

바닐라 자바스크립트

chapter03

자바스크립트 기본 문법

제공된 자료는 훈련생의 수업을 돕기 위한 것으로, 타인과 공유하시면 안됩니다.

Contents

part.1

변수

part.2

데이터 타입

part.3

연산자

part.4

조건문과 반복문

part.5

함수

변수란 데이터를 저장하기 위해 공간을 생성하는 것을 말한다.
변수는 var, let, const 라는 키워드를 사용하여 선언할 수 있다.

const

const는 상수를 선언하는 키워드이다.

const 키워드로 선언된 변수는 값을 변경 할 수 없다.

```
const a = 12;  
a = 999;           //에러남
```

let

let 키워드로 선언된 변수는 값을 변경할 수 있다.

```
let b = 12;  
b = 999;
```

var는 const나 let이 나오기 전에 사용된 변수 선언 키워드로 호이스팅, 유효범위 등 다양한 이슈가 있어 사용하는 것을 권장하지 않는다.

다음 예시를 보면 let은 유효범위에 따라 전역변수와 지역변수 두개가 선언되었다.

```
let puppy = "cute";
{
  let puppy = "so cute";
}
console.log(puppy); // cute
```

메모리 공간

puppy

```
{
  puppy
}
```

하지만 var는 블록스코프{ }을 무시하기 때문에 변수가 한 개만 선언되었다.

```
var cat = "cute";
{
  var cat = "so cute";
}
console.log(cat); // so cute
```

cat

let 키워드를 사용하여 변수를 선언하면, 변수가 선언되기 전에 변수를 사용하려고 할 때 에러가 발생한다.
반면, var 키워드를 사용하여 변수를 선언하면, 변수가 선언되기 전에 변수를 사용해도 에러가 발생하지 않는다.
따라서 var 키워드로 변수를 선언하면, 문제가 발생할 수 있는 상황을 미리 방지하기 어렵다.

```
console.log(letKeyword);  
let letKeyword = 'let keyword';
```

letKeyword is not defined

```
console.log(varKeyword);  
var varKeyword = 'var keyword';
```

undefined

자바스크립트에서 사용되는 데이터 종류(자료형)는 다음과 같다.

String, Number, Boolean, Undefined, Null, Object, Array, Function

자바스크립트는 변수를 선언할 때 자료형을 지정하지 않는다.
값을 대입하는 순간 동적으로 변수의 자료형이 정해진다.

```
let puppy = "cute";    //String 타입  
puppy = 10;           //Number 타입
```

자료형	설명	예시
String	문자 데이터는 큰따옴표("), 작은따옴표('), 백틱(`)과 함께 표기한다.	"둘리" 'user@naver.com' `Hello \${myName}`
Number	숫자데이터는 정수와 소수를 표현한다.	100 1.123
Boolean	논리 데이터는 true(참)과 false(거짓)이라는 두 가지 값으로 구분된다. 데이터를 예 또는 아니오로 구분해야 하는 상황에서 사용한다.	true false
undefined	아무것도 정의되지 않는 상태를 의미한다. 변수에 값을 대입하지 않으면 자동으로 가지는 값이다.	
null	아무것도 없는 상태를 의미한다. undefined와 달리 개발자가 직접 대입해야 한다.	

자료형	설명	예시
Object	객체는 여러 개의 프로퍼티를 저장하는 구조이다. 키(key)와 값(value)의 쌍으로 데이터를 저장한다.	<div>{ name: '둘리', age: 10 };</div> <div>Json 형태</div>
Array	배열은 여러 데이터를 순서대로 나열하여 저장하는 구조이다. 데이터를 입력할 때는 대괄호[]를 사용한다.	['Apple', 'Banana', 'Cherry'];

Q1. 회원, 제품, 주문 정보를 담을 수 있는 객체를 만들어주세요.

회원	아이디
	패스워드
	등급
	이름
	전화번호

제품	제품명
	가격
	상세설명
	카테고리

주문	제품명
	수량
	총주문금액
	주문자
	주문일자
	배송지

Q2-1. 사람 객체의 정보를 수정하여, 아래와 같이 출력하세요.

```
const person = {  
  age : 30,  
  name : "둘리",  
  address : "서울"  
}
```

결과

10
또치
인천

Q2-2. 사람 객체의 정보를 수정하여, 아래와 같이 출력하세요.

```
const product = {  
  name: "노트북",  
  price: 1500000,  
  brand: "삼성",  
}
```

결과

노트북
2000000
LG

Q3. 다음 문자열 배열을 사용하여 첫번째 요소와 마지막요소를 합쳐서 출력하세요

```
const foodList = ['Pizza', 'Hamburger', 'Chicken', 'Steak'];
```

결과

Pizza Steak

연산자 종류	설명	예시
대입 연산자	오른쪽 값을 왼쪽 변수에 저장한다.	$x = 1$
비교 연산자	값들을 비교하고 논리값을 반환한다.	$x == y$ $x != y$ $x > y$ $x <= y$
산술 연산자	숫자값으로 계산되며 덧셈, 뺄셈, 곱셈, 나눗셈 연산이 있다.	$2 + 3$ $5 - 1$ $2 * 3$ $10 / 2$ $10 \% 2$
부호 연산자	값의 부호를 반대로 바꾼다.	$x = 3$ $-x \rightarrow -3$

연산자

연산자

연산자 종류	설명	예시
증감 연산자	아무것도 정의되지 않는 상태를 의미한다. 변수에 값을 대입하지 않으면 자동으로 가지는 값이다.	x++ x--
논리 연산자	boolean 값을 비교하여 참, 거짓을 반환한다	true && true true false !true
문자열 결합 연산자	문자열 여러 개를 연결하여 새로운 문자열을 반환한다	"바닐라" + "JS"
삼항연산자	조건에 따라 하나의 값을 반환한다	point = 92 grade = (point>=90) ? 'A' : 'B'

조건문은 조건에 따라 다른 코드를 실행하는 문법으로, if와 switch가 있다.

조건문은 조건이 참이면 코드 블록을 실행하고, 거짓이라면 실행하지 않는다.

```
if (조건식) {  
    실행할 코드 블록  
}
```

if 조건문 다음에 else 키워드를 추가하면 조건에 따라 실행할 코드 블록을 나눌 수 있다.

조건이 참일때, 거짓일때 각각 다른 코드 블록이 실행되도록 분기된다.

```
if(조건식){  
    조건을 만족할 때 실행할 코드 블록  
} else {  
    조건을 만족하지 않을 때 실행할 코드 블록  
}
```

조건문과 반복문

연습문제

Q4-1. 변수 x가 10보다 크고 20보다 작을 때 x를 출력하는 조건식을 작성하세요

Q4-2. 문자열 str의 길이가 5보다 길면 str을 출력하는 조건식을 작성하세요

Q4-3. 객체 person의 나이가 20살 미만이고 성별이 남자이면, "남학생입니다"를 출력하는 조건식을 작성하세요

```
const person = {  
  age: 15,  
  genger: "M",  
  name: "둘리",  
  address: "서울",  
};
```

————→ 남학생입니다

Q4-4. 배열에 숫자 10이 포함되어 있으면 "배열에 10이 있습니다"를 출력하고,
없으면 "10이 배열에 없습니다"를 출력하는 조건식을 작성하세요

```
const arr = [5, 12, 8, 130, 44]; —————→ 10이 배열에 없습니다
```

조건문과 반복문

연습문제

Q5-1. 반복문을 사용하여 1부터 10까지 수 중에서 짝수만 출력하세요

결과: 2 4 6 8 10

Q5-2. 반복문을 사용하여 1부터 10까지 수 중에서 홀수만 큰수부터 출력하세요

결과: 9 7 5 3 1

Q5-3. 반복문을 사용하여 1부터 10까지 수의 합을 출력하세요

결과: 55

Q5-4. 반복문을 사용하여 1부터 20까지 수 중에서 3의배수 또는 5의 배수의 합을 구하세요

결과: 98

Q6. 배열에 있는 모든 숫자를 더해서 합을 구하세요.

```
const arr = [1, 'aa', true, 5, 10];
```

결과

16

조건문과 반복문

연습문제

Q7-1. 5 X 3 크기의 사각형을 별(*)로 출력하세요.
너비가 5이고 높이가 3입니다

```
* * * * *  
* * * * *  
* * * * *
```

Q7-2. 5 X 5 크기의 삼각형을 별(*)로 출력하세요.

```
*  
* *  
* * *  
* * * *  
* * * * *
```

함수 정의

함수는 특정 기능을 수행하는 코드블록이며, 외부에서 입력받은 값을 처리하여 결과를 반환한다.

함수에는 이름이 있을 수도 없을 수도 있다.

함수 사용

함수를 호출하면, 코드가 실행되고 결과를 반환한다.

함수의 이름 파라미터

```
function sum (n1, n2) {  
  let sum = n1 + n2;  
  return sum;  
}
```

반환값

let result = sum(10, 20); → 함수 호출

자바스크립트에서 함수를 정의하는 두 가지 방법이 있다.

첫번째로 함수 선언식(Function Declaration)으로, 함수를 이름과 함께 정의하는 방식이다.

두번째로 함수 표현식(Function Expression)으로, 함수를 변수에 할당하는 방식이다.

자바스크립트에서 함수는 데이터로 취급하기 때문에 변수에 저장할 수 있다.

이때, 함수에 이름이 없을 경우 익명 함수라고 한다.

함수의 이름

↑

```
function hello() {  
  console.log('안녕하세요~');  
}
```

함수 선언식

변수 함수 할당

↑ ↑

```
const hello = function () {  
  console.log('안녕하세요~');  
}
```

함수 표현식

Q8. 두 개의 정수를 입력 받아, 첫번째 수에서 두번째 수를 빼는 함수를 만드세요.

입력: 5, 3

반환: 2

Q9. 두 개의 정수를 입력 받아, 그 사이의 모든 숫자의 합을 반환하는 함수를 만드세요.

입력: 3, 5

반환: 12 (3 + 4 + 5)

Q10. 숫자 배열을 입력 받아, 배열에서 가장 큰 숫자를 반환하는 함수를 만드세요.

입력: [-1, 5, 10, -3]

반환: 10

Q11. 배열을 입력 받아, 모두 같은 자료형으로 이루어져 있는지 확인하는 함수를 만드세요.

입력: ["a", "b", "c"]

반환: true

Q12. 문자열 배열과 특정 문자열을 입력 받아, 특정값이 있는지 확인하고 있으면 true 없으면 false를 반환하는 함수를 만드세요.

입력: ["apple", "banana", "cherry"], "banana"

반환: true

Q13. 정수를 입력 받아, 그 숫자의 구구단을 출력하는 함수를 만드세요

입력: 2

출력:

```
2 * 1 = 2
2 * 2 = 4
2 * 3 = 6
2 * 4 = 8
2 * 5 = 10
2 * 6 = 12
2 * 7 = 14
2 * 8 = 16
2 * 9 = 18
```

Q14. 다양한 값으로 구성된 배열을 입력 받아, 그 중에서 문자열만 골라서 새로운 배열로 반환하는 함수를 만드세요

입력: [1, "apple", true, "banana", 42, "cherry"]

반환: ["apple", "banana", "cherry"]

Q15. 숫자를 입력 받아, 팩토리얼을 계산하여 반환하는 함수를 만드세요

입력: 5

반환: 120 (1*2*3*4*5)

Q16. 학생 객체 배열을 입력 받아, 특정 과목의 평균 점수를 반환하는 함수를 만드세요

입력: students, 'math'

반환: 80

```
const students = [  
  { name: "둘리", math: 90, english: 85 },  
  { name: "또치", math: 80, english: 95 },  
  { name: "도우너", math: 70, english: 75 },  
];
```

Q17. 직원 객체 배열을 입력 받아, 특정 직급의 평균 월급을 반환하는 함수를 만드세요

입력: employees, '사원'

반환: 240

```
const employees = [  
  { name: "짱구", position: "부장", salary: 500 },  
  { name: "철수", position: "차장", salary: 380 },  
  { name: "훈이", position: "사원", salary: 260 },  
  { name: "맹구", position: "차장", salary: 420 },  
  { name: "유리", position: "사원", salary: 220 },  
];
```