

2024년 상반기 K-디지털 트레이닝

자바스크립트의 변수

[KB] IT's Your Life

1 자바스크립트의 변수

✓ 변수의 종류

○ 프리미티브 타입 primitive type

- 사전적 의미: 초기의, 원시적인 단계의 → 언어 설계 당시에 정한 타입(크기 고정)
- 숫자로 표현되는 데이터 → 크기가 고정되어 있음
 - number, boolean, null, undefined, infinite, NaN
- 변수의 값을 그대로 저장
- = 연산자 사용시 값이 복사됨

○ 참조 타입 reference type

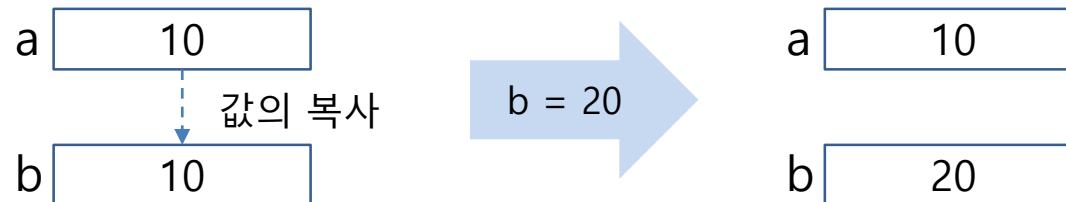
- 값을 저장하는 것이 아니고, 실제 데이터가 있는 메모리 주소(참조값)를 저장
- 프리미티브 타입이 아닌 모든 데이터
 - string, array, object, function ...
- 해당 변수에 대해 읽기/쓰기를 하는 경우 실제 데이터에 대해서 동작
- = 연산자 사용시 참조값이 복사됨 → 값은 공유하게 됨
- 실제 데이터에 대한 참조가 없어지면 쓰레기(garbage) 값이 됨
- 자바스크립트 인터프리터 내부에 garbage collector가 주기적으로 garbage 메모리를 찾아서 해제함(반납함)

✓ 프리미티브 타입

```
let a = 10;
```

```
let b = a;
```

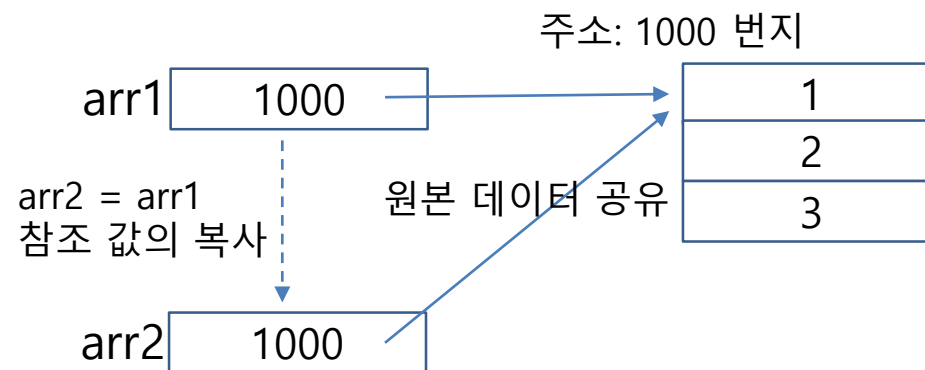
```
b = 20;
```



✓ 참조 타입

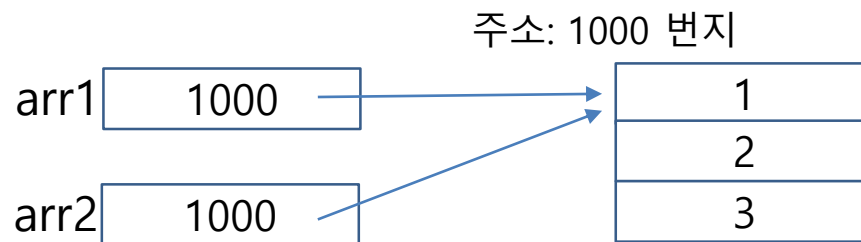
```
let arr1 = [ 1, 2, 3 ];
```

```
let arr2 = arr1;
```



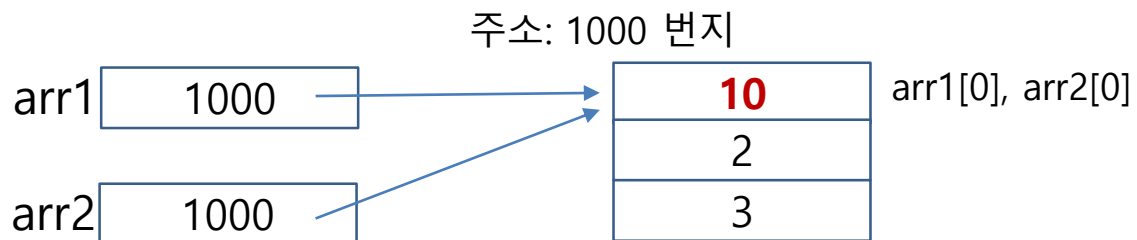
✓ 참조 타입

```
let arr1 = [ 1, 2, 3 ];  
let arr2 = arr1;
```



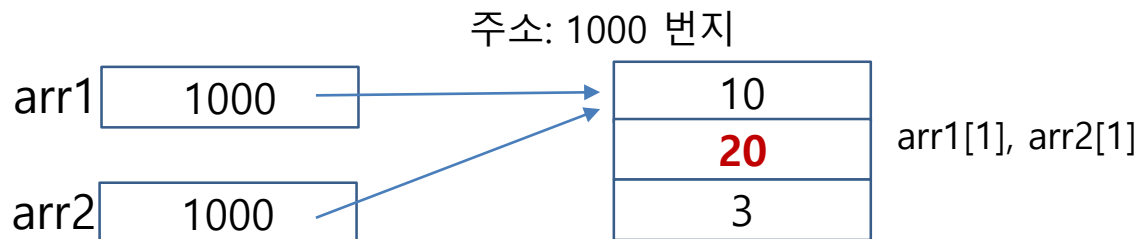
```
arr1[0] = 10;
```

```
console.log(arr1[0], arr2[0]);  
→ 10 10
```



```
arr2[1] = 20;
```

```
console.log(arr1[1], arr2[1]);  
→ 20 20
```

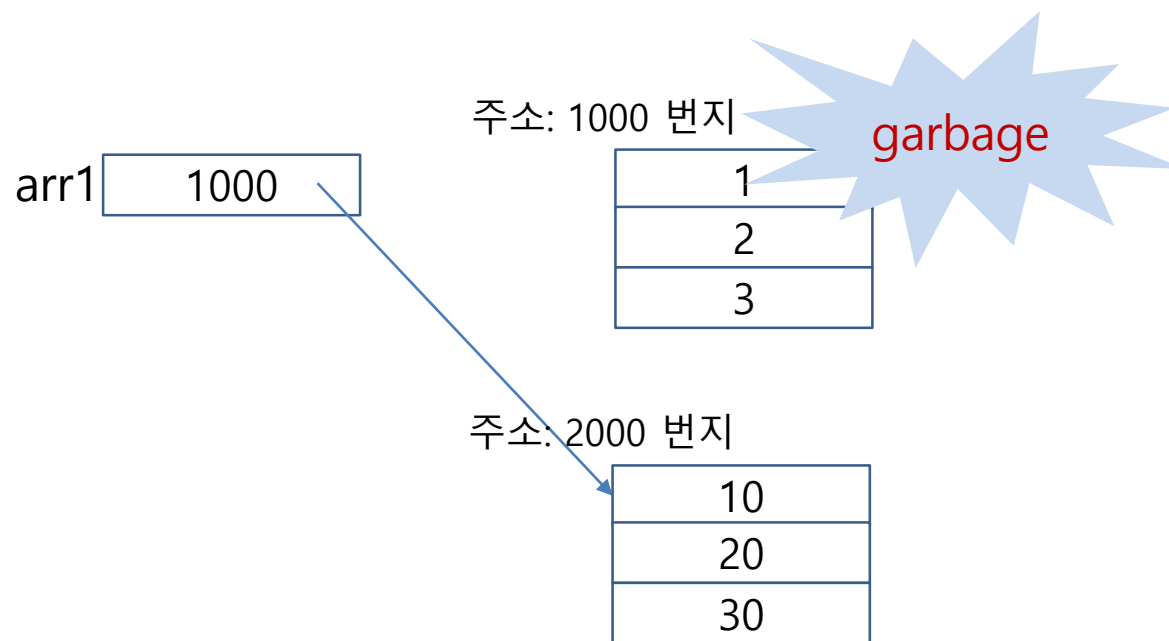
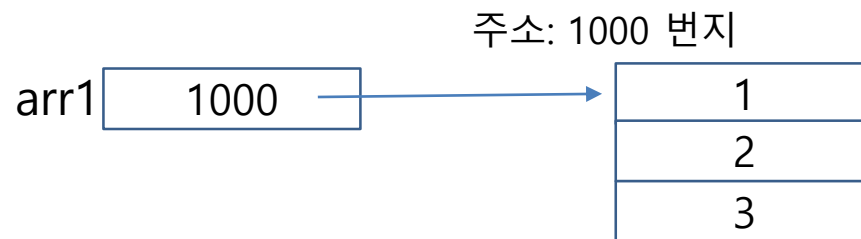


✓ garbage

```
let arr1 = [ 1, 2, 3 ];
```

...

```
arr1 = [10, 20, 30 ];
```



✓ 함수의 매개변수 전달 방식

○ 값에 의한 호출 call by value

- primitive 타입을 매개변수로 사용한 경우
- 인수의 값이 매개변수로 = 연산자 동작 → 값이 복사됨

```
function swap(x, y) {  
  let temp = x;  
  x = y;  
  y = temp;  
  console.log(`swap 함수 안 -> x: ${x}, y:${y}`);  
}
```

```
let a = 100;  
let b = 200;
```

```
console.log(`swap 호출 전 -> a: ${a}, b:${b}`);  
swap(a, b);  
console.log(`swap 호출 후 -> a: ${a}, b:${b}`);
```

✓ 함수의 매개변수 전달 방식

○ 참조에 의한 호출 call by reference

- 참조 타입을 매개변수로 사용한 경우
- 인수의 값이 매개변수로 = 연산자 동작 → 참조값이 복사됨

```
function swap(numbers) {  
  let temp = numbers[0];  
  numbers[0] = numbers[1];  
  number[1] = temp;  
  console.log(`swap 함수 안 -> numbers: ${numbers}`);  
}
```

```
let arr = [100, 200];
```

```
console.log(`swap 호출 전 -> arr : ${arr}`);  
swap(arr);  
console.log(`swap 호출 후 -> arr : ${arr}`);
```

✓ 전역(global) 변수 vs 지역(local) 변수

○ 전역 변수

- 함수 밖에서 전역 레벨에서 선언된 변수
- 모든 함수에서 사용가능

○ 지역 변수

- 함수 내(블록 내)에서 선언된 변수
 - 함수의 매개변수도 지역 변수임
 - 함수가 호출되고 선언문이 실행됐을 때 메모리에 생성
 - 함수가 끝나거나 블록이 끝날 때 메모리에서 제거됨
- 함수(블록) 밖에서는 사용할 수 없음(해당 변수는 존재하지 않음)

○ 이름 충돌

- 전역 변수와 지역 변수의 이름이 같은 경우 지역 변수로 간주

```
let x = 100;    // 전역 변수
let y = 200;    // 전역 변수
```

```
function swap(x, y) { // 지역 변수 x, y
  let temp = x;       // 지역 변수 temp
  x = y;
  y = temp;
  console.log(`swap 함수 안 -> x: ${x}, y:${y}`);
}
```

```
console.log(`swap 호출 전 -> x: ${x}, y:${y}`);
swap(x, y);
console.log(`swap 호출 후 -> a: ${x}, y:${y}`);
```