

HTML/CSS

웹 프로그래밍에서 기본적으로 갖춰야 할 HTML/CSS 관련 지식을 공부합시다

HTML 기본

HTML Basic

웹 페이지의 구조를 다루는 HTML에 대해 배웁니다

HTML은?

Hyper Text Markup Language

마크업 언어

태그 등을 이용하여 문서나 데이터의 구조를 명기하는 언어

“하이퍼 텍스트”

웹 페이지를 서로 연결하는 것을 의미

월드 와이드 웹(WWW)의 웹 페이지를 서로 연결하며,
마크업을 통해 웹 페이지의 콘텐츠를 표현합니다

HTML의 기본구조

HTML Structure

HTML5 형식임을 의미합니다

HTML문서의 시작과 끝을 의미합니다

HTML문서에 관한 기본정보를 포함합니다

브라우저의 제목표시줄에 출력될 내용입니다

문서의 본문에 해당합니다

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

주석

Comment

```
<!DOCTYPE html>
<html>
<head>
  <title>Document</title>
</head>
<body>
  <!-- 이 사이에 넣는 내용은 주석이 됩니다 -->
</body>
</html>
```

자주, 습관적으로 사용하는 것이 좋습니다!

주석은 화면에는 표시되지 않으며, 소스에서만 확인할 수 있습니다.

개발기간이 길어지고, 소스가 많아질수록 작업한 내용을 주석으로 정리해야 나중에 소스를 고치거나 참고할 때 효율성이 증가합니다.

태그의 요소와 속성

Element & Attribute

HTML 태그의 구성

<요소 속성="값">또는

<요소 속성="값">내용</요소>

스스로 닫는 태그 (담을 내용이 없음)

<열리는태그>와 </닫히는태그>로 이루어진 태그

예제

네이버 홈으로 가기

href는 Hyper Reference(참조), src는 Source(출처, 소스)의 줄임말입니다.

절대경로와 상대경로

Absolute path & Relative path

절대경로

http://, https://로 시작하는 전체 주소입니다

```

```

상대경로

해당 HTML파일을 기준으로 한 경로입니다

```

```

상대경로에서, 하위 폴더로는 “폴더명/파일명”을 사용하며,
상위 폴더로 올라갈때는 “../”으로 시작합니다.
“/”로 시작한다면 가장 최상위 폴더에서부터 시작합니다.

head 요소

문서의 메타데이터 집합

웹 페이지에 직접적으로 보이지 않는 정보

```
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <link rel="stylesheet" href="style.css">
  <script src="script.js" charset="utf-8"></script>
  <title>Document</title>
</head>
```

웹 페이지의 인코딩 방식을 정의합니다

IE에서의 렌더링 방식을 최신으로 설정합니다

CSS파일을 연결합니다

JavaScript파일을 연결합니다

문서의 제목을 나타냅니다

여기있는 내용에 더해, 다양한 meta태그가 존재할 수 있습니다.

페이스북에 링크했을때의 제목, 설명, 커버이미지를 보여주는 meta태그나, 검색엔진에서 주로 사용할 내용, 모바일에서의 확대/축소여부 등 여러가지 설정값을 지정할 수 있습니다.

또한, CSS나 JS파일의 링크 역시 head요소 내부에 link, style, script요소를 사용해 나타냅니다.

블록과 인라인

Block, Inline

요소의 형태는 두 가지로 나누어집니다

블록 요소

Block Level Elements

줄 바꿈이 일어나는 형태

`<h1>블록 요소</h1>`

`<p>p요소는 블록 형태입니다</p>`

`<div>div요소도 블록 형태입니다</div>`

블록 요소

p 요소는 블록 형태입니다.

div 역시 블록 형태입니다.

블록 요소는 줄 바꿈이 일어나는 형태이며, 기본적으로 width가 전체 너비의 값을 가집니다.

해당 요소에 배경색(background-color)을 지정하게 되면, 실제로 들어있는 글자 내용과는 별개로 전체 너비 영역만큼을 차지한다는 것을 알 수 있습니다.

인라인 요소

Inline Elements

줄 바꿈이 일어나지 않는 형태

```
<strong>strong 요소</strong>
```

```
<a href="">a요소</a>
```

```
<span>span요소</span>
```



인라인 요소는 줄바꿈 없이, 기본적으로는 자신의 내용만큼의 가로너비를 가집니다.

블록 요소는 인라인 요소를 포함할 수 있지만, 인라인 요소는 블록 요소를 포함할 수 없습니다.

HTML 태그

HTML Tags

HTML을 구성하는 태그들에 대해 배웁니다

줄 바꾸기

Line breaks

p태그 (Paragraph, 문단)

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Repudiandae sequi ratione tenetur reiciendis cum in totam atque ipsum similique quae.</p>

<p>Lorem ipsum dolor sit amet.</p>

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Repudiandae sequi ratione tenetur reiciendis cum in totam atque ipsum similique mollitia accusamus provident officiis modi recusandae porro molestiae dicta saepe aperiam voluptatibus numquam, qui ex dolore nemo dolor sapiente. Accusantium, quae.

Lorem ipsum dolor sit amet.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Repudiandae sequi ratione tenetur reiciendis cum in totam atque ipsum similique mollitia accusamus provident officiis modi recusandae porro molestiae dicta saepe aperiam voluptatibus numquam, qui ex dolore nemo dolor sapiente. Accusantium, quae.

Lorem ipsum dolor sit amet.

p | 1904 x 38

각 p태그 아래에 공백이 생깁니다

br태그 (Linebreak, 줄 바꾸기)

Lorem ipsum dolor sit amet.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Tenetur, ab.

Lorem ipsum dolor sit amet.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Tenetur, ab.

줄바꿈 간에 여백이 없습니다

헤드라인

Headline

웹 페이지의 개요를 나타냄

h1 ~ h6

HTML

역사

개발

1980년, 유럽 입자 물리 연구소(CERN)의 계약자였었던 물리학자 팀 버너스리가 HTML의 원형인 인콰이어를 제안하였다.
~ 중략 ~
버너스리의 개인적인 기록에 1990년부터 "하이퍼텍스트가 사용되는 여러 분야의 일부"를 열거했고 백과사전을 그 목록의 첫 번째로 두었다.

최초 규격

HTML 최초의 일반 공개 설명은 1991년 말에 버너스리가 처음으로 인터넷에서 문서를 "HTML 태그"(HTML tag)로 부르면서 시작되었다.

HTML 코드

<h1>HTML</h1>

<h2>역사</h2>

<h3>개발</h3>

<h3>최초규격</h3>

중요도 순으로 개요를 나타낼 때 사용합니다.

학술문서나 검색엔진에서 검색시에 중요하게 사용됩니다.

실제 글자크기등은 CSS에서 만들고자 하는 웹 페이지에 맞춰서 새로 설정하므로, 단계별로 구분할 제목이 있다면 hn태그를 사용하는 것이 좋습니다.

링크

Anchor

a

새 창에서 링크를 열어줍니다

```
<a href="http://www.naver.com" target="_blank" title="네이버 열기">sdf</a>
```

href : 이동할 페이지 주소

target : 링크 걸린 페이지를 여는 방법 (_self, _blank)

title : 마우스를 올렸을 때 보여줄 제목

링크를 클릭한 창을의 주소를 바꿉니다

****target에는 추가로 parent, top, <frameName>이 있지만, 최근 HTML에서는 Frame레이아웃을 잘 사용하지 않기때문에 설명에서 제외했습니다**

이미지

Image

img

```

```

src : 이미지의 경로

width, height : 이미지의 가로/세로 크기(px단위)

색상

Color

```
<p style="color: #333;">Color #333</p>  
<p style="color: DarkGreen;">Color DarkGreen</p>
```

색상은 Hex code를 사용한 #000000 ~ #FFFFFF까지의 값과,
HTML규격에 미리 정의된 ColorName을 사용할 수 있습니다.

HTML 색상표(2) color name을 이용한 색상표		
Color Name	Color HEX	Color
AliceBlue	#F0F8FF	
AntiqueWhite	#FAEBD7	
Aqua	#00FFFF	
Aquamarine	#7FFFD4	
Azure	#F0FFFF	
Beige	#F5F5DC	
Bisque	#FFE4C4	
Black	#000000	
BlanchedAlmond	#FFEBCD	
Blue	#0000FF	
BlueViolet	#8A2BE2	
Brown	#A52A2A	
BurlyWood	#DEB887	
CadetBlue	#5F9EA0	
Chartreuse	#7FFF00	
Chocolate	#D2691E	
Coral	#FF7F50	
CornflowerBlue	#6495ED	
Cornsilk	#FFF8DC	
Crimson	#DC143C	
Cyan	#00FFFF	
DarkBlue	#00008B	
DarkCyan	#008B8B	
DarkGoldenRod	#B8860B	
DarkGray	#A9A9A9	
DarkGrey	#A9A9A9	
DarkGreen	#006400	
DarkKhaki	#BDB76B	
DarkMagenta	#8B008B	
DarkOliveGreen	#556B2F	
Darkorange	#FF8C00	
DarkOrchid	#9932CC	
DarkRed	#8B0000	
DarkSalmon	#E9967A	

http://www.homejjang.com/03/color_name.php
http://www.w3schools.com/colors/colors_names.asp

목록

Ordered List, Unordered List

Ordered List

```
<ol>  
  <li>항목</li>  
  <li>항목</li>  
  <li>항목</li>  
  <li>항목</li>  
  <li>항목</li>  
</ol>
```

Unordered List

```
<ul>  
  <li>항목</li>  
  <li>항목</li>  
  <li>항목</li>  
  <li>항목</li>  
  <li>항목</li>  
</ul>
```

목록 형태로 나타나는 요소들은 ol, ul태그를 사용하여 구현 후, CSS로 제작하고자 하는 디자인에 맞게 스타일을 지정해줍니다.

정의 목록

Description List

정의 목록 태그

dt는 목록 중 개념을 나타냅니다

<dl>

<dt>HTML</dt>

<dd>HyperTextMarkupLanguage</dd>

<dd>웹 페이지를 구현하는 마크업 언어이다</dd>

dd는 해당 개념의 정의를 나타냅니다

<dt>CSS</dt>

<dd>Cascading Style Sheet</dd>

<dd>HTML의 형태를 지정하는 언어이다</dd>

</dl>

목록과 정의 목록은 서로 중첩해서 사용이 가능합니다.

레이아웃 요소

div, span

```
<div>  
  <p>블럭요소 내부에 <span>인라인 요소를 사용합니다</span></p>  
</div>
```

지금까지 배운 요소들은 각각 의미와 사용해야 하는 곳이 정해져 있지만, div와 span은 오직 Block과 Inline방식의 레이아웃을 구현하는데에 사용합니다.

클래스와 아이디 속성

class, id

네이밍

첫 글자는 알파벳으로 시작

두 번째부터는 알파벳, 숫자, -, _를 사용 가능

대소문자를 구분

클래스와 아이디의 차이

id(아이디)는 페이지에서 딱 한번만 선언 가능, 요소의 unique한 특성을 나타냄

class(클래스)는 여러번 사용 가능, 범용적인 부분을 나타냄

ID와 Class의 적절한 사용 예

```
<div class="chapter" id="chapter1">  
  <h2>HTML</h2>  
  <p>HTML강의를 시작해봅시다</p>  
</div>
```

```
<div class="chapter" id="chapter2">  
  <h2>CSS</h2>  
  <p>Chapter2는 CSS입니다!</p>  
</div>
```

```
<div class="chapter" id="chapter3">  
  <h2>JavaScript</h2>  
  <p>JavaScript는 야무님이죠!</p>  
</div>
```

class와 id의 이름은 짧게짓기보다는, 길더라도 해당 요소의 의미에 적합하게 짓는 것이 더 좋습니다.

CSS 기본

CSS Basic

CSS란 무엇이며, 어떻게 사용하는 것인지?

CSS란?

Cascading Style Sheet

마크업 언어(HTML)가 실제 표시되는 방법을 기술하는 언어
레이아웃과 스타일을 정의할 때 주로 사용

HTML과 CSS를 분리해서 사용해야 하는 이유는?

과거의 HTML

혼돈! 파괴! 망각각!

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  </head>
  <body bgcolor="Azure">
    <font size="4" face="arial" color="DarkBlue">Chaos! Destruction! Oblivion!</font>
    <hr size="1" width="100%" color="DarkCyan">
    <p font-size="20px">CSS가 없는 혼돈의 세계입니다</p>
    <br>
    <hr align="right" width="40%" color="DarkSlateGray">
    <p align="right">여러분은 이렇게 사용하시면 안됩니다</p>
  </body>
</html>
```

HTML에는 스타일을 제외한, 문서의 구조만이 명확히 나타나야 함

과거 브라우저에서 사용하던 font, center 등의 태그는 HTML5에서는 더 이상 사용하지 않습니다

http://www.w3schools.com/html/html5_intro.asp

CSS문법

CSS Syntax Basic



CSS문법

CSS Syntax Basic - Example

```
#body-title {  
    font-size: 14px;  
    font-weight: bold;  
    color: DarkSlateGrey;  
}
```

HTML Color Names

http://www.w3schools.com/colors/colors_names.asp

CSS사용법

내부 스타일 시트 (Internal Style Sheet)

```
<html lang="en">  
<head>  
  <style type="text/css">  
    #body-title {  
      font-size: 14px;  
      font-weight: bold;  
      color: DarkSlateGrey;  
    }  
  </style>  
</head>  
<body>  
  <p id="body-title">Internal Style Sheet</p>  
</body>  
</html>
```

head 안쪽, style태그 내부에 작성

CSS사용법

인라인 스타일 시트 (Inline Style Sheet)

```
<html lang="en">
<head>
</head>
<body>
  <p id="body-title" style="font-size:14px; font-weight: bold; color: DarkSlateGrey">Inline Style Sheet</p>
</body>
</html>
```

사용할 요소의 style속성에 정의

인라인 스타일은 내용과 스타일이 분리되지 않으므로 권장되지 않습니다

CSS사용법

외부 스타일 시트 (External Style Sheet)

```
<html lang="en">
<head>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <p id="body-title">External Style Sheet</p>
</body>
</html>
```

link태그를 사용, href속성에 경로를 입력

link태그를 사용하여 외부 CSS파일을 HTML문서에 연결합니다

CSS 선택자

CSS Selector

CSS를 적용할 요소

CSS선택자

Universal Selector (전체 선택자)

별표 기호

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

HTML페이지 내부의 모든 요소에 같은 CSS속성을 적용합니다.

따라서 margin이나 padding값을 초기화하는 등, 기본값을 정할 때 주로 사용합니다.

다만 문서의 모든 요소를 읽기 때문에 페이지 로딩시간이 길어질 수 있으니 자주 사용하는 것은 좋지 않습니다.

CSS선택자

Type Selector (타입 선택자)

HTML 태그명

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

HTML 태그명

```
p {  
  color: gray;  
  margin-bottom: 10px;  
}
```

해당하는 모든 HTML태그 요소를 지정합니다.

CSS선택자

Class Selector (클래스 선택자)

마침표 기호

CSS

```
.section {  
  color: #333;  
  margin-bottom: 40px;  
}  
p.section-title {  
  font-size: 18px;  
}  
p.section-content {  
  font-size: 13px;  
  line-height: 13px;  
  color: #999;  
}
```

앞에 TAG명 입력 가능

HTML

```
<body>  
  <div class="section">  
    <p class="section-title">Lorem ipsum dolor sit amet.</p>  
    <p class="section-content">Lorem ipsum dolor sit amet</p>  
  </div>  
  
  <div class="section">  
    <p class="section-title">Lorem ipsum dolor sit amet.</p>  
    <p class="section-content">Lorem ipsum dolor sit amet</p>  
  </div>  
</body>
```

CSS에서는 마침표 기호로 나타내며, HTML에서는 주어진 값을 class속성값으로 가진 요소를 선택합니다.
마침표 앞에 태그를 붙여주면 범위는 지정한 태그에 한합니다.

CSS선택자

ID Selector (ID 선택자)

기호

CSS

```
#index-title {  
  font-size: 18px;  
}  
p#index-description {  
  font-size: 12px;  
  color: #999;  
}
```

앞에 TAG명 입력 가능

HTML

```
<body>  
  <h3 id="index-title">Lorem ipsum dolor sit.</h3>  
  <p id="index-description">Lorem ipsum doloro?</p>  
</body>
```

CSS에서는 #기호로 나타내며, HTML에서는 주어진 값을 id속성값으로 가진 요소를 선택합니다.

HTML에서 id값은 오직 하나만 존재해야 합니다.

클래스 선택자와 같이 앞에 TAG명을 입력할 수 있습니다.

ID선택자의 우선순위가 Class선택자의 우선순위보다 높으므로, 같은 속성에 서로 다른 값을 지정할 경우 ID선택자의 값이 적용됩니다.

CSS선택자

Combinator Selector (복합 선택자)

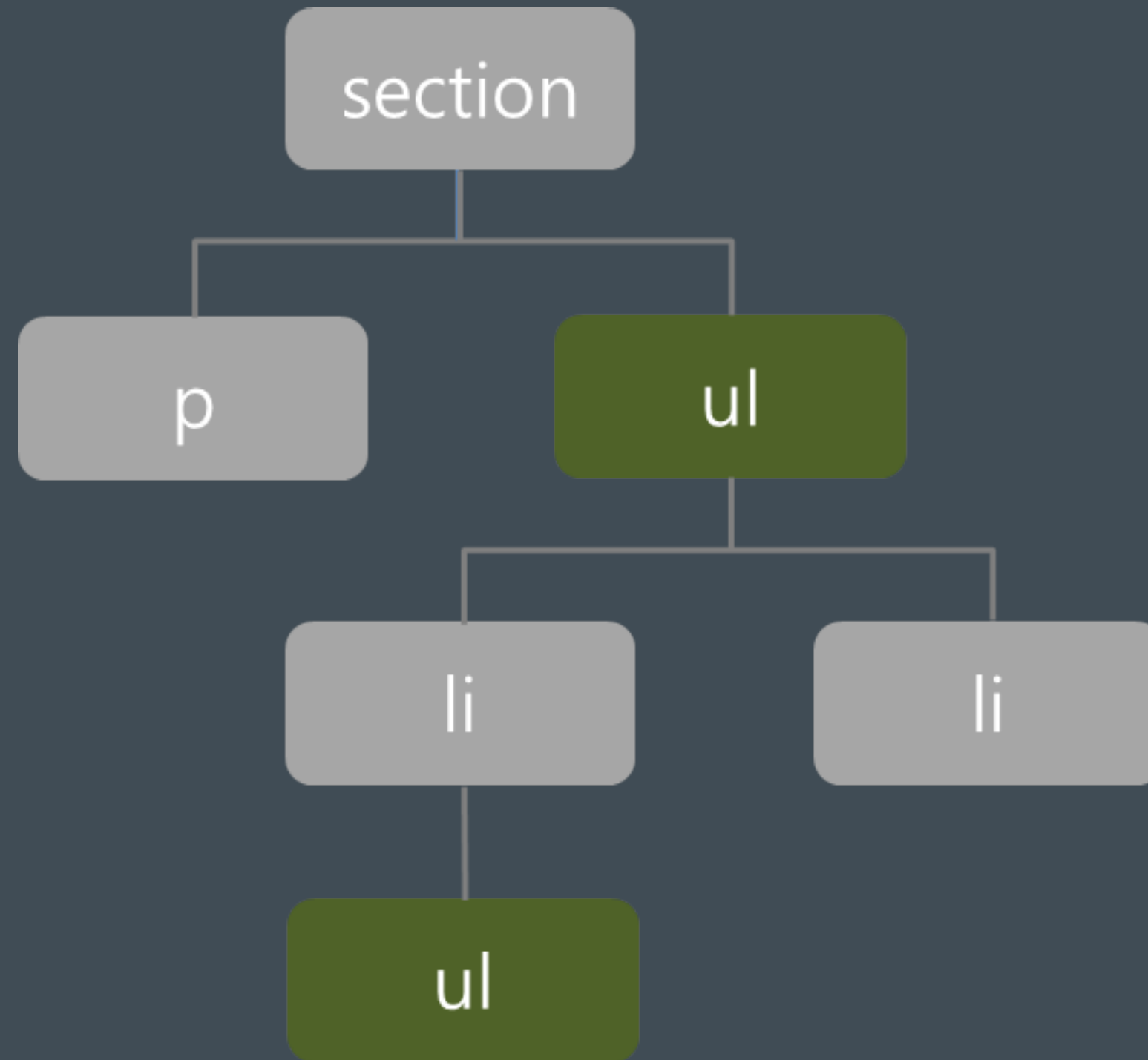
하위선택자와 자식선택자 (Descendant and Child)

하위 선택자

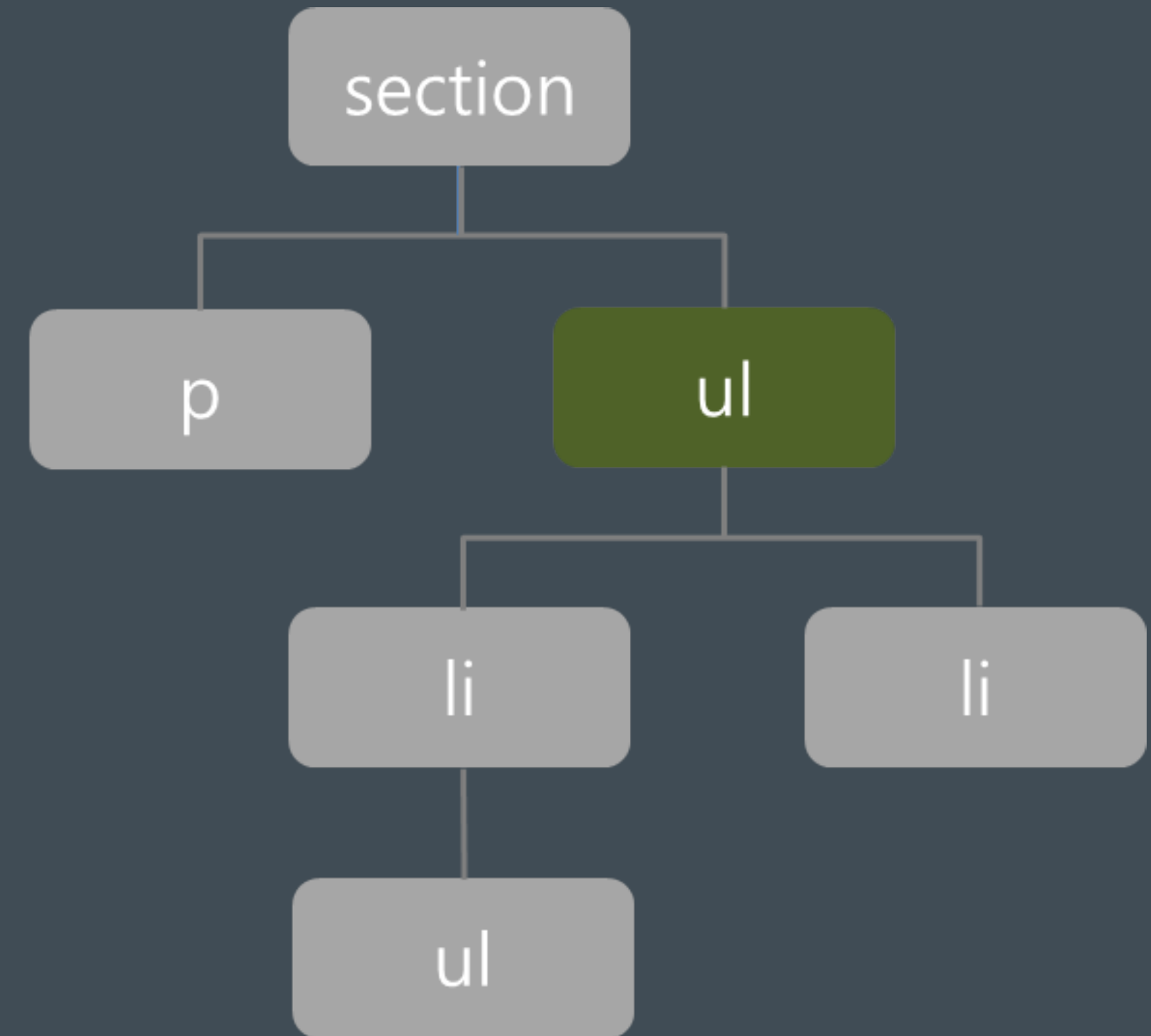
```
section ul {  
  border: 1px solid black;  
}  
section>ul {  
  border: 1px solid black;  
}
```

자식 선택자

하위 선택자



자식 선택자



포함 관계를 가지는 태그들 사이에서, 포함하는 요소는 '부모 요소', 포함되는 요소는 '자식 요소'라고 합니다.

하위 선택자는 부모요소에 포함된 '모든' 하위 요소를 지정하며,

자식 선택자는 부모요소의 '바로 아래' 자식 요소만을 지정합니다.

CSS선택자

Combinator Selector (복합 선택자)

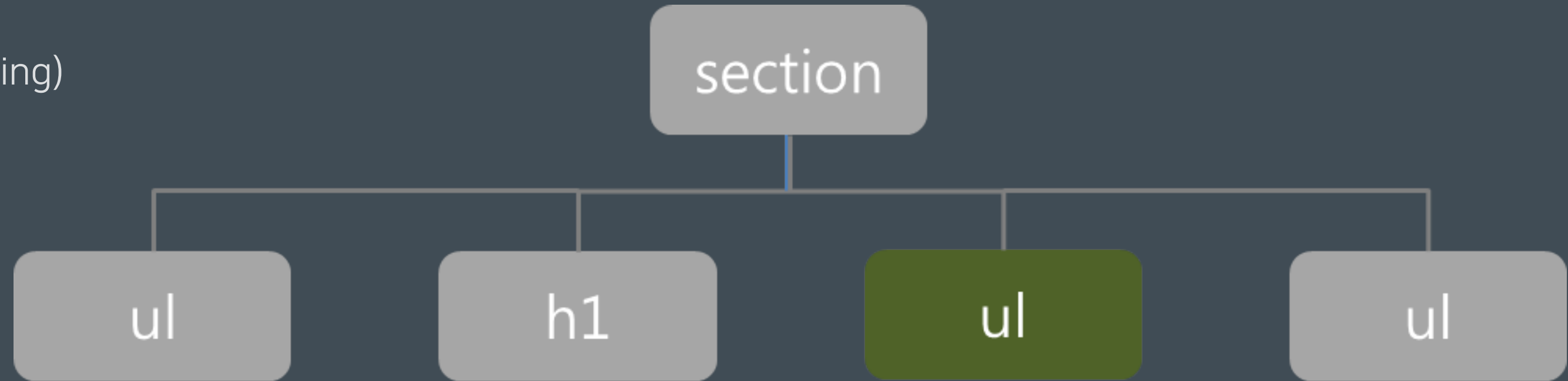
인접 형제 선택자와 일반 형제 선택자 (Adjacent Sibling & General Sibling)

인접 형제 선택자

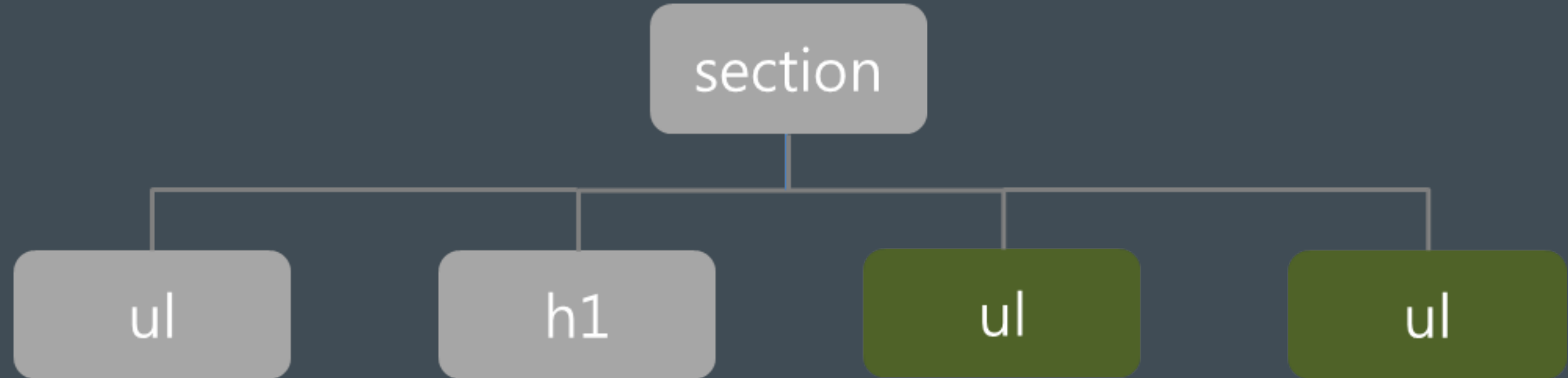
```
h1+ul {  
  background: Azure;  
  color: DarkBlue;  
}  
h1~ul {  
  background: Azure;  
  color: DarkBlue;  
}
```

일반 형제 선택자

인접 형제 선택자



일반 형제 선택자



같은 부모 요소를 가지는 요소들은 '형제 관계' 라고 부릅니다.

이 때, 먼저 나오는 요소를 '형 요소', 나중에 나오는 요소는 '동생 요소'라 합니다. (부모 요소 내부에서 보다 위 줄에 쓰여진 것을 의미합니다)

두 선택자 모두 형 요소에는 적용되지 않으며, 인접 형제 선택자는 조건을 충족하는 '첫 번째' 동생 요소만을 지정하며, 일반 형제 선택자는 조건을 충족하는 '모든' 동생 요소를 지정합니다.

CSS선택자

속성 선택자 (Attribute Selector)

태그 내의 속성에 따름

패턴	의미	예제
<code>E[attr]</code>	'attr'속성이 포함된 요소 E	<code><E attr>Lorem</E></code>
<code>E[attr="val"]</code>	'attr'속성의 값이 'val'인 요소 E	<code><E attr="val">Lorem</E></code>
<code>E[attr~="val"]</code>	'attr'속성의 값에 'val'이 포함되는 요소 E (공백으로 분리된 값이 일치해야 함)	<code><E attr="val">Lorem</E></code> <code><E attr="item val">Lorem</E></code> <code><E attr="item-val">Lorem</E></code>
<code>E[attr^="val"]</code>	'attr'속성의 값이 'val'로 시작하는 요소 E	<code><E attr="val">Lorem</E></code> <code><E attr="value">Lorem</E></code>
<code>E[attr\$="val"]</code>	'attr'속성의 값이 'val'로 끝나는 요소 E	<code><E attr="val">Lorem</E></code> <code><E attr="item-val">Lorem</E></code>
<code>E[attr*="val"]</code>	'attr'속성의 값에 'val'이 포함되는 요소 E (공백이나 Dash(-)에 영향받지 않음)	<code><E attr="val">Lorem</E></code> <code><E attr="many itemval">Lorem</E></code> <code><E attr="item-val">Lorem</E></code>
<code>E[attr ="val"]</code>	'attr'속성의 값이 'val'이거나, 'val-'로 시작하는 요소 E	<code><E attr="val">Lorem</E></code> <code><E attr="val-num3">Lorem</E></code> <code><E attr="value">Lorem</E></code>

CSS선택자

가상 클래스 선택자 (Pseudo-Classes Selector), 가상 엘리먼트 선택자 (Pseudo-Elements Selector)

HTML소스에는 존재하지 않지만 필요에 의해 가상의 선택자를 지정

패턴	의미
E:link	방문하지 않은 링크 E
E:visited	방문한 링크 E
E:active	E 요소에 마우스 클릭 또는 키보드 엔터가 눌린 동안
E:hover	E 요소에 마우스가 올라가 있는 동안
E:focus	E 요소에 포커스가 머물러 있는 동안
E::first-line	E 요소의 첫 번째 라인
E::first-letter	E 요소의 첫 번째 문자
E::before	E 요소의 시작 지점에 생성된 요소
E::after	E 요소의 끝 지점에 생성된 요소

CSS 우선순위

CSS Cascading

스타일 적용의 우선순위를 알아봅니다

CSS스타일 적용 우선순위

특정도(specify)값이 높은 순서대로 적용됩니다
특정도는 가장 구체적인 값을 의미합니다

특정도 계산식

스타일	특정도
Inline (인라인 스타일)	1000
ID Selector (ID 선택자)	100
Class Selector (클래스 선택자)	10
TAG Selector (태그 선택자)	1

특정도는 CSS구문에서
해당하는 스타일의 수 * 특정도 값을 모두 더한 값입니다.

클래스 선택자가 3개 존재하며, 태그선택자가 있는 CSS구문이라면
ex) p.wrap.item>.active

클래스선택자의 갯수(3) * 클래스선택자의 특정도(10) + 태그선택자의 개수(1) * 태그 선택자의 특정도(1) = 31의 특정도를 가지게 됩니다.

예제)

CSS 구문	특정도
p { color: gray; }	1
p:first-line { color: black; }	2
.wrap { color: black; }	10
p.wrap { color: black; }	11
p.wrap>.item { color: black; }	22
#wrap { color: black; }	100
p#wrap { color: black; }	101
<!-- HTML --> <p style="color: black;">Example</p>	1000

우선순위의 흑마법

인라인 스타일과 important

from 위키백과 : 흑마술은 전통적으로 악의적, 이기적 목적을 위한 초자연적인 힘의 이용을 일컫는다.

특정도 계산식

스타일	특정도
important	Absolute
Inline (인라인 스타일)	1000
ID Selector (ID 선택자)	100
Class Selector (클래스 선택자)	10
TAG Selector (태그 선택자)	1

CSS

```
p {  
  font-size: 30px !important;  
  color: green !important;  
}
```

HTML

```
<p style="font-size: 10px; color: green;">important는 모든걸 무시하지</p>
```

important 쓴 것을 잊어버리면 나중에 왜 안되는지 헤매게 됨

!important값은 해당하는 요소에 지정된 어떤 특정도 무시하고 가장 우선순위로 적용됩니다.
이는 테스트시에는 유용할 수 있지만, 추후 유지보수나 전체적 스타일 흐름을 방해하기 때문에 이용하지 않는 것이 좋습니다.

CSS 서체

CSS Typography

CSS로 서체에 스타일을 지정합니다

서체 색상

Color

```
h1 {  
  color: #ff0000;  
}
```

color속성은 글자 색을 지정합니다.

HEX코드, rgb(), rgba(), 색상명이 올 수 있습니다.

서체 지정

Font family

```
body {  
    font-family: “돋움”, dotum, “굴림”, gulim, arial, helvetica, sans-serif;  
}
```

웹 페이지를 방문한 사용자가 없을 경우를 대비해, 다양한 서체들을 선언해놓습니다.

순서대로 해당 서체가 없을 경우 다음 서체가 사용자의 컴퓨터에 설치되어 있는지 확인 후, 순서 중 가장 먼저 찾은 폰트를 사용하게 됩니다.

서체의 종류

Font types

Serif

글자에 꺾쇠가 붙어있는 서체, 인쇄물에 많이 사용
영어 서체에는 Georgia, Times New Roman
한글 서체에는 바탕체, 궁서체, 명조체가 있다

Python, 파이썬

Sans-Serif

Sans는 "없음"을 뜻하는 프랑스어.
Serif가 없는 글시체를 말함
영어 서체에는 Arial, Helvetica
한글 서체에는 돋움, 굴림, 나눔 고딕등이 있다.

Python, 파이썬

Monospace

고정 폭 서체
글자들 간 구분이 쉬워야 하는
프로그래밍 코드를 나타낼 때 주로 사용

Python, 파이썬

Cursive

커브가 많이 들어간 서체
필기체를 말함
가독성이 떨어지니 일부분에만 사용하는 것이 좋다

Python, 파이썬

글자 크기

Font size

```
body {  
    font-size: 14px;  
}  
h1 {  
    font-size: 28px(2em);  
}
```

** em은 부모 요소로부터의 비율입니다.

부모(body)의 font-size가 14px이며, h1의 font-size가 2em이면 $14\text{px} \times 2.0 = 28\text{px}$ 이 적용됩니다.

글자 스타일

Font style

```
p {  
    font-style: italic;  
}
```

italic과 oblique는 둘 다 기울임꼴을 나타내지만, italic은 별도의 기울어진 폰트가 있을 경우 해당 폰트를 사용합니다.
inherit은 상위 요소의 font-style을 물려받아 나타냅니다.

글자 굵기

Font weight

```
p {  
  font-weight: bold;  
}  
p {  
  font-weight: 700;  
}
```

lighter
normal
bold
bolder
inherit

<400
400
700
>700

lighter와 bolder는, 해당 서체의 Lighter또는 Bolder서체가 있어야 표현되며,
해당하는 서체가 없을 경우 normal, bold와 동일 굵기로 나타납니다.

줄 간격

Line height

```
p {  
  line-height: 1.5;  
}
```

숫자만 입력할 경우 해당 font-size
에 곱한 값이 줄 간격이 됩니다

** font-size가 px로 고정되어 있다면, line-height에도 고정된 px을 사용할 수 있습니다.

문자 꾸미기

text-decoration

```
p.item1 {  
  text-decoration: none;  
}  
p.item2 {  
  text-decoration: underline;  
}  
p.item3 {  
  text-decoration: overline;  
}  
p.item4 {  
  text-decoration: line-through;  
}
```

밑줄

윗줄

취소선

문자 정렬

text-align

```
p.item1 {  
    text-align: left;  
}  
p.item2 {  
    text-align: center;  
}  
p.item3 {  
    text-align: right;  
}  
p.item4 {  
    text-align: justify;  
}
```

양쪽정렬

** justify는 우측 끝 부분을 깔끔하게 양쪽 정렬해주지만, 줄의 내부 간격이 뒤트리로 잘 사용하지 않습니다.

문자 들여쓰기

text-indent

```
p {  
  text-indent: 1em;  
}
```

-값일 경우, 내어쓰기

대소문자 변환

text-transform

```
p.item1 {  
  text-transform: capitalize;  
}  
p.item2 {  
  text-transform: uppercase;  
}  
p.item3 {  
  text-transform: lowercase;  
}
```

첫 글자만 대문자로

모두 대문자로

모두 소문자로

당연히 영문에만 적용됩니다.

자간

letter-spacing

```
p {  
  letter-spacing: 5px;  
}
```

각 글자간의 간격

단어 간격

word-spacing

글자가 아닌, 단어간의 간격을 조절합니다

```
p {  
  word-spacing: 5px;  
}
```

각 단어간의 간격

요소간 수직 정렬

vertical-align

박스 내에서의 수직 정렬이 아닌, 나란히 오는 인라인 요소간의 정렬

```
<div>
  <strong>baseline : </strong>
  <span class="ori">align</span>
  <span class="vitem item1">Text</span>
</div>
```

```
<div>
  <strong>sub : </strong>
  <span class="ori">align</span>
  <span class="vitem item2">Text</span>
</div>
```

```
<div>
  <strong>super : </strong>
  <span class="ori">align</span>
  <span class="vitem item3">Text</span>
</div>
```

```
<div>
  <strong>top : </strong>
  <span class="ori">align</span>
  <span class="vitem item4">Text</span>
</div>
```

```
<div>
  <strong>text-top : </strong>
  <span class="ori">align</span>
  <span class="vitem item5">Text</span>
</div>
```

```
<div>
  <strong>bottom : </strong>
  <span class="ori">align</span>
  <span class="vitem item6">Text</span>
</div>
```

```
<div>
  <strong>text-bottom : </strong>
  <span class="ori">align</span>
  <span class="vitem item7">Text</span>
</div>
```

```
<div>
  <strong>middle : </strong>
  <span class="ori">align</span>
  <span class="vitem item8">Text</span>
</div>
```

```
.ori {
  font-size: 20px;
}
.vertical-item {
  font-size: 10px;
}
.item1 {
  vertical-align: baseline;
}
.item2 {
  vertical-align: sub;
}
.item3 {
  vertical-align: super;
}
.item4 {
  vertical-align: top;
}
.item5 {
  vertical-align: text-top;
}
.item6 {
  vertical-align: bottom;
}
.item7 {
  vertical-align: text-bottom;
}
.item8 {
  vertical-align: middle;
}
```


줄 바꿈 설정

text-align

```
p.item1 {  
  white-space: nowrap;  
}  
p.item2 {  
  white-space: pre;  
}  
p.item3 {  
  white-space: pre-line;  
}  
p.item4 {  
  white-space: pre-wrap;  
}
```

줄 바꿈이 없습니다

줄 바꿈, 띄어쓰기, 공백등 모든것이 보여지며,
박스를 벗어나도 줄 바꿈이 일어나지 않습니다

줄 바꿈만 보여주며, 띄어쓰기는 무시합니다
자동으로 줄바꿈이 됩니다

줄 바꿈과 띄어쓰기를 모두 보여주며
자동으로 줄바꿈이 됩니다