

웹 앱 개발을 위한 Javascript 기초

6. 브라우저 및 문서 객체

학습내용

1. 브라우저 객체 모델
2. 문서 객체 모델

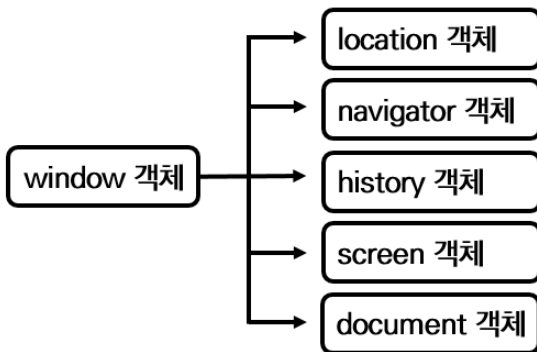
학습목표

1. 자바스크립트의 브라우저 객체 모델에 대해 설명할 수 있다.
2. 자바스크립트의 문서 객체 모델에 대해 설명할 수 있다.

1. 브라우저 객체 모델

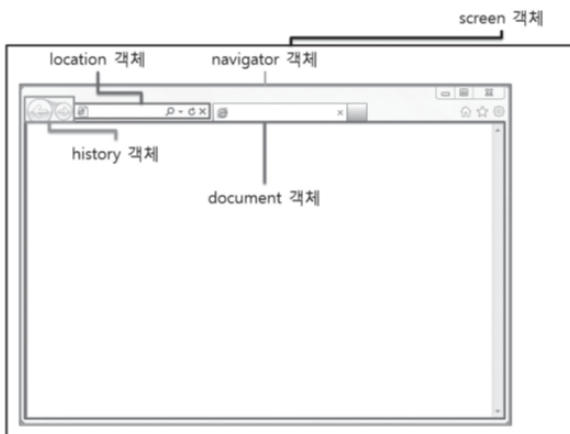
📖 브라우저 객체 모델

- * 웹 브라우저와 관련된 객체의 집합
- * 대표적인 브라우저 객체 모델
 - window, location, navigator, history, screen, document 객체



📖 브라우저 객체 모델

- * 브라우저 관련 객체 정리



1. 브라우저 객체 모델

window 객체

- * window 객체의 속성
- * 많은 속성이 있음
- * 일부 익스플로러에서는 실행되지 않음

window 객체

- * window 객체는 브라우저 기반 자바스크립트의 최상위 객체
- * alert(), prompt () 함수 모두 window 객체의 메소드

이 페이지 내용:

글자를 입력해주세요

확인

취소

이 페이지 내용:

입력했어요

확인

1. 브라우저 객체 모델

새로운 window 객체 생성

- * window 객체 생성
 - open () 메소드는 네 개의 매개변수가 있음
 - 입력해도 되고 입력하지 않아도 되는 매개변수를 옵션이라고 함
 - open () 메소드의 모든 매개변수는 옵션

메서드 이름	설명
open(URL, name, features, replace)	새로운 window 객체를 생성합니다.

새로운 window 객체 생성

- * open() 메소드의 옵션을 사용하지 않은 예
- * 옵션을 사용한 open() 메소드와 차이가 없음

```
<script>
  window.open();
</script>
```

```
<script>
  window.open('http://hanb.co.kr','child', 'width=600, height = 300', true);
</script>
```



1. 브라우저 객체 모델

새로운 window 객체 생성

- * open() 메소드의 첫 번째 매개변수
 - 열고자 하는 HTML 페이지의 URL
- * open() 메소드의 두 번째 매개변수
 - 윈도우 간 통신하는데 사용하는 윈도우 이름
- * open() 메소드의 세 번째 매개변수
 - 윈도우 출력 모양을 지정하는 옵션
- * open() 메소드의 네 번째 매개변수
 - URL이 새로운 것인가 기존의 문서를 대체할 것인가를 나타내는 부울값

새로운 window 객체 생성

옵션 이름	설명	입력할 수 있는 값
height	새 윈도우의 높이	픽셀 값
width	새 윈도우의 너비	픽셀 값
location	주소 입력창의 유무	yes, no, 1, 0
menubar	메뉴의 유무	yes, no, 1, 0
resizable	화면 크기 조절 가능 여부	yes, no, 1, 0
status	상태 표시줄의 유무	yes, no, 1, 0
toolbar	상태 표시줄의 유무	yes, no, 1, 0

1. 브라우저 객체 모델

📄 새로운 window 객체 생성

* open() 메소드의 속성과 메소드

```
<script>
  // 변수를 선언합니다.
  var child = window.open("", "", 'width=300, height=200');

  // 출력합니다.
  child.document.write('<h1>Form Parent Window</h1>');
</script>
```

📄 window 객체의 기본 메소드

* window 객체는 자신의 형태와 위치를 변경할 수 있도록 메소드 제공

메서드 이름	설명
moveBy(x,y)	윈도우의 위치를 상대적으로 이동합니다.
moveTo(x,y)	윈도우의 위치를 절대적으로 이동합니다.
resizeBy(x,y)	윈도우의 크기를 상대적으로 지정합니다.
resizeTo(x,y)	윈도우의 크기를 절대적으로 지정합니다.

1. 브라우저 객체 모델

window 객체의 기본 메소드

- * window 객체는 자신의 형태와 위치를 변경할 수 있도록 메소드 제공

메서드 이름	설명
scrollBy(x,y)	윈도우 스크롤의 위치를 상대적으로 이동합니다.
scrollTo(x,y)	윈도우 스크롤의 위치를 절대적으로 이동합니다.
focus()	윈도우에 초점을 맞춥니다.
blur()	윈도우에 초점을 제거합니다.
close()	윈도우를 닫습니다.

window 객체의 기본 메소드

- * 상대 이동과 절대 이동 (새로운 창이 지정 경로로 이동)
 - **OOBy () 형태의 메소드**
 - ➡ 현재의 윈도우를 기준으로 상대적으로 속성을 변화
 - **OOTo () 형태의 메소드**
 - ➡ 절대적인 기준으로 속성을 변화

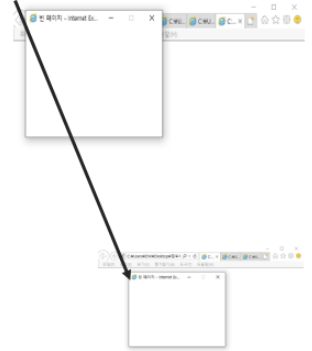
1. 브라우저 객체 모델

📖 window 객체의 기본 메소드

```
<script>
  // 변수를 선언합니다.
  var child = window.open("", "", 'width=300, height=200');

  child.moveTo(0,0);

  // 1초마다 함수를 실행합니다.
  setInterval(function () {
    child.moveBy(10, 10);
  }, 1000);
</script>
```



📖 screen 객체

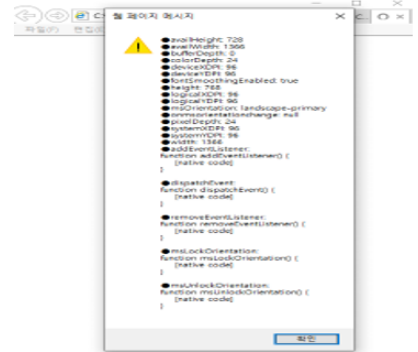
- * 웹 브라우저의 화면이 아닌 운영체제 화면의 속성을 가지는 객체
- * screen 객체의 속성

속성 이름	설명
width	화면의 너비
height	화면의 높이
availWidth	실제 화면에서 사용 가능한 너비
availHeight	실제 화면에서 사용 가능한 높이
colorDepth	사용 가능한 색상 수
pixelDepth	한 픽셀당 비트 수

1. 브라우저 객체 모델

screen 객체

```
<script>
  // 출력합니다.
  var output = '';
  for (var key in screen) {
    output += '●' + key + ';' + screen[key] + '\n';
  }
  alert(output);
</script>
```



location 객체

- * 브라우저의 주소 표시줄과 관련된 객체
- * location 객체는 프로토콜의 종류, 호스트 이름, 문서 위치 등의 정보가 있음
- * location 객체 속성

1. 브라우저 객체 모델

location 객체

속성 이름	설명	예
href	문서의 URL주소	http://www.example.com/
host	호스트 이름과 포트 번호	localhost:30763
hostname	호스트 이름	localhost
port	포트 번호	30763
pathname	디렉토리 경로	/Projects/Location.htm
hash	앵커 이름(#~)	#beta
search	요청 매개변수	?param=10
protocol	프로토콜 종류	http:

location 객체

- * replace () 메소드는 뒤로 가기 버튼을 사용할 수 없음

속성 이름	설명
assign(link)	현재 위치를 이동합니다.
reload()	새로고침합니다.
replace(URL)	현재 위치를 이동합니다.

1. 브라우저 객체 모델

location 객체

* navigator 속성

속성 이름	설명
appName	브라우저의 코드명
appName	브라우저의 이름
appVersion	브라우저의 버전
platform	사용 중인 운영체제의 시스템 환경
userAgent	브라우저의 전체적인 정보

onload 이벤트 속성

- * 문서 객체 속성 중 'on'으로 시작하는 속성을 이벤트 속성이라 부르고 함수를 할당해야 함

* onload 이벤트 속성

```
<script>
  window.onload = function () {

  };
</script>
```

1. 브라우저 객체 모델

📖 window 객체의 로드 완료

* window 객체 로드가 완료되는 때는?

➡ HTML 페이지에 존재하는 모든 태그가 화면에 올라가는 순간이 로드가 완료되는 순간

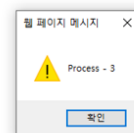
📖 window 객체의 로드 완료

* HTML 페이지 생성 순서

```
<!DOCTYPE html>
<html>
<head>
  <title>Document Object Model</title>
  <script>alert('Process - 0')</script>
</head>
<body>
  <h1>Process - 1</h1>
  <script>alert('Process - 2')</script>
  <h2>Process - 2</h2>
  <script>alert('Process - 3')</script>
</body>
</html>
```

Process - 1

Process - 2



2. 문서 객체 모델

📖 문서 객체 모델

* 넓은 의미

👉 웹 브라우저가 HTML 페이지를 인식하는 방식

* 좁은 의미

👉 document 객체와 관련된 객체의 집합

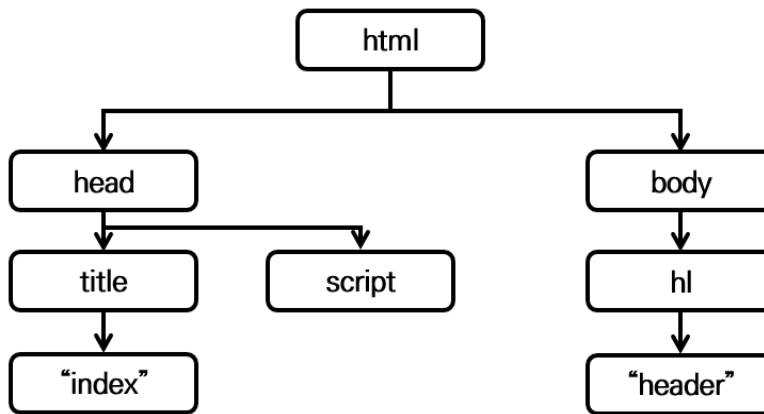
* 문서 객체 모델을 사용하면 HTML 페이지에 태그를 추가, 수정, 제거할 수 있음

* 태그

- HTML 페이지에 존재하는 **html**이나 **body** 태그를 '태그'라고 부름

* 문서 객체

- **html**이나 **body** 태그를 자바스크립트에서 이용할 수 있는 객체로 만들면 문서 객체임



2. 문서 객체 모델

📖 문서 객체 모델

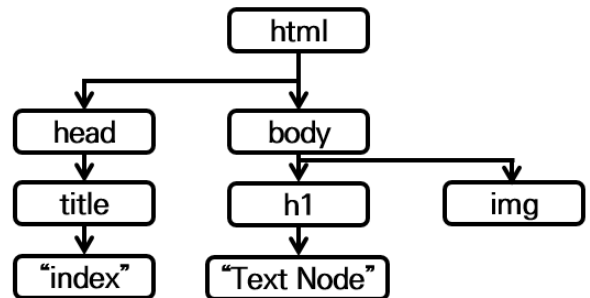
* 노드

● 요소 노드

➡ HTML 태그를 의미

● 텍스트 노드

➡ 요소 노드 안에 들어 있는 글자를 의미



📖 문서 객체 모델

* 문서 객체 생성

● 정적으로 문서 객체를 생성

➡ 처음 HTML 페이지에 적혀 있는 태그들을 읽으며 생성

● 동적으로 문서 객체를 생성

➡ 자바스크립트로 원래 HTML 페이지에는 없던 문서 객체를 생성

2. 문서 객체 모델

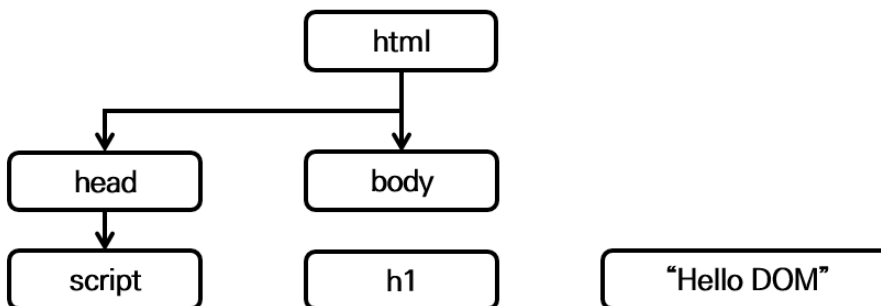
📖 문서 객체 만들기

- * 텍스트 노드를 갖는 문서 객체
 - 요소 노드와 텍스트 노드 생성 후 텍스트 노드를 요소 노드에 붙여 줌
- * document 객체의 노드 생성 메소드

메서드 이름	설명
<code>createElement(tagName)</code>	요소 노드를 생성합니다.
<code>createTextNode(text)</code>	텍스트 노드를 생성합니다.

📖 문서 객체 만들기

- * 현재 상황



2. 문서 객체 모델

📄 문서 객체 만들기

- * 화면에 문서를 출력하려면 생성한 문서 객체를 body 문서 객체에 연결

메서드 이름	설명
appendChild(node)	객체에 노드를 연결합니다.

- * 텍스트 노드를 갖지 않는 문서 객체
- * 텍스트 노드를 갖지 않는 대표적인 HTML 태그는 img
 - img 태그는 텍스트 노드 대신에 많은 속성이 있음
 - img 태그 생성 후 body 문서 객체에 연결
- * 문서 객체 생성

```
<script>
  window.onload = function () {
    // 변수를 선언합니다.
    var img = document.createElement('img');
    // 노드를 연결합니다.
    document.body.appendChild(img)
  };
</script>
```

- * 텍스트 노드를 갖지 않는 문서 객체
- * img 태그에 이미지를 넣으려면 src 속성 지정
 - 코드를 실행하면 이미지 출력

2. 문서 객체 모델

📄 문서 객체 만들기

* 문서 객체의 속성 지정

```
<script>
  window.onload = function () {
    // 변수를 선언합니다.
    var img = document.createElement('img');
    img.src = 'Penguins.jpg';
    img.width = 500;
    img.height = 350;

    // 노드를 연결합니다.
    document.body.appendChild(img)
  };
</script>
```

📄 문서 객체 만들기

* innerHTML

- 문서 객체 내부의 글자를 나타냅니다

* 문서 객체의 innerHTML 속성

<u><h1></u>	<u>Hello DOM..!</u>	<u></h1></u>
여는 태그	innerHTML 속성	닫는 태그

2. 문서 객체 모델

📄 문서 객체 만들기

```
<script>
  window.onload = function () {
    // 변수를 선언합니다.
    var output = '';

    // innerHTML 속성에 문자열을 할당합니다.
    document.body.innerHTML = output;
  };
</script>
```

📄 문서 객체 만들기

- * HTML 태그를 자바스크립트로 가져오는 방법
 - document 객체가 가지는 아래표 메소드를 사용
 - document 객체의 getElementById() 메소드는 id 속성을 갖는 태그만 가져올 수 있으므로 id 속성을 입력

메서드 이름	설명
getElementById(id)	태그의 id 속성이 id와 일치하는 문서 객체를 가져옵니다.

- header1과header2는 문서 객체이므로 뒤에 점을 찍어주면
문서 객체의 속성과 메소드를 살펴볼 수 있음

2. 문서 객체 모델

문서 객체 만들기

* header1 객체의 속성

```

window.onload = function () {
    // 문서 객체를 가져옵니다.
    var header1 = document.getElementById('header-1');
    var header2 = document.getElementById('header-2');

    header1.inn
};

```

문서 객체 만들기

* HTML 태그를 자바스크립트로 가져오는 방법

● innerHTML 속성 변경

```

<script>
    window.onload = function () {
        // 문서 객체를 가져옵니다.
        var header1 = document.getElementById('header-1');
        var header2 = document.getElementById('header-2');

        // 문서 객체의 속성을 변경합니다.
        header1.innerHTML = 'with getElementById()';
        header2.innerHTML = 'with getElementById()';
    };
</script>

```

출력

with getElementById()
with getElementById()

2. 문서 객체 모델

📖 문서 객체 만들기

- * 여러 개의 문서 객체 가져오는 방법
 - document 객체의 getElementById () 메소드는 **한 번에 한 가지 문서 객체만** 가져올 수 있음
 - 아래 표를 이용해서 여러 개의 객체를 가져올 수 있음

메서드 이름	설명
getElementsByTagName(name)	태그의 name 속성이 name과 일치하는 문서 객체를 배열로 가져옵니다.
getElementsByTagName(tagName)	tagName과 일치하는 문서 객체를 배열로 가져옵니다.

📖 문서 객체 만들기

- * 여러 개의 문서 객체 가져오는 방법

```
<!DOCTYPE html>
<html>
<head>
  <title>Index</title>
  <script>
    window.onload = function () {
      // 문서 객체를 가져옵니다.
      var headers = document.getElementsByTagName('h1');
    };
  </script>
```

(스크립트 이어짐)

2. 문서 객체 모델

문서 객체 만들기

- * 여러 개의 문서 객체 가져오는 방법

```
</head>
<body>
  <h1>Header</h1>
  <h1>Header</h1>
</body>
</html>
```

문서 객체의 스타일 조작

- * 문서 객체의 스타일 변경
- * style 속성 사용
- * getElementById () 메소드로 문서 객체를 가져옴
- * style 속성에 있는 border, color, fontFamily 속성을 지정
- * CSS에 입력하는 것과 같은 형식으로 입력

2. 문서 객체 모델

문서 객체의 스타일 조작

* 문서 객체의 style 속성

메서드 이름	설명
removeChild(child)	문서 객체의 자식 노드를 제거합니다.

* h1 태그 제거

```
<body>
  <h1 id = "will-remove">Header</h1>
</body>
```

문서 객체의 스타일 조작

* 문서 객체의 제거

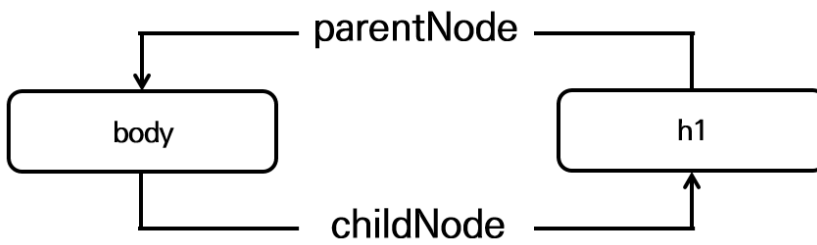
```
<script>
  window.onload = function () {
    // 문서 객체를 가져옵니다.
    var willRemove = document.getElementById('will-remove');

    // 문서 객체를 제거합니다.
    document.body.removeChild(willRemove);
  };
</script>
```

2. 문서 객체 모델

문서 객체의 스타일 조작

* parentNode와 childNode의 관계



정리하기

1. 브라우저 객체 모델

- window 객체
- 많은 속성이 있음, 일부 익스플로러에서는 실행되지 않음
- window 객체는 자신의 형태와 위치를 변경할 수 있도록 메소드 제공
- screen 객체 : 웹 브라우저의 화면이 아닌 운영체제 화면의 속성을 가지는 객체
- location 객체
- 브라우저의 주소 표시줄과 관련된 객체
- location 객체는 프로토콜의 종류, 호스트 이름, 문서 위치 등의 정보가 있음
- navigator 객체 : navigator 객체는 웹 페이지를 실행하고 있는 브라우저에 대한 정보가 있음
- window 객체의 onload 이벤트 속성 : 문서 객체 속성 중 'on'으로 시작하는 속성을 이벤트 속성이라 부르고 함수를 할당해야 함

정리하기

2. 문서 객체 모델

- 문서 객체 만들기
- 넓은 의미로 웹 브라우저가 HTML 페이지를 인식하는 방식
- 좁은 의미로 document 객체와 관련된 객체의 집합
- 텍스트 노드를 갖는 문서 객체
- 요소 노드와 텍스트 노드 생성 후 텍스트 노드를 요소 노드에 붙여 줌
- 문서 객체 가져오기
- HTML 태그를 자바스크립트로 가져오는 방법
- document 객체가 가지는 아래표 메소드를 사용
- 문서 객체의 스타일 조작
- 문서 객체의 스타일 변경
- style 속성 사용
- 문서 객체 제거
- 제거 메소드 사용
- h1 태그 제거
- 문서 객체의 제거