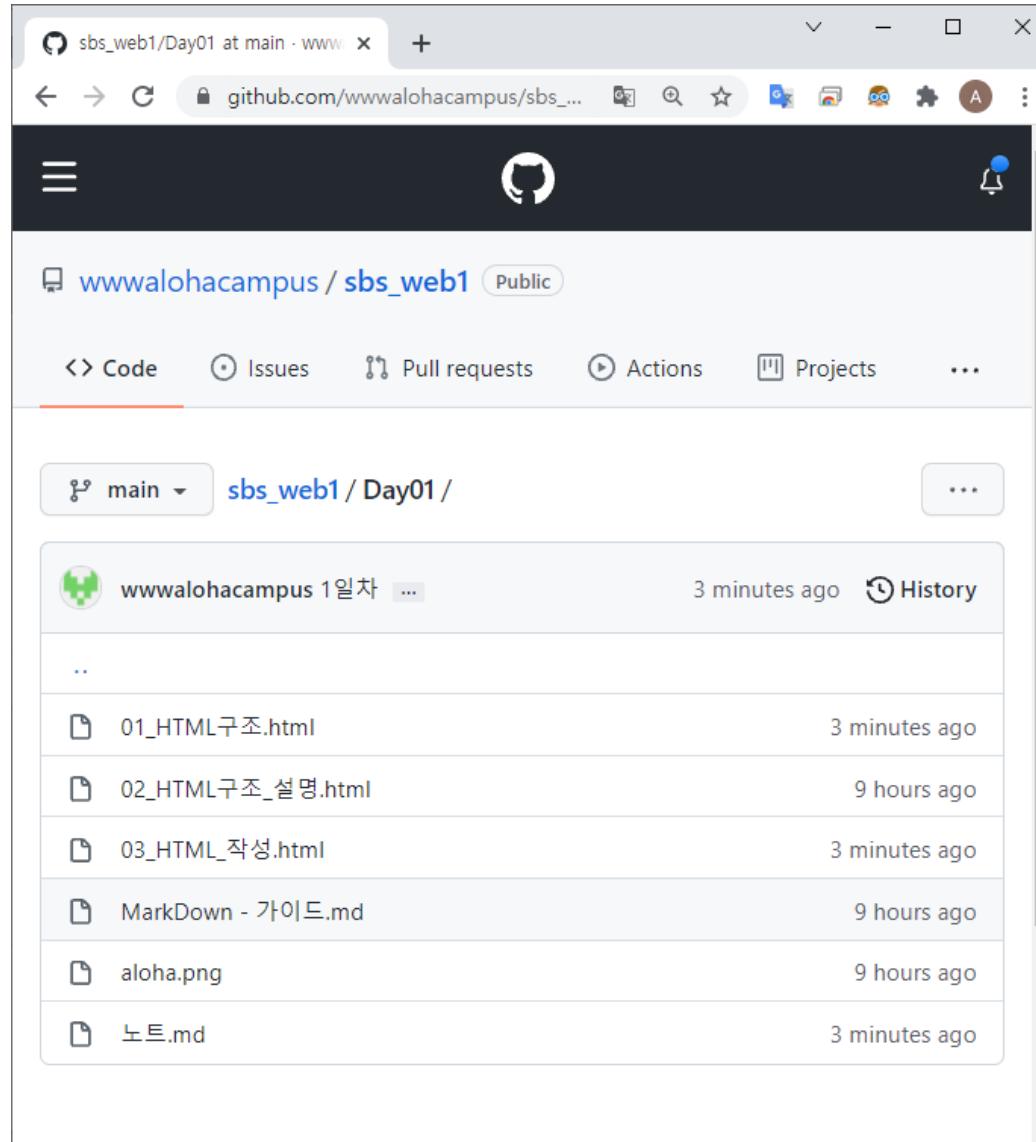
실습04

Git 커밋하고 푸시하기



실습05

GitHub 저장소 확인하기



과제01

개인 PC에 복습환경 만들기

1. VS Code 설치
2. Git 설치
3. Git Clone 하기



웹 표준, 기본 태그

#3



웹 표준

Web Standard



- 웹에서 사용되는 표준 기술이나 규칙
 - W3C 권고안에 나오는 기술들

 - 필요성
 - 과거에는 브라우저마다 웹 페이지가 다르게 보여지거나 멀티미디어 기능이 작동하지 않는 문제 발생
- 모든 브라우저에서
웹페이지가 동일하게 보이고 정상 작동





웹 접근성

Web Accessibility

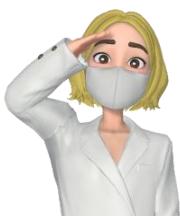


- 모든 사용자가 신체적, 환경적 조건에 관계없이 웹에 접근하여 이용할 수 있도록 보장하는 것

신체적 - 일반인, 장애인, 고령자 등

환경적 - 브라우저, 운영체제, 디바이스

ex) 마우스 또는 키보드만으로 웹 사용가능,
이미지에 대체 텍스트 제공 등





웹 호환성

Cross Browsing



- 여러 브라우저에서 동일한 사용자 경험을 줄 수 있도록 웹 페이지를 제작하는 기술, 방법
- 주로 MSIE (인터넷 익스플로러)에서도 동작하도록 웹 페이지를 제작하는 기술, 방법을 의미





기본 태그

- 텍스트
- 리스트
- 링크
- 이미지
- 테이블





텍스트 태그

- block 태그

- h1~h6
- p
- br
- hr
- blockquote
- pre

- inline 태그

- strong, b
- i, em
- q
- mark
- span
- ruby
- abbr, cite, code, ...





HTML 엔터티

출력 결과	설명	엔터티 이름	엔터티 번호
	공백	 	
<	보다 작다	<	<
>	보다 크다	>	>
&	앰퍼샌드	&	&
"	인용부호	"	"
'	어퍼스트로피	'	'
¢	센트	¢	¢
£	파운드	£	£
¥	옌	¥	¥
€	유로	€	€
©	저작권	©	©
®	등록상표	®	®





리스트 태그

-

- 순서가 없는 목록 태그

(Unordered List)

-

- 순서가 있는 목록 태그

(Ordered List)

- <dl>

- 정의 목록 태그

(Definition List)





링크 태그

-
 - 절대 경로 : href="http://www.naver.com"
 - 상대 경로 : href=". /01_text.html"
 - 내부 파일 : href="#top"
- target="이동 방식"
 - _self : 현재 페이지에서 열기
 - _blank : 새로운 템, 창에서 열기
 - _parent : 부모프레임으로 페이지 이동
 - _top : 최상위 프레임에서 열기
- title="설명"





이미지 태그

-
 - width : 가로 크기
 - height : 세로 크기
 - 이미지의 종류 : JPEG, PNG, GIF
- 관련태그 : <figure>, <figcaption>
- 경로
 - 절대경로 : 최상위 위치 부터 모든 상위경로를 포함한 고유 경로
 - C:/SBS_WEB1/sbs_web1
 - 상대경로 : 해당 파일이 위치한 곳을 기준으로한 경로
 - ./05_img.html



/	루트
./	현재 경로
../	상위 경로





테이블 태그

<table>

- 표 형태의 데이터를 표시하는 태그
- tr
- th, td
- caption

- thead, tbody, tfoot
- colgroup



멀티미디어 태그 입력 양식 태그

#4



멀티미디어 태그

- 오디오
- 비디오
- iframe





오디오 태그

- HTML5에서 오디오를 재생할 수 있는 태그
- 속성
 - autoplay, controls, loop, preload, src, volume
- 오디오 파일 형식
 - mp3, wav, ogg
 - 브라우저마다 지원되는 파일 형식이 다르다.
 - → <source> : 여러 가지 오디오 파일 형식을 동시에 제공





비디오 태그

- HTML5에서 비디오를 재생할 수 있는 태그
- 속성
 - autoplay, controls, loop, poster, preload, muted, src, volume, width, height
- 비디오 파일 형식
 - mp4, WebM, ogg
 - 브라우저마다 지원되는 파일 형식이 다르다.
 - → <source> : 여러 가지 비디오 파일 형식을 동시에 제공



iframe

- 페이지 내부에 다른 웹 페이지를 표시하는 태그
- iframe (Inline Frame)
- 속성
 - src, width, height





입력 양식 태그

- <form>
 - HTML 입력 양식을 생성하는 태그
 - 속성
 - action, method, enctype
- 입력
 - <input>, <textarea>, <select>, <fieldset>, <label>, ...



레이아웃 태그

#5



레이아웃 태그

- 블록 요소와 인라인 요소
- div & span
- 시맨틱 요소





블록 요소 vs 인라인 요소

block

- 가능한 전체 너비를 차지
- 항상 새로운 줄에서 시작
- width, height, margin, padding 등을 사용하여 크기와 간격을 변경할 수 있음





블록 요소 vs 인라인 요소

block

- 블록 요소의 종류
 - address, article, aside, audio, blockquote, canvas, dd, div, dl, fieldset, figcaption, figure, footer, form, h1, h2, h3, h4, h5, h6, header, hgroup, hr, noscript, ol, output, p, pre, section, table, ul, video





블록 요소 vs 인라인 요소

inline

- 컨텐츠 크기만큼 너비를 차지
- 새로운 줄에서 시작하지 않음
- width, height 로 크기를 변경할 수 없음
- line-height 로 줄의 높이를 변경
- text-align 으로 수평 정렬





블록 요소 vs 인라인 요소

inline

- 인라인 요소의 종류

- a, abbr, acronym, b, bdo, big, br, button, cite, code, dfn, em, i, img, input, kbd, label, map, object, q, samp, small, script, select, span, strong, sub, sup, textarea, tt, var





공간 분할 태그

- 블록 형식으로 공간 분할





공간 분할 태그

- 인라인 형식으로 공간 분할





시맨틱 요소

Semantic element

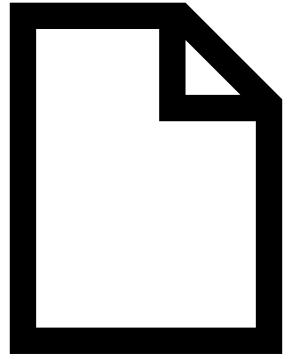
- 브라우저에 의미나 목적을 명확하게 알려주는 요소

태그	설명
<header>	문서의 머리말
<hgroup>	h1~h6 요소의 그룹
<nav>	네비게이션 링크
<article>	문서의 내용이나 포스트
<section>	문서의 섹션
<aside>	사이드 바
<footer>	문서의 꼬리말
<figure>	그림이나 도표
<time>	날짜와 시간 표시



CSS 기초 선택자와 적용 방법

#6



HTML

웹 페이지의 구조



CSS

웹 페이지의 스타일





Cascading Style Sheets

- 웹 페이지의 스타일을 지정하는 언어
- HTML 요소가 표시되는 방식을 설명





CSS 기초

- **스타일**

- 스타일 속성을 이용하여 웹 페이지를 디자인 하는 것

- **스타일 시트**

- 반복되는 스타일들을 모아 놓은 것





선택자

Selector

- HTML 문서의 특정 영역이나 요소 등을 선택해줄 수 있는 키워드
- 종류

기본

- 전체 선택자
- 태그 선택자
- 클래스 선택자
- 아이디 선택자
- 그룹 선택자

관계

- 종속 선택자
- 하위 선택자
- 자식 선택자
- 인접 선택자
- 형제 선택자

기타

- 가상 선택자
- 속성 선택자





스타일 적용 방법

- 인라인 스타일

```
<h1 style="color : red;">Hello</h1>
```

- 내부 스타일 시트

```
<style></style>
```

- 외부 스타일 시트

```
<link href="css/style.css">
```





스타일 적용 우선 순위

- ① 웹 브라우저 디폴트 값
- ② 인라인 스타일
- ③ 내부 스타일 시트
- ④ 외부 스타일 시트





스타일 적용 우선 순위

- ① !important 지정 속성
- ② 인라인 스타일
- ③ id 스타일
- ④ class 스타일
- ⑤ 태그 스타일

선택자 중복 지정 시,
마지막에 지정한 스타일이 적용된다.





CSS 스타일 기초

- 색상
- 폰트
- 텍스트 스타일





선택자의 종류

종류	내용	예시
전체 선택자	전체 문서 선택	*
태그 선택자	HTML 태그 선택	p, table, body, nav 등
아이디 선택자	# (id 속성)	#header, #footer
클래스 선택자	. (class 속성)	.btn, .container, .item
그룹 선택자	여러 선택자를 복합 적용	p .class, #id, table





선택자의 종류

종류	내용	예시
종속 선택자	선택자에 종속된 선택자	p.content, div#wrap
하위 선택자	선택자 내부의 후손 선택자	p a, .container h2, p #id
자식 선택자	선택자 바로 밑의 자식 선택자	div > p, p > a





선택자의 종류

종류	내용	예시
인접 선택자	특정요소와 붙어있는 선택자 지정	h1 + p
형제자매 선택자	형제자매 관계에 있는 선택자 지정	div~table
가상 선택자	특정 이벤트에 대한 선택자 지정	a:hover, button:active
속성 선택자	태그의 속성을 지정하는 선택자	[name='email'], [data='1']



가상 선택자

cSS1, cSS2

가상 (클래스) 선택자; 의사 (클래스) 선택자

종류	내용
:hover	마우스 커서가 링크에 올라가 있는 상태
:active	마우스 커서를 클릭한 순간부터 놓기 직전까지 상태
:link	링크를 클릭하지 않은 상태
:visited	링크를 눌러서 방문한 후 상태
:focus	입력 폼에서 키보드 입력을 기다리는 상태
:first-line	첫 번째 문장에만 적용
:first-letter	첫 번째 글자에만 적용
:before	문장이 시작되기 전
:after	문장이 끝난 다음
:lang	언어별로 속성을 적용하기 위한 선택자



가상 선택자

css3

가상 (클래스) 선택자; 의사 (클래스) 선택자

종류	내용
:first-of-type	선택자 중 맨 처음에 나오는 요소를 선택
:last-of-type	선택자 중 맨 마지막에 나오는 요소를 선택
:only-of-type	지정한 태그 선택자로만 구성된 요소 하나만을 선택
:only-child	지정한 태그 선택자의 오직 자식만을 선택
:nth-child	지정한 태그의 n번째 자식을 선택
:nth-last-child(n)	마지막 자식을 선택
:nth-of-type(n)	특정 태그의 n번째 자식을 선택
:nth-last-of-type(n)	맨 마지막부터 계산하여 특정 태그의 n번째 자식을 선택
:last-child	마지막 자식을 선택

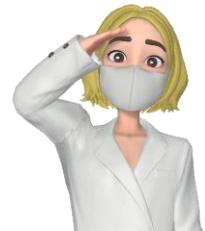


가상 선택자

css3

가상 (클래스) 선택자; 의사 (클래스) 선택자

종류	내용
:root	웹 페이지의 최상위 root를 선택
:empty	아무런 내용이 없는 특정 태그를 선택
:target	내부에서 링크를 이용하여 문서의 특정 부분으로 이동할 때 사용
:enabled	활성화한 태그를 선택
:disabled	비활성화한 태그를 선택
:checked	체크된 태그를 선택
:not(x)	x 선택자를 제외한 나머지를 선택
::section	웹 페이지에서 마우스를 이용하여 텍스트 등을 선택한 경우를 선택



텍스트 태그 및 타이포그래피 속성

#7



텍스트 태그

- block 태그
 - h1~h6
 - p
 - br
 - hr
 - blockquote
 - pre
- inline 태그
 - strong, b
 - i, em
 - q
 - mark
 - span
 - ruby
 - abbr, cite, code, ...





타이포그래피

- 활자 서체의 배열
- 활자의 서체나 글자 배치 따위를 구성하고 표현하는 일





타이포그래피 속성

▪ 글자 속성 (font)

- font-family
- font-size
- line-height
- font-weight
- font-style
- font-variant

▪ 문단 속성

- text-align
- text-indent
- text-decoration
- text-transform
- vertical-align
- letter-spacing
- word-spacing
- white-space
- word-break



font

단축 속성

글자와 관련된 스타일을 지정하는 속성

`font: italic small-caps bold 12px/30px Georgia, serif;`

`font: [font-style], [font-variant], [font-weight],
[font-size/line-height], [font-family]`



font

단축 속성

속성 값	설명	기본값
font-style	기울기 지정	normal
font-weight	두께 지정	normal
font-size	크기 지정	medium (16px)
line-height	줄 높이 지정	normal
font-family	글꼴 지정	OS 따라 상이
font-variant	소문자→작은 대문자 변환여부 지정	normal



font-family

폰트를 지정하는 속성

- 적용방식

1. 시스템 폰트

- 운영체제에 있는 글꼴을 지정

2. 웹 폰트

- 서버에 업로드된 폰트를 다운받아서 지정

3. 이미지 폰트

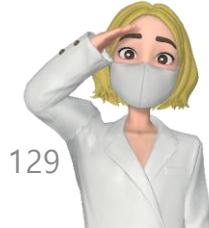
- 폰트를 이미지로 제작해서 지정

- 기본 폰트

- Win : 굴림, 돋움, 궁서, 바탕, 맑은 고딕
- Mac : 애플고딕, 애플명조, PC 명조, 필기체, 애플 SD, 산돌고딕, Neo, 나눔고딕체

- 폰트 분류

- 산 세리프(san-serif) : 끝부분에 빼침 장식 X
- 세리프(serif) : 끝부분에 빼침 장식 O



font-family

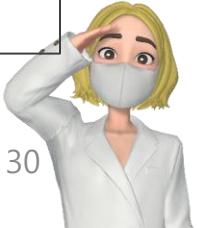
폰트를 지정하는 속성

font-family: Arial, Helvetica, sans-serif;

- 여러 글꼴 지정 가능
- 브라우저가 1 번째 글꼴 지원 X → 다음 글꼴
- 유형
 - family-name (글꼴 이름)
 - generic-family (글꼴 유형)



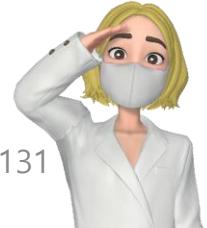
- | | |
|--------------|-----------------|
| ▪ serif | : 빠침O, 명조계열의 글꼴 |
| ▪ sans-serif | : 빠침X, 고딕계열의 글꼴 |
| ▪ monospace | : 폭과 간격이 일정한 글꼴 |
| ▪ cursive | : 필기계열의 글꼴 |
| ▪ fantasy | : 화려한 글꼴 |



font-size

폰트를 크기를 지정하는 속성

- 절대 단위, 상대 단위 값 지정 가능
- 부모 요소로부터 상속되는 속성
- 제목 태그, 폼 태그는 font-size 상속되지 않음



font-size

폰트를 크기를 지정하는 속성

■ 절대단위

- cm : 센티미터
- mm : 밀리미터, 1/10cm
- pt : 1/72 in
- in : 1 in = 2.54.cm
- pc : 1pc = 12pt

■ 상대단위

- px : 해상도에 따라
- % : 기본 크기 100% 기준
- em : 기본 크기 1em 기준
- rem : html(루트) 1rem 기준
- ex : 소문자 x 높이를 1 기준
- vw & vh : 화면 너비/높이 100% 기준
- vmin & vmax : 너비/높이 중 짧은/긴 길이를 기준
- medium 류 : xx-small, x-small, small, large, x-large, xx-large (medium 기준)



line-height

텍스트 라인의 높이를 지정하는 속성

`line-height: normal;`

`line-height: 1.6;`

`line-height: 200%;`



font-weight

텍스트의 굵기 정도를 지정하는 속성

`font-weight: normal;`

`font-weight: bold;`

`font-weight: 900;`



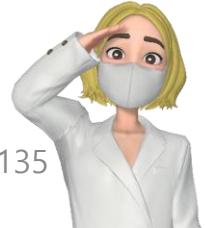
font-style

글꼴 스타일(기울기)을 지정하는 속성

`font-style: normal;`

`font-style: italic;` ↗ 손글씨

`font-style: oblique;` ↗ 비스듬한



font-variant

소문자를 작은 대문자로의 표현여부 지정

`font-variant: normal;`

`font-variant: small-caps;`



text-align

텍스트의 수평 정렬을 지정하는 속성

`text-align: center;`

`text-align: left;`

`text-align: right;`

`text-align: justify;`



text-indent

첫 줄의 들여쓰기를 지정하는 속성

`text-indent: 50px;`

`text-indent: -2em;`

`text-indent: 30%;`



text-decoration

텍스트의 추가 장식을 지정하는 속성

`text-decoration: overline;`

`text-decoration: line-through;`

`text-decoration: underline;`

`text-decoration: underline overline;`



text-transform

대소문자를 변형하는 속성

`text-transform: uppercase;`

`text-transform: lowercase;`

`text-transform: capitalize;`



vertical-align

요소의 수직 정렬을 지정하는 속성

`vertical-align: baseline;`

`vertical-align: text-top;`

`vertical-align: text-bottom;`

`vertical-align: sub;`

`vertical-align: super;`



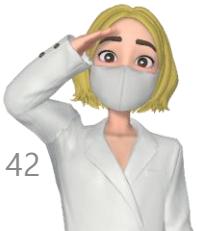
letter-spacing

문자 간격의 크기를 지정하는 속성

`letter-spacing: 3px;`

`letter-spacing: 2px;`

`letter-spacing: -1px;`



word-spacing

단어 사이의 공백 크기를 지정하는 속성

`word-spacing: 30px;`



white-space

요소 내부의 공백을 처리 방법을 지정하는 속성

`white-space: nowrap;` ↗ 단일 공백, 개행 X

`white-space: normal;`

`white-space: pre;` ↗ 연속된 공백도 그대로 표시



word-break

줄 바꿈 기준을 지정하는 속성

`word-break: normal;`

`word-break: break-all;` ↗ 글자 기준으로 줄바꿈

`word-break: keep-all;` ↗ 단어 기준으로 줄바꿈



박스 모델

#8

box model

html 요소를 감싸는 상자

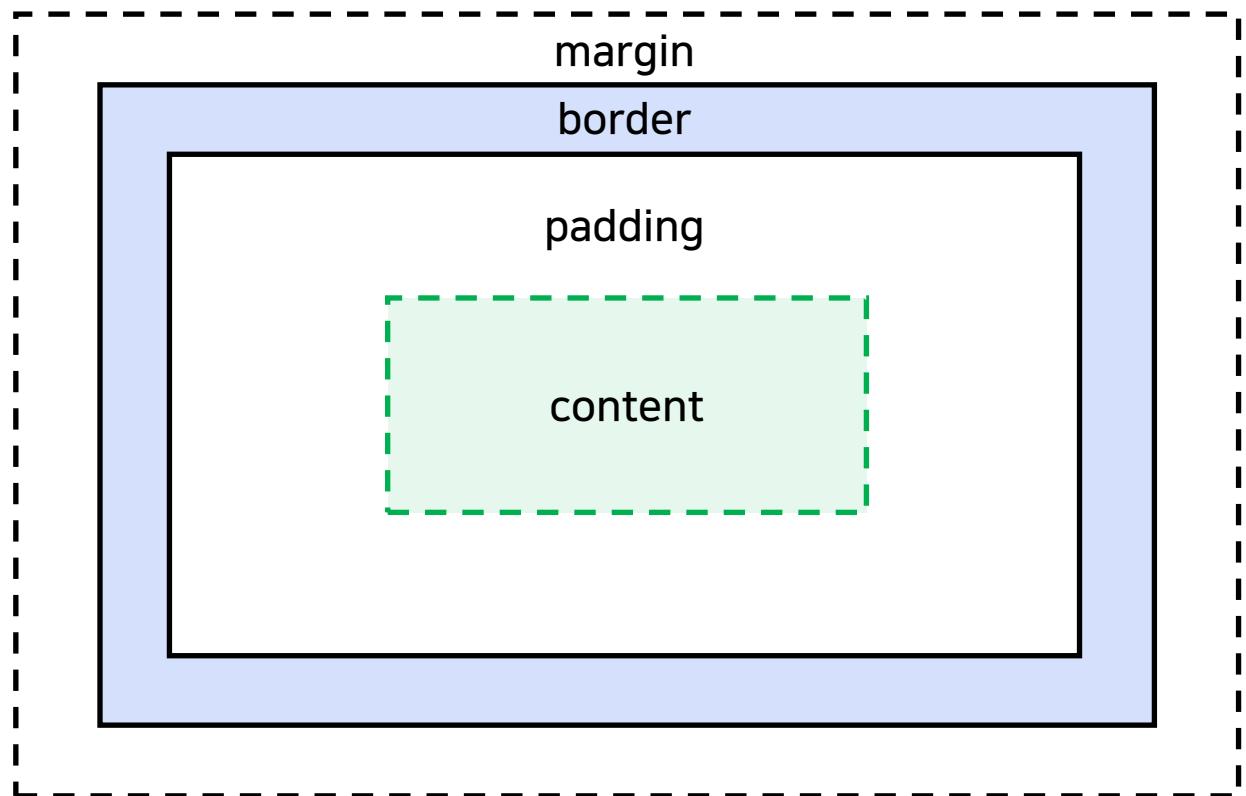
- margin 외부 여백
 - border 테두리
 - padding 내부 여백
 - content 내용 - text, image



box model

html 요소를 감싸는 상자

- margin 외부 여백
- border 테두리
- padding 내부 여백
- content 내용 - text, image



margin

요소의 외부 여백을 지정하는 속성

- 속성값

- auto
- length
- %
- inherit

브라우저가 길이를 계산
px, em, pt, cm 등
상위 요소에 대한 비율 지정
부모 요소를 길이로 지정

- 단축 속성

- margin

top, right, bottom, left (시계방향)

- 개별 속성

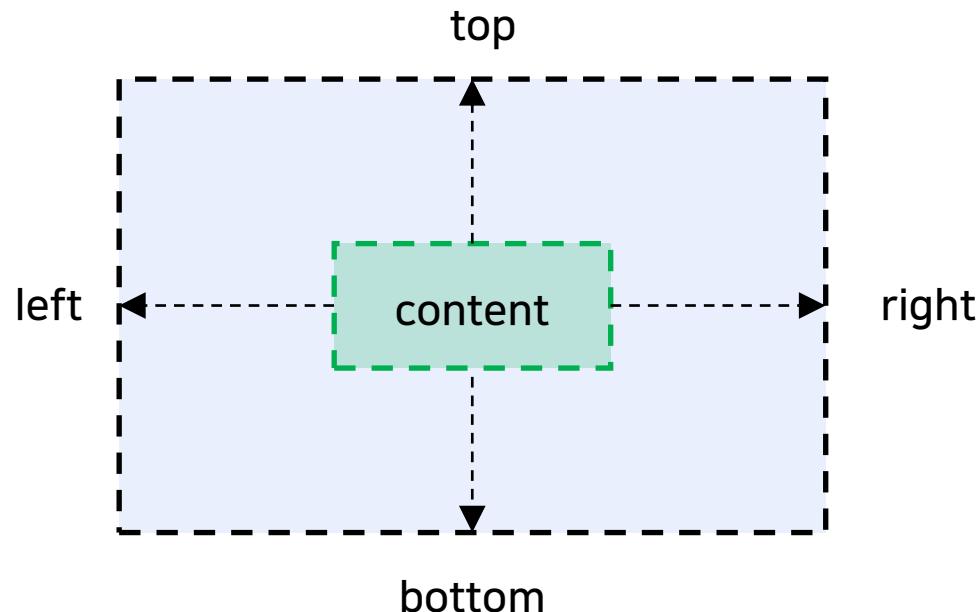
- margin-top
- margin-right
- margin-bottom
- margin-left

위쪽 외부 여백
오른쪽 외부 여백
아래쪽 외부 여백
왼쪽 외부 여백



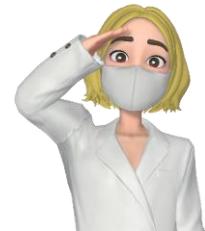
margin

요소의 외부여백을 지정하는 속성



축약형으로 지정하는 방법 4가지

```
1  margin : top right bottom left;  
2  margin : top [left, right] bottom;  
3  margin : [top bottom], [left, right];  
4  margin : [all];  
5  
6
```



margin

요소의 외부여백을 지정하는 속성



03_박스 모델 > 01_margin



01_margin.html

```
1  
2 margin : 10px 20px 30px 40px;  
3 margin : 10px 20px 30px;  
4 margin : 10px 20px;  
5 margin : 10px;  
6
```



margin

요소의 외부여백을 지정하는 속성

* `auto`

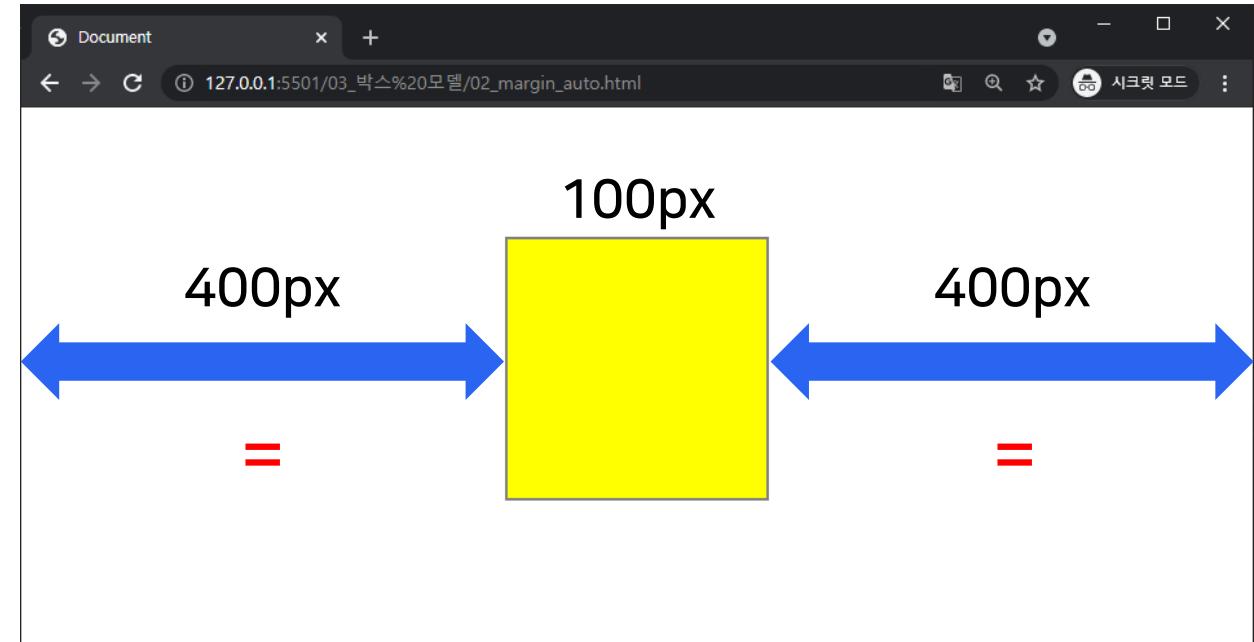
- 컨테이너 내에 요소를 가운데 배치
- 좌우 margin 을 균등 분할

```
1  
2     margin : auto;  
3     margin : 0 auto;  
4
```

03_박스 모델 > 01_margin

02_margin_auto.html

900px



margin

요소의 외부여백을 지정하는 속성

* [inherit](#)

- 부모 요소의 크기를 상속



03_박스 모델 > 01_margin



03_margin_inherit.html

```
1
2  div { margin-left: 50px; }
3  div p { margin-left: inherit; }
4
```



margin

요소의 외부여백을 지정하는 속성

* 개별 속성

- 상, 하, 좌, 우 외부 여백을 각각 지정



03_박스 모델 > 01_margin



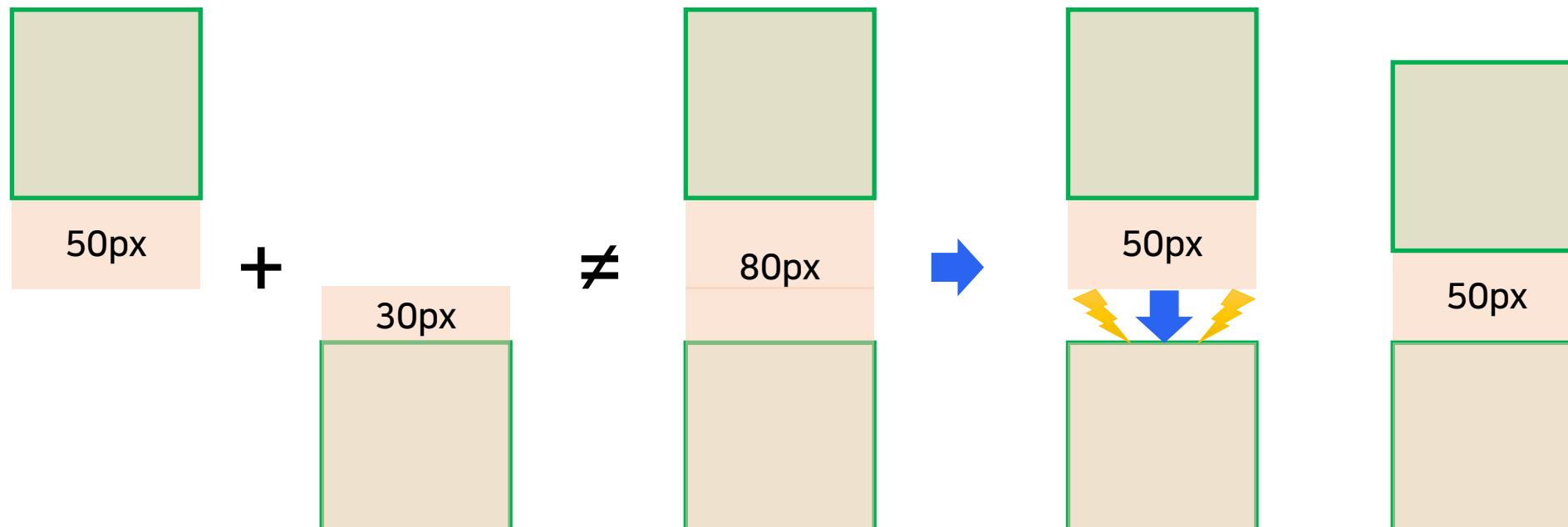
04_margin_property.html

```
1
2  /* margin : 10px 20px 30px 40px */
3
4  margin-top      : 10px;
5  margin-right    : 20px;
6  margin-bottom   : 30px;
7  margin-left     : 40px;
8
```



margin collapse

상단 및 하단 여백이 둘 중 더 큰 여백으로 축소되는 현상



margin collapse

상단 및 하단 여백이 둘 중 더 큰 여백으로 축소되는 현상

- 부모 자식 요소 간 발생
- 인접한 형제 요소 간 발생
- 비어있는 요소에서 발생



```
1 <div style="margin-bottom: 50px;">  
2   <div style="margin-top: 30px;"></div>  
3 </div>
```



```
1 <div style="margin-bottom: 50px;"></div>  
2 <div style="margin-top: 30px;"></div>
```



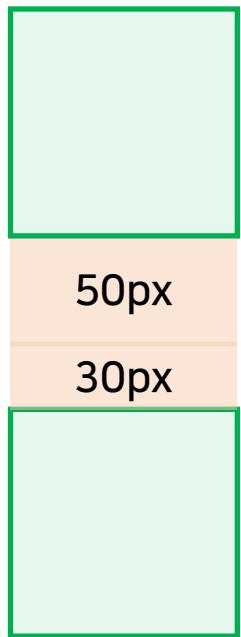
```
1 <div style="margin-bottom: 50px;"></div>  
2 <p style="margin-top: 30px;">margin collapse</p>
```



margin collapse

해결하기

원인 : 위, 아래로 블록요소가 있음

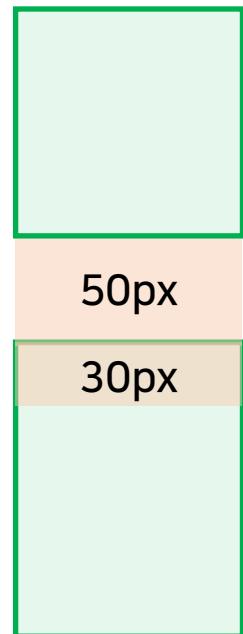


01_박스모델 > 01_margin

05_margin_collapse.html

margin 처리 맞닿지 않게 한다!

① padding 으로 처리



② border 를 처리



margin collapse

마진 상쇄 예외 상황

- position : absolute;
- float : left/right;
- display : flex; 로 지정된 컨테이너의 자식요소 일 때
- display : grid; 로 지정된 컨테이너의 자식요소 일 때



padding

요소의 내부 여백을 지정하는 속성

- 속성값

- auto
- length
- %
- inherit

브라우저가 길이를 계산
px, em, pt, cm 등
상위 요소에 대한 비율 지정
부모 요소를 길이로 지정

- 단축 속성

- padding

top, right, bottom, left (시계방향)

- 개별 속성

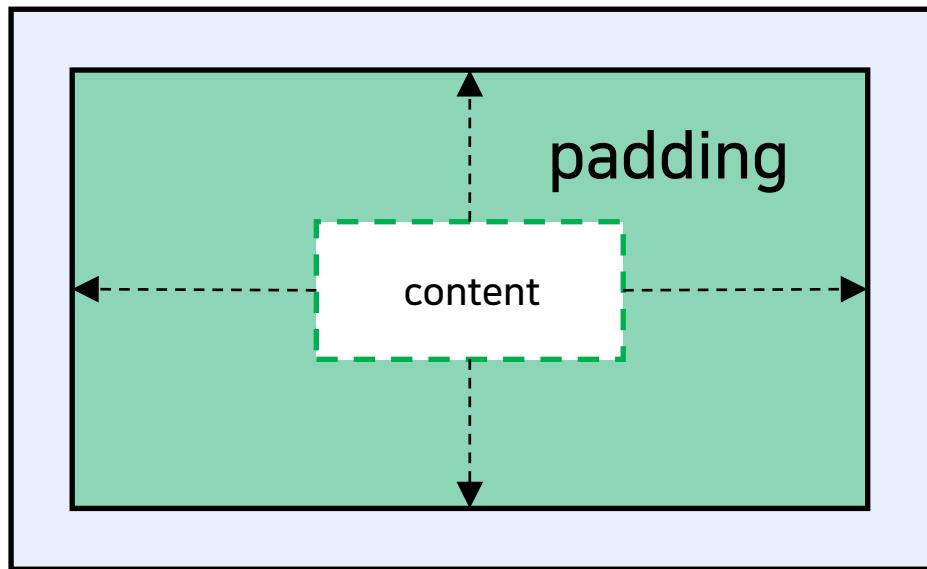
- padding-top
- padding-right
- padding-bottom
- padding-left

위쪽 내부 여백
오른쪽 내부 여백
아래쪽 내부 여백
왼쪽 내부 여백



padding

요소의 **내부** 여백을 지정하는 속성



```
1
2   padding : top right bottom left;
3   padding : top [left, right] bottom ;
4   padding : [top, bottom], [left, right];
5   padding : [all];
6
```



padding

요소의 [내부](#) 여백을 지정하는 속성



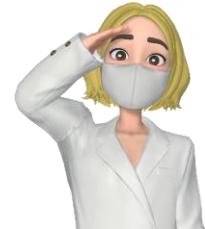
03_박스 모델 > 02_padding



01_padding.html



```
1  
2 padding : 10px 20px 30px 40px;  
3 padding : 10px 20px 30px;  
4 padding : 10px 20px;  
5 padding : 10px;  
6
```



padding

요소의 [내부](#) 여백을 지정하는 속성

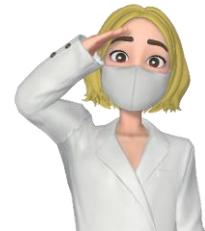


03_박스 모델 > 02_padding



02_padding_property.html

```
1  
2 padding-top      : 10px;  
3 padding-right    : 20px;  
4 padding-bottom   : 30px;  
5 padding-left     : 40px;  
6
```

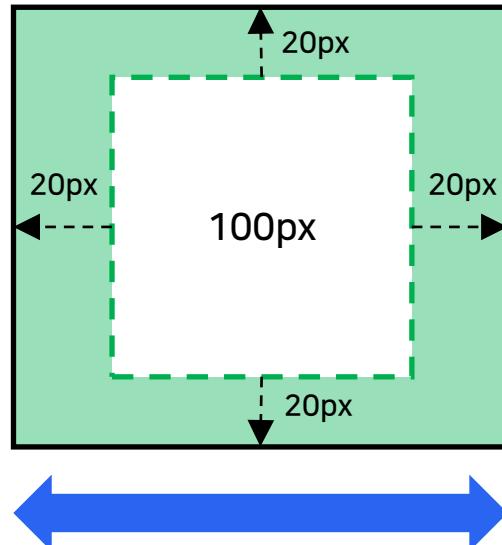


padding

요소의 **내부** 여백을 지정하는 속성



```
1 .box1 {  
2   padding: 20px;  
3   width: 100px;  
4   height: 100px;  
5 }
```



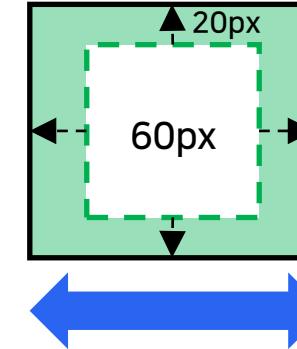
03_박스 모델 > 02_padding



03_padding_box.html



```
1 .box1 {  
2   padding: 20px;  
3   width: 100px;  
4   height: 100px;  
5   box-sizing: border-box;  
6 }
```



box-sizing: border-box;

width와 height 의
border, padding 을 포함시키는 속성



width/height 로 지정한 크기가
실제 요소의 크기가 되도록 !!!



배치 속성

#9



배치 속성

- float
- clear
- 중첩박스
- box-sizing



float

요소를 띄우는 방식을 지정하는 속성

- 속성값

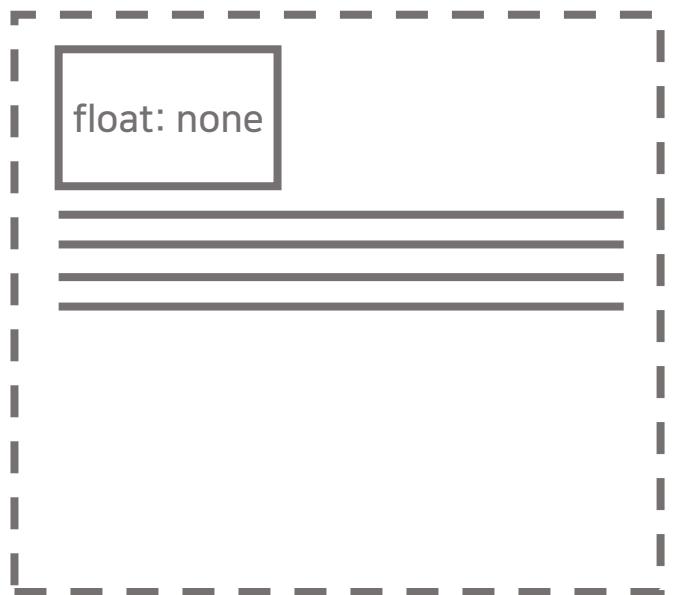
- left 요소를 왼쪽으로 띄워서 배치
- right 요소를 오른쪽을 띄워서 배치
- none 요소를 띄우지 않음 (기본값)
- inherit 부모 요소를 따름



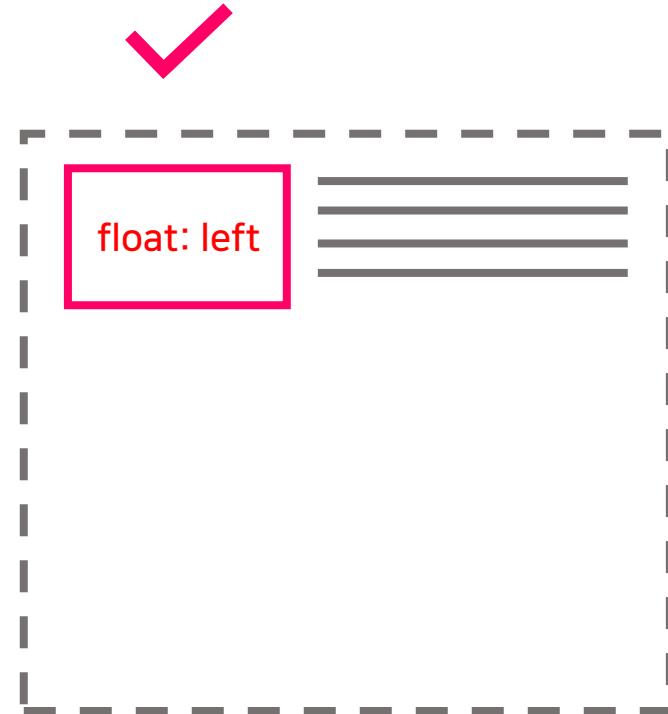
float

float : left

- 요소를 띄워서 왼쪽으로 배치



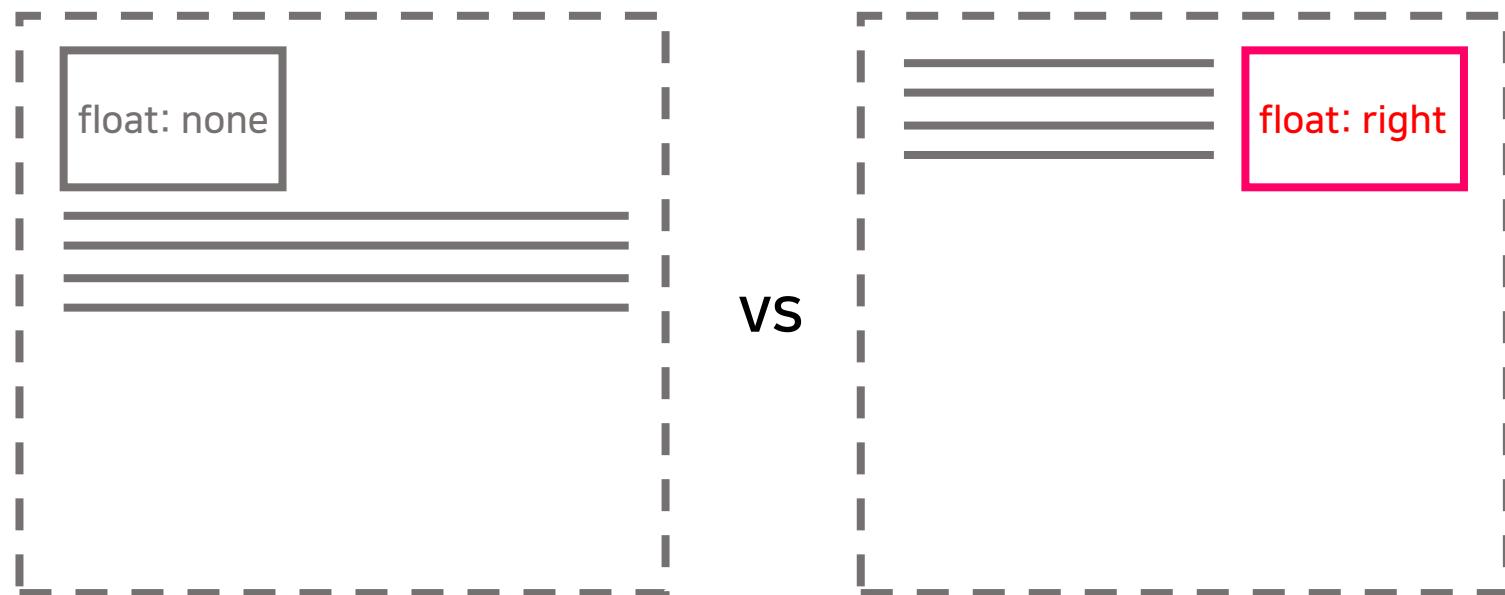
vs



float

float : right

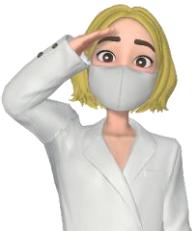
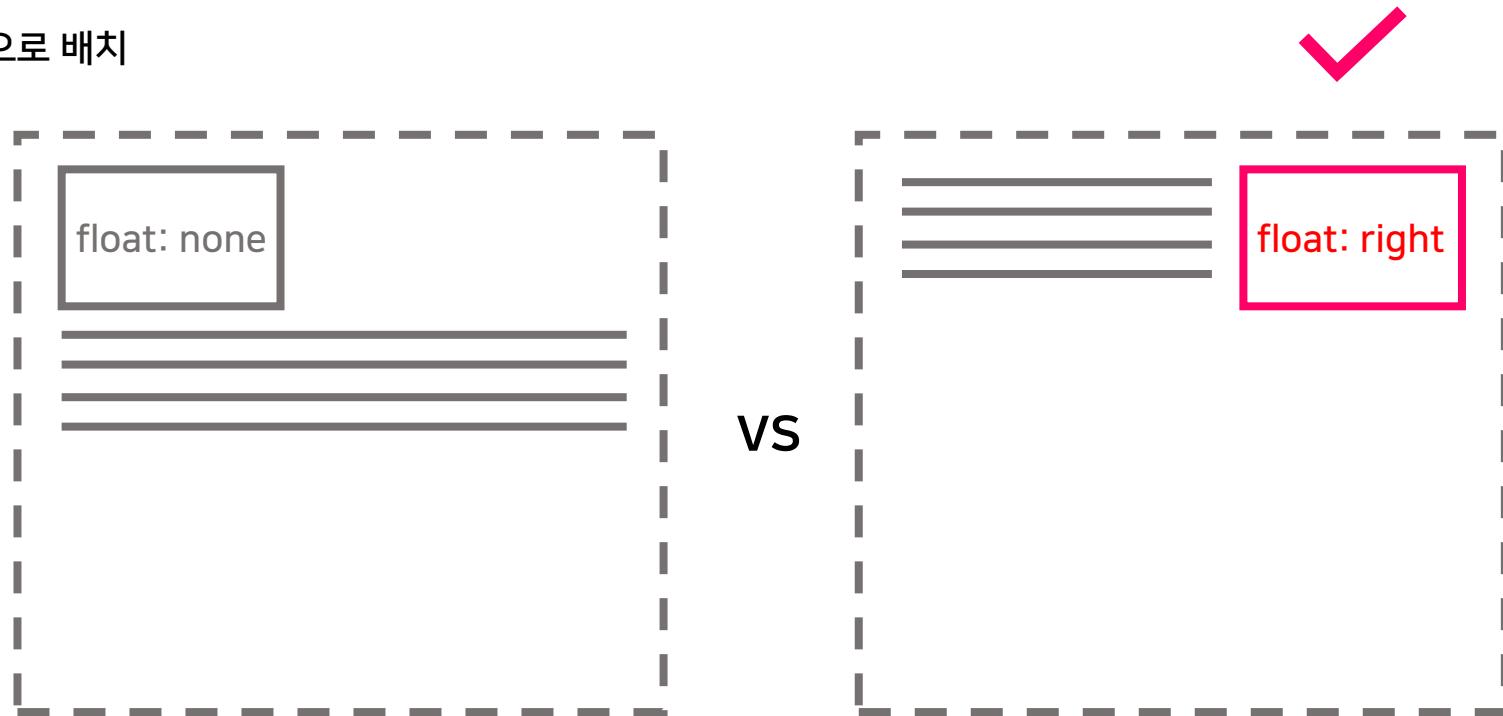
- 요소를 띄워서 오른쪽으로 배치



float

float : right

- 요소를 띄워서 오른쪽으로 배치



clear

float 지정 요소 다음 요소의 배치를 지정하는 속성

- 속성값

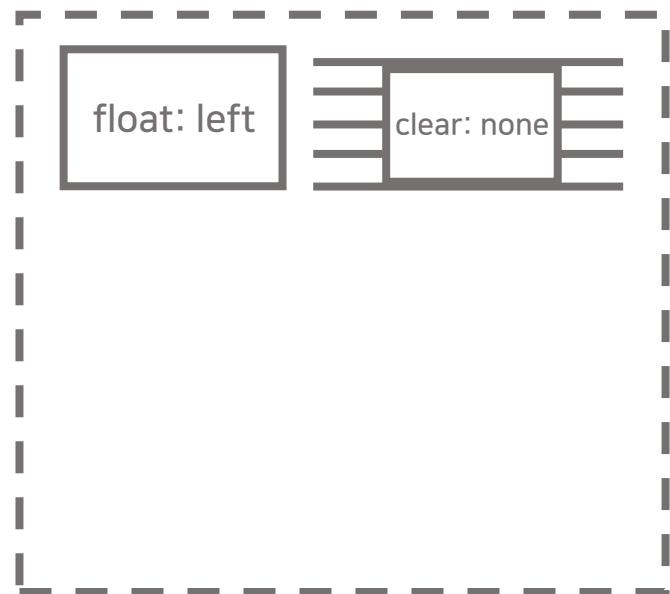
- left float:left 요소 다음 블록에 배치
- right float:right 요소 다음 블록에 배치
- both float:left,right 요소 다음 블록에 배치
- none float 요소 다음으로 배치되지 않음 (기본값)
- inherit 부모 요소를 따름



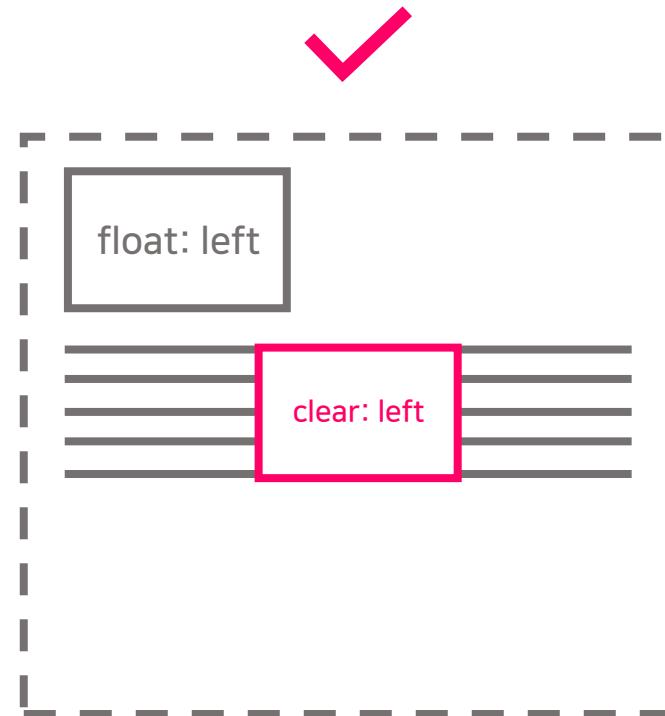
clear

clear : left

- float:left 요소 다음 블록에 배치



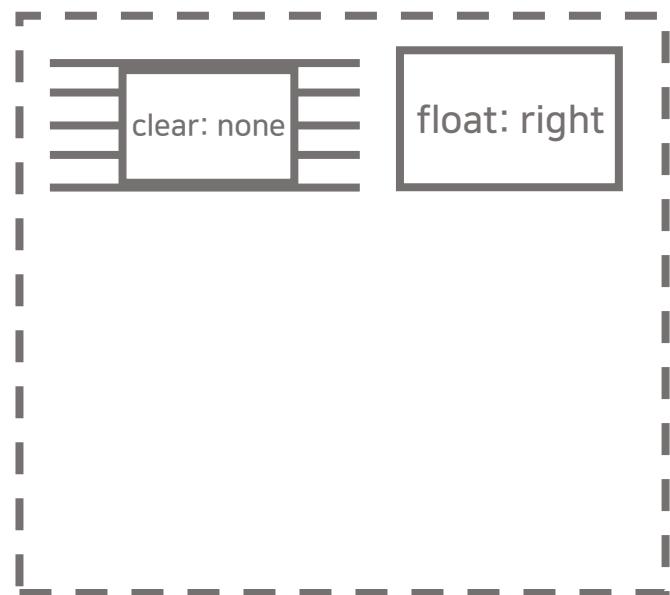
vs



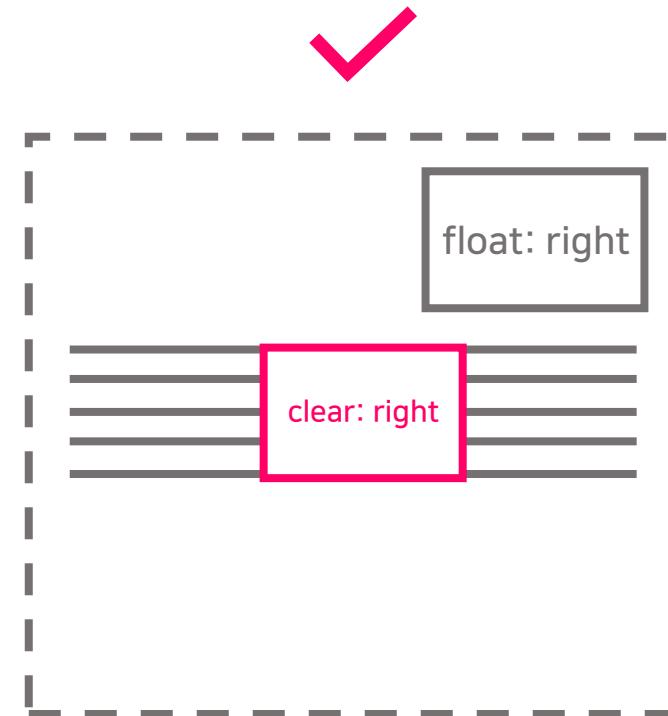
clear

clear: right

- float:right 요소 다음 블록에 배치



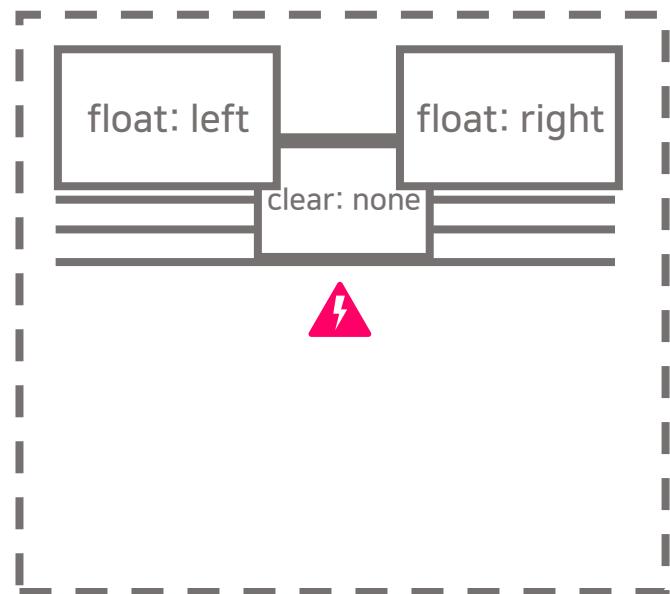
vs



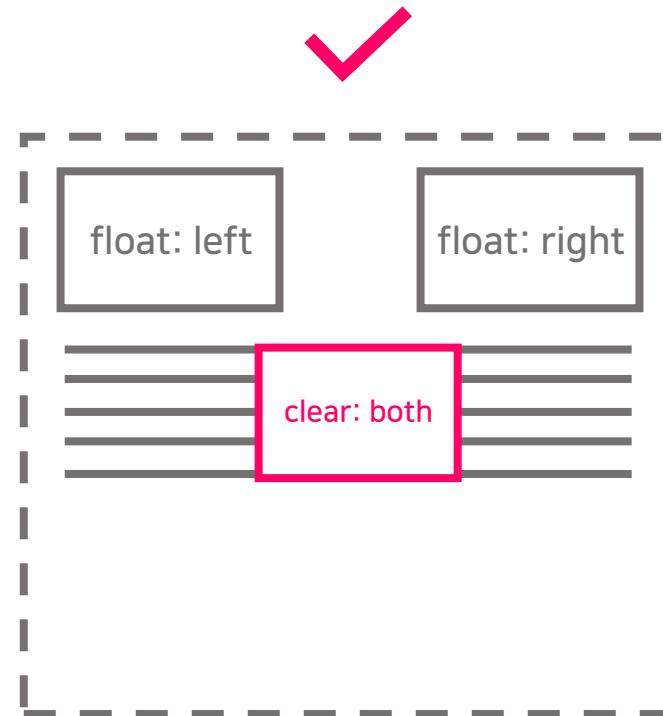
clear

`clear : both`

- `float:left,right` 요소 다음 블록에 배치



VS

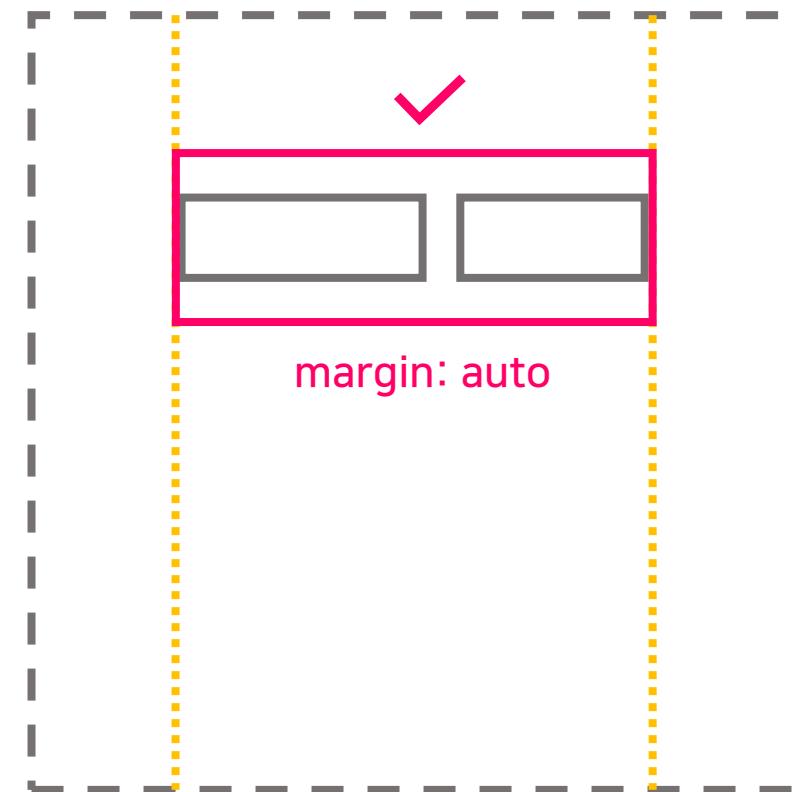
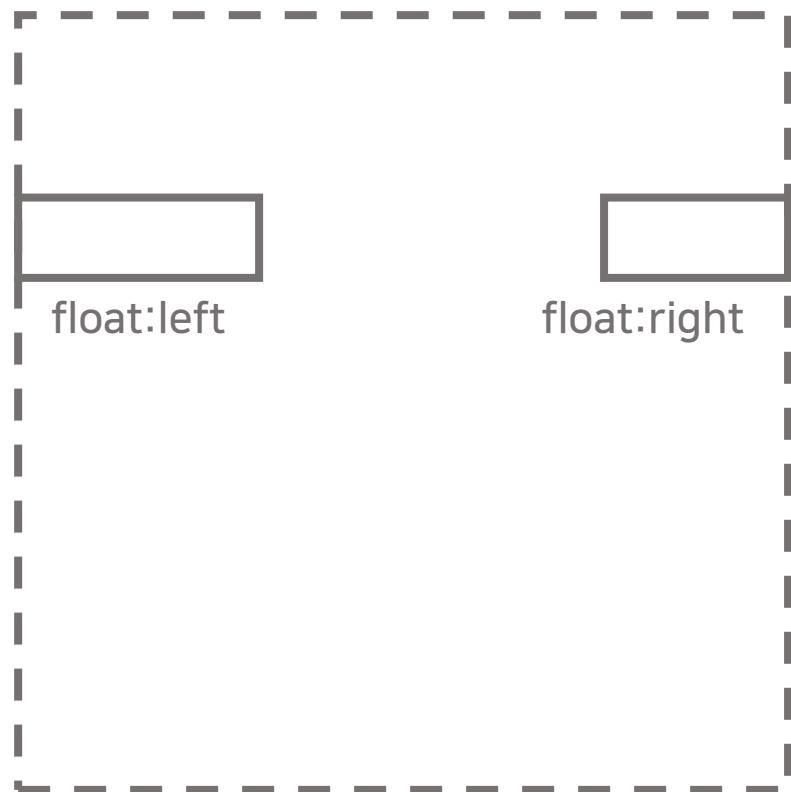




중첩 박스

두 박스를 왼쪽, 오른쪽 배치를 유지한 채로
페이지에 가로 가운데 정렬하려면?

구조적으로 배치하기 위해 중첩된 박스



box-sizing

가로/세로 크기에 padding과 border 포함여부를 지정하는 속성

- 속성값

- content-box
- border-box

컨텐츠 영역만 가로/세로 크기로 지정 (기본값)

border, padding 영역을 포함하여 가로/세로 크기 지정



표현 속성 리스트 속성

#10



표현 속성

- display
- overflow



display

요소의 유형을 지정하는 속성

- 속성값

- | | |
|----------------|-----------|
| ▪ block | 블록 요소 |
| ▪ inline | 인라인 요소 |
| ▪ inline-block | 인라인 블록 요소 |
| ▪ table | 테이블 요소 |
| ▪ grid | 2차원 레이아웃 |
| ▪ flex | 반응형 레이아웃 |
| ▪ none | 표시 하지 않음 |



overflow

넘치는 컨텐츠에 대한 처리유형을 지정하는 속성

- 속성값

- | | |
|-----------|---------------------|
| ▪ visible | 컨테이너 외부에 출력 |
| ▪ scroll | 스크롤 영역에서 출력 |
| ▪ auto | 넘치는 경우에만 스크롤 영역에 출력 |
| ▪ hidden | 숨김 |
| ▪ initial | 기본값 |
| ▪ inherit | 부모 요소를 따름 |





리스트 속성

- 목록 스타일 속성
 - list-style
 - list-style-type
 - list-style-position
 - list-style-image



list-style

리스트 요소의 스타일을 지정하는 속성

- 단축 속성

- [list-style-type] 마커 유형을 지정
- [list-style-position] 마커의 위치를 지정
- [list-style-image] 마커 대체 이미지 지정



배경과 색상 표현

#11



배경과 색상 표현

- 배경 이미지 속성
- 웹 색상 표현
- 그레디언트



background

요소의 배경 스타일을 지정하는 속성

- 단축 속성

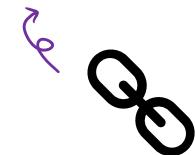
- | | |
|-------------------------|-----------|
| ▪ background-color | 배경 색상 |
| ▪ background-image | 배경 이미지 |
| ▪ background-repeat | 이미지 반복 여부 |
| ▪ background-attachment | 스크롤/고정 여부 |
| ▪ background-position | 이미지 시작 위치 |
| ▪ background-size | 이미지 크기 |





웹 색상 표현

색상 팔레트 사이트



■ 색상 표현 방법

유형	설명
#000000	hex 색상 코드 단위 - RGB 256색을 16진수로 짧게 표시
black, red, ...	컬러 이름으로 표현
rgb(0, 0, 0)	red, green, blue(0~255) 색상 단위
rgba(0, 0, 0, 0)	red, green, blue(0~255), alpha(0.0~1.0) 색상 단위
hsl (0, %, %)	Heu(색조), Saturation(채도,%), Lightness (밝기,%) 색상 단위
hsla (0, %, %, 0)	hsl + alpha 색상 단위





웹 색상 표현

■ 색상 코드

- #000000 : black
- #FFFFFF : white
- #FF0000 : red
- #00FF00 : green
- #0000FF : blue

#A5 = 165

16^1	$16^0 (=1)$
A	5
A X 16	5 X 1
(10 X 16) + (5 X 1) = 165	

16진수 표기법

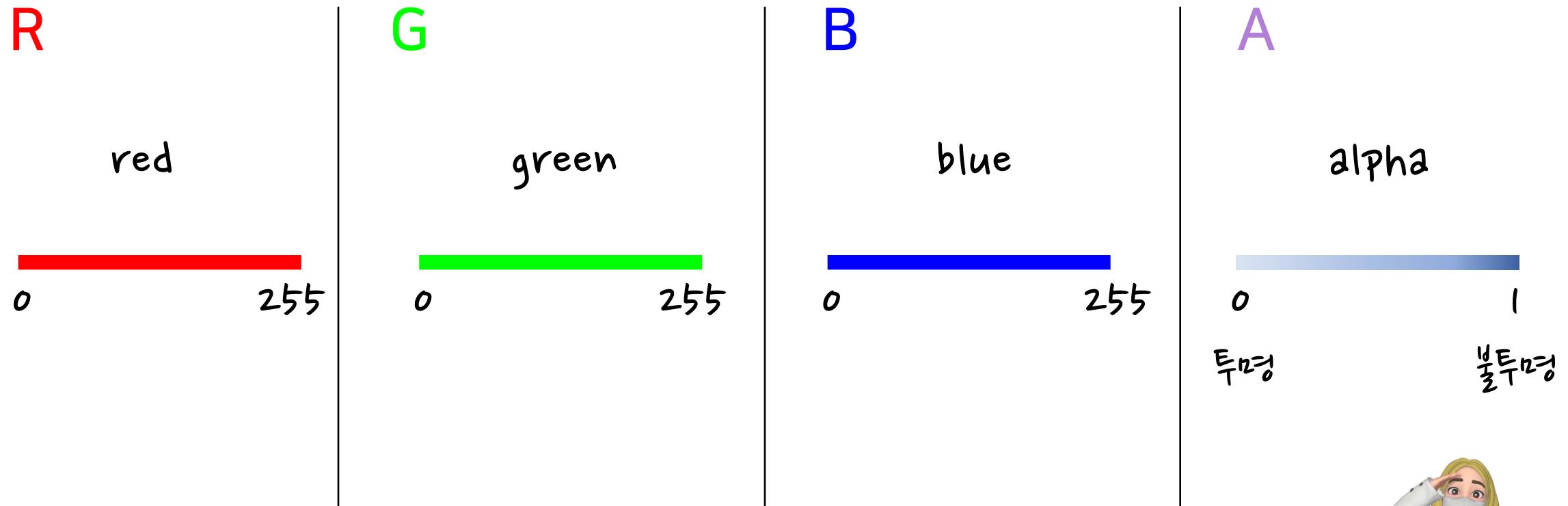
1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15





웹 색상 표현

rgba (0, 0, 0, 0)





웹 색상 표현

hsla (0, %, %, 0)





웹 색상 표현

그레디언트

- linear-gradient
- radial-gradient
- conic-gradient



linear-gradient()

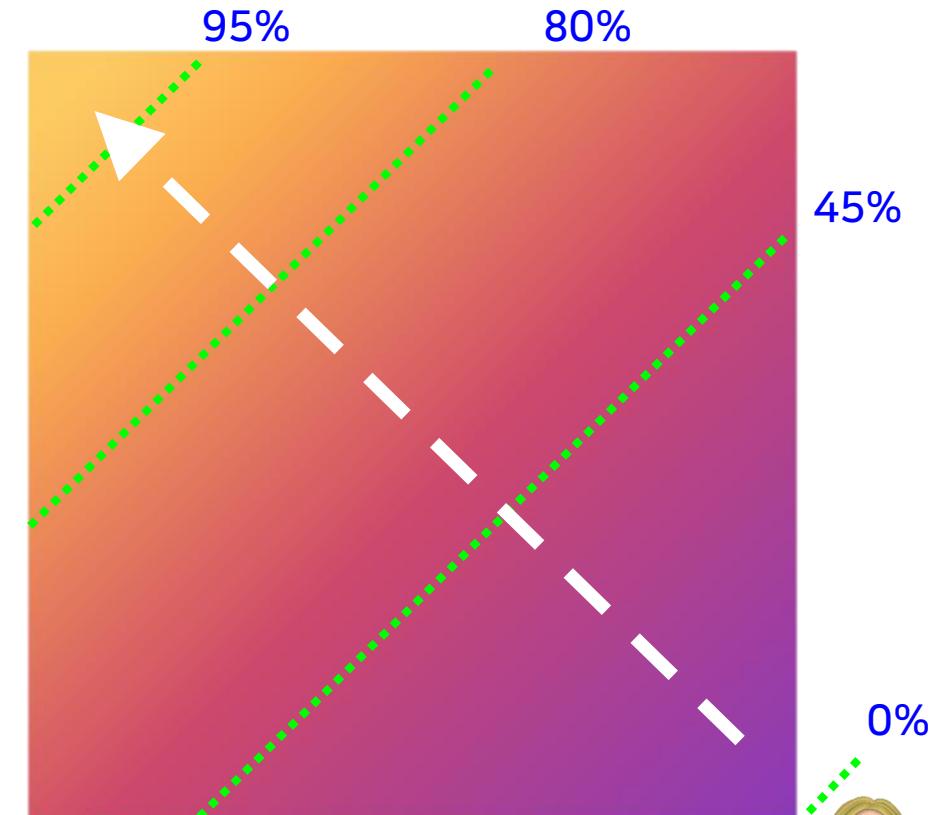
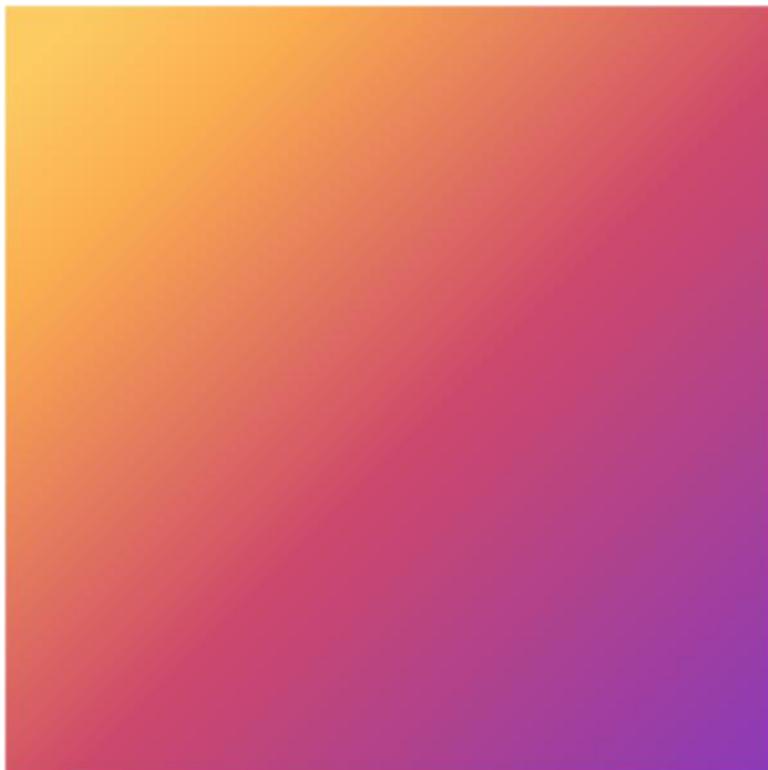
선형 그라데이션 지정 함수

- 최소 2개 색상 정지점을 지정해야함
- 속성값
 - direction, color-stop1, color-stop2 ...
 - direction
 - to bottom (기본값), to top, to left, to right
 - to (top/bottom) (left/right) (수직/수평)
 - 0 ~ 360deg



linear-gradient()

```
linear-gradient(to top left, #8a3ab9 0%, #cd486b 45%, #fbad50 80%, #fccc63 95%);
```



radial-gradient()

방사형 그라데이션 지정 함수

- 최소 2개 색상 정지점을 지정해야함
- 속성값
 - shape size, color-stop1, color-stop2 ...
 - shape
 - circle (정원), ellipse (타원)
 - size
 - closest-side
 - farthest-side
 - closest-corner
 - farthest-corner



radial-gradient()

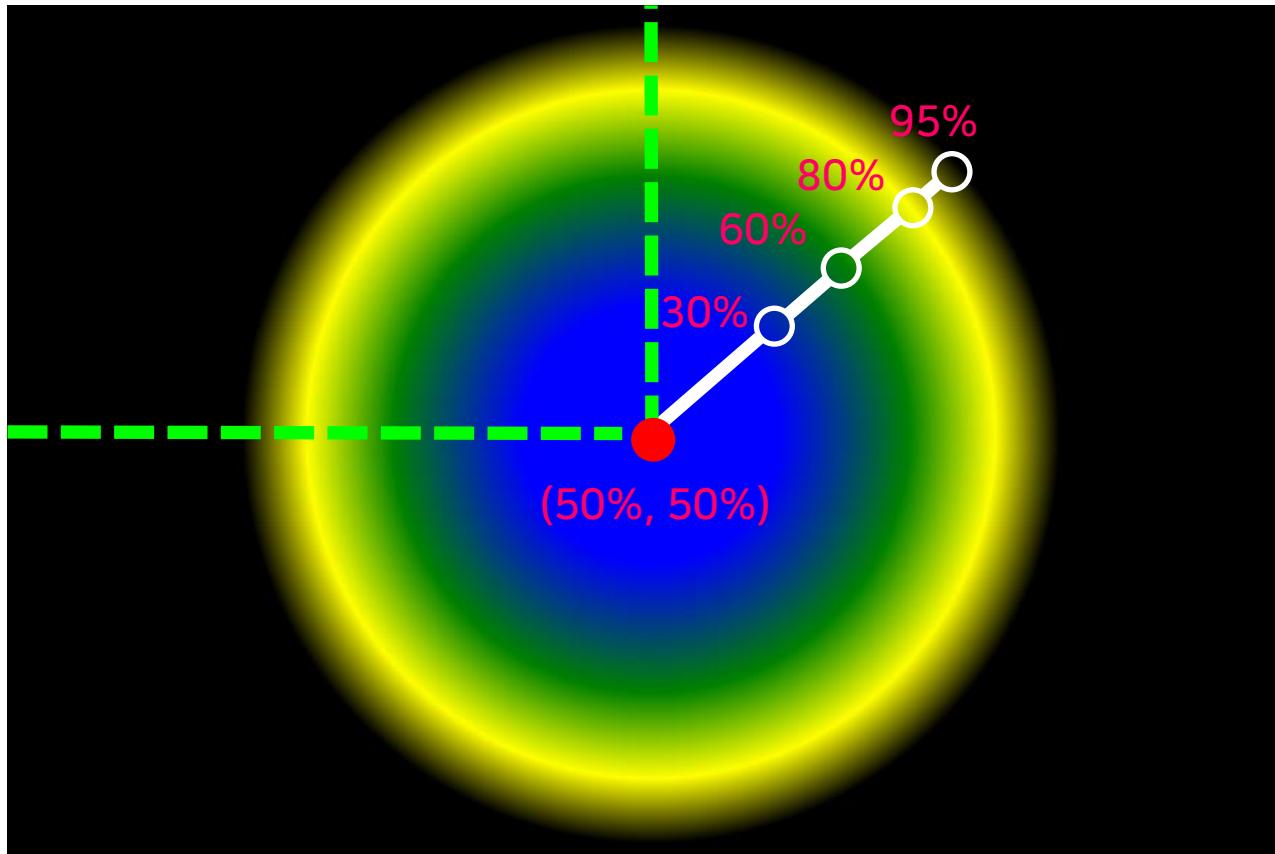
```
radial-gradient(circle closest-side at 50% 50%, blue 30%, green 60%, yellow 80%, black 95%);
```

로우

사이즈

중심점 위치 - (가로%, 세로%)

색상 시각위치(%)



radial-gradient()

radial-gradient(모양 사이즈 at 가로 세로, 컬러1 위치1, 컬러2 위치2, …);

모양	설명
ellipse	타원
circle	원

사이즈	설명
closest-corner	시작점(중앙)에서 둘러싸는 상자의 가장 가까운 모서리까지의 거리
closest-side	시작점(중앙)에서 둘러싸는 상자의 가장 가까운 면까지의 거리
farthest-corner	시작점(중앙)에서 둘러싸는 상자의 가장 먼 모서리까지의 거리
farthest-side	시작점(중앙)에서 둘러싸는 상자의 가장 먼 면까지의 거리



conic-gradient()

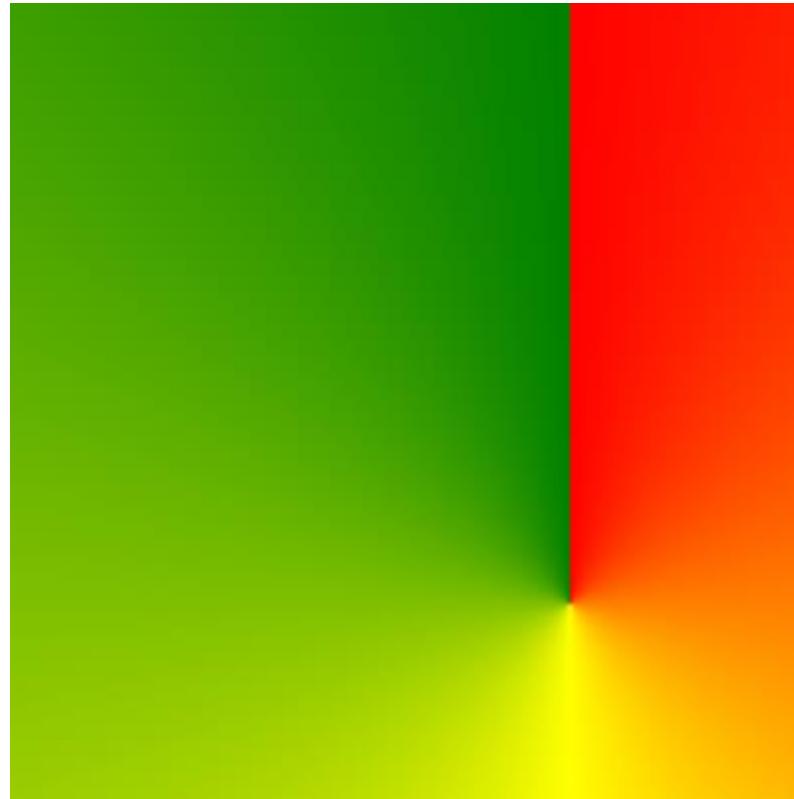
원뿔형 그라데이션 지정 함수

- 최소 2개 색상 정지점을 지정해야함
- 속성값
 - from angle, color deg, color deg ...



conic-gradient()

```
background-image: conic-gradient(at 70% 75%, red, yellow, green);
```



위치 속성

#12

position

요소에 대한 위치 지정방법의 기준을 설정하는 속성

- 속성값

- | | |
|------------|----------------|
| ▪ static | (기본값) 기준 : 없음 |
| ▪ relative | 기준 : 초기 위치 |
| ▪ absolute | 기준 : 부모 요소 |
| ▪ fixed | 기준 : 브라우저(뷰포트) |
| ▪ sticky | 기준 : 스크롤 영역 |



position

static

- 배치 기준 없음
- 페이지의 정상적인 흐름에 따라 배치
- top, bottom, left, right 속성에 영향 받지 않음



position

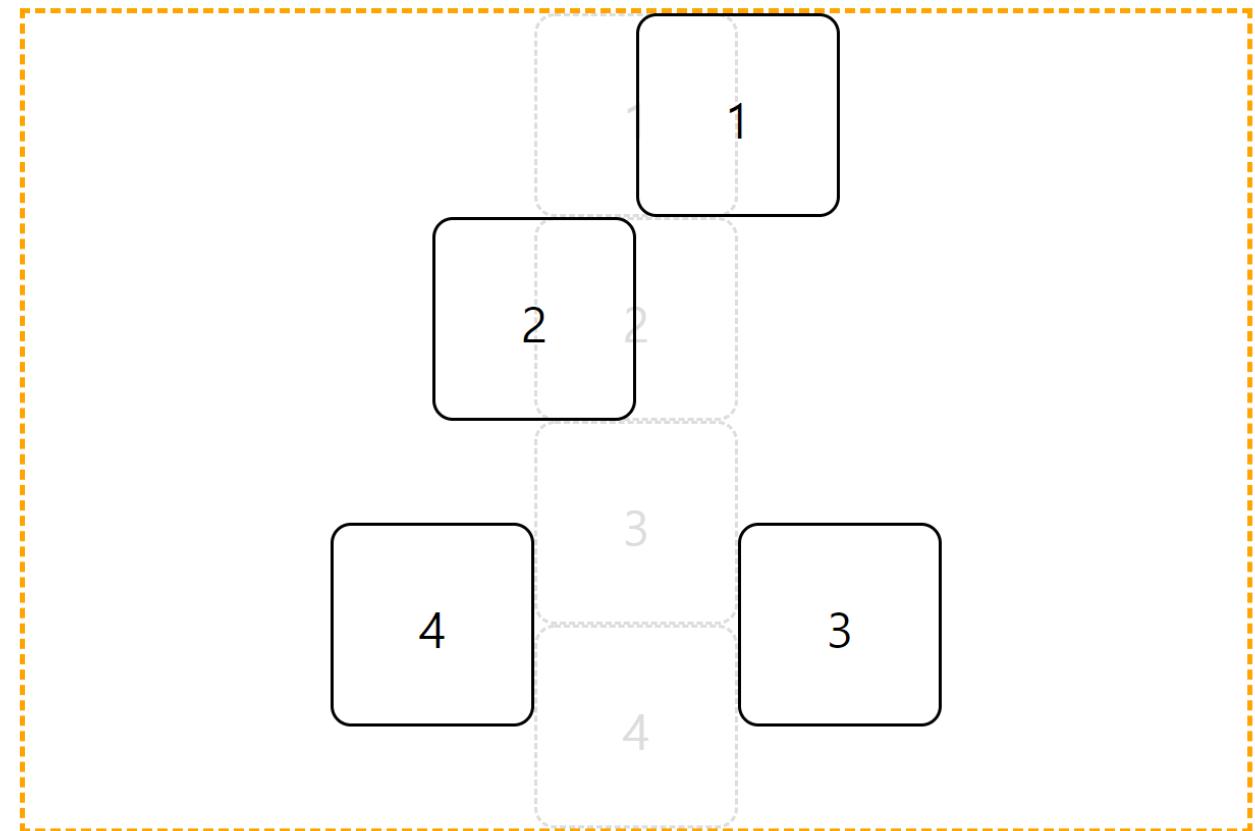
relative

- 초기 위치를 기준으로 상대적으로 배치
- top, bottom, left, right로 지정한 길이만큼
초기 위치에서 떨어진 위치에 배치



position

position : relative;



position

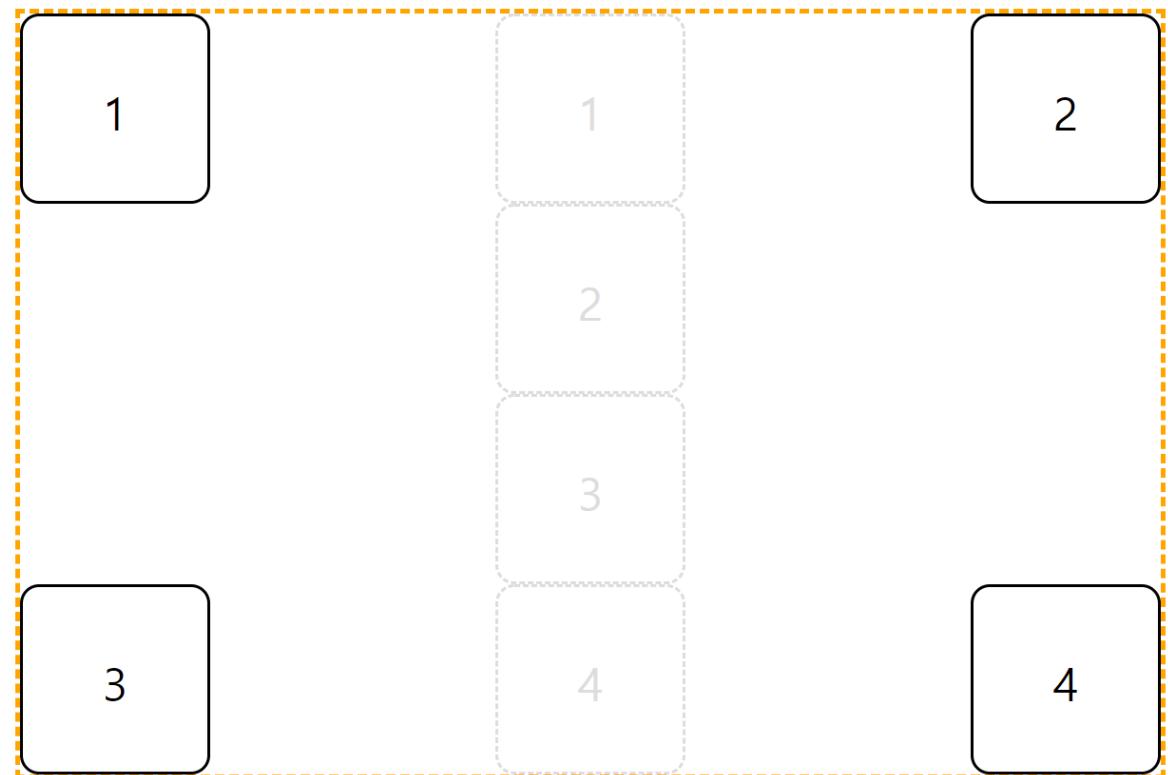
absolute

- 부모 요소를 기준으로 절대적인 위치로 배치
- 페이지의 정상적인 흐름에서 제외
- 다른 요소와 겹쳐져서(over-lap) 배치될 수 있음
- top, bottom, left, right 로 지정한 길이만큼
부모 요소에서 떨어진 위치에 배치



position

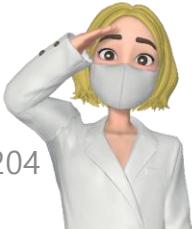
position : absolute;



position

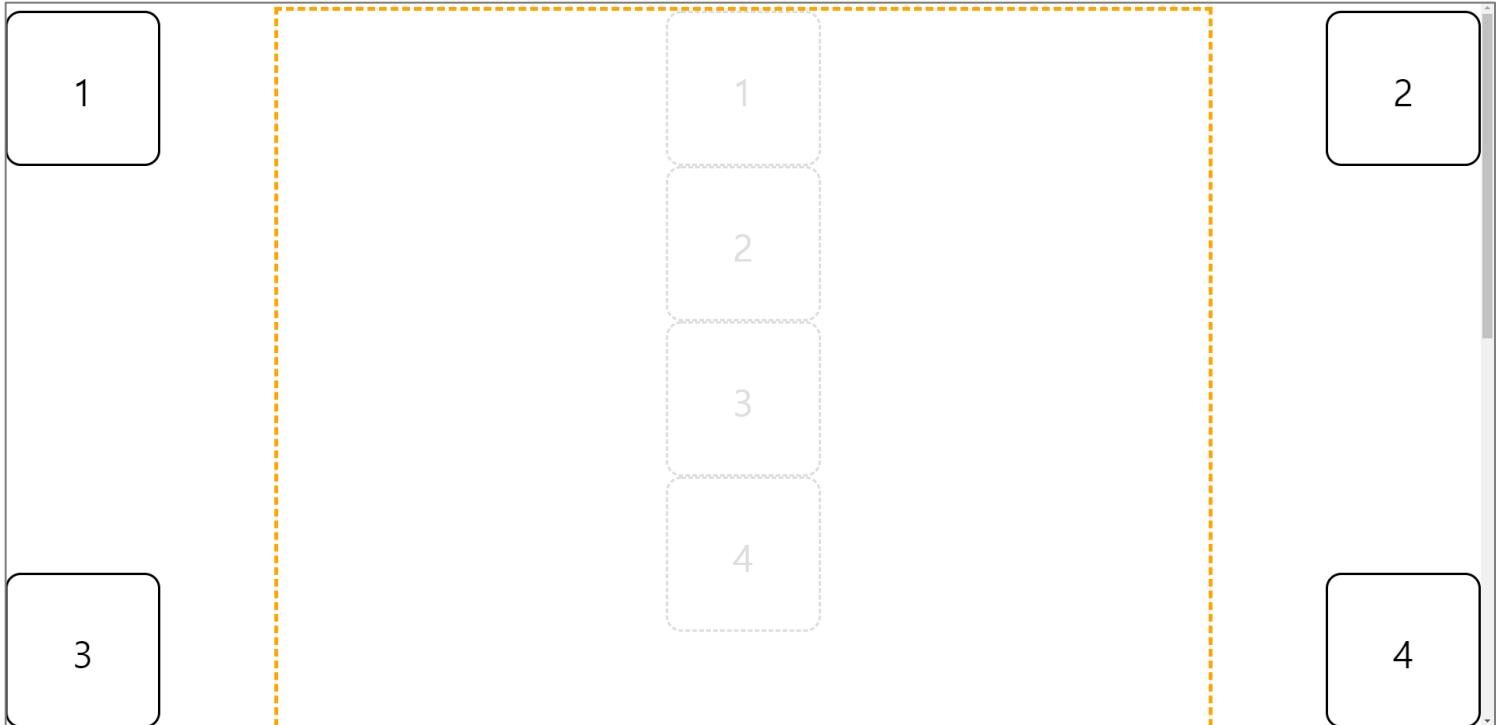
fixed

- 브라우저(뷰포트)를 기준으로 상대적인 위치로 배치
- 페이지를 스크롤해도 고정된 위치를 유지
- top, bottom, left, right 로 지정한 길이만큼
페이지의 테두리에서 떨어진 위치에 배치



position

position : fixed;



position

sticky

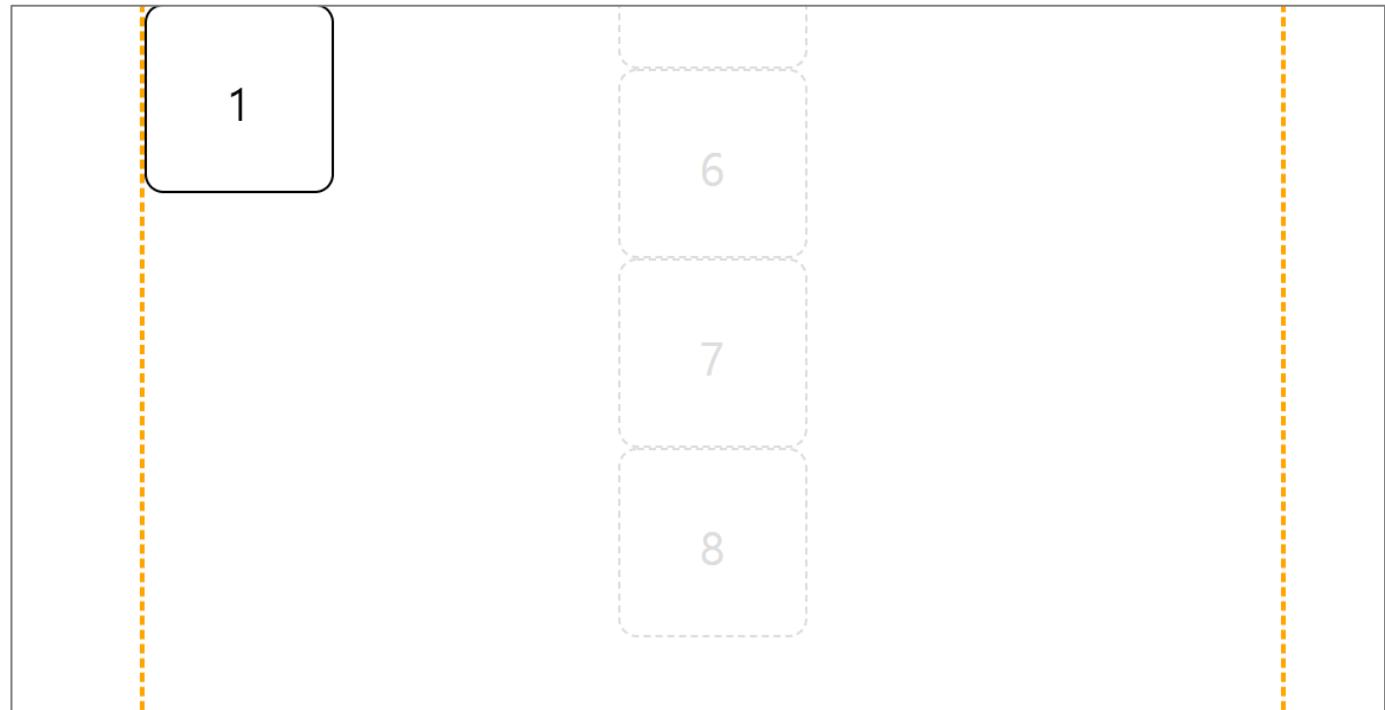
☞ relative + fixed

- 스크롤 위치를 기준으로 배치
- 스크롤 위치
 - 페이지 시작~요소의 위치 : relative
 - 그 이후 : fixed
- IE 에서는 지원하지 않음
- Safari 에서는 -webkit- prefix 가 필요함



position

position : sticky;



top bottom left right

position 기준에 따라 '상,하,좌,우'로부터의 거리를 지정하는 속성

- 속성값

- auto (기본값)
- length – px, cm, em
- %
 - top, bottom - 부모 요소의 세로 크기의 비율로 지정
 - left, right - 부모 요소의 가로 크기의 비율로 지정



과제01

Github 페이지 만들기😊

1. Github 사이트로 이동한다.
2. 로그인 한다.
3. New 버튼을 누른다.
4. 새로운 레포지토리(저장소)를 만든다.
5. VS code에서 [Git clone] 한다.
6. Index.html 파일하나 만들고 [커밋/푸시] 한다.
7. Settings 메뉴 > Page 메뉴로 이동한다.
8. main 을 선택하고 [Save] 한다.
9. 생성된 github 링크로 이동한다.



<https://github.com/>



GitHub x +

github.com

코리아교육그룹 웹... 코리아교육그룹 통...

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

wwalohacampus New

Repositories Find a repository...

wwalohacampus / sbs_web1
wwalohacampus / sbs_web2
wwalohacampus / mypage
wwalohacampus / barun
wwalohacampus / myprofile
wwalohacampus / matsam
wwalohacampus / test

Show more

Recent activity

When you take actions across GitHub, we'll provide links to that activity here.

Learn Git and GitHub without any code!

Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.

Read the guide Start a project

All activity

Introduce yourself

The easiest way to introduce yourself on GitHub is by creating a README in a repository about you! You can start here:

wwalohacampus / README.md

```
1 - 🌟 Hi, I'm @wwalohacampus
2 - 💬 I'm interested in ...
3 - 🌱 I'm currently learning ...
4 - 🚀 I'm looking to collaborate on ...
5 - 📩 How to reach me ...
6 -
```

Dismiss this Continue

Discover interesting projects and people to populate your personal news feed.

Your news feed helps you keep up with recent activity on repositories you [watch](#) or [star](#) and people you [follow](#).

Explore GitHub

ProTip! The feed shows you events from people you [follow](#) and repositories you [watch](#) or [star](#).

Subscribe to your news feed

© 2021 GitHub, Inc. Bloo API Terms

Copyright © ALOHA CLASS. All Rights Reserved

Create a New Repository x +

github.com/new

코리아교육그룹 웹... 코리아교육그룹 통...

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

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Owner * wwwalohacampus Repository name * profile

Great repository names are short and memorable. Need inspiration? How about miniature-octo-fiesta?

Description (optional) 나의 프로필

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](#).

Choose a license A license tells others what they can and can't do with your code. [Learn more](#).

This will set `main` as the default branch. Change the default name in your [settings](#).

Create repository

Copyright © ALOHA CLASS. All Rights Reserved

▶

보기 > 명령 팔레트

> git clone

> 개요

최근 항목

최근에 사용한 항목

기타 명령

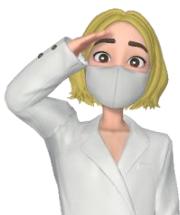
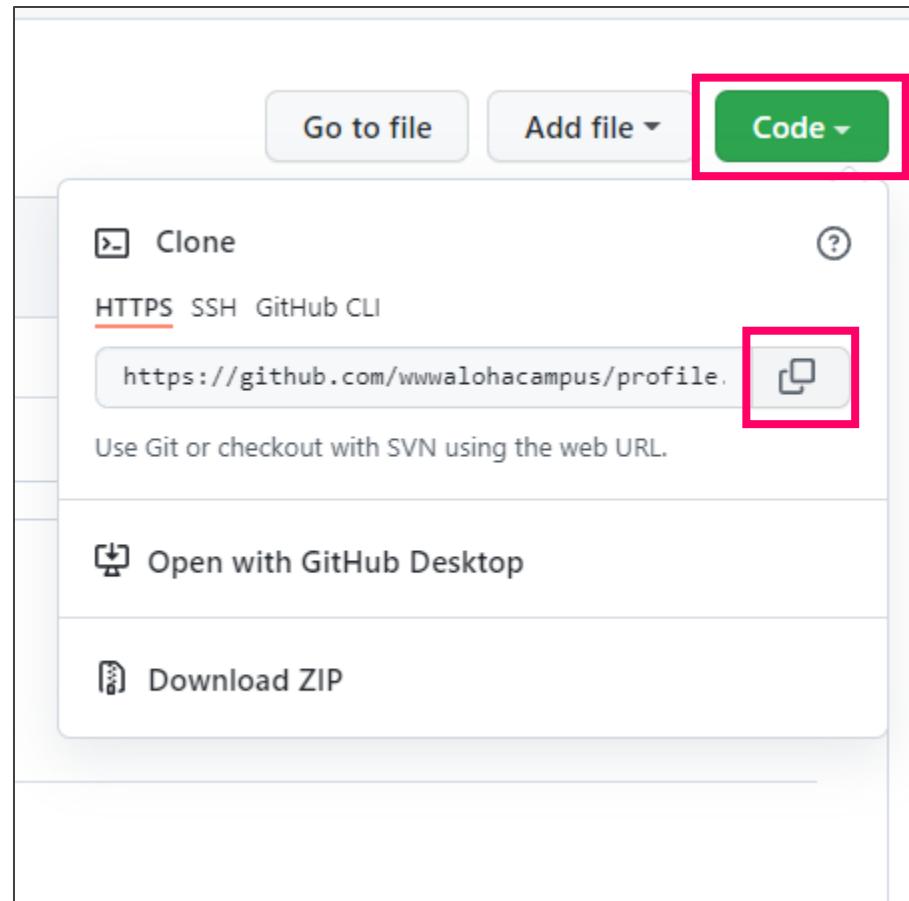
VS Code를 자신의 것으로 만들기 위한 가장 적합한 사용자 지정을 발견하세요.

기본 사항 알아보기

VS Code 오른쪽으로 이동하여 필수 기능에 대한 개요를 확인하세요.

생산성 향상

Copyright © ALOHA CLASS. All Rights Reserved

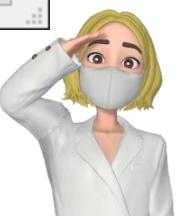
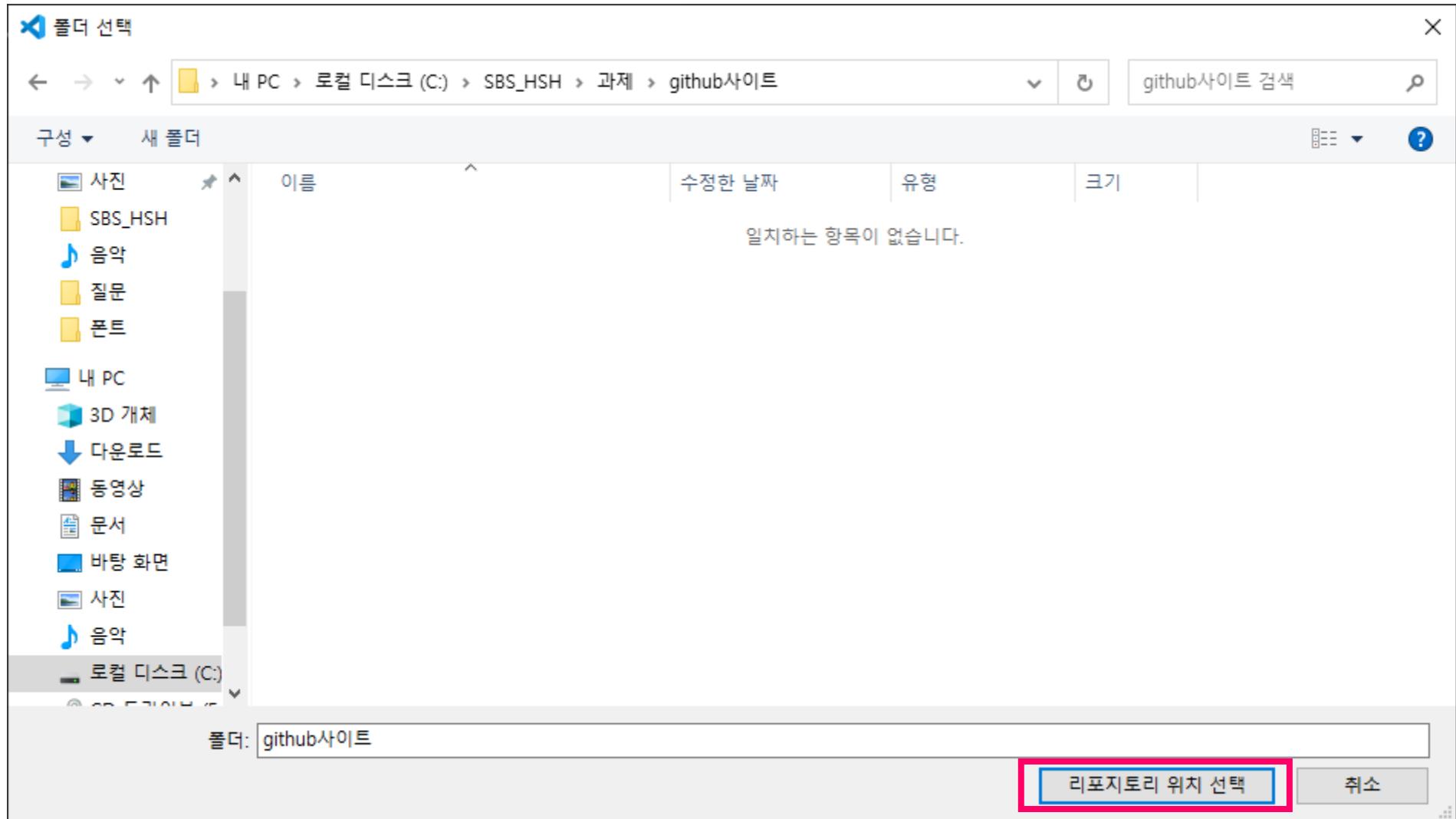




소스 제어 <https://github.com/wwwalohacampus/profile.git>

메시지('main'에서) 리포지토리 URL <https://github.com/wwwalohacampus/profile.git>

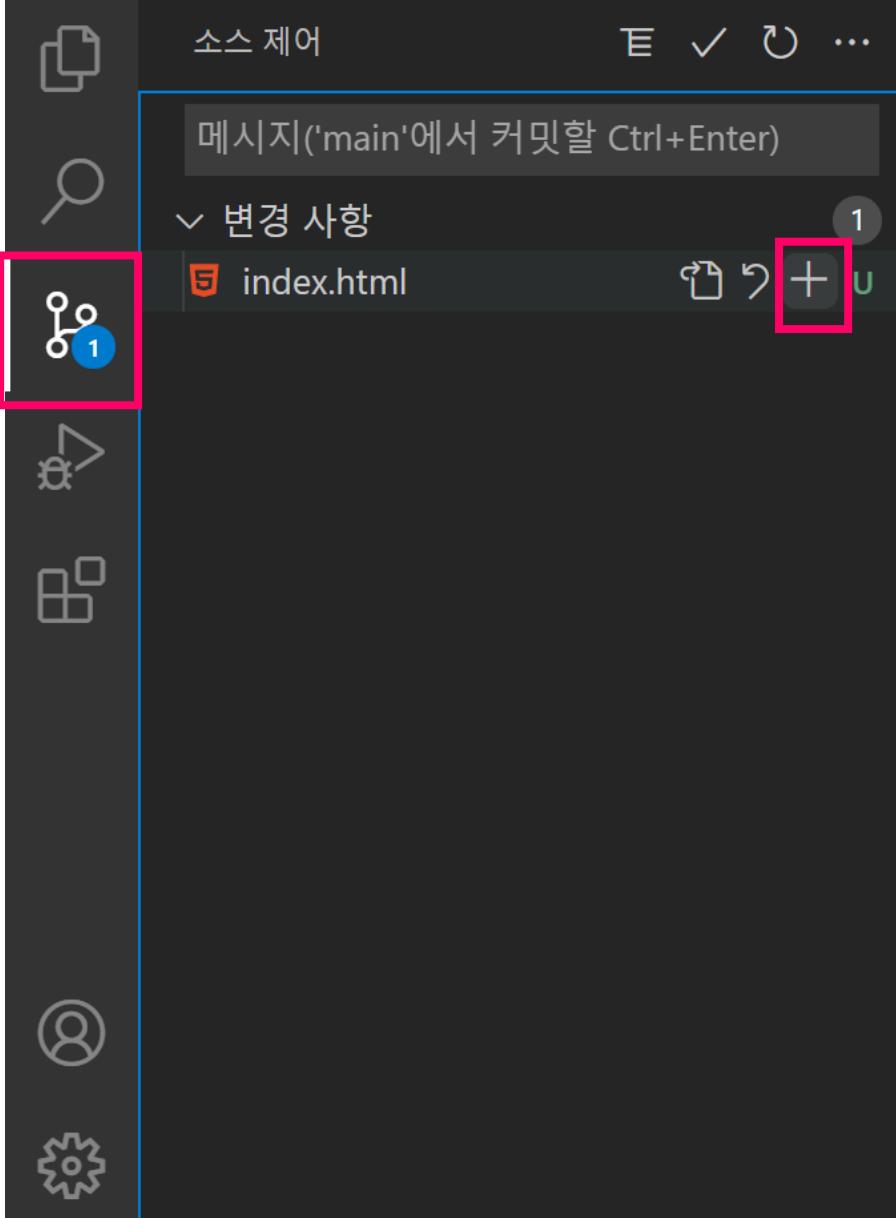
변경 사항 GitHub에서 복제



The screenshot shows the Visual Studio Code interface with the following details:

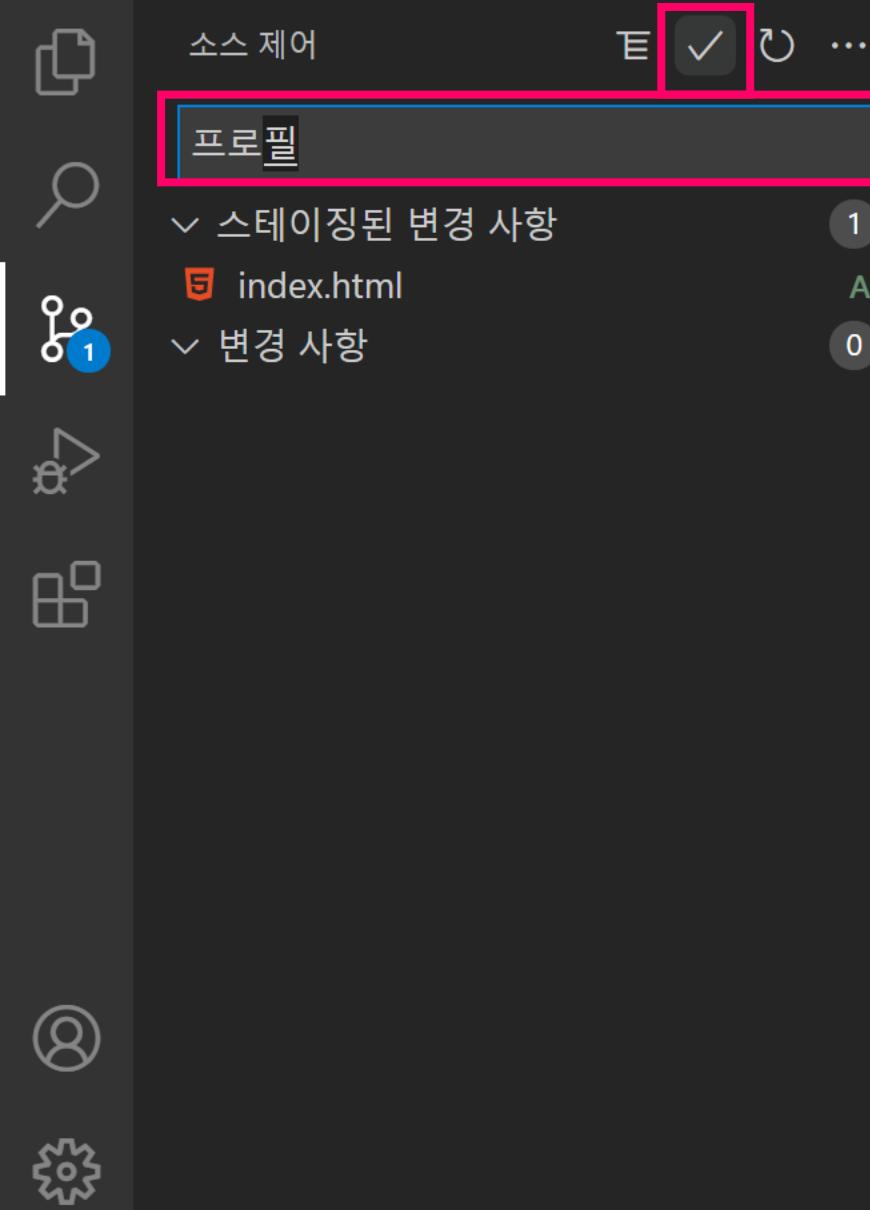
- Left Sidebar:** Contains icons for File Explorer, Search, Problems (with 1 notification), and others.
- File Explorer:** Shows a folder named "PROFILE" containing "index.html" (selected) and "README.md".
- Central Editor:** Displays the content of "index.html".

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7     <title>나의 프로필</title>
8   </head>
9   <body>
10    <h1>SBS 아카데미</h1>
11    <h3>ALOHA 의 프로필 페이지 입니다:)</h3>
12  </body>
13 </html>
```
- Bottom Status Bar:** Shows "main*" in the bottom-left, and "줄 1, 열 16" along with other status information like encoding and line endings.



The main editor window displays the file 'index.html' with the following content:

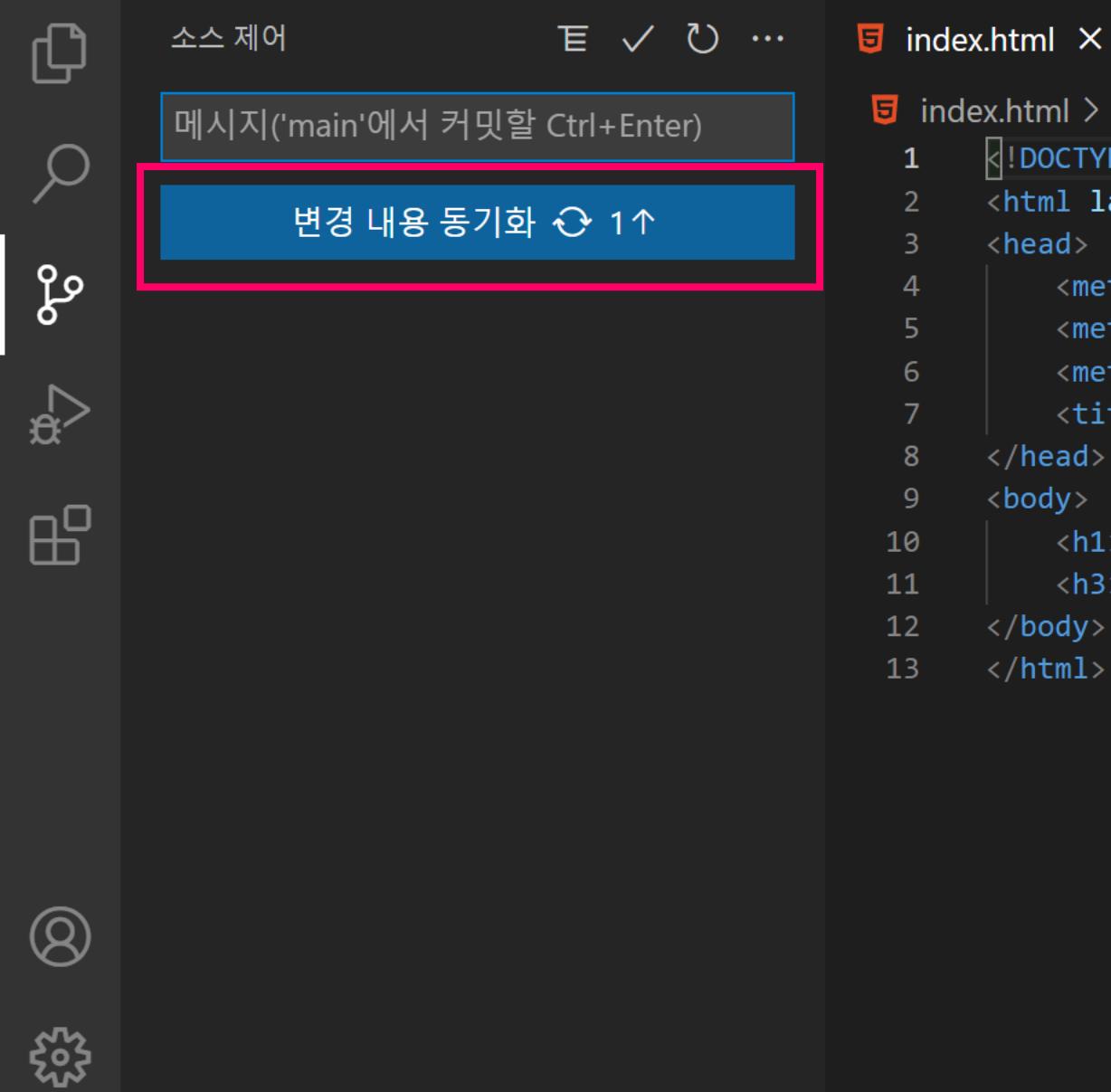
```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7     <title>나의 프로필</title>
8 </head>
9 <body>
10    <h1>SBS 아카데미</h1>
11    <h3>ALOHA 의 프로필 페이지 입니다:)</h3>
12 </body>
13 </html>
```



index.html A X

index.html > ...

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7     <title>나의 프로필</title>
8 </head>
9 <body>
10    <h1>SBS 아카데미</h1>
11    <h3>ALOHA 의 프로필 페이지 입니다:)</h3>
12 </body>
13 </html>
```



index.html ×

index.html > ...

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0, shrink-to-fit=no">
7     <title>나의 프로필</title>
8   </head>
9   <body>
10    <h1>SBS 아카데미</h1>
11    <h3>ALOHA 의 프로필 페이지 입니다:)</h3>
12  </body>
13 </html>
```



wwwwalohacampus/profile: 나의 프로필

github.com/wwwwalohacampus/profile

코리아교육그룹 웹... 코리아교육그룹 투자자 관리

일기 목록

Search or jump to...

Pull requests Issues Marketplace Explore

wwwwalohacampus / profile Public

Unwatch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main branch 0 tags

Go to file Add file Code

wwwwalohacampus 프로필 c21f1f8 28 seconds ago 2 commits

README.md Initial commit 4 minutes ago

index.html 프로필 28 seconds ago

README.md

profile

나의 프로필

About

나의 프로필

Readme

Releases

No releases published Create a new release

Packages

No packages published Publish your first package

Languages

HTML 100.0%

Options x +

github.com/wwwalohacampus/profile/settings

코리아교육그룹 웹... 코리아교육그룹 통...

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

wwwalohacampus / profile Public

Unwatch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Options

Manage access

Security & analysis

Branches

Webhooks

Notifications

Integrations

Deploy keys

Actions

Environments

Secrets

Pages

Moderation settings

Settings

Repository name profile Rename

Template repository

Template repositories let users generate new repositories with the same directory structure and files. [Learn more.](#)

Social preview

Upload an image to customize your repository's social media preview.

Images should be at least 640×320px (1280×640px for best display). [Download template](#)

Edit

Features

Pages

github.com/wwwalohacampus/profile/settings/pages

코리아교육그룹 웹... 코리아교육그룹 틀...

wwwalohacampus / profile Public

Unwatch 1 Star 0 Fork 0

Pages Issues Pull requests Actions Projects Wiki Security Insights Settings

Options

Manage access

Security & analysis

Branches

Webhooks

Notifications

Integrations

Deploy keys

Actions

Environments

Secrets

Pages

Moderation settings

GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Source

GitHub Pages is currently disabled. Select a source below to enable GitHub Pages for this repository. [Learn more.](#)

Branch: main / (root) Save

Select branch

Select branch

main

None

Try it risk-free for 14 days using a GitHub Enterprise organization, or learn more about changing the visibility of your GitHub Pages site.



Pages

github.com/wwwalohacampus/profile/settings/pages

코리아교육그룹 웹... 코리아교육그룹 통...

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

GitHub Pages source saved.

wwwalohacampus / profile Public

Unwatch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Options

Manage access

Security & analysis

Branches

Webhooks

Notifications

Integrations

Deploy keys

Actions

Environments

Secrets

Pages

Moderation settings

GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Your site is ready to be published at <https://wwwalohacampus.github.io/profile/>

Source

Your GitHub Pages site is currently being built from the `main` branch. [Learn more](#).

Branch: main / (root) Save

Theme Chooser

Select a theme to publish your site with a Jekyll theme. [Learn more](#).

Choose a theme

Custom domain

Custom domains allow you to serve your site from a domain other than `wwwalohacampus.github.io`. [Learn more](#).

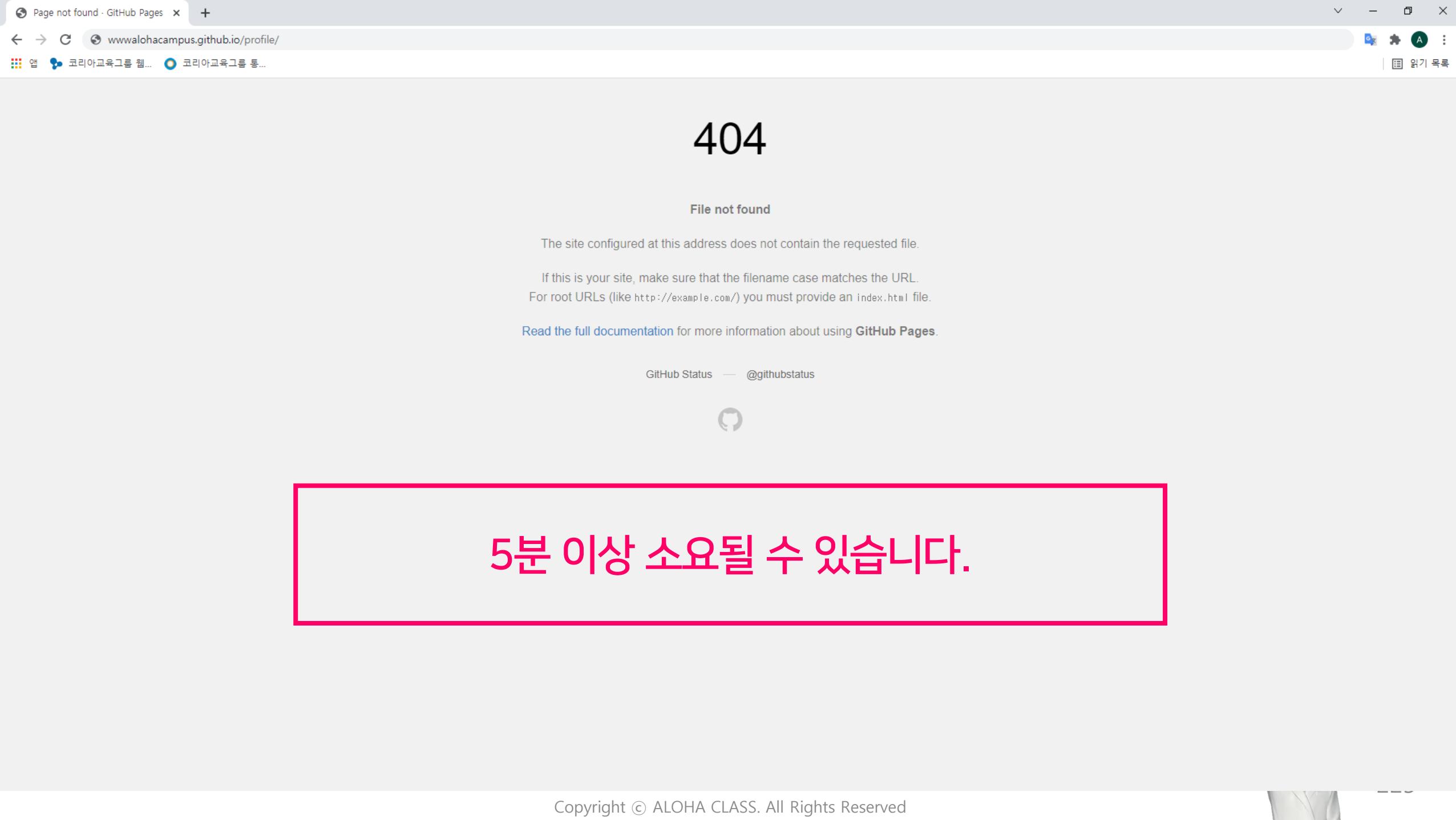
Save Remove

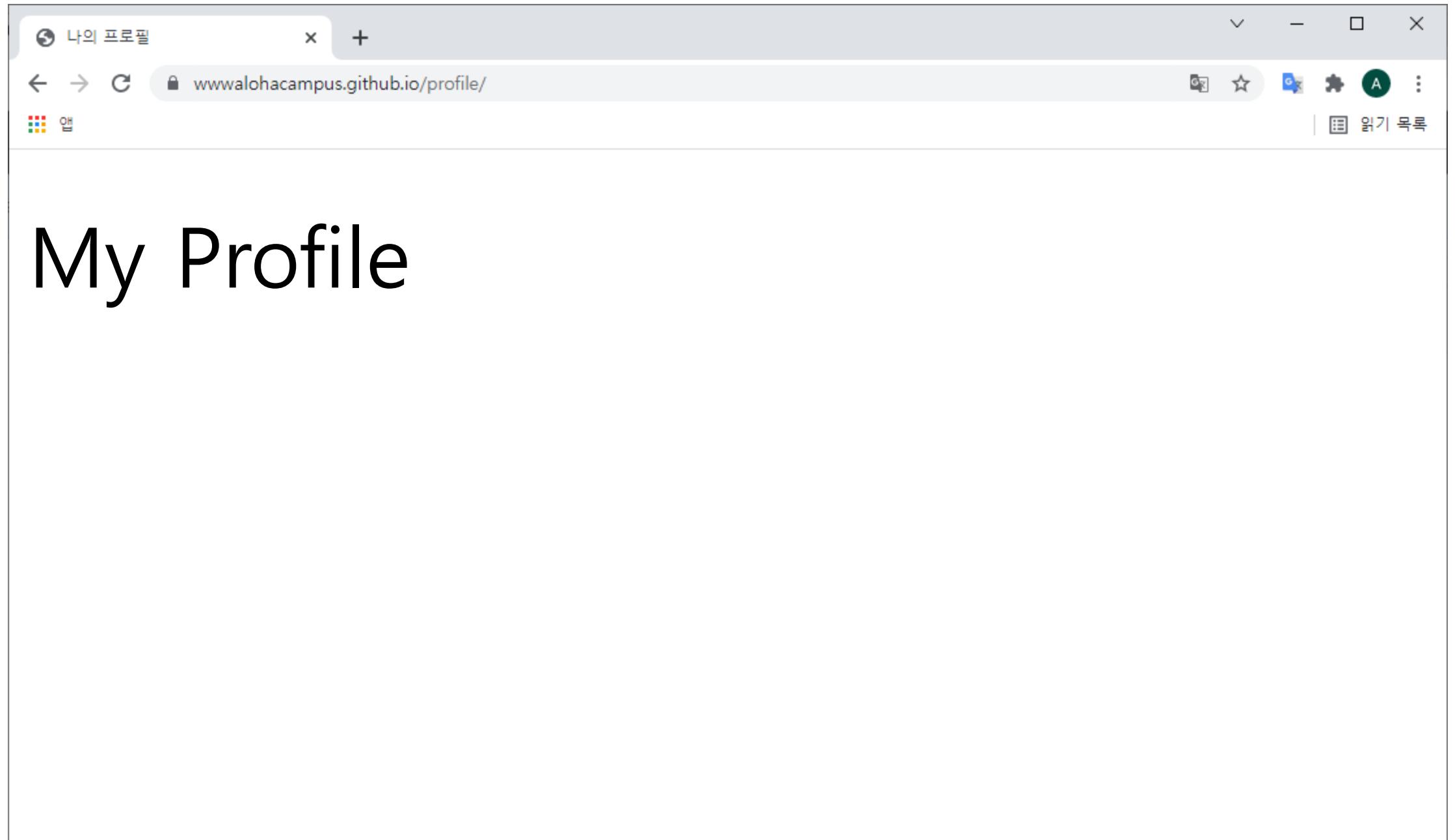
Enforce HTTPS

Required for your site because you are using the default domain (`wwwalohacampus.github.io`)

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](#).

Copyright © ALOHA CLASS. All Rights Reserved





과제 끝!



변형 속성

#13

transform

요소를 변형시키는 속성

transform: 변형함수1 변형함수2 ...;

- **변형**

- 이동 - translate
- 회전 - rotate
- 크기조정 - scale
- 기울임 - skew

- ...



transform

2차원 변형 함수

함수	설명	단위
translate(x, y)	이동(X,Y)	기본 단위
translateX(x)	이동(X)	기본 단위
translateY(y)	이동(Y)	기본 단위
scale(x, y)	크기조정(X,Y)	배율
scaleX(x)	크기조정(X)	배율
scaleY(y)	크기조정(Y)	배율
rotate(deg)	회전(각도)	deg
skew(x-deg, y-deg)	기울임(X,Y)	deg
skewX(x-deg)	기울임(X)	deg
skewY(y-deg)	기울임(Y)	deg
matrix(n,n,n,n,n,n)	2차원 변형 (6 개의 값)	-



transform

3차원 변형 함수

함수	설명	단위
translate3d(x, y)	이동(X,Y)	기본 단위
translateZ(z)	이동(Z)	기본 단위
scale3d(x, y, z)	크기조정(X,Y,Z)	배율
scaleZ(z)	크기조정(Z)	배율
rotate3d(x, y, z, a)	회전(X,Y,Z,deg)	- , deg
rotateZ(z)	회전(Z)	deg
perspective(n)	원근법	deg
matrix(n,n,n,n,n,n)	2차원 변환 (6 개의 값)	-



변환 속성

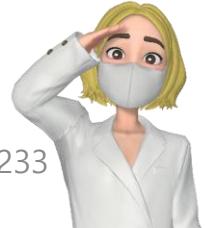
#14

transition

요소의 움직임을 구현하는 속성

```
transition: width 2s linear 1s;
```

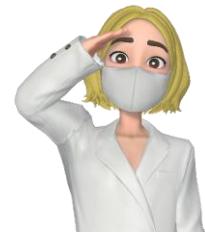
- 단축 속성
 - [property] [duration] [timing] [delay]
 - 개별속성
 - transition-property 스타일 속성 지정
 - transition-duration 지속시간 지정
 - transition-timing-function 속도 곡선 지정
 - transition-delay 자연시간 지정



transition

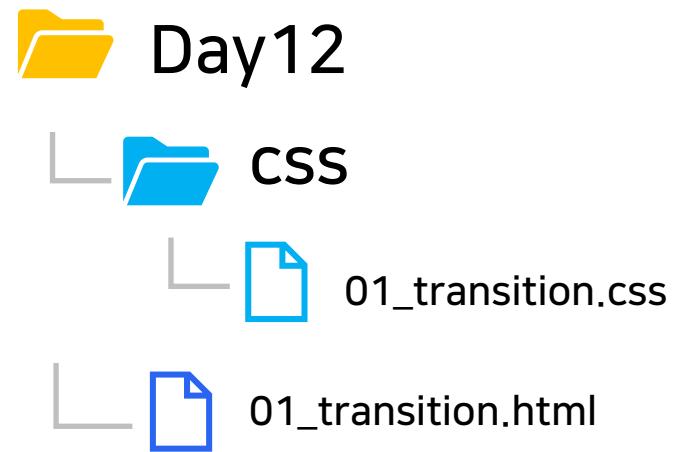
변환 속성과 함께 사용하는 스타일 속성

함수	설명
위치 속성	top, left, bottom, right
크기 속성	width, height
박스 속성	margin, padding
테두리 속성	border-width, border-radius, border-color
색상 속성	color, background-color
투명도 속성	opacity
변형 속성	transform



실습

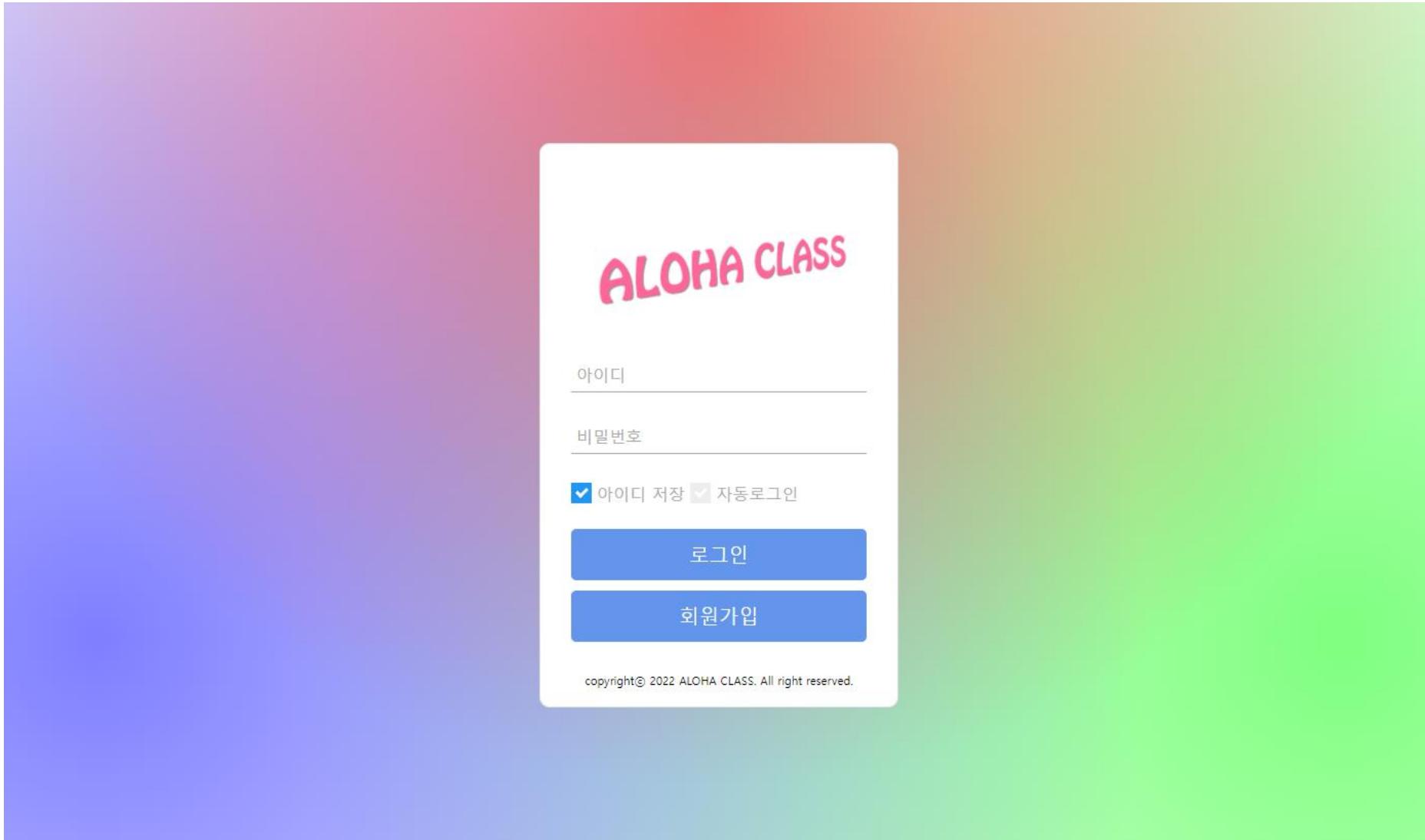
여러가지 변형 속성



로그인 페이지 만들기

#15

로그인 페이지 만들기



박스 수평/수직 가운데 배치하기



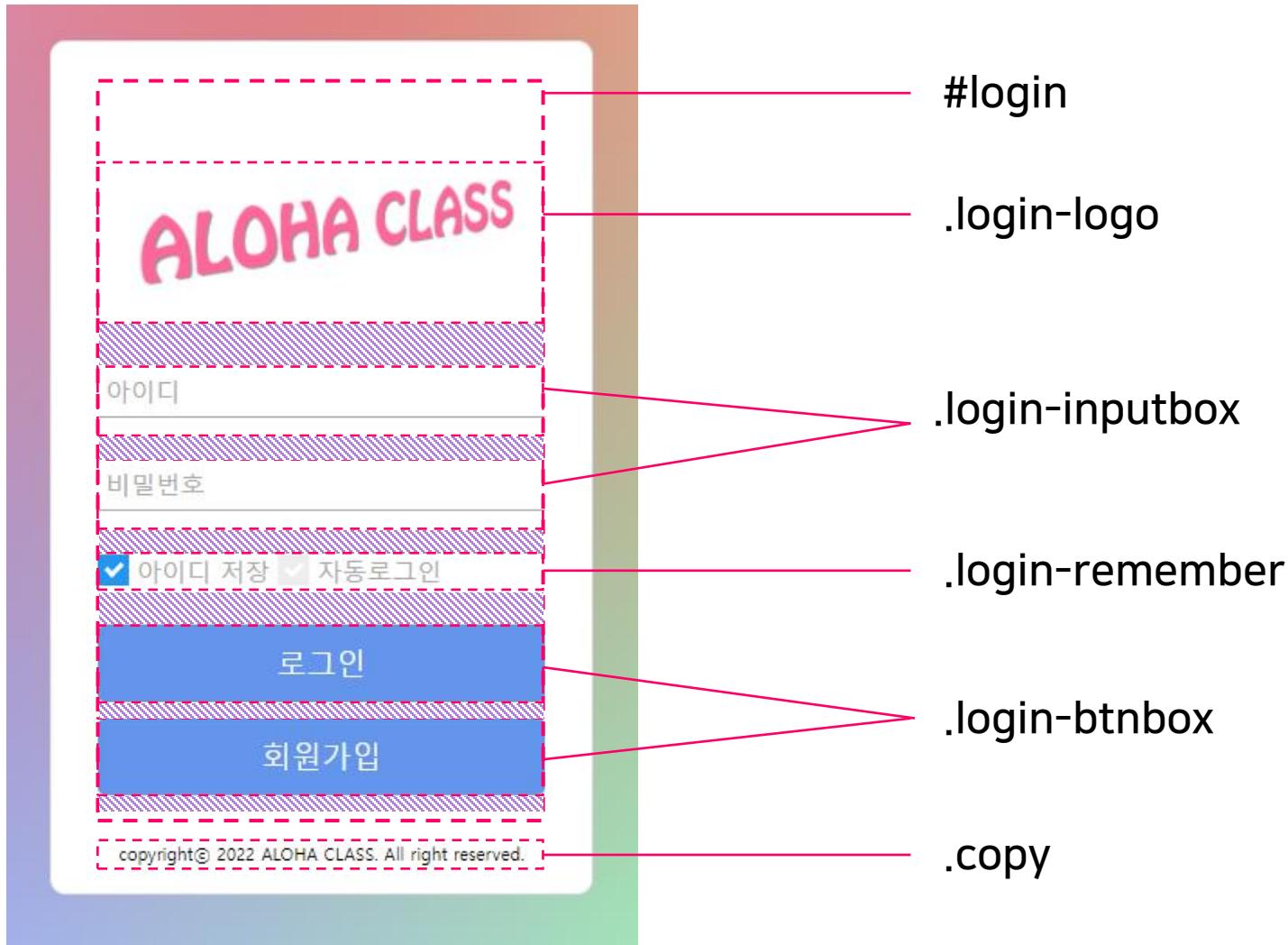
```
1 <div class="container">
2   <form action="" method="post">
3     <div class="login-wrap">
4
5     </div>
6   </form>
7 </div>
```



```
1 .login-wrap {
2   position: absolute;
3   top: 50%;
4   left: 50%;
5   transform: translate(-50%, -50%);
6   width: 350px;
7   height: 550px;
8   padding: 30px;
9   background-color: #ffffff;
10  border: 0.2px solid #ddd;
11  border-radius: 10px;
12  box-sizing: border-box;
13 }
```



영역 배치



실습

로그인 페이지 만들기

📁 Day13

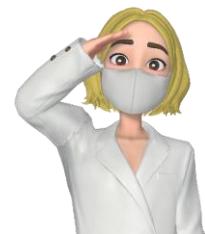
 └ 📄 login.html

 └ 📁 CSS

 └ 📄 login.css

 └ 📁 img

 └ 📄 logo.png



애니메이션 속성

#16



애니메이션 속성

속성	설명
animation	모든 animation 속성을 지정하는 단축 속성
animation-delay	이벤트 발생 후, 재생 전까지 지연시간을 지정하는 속성
animation-direction	애니메이션 진행 방향을 지정하는 속성
animation-duration	애니메이션 재생 시간을 지정하는 속성
animation-iteration-count	애니메이션 반복 횟수를 지정하는 속성
animation-name	애니메이션 이름을 지정하는 속성
animation-play-state	애니메이션 재생 상태를 지정하는 속성
animation-timing-function	수치 변형 함수를 지정하는 속성



@keyframes

애니메이션이 변경되는 지점을 설정하는 속성

- 0% or from : 시작 위치
- 100% or to : 종료 위치



@keyframes

애니메이션이 변경되는 지점을 설정하는 속성

- 0% or from : 시작 위치
- 100% or to : 종료 위치

```
● ● ●  
#wrap {  
    animation-name : 애니메이션 속성 이름;  
}  
  
@keyframes 애니메이션 이름 {  
    0% {}  
    50% {}  
    100% {}  
}
```



CSS 레이아웃 이해하기

#17



CSS 레이아웃

- 정적 레이아웃
- 동적 레이아웃
- 적응형 레이아웃
- 반응형 레이아웃





정직 레이아웃



영역의 크기가 고정됨

- 크기가 변하지 않고 고정된 레이아웃

The screenshot shows the Naver homepage with several fixed layout elements:

- Header:** A large green NAVER logo with a yellow bird icon on the left, a search bar with a green border in the center, and a green search button on the right.
- Top Navigation Bar:** A horizontal bar with links for Mail, Cafe, Blog, etc., and a weather forecast showing 15.0°C.
- Advertisement:** A large banner for "sofa new products early bird special" featuring a woman sitting on a sofa.
- User Authentication:** A green "NAVER 로그인" button on the right side.
- Footer:** A navigation bar at the bottom with links for Newsstand, News, Sports, and Economy, along with a footer menu.





동적 레이아웃



영역 크기를 브라우저 너비에 비율로 적용

- 브라우저 너비와 상관없이 화면이 꽉 채워지는 레이아웃

The screenshot shows a web browser window with a dark theme. The address bar displays 'dinfree.com/lecture/backend/javaweb_1.1.html'. The page content is a Java web development tutorial. On the left, there's a sidebar with a navigation menu:

- 자바 웹 기초
 - 1. 자바 웹 개발 시작하기
 - 1.1 자바 웹 개발 개요
 - 1.2 개발 환경 구축
 - 1.3 HTTP, WAS, 웹애플리케이션
 - 2. 서블릿과 JSP
 - 2.1 서블릿
 - 2.2 JSP
 - 2.3 내장객체와 속성관리
 - 3. MVC 패턴과 컨트롤러
 - 3.1: MVC 패턴





적응형 레이아웃



영역의 크기가 고정됨

- 브라우저 너비에 따라 변경되는 레이아웃

The screenshot shows a web browser window displaying the [w3schools.com/html/default.asp](https://www.w3schools.com/html/default.asp) page. The browser interface includes a title bar, address bar, and various toolbars. The main content area features a sidebar on the left with a navigation menu for the 'HTML Tutorial' section, including links like 'HTML HOME', 'HTML Introduction', and 'HTML Editors'. The main content area has a large title 'HTML Tutorial' and two green buttons: '< Home' and 'Next >'. Below the title, there is a green box containing text about HTML being a standard markup language for Web pages and its use in creating websites. A purple hand-drawn arrow points from the text '영역의 크기가 고정됨' to the sidebar area, indicating that the sidebar's width remains constant regardless of the browser window size.





반응형 레이아웃



네비에 따라 비율 적용도 되면서,
배치의 변화도 있는 레이아웃

- 디바이스(디스플레이)의 종류에 따라 영역의 크기가 변하는 레이아웃

The screenshot shows a Google Classroom page titled 'WEB 2 과정 SBS Academy 학습'. The top navigation bar includes tabs for '스트림', '수업' (selected), '사용자', and '성적', along with various icons. On the left, there's a sidebar with categories like '모든 주제', '가이드' (selected), '수업자료', '실습자료', '과제', and 'TIP'. The main content area has three sections: '가이드' (with a card for 'MAC 사용자 설치가이드 - VSCode, Homebre...'), '수업자료' (with a card for '[WEB 2] 1일차 수업자료'), and '실습자료'. Each card includes a preview image, a title, a timestamp, and a three-dot menu icon. A cartoon character is visible in the bottom right corner.

CSS 기본 레이아웃 만들기

#18

기본 레이아웃



Web Design,Your Future

끊임없이 연습하고, 빈틈없이 공부해요.:)

웹 디자인서 | 웹 디자인너 |



1 HTML, 기초 개념과 실습
필수적인 내용과 다양한 실무 연습
책쓰고 써온 저마다운 대(大)작자
그들은 서재에서 책 모으고 코드를 만드는데

2 이론적인 것보다
실제에 적용되는 CSS 레이아웃
설계자, 디자인자, 프론트엔드 개발자에게
웹 과정 학습을 위한 전략과 방법을 배우세요.

3 웹 디자인 기능사
작업할 실기 대비도 OK!
2022년 새롭게 바뀐 평가기준에 대비합니다.
작업을 실기는 평가를 공유하고 해야 한 번 헷갈리지
않도록 좋은 품질로 보여줄 수 있습니다.

This is a screenshot of a website page. At the top, there is a section titled "PROBLEMS" with the subtext "이런 고민을 가진 분들을 추천합니다.". Below this, there are four circular icons, each with a symbol and a brief description: 1. "마이크로소프트 키보드" (Microsoft Keyboard), "마이크로소프트 디스플레이" (Microsoft Display), "Photoshop, Illustrator 등 소프트웨어 사용법", and "인터넷 끊김 문제" (Internet Disconnection Problem).

수강생	WEB 02	기간	2021.10~2021.11
수강료	11,000원	교재가격	100% 저작권자
개강일	2021.10~2021.11	수강여부	WEB 전문 튜토링수강

Copyright 2021. ALOHA CLASS. All Rights Reserved.



기본 레이아웃

Web Design, Your Future

끊임없이 연습하고, 빈틈없이 공부해요:)

| 웹 피플리서 | 웹 디자이너 |



1 HTML 기초 개념과 실습
필수적인 내용과 다양한 실무 연습
베스트 대고부터 레이아웃 대고까지
그동안 웹 사이트에서 보던 모든 코드를 만나요

2 이론적인 것보다
실제에 적용되는 CSS 레이아웃
업무에서도 쉽게 활용할 수 있는 난이도
웹 과정 입문에 가장 적합한 학습방법을 제시합니다.
다음 과정 학습을 위한 연계에도 많은 도움이 됩니다.

3 웹 디자인 기능사
작업형 실기는 평판을 공부하고 여러 번 반복하여
연습해야 좋은 결과를 얻을 수 있습니다.
2022년 새롭게 바뀐 출제기준에 대비합니다.

PROBLEMS

이런 고민을 가진 분들께 추천합니다.



미디어쿼리와 같은 미리보기 기능이 아예 웹 디자인 기능에서 누락됨
Photoshop, Illustrator 등을 사용하는 차이로 인해 수평선과 같은 기본적인 공유한 것들은 서로 다른 표기로 표시되는 경우
웹 디자인 조건에서 항상 아름답게 실수하는 일은 수험생에게는 문제
미디어쿼리와 같은 미리보기 기능이 아예 웹 디자인 기능에서 누락됨
Photoshop, Illustrator 등을 사용하는 차이로 인해 수평선과 같은 기본적인 공유한 것들은 서로 다른 표기로 표시되는 경우
웹 디자인 조건에서 항상 아름답게 실수하는 일은 수험생에게는 문제

수강생	WEB 02	기간	2021.01~2021.12
수강료	10,000원	교재기관	SBS 아카데미
접수기간	2021.01~2021.12	수강대상	WEB 일문 및 WEB 01 수료생

Copyright 2022. ALOHA CLASS. All Rights Reserved.



Web Design, Your Future

끊임없이 연습하고, 빈틈없이 공부해요:)

| 웹 피플리서 | 웹 디자이너 |



1 HTML 기초 개념과 실습
필수적인 내용과 다양한 실무 연습
베스트 대고부터 레이아웃 대고까지
그동안 웹 사이트에서 보던 모든 코드를 만나요

2 이론적인 것보다
실제에 적용되는 CSS 레이아웃
업무에서도 쉽게 활용할 수 있는 난이도
웹 과정 입문에 가장 적합한 학습방법을 제시합니다.
다음 과정 학습을 위한 연계에도 많은 도움이 됩니다.

3 웹 디자인 기능사
작업형 실기는 평판을 공부하고 여러 번 반복하여
연습해야 좋은 결과를 얻을 수 있습니다.
2022년 새롭게 바뀐 출제기준에 대비합니다.

PROBLEMS

이런 고민을 가진 분들께 추천합니다.



미디어쿼리와 같은 미리보기 기능이 아예 웹 디자인 기능에서 누락됨
Photoshop, Illustrator 등을 사용하는 차이로 인해 수평선과 같은 기본적인 공유한 것들은 서로 다른 표기로 표시되는 경우
웹 디자인 조건에서 항상 아름답게 실수하는 일은 수험생에게는 문제
미디어쿼리와 같은 미리보기 기능이 아예 웹 디자인 기능에서 누락됨
Photoshop, Illustrator 등을 사용하는 차이로 인해 수평선과 같은 기본적인 공유한 것들은 서로 다른 표기로 표시되는 경우
웹 디자인 조건에서 항상 아름답게 실수하는 일은 수험생에게는 문제

수강생	WEB 02	기간	2021.01~2021.12
수강료	10,000원	교재기관	SBS 아카데미
접수기간	2021.01~2021.12	수강대상	WEB 일문 및 WEB 01 수료생

Copyright 2022. ALOHA CLASS. All Rights Reserved.

레이아웃 모듈

- 헤더, 메뉴

#19



레이아웃 모듈

- 헤더
- 메뉴
- 섹션
- 푸터



레이아웃 모듈

- 섹션, 푸터

#20



레이아웃 모듈

- 헤더
- 메뉴
- 섹션
- 푸터



Flex 레이아웃

#21



Flexbox 이전의 레이아웃

- block section
- inline text
- table
- position

이제 `float`, `position` 속성으로

레이아웃을 지정하지 않아도 유연한 반응형 레이아웃을 만들 수 있다!





Flexible box

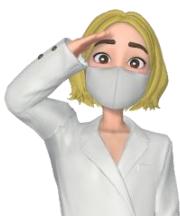
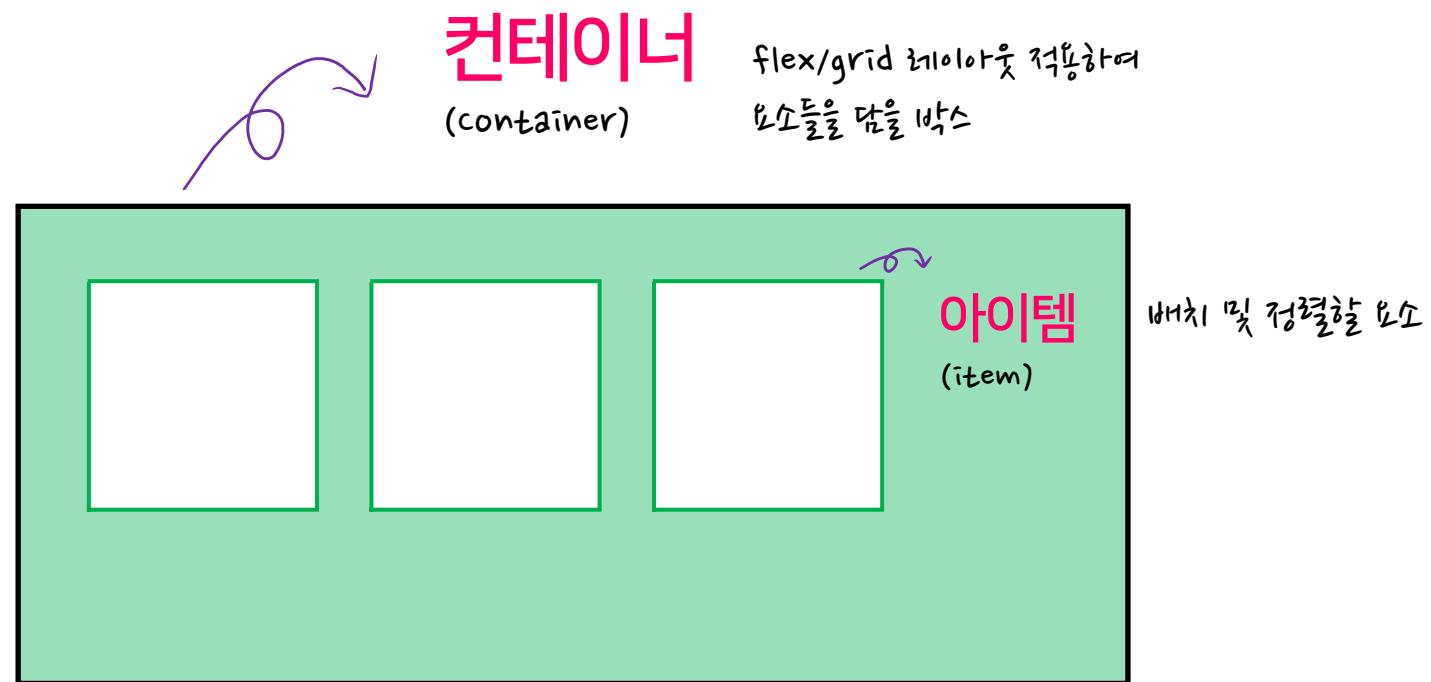
- **요소의 크기가 동적인 경우에도 요소를 배치 및 정렬할 수 있는 여러 가지 효율적인 방법을 제공하는 레이아웃 모델**



🌿 Container & item



▪ 핵심 요소





Flex Container

Container 를 위한 속성

속성	설명
display	Flex Container를 정의
flex-flow	flex-direction와 flex-wrap의 단축 속성
flex-direction	Flex Items의 주 축(main-axis)을 설정
flex-wrap	Flex Items의 여러 줄 묶음(줄 바꿈) 설정
justify-content	주 축(main-axis)의 정렬 방법을 설정
align-content	교차 축(cross-axis)의 정렬 방법을 설정(2줄 이상)
align-items	교차 축(cross-axis)에서 Items의 정렬 방법을 설정(1줄)



Flex Container

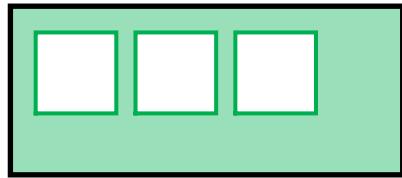
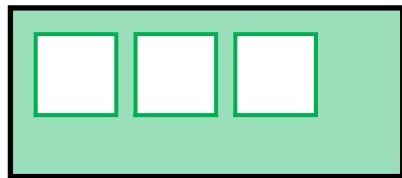
display

속성값	설명
flex	Block 특성의 Flex Container를 정의
inline-flex	Inline 특성의 Flex Container를 정의

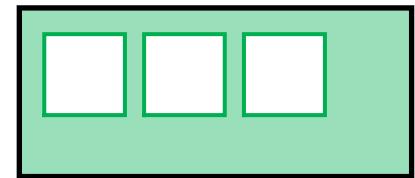
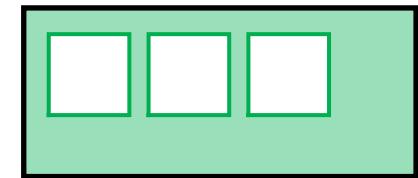


display

`display : flex;`



`display : inline-flex;`





Flex Container

flex-flow

단축 속성

- **flex-direction**
 - item 의 주 축(row, column) 방향을 지정

- **flex-wrap**
 - item 의 wrap 여부 (줄바꿈)를 지정





Flex Container

flex-direction

주축의 방향을 지정

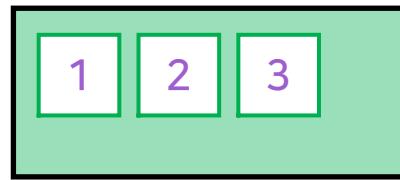
속성	설명
row	Item 을 수평축(왼쪽에서 오른쪽으로)으로 표시
row-reverse	Item 을 row의 반대 축으로 표시
column	Item 을 수직축(위에서 아래로)으로 표시
column-reverse	Item 을 column의 반대 축으로 표시



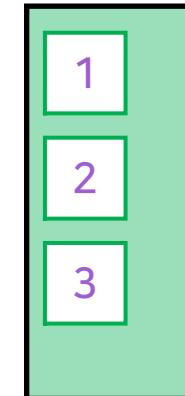
flex-direction

flex-direction : row;

기본값



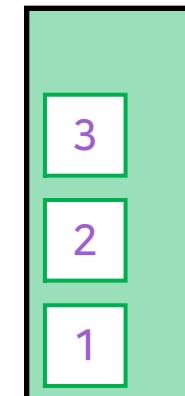
flex-direction : column;



flex-direction : row-reverse;



flex-direction : column-reverse;





Flex Container

flex-wrap

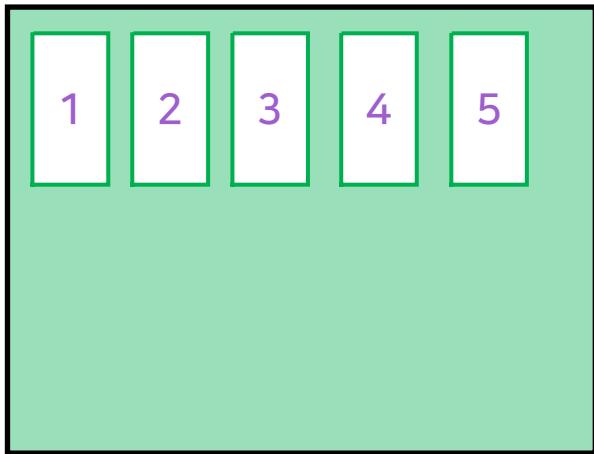
Item 여러 줄을 여부를 지정

속성	설명
nowrap	모든 item 을 여러 줄로 묶지 않음(한 줄에 표시)
wrap	Item 을 여러 줄로 묶음
wrap-reverse	Item 을 wrap의 역 방향으로 여러 줄로 묶음



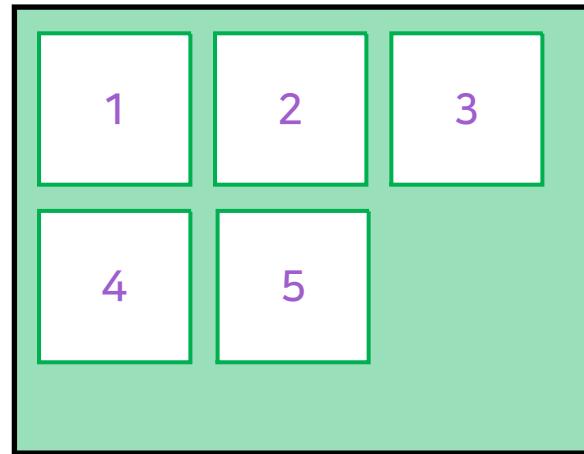
flex-wrap

`flex-wrap : nowrap;`

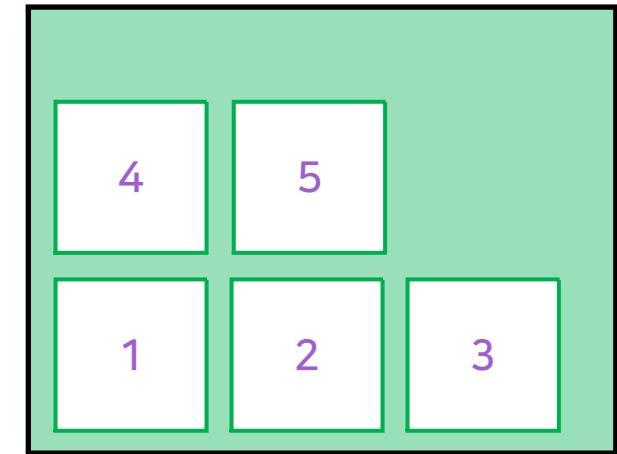


기본값

`flex-wrap : wrap;`



`flex-wrap : wrap-reverse;`





Flex Container

justify-content

주축의 정렬 방법을 지정

속성	설명
flex-start	item 을 시작점(flex-start)으로 정렬
flex-end	item 을 끝점(flex-end)으로 정렬
center	item 을 가운데 정렬
space-between	시작 Item은 시작점에, 마지막 Item은 끝점에 정렬되고 나머지 Items는 사이에 고르게 정렬됨
space-around	Item 시작점과 끝점의 좌우 여백만 작게 지정되고, item 사이의 여백은 균등하게 정렬됨
space-evenly	item 을 균등한 여백을 포함하여 정렬



justify-content

justify-content : flex-start;

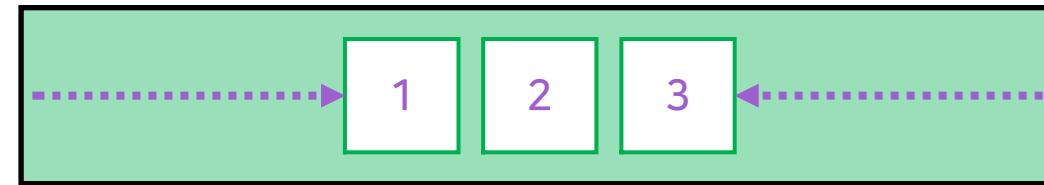


기본값

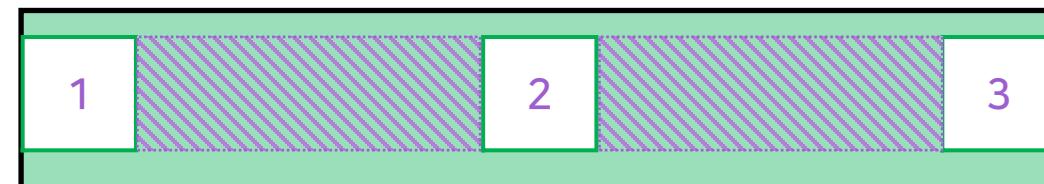
justify-content : flex-end;



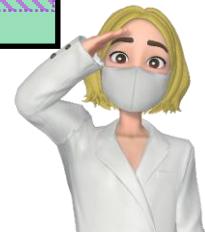
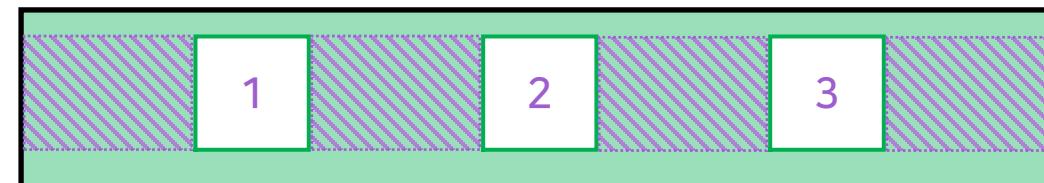
justify-content : center;



justify-content : space-between;



justify-content : space-around;



Flex Container

조건
- 2줄 이상
- 여백

align-content

교차 축의 정렬 방법을 지정

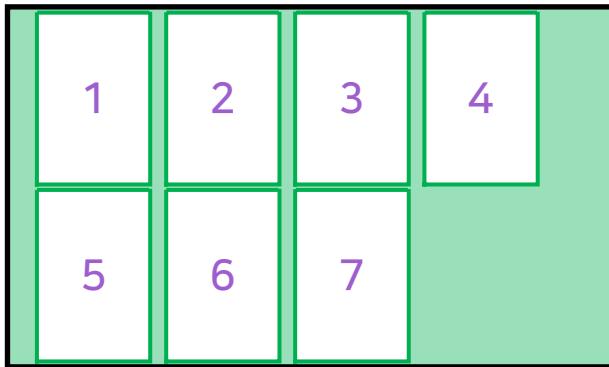
속성	설명
stretch	Container의 교차 축을 채우기 위해 item 을 늘림
flex-start	Item 을 시작점(flex-start)으로 정렬
flex-end	Item 을 끝점(flex-end)으로 정렬
center	Item 을 가운데 정렬
space-between	시작 Item 은 시작점에, 마지막 Item 은 끝점에 정렬되고 나머지 Items 는 사이에 고르게 정렬됨
space-around	Item 시작점과 끝점의 좌우 여백만 작게 지정되고, item 사이의 여백은 균등하게 정렬됨

* stretch 는 교차 축의 크기(width or height) 가 기본값(auto) 이어야 적용됩니다.

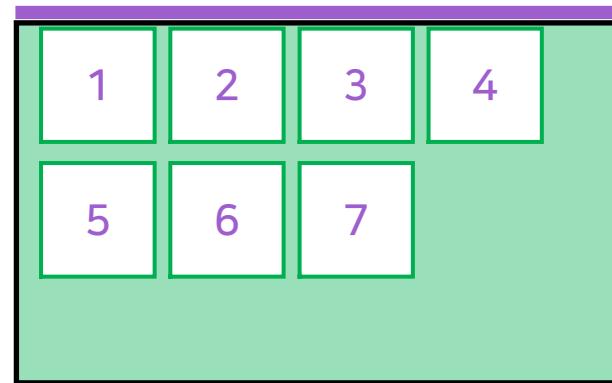


align-content

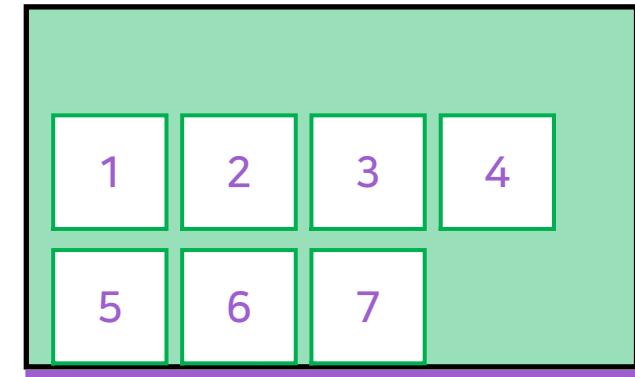
align-content : stretch; 기본값



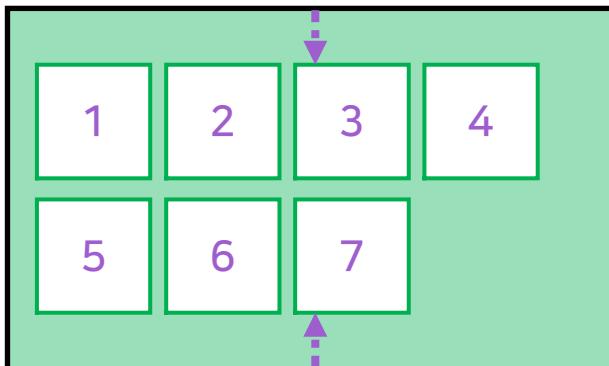
align-content : flex-start;



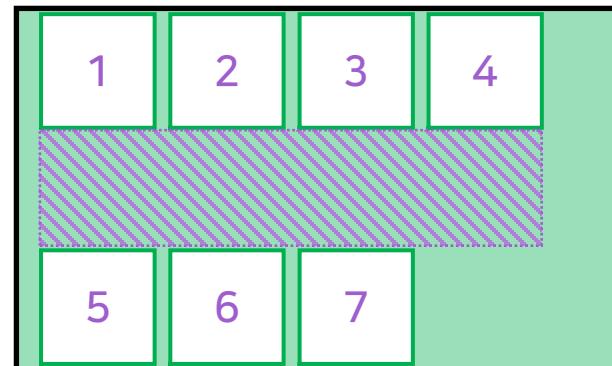
align-content : flex-end;



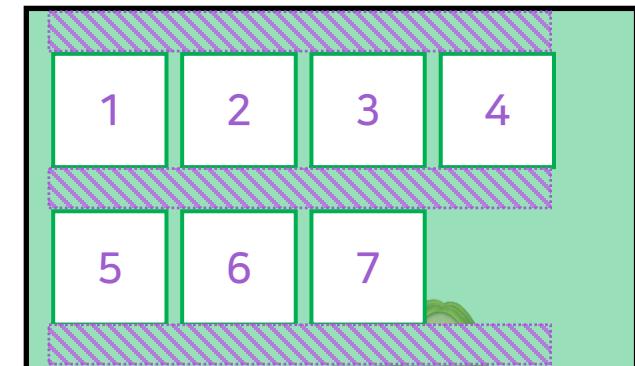
align-content : center;



align-content : space-between;



align-content : space-around;



Flex Container

적용 우선순위

1. align-content
2. align-items

align-items

item 이 한 줄 일 때!

교차 축의 item 들의 정렬 방법 지정

속성	설명
stretch	Container의 교차 축을 채우기 위해 item 을 늘림
flex-start	item 을 각 줄의 시작점(flex-start)으로 정렬
flex-end	item 을 각 줄의 끝점(flex-end)으로 정렬
center	item 을 가운데 정렬
baseline	item 을 문자 기준선에 정렬

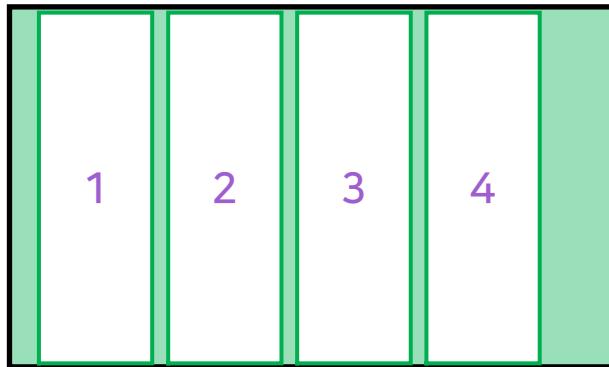
* align-content : stretch(기본값); 이어야 적용됩니다.



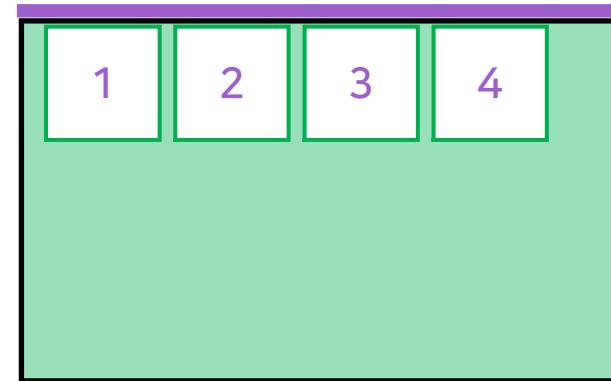
align-items

align-items : **stretch**;

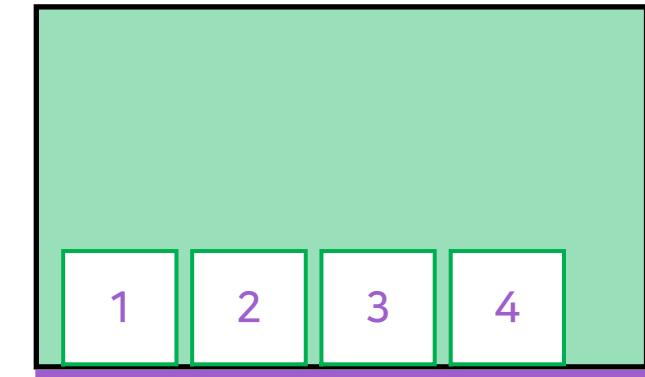
기본값



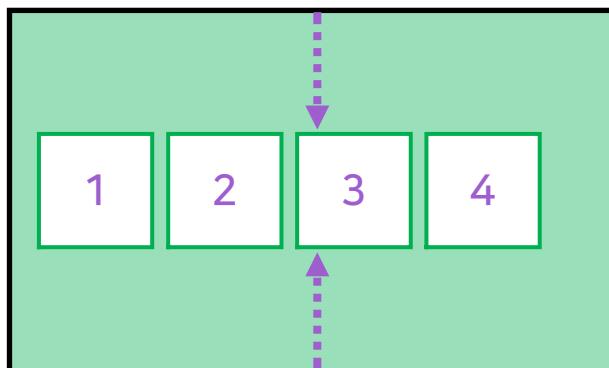
align-items : **flex-start**;



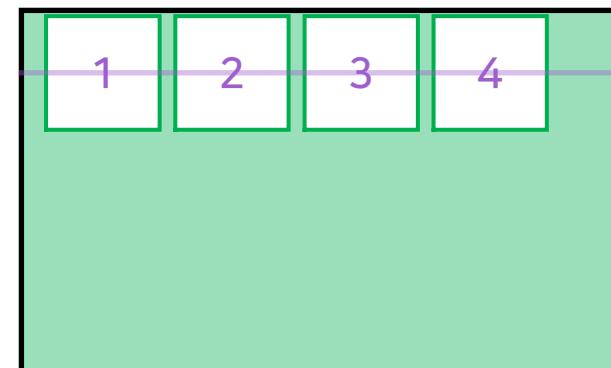
align-items : **flex-end**;



align-items : **center**;



align-items : **baseline**;





Flex item

Item 을 위한 속성

속성	설명
order	Flex Item 의 순서를 설정
flex	flex-grow, flex-shrink, flex-basis의 단축 속성
flex-grow	Flex Item 의 증가 너비 비율을 설정
flex-shrink	Flex Item 의 감소 너비 비율을 설정
flex-basis	Flex Item 의 (공간 배분 전) 기본 너비 설정
align-self	교차 축(cross-axis)에서 Item 의 정렬 방법을 설정





Flex item

order

order : 순서;

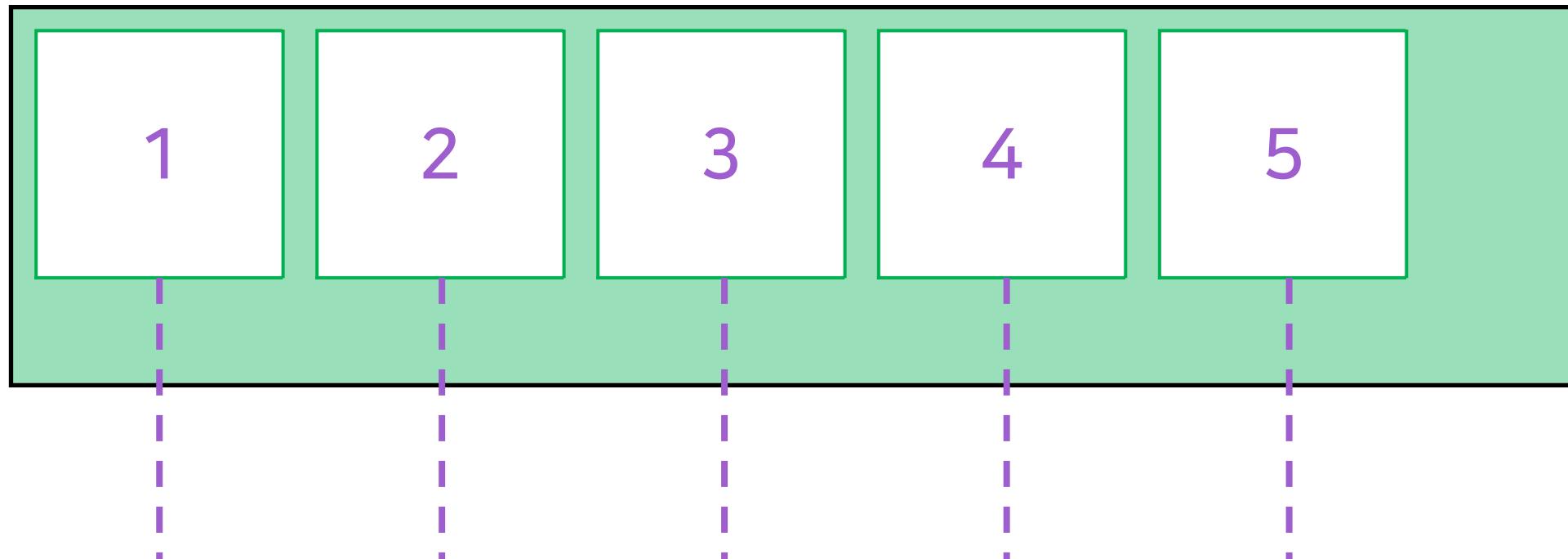
- 기본값 : 0



order

order : 0

기본값



order : -1

order : 1

order : 2

order : 2

order : 50



Flex item

flex

Item 너비를 지정하는 속성

단축 속성

flex : 증가너비 감소너비 기본너비 ;

속성	설명	기본값
flex-grow	Item 의 증가 너비 비율을 설정	0
flex-shrink	Item 의 감소 너비 비율을 설정	1
flex-basis	Item 의 (공간 배분 전) 기본 너비 설정	auto

- flex-grow 이외의 개별 속성 생략 가능
- flex-basis 를 생략하면, auto 가 아니라 0 이 된다.



flex-grow

flex-grow : 1;

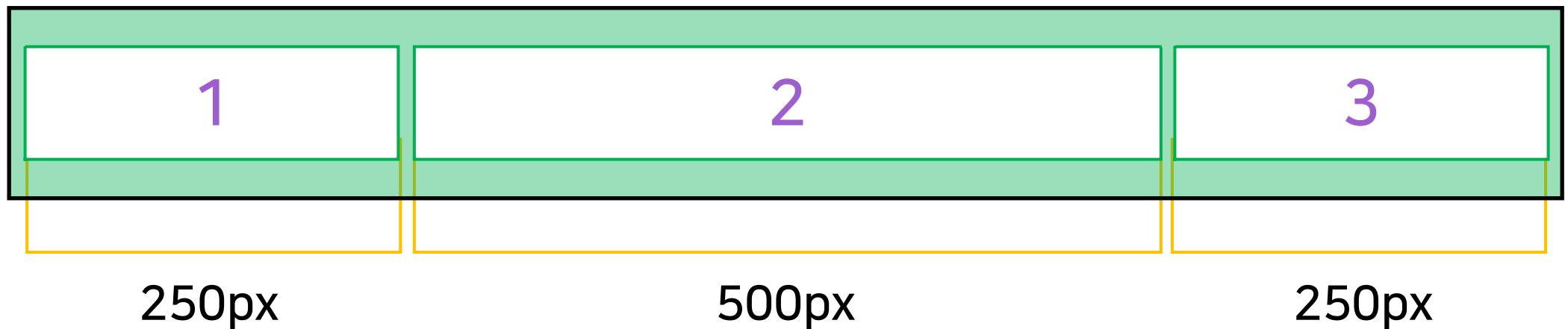
1/4 (25%)

flex-grow : 2;

2/4 (50%)

flex-grow : 1;

1/4 (25%)



flex-shrink

`flex-shrink : 1;`

250px

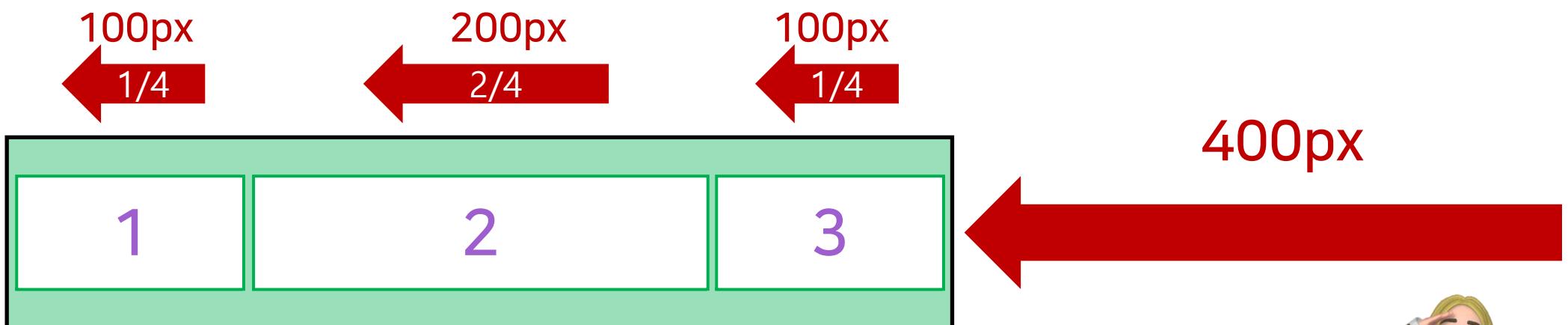


`flex-shrink : 2;`

500px

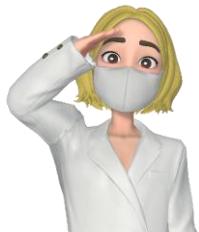
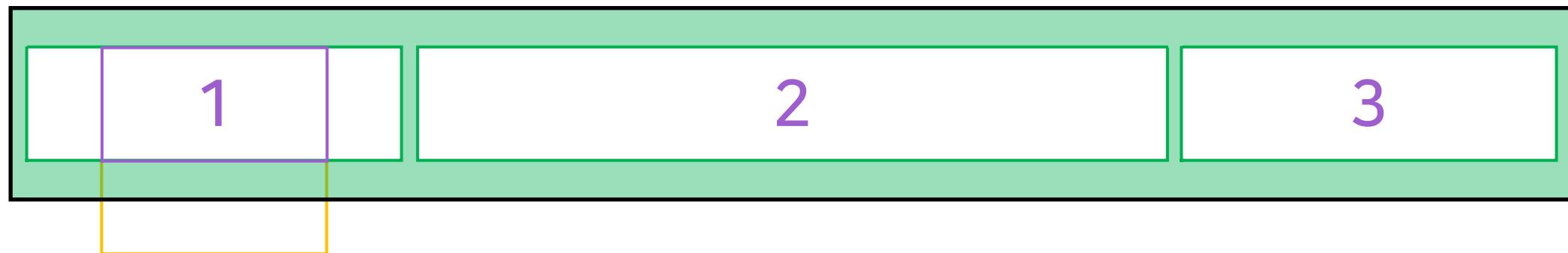
`flex-shrink : 1;`

250px



flex-basis

`flex-basis : 150px;`





Flex item

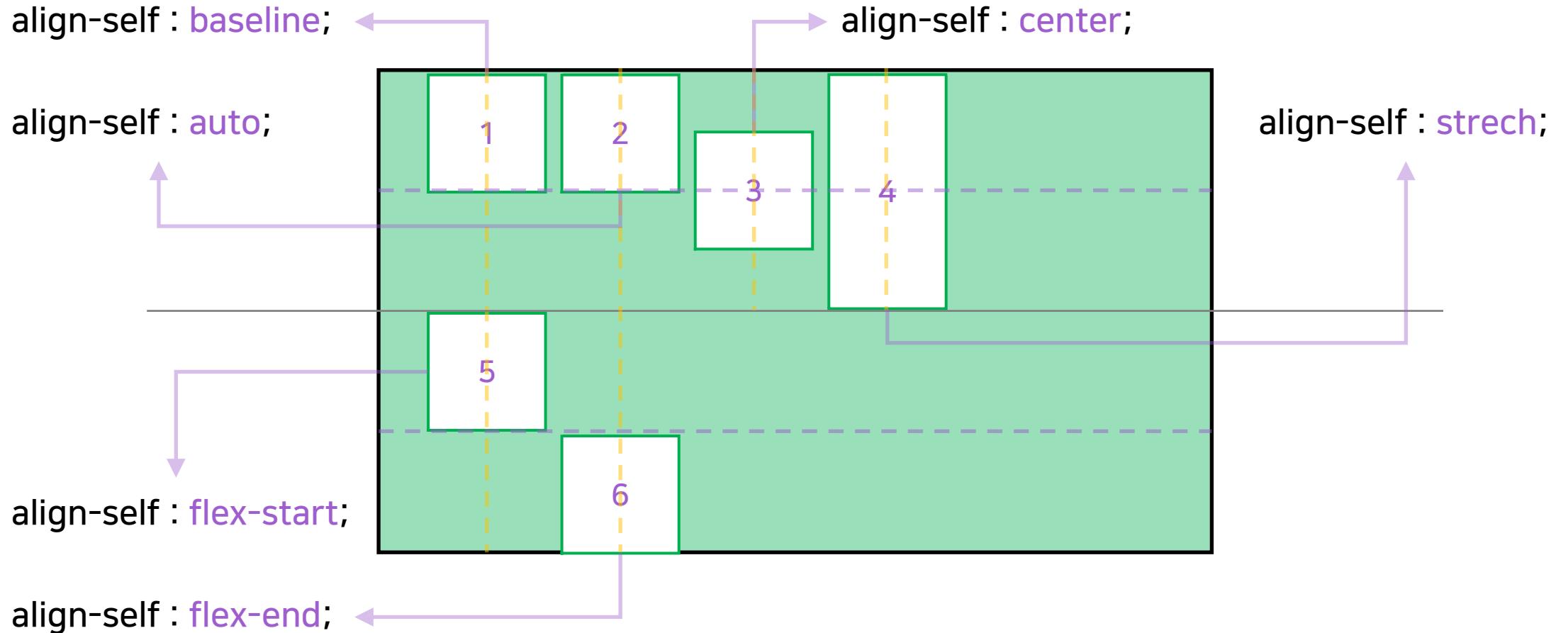
align-self

교차 축의 개별 item 정렬 방법 지정

속성	설명
auto	Container의 align-items 속성을 상속받음
stretch	Container의 교차 축을 채우기 위해 Item을 늘림
flex-start	Item 을 각 줄의 시작점(flex-start)으로 정렬
flex-end	Item 을 각 줄의 끝점(flex-end)으로 정렬
center	Item 을 가운데 정렬
baseline	Item 을 문자 기준선에 정렬



align-self



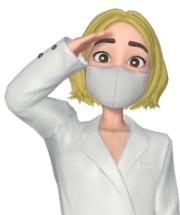
Grid 레이아웃

#22

Grid 레이아웃

2차원(행과 열)의 레이아웃 시스템

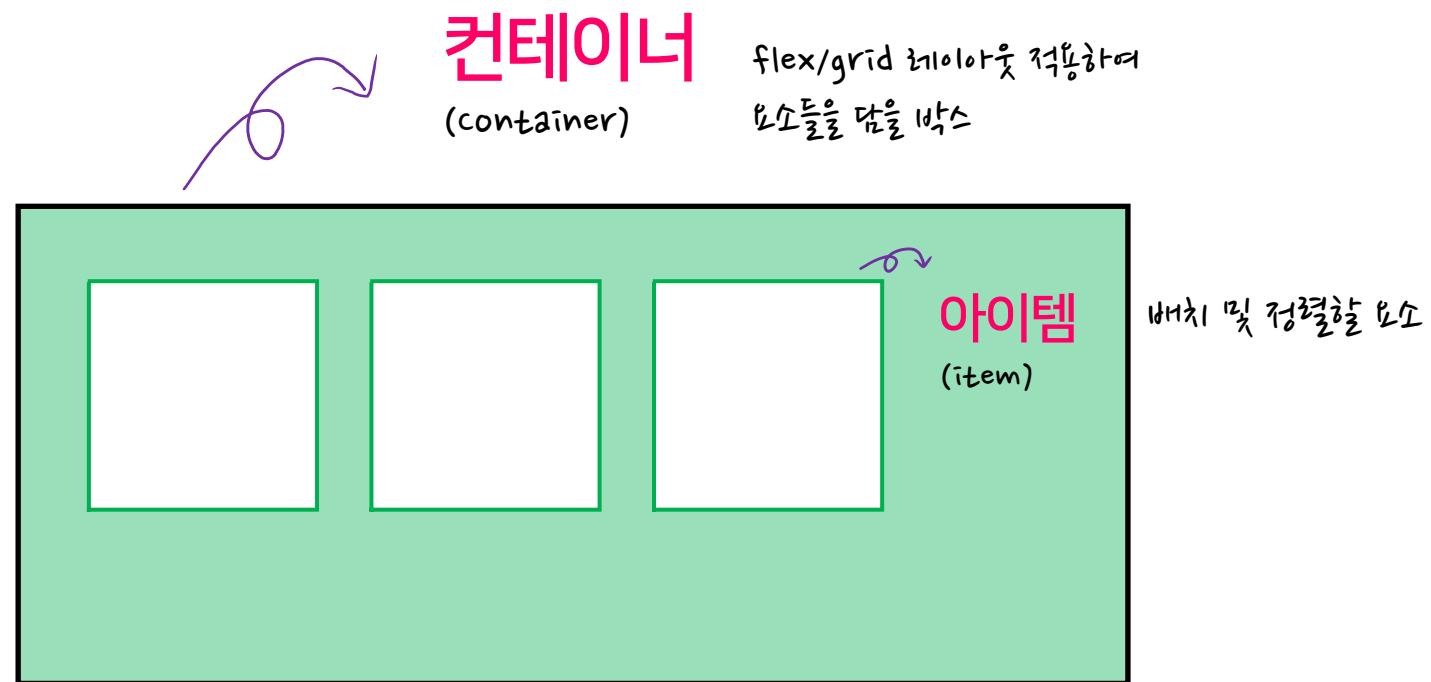
- 세로 열과 가로 행을 기준으로 요소를 정렬 레이아웃



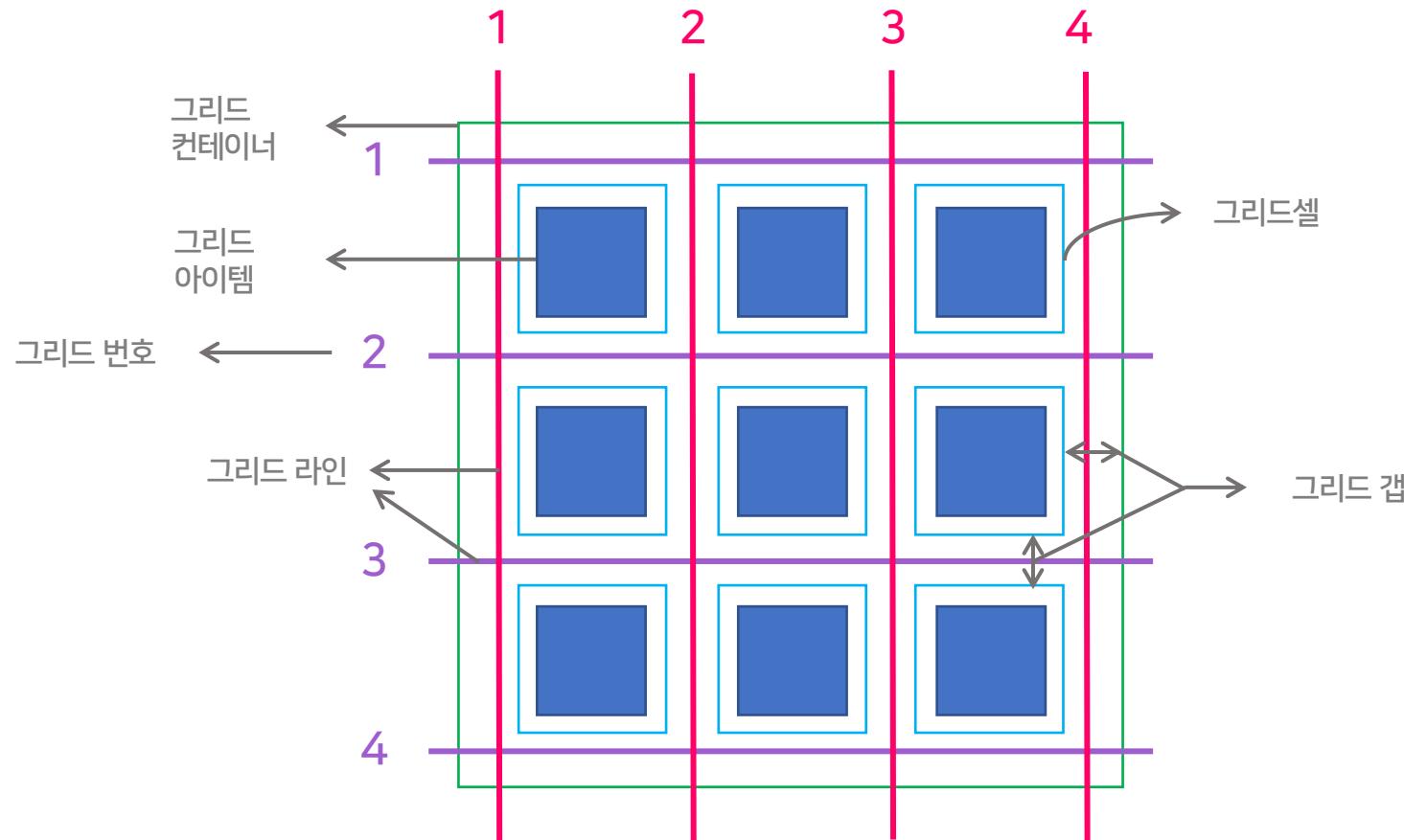
🌿 Container & item



▪ 핵심 요소



Grid 용어





Grid Container

Container 를 위한 속성

속성	설명
display	그리드 컨테이너(Container)를 정의
grid-template	grid-template-xxx의 단축 속성
grid-template-rows	명시적 행(Track)의 크기를 정의
grid-template-columns	명시적 열(Track)의 크기를 정의
grid-template-areas	영역(Area) 이름을 참조해 템플릿 생성
gap(grid-gap)	xxx-gap의 단축 속성
row-gap (grid-row-gap)	행과 행 사이의 간격(Line)을 정의
column-gap (grid-column-gap)	열과 열 사이의 간격(Line)을 정의



Container 를 위한 속성

속성	설명
grid-auto-rows	암시적인 행(Track)의 크기를 정의
grid-auto-columns	암시적인 열(Track)의 크기를 정의
grid-auto-flow	자동 배치 알고리즘 방식을 정의
grid	grid-template-xxx과 grid-auto-xxx의 단축 속성
place-content	align-content와 justify-content의 단축 속성
align-content	그리드 콘텐츠(Grid Contents)를 수직(열 축) 정렬
justify-content	그리드 콘텐츠를 수평(행 축) 정렬
place-items	align-items와 justify-items의 단축 속성
align-items	그리드 아이템(Items)들을 수직(열 축) 정렬
justify-items	그리드 아이템들을 수평(행 축) 정렬





fr

fractional unit (단위 분수)

- 분자가 1이고 분모가 양의 정수인 분수
- 남은 공간을 나누어 갖는 비율

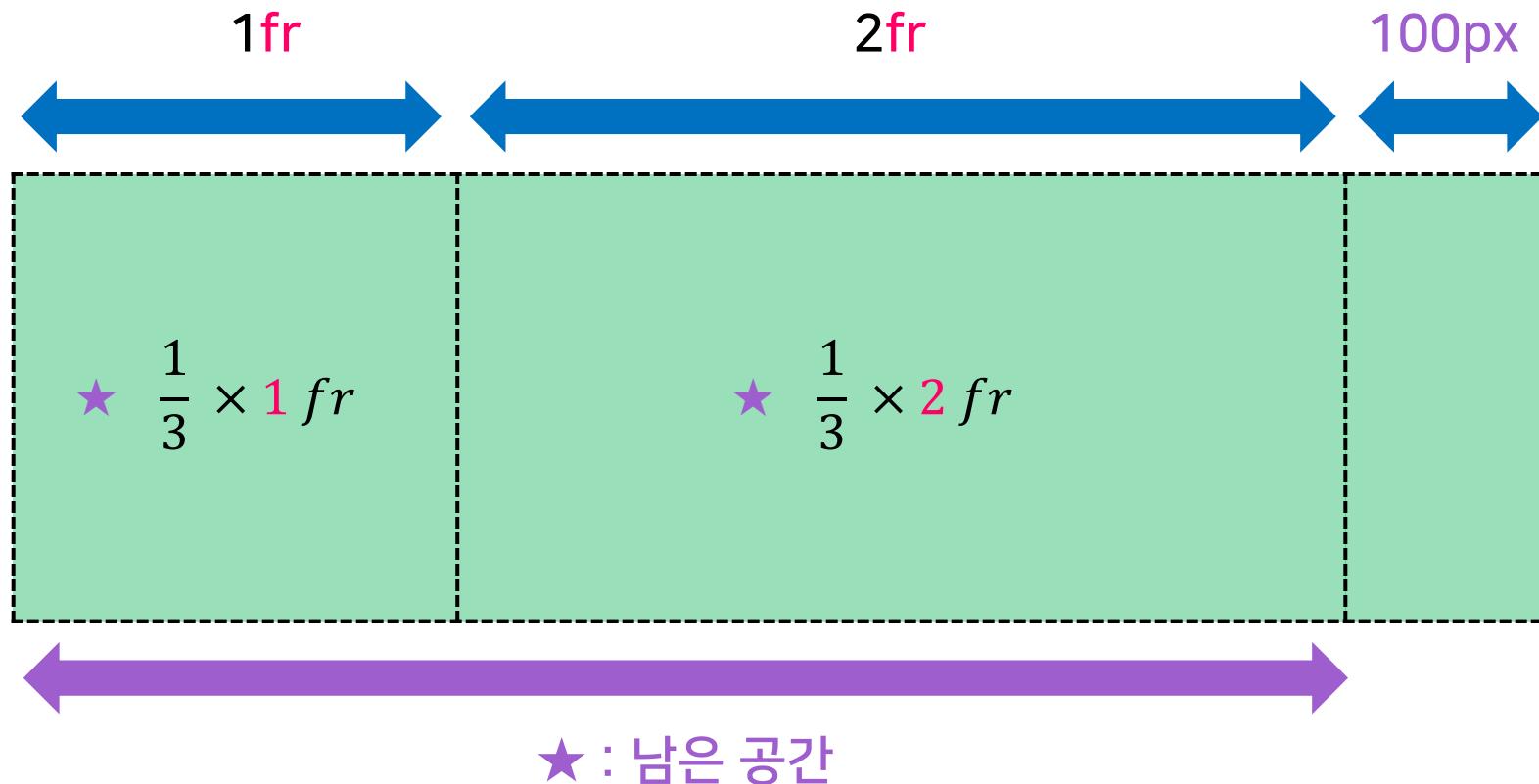
* Fraction
: 분수



fr

남은 공간을 나누어 갖는 비율

`grid-template-columns: 1fr 2fr 100px;`



Grid Container

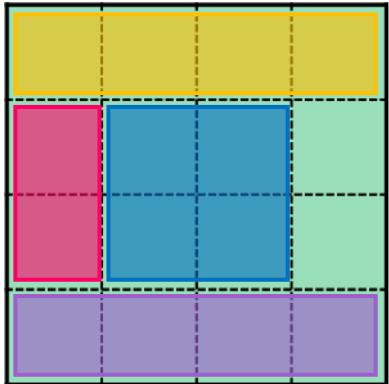
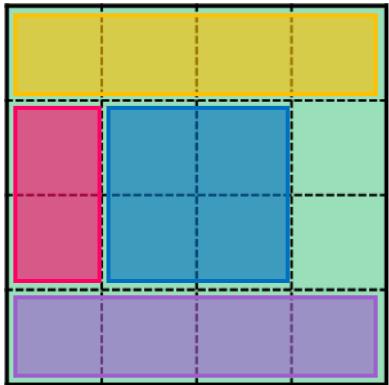
display

속성값	설명
grid	Block 특성의 Grid Container를 정의
inline-grid	Inline 특성의 Grid Container를 정의

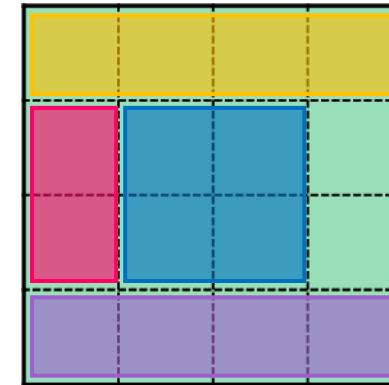
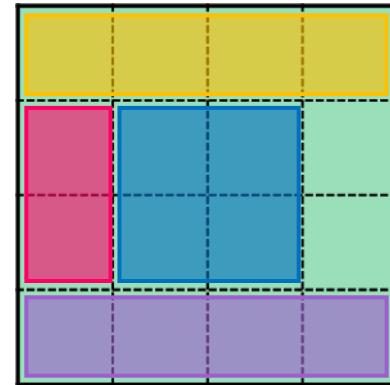


display

`display : grid;`



`display : inline-grid;`



Grid Container

grid-template

단축 속성

- grid-template-rows
grid-template-columns 의 단축 속성





Grid Container

grid-template-rows

- 라인의 이름과 행의 크기를 지정하는 속성
- fr (fraction) 단위 사용가능
- repeat() 함수 사용가능

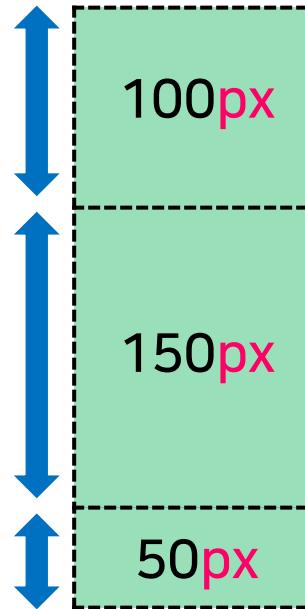
* 라인의 이름은 행/열 개수대로 자동으로 지정된다. (선택적)



grid-template-rows

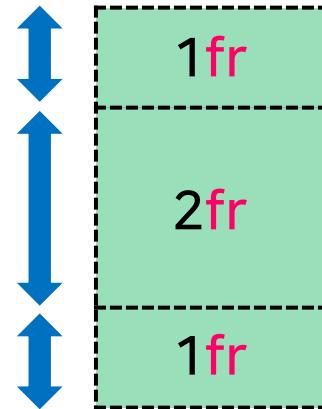
grid-template-rows

: 100px 150px 50px;



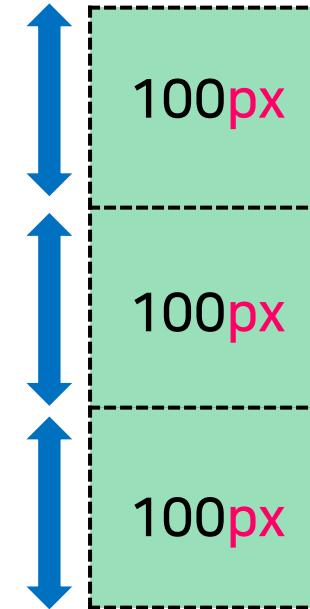
grid-template-rows

: 1fr 2fr 1fr;



grid-template-rows

: repeat(3, 100px)



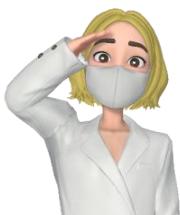


Grid Container

grid-template-columns

- 라인의 이름과 **열의 크기**를 지정하는 속성
- fr (fraction) 단위 사용가능
- repeat() 함수 사용가능

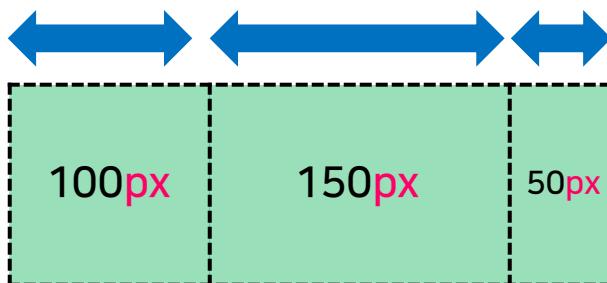
* **라인의 이름은 행/열 개수대로 자동으로 지정된다.** (선택적)



grid-template-columns

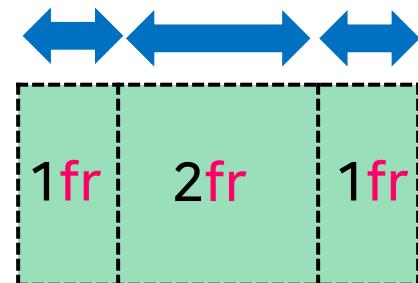
grid-template-columns

: 100px 150px 50px;



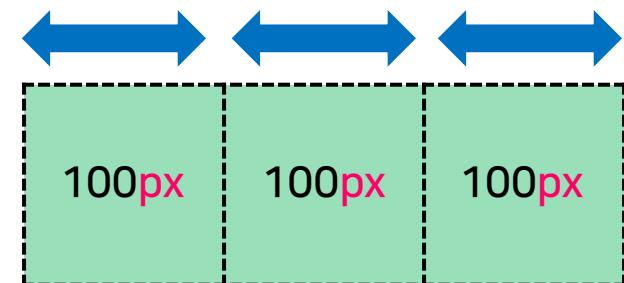
grid-template-columns

: 1fr 2fr 1fr;



grid-template-columns

: repeat(3, 100px)





Grid Container

grid-template-areas

- Grid 레이아웃의 영역을 지정하는 속성
- Grid 아이템에서 grid-area 속성을 사용하여, 지정한 항목의 이름을 참조하여 영역을 지정한다.

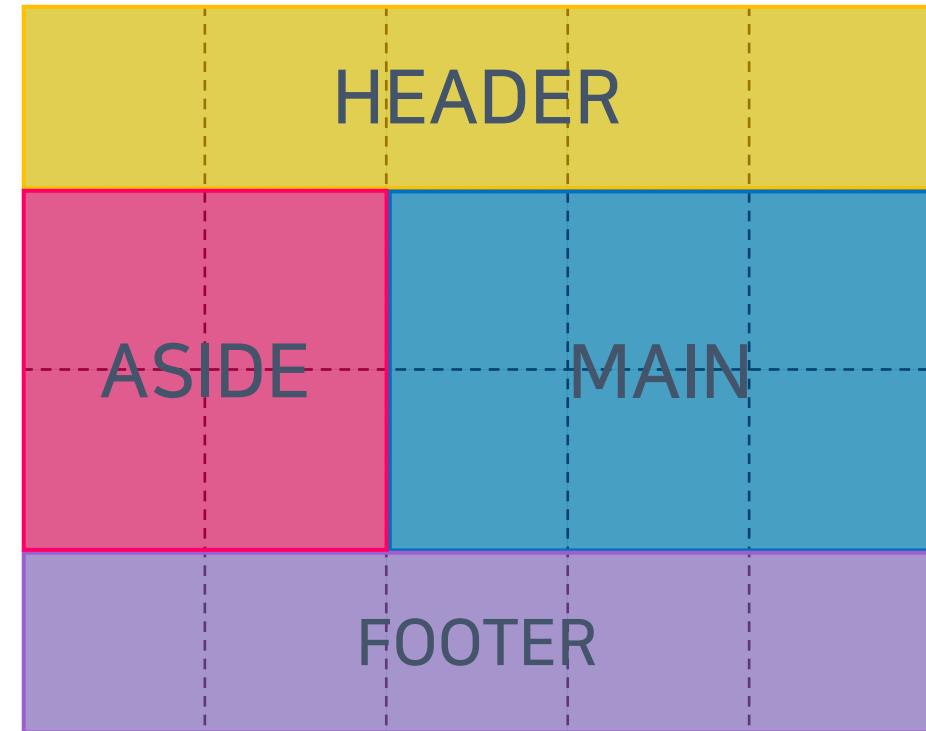
* . 기호를 사용하여 빈 영역을 지정할 수 있다.



grid-template-areas

Grid 레이아웃의 영역을 지정하는 속성

```
●●●  
1 header { grid-area: header; }  
2 aside { grid-area: aside; }  
3 main { grid-area: main; }  
4 footer { grid-area: footer; }  
5  
6  
7 .grid-container {  
8   display: grid;  
9   grid-template-areas:  
10    'header header header header header'  
11    'aside aside main main main'  
12    'aside aside main main main'  
13    'footer footer footer footer footer';  
14 }
```

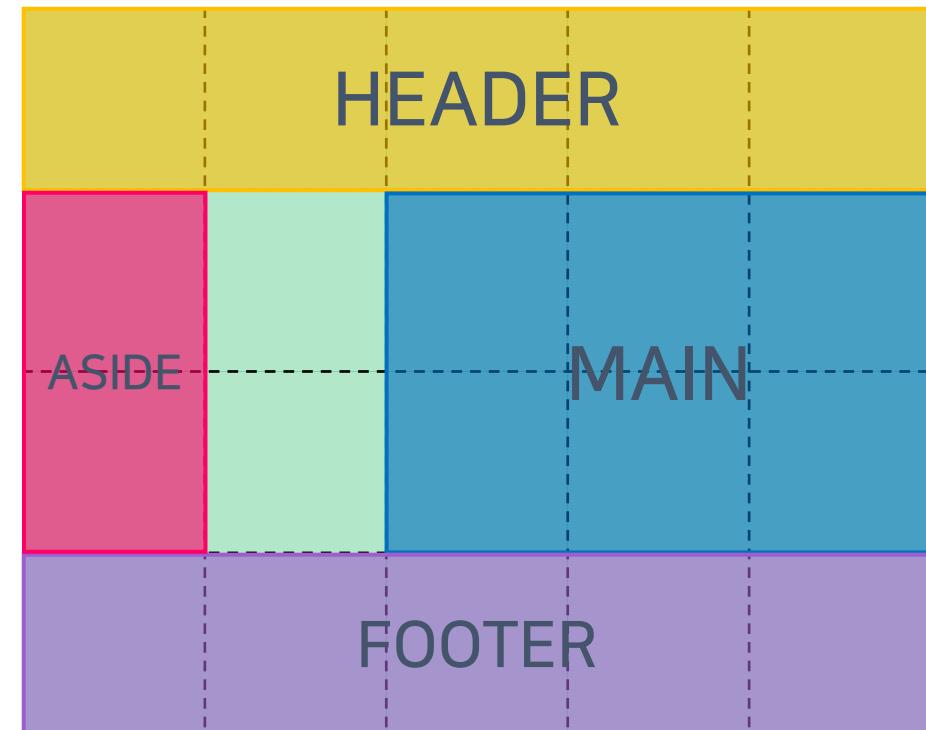


grid-template-areas

Grid 레이아웃의 영역을 지정하는 속성

. 기호를 사용하여 빈 영역을
지정할 수 있다.

```
1 header { grid-area: header; }
2 aside { grid-area: aside; }
3 main { grid-area: main; }
4 footer { grid-area: footer; }
5
6
7 .grid-container {
8   display: grid;
9   grid-template-areas:
10    'header header header header header'
11    'aside . main main main'
12    'aside . main main main'
13    'footer footer footer footer footer';
14 }
```



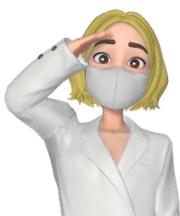
Grid Container

grid-template

단축 속성

- grid-template-rows
grid-template-columns
grid-template-areas 속성을 지정하는 단축 속성

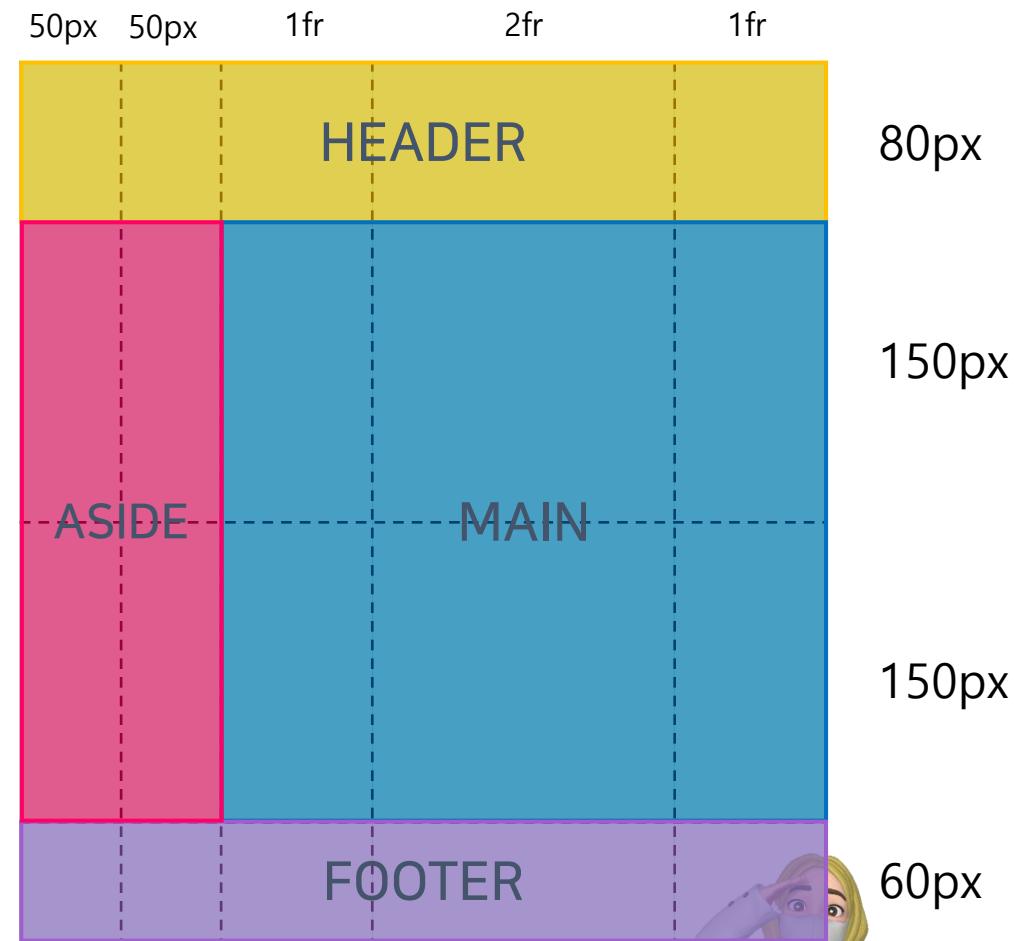
* . 기호를 사용하여 빈 영역을 지정할 수 있다.



grid-template

grid-template-XXX 단축 속성

```
1 header { grid-area: header; }
2 aside { grid-area: aside; }
3 main { grid-area: main; }
4 footer { grid-area: footer; }
5
6
7 .grid-container {
8   display: grid;
9   grid-template:
10    'header header header header header' 80px
11    'aside aside main main main' 150px
12    'aside aside main main main' 150px
13    'footer footer footer footer footer' 60px
14   / 50px 50px 1fr 2fr 1fr;
15 }
```



Grid Container

grid-row-gap

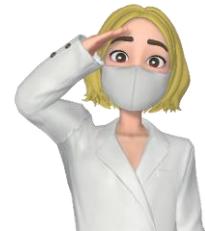
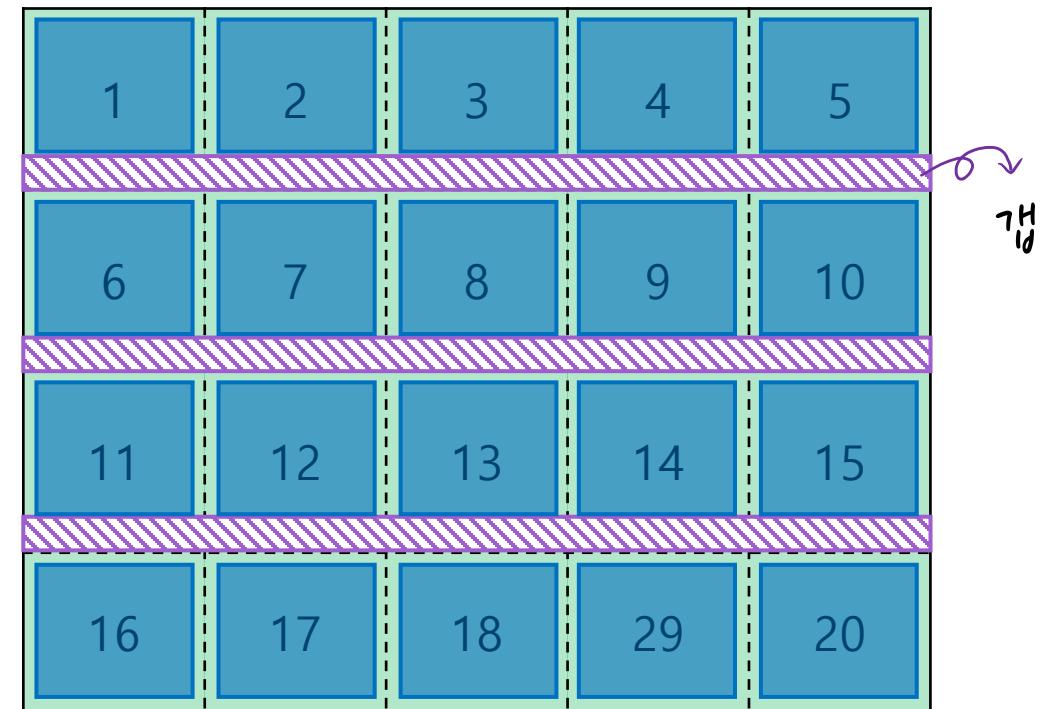
- Grid 레이아웃에서 행 사이의 간격을 지정하는 속성
- 기본값 : 0



grid-row-gap

그리드 행 사이의 간격을 지정하는 속성

```
1 .grid-container {  
2   display: grid;  
3   grid-row-gap : 50px;  
4 }
```





Grid Container

grid-column-gap

- Grid 레이아웃에서 열 사이의 간격을 지정하는 속성
- 기본값 : 0

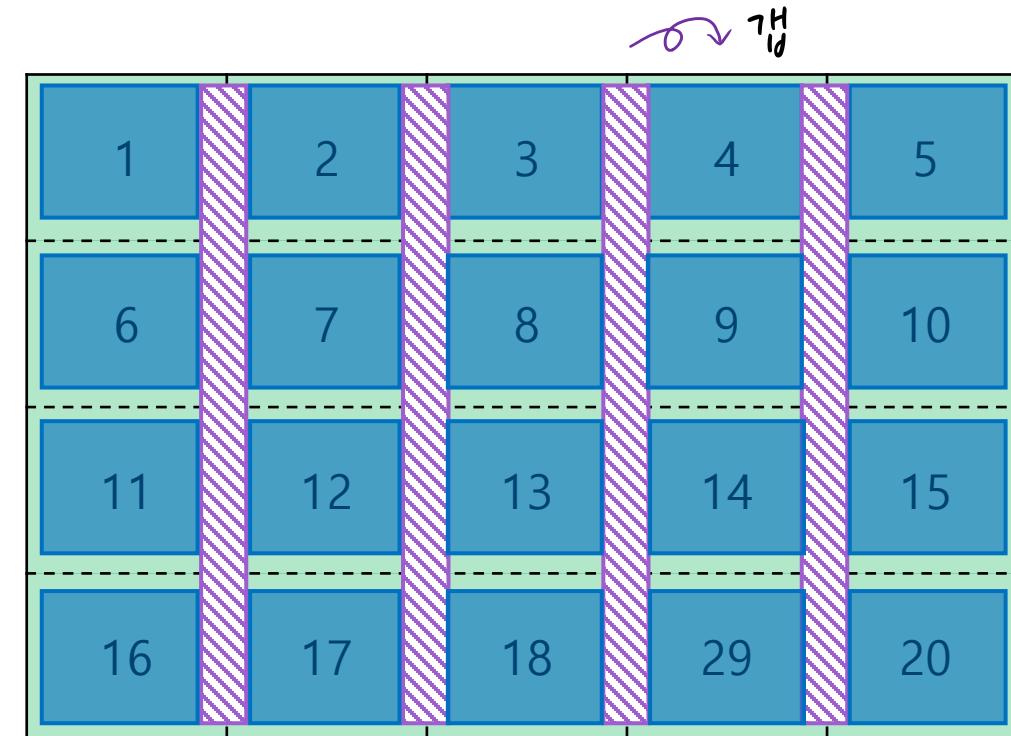


grid-column-gap

그리드 열 사이의 간격을 지정하는 속성



```
1 .grid-container {  
2   display: grid;  
3   grid-column-gap : 50px;  
4 }
```



Grid Container

grid-auto-rows

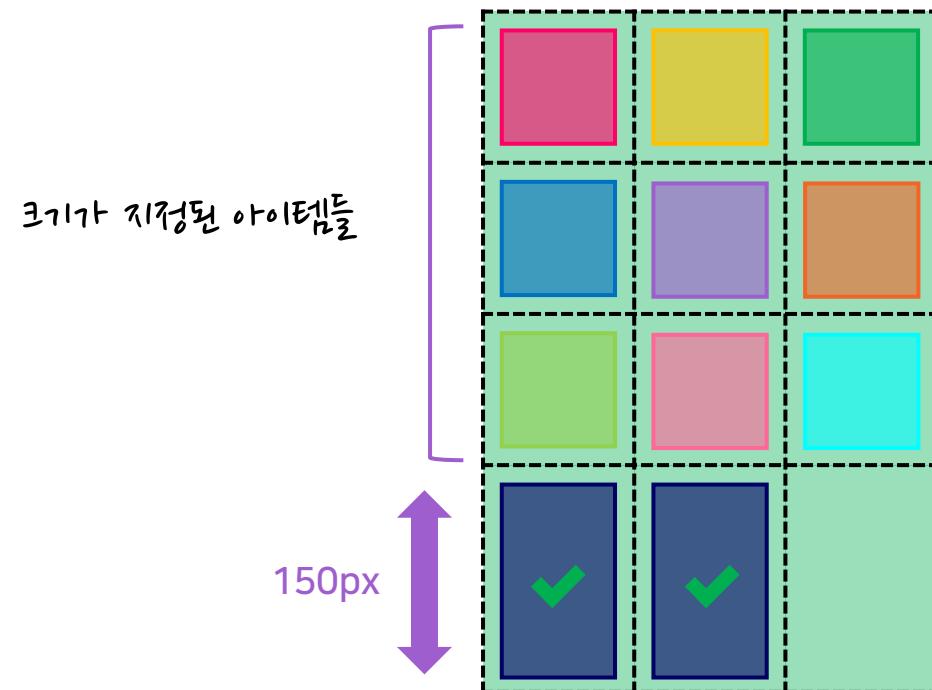
- 크기가 설정되지 않은 행의 크기를 지정하는 속성



grid-auto-rows

크기가 설정되지 않은 행의 크기를 지정하는 속성

grid-auto-rows : 150px;



Grid Container

grid-auto-columns

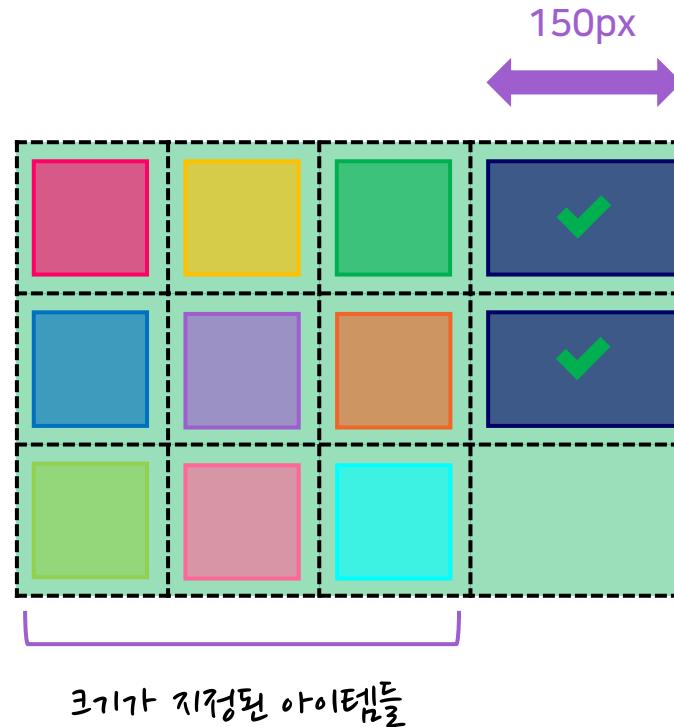
- 크기가 설정되지 않은 열의 크기를 지정하는 속성



grid-auto-columns

크기가 설정되지 않은 열의 크기를 지정하는 속성

grid-auto-columns : 150px;





Grid Container

grid-auto-flow

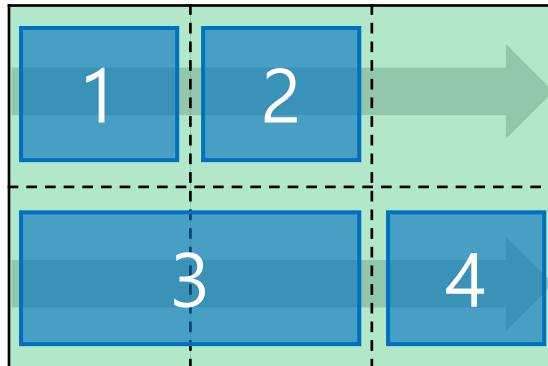
- 크기나 영역을 지정하지 않은 아이템이 그리드에 삽입되는 방식을 지정하는 속성

속성값	의미
row	각 행 축을 따라 차례로 배치 ($\rightarrow \downarrow$)
column	각 열 축을 따라 차례로 배치 ($\downarrow \rightarrow$)
row dense	행 축을 따라 배치, 앞 선 행의 빈 영역을 채움
column dense	열 축을 따라 배치, 앞 선 열의 빈 영역을 채움

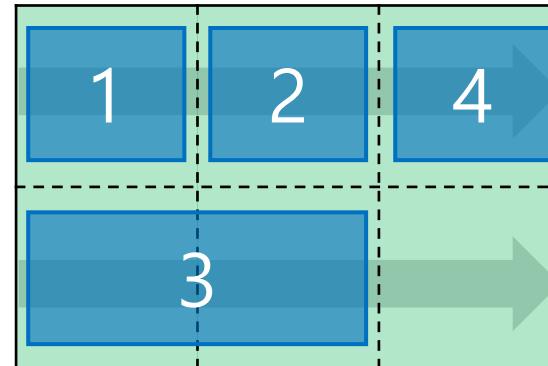


grid-auto-flow

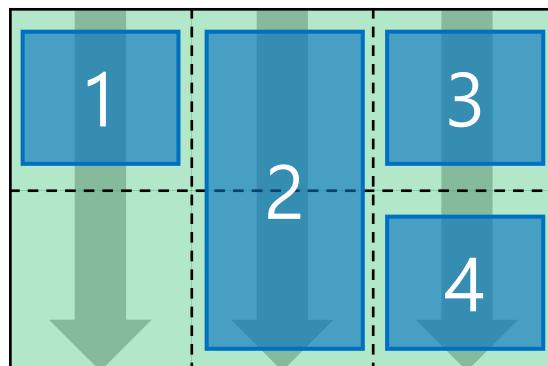
grid-auto-flow : **row**;



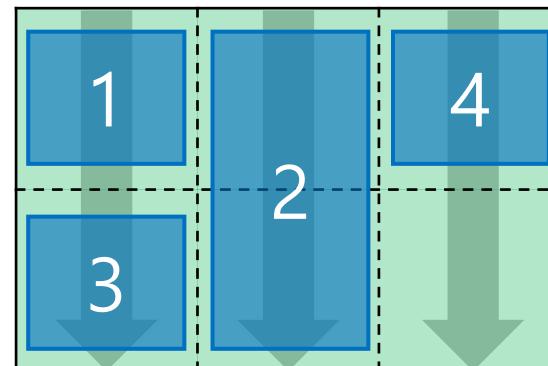
grid-auto-flow : **row dense**;



grid-auto-flow : **column**;



grid-auto-flow : **column dense**;





Grid Container

grid

- grid-template-xxx과 grid-auto-xxx의 단축 속성입니다.

```
1 .grid-container {  
2     grid: <grid-template>;  
3  
4     grid: <grid-template-rows> / <grid-auto-flow> <grid-auto-columns>;  
5  
6     grid: <grid-auto-flow> <grid-auto-rows> / <grid-template-columns>;  
7 }
```





Grid Container

align-content

- Grid 컨텐츠의 수직 정렬 방식을 지정하는 속성

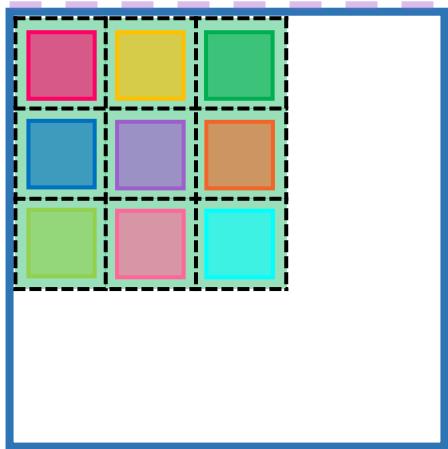
속성값	의미
normal	stretch와 같습니다.
start	시작점(위쪽) 정렬
center	수직 가운데 정렬
end	끝점(아래쪽) 정렬
space-around	각 행 위아래에 여백을 고르게 정렬
space-between	첫 행은 시작점에, 끝 행은 끝점에 정렬되고 나머지 여백으로 고르게 정렬
space-evenly	모든 여백을 고르게 정렬
stretch	열 축을 채우기 위해 그리드 컨텐츠를 늘림



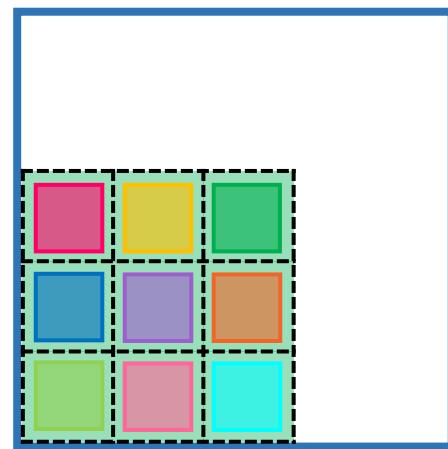
align-content

Grid 컨텐츠의 수직 정렬 방식을 지정하는 속성

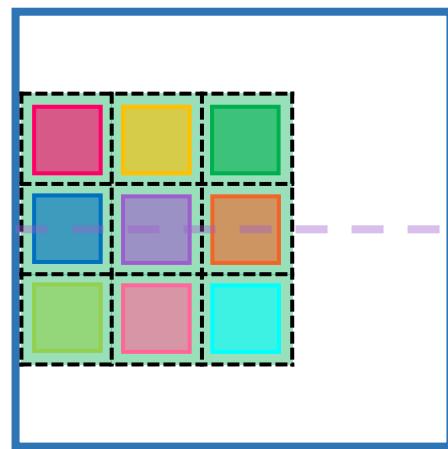
align-content : start;



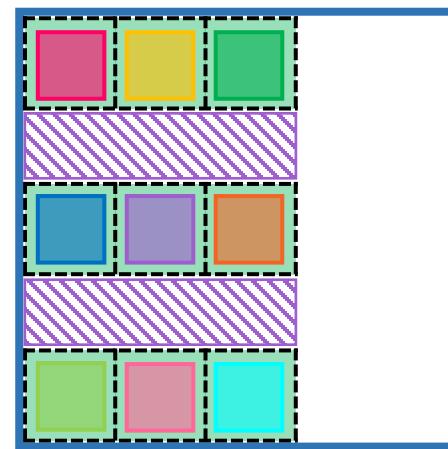
align-content : end;



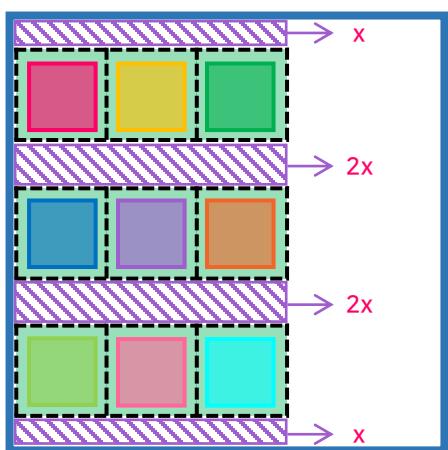
align-content : center;



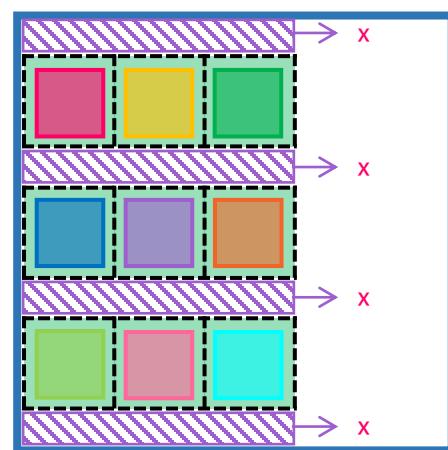
align-content : space-between;



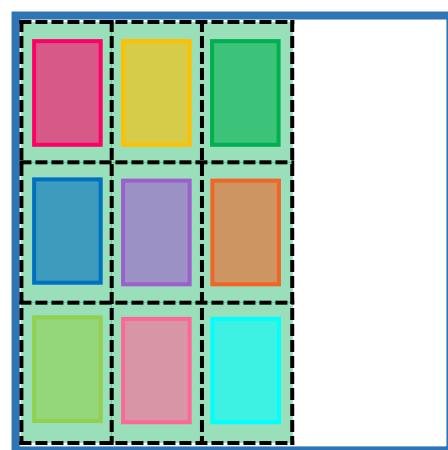
align-content : space-around;



align-content : space-evenly;



align-content : stretch;





Grid Container

justify-content

- Grid 컨텐츠의 수평 정렬 방식을 지정하는 속성

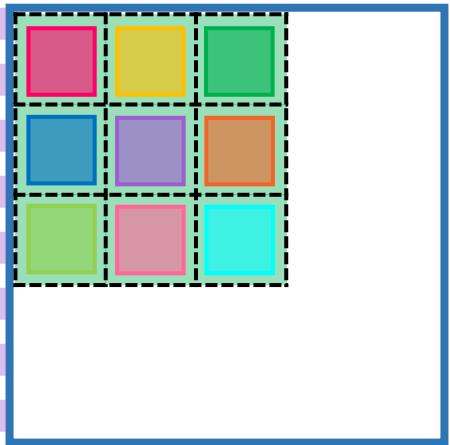
속성값	의미
normal	stretch와 같습니다.
start	시작점(왼쪽) 정렬
center	수평 가운데 정렬
end	끝점(오른쪽) 정렬
space-around	각 열 좌우에 여백을 고르게 정렬
space-between	첫 열은 시작점에, 끝 열은 끝점에 정렬되고 나머지 여백으로 고르게 정렬
space-evenly	모든 여백을 고르게 정렬
stretch	행 축을 채우기 위해 그리드 컨텐츠를 늘림



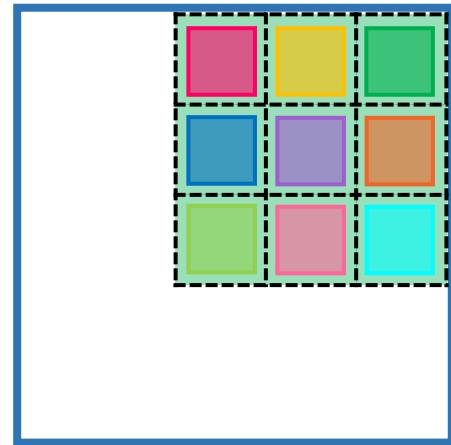
justify-content

Grid 컨텐츠의 수평 정렬 방식을 지정하는 속성

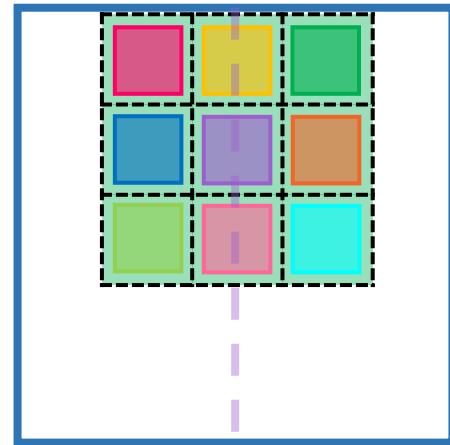
justify-content : start;



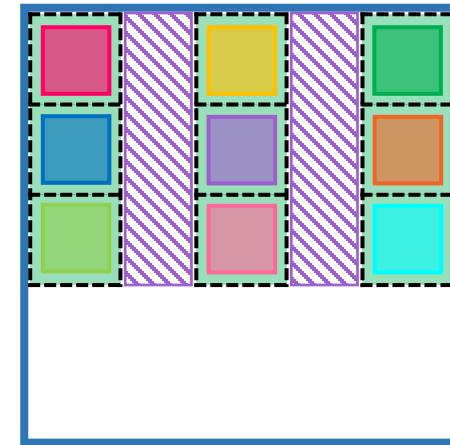
justify-content : end;



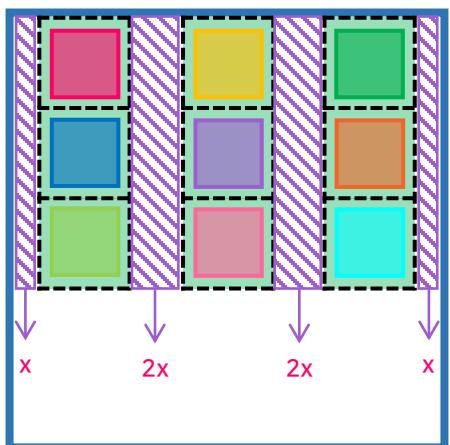
justify-content : center;



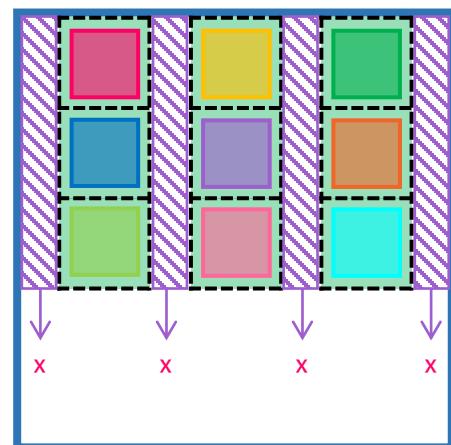
justify-content : space-between;



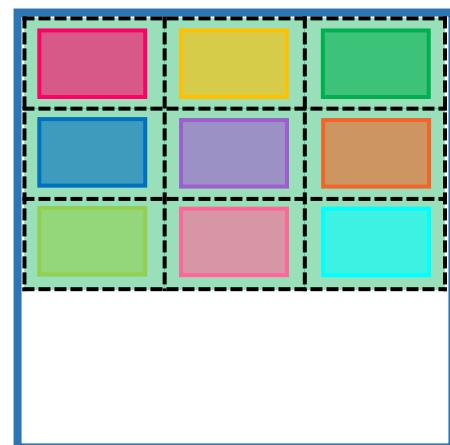
justify-content : space-around;



justify-content : space-evenly;



justify-content : stretch;



Grid Container

place-content

- align-content 와 justify-content 의 단축 속성

```
1 .grid-container {  
2  
3   place-content: <align-content> <justify-content>;  
4  
5 }
```



Grid Container

align-items

- Grid 아이템들의 수직 정렬 방식을 지정하는 속성

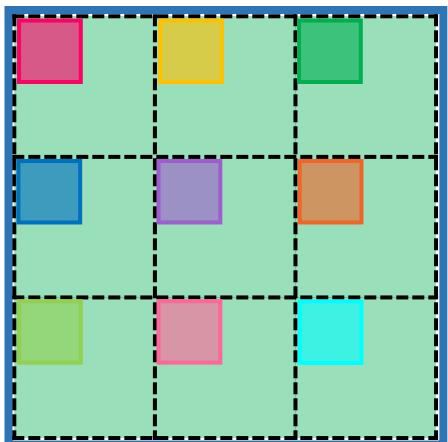
속성값	의미
normal	stretch와 같습니다.
start	시작점(위쪽) 정렬
center	수직 가운데 정렬
end	끝점(아래쪽) 정렬
stretch	열 축을 채우기 위해 그리드 아이템을 늘림
normal	stretch와 같습니다.



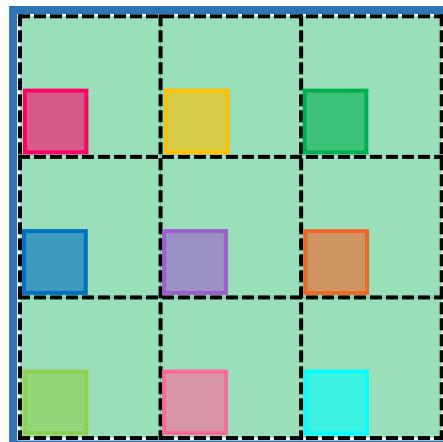
align-items

Grid 아이템들의 수직 정렬 방식을 지정하는 속성

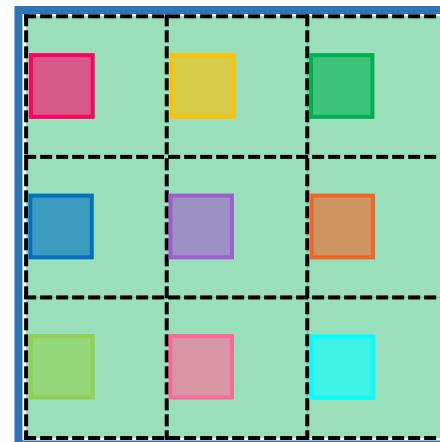
align-items : start;



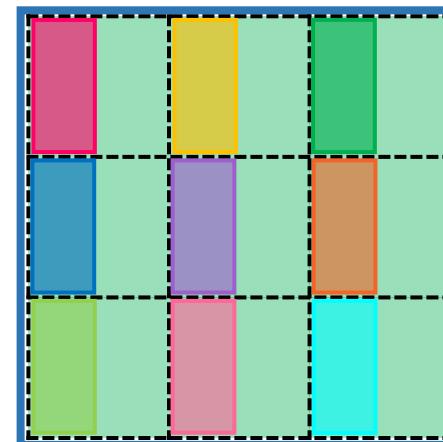
align-items : end;



align-items : center;



align-items : stretch;



Grid Container

justify-items

- Grid 아이템들의 수평 정렬 방식을 지정하는 속성

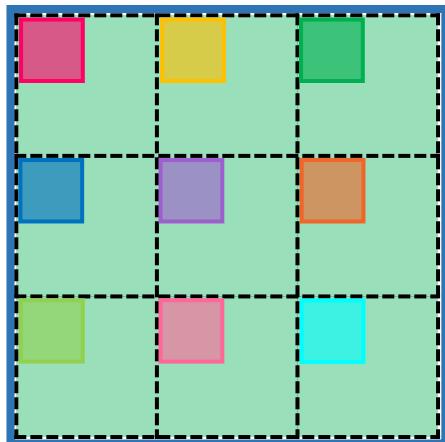
속성값	의미
normal	stretch와 같습니다.
start	시작점(왼쪽) 정렬
center	수평 가운데 정렬
end	끝점(오른쪽) 정렬
stretch	행 축을 채우기 위해 그리드 아이템을 늘림



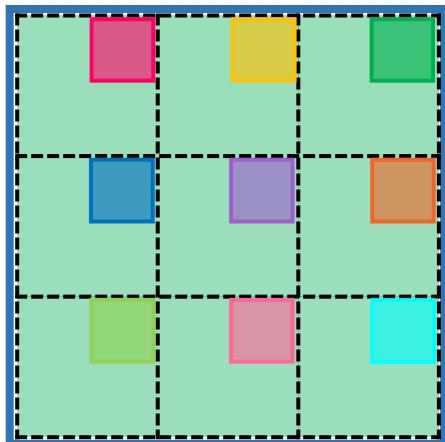
justify-items

Grid 아이템들의 수평 정렬 방식을 지정하는 속성

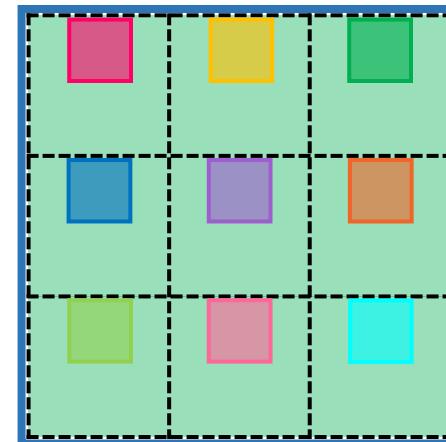
justify-items : start;



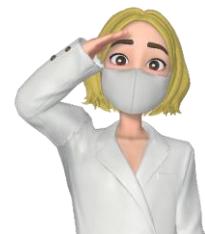
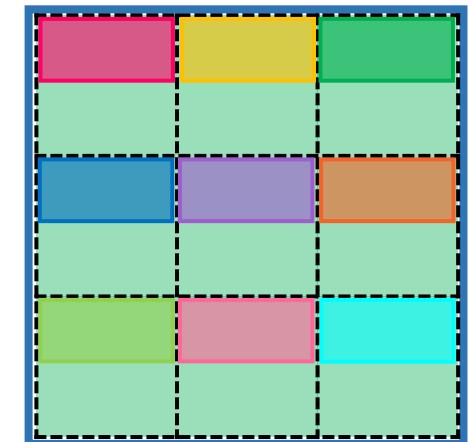
justify-items : end;



justify-items : center;



justify-items : stretch;



Grid Container

place-items

- align-items 와 justify-items 의 단축 속성



```
1 .grid-container {  
2   place-items: <align-items> <justify-items>;  
3 }
```





Grid Item

item 를 위한 속성

속성	설명
grid-row-start	그리드 아이템(Item)의 행 시작 위치 지정
grid-row-end	그리드 아이템의 행 끝 위치 지정
grid-row	grid-row-xxx의 단축 속성(행 시작/끝 위치)
grid-column-start	그리드 아이템의 열 시작 위치 지정
grid-column-end	그리드 아이템의 열 끝 위치 지정
grid-column	grid-column-xxx의 단축 속성(열 시작/끝 위치)



Grid Item

item 을 위한 속성

속성	설명
grid-area	영역(Area) 이름을 설정하거나, grid-row와 grid-column의 단축 속성
align-self	단일 그리드 아이템을 수직(열 축) 정렬
justify-self	단일 그리드 아이템을 수평(행 축) 정렬
place-self	align-self와 justify-self의 단축 속성
order	그리드 아이템의 배치 순서를 지정
z-index	아이템이 쌓이는 순서를 지정하는 속성





Grid Item

grid-row, grid-column

- **grid-row** : grid-row-start과 grid-row-end의 단축 속성
 - **grid-column** : grid-column-start과 grid-column-end의 단축 속성
-
- **grid-row-start** : 그리드 **수직** 라인의 **시작** 위치를 지정하는 속성
 - **grid-row-end** : 그리드 **수직** 라인의 **끝** 위치를 지정하는 속성
 - **grid-column-start** : 그리드 **수평** 라인의 **시작** 위치를 지정하는 속성
 - **grid-column-end** : 그리드 **수평** 라인의 **끝** 위치를 지정하는 속성



grid-row

그리드 아이템 수직 라인의 시작/끝 위치를 지정하는 속성

span

: 확장할 라인의 개수를
지정하는 키워드



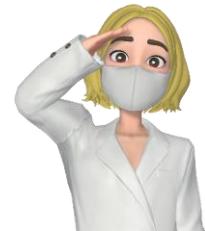
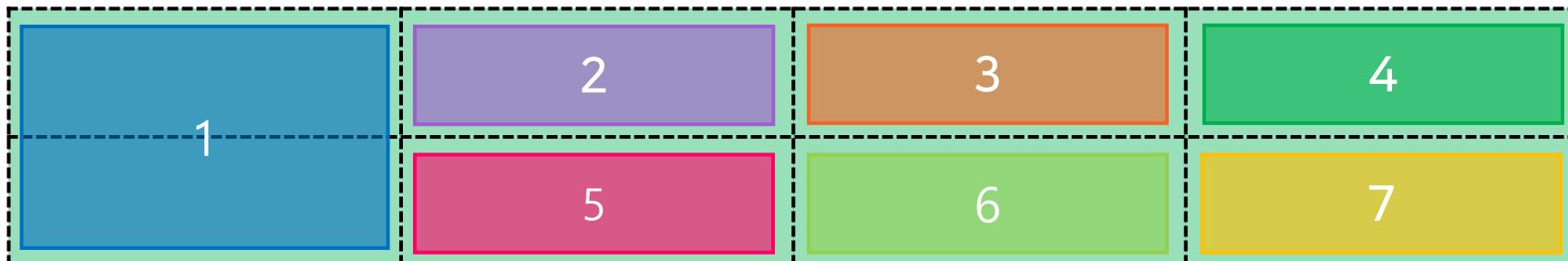
```
1 .grid-container {  
2   display: grid;  
3   ...  
4 }  
5  
6 .item1 {  
7   grid-row-start: 1;  
8   grid-row-end: 3;  
9 }
```



```
1 .grid-container {  
2   display: grid;  
3   ...  
4 }  
5  
6 .item1 {  
7   grid-row: 1 / 3;  
8 }
```



```
1 .grid-container {  
2   display: grid;  
3   ...  
4 }  
5  
6 .item1 {  
7   grid-row: 1 / span 2;  
8 }
```



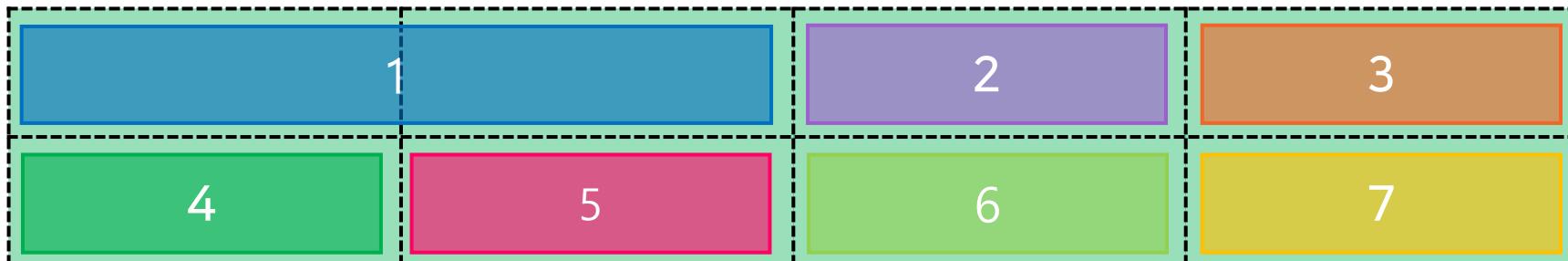
grid-column

그리드 아이템 수평 라인의 시작/끝 위치를 지정하는 속성

```
1 .grid-container {  
2   display: grid;  
3   ...  
4 }  
5  
6 .item1 {  
7   grid-column-start: 1;  
8   grid-column-end: 3;  
9 }
```

```
1 .grid-container {  
2   display: grid;  
3   ...  
4 }  
5  
6 .item1 {  
7   grid-column: 1 / 3;  
8 }
```

```
1 .grid-container {  
2   display: grid;  
3   ...  
4 }  
5  
6 .item1 {  
7   grid-column: 1 / span 2;  
8 }
```



Grid Item

grid-area

- grid-row와 grid-column의 단축 속성

```
● ● ●  
1 .grid-item {  
2   grid-area: <grid-row-start> / <grid-column-start> / <grid-row-end> / <grid-column-end>;  
3 }
```

- grid-template-areas가 참조할 영역 이름을 지정하는 속성

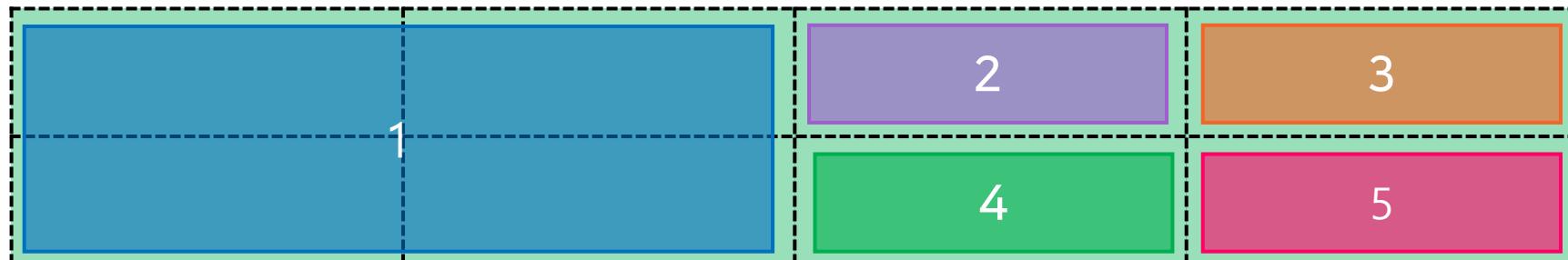
```
● ● ●  
1 .grid-item {  
2   grid-area: 영역이름;  
3 }
```



grid-area

grid-row와 grid-column의 단축 속성

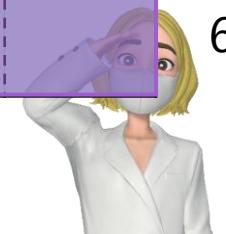
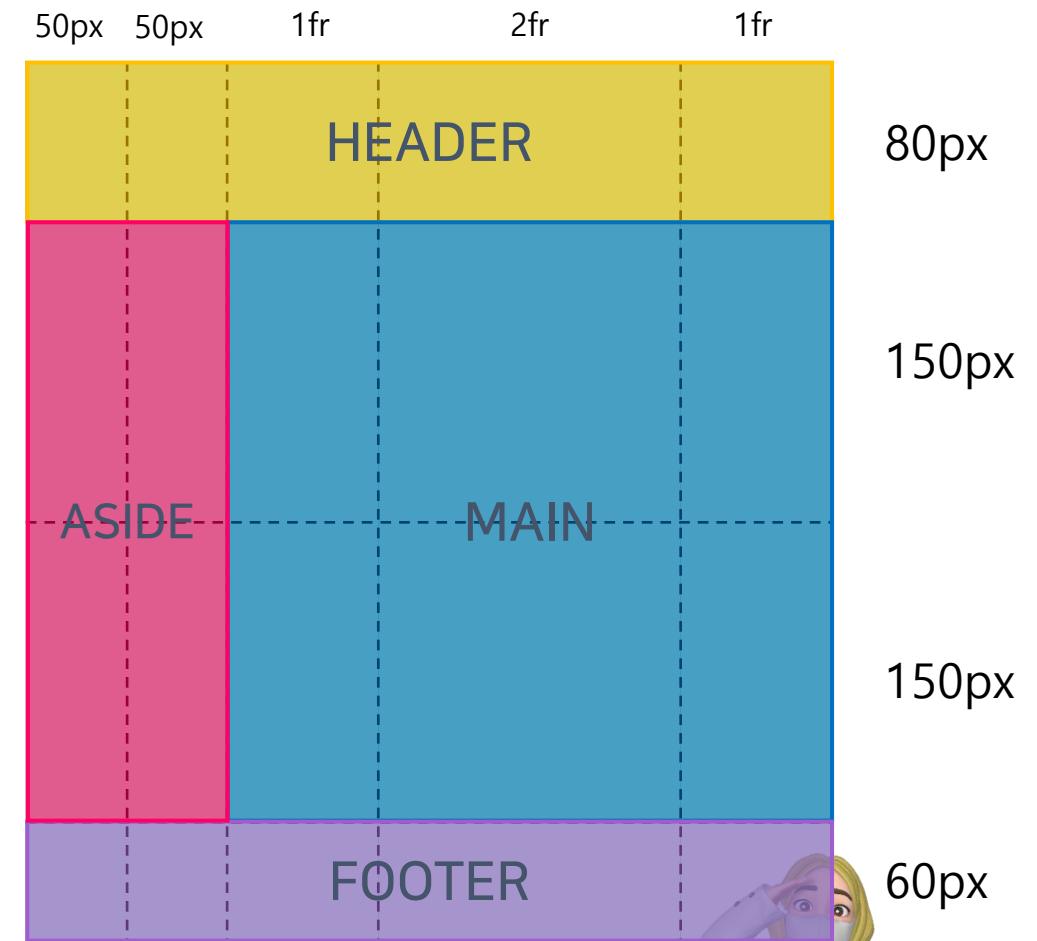
```
1 .item {  
2   grid-row: 1 / 3;  
3   grid-column: 1 / 3;  
4 }  
5 .item {  
6   grid-area: 1 / 1 / 3 / 3;  
7 }
```



grid-area

grid-template-areas가 참조할 영역 이름을 지정하는 속성

```
 1 header { grid-area: header; }
 2 aside { grid-area: aside; }
 3 main { grid-area: main; }
 4 footer { grid-area: footer; }
 5
 6
 7 .grid-container {
 8   display: grid;
 9   grid-template:
10     'header header header header header' 80px
11     'aside aside main main main' 150px
12     'aside aside main main main' 150px
13     'footer footer footer footer footer' 60px
14   / 50px 50px 1fr 2fr 1fr;
15 }
```



Grid Item

align-self

- 단일 그리드 아이템(Item)을 수직(열 축) 정렬하는 속성

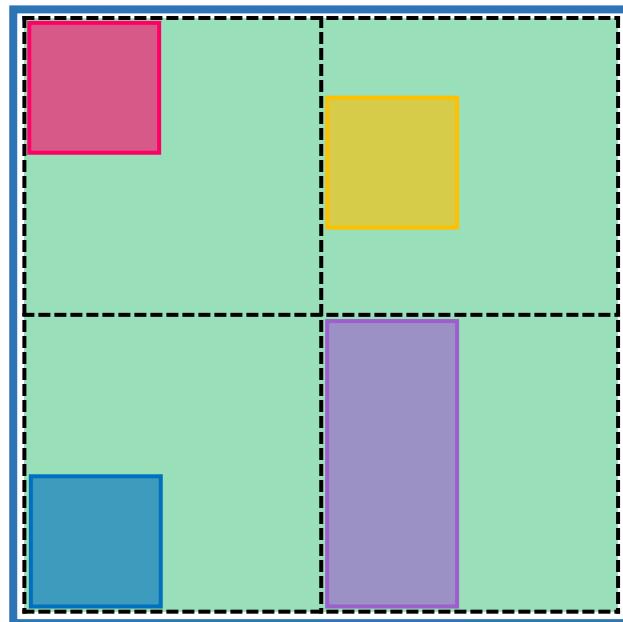
속성	설명
normal	(=stretch)
start	시작점(위쪽) 정렬
center	수직 가운데 정렬
end	끝점(아래쪽) 정렬
stretch	열 축을 채우기 위해 그리드 아이템을 늘림



align-self

단일 그리드 아이템의 수직 정렬 방식을 지정하는 속성

align-self : start;



align-self : center;

align-self : end;

align-self : stretch;





Grid Item

justify-self

- 단일 그리드 아이템(Item)을 수평(행 축) 정렬하는 속성

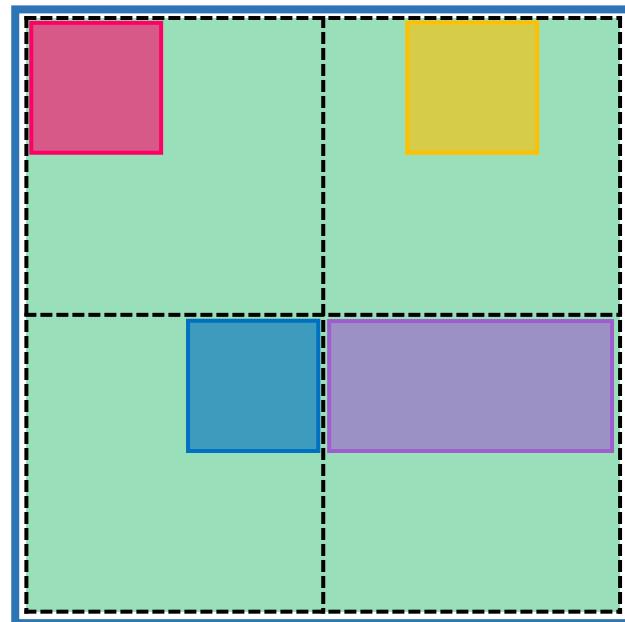
속성	설명
normal	stretch와 같습니다.
start	시작점(왼쪽) 정렬
center	수평 가운데 정렬
end	끝점(오른쪽) 정렬
stretch	행 축을 채우기 위해 그리드 아이템을 늘림



justify-self

단일 그리드 아이템의 수직 정렬 방식을 지정하는 속성

justify-self : start;



justify-self : center;

justify-self : end;

justify-self : stretch;





Grid Item

place-self

- align-self와 justify-self의 단축 속성

속성	설명
normal	stretch와 같습니다.
start	시작점(왼쪽) 정렬
center	수평 가운데 정렬
end	끝점(오른쪽) 정렬
stretch	행 축을 채우기 위해 그리드 아이템을 늘림



Grid Item

order

- 그리드 아이템이 자동 배치되는 순서를 지정하는 속성



Grid Item

z-index

- 아이템이 쌓이는 순서를 지정하는 속성



수고하셨습니다😊



한성호 강사
(ALOHA)