

[ITEC] JS Study

2 week

연산자, 조건문, 반복문, 함수 (arrow func, 호이스팅, 스코프)

© yoon sang seok all rights reserved.

Contents

0. 챌린지 코드 리뷰

1. 연산자

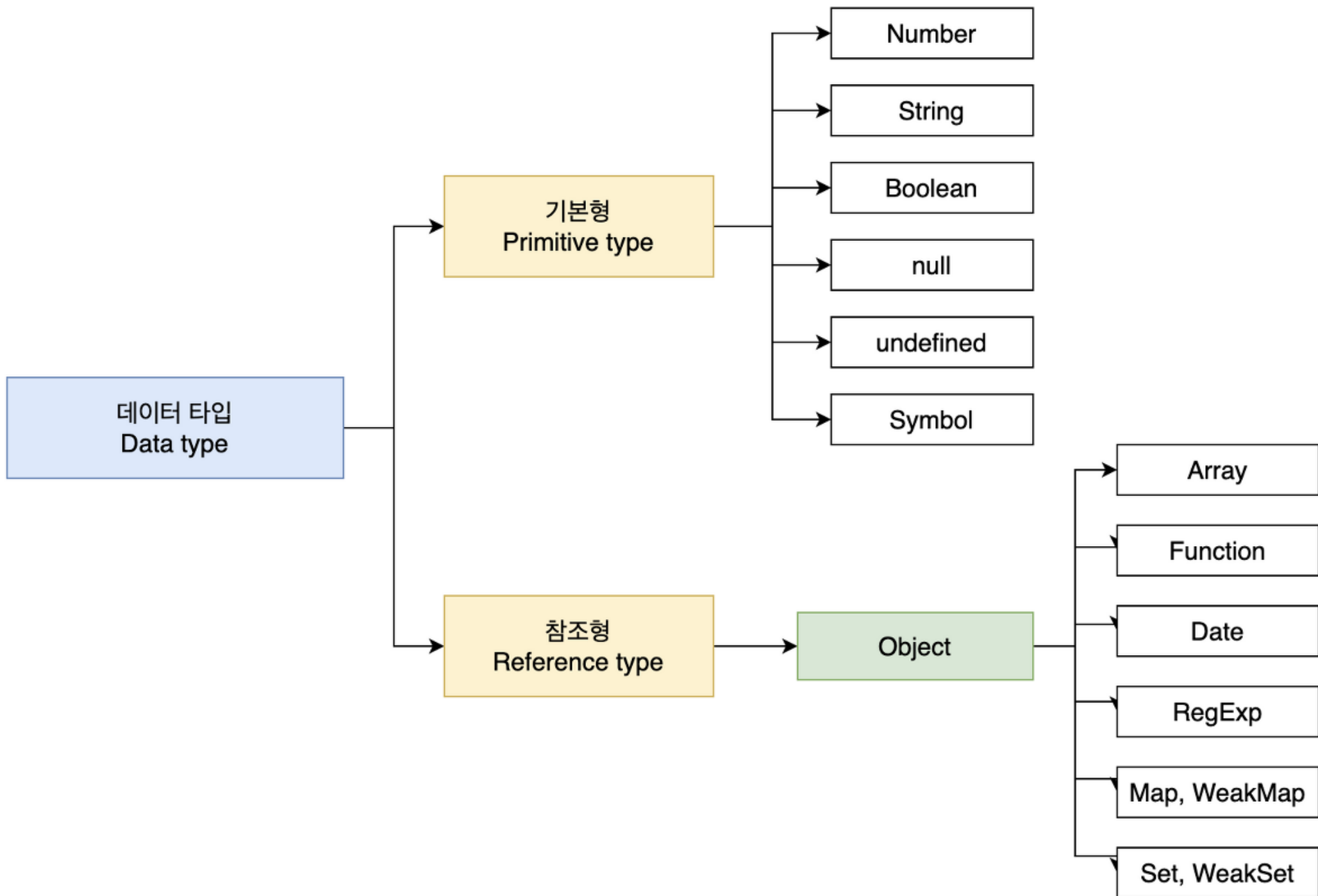
2. 조건문

3. 반복문

4. 함수

0. 챌린지 코드 리뷰

```
function solution(numbers) {  
  let answer = [];  
  for (let i = 0; i < numbers.length; i++) {  
    for (let j = i + 1; j < numbers.length; j++) {  
      answer.push(numbers[i] + numbers[j]);  
    }  
  }  
  answer = new Set(answer);  
  answer = Array.from(answer);  
  answer.sort((a, b) => a - b);  
  return answer;  
}  
  
console.log(solution([1, 2, 3]));  
// [3, 4, 5]
```



1. 연산자

- true, false, null
- &&, ||, ?
- <=, >=, <, >
- ===, !==

```
if ( [] == 0 ) {  
    console.log("????????");  
}
```

```
/*  
[출력 결과]  
?????????  
*/
```

2. 조건문

- `if (...) { } else if (...) { } else { }`
- 삼항연산자 (ternary operator)

```
if (...) { } else if (...) { } else { }
```

```
const age = 24;

if (age < 0) {
  console.log("태어나지도 않았습시다.");
} else if (age <= 19) {
  console.log("나이가 19세보다 작거나 같습니다.");
} else if (age <= 50) {
  console.log("으른");
} else {
  console.log("나이가 50이상??");
}
```

삼항연산자 (ternary operator)

```
let result = condition ? value1 : value2;  
// condition이 true라면 value1, false라면 value2가 반환
```

```
const age = 24;  
  
let accessAllowed = age > 18 ? console.log("1") : console.log("2");
```


3. 반복문 (loop)

- for
- while
- for...of
- for...in

for

```
const myArray = [1, 2, 3, 4];

for (let i = 0; i < myArray.length; i++) {
  console.log(i);
}

/*
[출력 결과]
0 1 2 3
*/
```

while

```
let n = 0;  
  
while (n < 3) {  
  n++;  
}  
  
console.log(n);  
// 3
```

for...of

```
const numbers = [17, 19, 32, 97, 103, 7];

for (const number of numbers) {
  console.log(number);
}
/*
[출력 결과]
17 19 32 97 103 7
*/
```

```
const string = "Hello World!";

for (const char of string) {
  console.log(char);
}
/*
[출력 결과]
H e l l o   W o r l d !
*/
```

for...in

```
const object = { a: 1, b: 2, c: 3 };

for (const property in object) {
  console.log(`${property}: ${object[property]}`);
}

// expected output:
// "a: 1"
// "b: 2"
// "c: 3"
```

별찍기

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

4. 함수

함수 선언식 (function declaration)

```
function printNumber(number) {  
  if (number) {  
    console.log(number);  
  } else {  
    console.log("?");  
  }  
}
```

```
printNumber(7); // 7  
printNumber(null); // ?
```


함수 표현식 (function expressions)

- 함수 선언식은 호이스팅에 영향을 받지만, 함수 표현식은 호이스팅에 영향을 받지 않는다.
- 함수 표현식은 콜백 함수로 사용된다. (다른 함수의 인자로 넘길 수 있음)

```
const printNumber = function (number) {  
  console.log(number);  
};  
  
printNumber(17); // 17
```

```
printNumber1(17); // 17  
printNumber2(18); // Error  
  
function printNumber1(number) {  
  console.log(number);  
}  
  
const printNumber2 = function (number) {  
  console.log(number);  
};
```

챌린지

1. [모던 JS](#) - 2.18까지 정독
2. 시뮬레이션 게임 개발 (주제는 자유) [레퍼런스](#)



© yoon sang seok all rights reserved.