

# Vanilla-JS 챌린지 Day 4

## 1. Document

브라우저는 HTML 문서를 로드하기 전에 document 객체를 만든다.

이때, document 객체는 프로퍼티로 HTML 문서의 전반적 속성을 나타내고 이를 통해 자바 스크립트에서 메소드로 HTML 문서의 전반적 제어를 지원한다.

아래의 사진은 document 객체가 가지는 HTML의 프로퍼티를 directory 형식으로 나타낸 것이다.

```
> console.dir(document);  
  
▼ #document  
  URL: "https://codingwanee.tistory.com/entry/JavaScript-%EC%9E%90%EB%B0%94%EC%8A%A4%ED..."  
  ▶ activeElement: body#tt-body-page.thema_black  
  ▶ adoptedStyleSheets: []  
  ▶ alinkColor: ""  
  ▶ all: HTMLAllCollection(726) [html, head, link, link, link, script, script, script, scri...]  
  ▶ anchors: HTMLCollection []  
  ▶ applets: HTMLCollection []  
  ▶ aswift_0: (...)  
  ▶ aswift_1: (...)  
  ▶ baseURI: "https://codingwanee.tistory.com/entry/JavaScript-%EC%9E%90%EB%B0%94%EC%8A%A4%ED..."  
  ▶ bgColor: ""  
  ▶ body: body#tt-body-page.thema_black  
  ▶ characterSet: "UTF-8"  
  ▶ charset: "UTF-8"  
  ▶ childElementCount: 1  
  ▶ childNodes: NodeList(2) [<!DOCTYPE html>, html]  
  ▶ children: HTMLCollection [html]  
  ▶ closure_uid_202482383: 15  
  ▶ compatMode: "CSS1Compat"  
  ▶ contentType: "text/html"  
  ▶ cookie: "__T_=1; __gads=ID=23d8f3e385ddca91-22c5d8160fc50022:T=1607256959:RT=16072569..."  
  ▶ currentScript: null  
  ▶ defaultView: Window {0: global, 1: Window, 2: global, 3: Window, 4: global, 5: global...}  
  ▶ designMode: "off"  
  ▶ dir: ""  
  ▶ doctype: <!DOCTYPE html>  
  ▶ documentElement: html  
  ▶ documentURI: "https://codingwanee.tistory.com/entry/JavaScript-%EC%9E%90%EB%B0%94%EC%8A%A4%ED..."  
  ▶ domain: "codingwanee.tistory.com"  
  ▶ embeds: HTMLCollection []  
  ▶ featurePolicy: FeaturePolicy {}  
  ▶ fgColor: ""  
  ▶ firstChild: <!DOCTYPE html>  
  ▶ firstElementChild: html  
  ▶ fonts: FontFaceSet {onloading: null, onloadingdone: null, onloadingerror: null, ready...
```

사진처럼 HTML문서가 가지는 정보를 : 을 기준으로 왼쪽에 key값으로 오른쪽에 value 값으로 나타내어 보인다.

## 2. 자바스크립트에서 Document Elements 찾기

HTML은 자바스크립트와 연결 되어있어서, 이전 페이지와 같이 document는 자바스크립트에서 HTML의 정보를 가지고 전반적인 제어를 한다.

그런데, 자바스크립트를 통해 HTML을 제어 할 때, 필요한 element만 가져와야 효율적인 제어가 가능할 것이다.

이를 위해 ID 또는 classname 또는 tag를 통해서 특정 element만 찾는 것을 돕는 함수가 있다.

1) document.getElementById()

특정 id를 통해서 필요한 element를 가져온다.

인자에는 “ ” 사이에 목표하려는 element의 id를 적는다.

2) document.getElementsByClassName()

특정 classname을 통해서 필요한 element를 가져온다.

인자에는 “ ” 사이에 목표하려는 element의 classname을 적는다.

3) document.querySelector()

특정 tag를 통해서 필요한 element를 가져온다.

인자에는 목표하려는 element의 tag를 적는다.

## 3. 자바스크립트 Event

자바스크립트는 HTML을 제어함으로써 역동적인 효과를 주기위해 만들어진 언어이다.

이때, HTML에 동적인 효과를 주는 방법중 하나로 event가 있다.

click, window resize, mouseenter 등 여러 가지 event를 통해 HTML을 변화 할 수 있다.

형식은 다음과 같다.

```
function mouse_enter() {  
    title.innerText = "The mouse is here!";  
}  
  
const title = document.querySelector("h1");  
  
title.addEventListener("mouseenter", mouse_enter);
```

이벤트의 종류는 다양하고 이를 통해 우리는 역동적인 웹페이지를 만들 수 있다.