

웹브라우저 자바스크립트

WebBrowser 구성 요소

- HTML : 웹 화면을 표현
- CSS : 웹 화면을 꾸밈
- JavaScript : 웹브라우저, HTML을 제어

HTML에서 Javascript 로드

Inline

- 태그에 직접 Javascript를 기술하는 방식

```
<input type="button" onclick="alert('Hello world')" value="Hello world" />
```

Script

- `<script>` `</script>` 태그를 만들어서 여기에 자바스크립트 코드를 삽입하는 방식
- HTML코드와 Javascript코드를 분리 할 수 있음.

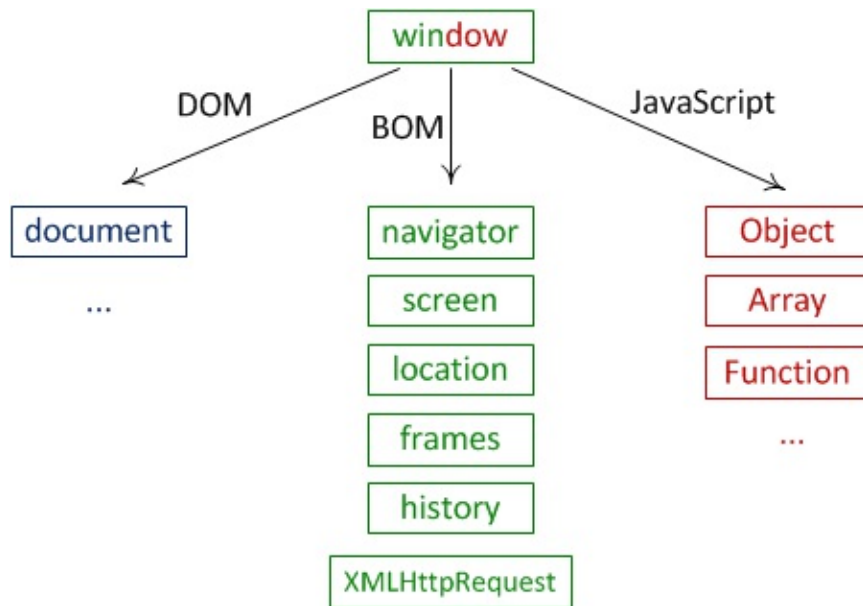
```
<html>
<body>
  <input type="button" id="hw" value="Hello world" />
  <script type="text/javascript">
    var hw = document.getElementById('hw');
    hw.addEventListener('click', function(){
      alert('Hello world');
    })
  </script>
</body>
</html>
```

외부 파일

- Javascript를 별도의 파일로 분리하여 사용
- js파일을 여러 웹페이지에서 사용하여 재활용성을 높일 수 있음.
- 유지보수가 용이

Window 객체

Window 객체



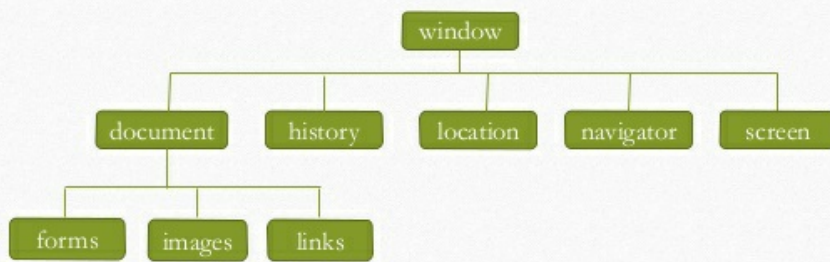
- 전역객체
- 모든 전역 자바스크립트 객체, 함수, 변수들은 자동으로 window 객체의 멤버가 된다.
 - 모든 객체는 window의 자식
 - 전역 변수는 window객체의 프로퍼티가 되고,
 - 전역 함수는 window객체의 메소드가 된다.
- window 객체는 생략 가능하다.

```
alert('hello world');  
window.alert('hello world');  
/* 둘다 같은 내용 */
```

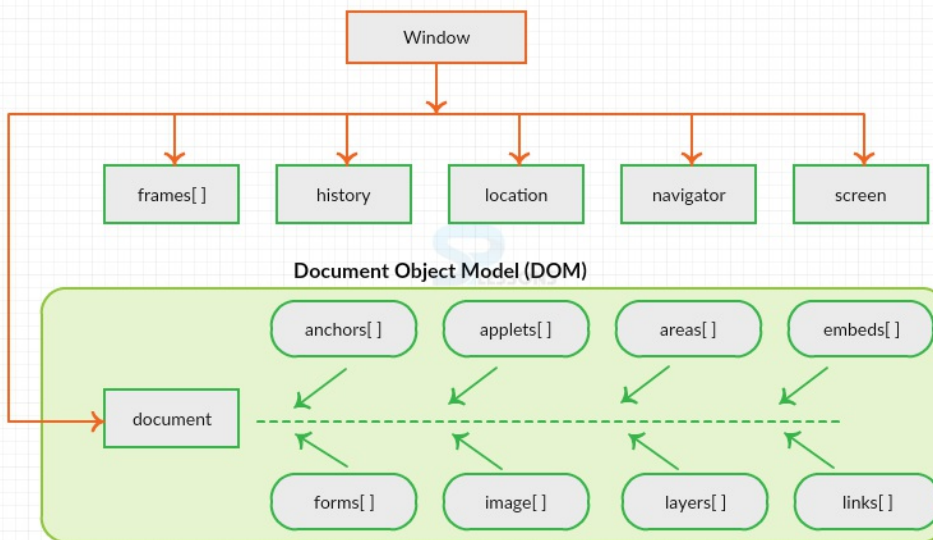
Ex : EX_window.html

BOM

Browser Object Model (BOM)



Browser Object Model (BOM)



- Browser Object Model
- 웹브라우저의 창이나 프레임을 추상화 하여 프로그래밍적으로 제어하는 수단.
- 웹페이지 내용을 제외한 브라우저의 각종요소들을 객체화 시킨것.
- 웹브라우저 창을 관리할 목적으로 제공되는 객체모음
- 전역객체 window의 프로퍼티에 속한 객체들이 이에 속한다.
 - DOM은 document(현재문서)에 대한 변형과 제어를 담당
 - BOM은 window 프로퍼티에 속해 window 제어를 담당
 - ex) alert, window.location, window.open, window.close
- 각 브라우저 마다 세부사항이 다르다.

BOM의 객체

window 객체

- 브라우저에서 열려있는 창을 나타냅니다.

location 객체

- 현재 URL에 대한 정보가 들어있습니다.

navigator 객체

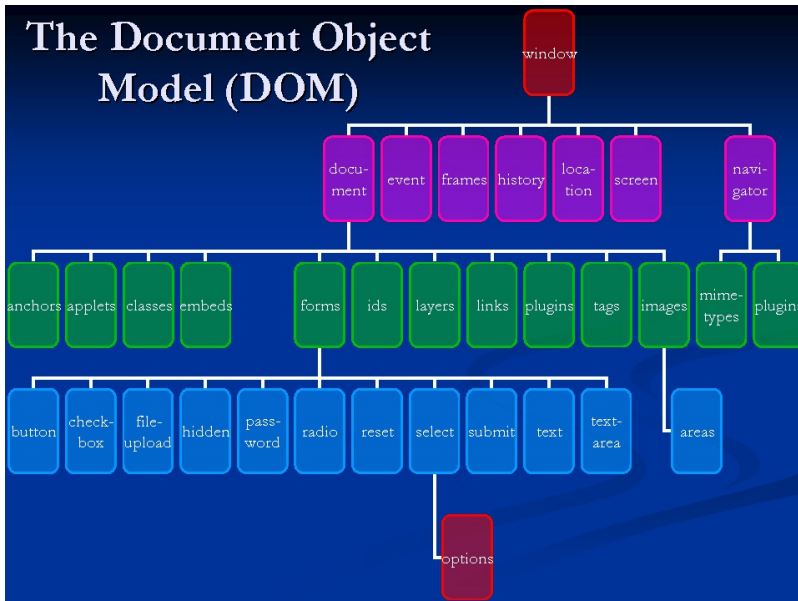
- 브라우저에 대한 정보가 들어있습니다.

[참고]

[The window object] : https://www.w3schools.com/jsref/obj_window.asp [The screen object] :
https://www.w3schools.com/jsref/obj_screen.asp [The history object] :
https://www.w3schools.com/jsref/obj_history.asp [The location object] :
https://www.w3schools.com/jsref/obj_location.asp [The navigator object] :
https://www.w3schools.com/jsref/obj_navigator.asp

DOM

DOM



- Document Object Model
- 웹 페이지의 내용을 제어한다.
- 문서의 구조적 형태를 제공하므로 자바스크립트와 같은 스크립트 언어를 사용하여 문서 내용과 시각적 표현을 수정할 수 있습니다.
- window의 프로퍼티인 document 프로퍼티에 할당된 Document객체가 담당
- Document 객체의 프로퍼티는 문서 내의 주요 엘리먼트에 접근할 수 있는 객체를 제공 [예제]
- 특정한 엘리먼트의 객체를 획득할 수 있는 메소드도 제공 [예제]

[참고] : <https://developer.mozilla.org/ko/docs/DOM>

제어대상 찾기

- 문서를 자바스크립트로 제어 하려면 먼저 제어 할 대상에 해당되는 객체를 찾는다.
- 문서 내 객체를 찾는 방법은 document 객체의 메소드를 이용한다.
- document.getElementsByTagName
 - 태그의 이름으로 엘리먼트들을 가져온다.
 - 가져온 값은 nodeList라는 유사배열에 저장한다.
- document.getElementsByClassName
 - 클래스 이름으로 엘리먼트들을 가져온다.
- document.getElementById
 - ID값으로 엘리먼트를 가져온다.

- document.querySelector
 - css의 선택자(selector)를 인자로 선택자의 엘리먼트를 가져온다. (하나만)
- document.querySelectorAll
 - document.querySelector와 같으나 유사배열에 담아 반환. (여러개)

BOM과 DOM 비교

BOM	DOM
웹브라우저의 창이나 프레임을 추상화 하여 프로그래밍적으로 제어 할 수 있도록 제공하는 수단.	웹 페이지를 Javascript로 제어하기 위한 객체 모델을 의미.
전역객체인 Window의 프로퍼티와 메소드들을 통해서 제어할 수 있다.	윈도우에 로드된 문서를 의미한다.

이벤트

Event

이벤트는 어떤 사건(action)이 발생하는것을 의미한다.

- event target
 - event가 일어날 객체.
- event type
 - event의 종류.
- event handler
 - event가 발생했을시 동작할 코드.

이벤트 등록 방법

- inline

```
<input type="button" onclick="alert('Hello world');" value="button" />
```

- 프로퍼티 리스너
 - 이벤트 대상에 해당하는 객체의 프로퍼티로 이벤트를 등록하는 방식
 - 단점으로 여러개의 이벤트 핸들러를 등록해도 단 하나만 실행된다.

```
<input type="button" id="target" value="button" />
<script>
  var t = document.getElementById('target');
  t.onclick = function(){
    alert('Hello world');
  }
</script>
```

- addEventListener
 - 가장 많이 사용됨
 - 하나의 이벤트 대상에 복수의 동일 이벤트 타입 리스너를 등록할 수 있다

이벤트 종류들은 아래와 같다.

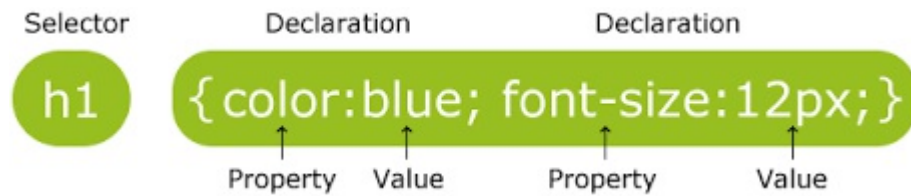
- 마우스 이벤트
- 키보드 이벤트
- 프레임/오브젝트 이벤트
- 폼 이벤트

[이벤트 참고] : https://www.w3schools.com/jsref/dom_obj_event.asp

CSS 기본

CSS는 문서의 콘텐츠와 레이아웃, 글꼴 및 시각적 요소들로 표현되는 문서의 외관(디자인)을 분리하기 위한 목적으로 만들어졌다.

기본 문법



- css는 선택자(selector)와 선언부(declaration)로 나뉜다.
- 선언부는 다시 속성(property)과 속성값(value)으로 나뉜다.

[예제] button을 꾸미는 css

```
<style>
  button {
    color:#fff;
    font-weight:bold;
    padding: 5px 20px;
    background: #4ca5d0;
    border:1px solid #327ea3;
  }
</style>
```


jQuery

자바스크립트를 쉽게 사용 할 목적으로 개발된, 적은 코드로 많은 일을 하기 위한 자바스크립트로 만들어진 라이브러리.

주요 기능

- HTML/DOM 조작
- CSS 조작
- HTML 이벤트
- 효과, 애니메이션
- 유틸리티

jQuery 구문

```
$(제어대상).method_1().method_n();

$('.welcome').html('hello world! coding everybody!').css('background-color', 'yellow');
```

wrapper

- jQuery(엘리먼트 오브젝트 | 'CSS스타일 선택자')
- 인자로 전달된 요소들에 jQuery의 기능성을 부가해서 반환

```
jQuery(document.body).css( "background-color", "black" );
$(document.body).css( "background-color", "black" );
// 둘은 같은 내용이나 $를 사용하는 다른 라이브러리와 충돌 가능성으로 jQuery 라고 사용
```

선택자

- 제어하려는 엘리먼트를 빠르고 정확하게 지정

chain

- jQuery의 메소드들은 반환값으로 자기 자신을 반환해야 한다는 규칙을 가지고 있다.
- 이를 이용하면 한번 선택한 대상에 대해서 연속적인 제어를 할 수 있다.

```
<html>
  <body>
    <a id="tutorial" href="http://jquery.com" target="_self">jQuery</a>
    <script type="text/javascript"
src="https://ajax.googleapis.com/ajax/libs/jquery/1.6.2/jquery.min.js">
```

```
</script>
    <script type="text/javascript">
        jQuery('#tutorial').attr('href',
'http://jquery.org').attr('target', '_blank').css('color', 'red');
    </script>
</body>
</html>
```

event

element 제어

form

탐색

애니메이션

ajax

Bootstrap

[site] : <http://bootstrapk.com/>

- 부트스트랩이란 하나의 웹 사이트 개발 프레임워크로서, 디자인을 잘 못하는 프로그래머나 HTML을 거의 알지 못하는 사람들도 쉽게 홈페이지를 개발할 수 있도록 해줍니다.
- 부트스트랩은 반응형이며 모바일 우선인 웹프로젝트 개발을 위한 가장 인기있는 HTML, CSS, JS 프레임워크입니다.
- 부트스트랩의 스타일시트는 웹페이지를 만들기 위한 거의 모든 요소에 대해서 정의를 해놓았기 때문에 손쉽게 웹사이트를 만들 수 있다는 장점이 있음.