

06

문자열

- 정규표현식
- Object 객체
- Number 객체
- String 객체
- Date rorcp

❖ 정규표현식

- 입력요소에 데이터를 규칙에 맞게 작성했는지 알려주는 객체

생성 : `let 변수 = new RegExp`

`let 변수 = /패턴/검색옵션`

검색옵션	설명		
a*	a가 0개 또는 여러개	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		

❖ 정규표현식

- 입력요소에 데이터를 규칙에 맞게 작성했는지 알려주는 객체

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		
.	아무글자		
[abc]	괄호 안의 글자		
[^abc]	괄호 안의 글자 제외		
[a-z]	알파벳 a에서 z까지		
[0-9]	숫자 0에서 9까지		

❖ 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
isPrototypeOf(object)	객체가 object의 프로토타입인지 검사
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 : <code>length</code>	문자열의 길이, 글자수
<code>charAt(index)</code>	Index 위치의 한글자 추출
<code>charCodeAt('찾을 문자')</code>	문자의 아스키 코드값 리턴
<code>fromCharCode(아스키코드)</code>	코드에 대한 문자 리턴
<code>concat(args)</code>	매개변수 문자열을 이어서 리턴
<code>indexOf('찾을 문자')</code>	앞-문자의 위치 리턴, 없으면 -1 리턴
<code>lastIndexOf('찾을 문자')</code>	뒤-문자의 위치 리턴
<code>match('찾을 문자')</code>	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
trim()	문자 앞뒤 공백제거

❖ 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

❖ 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 = 1000 * 60sc
- 1시간 = 1000 * 60 * 60m
- 1일 = 1000 * 60 * 60 * 24

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

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

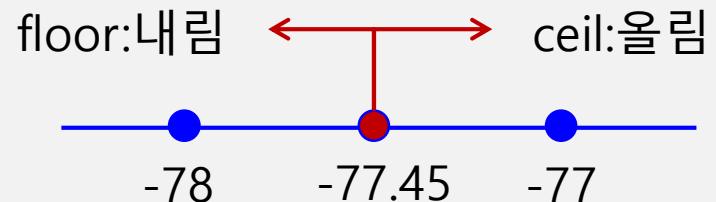
```
let remain = Math.floor();
```

```
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 i=0; i<5; i++){
  for ( let j=0; j<6; j++){
    lotto += Math.floor(Math.random()*45 + 1 ) + 'Wt';
  }
  lotto += 'Wn';
}
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