


# 자바스크립트

## 모던 JavaScript 튜토리얼

모던 자바스크립트 튜토리얼은 클로저, 문서 객체 모델, 이벤트, 객체 지향 프로그래밍 등의 다양한 주제에 대한 설명과 예시, 과제를 담고 있습니다.

 <https://ko.javascript.info/>



## Javascript

- Javascript는 브라우저를 동적으로 사용하기 위해 고안된 언어
- Java와 Javascript는 서로 완전히 다른 언어다.

## ES6

- ECMA Script(ES)는 브라우저마다 파편화 되어가는 JS 생태계를 묶기 위해 ECMA가 제시한 표준안
- ES6는 ECMA가 제안하는 JS의 표준 Ver.6 (현재 ES6 를 사용함)

## Vanilla JS

- Vanilla JS : 브라우저 콘솔에서 바로 사용할 수 있는 JS
- Vanilla JS는 프로그래밍을 통해 BOM 조작, DOM 구조를 조작할 수 있다.

## Vanilla JS vs Node.js

- 브라우저라는 제한된 환경을 넘어 브라우저 밖(컴퓨터)에서 JS를 구동할 수 있는 새로운 실행 환경인 `Node.js`가 등장한다.
- Vanilla JS
  - 브라우저 콘솔 환경에서 사용하는 가장 순정의 JS를 의미
  - 어떠한 라이브러리/프레임워크에도 의존하지 않는 Original Browser JS
  - BOM(Browser Object Model) 조작, DOM(Document Object Model) 조작, 프로그래밍 가능
- Node.js
  - 브라우저 밖에서 JS를 구동할 수 있는 새로운 JS 실행환경(runtime environment)인 `Node.js`가 등장

- V8 엔진을 기반으로 제작되었으며, Node.js의 등장을 통해 JS 만으로 클라이언트-서버 를 모두 제작 가능

#### V8 엔진(자바스크립트, NodeJS, Deno, WebAssembly) 내부의 메모리 관리 시각화하기

이번 장에서는 NodeJS, Deno와 Electron과 같은 런타임 및 Chrome, Chromium, Brave, Opera, Microsoft Edge와 같은 웹 브라우저에서 사용되는 ECMAScript와 WebAssembly를 위한 V8 엔진의 메모리 관리 방법에 대해 살펴볼 것이다.

 [https://ui.toast.com/weekly-pick/ko\\_20200228](https://ui.toast.com/weekly-pick/ko_20200228)

