

OSS기반 시프로그래밍

## 2장. 자바스크립트 객체 다루기

동아대학교 컴퓨터·시공학부  
양 선

- 자바스크립트 객체를 이해
- Array, String, Date, Math 객체의 특징
- 사용자 정의 객체를 만드는 방법

교재의 7장에 해당합니다.

# 01 자바스크립트 객체 이해하기

## ■ 자바스크립트 객체

### 2. 자바스크립트 객체의 구성

- 자바스크립트 객체는 데이터를 의미하는 속성(property)과 데이터를 조작하는 메서드로 구성된 집합



객체.속성에  
값을 담으면 돼.

`object.property = value`

(a) 속성



괄호가 있으면  
메서드야.

`object.method(value)`

(b) 메서드

그림 7-3 자바스크립트 객체의 구성

# 01 자바스크립트 객체 이해하기

## ■ 자바스크립트 객체

### 2. 자바스크립트 객체의 구성

표 7-1 자바스크립트 객체의 메서드

| 메서드                           | 내용                       |
|-------------------------------|--------------------------|
| <code>document.write()</code> | document 객체의 write() 메서드 |
| <code>Math.floor()</code>     | Math 객체의 floor() 메서드     |
| <code>Math.random()</code>    | Math 객체의 random() 메서드    |

표 7-2 객체의 속성값 변경

|  |                                  |
|--|----------------------------------|
| <code>style.color = "red"</code>       | style 객체 color 속성을 red로 변경       |
| <code>style.background = "aqua"</code> | style 객체 background 속성을 aqua로 변경 |

# 01 자바스크립트 객체 이해하기

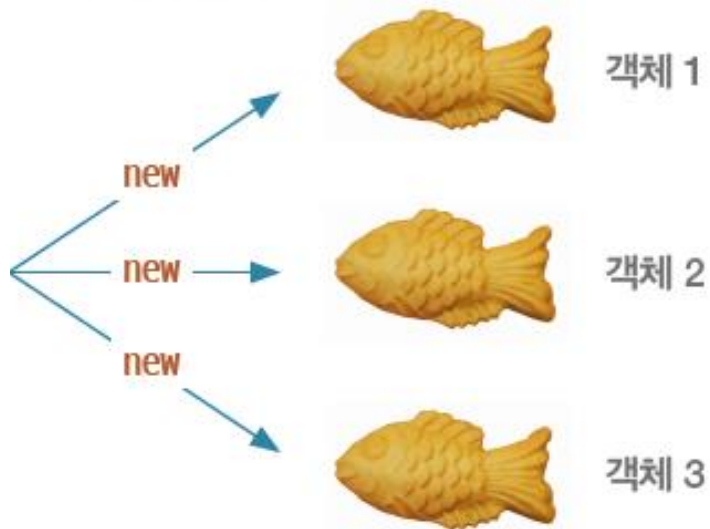
## ■ 자바스크립트 객체

### 3. 객체 선언과 객체 생성

#### 객체 선언

```
class 금붕어 {  
  
    크기  
    색  
    ...  
  
    헤엄친다()  
    호흡한다()  
    ...  
}
```

#### 객체 생성



new 키워드로  
객체를 생성해.



# 01 자바스크립트 객체 이해하기

## ■ 자바스크립트 객체

### 4. 전역객체

- HTML에서 최상위 객체는 window

- 전역객체 메서드

표 7-3 전역객체 메서드

| 전역객체 메서드                         | 객체 이름 생략                  | 내용                            |
|----------------------------------|---------------------------|-------------------------------|
| <code>window.alert()</code>      | <code>alert()</code>      | 알림 메시지                        |
| <code>window.prompt()</code>     | <code>prompt()</code>     | 사용자로부터 입력                     |
| <code>window.confirm()</code>    | <code>confirm()</code>    | 확인/취소 버튼(확인= true, 취소= false) |
| <code>window.eval()</code>       | <code>eval()</code>       | 수식을 계산하여 결과를 반환               |
| <code>window.parseInt()</code>   | <code>parseInt()</code>   | 정수로 변환                        |
| <code>window.parseFloat()</code> | <code>parseFloat()</code> | 실수로 변환                        |
| <code>window.isNaN()</code>      | <code>isNaN()</code>      | 숫자이면 false, 숫자가 아니면 true      |

## style.css를 만들어 놓으세요. (쓰고 싶을 때만 link)

---

```
body {  
  font-family: consolas;  
  font-size: 19px;  
  color: #004;  
  padding: 12px 25px;  
  border: 2px solid #cef;  
  border-radius: 15px;  
}  
  
hr {  
  border: 1px dotted #fea;  
}
```

# 예제 1.html

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8"> <title>전역객체 메서드 사용하기</title>
  <link rel="stylesheet" href="style.css">
</head> <body>
  <script>
    const jud = confirm("confirm() 앞에 window.이 생략된 것을 이해했나요?");
    document.write("이렇게 대답하셨습니다: " + (jud==true?'예':'아니오') + "<hr>");
    document.write(isNaN(1) + "<hr>");
    document.write(isNaN('0.5') + "<hr>");
    document.write( isNaN('str') + "<hr>");
    document.write( eval(3 ** 4));    // document.write( 3 ** 4 );
  </script>
</body>
</html>
```

이렇게 대답하셨습니다: 예

false

false

true

81

# 01 자바스크립트 객체 이해하기

## ■ 배열과 Array 객체

### - 배열 생성하기

```
let arr = [10, 20, 30, 40, 50];  
document.write(arr);
```



10,20,30,40,50

```
let arr = [1, "첫번째", 2.34, true];  
document.write(arr);
```



1,첫번째,2.34,true



# 01 자바스크립트 객체 이해하기

## ■ 배열과 Array 객체

- 배열 원소에 접근하기

```
let a = [10, 20, 30, 40, 50];  
for (i=0; i<5; i++)  
    document.write( a[i]*10 + ' ');
```

```
100 200 300 400 500
```

# 01 자바스크립트 객체 이해하기

## ■ 배열과 Array 객체

### - 빈 배열 선언, 인덱스 범위

```
let a = [], sum = 0    // 세미콜론? 쉼표?
```

```
a[0]=3.141592, a[1] = 100, a[2] = "Hello"
```

```
document.write( a+"<br>" )
```

```
for(i=0; i<=a.length; i++) {
```

```
    document.write(` a[${i}] = ${a[i]} <br>` )    // a[변수] = 변수 <br>
```

```
    sum += a[i]
```

```
}
```

```
document.write( "sum: " + sum )
```

```
3.141592,100,Hello
```

```
a[0] = 3.141592
```

```
a[1] = 100
```

```
a[2] = Hello
```

```
a[3] = undefined
```

```
sum: 103.141592Helloundefined
```

# 01 자바스크립트 객체 이해하기

표 7-5 Array 객체 메서드

| 메서드                             | 내용   |
|---------------------------------|--|
| <code>arr1.concat(arr2)</code>  | arr1 배열 뒤에 arr2 배열을 합치고 그 복사본을 반환함   |
| <code>filter(function())</code> | function()의 조건에 맞는 배열의 복사본을 반환함  |
| <code>indexOf(x, i)</code>      | i번째 인덱스 원소부터 시작하여 문자열 x가 처음 나타나는 위치를 찾아 반환하고 x가 없으면 -1을 반환함(i를 생략하면 처음부터 찾음) |
| <code>join(s)</code>            | 배열의 모든 원소를 합친 문자열을 반환(괄호 안에 s가 있으면 원소 사이에 s를 삽입하여 합친 문자열을 반환)                |
| <code>pop()</code>              | 배열 맨 뒤의 값을 삭제함   |
| <code>push(x)</code>            | 배열 맨 뒤에 x를 삽입함   |
| <code>reverse()</code>          | 배열의 원소 순서를 거꾸로 바꿈  |
| <code>shift()</code>            | 배열 맨 앞의 원소를 제거하고 그 값을 반환   |
| <code>slice(ix, iy)</code>      | ix부터 iy 직전까지의 복사본을 반환함(iy를 생략하면 ix부터 끝까지의 복사본을 반환)                           |
| <code>sort()</code>             | 모든 원소를 정렬함   |
| <code>toString()</code>         | 배열의 각 원소를 문자열로 바꾸어 반환함   |
| <code>unshift(x)</code>         | 배열 앞부분에 x값을 삽입함  |

## 예제2.html (1)

```
<!DOCTYPE html><html>
<head> <meta charset="UTF-8">
  <title>Array 객체의 메서드 사용하기</title>
</head> <body>
  <script>
    let a1 = [0, 1, 2, 3, 4];
    let a2 = new Array(5,6,7,8,9);
    document.write(a1 + '<br>');
    document.write(a2 + '<hr>');

    document.write(a1.concat(a2) + '<br>');
    function gt(a) { return(a >= 7) };
    document.write(a2.filter(gt) + '<br>');
    document.write(a2.indexOf(6) + '<br>');
    document.write(a2.slice(1,4) + '<br>');
    document.write(a2.join("+") + '<hr>');
```

---

0,1,2,3,4

5,6,7,8,9

---

0,1,2,3,4,5,6,7,8,9

7,8,9

1

6,7,8

5+6+7+8+9

---

## 예제2.html (2)

```
document.write(a2.reverse() + '<br>');  
a2.push(4);      document.write(a2 + '<br>');  
a2.pop();        document.write(a2 + '<br>');  
a2.shift();      document.write(a2 + '<br>');  
a2.unshift(10); document.write(a2 + '<hr>');
```

9,8,7,6,5  
9,8,7,6,5,4  
9,8,7,6,5  
8,7,6,5  
10,8,7,6,5

---

```
document.write(a2.sort() + '<br>');  
document.write(a2.sort( (a,b) => a-b ) + '<hr>');
```

10,5,6,7,8  
5,6,7,8,10

---

```
</script>  
</body>  
</html>
```

# 01 자바스크립트 객체 이해하기

## - String 객체

```
let s = "문자열"
```

```
document.write(s[0] + ';' + s[1] + ';' + s[2] + '<br>')
```

```
document.write(s.length)
```

---

문,자,열  
3

# 01 자바스크립트 객체 이해하기

## - String 객체의 메서드

표 7-6 String 객체 메서드

| 메서드                            | 내용  |
|--------------------------------|---|
| <code>charAt(i)</code>         | i번째 인덱스의 문자를 반환함  |
| <code>str1.concat(str2)</code> | str1에 str2 문자열을 합침  |
| <code>indexOf(s, i)</code>     | i번째부터 시작하여 문자 s가 처음 나타나는 위치를 찾아 반환하고 s가 없으면 -1을 반환함(i를 생략하면 처음부터 찾음)              |
| <code>replace(s, t)</code>     | 문자 s를 문자 t로 변경함   |
| <code>search(s)</code>         | 문자 s가 처음 나타나는 위치를 찾아 반환함. <code>indexOf()</code> 와 기능이 같지만 시작 위치를 지정할 수 없다는 점이 다름 |

# 01 자바스크립트 객체 이해하기

|                                |  |
|--------------------------------|--|
| <code>slice(ix, iy)</code>     | ix부터 iy 직전까지의 문자열을 반환함(iy를 생략하면 ix부터 끝까지 반환). <code>substring()</code> 과 기능이 같지만 iy가 음수일 수 있다는 점이 다름 |
| <code>split(s, limit)</code>   | s를 분리자로 하여 문자열을 분리하고 배열을 반환함. limit는 반환되는 배열 크기를 제한함(" "를 입력하면 한 문자씩 분리)                             |
| <code>substr(i, len)</code>    | i번째 인덱스부터 len만큼의 문자열을 배열로 반환함(len을 생략하면 끝까지 반환)  |
| <code>substring(ix, iy)</code> | ix부터 iy 직전까지의 배열을 반환함(iy를 생략하면 ix부터 끝까지 반환). <code>slice()</code> 와 비슷하지만 iy는 언제나 양수라는 점이 다름         |
| <code>toLowerCase()</code>     | 소문자로 변환함   |
| <code>toUpperCase()</code>     | 대문자로 변환함   |
| <code>trim()</code>            | 양끝의 공백 문자를 제거함   |



## 예제3.html (1)

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>String 객체의 메서드 사용하기</title>
<body>
  <script>
    const s1 = new String("OSS기반 ");
    const s2 = "AI Programming";
    document.write(` s1 = "${s1}"<br>`);
    document.write(` s2 = "${s2}"<hr>`);

    document.write(` ${s1.charAt(0)}<br>`);
    document.write(` ${s1.concat(s2)}<br>`);
    document.write(` ${s2.indexOf("i")}<hr>`);
```

```
s1 = "OSS기반 "
s2 = "AI Programming"
```

```
0
OSS기반 AI Programming
11
```

## 예제3.html (2)

```
document.write(` ${s2.replace("m", "M")}<br>` );  
document.write(` ${s2.search("i")}<br>` );  
document.write(` ${s2.split("a")}<hr>` );
```

```
document.write(` ${s2.substr(3, 5)}<br>` );  
document.write(` ${s2.substring(3, 5)}<br>` );  
document.write(` ${s2.substr(3)}<br>` );  
document.write(` ${s2.substring(3)}<br>` );  
document.write(` ${s2.slice(3, -2)}<hr>` );
```

```
document.write(` ${s2.toLowerCase()}<br>` );  
document.write(` ${s2.toUpperCase()}<br>` );  
document.write(` "${s1.trim()}"` );
```

```
</script>  
</body>  
</html>
```

AI PrograMming  
11  
AI Progr,mming

Progr  
Pr  
Programming  
Programming  
Programmi

ai programming  
AI PROGRAMMING  
"OSS기반"

# 01 자바스크립트 객체 이해하기

## - String 객체의 메서드

```
let s = "Programmng"  
document.write( s.split('a') + '<br>')  
document.write( s.split("") + '<br>')  
document.write( s.split('a',1) ) // 배열크기 제한
```

표 7-7 문자열의 음수 인덱스

|        |     |     |    |    |    |    |    |    |    |    |    |
|--------|-----|-----|----|----|----|----|----|----|----|----|----|
| 인덱스    | 0   | 1   | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | 10 |
| 문자     | P   | r   | o  | g  | r  | a  | m  | m  | i  | n  | g  |
| 음수 인덱스 | -11 | -10 | -9 | -8 | -7 | -6 | -5 | -4 | -3 | -2 | -1 |

## 예제4.html

```
<!DOCTYPE html>
<head>
  <meta charset="UTF-8">
  <title>문자열 회전시키기</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <script>
    let s = new String("Programming")
    let firstChar

    for (let i = 0; i <= s.length; i++) { // 오타 아님
      document.write(s + "<br>")
      firstChar = s[0]
      s = s.slice(1) + firstChar
    }
  </script>
</body>
</html>
```

Programming  
rogrammingP  
ogrammingPr  
grammingPro  
rammingProg  
ammingProgr  
mmingProgra  
mingProgram  
ingProgramm  
ngProgrammi  
gProgrammin  
Programming

# 01 자바스크립트 객체 이해하기

## ■ 문자열과 Array 객체

### 2. for-of 반복문

```
const dayString = "일월화수목금토";  
  
for (let day of dayString)  
  document.write(day + "요일, ");
```

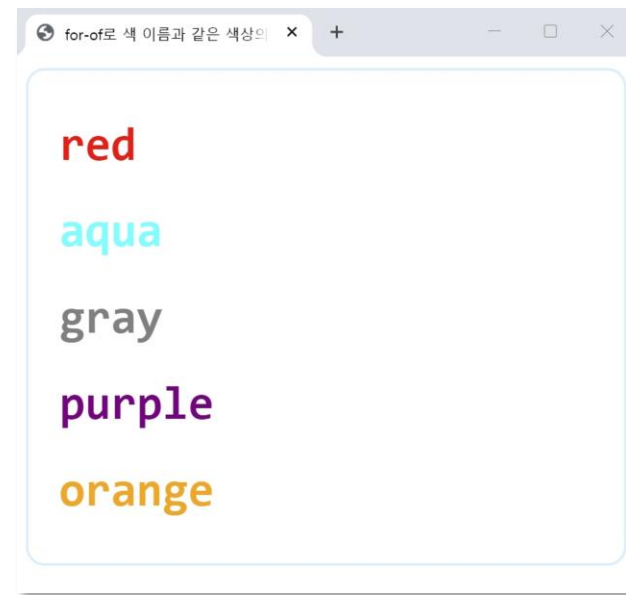


일요일, 월요일, 화요일, 수요일, 목요일, 금요일, 토요일,

# 예제5.html

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>for-of로 색 이름과 같은 색상의 문자 만들기</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <script>
    const colorArray = ["red", "aqua", "gray", "purple", "orange"];

    for (let co of colorArray)
      document.write("<h1 style='color:" + co + "'>" + co +
"</h1>");
  </script>
</body>
</html>
```



- 
- 내용이 더 추가됩니다.