

SPOTFIRE PND SUIT GUIDE.

PND SUIT

WEB 가이드

www.pndsolution.com



01 웹 브라우저 역사



02 DOM (Document Object Model) 이란?

1. 정의

문서 객체 모델(The Document Object Model, 이하 DOM) 은 HTML, XML 문서의 프로그래밍 interface 이다.

2. 용도

DOM은 문서의 구조화된 표현(structured representation)을 제공하며 프로그래밍 언어가 DOM 구조에 접근할 수 있는 방법을 제공하여 그들이 문서 구조, 스타일, 내용 등을 변경할 수 있게 돕는다.

3. 구조

nodes와 objects로 구성된다.

4. 발전사항

초창기 DOM은 JS와 밀접하게 연관되어 있었지만 지금은 분리해서 각각 발전되었다.
따라서 페이지 콘텐츠들은 DOM에 저장되고 이를 JS로 접근 및 조작하는 방식이 대표적이다.

03 웹 브라우저에서 HTML문서 렌더링 과정

1. 불러오기

로더(Loader)가 서버로부터 전달 받는 리소스 스트림을 읽는 과정.
읽으면서 어떤 파일인지, 데이터인지 파일을 다운로드할 것인지 등을 결정한다.

2. 파싱 (Phasing)

웹 엔진이 가지고 있는 HTML / XML 파서가 문서를 파싱해서 HTML코드는 DOM Tree , CSS는 CSSOM Tree를 만든다.

03 웹 브라우저에서 HTML문서 렌더링 과정

3. 렌더링 트리 만들기

DOM Tree는 내용을 저장하는 트리로 자바스크립트에서 접근하는 DOM객체를 쓸 때 이용하는 것이고 별도로 그리기 위한 트리가 만들어져 하는데 그것이 렌더링 트리이다.
(그릴 때 필요없는 head,title,body태그등이 없음 + display:none처럼 DOM에는 있지만 화면에서는 걸러내야할 것들을 걸러냄)

4. CSS 결정

CSS는 선택자에 따라서 적용되는 태그가 다르기 때문에 모든 CSS 스타일을 분석해 태그에 스타일 규칙이 적용되게 결정한다.

03 웹 브라우저에서 HTML문서 렌더링 과정

5. 레이아웃

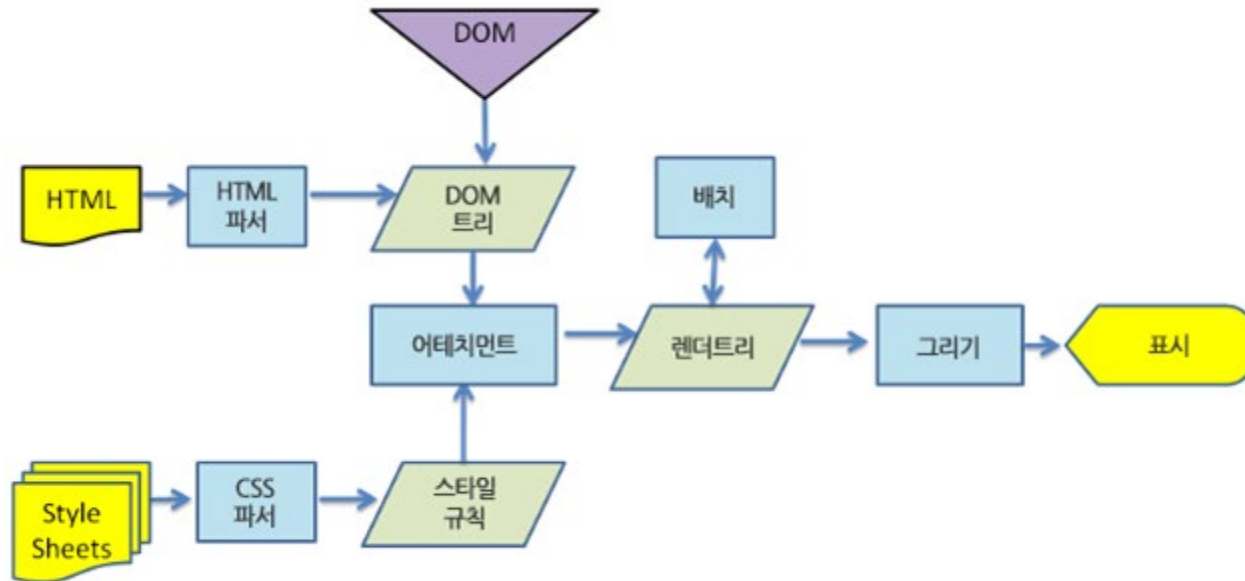
렌더링 트리에서 위치나 크기를 가지고 있지 않기 때문에 객체들에게 위치와 크기를 정해주는 과정

6. 그리기

렌더링 트리를 탐색하면서 그리기.

* 참고로 렌더링 엔진은 가능하면 HTML문서가 파싱 될 때까지 기다리지 않고, 배치와 그리기 과정을 시작한다.

03 웹 브라우저에서 HTML문서 렌더링 과정



04 HTML문서에서 CSS / JAVASCRIPT 위치

1. CSS

문서를 파싱해서 DOM Tree를 만들어도 스타일 규칙이 없으면 렌더링 할 수가 없다.

즉, 최대한 빨리 스타일 규칙을 알아야 렌더링 트리가 완전히 만들어지므로 CSS파일을 모두 다운로드 시키기 위해 `<head></head>` 태그 사이에 놓는 것이다.

2. JAVASCRIPT

자바스크립트는 DOM객체를 이용해서 컴포넌트들을 조작하는데 `<head></head>` 태그에 놓게 되면 HTML 파서가 파싱을 멈추고 스크립트 파일을 읽기 때문에 위에서 스크립트파일이 많거나 파일이 크면 읽는 시간이 오래걸려 사용자 입장에서 웹페이지가 느리게 보이게 되므로 느리다고 느낄 수 있다.
그래서 최대한 맨 밑 `</body>` 태그 위에 스크립트를 모아둔다.

05 PWA (Progressive Web Apps)

1. 정의

PWA는 구글에서 개발하고 적용HTML, CSS, Javascript와 같은 웹 기술로 만드는 앱입니다.
PWA는 네이티브 앱보다 빨리 개발할 수 있으며 푸시 알림이나 오프라인 지원과 같은 네이티브 앱의 특징들을 모두 제공합니다

2. 배경

2015년 구글 크롬의 엔지니어 알렉스 러셀(Alex Russel)이 차세대 웹의 개념,
즉 웹은 웹인데 점진적(progressive)으로 네이티브 앱 수준으로 근접해가는 웹이라는 개념을 처음 제시하였고
그 다음 해인 2016년 Google I/O 개발자 콘퍼런스에서 PWA를 미래의 웹 앱으로 소개했습니다.

05 PWA (Progressive Web Apps)

3. 장점

- 다양한 앱스토어에 출시하기 위해서 별도의 프로세스를 거치지 않아도 된다.
- 일반적인 웹 기술을 활용해서 PWA를 만들 수 있다.
- 기존의 웹사이트를 앱으로 만들 수 있기 때문에, 추가로 유지관리해야 하는 코드베이스(codebase, 소스코드 일체)가 적다.
- PWA는 기본적으로 반응형(responsive)이기 때문에, 다양한 화면 크기에도 잘 동작한다.
- PWA는 부드럽고 빠르며 가볍다.
- 구글이나 애플에게 거액의 돈을 줄 필요가 없다.
- 일반적인 웹사이트와는 달리, 오프라인에서도 작동한다.
- PWA는 검색엔진을 통해서 찾을 수 있다. (앱스토어보다는 검색엔진을 사용하는 사람들이 훨씬 더 많다. 그리고 원한다면 앱스토어를 통해서도 PWA를 배포할 수 있다.)
- 푸시 알림을 통해서 사용자들의 관심을 끌 수 있다.
- PWA를 설치하게 함으로써 보다 높은 참여를 이끌어낼 수 있다.

05 PWA (Progressive Web Apps)

4. PWA를 적용한 기업들

- 스타벅스 - <https://app.starbucks.com/>
- 트위터 - <https://twitter.com/home>
- 핀터레스트 - <https://www.pinterest.co.kr/>