

07

내장객체(Built-in Object)

- Date 객체
- Math 객체
- 정규표현식
- Object 객체
- Number 객체
- String 객체

❖ 자바스크립트 객체의 종류

- 객체는 자바스크립트 프로그램에서 인식할 수 있는 모든 대상
- 객체는 하나의 변수에 다양한 정보를 저장할 수 있는 자료형

객체 종류	설명
내장 객체 (Built-in Object)	자바스크립트 프로그래밍을 할 때 자주 사용하는 요소를 미리 정의 해 놓은 객체 Object, Number, String, Date, 정규표현식, Boolean 등
문서 객체 모델 (DOM)	웹 문서의 모든 요소를 자바스크립트가 이용할 수 있도록 객체로 해석한 것 즉, 웹 문서 안의 이미지나 텍스트 등이 모두 별 도의 객체
브라우저 객체 모델 (BOM)	웹 브라우저 전체를 객체로 관리하는 것 웹 브라우저 기반의 최상위 객체
사용자 정의 객체	사용자가 필요할 때마다 정의해서 사용하는 객체

❖ Date 객체

- 날짜와 시간을 표시하는 객체

생성 : let date = new Date(); -- 현재 날짜

let date = new Date(2022, 11, 25, 12, 25, 0, 0)

```
let date = new Date();
console.log(date.getFullYear());
console.log(date.getMonth());
console.log(date.getDate());
console.log(date.getDay());
console.log(date);
console.log(new Date(1))
console.log(new Date(1671938700000));
```

```
console.log(date.getHours());
console.log(date.getMinutes());
console.log(date.getSeconds());
console.log(date.getMilliseconds());
console.log(date.toGMTString());
console.log(date.getTime());
```

Unix time : 1970.1.1 12시 자정 기준

* 날짜와 시간 정보 가져올 때 : get, 설정할 때 : set

❖ Date 객체

- 날짜와 시간을 GMT (그리니치 표준시)시간으로 표시
- `getMonth()` : 0 ~ 11, 0 → 1월
- `getDay()` : 0 ~ 6, 0 → 일요일

```
console.log('-----to*String()-----')  
console.log(date.toString());  
console.log(date.toGMTString());  
console.log(date.toUTCString());  
console.log(date.toLocaleDateString());  
console.log(date.toLocaleString());  
console.log(date.toLocaleTimeString());  
console.log(date.toString());  
console.log(date.getTimeString());
```

❖ 크리스마스까지 몇 일 남았을까요?

- 1초 = 1000msc
- 1분 = 60초
- 1시간 = 60분
- 1일 = 24시간

※ 날짜 계산 결과 : 1000msc 로 나오므로 원하는 형태로 변경해야 함

```
let now = new Date();
```

```
let xmax = new Date();
```

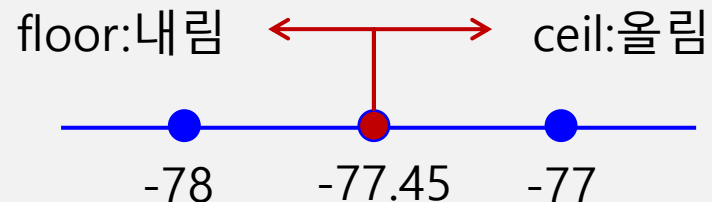
```
let remain = Math.floor();
```

```
console.log(`크리스마스 까지 ${remain}일 남았습니다`)
```

❖ Math 객체

- 생성자 함수를 사용하지 않는 객체
- 메서드를 변수에 저장해서 함수처럼 쉽게 사용

```
let max = Math.max;  
console.log(max(52, 512, 103, 777, 57, 32));  
console.log(Math.max(5,7,9,4));  
console.log(Math.min(4,5,9,10));  
  
console.log(Math.ceil(77.45));  
console.log(Math.ceil(-77.45));  
console.log(Math.floor(77.45));  
console.log(Math.floor(-77.45));  
console.log(Math.round(77.45)); -- 반올림
```



❖ Math 객체

```
console.log(Math.PI);    -- 3.141592653589793
```

```
console.log(Math.SQRT2); -- 1.4142135623730951
```

```
console.log(Math.abs(-5));
```

```
console.log(Math.pow(3,2));
```

```
console.log(Math.random());
```

```
console.log(Math.sqrt(16));
```

- 원하는 범위의 난수 만들기

$\text{Math.floor}(\text{Math.random()} * (\text{최대값} - \text{최소값} + 1)) + \text{최소값}$

❖ 로또번호 발생

- 1~45까지의 무작위 수 발생
- 6개씩 5줄 만들기

```
let lotto = "";
```

```
 {  
  for ( let j=0; j<6; j++){
```

```
    
```

```
  }
```

```
}
```

```
console.log(lotto);
```

21	24	36	43	43	19
33	13	32	10	9	37
41	14	33	26	36	8
11	11	36	5	8	20
8	42	7	14	22	19

도전 : 중복된 값이 발생하지 않도록 수정 해 보세요

❖ 정규표현식

- 일정한 패턴을 가진 문자열의 집합을 표현하기 위해 사용
- 패턴을 정의하고 테스트하여 간단히 데이터 체크
- 특정 패턴과 일치하는 문자열 검색, 추출, 치환할 수 있는 기능
- 주석이나 공백을 허용하지 않고,
- 여러 가지 기호 사용하여 가독성이 좋지 않음

생성 : let 변수 = new RegExp

let 변수 = /패턴/검색옵션

```
let tel = '010-1234-567팔';  
  
const regExp = /^d{3}-d{4}-d{4}$/;  
  
console.log(regExp.test(tel));
```

❖ 정규표현식

검색옵션	설명		
\d	숫자	\D	숫자 아님
\w	단어(알파벳, 숫자, _ 기호)	\W	단어 아님
\s	공백문자(탭, 띄어쓰기, 개행)	\S	공백 문자 아님
a\$	맨 뒤 문자가 a		
^a	맨 앞 문자가 a		
.	아무 글자		
[abc]	괄호 안의 글자		
[^abc]	괄호 안의 글자 제외		
[a-z]	알파벳 a에서 z까지		
[0-9]	숫자 0에서 9까지		

❖ 정규표현식

검색옵션	설명		
a*	a가 0개 또는 여러 개	a{5}	a가 5개
a+	a가 적어도 1개 이상	a{2,}	a가 2개 이상
a?	a가 0개 또는 1개	a{.2}	a가 2개 이하
i	대소문자 구분 없이 비교		
?	0 또는 1회 규칙에 맞는 문자 검색		
g	규칙에 맞는 모든 문자		
m	여러 줄의 검사 수행		

```
const target = 'Is this all there is?';
const regExp2 = /is/ig;
console.log(target.match(regExp2));
console.log(target.replace(regExp2, 'LOVE'));
```

❖ Object 객체

- 자바스크립트의 최상위 객체, 모든 객체가 활용

생성 : `let object = {};`

`let object = new Object();`

메서드	설명
constructor	객체의 생성자 함수
hasOwnProperty('name')	객체가 name 속성이 있는지 확인
propertyIsEnumerable('name')	반복문으로 열거할 수 있는지 확인
toLocaleString()	객체를 호스트 환경에 맞는 문자열로 변환
toString()	객체를 문자열로 변환
valueOf()	객체의 값

❖ Number 객체

- 가장 단순한 객체, 숫자 표현

생성 : `let num = 273`

`let numobj = new Number(273);`

메서드	설명
<code>toExponential()</code>	숫자를 지수로 표현(소수점 자릿수)
<code>toFixed()</code>	숫자를 고정 소수점 으로 표현(소수점 자릿수)
<code>toPrecision()</code>	숫자 길이에 따라 지수, 고정소수점으로 표현 유효 숫자의 자릿수

```
let num = 273.210332;
console.log(num.toFixed(2));
console.log(num.toExponential(3));
console.log(num.toPrecision(5));
console.log(num.toPrecision(2));
```

273.21

2.732e+2

273.21

2.7e+2

❖ String 객체

- 가장 많이 사용하는 내장객체

생성 : `let str = 'hello';`

`let strobj = new String('hello');`

메서드	설명
속성 : length	문자열의 길이, 글자수
charAt(index)	Index 위치의 한 글자 추출
charCodeAt(index)	Index 위치문자의 아스키 코드 값
fromCharCode(아스키코드)	코드에 대한 문자 리턴
concat(args)	매개변수 문자열을 이어서 리턴
indexOf('찾을 문자')	앞-문자의 위치 리턴, 없으면 -1 리턴
lastIndexOf('찾을 문자')	뒤-문자의 위치 리턴
match('찾을 문자')	문자열 안에 위치 리턴, 없으면 null

❖ String 객체

메서드	설명
replace(a, b)	a를 b로 변경
search('찾을 문자')	문자의 위치 리턴, 없으면 -1 리턴
slice(start, end)	start에서 end-1 까지 문자열 추출 리턴
split(separator)	문자열을 separator로 잘라서 배열로 리턴
substr(start, count)	start에서 count 만큼 문자열 잘라서 리턴
substring(start, end)	start에서 end-1 까지 문자열 잘라서 리턴
toLowerCase()	소문자로 변환
toUpperCase()	대문자로 변환
toString()	데이터를 문자열로 변환
trim()	문자 앞뒤 공백제거

❖ String 객체

//toString() : 데이터를 문자열로 변환

```
const num = 7788;  
console.log(num.toString());
```

```
const arr = [ 1, 2, 3, 'a', 'b'];  
console.log(arr.toString());
```

```
const obj = { key : 'data', age : 15};  
console.log(obj.toString());  
console.log(obj.key.toString());  
console.log(obj.age.toString());
```

7788

1,2,3,a,b

[object Object]

data

15

❖ String 객체

```
let str = ' 1A34b 7c9z ';
console.log(str.charAt(8));
console.log(str.charCodeAt(2));
console.log(String.fromCharCode(65));
console.log(str.concat('XYZ'));
console.log(str.indexOf('b'));
console.log(str.match('b'));
console.log(str.replace('4b', 'b5'));
```

```
console.log(str.search('y'));
console.log(str.slice(3,7));
console.log(str.split(' '));
console.log(str.substr(3,7));
console.log(str.substring(3,7));
console.log(str.toLowerCase());
console.log(str.toUpperCase());
console.log(str.trim());
```

c

65

A

1A34b 7c9z XYZ

5

▶ ['b', index: 5, input: ' 1A34b 7c9z ', groups: undefined]

1A3b5 7c9z

-1

34b

▶ (4) ['', '1A34b', '7c9z', '']

34b 7c9

34b

1a34b 7c9z

1A34B 7C9Z

1A34b 7c9z

❖ 도전! 문제

❖ 기념일 계산하기

우리 만난 지	
#accent	
기념일 계산	
100일	#date100
200일	#date200
1년	#date365
500일	#date500

1. html 구조 만들기
2. CSS 디자인
3. 날짜 계산하기
 - ① 오늘날짜 Date 객체 생성
 - ② 처음 만난 날 객체 생성
 - ③ 오늘날짜 - 처음 만난 날
 - ④ 결과 일로 변환하기
 - ⑤ 화면에 표시하기
 - ⑥ 기념일 계산 화면에 표시

문제1 기념일 계산하기

우리 만난 지

1579일

기념일 계산

100일	2018년 7월 1일
<hr/>	
200일	2018년 10월 9일
<hr/>	
1년	2019년 3월 23일
<hr/>	
500일	2019년 8월 5일