

Ajax란

Asynchronous JavaScript and XML의 약자이다.

빠르게 동작하는 동적인 웹 페이지를 만들기 위한 개발 기법의 하나.

웹 페이지 전체를 다시 로딩하지 않고도, 웹 페이지의 일부분만을 갱신할 수 있다.

Ajax를 이용하면 백그라운드 영역에서 서버와 통신하여, 그 결과를 웹 페이지의 일부분에만 표시할 수 있다.

Ajax의 장점

Ajax를 이용하면 다음과 같은 장점이 있습니다.

1. 웹 페이지 전체를 다시 로딩하지 않고도, 웹 페이지의 일부분만을 갱신할 수 있습니다.
2. 웹 페이지가 로드된 후에 서버로 데이터 요청을 보낼 수 있습니다.
3. 웹 페이지가 로드된 후에 서버로부터 데이터를 받을 수 있습니다.
4. 백그라운드 영역에서 서버로 데이터를 보낼 수 있습니다.

Ajax의 한계

1. Ajax는 클라이언트가 서버에 데이터를 요청하는 클라이언트 폴링 방식을 사용하므로, 서버 푸시 방식의 실시간 서비스는 만들 수 없습니다.
2. Ajax로는 바이너리 데이터를 보내거나 받을 수 없습니다.
3. Ajax 스크립트가 포함된 서버가 아닌 다른 서버로 Ajax 요청을 보낼 수는 없습니다.
4. 클라이언트의 PC로 Ajax 요청을 보낼 수는 없습니다.

jQuery

제이쿼리는 오픈 소스 기반의 자바스크립트 라이브러리이다.

제이쿼리를 이용하면 문서 객체 모델(DOM)과 이벤트에 관한 처리를 손쉽게 구현할 수 있다
Ajax 응용 프로그램 및 플러그인도 제이쿼리를 활용하여 빠르게 개발할 수 있다.

V8 엔진

웹 브라우저를 만드는 데 기반을 제공하는 오픈 소스 자바스크립트 엔진이다.

구글 크롬 브라우저와 안드로이드 브라우저에 탑재되어 있다.

ECMAScript(ECMA - 262) 3rd Edition 규격의 C++로 작성되었으며, 독립적으로 실행이 가능하다. 또한 C++로 작성된 응용 프로그램의 일부로 작동할 수 있다.

V8은 자바스크립트를 바이트코드(bytecode)로 컴파일하고 실행하는 방식을 사용한다.

Nodejs

오픈소스 JavaScript 엔진인 크롬 v8에 비동기 이벤트 처리 라이브러리인 libuv를 결합한 플랫폼이다. -> 뭔 소린지 모르겠다.

다시 말해, JavaScript로 브라우저 밖에서 서버를 구축하는 등의 코드를 실행할 수 있게 해주는 런타임 환경이다. 처음엔 리눅스와 macos만 지원되었으나 2011 7월에 windows버전도 발표됨.

빈번한 I/O 처리에 있어서의 우수한 성능, 서버 확장의 용이성, 무엇보다도 JavaScript라는 프론트엔드 필수 언어로 백엔드까지 작성할 수 있다는 엄청난 장거임 때문에 출시 이후로 빠르게 점유율을 높여가고 있다. 특히 넷플릭스처럼 엄청나게 많은 양의 I/O데이터를 처리해야 하는 서비스에 있어서 강점이 두드러진다.

SPA 프레임워크

SPA는 Single Page Application의 약자로 단일 페이지로 구성된 웹 애플리케이션을 말한다. SPA가 등장하기 전 웹 애플리케이션을 구성하던 방식인 서버 사이드 렌더링(SSR)의 경우,

화면에 보여질 리소스를 서버로 요청하고, 서버로 부터 받아온 리소스를 렌더링 했다.

하지만 SPA의 경우 렌더링의 역할을 서버에게 넘기지 않고 브라우저에서 처리하는 방식이다. 웹 애플리케이션에 필요한 모든 정적 리소스를 최초에 한번 다운로드 하고, 이후 새로운 페이지 요청 시 페이지 갱신에 필요한 데이터만을 전달받아 페이지를 갱신하게 된다

SPA의 장점

새로운 페이지 요청 시 전체를 렌더링 하지 않고 변경되는 부분만 갱신하기 때문에 전체적인 트래픽 감소와 렌더링에서 좋은 효율을 가진다.

새롭게 갱신되는 부분만 렌더하기 때문에 새로고침이 발생하지 않아 화면 깜빡임 없이 빠른 화면 이동이 가능하다.

트래픽의 감소와 속도,반응성의 향상으로 인해 앱처럼 자연스러운 사용자 경험(UX)를 제공하며 모바일 사용이 증가하는 시점에 이는 큰 장점이 된다.

모듈화 또는 컴포넌트별 개발이 용이하다.

백엔드와 프론트엔드가 비교적 명확하게 구분된다.

SPA의 단점

웹 애플리케이션에 필요한 정적 리소스를 한번에 다운로드 하기 때문에 초기 구동 속도가 느리다.

SPA 구조 상 데이터 처리를 클라이언트에서 하는 경우가 많은데, 해당 로직들은 JavaScript를 통해 구현되므로 코드가 외부에 노출되는 보안적인 문제가 있다.

검색엔진 최적화(SEO)가 어렵다.

// 각각의 console.log의 출력 값을 알아오시오.

```
console.log(a);
```

```
var a;
```

```
a= 11;
```

```
console.log(a);
```

답

undefined

11

// 각각의 console.log의 출력 값을 알아오시오.

```
console.log(a);
```

```
a = 10;
```

```
var a;
```

```
console.log(a);
```

답

11

10

헝가리안 케이스

프로그래밍 언어에서 변수 및 함수의 인자 이름 앞에 데이터 타입을 명시하는 코딩 규칙

헝가리인 찰스 시모니가 ms사의 개발책임자 재직 당시 제안

최근에는 헝가리안 표기법을 쓰지 않는 추세이다.