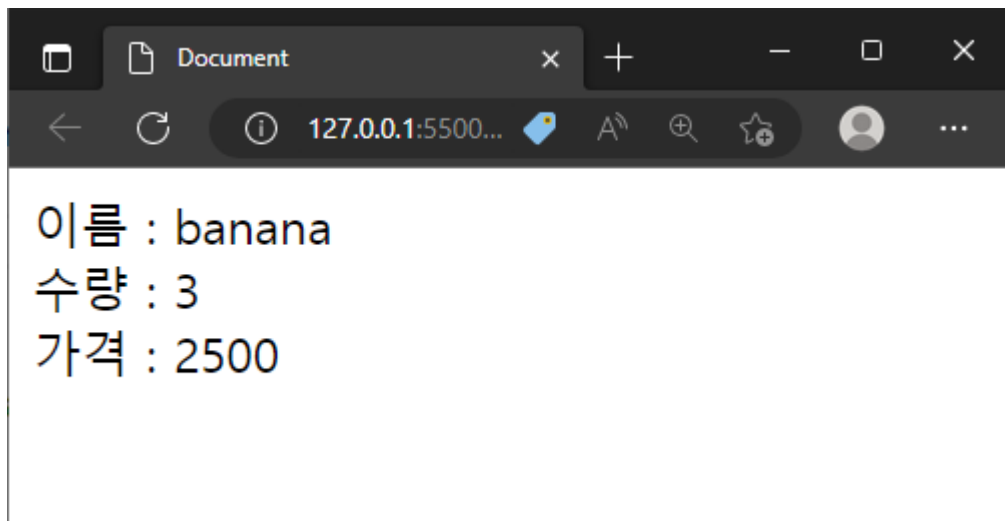


자바스크립트 실습 #고차함수

1. find() 함수를 이용하여 특정 과일을 검색한 후 결과를 출력하는 프로그램을 만들어보세요.

- 조건 : 여러 개의 과일 정보를 객체에 저장

* 실행 결과



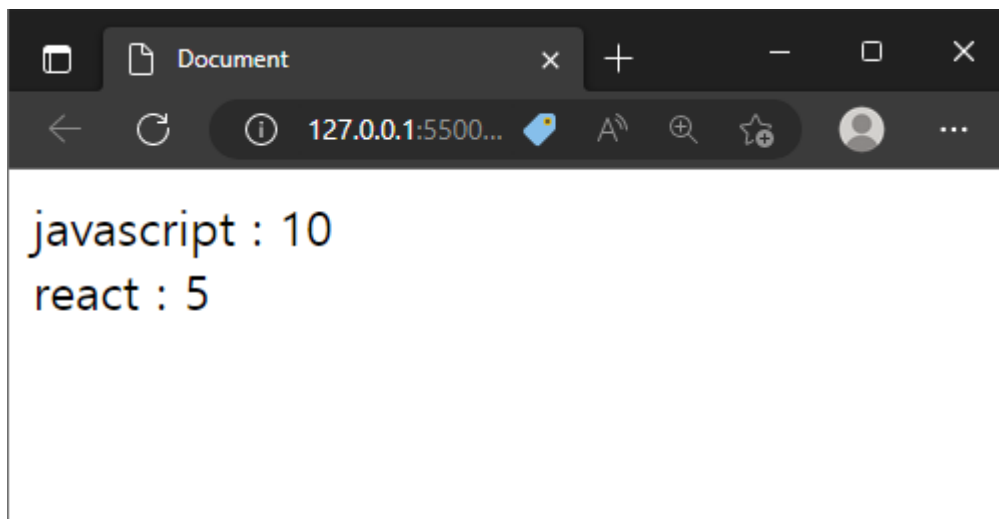
2. 다음 배열의 아이템 중 글자수가 5자 이상 되는 아이템을 검색하세요.

* 조건

- find() 함수를 이용
- 배열의 아이템과 글자수를 함께 출력

```
const array = ["javascript", "css", "react", "html"];
```

* 실행 결과



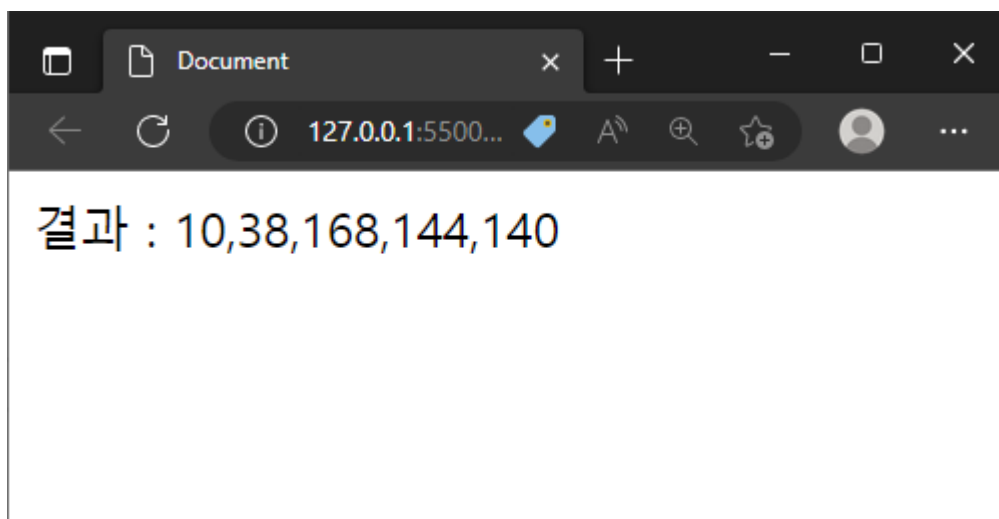
3. 다음 조건을 만족하는 프로그램을 만들어보세요.

* 조건

- map() 함수를 이용
- 배열 안 아이템을 각각 2배 곱해서 저장
- 새로운 배열의 아이템이 3의 배수이면 2배, 그렇지 않으면 그냥 출력
- 배열

```
const numbers = [5, 19, 42, 36, 70];
```

* 실행 결과



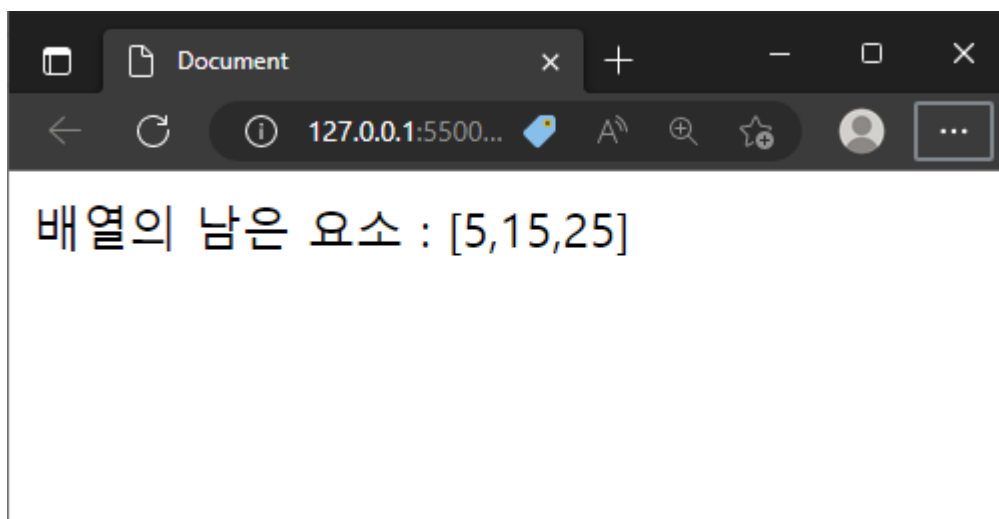
4. 다음 조건을 만족하는 프로그램을 만들어보세요.

* 조건

- forEach() 함수를 이용
- 배열 안 아이템 중 짝수를 제거
- 새로운 배열을 생성하지 않고, 원본 배열을 수정할 것
- 배열

```
const numbers = [5, 10, 15, 20, 25, 30];
```

* 실행 결과



5. 다음 조건을 만족하는 프로그램을 만들어보세요.

* 조건

- filter() 함수를 이용
- 아래 코드를 참고(수정해도 됨!)

```
class Users {  
  constructor() {  
    this.members = [  
      { id: 1, name: "soo", addr: "seoul" },  
      { id: 2, name: "Lee", addr: "busan" },  
      { id: 3, name: "Kim", addr: "seoul" },  
    ];  
  }  
}
```

* 실행 결과

- ID가 1인 고객 정보 출력
- 지역이 seoul이 아닌 고객을 제거 후, 전체 고객 정보 출력



6. 다음 코드를 실행하고 결과를 출력해보세요.

* for...in문 : 객체의 모든 프로퍼티를 순회하며 열거

```
for ( 변수선언문 in 객체 ) {  
    // 코드  
}
```

```
const person = {  
    name: "soo",  
    age: 20,  
    addr: "seoul",  
};  
  
// for ... in문의 변수에 person 객체의 key가 할당  
for (const key in person) {  
    document.write(`${key} : ${person[key]}<br>`);  
}
```

7. 다음 코드를 실행하고 결과를 출력해보세요.

* for...of문 : 이터러블(배열, 문자열 등)을 순회하면서 요소를 변수에 할당

```
for ( 변수선언문 of 이터러블){  
    // 코드  
}
```

```
const array = [10, 20, 30, 40, 50];  
  
// for ... of문 변수에 array의 원소가 순차적으로 할당  
for (const item of array) {  
    document.write(`${item} `);  
}
```

```
const say = "Hello";  
  
// for ... of문 변수에 say의 글자가 순차적으로 할당  
for (const item of say) {  
    document.write(`${item} `);  
}
```

8. 다음 코드를 실행하고 결과를 출력해보세요.

```
const map = new Map();

const lee = { name: "Lee" };
const kim = { name: "Kim" };

// 객체도 key로 사용 가능
map.set(lee, "developer");
map.set(kim, "designer");

console.log(map);

console.log(map.get(lee));

console.log(map.has(kim));
console.log(map.has("soo"));

map.delete(lee);
map.clear();

console.log(map);
```


9. 다음과 같은 결과를 출력하는 프로그램을 만들어보세요.

* Map 객체에 적용할 수 있는 메서드

메서드	설명
keys()	Map 객체의 key를 반환
values()	Map 객체의 value를 반환

* 조건 : 다음 코드와 for...of문을 이용할 것

```
const map = new Map();

const lee = { name: "Lee" };
const kim = { name: "Kim" };

map.set(lee, "developer");
map.set(kim, "designer");
```

* 실행 결과

```
Map 객체의 key
{ name: 'Lee' }
{ name: 'Kim' }

Map 객체의 value
developer
designer
```