



배열

안 화 수

❖ 자바스크립트 배열

- 한가지 자료형의 데이터를 저장하기 위한 정적인 자료구조
- 자바스크립트의 배열 형식

1. `var str=new Array();`

`str[0]=80;`

`str[1]=90;`

`str[2]=70;`

2. `var str=new Array(80, 90, 70);`

3. `var str=[80, 90, 70];`

배열

```
<script>
    // 배열

    // 방법1.
    var a1 = new Array();
    a1[0] = 80;
    a1[1] = 90;
    a1[2] = 70;

    // length : 배열의 크기를 구해주는 속성
    for(var i=0; i<a1.length; i++){
        document.write(a1[i]+"\\t");
    }
    document.write("<br>");

    // 방법2.
    var a2 = new Array(80, 90, 70);

    for(var i=0; i<a2.length; i++){
        document.write(a2[i]+"\\t");
    }
    document.write("<br>");

    // 방법3.
    var a3 = [80, 90, 70];

    for(var i=0; i<a3.length; i++){
        document.write(a3[i]+"\\t");
    }
</script>
```

배열의 속성

❖ 자바스크립트 배열의 속성

- length : 배열의 크기

배열의 메소드

❖ 자바스크립트 배열의 메소드

join() : 문자('.')를 기준으로 배열 값들을 하나의 문자 데이터로 결합 해주는 함수

sort() : 배열 원소 값들을 오름차순으로 정렬 (문자 : 사전순 정렬) 해주는 함수
원본 배열의 원소들이 오름차순으로 정렬되어 있다.

reverse() : 배열 원소 값들의 순서를 반대로 바꾸어 주는 함수
원본 배열의 순서가 반대로 바뀐다.

slice(start index, end index) : start ~ end-1번 원소를 추출 해주는 함수

concat() : 2개의 배열 객체를 하나로 결합 해주는 함수

pop() : 배열의 마지막 원소를 삭제 해주는 함수

push() : 배열의 마지막 인덱스에 새로운 데이터를 삽입 해주는 함수

shift() : 배열의 첫번째 원소를 삭제 해주는 함수

unshift() : 배열의 첫번째 인덱스에 새로운 데이터를 삽입 해주는 함수

배열의 메소드

```
<script>
    var num = ['사당', '교대', '방배', '강남'];
    document.write(num+"<br>");           // 사당, 교대, 방배, 강남
    document.write(typeof(num)+"<br>");   // object

    // join() : 문자('-')를 기준으로 배열 값들을 하나의 문자 데이터로 결합
    document.write(num.join('-')+"<br>");   // 사당-교대-방배-강남
    document.write(typeof(num.join('-'))+"<br>");// string

    // reverse() : 배열 원소 값들의 순서를 반대로 바꾸어 주는 함수
    // 원본 배열의 순서가 바뀐다.
    document.write(num.reverse()+"<br>"); // 강남, 방배, 교대, 사당
    // document.write(typeof(num.reverse())+"<br>");// object
    document.write(num+"<br>");           // 강남, 방배, 교대, 사당

    // sort() : 배열 원소 값들을 오름차순으로 정렬 (문자 : 사전순 정렬)
    // 원본 배열의 원소들이 오름차순으로 정렬되어 있다.
    document.write(num.sort()+"<br>");     // 강남, 교대, 방배, 사당
    document.write(num+"<br>");           // 강남, 교대, 방배, 사당

    // 배열 원소 값들을 내림차순 정렬 (문자 : 사전역순 정렬)
    document.write(num.sort().reverse()+"<br>");// 사당, 방배, 교대, 강남
</script>
```

배열의 메소드

```
<script>
var greenLine = ['사당', '교대', '방배', '강남'];
var yellowLine = ['미금', '정자', '모란', '수서'];

// slice(start index, end index) : start ~ end-1번 원소를 추출
document.write(greenLine.slice(1,3)+"<br>"); // 교대, 방배

// concat() : 2개의 배열 객체를 하나로 결합 해주는 함수
document.write(greenLine.concat(yellowLine) + "<br>");
// 사당, 교대, 방배, 강남, 미금, 정자, 모란, 수서

// pop() : 배열의 마지막 원소를 삭제 해주는 함수
greenLine.pop(); // '강남' 삭제
document.write(greenLine+"<br>"); // 사당, 교대, 방배

// push() : 배열의 마지막 인덱스에 새로운 데이터를 삽입 해주는 함수
greenLine.push('삼성'); // '삼성' 삽입
document.write(greenLine+"<br>"); // 사당, 교대, 방배, 삼성

// shift() : 배열의 첫번째 원소를 삭제 해주는 함수
greenLine.shift(); // '사당' 삭제
document.write(greenLine+"<br>"); // 교대, 방배, 삼성

// unshift() : 배열의 첫번째 인덱스에 새로운 데이터를 삽입 해주는 함수
greenLine.unshift('신도림');
document.write(greenLine+"<br>"); // 신도림, 교대, 방배, 삼성
</script>
```