



# 08.14\_실습 강의자료

## 0교시

자기소개

프로그래밍 경험도

## 1교시

### 1-01 (메아리 - h태그)

#### ▼ <!DOCTYPE html> 의 의미

document type의 약어로 브라우저에게 HTML의 버전 및 웹브라우저 내용을 잘 출력할 수 있도록 도와주는 역할로 DOCTYPE을 선언하지 않으면 각 브라우저의 환경에 따라 html 결과물이 다르게 보일 수 있습니다.

브라우저 별 선언 방법, 호환태그 표

<https://dasima.xyz/doctype-html/>

#### ▼ h태그의 뜻과 사용처는?

<h>태그는 문서의 제목을 표시할 때 사용하는 태그로 heading의 의미를 가짐

<h>태그는 여러 검색엔진이 각 웹사이트의 내용을 h태그를 이용해서 키워드를 수집하고 내용을 파악 (SEO를 위함)

h1~h6까지 계층구조로 사용해야 하며 단순한 글씨 크기를 위한 사용을 지양

한 페이지에 h1태그는 한 번만 사용 - 구글로봇기준

▼ 왜 h6까지있을까요?

원한다면 더 만들 수 있지만 h태그의 본래 목적인 계층구조를 따른다면 그 이상은 거의 사용하지 않는다.

브라우저별 기본 h태그 값 : [https://www.w3schools.com/tags/tag\\_hn.asp](https://www.w3schools.com/tags/tag_hn.asp)

▼ H도 가능한가요?

HTML, CSS는 대소문자를 구분하지 않습니다.

하지만 소문자로 작성하는 것이 관례입니다.

## 1-02 (환영문구 - h태그 + inline style css)

▼ 색깔 만드는 방법은?

총 3가지 방법이 있다.

1. 색 이름으로 표기 ex) green
2. HEX값으로 표기 ex) #f9f9f9
  - a. 가장 많이 사용함
3. RGB값으로 표기 ex) rgb(0,0,255)
  - a. 투명도가 필요하면 rgba를 사용할 수 있다. ex) rgba(0,0,255,0.5)
    - i. 0 : 완전 투명 / 1 : 완전 불투명

▼ 스타일 적용방법은?

총 3가지 방법이 있습니다.

1. Inline Style Sheet
  - a. style적용 최우선순위로 다른거 다 무시하고 적용할 수 있음ex)

```
<h1 style= "color: red">제목</h1>
```

## 2. Internal Style Sheet

- a. header 태그안에 <style>태그를 사용해서 css를 하며 간단하게 테스트 코드를 작성할 때 편리

ex)

```
<head>
...
<style>
  h1 {
    color: orange;
  }
</style>
</head>
<body>
  <h1>제목</h1>
</body>
```

## 3. External Style Sheet

- a. 가장 많이 사용함 (가독성, 유지보수 편리)

ex)

```
/*index.css*/

h1 {
  color: orange;
}
...
```

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

# 2교시

## 1-03 (할로윈 파티 - CSS select )

### ▼ 자식 선택할 때 > 와 그냥 띄어쓰기 차이는?

자손 선택(띄어쓰기) : 하위 모든 태그 안에 해당하는 요소를 선택

자식 선택(>) : 바로 안의 요소들(자식들)중 해당하는 요소를 선택

### ▼ html tag를 선택하는 방법은?

type - X, class - .X, id - #X (우선순위 : id > class > type)

자식 && 자손 선택자 - X Y && X > Y

형제 선택자 - X ~ Y : X뒤부터 Y까지

인접 선택자 - X + Y : X뒤에 바로 Y

순서 선택자 - first-child, last-child, nth-child(n), nth-last-child(n)...

속성 선택자 - X[속성값], X[속성값 = 'vlaue'] : 동일한 값, X[속성값 ~= 'vlaue'] : 한 className으로 포함한 값, X[속성값 \*= 'vlaue'] : 어떤 조합이든 포함한 값

cf.

속성 선택자 참고 : <https://wallel.com/css-속성-선택자-정리-css-attribute-selector/>

반드시 기억해야 하는 CSS 선택자 30가지 :

<https://code.tutsplus.com/ko/tutorials/the-30-css-selectors-you-must-memorize--net-16048>

### ▼ font-weight : 700, font-weight: bold 차이는?

숫자 : 100~900 (normal == 400, bold == 700)

문자 : normal, bold, lighter-부모보다 얇게, bolder-부모보다 두껍게

하지만 폰트별 지원하는 정도에 따라 차이가 있다

<https://developer.mozilla.org/ko/docs/Web/CSS/font-weight>

## 1-04 (무지개 - background-color)

### ▼ background 와 background-color 차이는?

background는 background-color, background-image, background-repeat, background-position을 모두 포함하는 약어로 4가지 속성을 한번에 설정 해줄 수 있습니다.

### ▼ background-image와 img의 차이는?

img태그는 HTML을이용하여 이미지를 넣고

background-image는 CSS를 이용해서 이미지를 넣는 것입니다.

차이점1.

img 태그에 alt를 반드시 넣어주라고 배운 이유는 이미지가 에러났을경우 띄워줄 값을 정하기 위해서입니다.

하지만 background-image에는 기본적으로 노출될 값을 지정하는 것이 없습니다.

차이점2.

프린터시 background-image는 출력되지 않는다

img태그가 SEO, 성능 면에서 더 효율적이다

background-image가 디자인적으로 조작을 할때 더 편리하다

따라서 이미지가 사용자에게 컨텐츠 이해에 도움을 더 준다면 img 태그를

그렇지 않다면 background-image를 사용하는것이 좋습니다.

## 3교시

### 1-05 (CSS사용하기 - CSS의 기본 속성 활용)

#### ▼ font , text 선택기준은?

font는 글자에 대한 디자인적인 속성, text는 글자의 배치와 효과를 설정해줍니다.

▼ font 속성들은?

font-style : 글자 스타일

font-weight : 글자 두께

font-variant: 대소문자 변형 (small-caps)

font-size : 글자 크기

font-family : 글꼴 지정

font : size, family값은 항상 있어야 하며 순서가 중요함(... , size, family 순으로 써야 함)

font설명 : <https://developer.mozilla.org/ko/docs/Web/CSS/font>

▼ text 속성들은?

text-align : 글자의 수평 방향 정렬

text-transform : 대소문자 변경

text-indent : 단락 첫줄 들여쓰기 설정

text-decoration : 글자에 여러가지 효과 설정

text-shadow : 글자에 그림자 효과

font, text 정리된 글 : <https://inpa.tistory.com/entry/CSS-📖-글자-문단-font-text-word>

## 1-06 (Text 덮어쓰기 정렬, float, clear)

▼ 이 많은 CSS를 다 외워야하나요?

아무도 모든 CSS를 외워서 쓰지 않습니다.(뇌 메모리 터져요ㅜ) 반복해서 사용하다보면 자주 사용하는 것들은 저절로 외워지지만 대부분의 CSS속성들은 필요할때 그때그때 검색을 통해서 알아냅니다

▼ float 말고 다른 방법은? (심화) ⇒ 6주차 배웁니다.

레이아웃을 설정하는 방법은 3가지가 있습니다.(Float, Flex, Grid)

float는 초기의 방법으로 현재는 float를 거의 쓰지 않고 Flex와 Grid를 이용하여 레이아웃을 작성합니다.

강의자료에는 Flex와 Grid에 대한 영상이 없지만 레이아웃에 가장 많이 쓰이고 중요한 개념이니 꼭 학습하시길 바랍니다.

## 기본개념

Flex

Flex는 크게 두가지로 개념을 나눌 수 있습니다.

부모가되는Container, 자식이되는 Item이 있습니다.

즉, flex속성은 부모에게 적용되는 속성, 자식에게 적용되는 속성으로 나눌 수 있는데요.

먼저 부모에게 display 속성값을 Flex로 줌으로써 자식요소들을 flex속성을 사용할 수 있습니다.

### 부모에게 적용되는 속성

display: flex; // flex container라는 명시

flex-direction : ; // 배치 방향 설정

flex-wrap : ; // 줄넘김 처리 설정

justify-content: ; // 메인축 방향 설정

align-items: ; // 수직축 방향 설정

align-content: ; // 아이템들이 여러 줄일 때 수직축 방향 설정

...

### 자식에게 적용되는 속성

flex-basis : ; // 아이템의 기본 크기(row일때는 너비, column일때는 높이) 설정

flex-grow : ; // 아이템이 얼마나 커질 수 있는지 설정

flex-shrink : ; // 아이템이 얼마나 작아질 수 있는지 설정  
flex : ; // 위의 3가지 속성의 약어  
align-self : ; // 수직축으로 아이템 정렬 align-items보다 우선순위가 높음  
order : ; // 아이템 순서 설정  
z-index : ; // z축 정렬 설정  
...

등이 있습니다! Grid도 꼭 학습해 보세요~

Flex 자료 : <https://studiomeal.com/archives/197>  
<https://www.youtube.com/watch?v=7neASrWEFEM>  
Grid 자료 : <https://studiomeal.com/archives/533>  
<https://www.youtube.com/watch?v=nxi1EXmPHRs>

## 추가 자료

### ▼ 자식 선택자, 자손 선택자 우선순위

자식 선택자와 자손 선택자는 우선순위가 동일하며 더 나중에 적용하는 style이 기존것을 덮어씌우는 형태입니다.

참고자료 - 우선순위 계산 방법

- a = 인라인 요소
- b = id selector의 수
- c = class selector 나 id를 제외한 attribute selector의 수



- d = 요소의 이름(태그이름)이나 pseudo-요소(ex. :visited)의 수

위의 내용에서  $a > b > c > d$ 로 보고 숫자는 높을 수록 높게 됩니다.

ex)


```
* {} /* a=0 b=0 c=0 d=0 -> specificity = 0,0,0,0
/li {} / a=0 b=0 c=0 d=1 -> specificity = 0,0,0,1
/li:first-line {} / a=0 b=0 c=0 d=2 -> specificity = 0,0,0,2
/ul li {} / a=0 b=0 c=0 d=2 -> specificity = 0,0,0,2
/ul ol+li {} / a=0 b=0 c=0 d=3 -> specificity = 0,0,0,3
/h1 + *[rel=up]{} / a=0 b=0 c=1 d=1 -> specificity = 0,0,1,1
/ul ol li.red {} / a=0 b=0 c=1 d=3 -> specificity = 0,0,1,3
/li.red.level {} / a=0 b=0 c=2 d=1 -> specificity = 0,0,2,1
/#x34y {} / a=0 b=1 c=0 d=0 -> specificity = 0,1,0,0
/style="" / a=1 b=0 c=0 d=0 -> specificity = 1,0,0,0
```

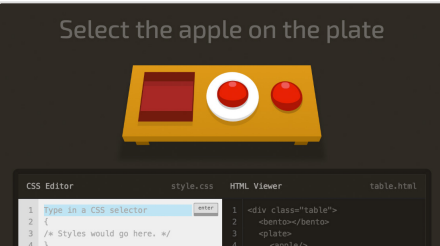
### ▼ css selector 연습하기

아래 링크에서 먼저 연습을 하면 css selector의 느낌을 어느정도 잡으실 수있을 겁니다!!

**CSS Diner**

You're about to learn CSS Selectors! Selectors are how you pick which element to apply styles to. Exhibit 1 - A CSS Rule `p { margin-bottom: 12px; }` Here, the "p" is the selector

 <https://flukeout.github.io/>



Hint)

- tag 선택자 : 태그이름
- class 선택자 : .

- id 선택자 : #
- 선택자 결합 : 붙여쓰기
  - **A.className** : A tag이면서 className을 가지고있는 요소 선택
- 여러개 선택하기 : ,
  - A,B : A, B 둘 다 선택
- 모두선택하기 : \*
- 인접 형제 선택자 : +
  - **A + B** : A와 B가 같은 계층에 있을 때 A 바로 뒤에 B를 선택자로 지정
- 일반 형제 선택자 : ~
  - **A ~ B** : A와 B가 같은 계층에 있을 때 A 뒤에 있는 모든 B를 선택자로 지정
- 자손 선택자 : 띄어쓰기
  - **A B** : A 자손 중 B 선택
- 자식 선택자 : >
  - **A > B** : A 자식 중 B 선택
- 특정 자식 선택자
  - 첫번째 자식 선택자 : :first-child
    - **A :first-child** : A의 첫번째 자식 선택
  - 유일한 자식 선택자 : :only-child
    - **A :only-child** : 자식이 1개인 A의 자식 선택
  - n번째 자식 선택지 : nth-child(n)
    - **A :nth-child(n)** : A의 n번째 자식 선택
- 특정 요소 제외 : :not(A)
  - **A :not(B)**: A 선택자 중 B는 제외
- 타입 첫번째 선택자 : :first-of-type
  - **A :first-of-type** : 첫번째 A 선택
- n번째 유형 선택자 : :nth-of-type( N)
  - 짝수 : even, 홀수 : odd, 몇번째 : n, 수식 : an+b번째 선택자(n은 0부터 시작)

- 자식이 없는 요소 선택자 : :empty
- 속성 선택자 : [ ]
  - a[href] : href속성을 가지고 있는 a태그 모두 선택
  - a[href="https://www.naver.com"] : href="https://www.naver.com"속성을 가지고 있는 a 태그 모두 선택
  - a[href^="https"] : href 값이 "https"으로 시작하는 a 태그 모두 선택
  - a[href\$=".com"] : href 값이 ".com"으로 끝나는 a 태그 모두 선택
  - a[href\*="naver"] : href 값에 "naver"를 포함하는 a 태그 모두 선택

#### ▼ 순서 상관없는 단축 속성


background, border 등이 있습니다.

#### ▼ Flex 연습하기

아래 링크에서 Flex 연습문제들을 풀며 Flex를 학습해보세요!!

**Flexbox Froggy**

A game for learning CSS flexbox

 <https://flexboxfroggy.com/#ko>

**FLEXBOX FROGGY**

Now use **align-items** to help the frogs get to the bottom of the pond. This CSS property aligns items vertically and accepts the following values:

- flex-start**: Items align to the top of the container.
- flex-end**: Items align to the bottom of the container.
- center**: Items align at the vertical center of the container.
- baseline**: Items align at the baseline of the container.
- stretch**: Items are stretched to fit the container.

Level 1  
align-items: flex-end

Submit