

## 2장

- 자바스크립트의 초반에는 각 회사들이 시장 점유율을 높이기 위해 자사 브라우저에서만 동작하는 기능을 추가하기 시작했다. 이로 인해 웹페이지가 브라우저에 따라 동작하지 않는 **크로스 브라우징** 이슈가 발생 했다. 결과적으로는 모든 브라우저에서 동작하는 웹 페이지를 구현하기가 어려워졌다.
- ECMA 인터내셔널에서 자바스크립트의 표준화 요청으로 자바스크립트는 **ECMAScript**로 명명되었다.
- 렌더링이란 HTML, CSS, 자바스크립트로 작성된 문서를 브라우저에 시각적으로 출력하는 것.
- 정적 웹페이지(HTML, CSS, JS)
  - Static Rendering 방식
  - 요청마다 서버에서 새로 조립하지 않음(HTML 재사용)
- SSR
  - 서버에서 데이터를 HTML로 변환해서 브라우저에게 전달하는 과정
  - 정적 웹페이지는 SSR은 아니지만 유사한 방식으로 렌더링 된다.
  - 요청 시점마다 동적으로 렌더링(새로운 HTML)
- Ajax - 비동기 통신 기능
  - 서버와 브라우저가 비동기 방식으로 데이터를 교환
  - 변경할 필요가 없는 부분은 다시 렌더링하지 않음
  - 필요한 데이터만 전송받아 변경
  - 부드러운 화면 전환
- Node.js
  - 브라우저 이외의 환경에서도 동작
  - 서버 사이드 애플리케이션 개발에 주로 사용
    - ▼ 서버 사이드란  
서버에서 돌아가며 클라이언트 요청을 처리하는 프로그램 전체, 백엔드 애플리케이션과 같은 의미

- 데이터를 실시간으로 처리하기 위해 비동기가 빈번하게 발생하는 SPA(single page application)에 적합
- 자바스크립트
  - 웹 브라우저에서 동작하는 유일한 프로그래밍 언어
  - 별도의 컴파일 작업을 수행하지 않는 인터프리터 언어
  - 모던 자바스크립트 엔진은 인터프리터와 컴파일러의 장점을 결합
  - 런타임에 컴파일되며 실행 파일이 생성되지 않고 인터프리터의 도움 없이 실행할 수 없기 때문에 컴파일러 언어라고 할 수는 없다.
  - 객체지향 프로그래밍