

배열의 메소드

toString 메소드

```
let a1 = [1, 2, 3];
let a2 = ["a", "b", "c"];

console.log(a1.toString()); //1, 2, 3
console.log(a2.toString()); //a, b, c
```

배열 객체의 toString 메소드는, 배열의 내부 값을 표현하는 string을 리턴한다.

splice 메소드

splice 메소드의 파라미터

배열.splice(삽입할_위치, 삭제할_항목_수, 삽입할_값_목록)

```
let a = [0, 1, 2];
a.splice(1, 0, "a");
console.log(a); //[ 0, 'a', 1, 2 ]
```

a.splice(1, 0, "a");

a 배열의 인덱스 1 위치에서, 0 개의 항목을 삭제하고, "a" 항목을 삽입한다.

```
let a = [0, 1, 2];
a.splice(1, 0, "a", "b");
console.log(a); //[ 0, 'a', 'b', 1, 2 ]
```

a.splice(1, 0, "a", "b");

a 배열의 인덱스 1 위치에서, 0 개의 항목을 삭제하고, "a", "b" 항목을 삽입한다.

```
let a = [0, 1, 2];
a.splice(1, 1, "a");
console.log(a); //[ 0, 'a', 2 ]
```

a.splice(1, 1, "a");

a 배열의 인덱스 1 위치에서, 1 개의 항목을 삭제하고, "a" 항목을 삽입한다.

```
let a = [0, 1, 2];
a.splice(1, 2, "a");
console.log(a); //[ 0, 'a' ]
```

a.splice(1, 2, "a");

a 배열의 인덱스 1 위치에서, 2 개의 항목을 삭제하고, "a" 항목을 삽입한다.

```
let a = [0, 1, 2];
a.splice(1, 1);
console.log(a); //[ 0, 2 ]
```

a.splice(1, 1);

a 배열의 인덱스 1 위치에서, 1 개의 항목을 삭제하고, 아무것도 삽입하지 않는다.

요약: 배열에 항목 한 개 삽입하기

배열.splice(삽입할_위치, 0, 삽입할_값);

```
let a = [0, 1, 2]
a.splice(1, 0, "a");
console.log(a);
```

요약: 배열에서 항목 한 개 삭제하기

배열.splice(삭제할_위치, 1);

```
let a = [0, 1, 2]
a.splice(1, 1);
console.log(a);
```

slice 메소드

slice 메소드의 파라미터

.slice(시작_위치, 끝_위치)

slice 메소드는 배열의 부분을 리턴한다.

리턴되는 부분 배열은, 원본 배열에서 시작_위치에서 끝_위치 사이의 항목들이다.

(시작위치 <= 위치 < 끝 위치)

원본 배열은 변경되지 않는다.

끝_위치 파라미터를 생략하면, 배열의 끝까지이다.

```
let a = [0, 1, 2, 3];
console.log(a.slice(0, 1)); //[ 0 ]
console.log(a.slice(0, 2)); //[ 0, 1 ]
console.log(a.slice(1, 2)); //[ 1 ]
console.log(a.slice(1, 3)); //[ 1, 2 ]
```

배열의 복제

```
let b = a.slice(0);
```

slice 메소드에 의해 생성되는 부분 배열은, a 배열의 처음부터(인덱스 0부터) 끝까지이다. 즉 a 배열이 복제되어 새 배열이 생성되고, 새 배열에 대한 참조가 지역 변수 b에 대입된다.

push, pop 메소드

```
let a = [];

for (let i = 0; i < 5; ++i)
  a.push(i);

console.log(a.toString()); //0,1,2,3,4
```

배열 객체의 push 메소드는, 배열의 끝에 항목을 추가한다.
 배열의 크기는 1 증가한다.
 push 메소드의 리턴 값은 배열의 새 크기이다.

```
let a = [1, 3, 5, 7];

while (a.length > 0) {
  let value = a.pop();
  console.log(value, a.toString());
}
```

```
7 "1,3,5"
5 "1,3"
3 "1"
1 ""
```

배열 객체의 pop 메소드는, 배열의 끝의 항목을 배열에서 제거하여 리턴한다.
 배열의 크기는 1 감소한다.

unshift, shift 메소드

```
let a = [];
```

```
for (let i = 0; i < 5; ++i)
  a.unshift(i);

console.log(a.toString()); //4,3,2,1,0
```

배열 객체의 unshift 메소드는, 배열의 선두에 항목을 추가한다.
 배열의 크기는 1 증가한다.
 unshift 메소드의 리턴 값은 배열의 새 크기이다.

```
let a = [1, 3, 5, 7];

while (a.length > 0) {
  let value = a.shift();
  console.log(value, a.toString());
}
```

```
1 "3,5,7"
3 "5,7"
5 "7"
7 ""
```

배열 객체의 shift 메소드는, 배열의 선두의 항목을 배열에서 제거하여 리턴한다.
 배열의 크기는 1 감소한다.

concat 메소드

두 배열을 결합한 새 배열을 생성하여 리턴한다.

```
let a1 = [1, 2];
let a2 = [3, 4];
let a3 = a1.concat(a2);
console.log(a1); //[ 1, 2 ]
```

```
console.log(a2); //[ 3, 4 ]  
console.log(a3); //[ 1, 2, 3, 4 ]
```

```
let a3 = a1.concat(a2);
```

a1 배열과 a2 배열을 결합한 새 배열을 생성하여 리턴한다.

a1 배열과 a2 배열은 변경되지 않는다.

sort

```
let a = [3, 1, 6, 2, 4, 8, 10, 5, 11, 7, 9];  
a.sort();  
console.log(a); //[ 1, 10, 11, 2, 3, 4, 5, 6, 7, 8, 9 ]
```

배열의 sort 메소드는 배열의 항목들을 정렬한다.

배열에 들어있는 값이 숫자이지만, 문자열인 것처럼 정렬한다.

그래서 정렬한 후 2 보다 10, 11 이 앞에 위치한다.

```
let a = [3, 1, 6, 2, 4, 8, 10, 5, 11, 7, 9];  
a.sort(compareNumber);  
console.log(a); //[ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 ]  
  
function compareNumber(i, j) {  
    return i - j;  
}
```

sort 메소드의 파라미터로 비교 함수를 전달하면,
이 비교 함수를 기준으로 정렬된다.

비교 함수는 정렬할 두 값을 비교한다.

첫째 파라미터가 둘째 파라미터 보다

크면(정렬 순서상 뒤이면) 양수를 리턴하고,

작으면 (정렬 순서상 앞이면) 음수를 리턴하고,
두 값이 같으면 0을 리턴해야 한다.

```
let a = [3, 1, 6, 2, 4, 8, 10, 5, 11, 7, 9];
a.sort(compareNumber);
console.log(a); //[ 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1 ]

function compareNumber(i, j) {
  return j - i;
}
```

비교함수 내부에서 i, j 순서를 바꿔서, 내림차순 정렬을 구현하였다.

reverse

```
reverse1.js
let a = [];

for (let i = 0; i < 5; ++i)
  a.push(i)

a.reverse()
console.log(a) //[ 4, 3, 2, 1, 0 ]
```

배열의 reverse 메소드는 배열의 항목들의 순서를 뒤집는다.

연습 문제

cars 배열 변수에 3개의 원소가 초기값으로 설정 되어 있습니다.

아래의 기능을 수행하는데 필요한 배열 객체의 메소드를 생각해 보세요

```
var cars = ['소형차', '중형차', '대형차'];
```

콘솔	
top ▼	기본 수준 ▼
1. 스포츠카, 소형차, 중형차, 대형차	10.연습문제2.html:12
2. 스포츠카, 소형차, 중형차, 대형차, 버스	10.연습문제2.html:15
3. 스포츠카, 소형차, 대형차, 버스	10.연습문제2.html:18
4. 대형차, 버스, 소형차, 스포츠카	10.연습문제2.html:21

답