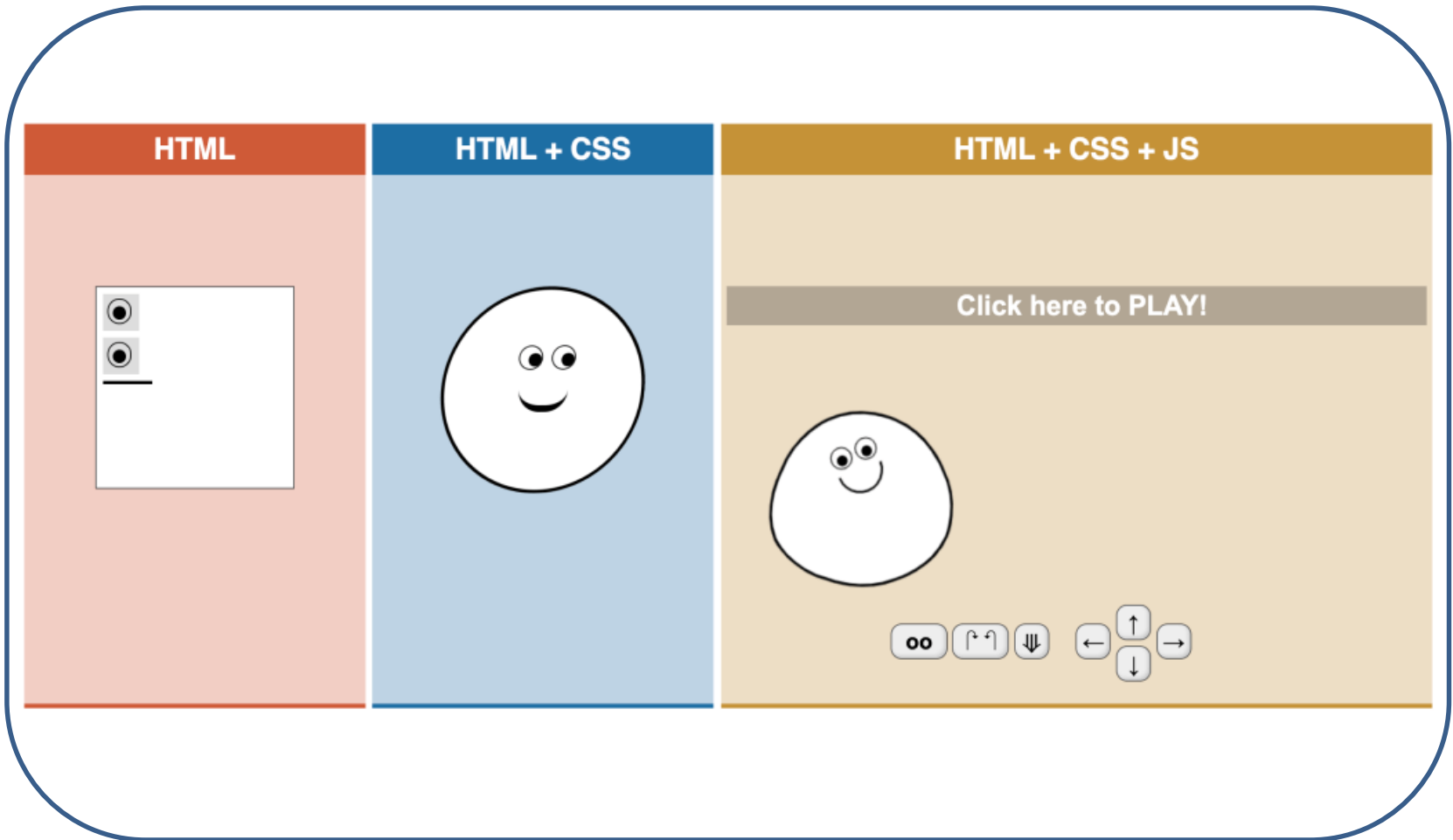


JAVASCRIPT

JavaScript - 개요

Javascript란



JavaScript - 개요

Javascript란

MS :p

up.do

Please

localhost:8080 내용:

아이디

아이디를 입력해 주세요

확인

비밀번호

비밀번호

비밀번호 확인

비밀번호 확인

이름

이름

이메일

이메일

주소

우편번호

우편번호 찾기

도로명 주소

상세주소

회원가입

아이디

ring333

비밀번호

....

비밀번호 확인

....

이름

순남2

이름을 확인해주세요

생년월일

19991232

생년월일을 확인해주세요 :)

이메일

ringpoo@naver.com

이메일이 가입된 이메일입니다 :p

휴대전화 (-~ 없이 번호만 입력해주세요)

01071283377

❌ 취소하기

♥ 가입하기

JavaScript - 개요

1993 - Mosaic

1994 - Netscape Navigator



Marc Andreessen(1971. 07 .09 ~
미국 위스콘신주 출생
최초의 GUI 브라우저, Mosaic 개발

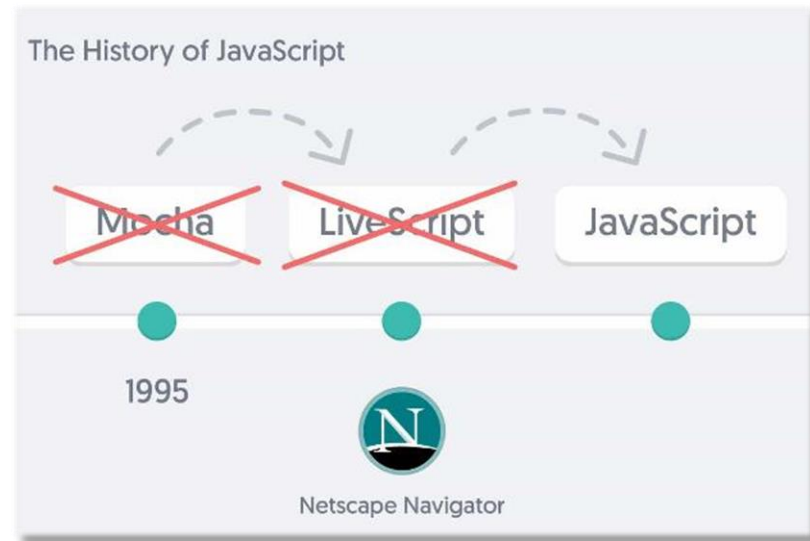


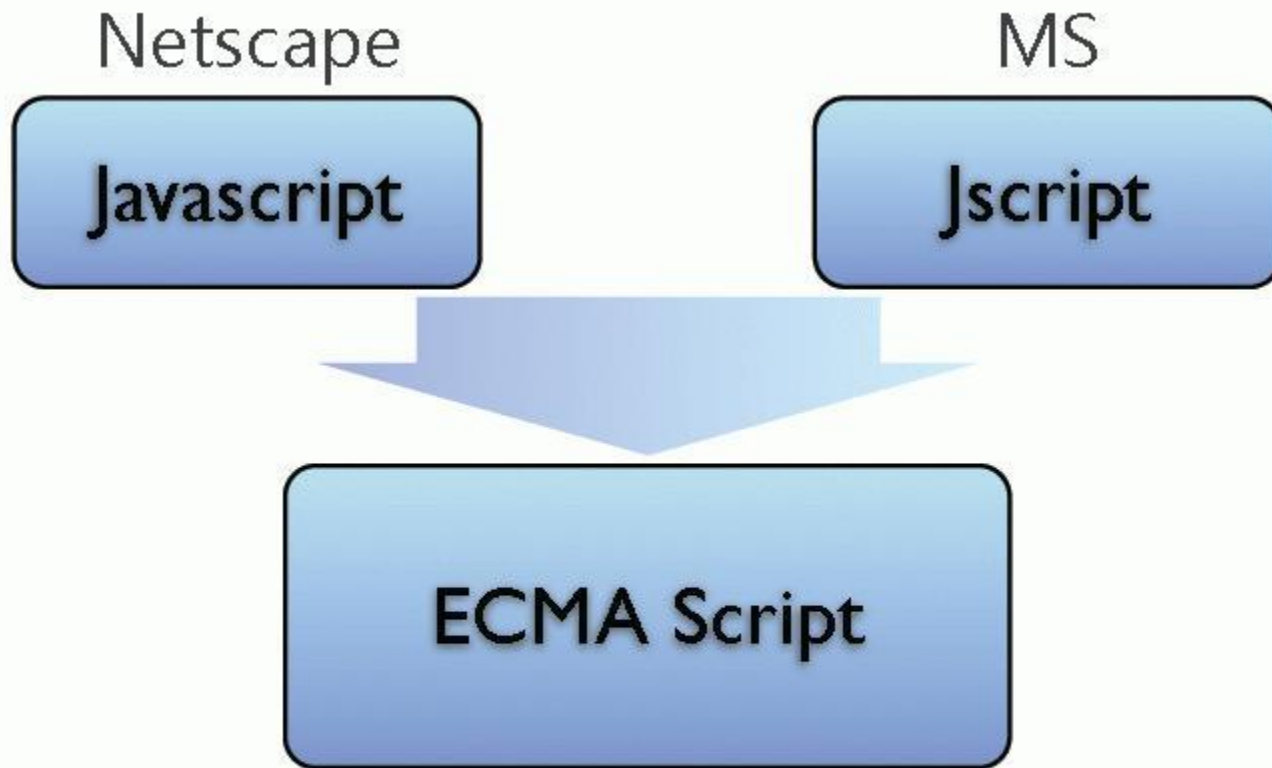
JavaScript - 개요

1995 - Netscape Navigator &
JavaScript-interpreter



Brendan Eich(1961. 07 .04 ~
미국 펜실베이니아주 출생
자바스크립트의 창조자





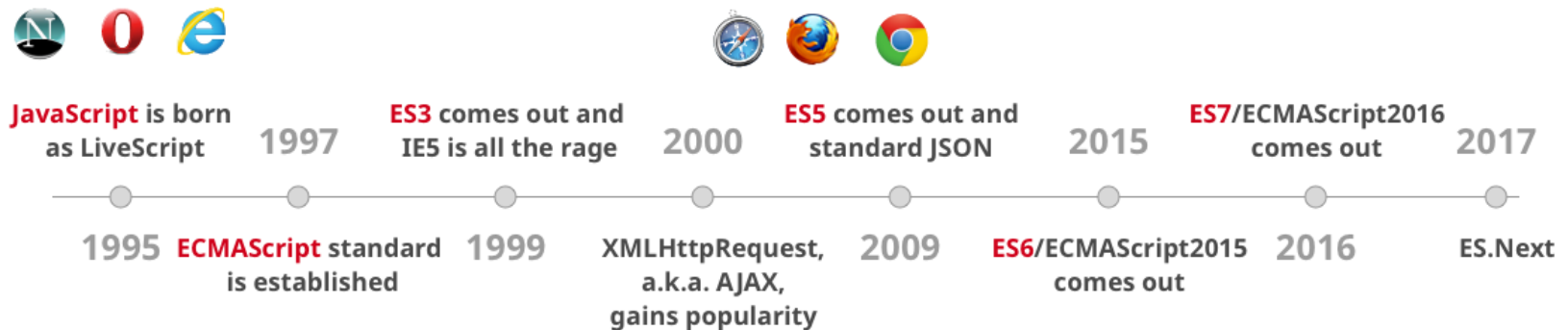
ECMA : Europe Computer Manufacturer Association

Javascript란

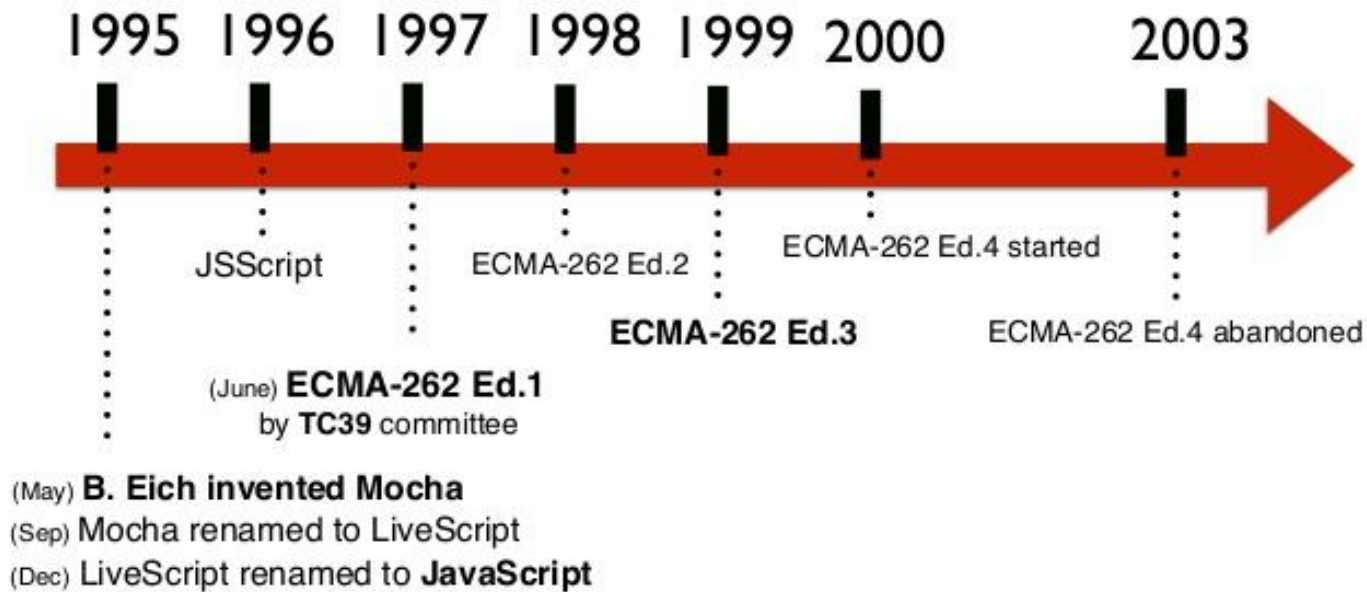
자바스크립트(JavaScript)는 웹 브라우저에서 많이 사용하는 인터프리터 방식의 객체지향 프로그래밍 언어이다.

자바스크립트는 ECMA 스크립트 표준을 따르는 대표적인 웹 기술이다.

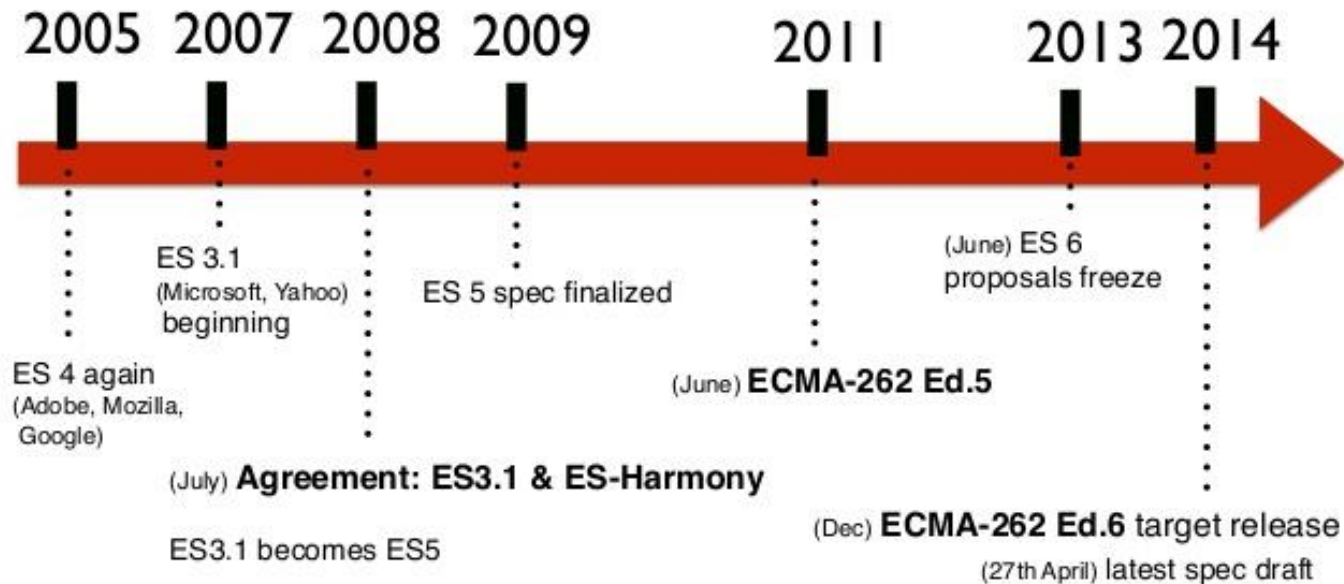
☞ ECMA(European Computer Manufacturers Association) : 표준화기구

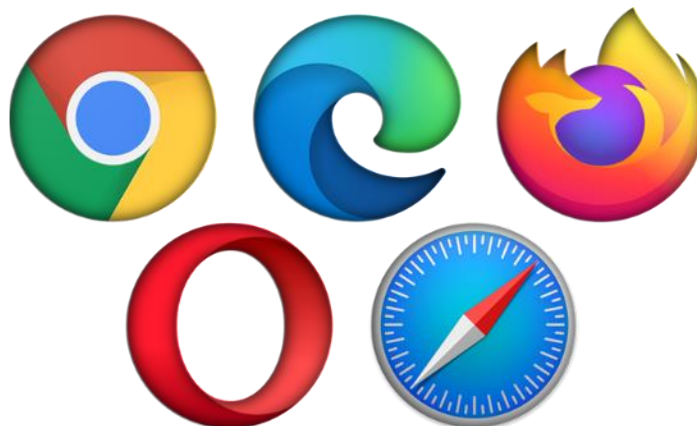


History



History





JS 사용법

브라우저 개발자 도구

- 브라우저별 개발자 도구(Develop Tools)가 있음
- 크롬/ IE 브라우저 **F12**키 눌러서 실행
- 브라우저 창에서 원하는 코드를 확인하고 싶으면 원하는 위치에서 우클릭 후 해당메뉴선택 (크롬 : 검사 / IE : 요소검사)
- 자바스크립트 소스코드 중 `console.log()`는 브라우저에 출력하는 것이 아니라 개발자 도구의 console패널에서 출력
- 스크립트 구문을 디버깅할때 자주 사용

자바스크립트 선언

- <SCRIPT></SCRIPT> 태그 사이에 자바스크립트 코드 문장을 작성하면 된다.
- HTML에서 제공하는 <script></script>태그를 사용하여, 자바 스크립트 작성영역을 설정
- type속성 브라우저 호환성을 위해 사용되나 default값으로 생략이 가능

<script type="text/javascript">

자바스크립트 내용

</script>

- 📖 language속성과 charset속성이 있었지만 language속성은 폐기 되고 charset속성은 meta태그가 적용되기 때문에 사용할 필요가 없음.

자바스크립트 위치

- <SCRIPT></SCRIPT>는 <head>,<body>안 어느 영역에나 작성 가능
- html태그 영역 밖에 작성 가능
하지만, 웹 표준과 웹 접근성을 고려해서 <head>나 <body>에 작성

<script>자바스크립트 내용</script>

<html>

<head><script>자바스크립트 내용</script></head>

<body><script>자바스크립트 내용</script></body>

</html>

<script>자바스크립트 내용</script>

자바스크립트 작성 방식

- inline 방식 : 자바스크립트 양이 한두 줄 정도로 소량일 때 사용, 태그에 이벤트 핸들러 속성을 이용하여 직접 실행 코드 작성
- internal 방식 : 가장 일반적인 방식, html파일 내 자바스크립트 소스를 작성(<head>,<body>)
- external 방식 : 자바스크립트의 양이 많은 경우 자바스크립트 코드 부분을 외부 파일로 저장하여 작성
<script src="경로">태그를 이용하여 내용을 삽입 후 사용

inline 방식

html 태그의 on이벤트 속성을 이용하여 내장 메소드를 호출하거나 개발자가 선언한 사용자정의 함수를 호출할때 사용

〈태그명 on이벤트 = “함수명();”〉

internal 방식

자바스크립트코드를 작성, 함수 단위로 소스코드를 작성하고 html태그에서 이벤트 핸들러를 통해 함수를 실행시키는 방식

〈script〉

실행할 소스코드 작성

〈/script〉

external 방식

외부에 별도의 자바스크립트 소스파일(*.js)을 작성하고 html에서 <script src="경로.js">태그를 이용하여 해당파일을 불러와 사용하는 방식

☞ 여러 개 html파일에서 공통적으로 사용하는 기능일 경우 사용

```
<script src="경로"></script>
```

〈noscript〉 태그

- 자바 스크립트가 지원되지 않는 브라우저를 대비 지원하지 않는 경우 〈noscript〉에 작성된 내용이 출력

〈noscript〉

지원하지 않을 경우 출력문구

〈/noscript〉

자바스크립트 실행방식

웹 브라우저에 내장되어 있는 자바스크립트 파서가 소스 코드를 한 줄씩 읽고 해석한다는 점에서 인터프리터 방식으로 소스를 해석하기 때문에 전체를 해석해 놓은 컴파일 언어와는 차이가 있다.

자바스크립트 실행은, 작성된 html 문서를 브라우저에서 읽어 들이면 바로 실행을 확인할 수 있다는 것이다.

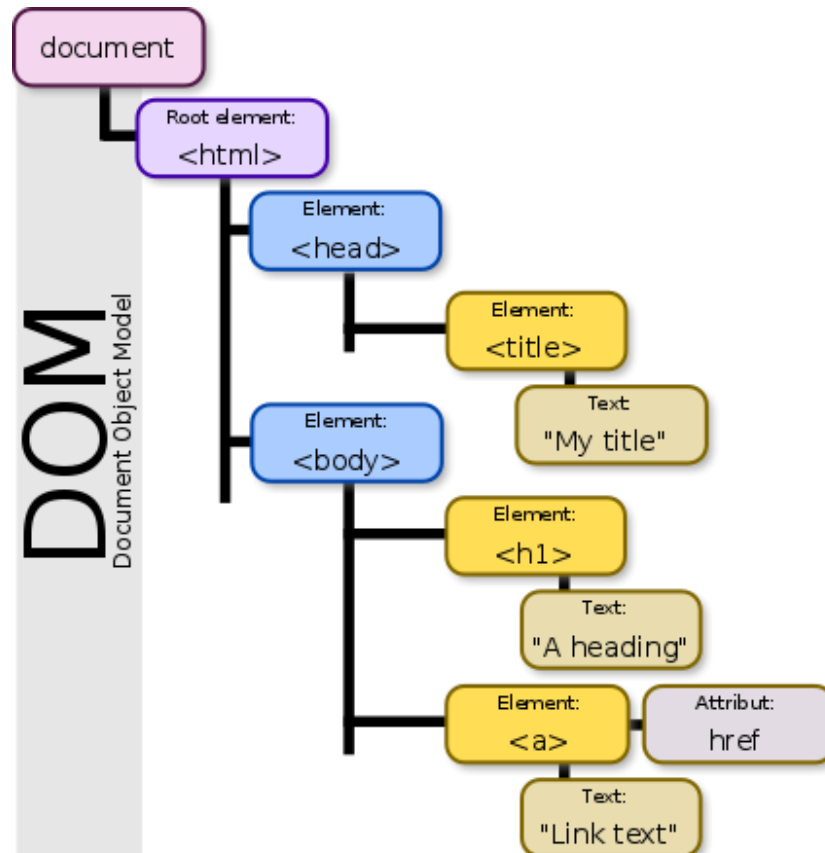
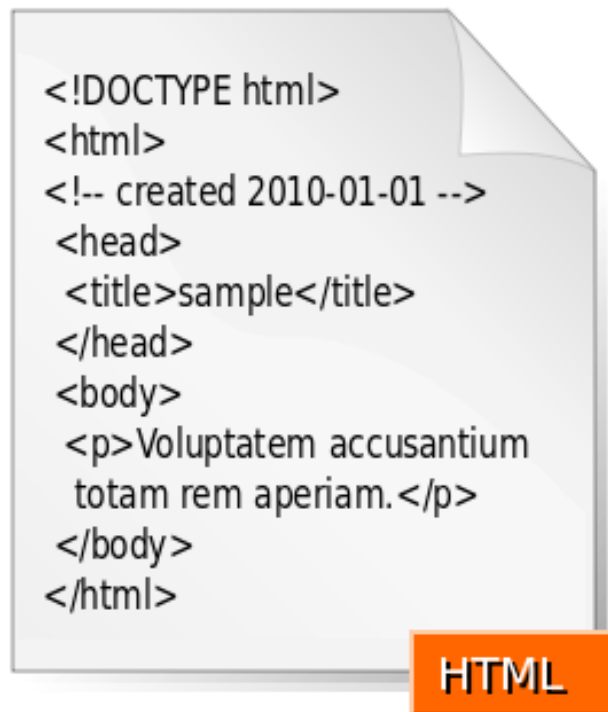
JS DOM

DOM(Document Object Model)

- 문서 객체 모델(The Document Object Model, 이하 DOM)은 HTML, XML 문서의 프로그래밍 interface 이다.
- 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.
- DOM은 구조화된 nodes와 property와 method를 갖고 있는 objects로 문서를 표현한다. 이들은 웹 페이지를 스크립트 또는 프로그래밍 언어들에서 사용될 수 있게 연결시켜주는 역할을 담당한다.

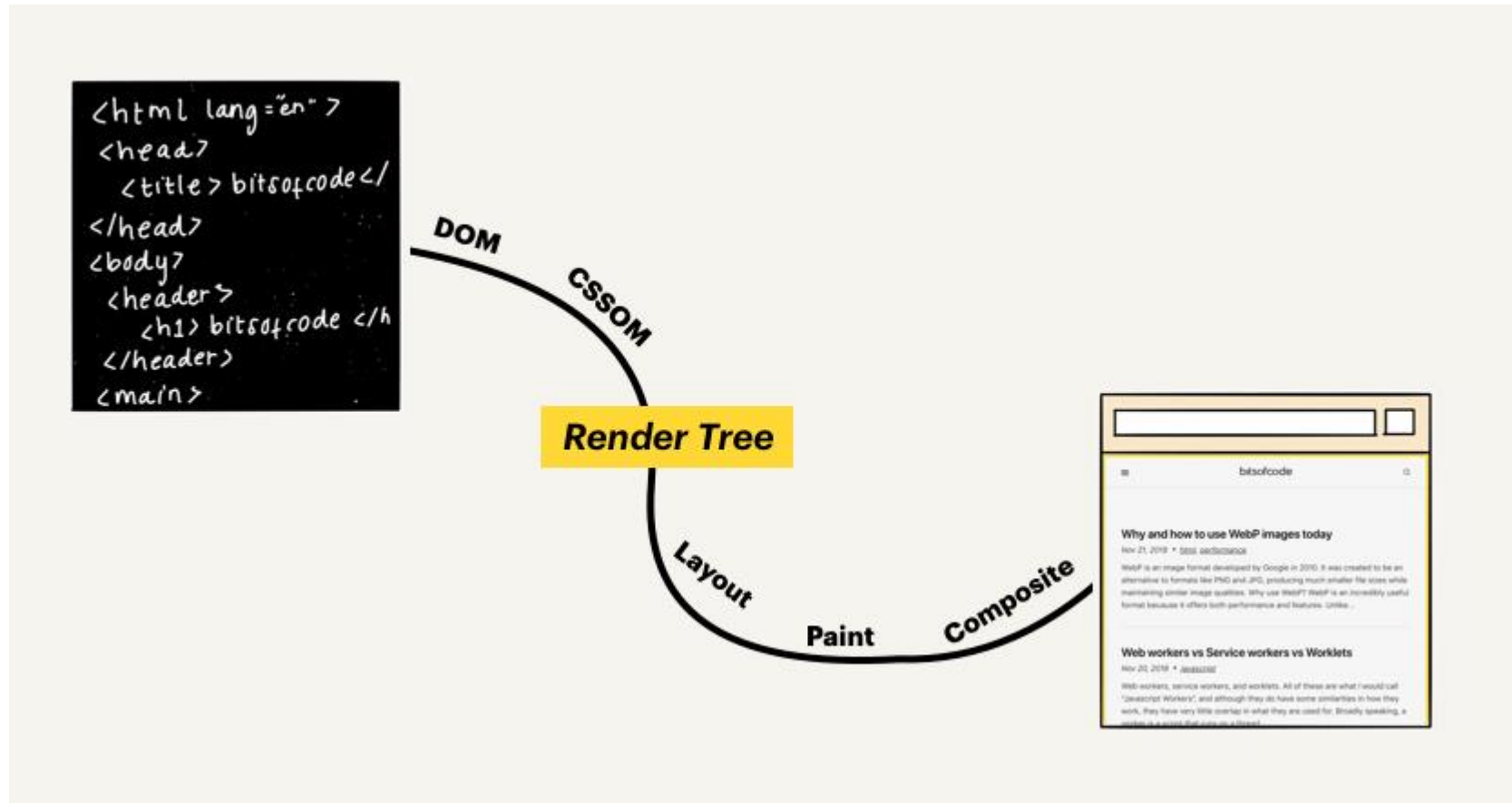
Javascript DOM

DOM(Document Object Model)



JavaScript DOM

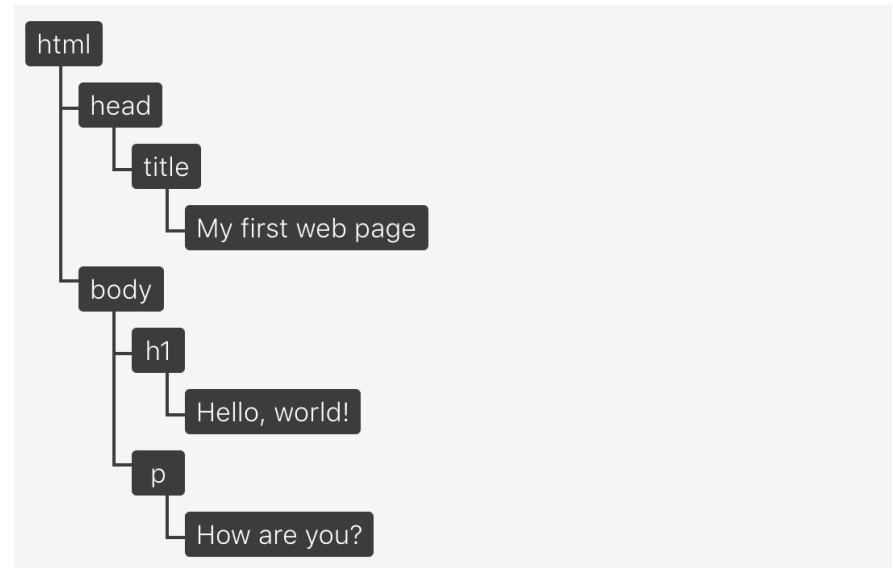
DOM(Document Object Model)



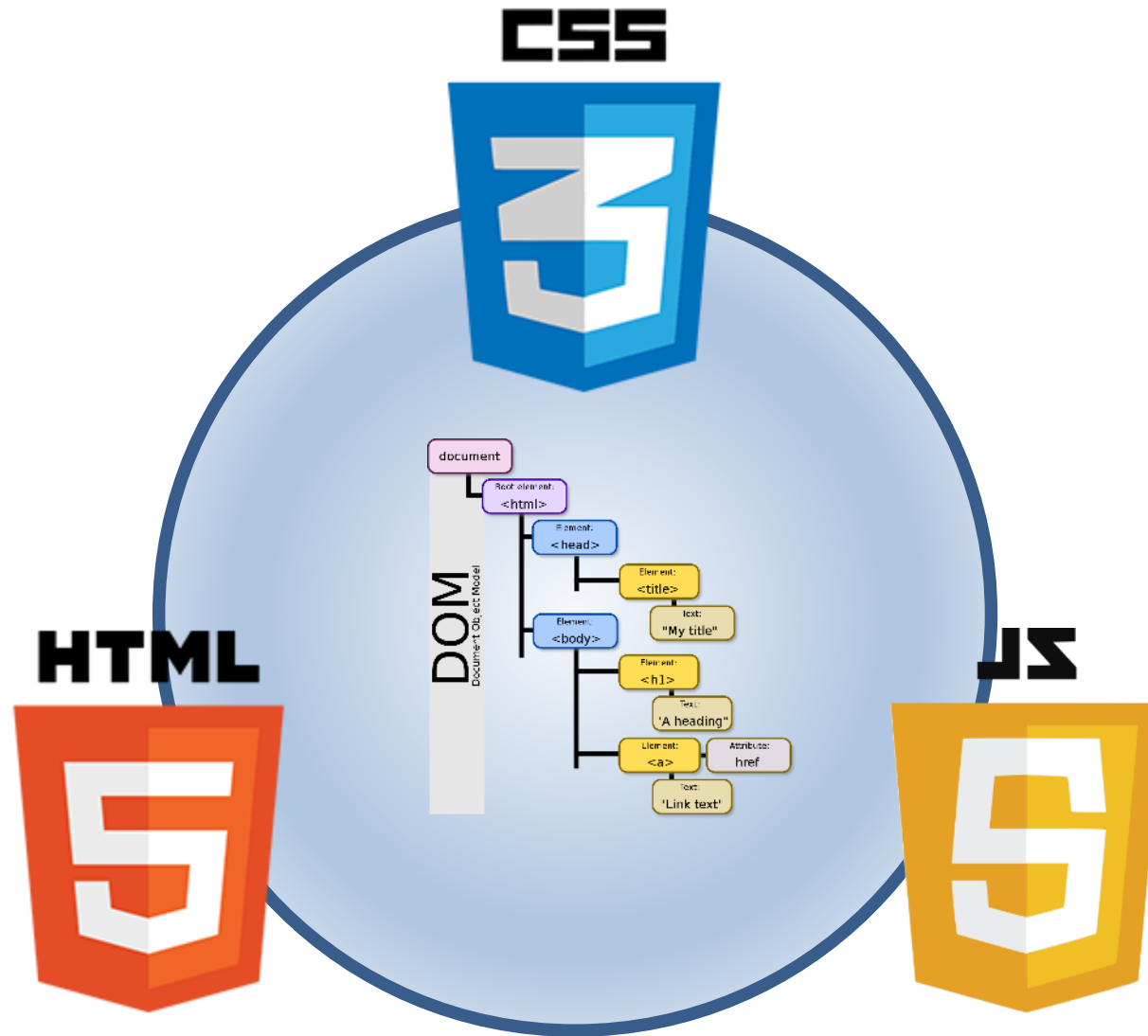
JavaScript DOM

DOM(Document Object Model)

```
<!doctype html>
<html lang="en">
  <head>
    <title>My first web page</title>
  </head>
  <body>
    <h1>Hello, world!</h1>
    <p>How are you?</p>
  </body>
</html>
```



JavaScript DOM



JS html태그 접근

메소드

메소드	설명
getElementById("아이디명")	태그의 id 속성 값을 이용해서 태그 엘리먼트 객체 정보를 가져온다.
getElementsByName("이름")	태그의 name 속성 값을 이용해서 태그 엘리먼트의 객체 정보를 배열 에 담아서 가져온다. 같은 이름의 태그가 여러 개 존재할 수 있개 때문에 기본적으로 배열로 리턴한다.
getElementsByTagName("태그명")	태그명을 이용해서 해당 태그들의 객체 정보를 배열 에 담아서 가져온다.

Javascript html태그 접근

```
document.getElementById()
```

- HTML태그의 id속성의 아이디명은 페이지에서 유일한 식별자 역할을 하도록 권장
- 리턴값 : 단일값(id는 중복허용 안함)

```
var 변수 = document.getElementById("아이디명");
```

☞ 변수는 객체를 의미하는 레퍼런스 변수(객체.항목.속성)

Javascript html태그 접근

```
document.getElementsByName()
```

- HTML태그의 name속성의 값(이름)으로 객체 정보를 가져올때 사용
- 리턴값 : 배열(name은 중복 가능)

```
var 변수 = document.getElementsByName("이름");
```

☞ 변수는 배열이 됨

Javascript html태그 접근

```
document.getElementsByTagName()
```

- HTML태그의 태그명을 이용해서 태그들을 한꺼번에 몽땅 가져와서 순서대로 제어
- 리턴값 : 배열(태그 중복가능)

```
var 변수 = document.getElementsByTagName("태그명");
```

☞ 변수는 배열이 됨

데이터 출력방법

코드	설명
<code>document.write(내용);</code>	브라우저 화면상의 페이지에 값을 출력한다.
<code>window.alert(내용);</code>	내용을 메시지창에 출력한다. <small>☞ window객체는 모두 적용되는 것으로 생략가능</small>
<code>innerHTML = 내용;</code>	태그 엘리먼트의 내용을 변경하여 출력한다.
<code>console.log(내용);</code>	개발자도구 화면의 콘솔에 출력한다.

데이터 입력방법

자바스크립트 내장객체인 window 객체가 제공하는 confirm(), prompt() 메소드를 사용하여 입력 받는 방법

HTML 태그에 접근하여 대상의 값을 읽는 방법

HTML form 태그의 input 입력양식을 통해 값을 입력 받는 방법

window.confirm()

어떤 질문에 대해 “예/아니오”의 결과를 얻을 때 사용

대화창에 메시지와 확인, 취소버튼 표시

리턴 값 : 확인(true), 취소(false)

```
var 변수 = [window.]confirm(“질문내용”);
```


Javascript 데이터 입력

```
window.prompt()
```

텍스트 필드와 확인/취소버튼이 있는 대화창 출력

리턴값 : 입력한 메시지 내용

```
var 변수 = [window.]prompt("메시지");
```