객체(object)

JS에서 객체란, 프로그램에서 인식할 수 있는 모든 대상이다.
or (원시 값을 제외한 모든 데이터의 종류)

JS에서 사용하는 객체
- 문서객체모델 (DOM)
- 브라우저객체모델 (BOM)
- 내장객체

객체는 참조 형태로 사용해야 한다!

⇒ instance를 만들어 사용한다.

ex) let now = new Date();

할당 ↑ Date 객체의 instance 생성

프로퍼티 : 객체의 특징이나 속성

메서드 : 객체에서 할 수 있는 동작

ex) 프로그램: 자동차 운전
프로퍼티: 제조사, 모델명, 색상 등
메서드: 시동걸기, 전진, 후진 등

인스턴스는 객체의 프로퍼티와 메서드를 그대로 물려받는다.

```
let now = new Date();
document.write(now.toLocaleString());
```

↓

instance 뒤에 . 을 붙여 프로퍼티와 메서드 작성

내장객체

Array

Array 객체로 인스턴스 생성 - 초깃값이 없을 때

```
let numbers = new Array();   // 크기 미지정
let numbers = new Array(4);  // 크기 : 4
```

Array 객체로 인스턴스 생성 - 초깃값이 있을 때

```
let numbers = ["1", "2", "3"];
let numbers = new Array("1", "2", "3");
```

property : length

↳ 배열 요소의 개수가 저장되어 있음

ex) `document.write(numbers.length);`

⇒ 3 출력

method : concat()

↳ 서로 다른 2개의 배열을 합친다.

```
let nums = [1, 2, 3];
let chars = ["a", "b", "c"];
document.write(nums.concat(chars));
```

⇒ 1, 2, 3, a, b, c 출력

method : join()

↳ 배열 요소를 연결하여 하나의 문자열로 만들어줌

`document.write(nums.join('/'));`

⇒ 1/2/3 출력

↳ '/'를 지정하지 않으면 1,2,3 으로 출력됨

method : push()

↳ 배열 맨 끝에 새로운 요소 추가

method : unshift()

↳ 배열 맨 앞에 새로운 요소 추가

ex) document.write (nums.unshift (4,5));

⇒ 4,5,1,2,3 출력

원본 배열 변경됨!

method : pop()

↳ 배열 마지막 요소를 꺼냄

ex) let popped = nums.pop();

document.write (popped);

⇒ 3 출력,

원본 배열(nums)는 [1,2]로 변경

method : shift()

↳ 배열 첫 요소를 꺼냄

원본 배열 변경됨!

method : splice()

↳ 배열의 중간에 요소 추가/삭제

- 원본 배열 변경됨!

인수가 1개인 경우	2개인 경우	3개인 경우
인스턴스 . splice (인덱스 값) ⇒ 해당 인덱스부터 맨 끝까지 잘라냄 ex) let newNums = nums.splice(1); ⇒ nums에는 [1] newNums 에는 [2,3] 저장	인스턴스.splice(인덱스, 잘라낼 개수) ⇒ 해당 인덱스부터 잘라낼 개수만큼 배열을 잘라냄	인스턴스.splice (인덱스, 잘라낼 개수 추가 요소) ⇒ 잘라낸 위치에 새로운 요소를 추가함

method : slice()

- 원본배열 변경 안됨!

↳ 기존 배열을 바꾸지 않고 요소만 꺼내는 메소드

인수가 1개 : 인수를 시작 인덱스 값으로 하여, 마지막 요소까지 복사

2개 : 첫 인수를 시작 인덱스, 두번째 인수를 '끝 인덱스-1'로 설정

ex) slice (2, 4)

⇒ 2번 인덱스부터 3번 인덱스까지 복사

Date

자바스크립트의 날짜, 시간 입력 방식

1. YYYY-MM-DD
2. YYYY-MM-DDTHH:MM:SS
3. MM/DD/YYYY
4. 이름형식 (JAN, MON 등)

Date의 method

구분		설명
날짜·시간 정보 가져오기	getFullYear()	연도를 4자리 숫자로 표시합니다.
	getMonth()	0~11 사이의 숫자로 월을 표시합니다. 0부터 1월이 시작되고 11은 12월입니다.
	getDate()	1~31 사이의 숫자로 일을 표시합니다.
	getDay()	0~6 사이의 숫자로 요일을 표시합니다. 0부터 일요일이 시작되고 6은 토요일입니다.
	getTime()	1970년 1월 1일 자정 이후의 시간을 밀리 초(1/1000초)로 표시합니다.
	getHours()	0~23 사이의 숫자로 시를 표시합니다.
	getMinutes()	0~59 사이의 숫자로 분을 표시합니다.
	getSeconds()	0~59 사이의 숫자로 초를 표시합니다.
	getMilliseconds()	0~999 사이의 숫자로 밀리초를 표시합니다.
날짜·시간 설정하기	setFullYear()	연도를 4자리 숫자로 설정합니다.
	setMonth()	0~11 사이의 숫자로 월을 설정합니다. 0부터 1월이 시작되고 11은 12월입니다.
	setDate()	1~31 사이의 숫자로 일을 설정합니다.
	setTime()	1970년 1월 1일 자정 이후의 시간을 밀리초로 설정합니다.
	setHours()	0~23 사이의 숫자로 시를 설정합니다.
	setMinutes()	0~59 사이의 숫자로 분을 설정합니다.
	setSeconds()	0~59 사이의 숫자로 초를 설정합니다.
	setMilliseconds()	0~999 사이의 숫자로 밀리초를 설정합니다.

Math

Date, Array 객체는 인스턴스를 생성해야 했지만, Math 객체는 인스턴스를 생성하지 않는다.

Math 객체의 property

E : 오일러 상수

PI : 약 3.1415926535

SQRT2 : $\sqrt{2}$

⋮

LOG10E : $\log_{10} e$

Math 객체의 method

표 16-4 Math 객체의 메서드

종류	설명
abs()	절댓값을 반환합니다.
acos()	아크 코사인(arc cosine)값을 반환합니다.
asin()	아크 사인(arc sine)값을 반환합니다.
atan()	아크 탄젠트(arc tangent)값을 반환합니다.
atan2()	아크 탄젠트(arc tangent)값을 반환합니다.
ceil()	매개변수의 소수점 이하 부분을 올립니다.
cos()	코사인(cosine)값을 반환합니다.
exp()	지수 함수를 나타냅니다.
floor()	매개변수의 소수점 이하 부분을 버립니다.
log()	매개변수에 대한 로그(log)값을 반환합니다.
max()	매개변수 중 최댓값을 반환합니다.
min()	매개변수 중 최솟값을 반환합니다.
pow()	매개변수의 지숫값을 반환합니다.
random()	0과 1 사이의 무작위 수를 반환합니다. $(0 \le x < 1)$
round()	매개변수의 소수점 이하 부분을 반올림합니다.
sin()	사인(sine)값을 반환합니다.
sqrt()	매개변수에 대한 제곱근을 반환합니다.
tan()	탄젠트(tangent)값을 반환합니다.

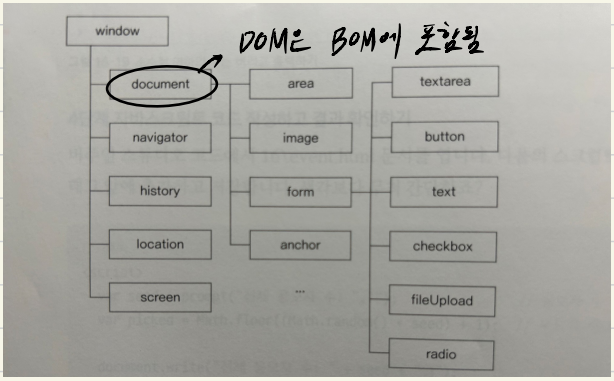
TIP Math 객체의 메서드는 developer.mozilla.org/ko/docs/Web/JavaScript/Reference/Global_ 서 확인할 수 있습니다.

브라우저 객체모델

웹 브라우저와 관련된 객체의 집합 (정확히는 JS가 제공하는 것이 아닌, 브라우저가 제공)
인스턴스를 생성하지 않고 바로 사용한다.

window

BOM의 최상위 객체



window의 property

종류	설명
document	브라우저 창에 표시된 웹 문서에 접근할 수 있습니다.
frameElement	현재 창이 다른 요소 안에 포함되어 있을 경우 그 요소를 반환하고, 반대로 포함되어 있지 않으면 null을 반환합니다.
innerHeight	내용 영역의 높이를 나타냅니다.
innerWidth	내용 영역의 너비를 나타냅니다.
localStorage	웹 브라우저에서 데이터를 저장하는 로컬 스토리지를 반환합니다.
location	window 객체의 위치 또는 현재 URL을 나타냅니다.

window의 method들

↳ alert(), prompt() 등은 window의 객체 이므로 원래는 window.을 붙여야 하지만, 생략가능하다. 마찬가지로 window의 메서드들에서 생략 가능

method : open()

↳ 웹 문서를 열 때 새창 (팝업)이 뜨게하는 메서드

window.open(경로, 창 이름, 창 옵션)

창 이름 지정 시, 새로 고침 해도 팝업은 1개만 나타날

ex) window.open("notice.html", "pop", "width=500,height=400")

※ 팝업 차단을 고려한 방법

```
let newWin = window.open("notice.html", "pop", "width=500,height=400")
if (newWin == null) {
        alert("팝업을 허제 해주세요");
}
```

method : close()

↳ 팝업 창을 닫는 메소드

navigator : 브라우저의 정보(플러그인, 온오프라인 등)를 뜻하는 객체

property : userAgent

↳ 사용자의 웹 브라우저 버전, JS의 엔진 등을 갖고있음

Navigator 객체는 브라우저 호환성을 위해서 주로 사용된다.

history : 방문한 사이트주소가 배열 형태로 저장되는 객체

property	length	방문한 사이트의 개수
method	back()	이전 페이지로
	forward()	다음 페이지로
	go()	현재 페이지 기준으로 상대적인 페이지를 불러옴 ex) history.go(1) ⇒ 다음 페이지로

location : 브라우저의 주소 표시줄과 관련된 객체

구분		설명
프로퍼티	hash	URL 중에서 #로 시작하는 해시 부분의 정보를 담고 있습니다.
	host	URL의 호스트 이름과 포트 번호를 담고 있습니다.
	hostname	URL의 호스트 이름이 저장됩니다.
	href	전체 URL입니다. 이 값을 변경하면 해당 주소로 이동할 수 있습니다.
	pathname	URL 경로가 저장됩니다.
	port	URL의 포트 번호를 담고 있습니다.
	protocol	URL의 프로토콜을 저장합니다.
	password	도메인 이름 앞에 username과 password를 함께 입력해서 접속하는 사이트의 URL일 경우에 password 정보를 저장합니다.
	search	URL 중에서 ?로 시작하는 검색 내용을 저장합니다.
	username	도메인 이름 앞에 username을 함께 입력해서 접속하는 사이트의 URL일 경우에 username 정보를 저장합니다.
메서드	assign()	현재 문서에 새 문서 주소를 할당해서 새 문서를 가져옵니다.
	reload()	현재 문서를 다시 불러옵니다.
	replace()	현재 문서의 URL을 지우고 다른 URL의 문서로 교체합니다.
	toString()	현재 문서의 URL을 문자열로 반환합니다.