



1 변수

2 문자열

3 조건문, 반복문

4 함수

1. 변수

✓ 변수란?

- 데이터를 저장하고 관리하는 공간
- 값을 할당하고 변경이 가능

✓ 변수 선언 방법

- var : 최초에 사용된 변수 선언 방법
- let : 블록 스코프(Block Scope)를 가지고 재할당이 가능하고 재선언을 불가능
- const : 블록 스코프(Block Scope)를 가지며 재할당 및 재선언 불가
- JavaScript 는 데이터 타입을 명시하지 않음

```
var name = "홍길동";
console.log(name); // 홍길동
var name = "이순신"; // 재선언 가능
console.log(name); // 이순신
```

```
let age = 25;
console.log(age); // 25
age = 30; // 재할당 가능
console.log(age); // 30
```

```
const pi = 3.14;
console.log(pi); // 3.14
// pi = 3.1415; // 오류 발생 (재할당 불가)
```

2. 문자열

✓ 문자열

- 문자열은 텍스트 데이터를 다루기 위한 기본 자료형이다.
- 작은 따옴표('), 큰따옴표("), 백틱(`)을 사용하여 문자열을 생성한다.

```
let str1 = 'Hello';
let str2 = "World";
let str3 = `JavaScript`;
```



✓ 문자열 메서드

■ 길이 확인 (length)

```
let text = "Hello, JavaScript!";
console.log(text.length); // 18
```

■ 문자열 결합(+, concat())

```
let first = "Hello";
let second = "World";
console.log(first + " " + second); // "Hello World"
console.log(first.concat(" ", second)); // "Hello World"
```



- 부분 문자열 추출
 - > slice(start, end), substring(start,end), substr(start, length)

```
let text = "JavaScript";
console.log(text.slice(0, 4)); // "Java"
console.log(text.substring(4, 10)); // "Script"
console.log(text.substr(4, 6)); // "Script"
```

■ 문자열 대소문자 변환 (toUpperCase(), toLowerCase())

```
let message = "Hello World";
console.log(message.toUpperCase()); // "HELLO WORLD"
console.log(message.toLowerCase()); // "hello world"
```



- 문자열 포함 여부 확인
 - includes(), indexOf(), startsWith(), endsWith()

```
let sentence = "Learning JavaScript is fun!";
console.log(sentence.includes("JavaScript")); // true
console.log(sentence.indexOf("JavaScript")); // 9
console.log(sentence.startsWith("Learning")); // true
console.log(sentence.endsWith("fun!")); // true
```

■ 문자열 공백 제거(trim(), trimStart(), trimEnd())

```
let text = " JavaScript ";
console.log(text.trim()); // "JavaScript"
console.log(text.trimStart()); // "JavaScript "
console.log(text.trimEnd()); // " JavaScript"
```

■ 문자열을 배열로 변환 (split())

```
let str = "apple,banana,grape";
let fruits = str.split(",");
console.log(fruits); // ["apple", "banana", "grape"]
```

■ 문자열을 숫자로 변환

```
let numStr = "123";
console.log(parseInt(numStr)); // 123
console.log(parseFloat("3.14")); // 3.14
console.log(Number("100")); // 100
```

■ 문자열 치환 (replace())

```
let text = "I love Java!";
console.log(text.replace("Java", "JavaScript")); // "I love JavaScript!"
```



■ 템플릿 리터럴(백틱 사용)

```
let name = "Alice";
let greeting = `Hello, ${name}!`;
console.log(greeting); // "Hello, Alice!"
```

문제

- 1. 문자열 "Hello, JavaScript! " 에서 "JavaScript"만 추출하는 코드 작성
- 2. 문자열 "Coding is fun!"의 앞뒤 공백을 제거하는 코드 작성
- 3. "apple,orange,banana" 문자열을 배열로 변환하는 코드 작성

3. 조건문, 반복문

✓ 조건문

■ if 문

```
let age = 20;
if (age >= 18) {
    console.log("성인입니다.");
}
```

• if ~ else 문

```
let score = 75;
if (score >= 80) {
    console.log("좋은 점수입니다.");
} else {
    console.log("더 노력하세요!");
}
```



■ if ~ else if ~ else 문

```
let temp = 30;
if (temp > 35) {
    console.log("너무 덥습니다.");
} else if (temp > 25) {
    console.log("적당한 온도입니다.");
} else {
    console.log("조금 쌀쌀합니다.");
}
```

■ 삼항 연산자

```
let num = 10;
let result = (num % 2 === 0) ? "짝수" : "홀수";
console.log(result); // "짝수"
```



■ switch 문

```
let day = "월요일";
switch (day) {
    case "월요일":
        console.log("한 주의 시작!");
        break;
    case "금요일":
        console.log("주말이 곧 옵니다!");
        break;
    default:
        console.log("평범한 날입니다.");
}
```



✓ 반복문

■ for 문

```
for (let i = 1; i <= 5; i++) {
    console.log("반복: " + i);
}
```

• while 문

```
let count = 0;
while (count < 3) {
    console.log("Count: " + count);
    count++;
}</pre>
```



• do ~ while 문

```
let number = 5;
do {
    console.log("会环: " + number);
    number--;
} while (number > 0);
```

• for ~ in 문

```
let person = {name: "Alice", age: 25, city: "Seoul"};
for (let key in person) {
   console.log(key + ": " + person[key]);
}
```



■ for ~ of 문

```
let fruits = ["apple", "banana", "cherry"];
for (let fruit of fruits) {
   console.log(fruit);
}
```

✓ 반복문 제어

• break 문

```
for (let i = 1; i <= 5; i++) {
    if (i === 3) break;
    console.log(i);
}
// 출력: 1, 2
```

■ continue 문

```
for (let i = 1; i <= 5; i++) {
    if (i === 3) continue;
    console.log(i);
}
// 출력: 1, 2, 4, 5
```

문제

- 1. 사용자 입력 숫자가 양수, 음수, 0 인지 판별
- 2. 1 부터 10까지 홀수만 출력하는 반복문 작성
- 3. 배열의 모든 요소를 출력하는 for ... of 반복문 작성

4. 함수

- ✓ 함수란?
 - 특정 작업을 수행하는 코드 블록
 - 코드의 재상요성을 높이고 유지보수를 쉽게 함
- ✓ 함수 선언 방법
 - function 키워드를 사용하여 함수를 선언함

```
function greet(name) {
return "안녕하세요, " + name + "님!";
}
console.log(greet("홈길동")); // 안녕하세요, 홈길동님!
```

✓ 함수 표현식

```
const greet = function(name) {
    return "안녕하세요, " + name + "님!";
};
console.log(greet("이순신")); // 안녕하세요, 이순신님!
```

✓ 화살표 함수

```
const greet = (name) => `안녕하세요, ${name}님!`;
console.log(greet("강감찬")); // 안녕하세요, 강감찬님!
```



- ✓ 함수의 매개변수 및 반환값
 - 기본 매개변수

```
function greet(name = "손님") {
    return `안녕하세요, ${name}님!`;
}
console.log(greet()); // 안녕하세요, 손님님!
console.log(greet("이순신")); // 안녕하세요, 이순신님!
```

■ 나머지 매개변수

```
function sum(...numbers) {
    return numbers.reduce((acc, num) => acc + num, 0);
}
console.log(sum(1, 2, 3, 4, 5)); // 15
```



✓ 콜백 함수

```
function processUserInput(callback) {
    let name = "홍길동";
    callback(name);
}
processUserInput(function(name) {
    console.log("안녕하세요, " + name + "님!");
});
```

