

HTML 기본 문법

<div> 레이아웃

태그	속성	비고
<div>	style	스타일
	width	가로 크기
	height	세로 크기
	border	테두리 굵기
	background-color	배경 색상
	float	정렬
	margin	여백

ex) `<p style="margin-top:45px;"></p>`

 : 줄바꿈

닫는 태그 x - 무엇인가를 설명하지 않는 태그는 감싸야하는 콘텐츠가 없기 때문
-> img, input, br, hr, meta

<hr> ----- 화면에 이렇게 선 긋는 것

단락<p></p> 내에 있는 content 중 짧은 content(short pieces)를 분리시키고 싶을 때 사용
->특정 content만 inline하고 싶거나 같은 줄 위에 다른 텍스트로서 있게 하고 싶을 때 사용

 목차를 나타낼 때 사용하는 태그, list / 의 항상 자식 태그

 서로 다른 목록을 경계지어 구분하게 해주는 태그, unordered list

 ordered list

<video src="myVideo.mp4" width="320" height="240" controls>

- controls attribute : 비디오 재생에 필요한 기본적인 조작 가능하게 해줌
pause, play, skip

<style>

* comment(주석)

form : <!-- comment -->

* table <table></table>

attribute of table tag : border="1" <!-- 경계 두께 조정-->

= table, td {
border: 1px solid black;
} (in CSS)

** CSS style table

1. table, th, td {

border: 1px solid black;
font_family: Arial, sans-serif; // sans-serif : 글자 획 끝 부분이 깔끔한 것
text-align: center;

}

2. <style>

table {
width: 100%;
border: 1px solid #444444;
border-collapse: collapse; // 테두리 사이의 간격
}

th, td {
border: 1px solid #444444;
padding : 10px; // 테두리와 내용 사이의 간격
}

</style>

+ border-top , border-bottom

**** 구조 표현하는 Tag**

<thead></thead>

<tbody></tbody>

<tfoot></tfoot> : 표 중 가장 아래 section

<tr></tr> : add row

<td></td> : add data in row

-> colspan="2" (attribute) 가로로 2칸 늘림, 주변 칸은 유지 rowspan="2", 세로로 2칸 늘림

<th></th> : table head -> add title to rows and columns

<th scope="col">Sunday</th>

=====

<form></form> :

<form action="/example.html" method="POST"></form>

action attribute => 정보를 어디로 보내는지 결정

method attribute => HTTP verbs(like POST) 꼭 대문자로 할 필요는 x

<input> : <form>에 입력 field가 필요한 경우 사용

- <input type="text" name="first-text-field" value="already pre-filled">

- input field에 입력된 text가 first-text-field에 저장됨

- value attribute로 default value 설정

- input type 속성에 password 입력 시 입력한 값을 화면에 * or 점 으로 대체해서 표현

- Number Input : <input id="years" name="years" type="number" step="1">

- type="range" : Range Input (slider 통한 입력)

-> <input id="volume" name="volume" type="range" min="0" max="100" step="1">

- type="checkbox" : 따로 따로 입력을 받는 체크 입력 다름

- type="radio" : 라디오 버튼 입력 (같은 name의 라디오 버튼 중 하나만 입력 받음)

- required attribute :

required 쓰면 user가 입력하지 않은채 제출을 누르면 제출되지 않고 경고창이 뜬.

<label></label> : input 등 기타에 대한 설명을 웹에 표시하기 위해 사용

- <label for="meal">What do you want to eat?"</label>

- for attribute : input의 id와 동일

<select></select> : <select id="lunch" name="lunch">

<option></option> : Dropdown list

- <option value="tacos">Tacos</option>



- `<datalist></datalist>` : `<select>`와 거의 유사하지만 user가 typing도 가능
- 마찬가지로 child tag로 `<option></option>` 사용
 - `<input>` list의 내용과 `<datalist>`의 id일치 해야함

`<textarea></textarea>` : 텍스트 상자

- `<textarea id="blog" name="blog" rows="5" cols="30">`

`<section class=""></section>`: ??

`<submit Form>`

- `<input type="submit" value="Send">` 전송 버튼 만들기

`<form>` action method="POST"

`<input>` type="range""checkbox""radio""list""submit""text""password"

[Form Validation]

*Text Length 체크

`<input id="summary" name="summary" type="text" minlength="5" maxlength="250" required>`

*Matching a Pattern

- user input이 특정 pattern 따르게 하고 싶은 경우 사용
- `<input id="payment" name="payment" type="text" required pattern="[0-9]{14,16}">`
`"[a-zA-Z0-9]+"`

id와 name의 차이

for과 id 왜 일치시켜야하는지

div와 section의 차이

pattern="[a-zA-Z0-9]+" 뒤에 +의 의미

drop-down list와 listbox(datalist)의 차이

2. 속성(attribute)

- 태그만으로 정보표현하기 부족한 경우 부가정보를 알려주는 역할

2종류 ① The name of the attribute ② The value of the attribute

id : content(element내의 raw text) 간의 구분을 짓게 해줌

src (source)

ex>

숫자(pixel단위) or %(원본기준으로 %계산해서 들어감)로 표현

따로 width height 지정하지 않으면 원본파일 들어감

width, height 둘 중 하나만 들어가면 입력된 하나에 맞춰 원본 가로 세로 비율에 맞춰 나머지 정보도 자동 설정됨.

CSS 기본 문법

- * inline styles(style attribute)

- <p style="color:red; font_size:20px;">I'm learning to code!</p>

- * <style> Tag

- 여러 element에 style 속성 적용시키고 싶을 때 사용
- <head></head>의 자식 tag로 <style></style> 사용한 후 안에 적용 시키고 싶은 내용 기입

```
-> <head>
      <style>
      p {
        color: red;
        font-size: 20px;
      }
    </style>
  </head>
```

- * html과 css 다른 파일로 분류

“”.css 에 <style></style>의 content에 해당되는 내용을 copy, paste it

- * 분류된 css file을 <link> Tag 이용해서 연결시키기

- <link> Tag

-> attribute

1. href="css파일" 2. type="text/css" 3. rel="html과 css파일의 관계(=stylesheet)"

- * class name으로 element 디자인 적용

ex) html <p class="brand">Sole Shoe Company</p>

```
css
.brand{ }
```

ex) html <h1 class="green bold"></h1>

```
css
.green{ color:green; }
.bold{ font-weight:bold; }
```

- * id name으로 element 디자인 적용

ex) html <h1 id="large-title"></h1>

```
css #large-title{ }
```

- * css가 html에 디자인 적용시키는 순서

1. id
2. class
3. tag

- ** css 파일에 만드는 것들을 selector라고 함.

class selector에 적용시키는 건 더 많은 구역에 적용하려고 case를 나눈 것?

id는 딱 특정 구역만 이 디자인을 적용시키고 싶을 때 사용?

** Chaining Selectors (selector끼리 연결시키기)

- h1.special{ } (selector끼리 띄워쓰기 안됨)

** 클래스 내의 특정 tag만 하고 싶을 때

.className TagName { }

* selector order 중에 최고 우선 순위 !important

** CSS Virtual Rules

* CSS의 Structure

1. element 선택 (어떤 대상으로 selector할지)

2. :(colon) (앞-property/뒤-value)

3. :(semicolon) 선언의 끝을 나타냄

* Property

1. font-family : typeface변화

유의할 사항) 1) user의 컴퓨터에 있는 font 사용

2) default font = Times New Roman

3) typeface수 제한시키는게 webPage load 속도가 빨라지고 디자인적으로 좋다

4) typeface를 구성하는 글자 수가 2개 이상이면 “”로 묶는 것이 좋다.

2. font-size: 18px

3. font-weight: bold;

font의 굵기를 정한다. -> value: bold, normal

+ word-spacing: 0.25em;(default)

+ letter-spacing: 0.02em;

4. text-align: right;

텍스트의 정렬을 왼쪽, 오른쪽, 가운데 정렬 중 하나로 정한다.(left, center, right)

+text-transform: uppercase; lowercase; // 대문자, 소문자

+line-height = font-size + leading

+@font-face :

@font-face {

font-family: 'Space Mono';

font-style: normal;

font-weight: 400;

src: local('Space Mono'), local('SpaceMono-Regular'),

url(https://fonts.gstatic.com/s/spacemono/v1/adVweg3BJhE6r8jYmXseHQzyDMXhdD8sAj6OAJTFsBl.woff2) format('woff2');

unicode-range: U+0000-00FF, U+0131, U+0152-0153, U+02C6, U+02DA,

U+02DC, U+2000-206F, U+2074, U+20AC, U+2212, U+2215;

}

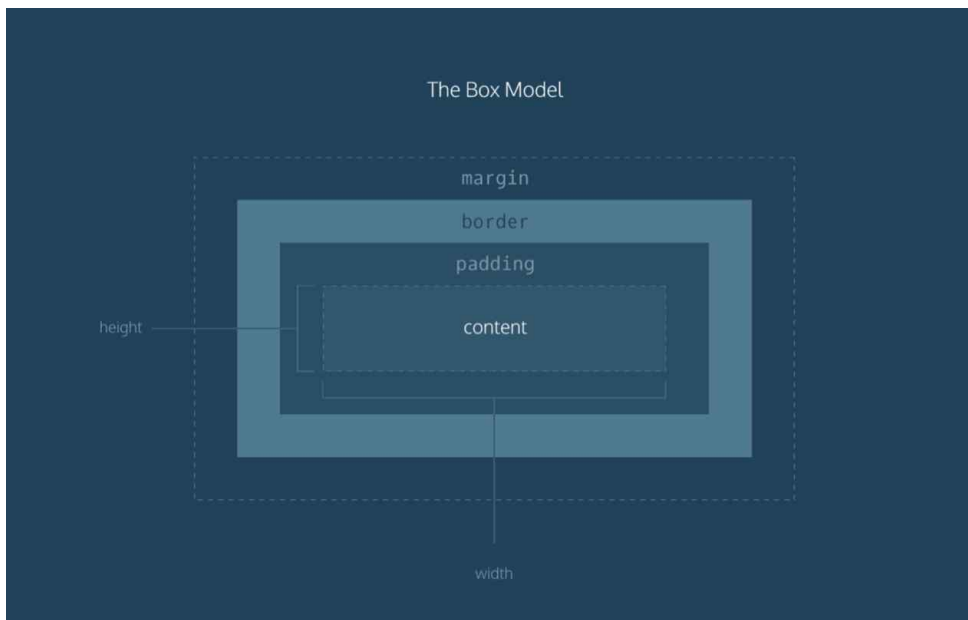
+외부 font 참고

<link rel="stylesheet" type="text/css"

href="https://www.google.~/css?family=Space+Mono:400(height),700(weight),700i>

5. color 배경색과 실제 색을 2가지 구분해서 속성으로 사용
color: red; (foreground color)
color: #000000 or #000 (Black)
color: #8FBC8F;(DarkSeaGreen) 보통은 이렇게 6개의 숫자와 #으로 구성(16진법)
color: rgb(23, 45, 23);
color: hsl(120, 60%, 70%); hsla() -> alpha(0~1) : opacity = color: transparent;
first-parameter-> 0~360 // hue 색깔원에서의 각도
second-> Saturation // 채도 0일 때 탁해짐, 색깔원 중심으로 감
third-> Lightness // 명도 0일때가 어두움
background-color: blue;
6. opacity (투명도 결정)
opacity: 0.5;
7. Background Image
background-image: url("https://www.example.com/image.jpg");

** CSS Box Model



- * height and width
 1. height: 80px;
 2. width: 240px;
- * border
 - border: 3px solid coral;
 - 1. width : thin, medium, thick
 - 2. style : non, dotted, solid 등 10개
 - 3. color : 140개정도
 - + default border -> medium none color
- * border-radius: 5px; <!-- corner of border box -->

ex) div.container{ border: 3px solid rgb(22,77,100); border-radius: 5px; }
ex) div.container{ height: 60px; width: 60px; border: 3px solid rgb(22,77,100);
border-radius: 100%; }

* padding

ex) padding: 10px; or padding-top: 10px;

1. padding-top
2. padding-right
3. padding-left
4. padding-bottom

ex) padding: 6px(top) 11px(right) 4px(bottom) 9px(left);
padding: 5px(top&bottom) 10px(left&right);

* margin

ex) margin: 20px; (top right bottom left)

1. margin-top
 2. margin-right
 3. margin-bottom
 4. margin-left
- + auto

-> margin: 0 auto;

// top과 bottom은 margin 0px, left와 right는 center가 될 때까지 알아서 조정

* Margin Collapse

수평적 left right는 서로의 margin을 더하는데

수직적으로 만나는 것은 margin을 만나는 두 margin 값 중 큰 값을 고름

* Min & Max Height and Width

min-width: 300px; max-width: 600px; min-height: 150px; max-height: 300px;

* Overflow

모든 box model은 border, margin, content, padding 다 합쳐서 총 px 계산됨

그래서 우리는 parent's containing area 보다 작게 나오도록 만들어야 되는데

실제로 child's containing area가 큰 경우 overflow attribute 사용해서 해결

property 1. hidden : 겹치는 부분 안보이게 함

2. scroll : 안보이는 부분 scroll

3. visible : parent's containing area 밖으로도 보이게 함

* Resetting Defaults

```
*{  
margin: 0;  
padding: 0;  
}
```

* Visibility

1. visibility: hidden;
2. visibility: visible;

+ display: none; 은

웹페이지에서 아예 삭제시키는 것이고 visibility: hidden;은 해당 공간은 존재하는 것

* box-sizing property

default-> content-box

1. content-box : left right padding + content width + border를 actual render

2. border-box : height width 설정한 것에 값을 고정시켜서 content width, height
auto로 설정, padding과 border두께는 변하지 않음

3. all box model 에 적용할 때

* { box-sizing: border-box; }

**** CSS display and position**

* position property

value : 1. static(default)

2. relative -> default position을 기준으로 top down left right pixel 결정
related property : top(move the element down) / bottom/ left/ right

3. absolute

4. fixed : parent element와 무관하게 position이 고정된다

*z-index:1(int);

어떤 element를 앞으로 또는 뒤로 보낼지 정하는데 사용

숫자가 클수록 web page 상에 앞에서 나옴

position:static;에서는 적용 안됨.

* **Display**

*inline Display

- display property - inline, block, inline-block

- inline elements(<a>)는 CSS로 height or width 조절 X

- CSS에서 어떤 element selector를 display: inline; 으로 지정하면 해당 element 들은
inline이 된다.

*Block Display

- <h1>~<h6>,<p>,<div>,<footer>

- strong{ display: block; }

*Inline-Block Display

display: inline-block;

*Float Property

Value

1. left : this value will move, or float, elements as far left as possible.

2. right : this value will move elements as far right as possible.

*Clear Property

Value

1. left

2. right

: the right side of the element will not touch any other element within the same containing element.

3. both

4. none