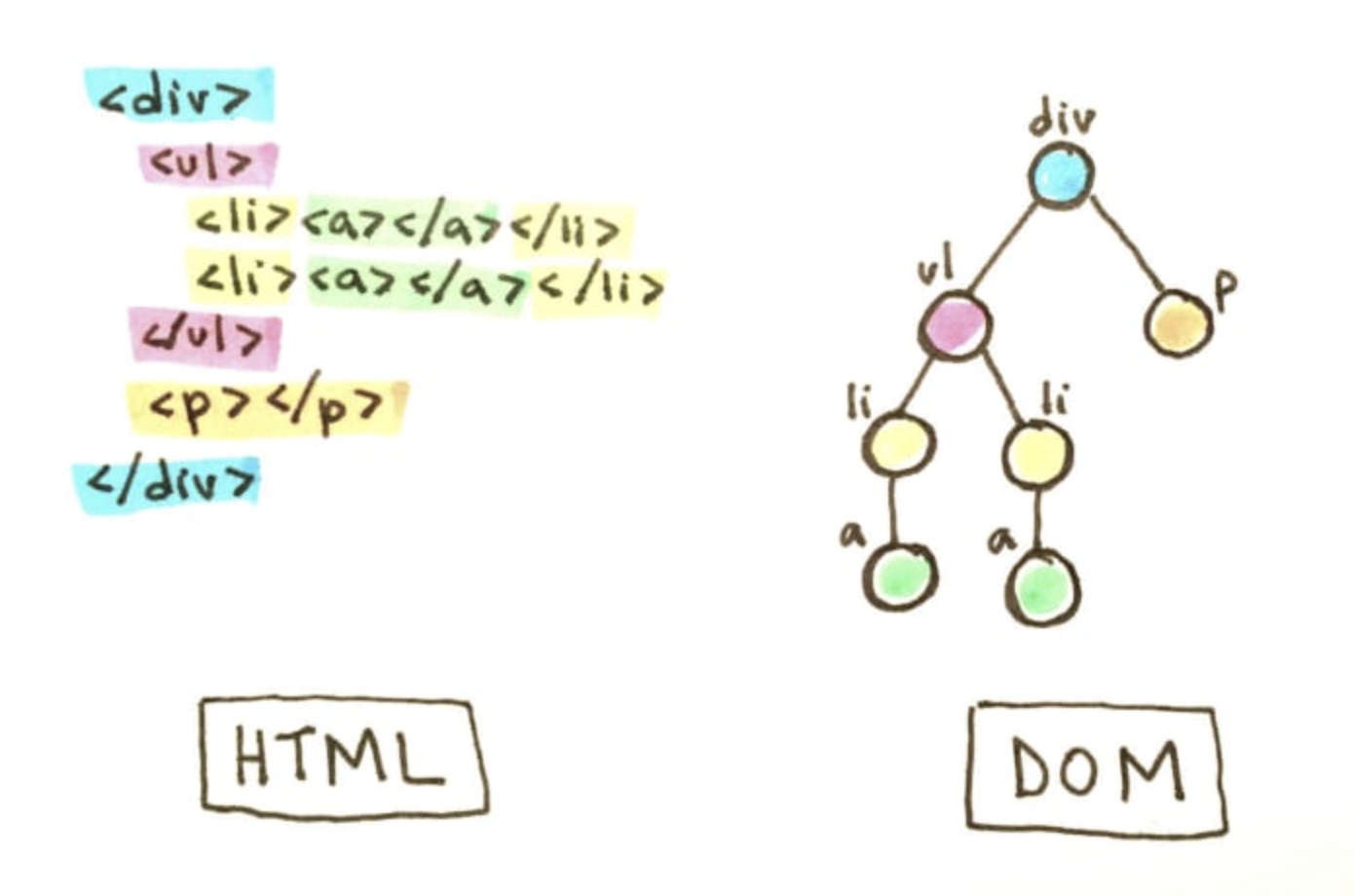
[2023.04.30]-DOM 복습

1. What is DOM?

DOM은 Document Object Model의 약자로, HTML 요소를 JavaScript Object처럼 조작할 수 있는 모델입니다.

DOM은 웹페이지의 콘텐츠 및 구조, 그리고 스타일 요소를 구조화 시켜 표현하여 프로그래밍 언어가 해당 문서에 접근하여 읽고 조작할 수 있도록 API를 제공하는 일종의 인터페이스입니다. 즉 자바스크립트와 같은 스크립팅 언어가 쉽게 웹 페이지에 접근하여 조작할 수 있게끔 연결시켜주는 역할을 담당합니다.

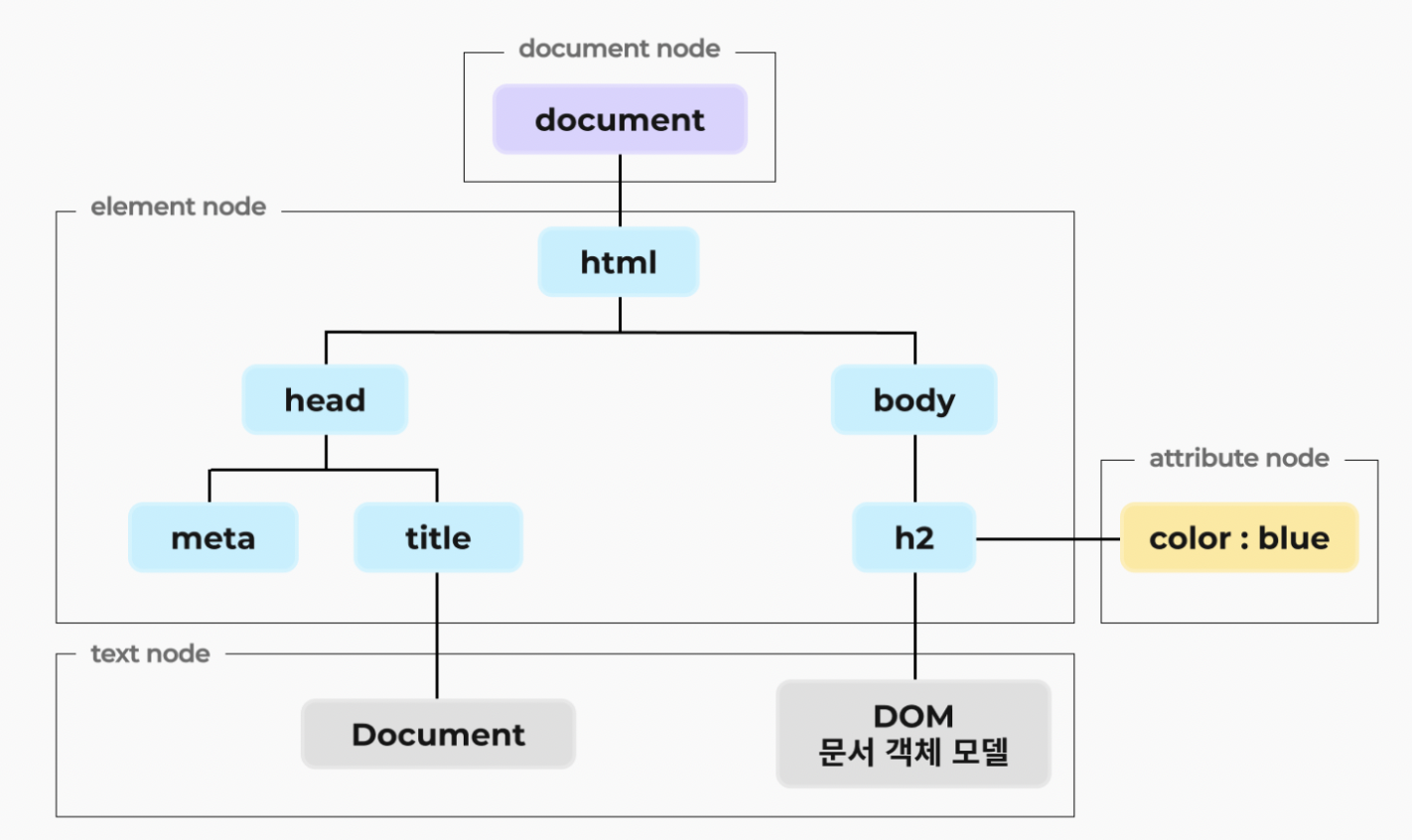
만들고자 하는 웹 사이트가 단순하게 정보의 전달만을 목적으로 하는 정적인 웹이라면 HTML과 CSS만으로도 충분합니다. 하지만 그 이상의 동적인 기능을 구현하려고 한다면 자바스크립트와 DOM이 반드시 필요하게 됩니다.



2. DOM의 구조

DOM은 HTML 문서를 계층적 구조와 정보로 표현하며, 이를 제어할 수 있는 프로퍼티와 메서드를 제공하는 트리 자료 구조입니다. 때문에 HTML DOM Tree라고 부르기도 합니다.

DOM은 트리 자료 구조로 구축되기 때문에 HTML 문서는 최종적으로 하나의 최상위 노드인 root 노드에서 시작해 자식 노드들을 가지며 아래로 뻗어나가는 구조로 만들어지게 됩니다.



3. 주요 노드 타입

* document node (문서 노드)
* element node (요소 노드)
* attribute node (속성 노드)
* text node (텍스트 노드)

**document node**는 Dom Tree에서 최상위 루트를 나타내며, document 객체를 가리킵니다. HTML 문서 전체를 나타내는 노드이기도 합니다. window 객체의 document 프로퍼티로 바인딩되어있어 window.document, document로 참조해 사용할 수 있습니다. HTML 문서에 이 노드는 오로지 1개만 존재합니다.

**element node**는 모든 HTML 요소(h1, p, body, div 등)가 해당하는 노드입니다. 속성 노드를 가질 수 있는 유일한 노드로서, 부모-자식 관계를 가지기에 계층적 구조를 이룰 수 있게 됩니다.

**attribute node**는 모든 HTML 요소의 속성이 해당하는 노드입니다. 요소 노드에 대한 정보를 가지고 있으며, 부모 노드가 아닌 해당 노드와 바인딩되어 있습니다.

**text node**는 HTML 문서의 모든 텍스트입니다. 텍스트 노드는 정보를 표현하며, 가장 마지막에 위치하는 자식 노드이기 때문에 잎사귀를 닮았다고 해서 리프 노드라고 부르기도 합니다.

4. DOM으로 HTML element 다루기

const tweetDiv = document.createElement('div')

위 코드를 통해 tweetDiv 변수에 새롭게 생성한 div element를 변수에 할당합니다.

하지만 아직 브라우저 화면에는 아무런 변화가 생기지 않습니다. tweetDiv에 할당된 div 요소는 현재 아무것도 연결이 되어있지 않은 상태이기 때문입니다. 이 요소를 확인하기 위해서는 APPEND를 통해 트리 구조와 tweetDiv를 연결해주는 과정이 필요합니다.

document.body.append(tweetDiv)

append를 통해 연결해주었지만 아직도 브라우저에는 별다른 변화가 생기지 않습니다. tweetDiv 요소에 아직 아무것도 입력해주지 않았기 때문에 보이는 내용이 없을 뿐, 크롬의 개발자 도구를 통해 살펴보면 잘 연결되어 있는 것을 확인할 수 있습니다.



새로 생성된 div가 잘 확인되는 모습

그렇다면 내용을 어떻게 입력해주면 될까요?

DOM으로 HTML 엘리먼트의 값을 조회하기 위해서는 querySelector을 활용하면 됩니다. 이 때 querySelector의 첫 번째 인자로 선택자를 전달해주어야 하는데, HTML요소와 id, class가 가장 많이 사용됩니다.

const oneTweet = document.querySelector('.tweet')

위 코드를 통해 querySelector로 클래스 이름이 tweet인 HTML 요소를 조회합니다.

위 코드는 변수 oneTweet에 할당된 하나의 요소를 가져오는 것입니다. 여러 개의 요소를 한 번에 가져오기 위해서는 **qureySelectorAll**을 사용합니다.

const tweets = document.querySelectorAll('.tweet')

querySelectorAll을 사용해 클래스 이름이 tweet인 모든 HTML 요소를 조회합니다.

const tweets = document.createElement('div')

tweets.textContent = 'tweet message';

지금까지 DOM을 활용해 요소를 생성하고 DOM 트리 구조에 연결까지 해주었다면 내용을 입력해 줄 차례입니다.

textContent를 사용해 tweets 요소에 내용을 추가해줍니다.

const container = document.querySelector('#container')

const tweetDiv = document.createElement('div')

container.append(tweetDiv)

tweetDiv.remove() *// 이렇게 append 했던 요소를 삭제할 수 있습니다.*

그렇다면 생성된 요소를 제거하려면 어떻게 하면 될까요?

remove 메서드를 사용하면 요소를 제거해 줄 수 있습니다.