

# 02장. 자바스크립트란?

## 2.1 자바스크립트의 탄생

- 자바스크립트란?

브레덴 아이크가 웹페이지의 보조적인 기능을 수행하기 위해 브라우저에서 동작하는 경량 프로그래밍 언어를 도입 한 언어

모카 → 라이브스크립트 → 자바스크립트

현재 모든 브라우저의 표준 프로그래밍 언어로 자리 잡았다.

## 2.2 자바스크립트의 표준화

- 크로스 브라우징 이슈란?

브라우저에 따라 웹페이지가 정상적으로 동작하지 않았다.

## 2.3 자바스크립트 성장의 역사

- 렌더링이란?

HTML, CSS, 자바스크립트로 작성된 문서를 해석해서 브라우저에 시각적으로 출력하는 것  
서버에서 데이터를 HTML로 변환해서 브라우저에게 전달하는과정(SSR: server side rendering)을 말하기도 함

- Ajax란?

자바스크립트를 이용해 서버와 브라우저가 비동기 방식으로 데이터를 교환할 수 있는 통신 기능

[전]

## 1. 방식

HTML 태그 → HTML 태그로 끝나는 완전한 HTML 코드를 서버로부터 전송받아 웹페이지 전체를 렌더링. 즉, 화면이 전환되면 서버로부터 새로운 HTML을 전송받아 웹페이지 전체를 처음부터 다시 렌더링

## 2. 단점

변경할 필요가 없는 부분까지 포함된 HTML 코드를 서버로부터 다시 전송 받아야 함, 불필요한 데이터 통신 발생, 성능 면에서 떨어짐, 화면이 전환되면 순간적으로 깜박이는 현상 발생

[후]

## 1. 방식

서버로부터 필요한 데이터만 전송받아 변경해야 하는 부분만 한정적으로 렌더링하는 방식이 가능

## 2. 장점

웹 브라우저에서도 데스크톱 애플리케이션과 유사한 빠른 성능과 부드러운 화면 전환 가능

### • JQuery란?

jQuery 등장으로 다소 번거롭고 논란이 있던 DOM을 더욱 쉽게 제어 가능 → 크로스 브라우저 이슈도 어느정도 해결

### • V8란?

자바스크립트로 웹 애플리케이션을 구축하려는 시도가 늘면서 빠른 성능을 보여줌

V8의 등장으로 자바스크립트는 데스크톱 애플리케이션과 유사한 사용자 경험을 제공할 수 있는 웹 애플리케이션 프로그래밍 언어로 정착, 웹 애플리케이션 개발에서 프론트엔드 영역이 주목받는 계기로 작용

### • Node.js란?

1. 라이언 달이 발표한 구글 V8 자바스크립트 엔진으로 빌드된 자바스크립트 런타임 환경

2. 브라우저의 자바스크립트 엔진에서만 동작하던 자바스크립트를 브라우저 이외의 환경에서도 동작할 수 있도록 자바스크립트 엔진을 브라우저에서 독립시킨 자바스크립트 실행 환경

3. 주로 개발에 사용되며, 필요한 모듈, 파일 시스템, HTTP 등 빌트인 API를 제공

4. 프론트엔드와 백엔드 영역에서 자바스크립트를 사용할 수 있다는 동형성은 별도의 언어를 학습하기 위한 시간을 덜 수 있다는 장점
5. 비동기 I/O를 지원하며 단일 스레드 이벤트 루프 기반으로 동작함으로써 요청처리 성능이 좋음  
→ 실시간으로 처리하기 위해 I/O가 빈번하게 발생하는 SPA(단일 페이지)에 적합

- **SPA 프레임워크란?**

CBD방법론을 기반으로 하는 SPA가 대중화되면서 Angular, React, Vue.js, Svelte 등 다양한 SPA 프레임워크/라이브러리 사용층을 확보

## 2.4 자바스크립트와 ECMAScript

자바스크립트의 표준 사양인 ECMA-262를 말하며, 프로그래밍 언어의 값, 타입, 객체와 프로퍼티, 함수, 표준 빌트인 객체 등 핵심 문법을 규정

## 2.5 자바스크립트의 특징

1. 웹 브라우저에서 동작하는 유일한 프로그래밍 언어, 개발자가 별도의 컴파일 작업을 수행하지 않는 인터프리터 언어
2. 명령형, 함수형, 프로토타입 기반 객체지향 프로그래밍을 지원하는 멀티 패러다임 프로그래밍 언어
3. 클래스 기반 객체지향 언어보다 효율적이면서 강력한 프로토타입기반의 객체지향언어