

# Front-End

---

## References

- [HTML tag reference - https://www.w3schools.com/tags/default.asp](https://www.w3schools.com/tags/default.asp)
- [HTML Input Types - https://www.w3schools.com/tags/tag\\_input.asp](https://www.w3schools.com/tags/tag_input.asp)
- [HTML validation - https://validator.w3.org/#validate\\_by\\_uri](https://validator.w3.org/#validate_by_uri)
- [웹 접근성 진단 서비스 - https://accessibility.kr/](https://accessibility.kr/)
- [WEBP 지원 브라우저 확인 - https://caniuse.com/webp](https://caniuse.com/webp)
- [Google Material Color Palette - https://m2.material.io/design/color/the-color-system.html#tools-for-picking-colors](https://m2.material.io/design/color/the-color-system.html#tools-for-picking-colors)
- [Adobe Color - https://color.adobe.com/ko/explore](https://color.adobe.com/ko/explore)

## Table of Contents

- [Front-End](#)
  - [References](#)
  - [Table of Contents](#)
  - [Section1. 강의 준비](#)
    - [설치](#)
  - [Section2. 모던 웹 기본 기술 이해](#)
    - [웹 기본 기술 이해와 정리](#)
    - [HTML](#)
      - [★ 주요 META NAME\(가장 일반적으로 많이 사용\)](#)
      - [호환성 관련 태그](#)
      - [반응형 웹 관련 태그](#)
      - [LINK](#)
      - [STYLE](#)
      - [주요 body 내의 태그](#)
      - [Semantic Web](#)
      - [알아둘 사항](#)
      - [이미지](#)
  - [Section3. 모던 웹의 핵심 상세한 CSS 기본](#)
    - [CSS 기본 정리](#)
      - [CSS란?](#)
      - [Selector\(CSS Selector 라고도 함\)](#)
      - [HTML CSS 연동 방법](#)
        - [1. 적용할 태그에 style 속성으로 넣기\(해당 태그에만 적용\)](#)
        - [2. HTML 문서 <head>에 <style>...</style> 태그로 넣기\(전체 문서에 적용\)](#)
        - [3. HTML 문서 <head>안에 CSS 파일로 링크하기\(전체 문서에 적용\)](#)
    - [Reset CSS](#)
    - [CSS 단위](#)
      - [CSS 크기 단위](#)
      - [반응형 Viewport 단위](#)
        - [Viewport 단위](#)

## ■ 색상 표현 단위

# Section1. 강의 준비

## 설치

- PDF
  - ☒ 00\_Preparation.pdf

# Section2. 모던 웹 기본 기술 이해

- Folder
  - ☒ 01\_head
- PDF
  - ☒ 01\_html정리.pdf

## 웹 기본 기술 이해와 정리

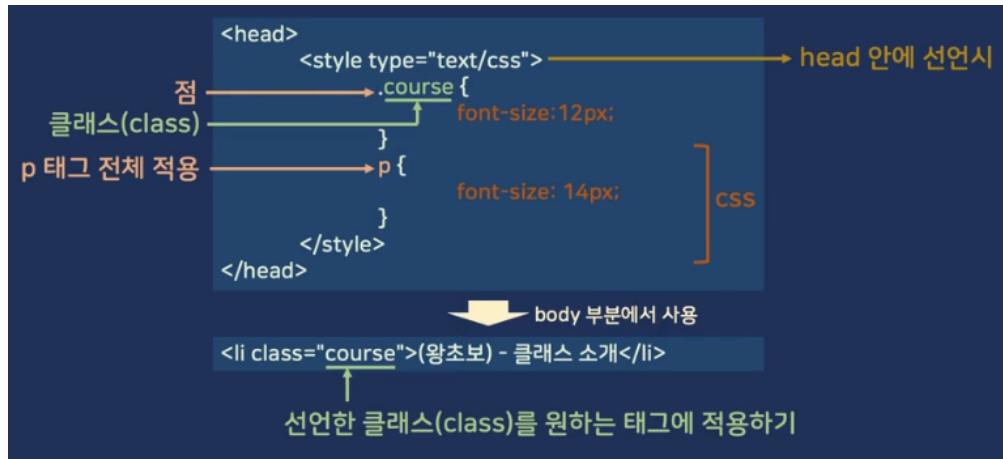
### 1. HTML : Hyper Text Markup Language의 약자로 웹사이트를 만들기 위한 **기본적인 언어**



- `<!DOCTYPE html>`: HTML5라는 신조어로 **문서를 선언**하는 태그
- `<html>...</html>`: HTML 문서의 **시작과 끝을 의미**하며 모든 HTML 태그들은 `<html>` 태그 안쪽으로 입력
- `<head>...</head>`: 웹사이트의 간단한 **요약정보**를 담는 영역이고 웹사이트에서 노출되지 않는 정보
- `<body>...</body>`: 웹사이트에서 눈에 보이는 **정보**를 담는 영역으로, 이미지나 텍스트 처럼 출력되는 정보
- `<meta charset="UTF-8">`: character setting의 약자를 나타내는 문자코드로, 모든 문자를 웹 브라우저에서 **깨끗이** 표시하겠다는 의미
- `<title>...</title>`: 웹사이트 탭에 나타나는 **제목**을 적는 태그

### 2. CSS 언어

- Cascading Style Sheets의 약자로 HTML 문서의 **스타일**을 지정하는 언어



#### ◦ CSS 언어 적용하기

1. 적용할 태그에 `style` 속성으로 넣기(해당 태그에만 적용)
2. HTML 문서 `<head>`에 `<style>...</style>` 태그로 넣기(전체 문서에 적용)
3. HTML 문서 `<head>`안에 CSS 파일로 링크하기(전체 문서에 적용) **Example**

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

### 3. javascript - ES6

- 웹사이트에 동적인 기능을 추가하는 언어

## HTML

- PDF
  - ☒ 01\_html정리.pdf
- 실습파일
  - ☒ html
  - ☒ html\_tag

### ★ 주요 META NAME(가장 일반적으로 많이 사용)

```
<meta name="desription" content="웹사이트 설명">
<meta name="keywords" content="키워드1, 키워드2, 키워드3">
<meta name="author" content="작성자">
```

- **desription**: 웹사이트 설명
- **keywords**: 해당 페이지가 어떤 키워드로 검색되는지
- **author**: 웹사이트 작성자

### 호환성 관련 태그

```
<meta http-equiv="X-UA-Compatible" content="IE=edge">
```

- 인터넷 익스플로러(ID)에서 최신 표준 모드로 렌더링 되도록 하는 설정

반응형 웹 관련 태그

```
<meta name="viewport" content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0" >
```

- viewport: 웹사이트가 보여지는 화면의 크기를 설정하는 태그

속성	설명	주요값
width	초기 뷰포트 너비 설정	<b>device-width</b> 또는 양의 정수(디바이스 너비 또는 특정 너비)
user-scalable	사용자가 뷰포트 크기를 조정할 수 있는지 여부	<b>yes</b> 또는 <b>no</b> (조정 가능 또는 불가능)
initial-scale	디바이스 너비와 뷰포트 너비 비율 설정	0.0과 10.0사이 수(주로 1.0을 많이 사용함)
maximum-scale	최대 확대/축소 비율 설정	0.0과 10.0사이 수(주로 1.0을 많이 사용함)
minimum-scale	최소 확대/축소 비율 설정	0.0과 10.0사이 수(주로 1.0을 많이 사용함)

LINK

```
<link rel="stylesheet" href="style.css">
<link rel="stylesheet" href="favicon.ico">
```

속성	설명	주요값
rel	html 문서와 외부 데이터와의 관계 표시	<b>stylesheet</b> (스타일시트), <b>icon</b> (아이콘)
href	외부 데이터의 경로	파일 경로(상대 경로 또는 절대 경로로 설정)

STYLE

```
<style>
  body {
    background-color: #f0f0f0;
    color: #333;
```

```
font-family: Arial, sans-serif;
}
```

## 주요 body 내의 태그

1. `<h1>` ~ `<h6>`: 제목 태그로, `<h1>`이 가장 큰 제목

현업에서는 웹브라우저 호환성을 위해, 태그에 표현 서식이 들어 있는 경우 모두 삭제하고 CSS style을 별도로 적용

2. `<p>`: 문단 태그로, 문단을 구분하는 태그
3. `<a>`: 링크 태그로, 다른 페이지로 이동하는 링크를 생성하는 태그

```
<a href="https://www.naver.com" target="_blank">네이버</a>
```

- `href`: 링크 주소
- `target`: 링크를 클릭했을 때 새 창으로 열지 여부
  - `_blank`: 새 창으로 열기
  - `_self`: 현재 창에서 열기(기본값)
  - `_parent`: 부모 프레임에서 열기
  - `_top`: 전체 창에서 열기

4. `<ol>`, `<ul>`, `<li>`: 리스트 태그로, 순서가 있는 리스트와 순서가 없는 리스트를 생성하는 태그

```
<ol>
  <li>1번</li>
  <li>2번</li>
  <li>3번</li>
</ol>
<ul>
  <li>1번</li>
  <li>2번</li>
  <li>3번</li>
</ul>
```

- `<ol>`: 순서가 있는 리스트
- `<ul>`: 순서가 없는 리스트

5. `<img>`: 이미지 태그로, 이미지를 삽입하는 태그

```

```

속성	설명	주요값
src	이미지 경로	상대경로 또는 절대경로로 설정
alt	이미지 설명	이미지가 로드되지 않을 때 대체 텍스트

alt는 웹접근성을 높이는 필수 속성처럼 다루는 것이 좋음(구글 SEO 문서에서도 강조)

## 6. <div>: 블록 레벨 태그로, 구역을 나누는 태그

- division의 약자로, html 문서의 특정 부분을 지정하는데 사용(화면에 표시가 달라지는 부분은 아님)
- div 태그는 CS 또는 javascript와 함께 사용하여 레이아웃을 구성하는 데 많이 사용됨

```
<div>
  <h1>제목</h1>
  <p>내용</p>
</div>
```

## 7. <table>: 테이블 태그로, 표를 생성하는 태그

```
<table>
  <thead>
    <tr>
      <th>제목1</th>
      <th>제목2</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>내용1</td>
      <td>내용2</td>
    </tr>
    <tr>
      <td>내용3</td>
      <td>내용4</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td>합계1</td>
      <td>합계2</td>
    </tr>
  </tfoot>
</table>
```

- <thead>: 테이블의 제목 부분
- <th>: 테이블의 제목 셀
- <tbody>: 테이블의 본문 부분
- <tr>: 테이블의 행(row)

- `<td>`: 테이블의 데이터 셀
- `<tfoot>`: 테이블의 마지막 행

thead나 tfoot은 생략 가능하거나 한번만 나와야하며, tfoot은 thead보다 뒤에 위치 tr과 td로만 구성해도 문제없음

8. `<form>`: 폼 태그로, 사용자 입력을 받는 폼을 생성하는 태그, `<input>` 태그와 함께 사용

속성	설명	주요값
action	폼 제출 시 이동할 URL	URL
method	폼 제출 방식	<b>GET</b> 또는 <b>POST</b>
target	폼 제출 시 이동할 프레임	<b>_blank</b> (새 창), <b>_self</b> (현재 창), <b>_parent</b> (부모 프레임), <b>_top</b> (전체 창)

9. `<input>`: 입력 태그로, 사용자 입력을 받는 태그

- 참고: [HTML Input Types](#)

```
<input type="text" name="이름" placeholder="이름을 입력하세요">
```

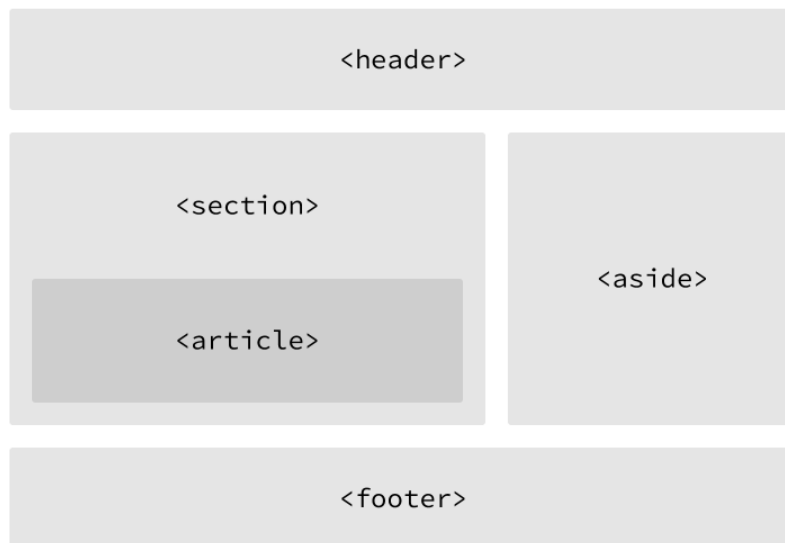
속성	설명	주요값
type	입력 타입	<b>text</b> (텍스트), <b>password</b> (비밀번호), <b>checkbox</b> (체크박스), <b>radio</b> (라디오 버튼)
maxlength	최대 길이	입력 필드에 입력할 수 있는 최대 문자 수
minlength	최소 길이	입력 필드에 입력할 수 있는 최소 문자 수
autofocus	자동 포커스	페이지 로드 시 자동으로 커서가 놓이도록 하는 설정
autocomplete	자동 완성 여부	<b>on</b> (자동 완성), <b>off</b> (자동 완성 안함)
name	입력 필드 이름	서버로 전송되는 데이터의 이름
value	기본값	입력 필드의 기본값
required	필수 입력 여부	<b>required</b> (필수 입력)
disabled	비활성화 여부	<b>disabled</b> (비활성화)

## Semantic Web

- html5에서 추가된 태그로, 웹사이트의 구조를 명확하게 정의하는 태그

- 구조화하여 검색 엔진이 이해할 수 있도록 하는 기술

시멘틱 웹 태그	설명
header	웹사이트의 머리말을 나타내는 태그
nav	웹사이트의 내비게이션을 나타내는 태그
aside	웹사이트의 사이드바를 나타내는 태그
section	본문의 여러 내용(article)을 포함하는 부분
article	본문의 주 내용이 들어가는 부분
footer	웹사이트의 바닥글을 나타내는 태그



## 알아둘 사항

1. 웹 표준
2. 웹 접근성
3. 크로스 브라우징 : 웹사이트가 다양한 브라우저에서 동일하게 보이도록 하는 것

## 이미지

- 이미지는 비트맵과 벡터 이미지로 구분
  - 비트맵: 픽셀로 구성된 이미지
    - 정교하고 다양한 색상을 표현할 수 있지만, **확대하면 픽셀이 깨짐**
    - JPG, PNG, GIF, BMP
  - 벡터 이미지: 수학적 공식으로 구성된 이미지
    - 확대해도 깨지지 않지만, **색상 표현이 제한적**
    - SVG, AI, EPS

### 1. JPEG(JPG) : 높은 압축률

- 손실 압축 방식(원본 이미지와 일부 다른 정보)

### 2. GIF



- 8비트 색상(256색)
- 비손실 압축 방식(원본 이미지와 동일한 정보)

### 3. PNG

- W3C 권장 이미지 포맷
- 비손실 압축 방식(원본 이미지와 동일한 정보), 단 파일 용량이 큼

### 4. ★ WEBP : 동영상, 투명도 지원되는 JPG, PNG, GIF 모두 지원

- Google 에서 개발한 이미지 포맷
- 일부 브라우저에서 미지원
- 손실/비손실 압축 방식 모두 지원

## Section3. 모던 웹의 핵심 상세한 CSS 기본

### CSS 기본 정리

- PDF
  - ☒ 02\_css\_기본정리
- 실습파일
  -

### CSS란?

- Cascading Style Sheets의 약자로 HTML 문서의 **스타일을 지정**하는 언어
- HTML로 구조화된 문서를 어떻게 브라우저 상에 렌더링할지 표현 방법을 정의하기 위한 언어
- 현재 사용하는 CSS는 CSS3로, CSS2의 모든 기능을 포함하고 있으며, CSS2.1에서 추가된 기능도 포함



### Selector(CSS Selector 라고도 함)

- CSS에서 스타일을 적용할 HTML 요소를 선택하는 방법



## HTML CSS 연동 방법

### 1. 적용할 태그에 `style` 속성으로 넣기(해당 태그에만 적용)

가급적 지양할 것

```
<!DOCTYPE html>
<html>
  <body>
    <h1 style="color:blue;">Hello World!</h1>
    <p style="color:red;">This is a paragraph.</p>
  </body>
</html>
```

### 2. HTML 문서 `<head>`에 `<style>...</style>` 태그로 넣기(전체 문서에 적용)

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      h1 {
        color: blue;
      }
      p {
        color: red;
      }
    </style>
  </head>
  <body>
    <h1>Hello World!</h1>
    <p>This is a paragraph.</p>
  </body>
</html>
```

### 3. HTML 문서 `<head>`안에 CSS 파일로 링크하기(전체 문서에 적용)

- home.html

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <h1>Hello World!</h1>
    <p>This is a paragraph.</p>
  </body>
</html>
```

- style.css

```
h1 {
  color: blue;
}
p {
  color: red;
}
```

## Reset CSS

- 웹 브라우저마다 기본 스타일이 다르기 때문에, 모든 브라우저에서 동일한 스타일을 적용하기 위해 사용

실무에서 필요에 의해 임의로 만든 설정

- 최근에는 normalize.css를 많이 사용
  - cdn(Contents Delivery Network)을 통해서 링크하거나, npm으로 설치하여 사용
  - <https://cdnjs.com/libraries/normalize>

```
<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/normalize/8.0.1/normalize.min.css">
```

- css / min.css
  - css: 일반 css 파일
  - min.css: 압축된 css 파일

## CSS 단위

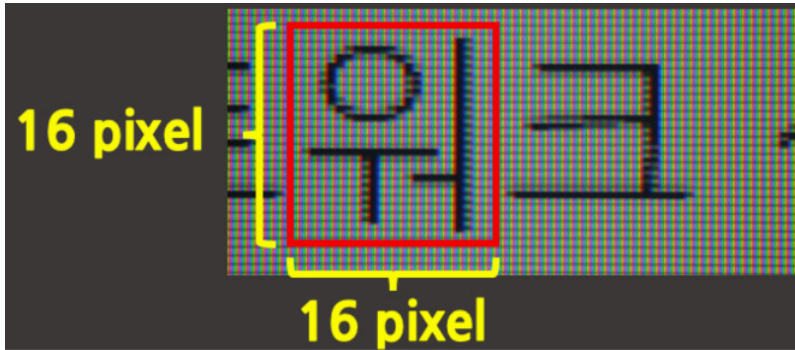
### CSS 크기 단위

-  PDF

- ☒ 03\_css\_단위정리.pdf
- 실습파일
  - ☒ css 단위
  - ☒ viewport 단위

## 1. px

- 픽셀(화소) 단위, 해상도에 따라 상대적인 크기를 가짐

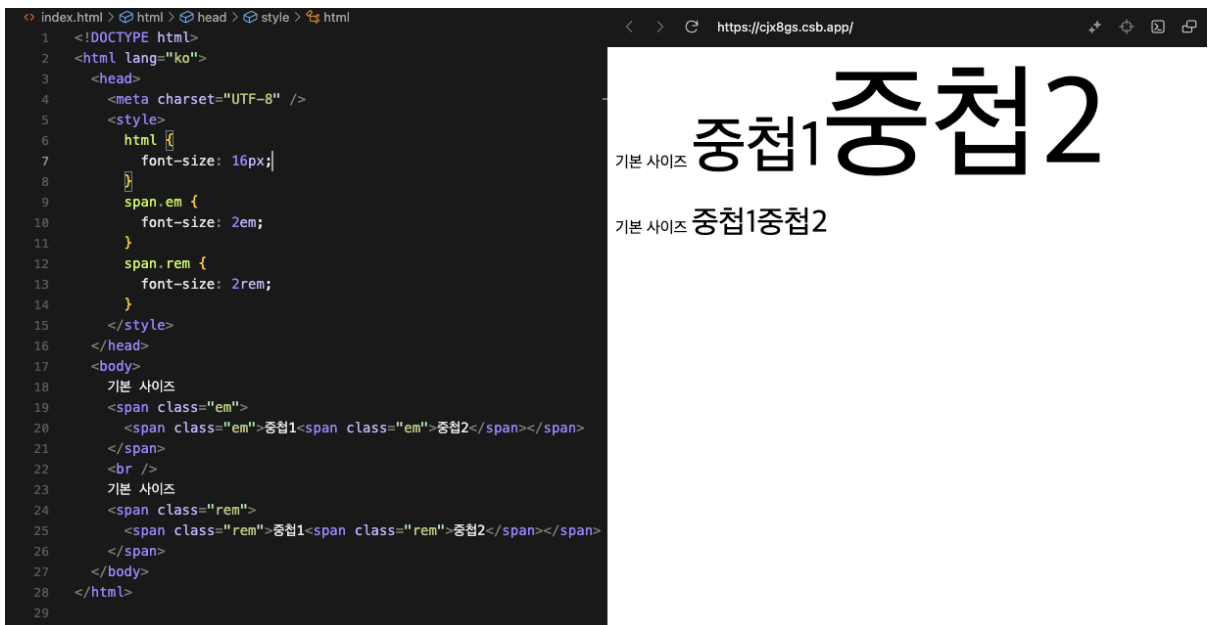


## 2. %(percent)

- 백분율 단위의 상대 단위, 지정 사이즈를 기반으로 상대적인 비율의 크기를 가짐

## 3. em

- 배수 단위, 지정 사이즈를 기반으로 배수로 계산된 크기를 가짐
- 중첩된 자식 요소에 em을 지정하면 모든 자식 요소 사이즈에 영향을 주므로 주의



## 4. rem

- em과 달리 rem은 root em으로, 최상위 요소(html) 사이즈를 기준으로 함

## 반응형 Viewport 단위

- viewport 단위는 화면 크기에 따라 상대적인 크기를 가짐

## Viewport 단위

단위	설명
vw	viewport width의 약자로, 뷰포트 너비의 1%를 의미
vh	viewport height의 약자로, 뷰포트 높이의 1%를 의미
vmin	viewport minimum의 약자로, 뷰포트 너비와 높이 중 작은 값의 1%를 의미
vmax	viewport maximum의 약자로, 뷰포트 너비와 높이 중 큰 값의 1%를 의미

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      .box {
        width: 50vw; /* 뷰포트 너비의 50% */
        height: 50vh; /* 뷰포트 높이의 50% */
        background-color: lightblue;
      }
    </style>
  </head>
  <body>
    <div class="box"></div>
  </body>
</html>
```

## 색상 표현 단위

- Google Material Color Palette - <https://m2.material.io/design/color/the-color-system.html#tools-for-picking-colors>
- Adobe Color - <https://color.adobe.com/ko/explore>

1. 색상 이름으로 표기하는 방법 : 주요 색상 이름 참고 [https://www.w3schools.com/colors/colors\\_names.asp](https://www.w3schools.com/colors/colors_names.asp)

2. 16진수로 표기하는 방법

- #RRGGBB 형식으로 표기
- R: Red, G: Green, B: Blue
- 00 ~ FF(16진수)로 표현
- 예) #FF0000(빨강), #00FF00(초록), #0000FF(파랑)



### 3. RGB로 표기하는 방법

- rgb(0 ~ 255, 0 ~ 255, 0 ~ 255) 형식으로 표기
- 예) rgb(255, 0, 0)(빨강), rgb(0, 255, 0)(초록), rgb(0, 0, 255)(파랑)

### 4. RGBA로 표기하는 방법

- rgba(0 ~ 255, 0 ~ 255, 0 ~ 255, 0.0 ~ 1.0) 형식으로 표기
- 예) rgba(255, 0, 0, 0.5)(빨강 반투명), rgba(0, 255, 0, 1.0)(초록 불투명), rgba(0, 0, 255, 0.2)(파랑 투명)