

# undefined vs null vs undeclared

👤 배정	
📌 상태	
🕒 속성	@2021년 8월 20일 오후 10:17
📌 언어	

변수를 선언, 초기화, 할당을 하다보면 `undefined` 와 `null` 이라는 타입을 많이 보게된다.  
근데 듣도보도 못한 `undeclared` 라는 타입도 있다고 한다.  
undefined와 null은 알겠는데 undeclared는 무엇일까?  
그 전에 자바스크립트에는 7가지 내장 타입이 있다는 것을 알고 있을 것이다.

## 자바스크립트의 7가지 내장타입

| null, undefined, boolean, number, string, object, symbol

이는 `typeof` 연산자로 타입이름을 알 수 있다.

### 변수 vs 값

변수는 타입이 없으나 값은 타입이 있다.  
타입이란? 값의 내재된 특성을 정의함

## undefined



자바스크립트 엔진이 변수를 초기화 할때 사용하는 값

ex)

```
let pokemon;  
console.log(pokemon); // undefined
```

접근 가능한 스코프에 변수가 선언되었으나 아무런 값도 할당되지 않은 상태로, 변수를 초기화 할때 사용하기 때문에 **선언된 변수에 할당할 수 있는 값**이라고 할 수 있다.

## null



변수에 값이 없다는 것을 의도적으로 명시할때 사용하는 값

ex)

```
let pokemon = 'Pikachu';  
  
pokemon = null; // 이전에 참조했던 'Pikachu' 라는 값을 더이상 참조하지 않겠다  
  
console.log(pokemon); // null
```

변수에 어떠한 값도 없다는 의도를 가지고 일부러 넣은 값이다.

그러나 명시적으로 할당되었기 때문에 undefined와는 다르다고 할 수 있다.

## 그렇다면 undeclared는 무엇일까??



접근 가능한 스코프 내에 변수 자체가 선언조차도 되어있지 않은 상태

ex)

```
console.log(digimon); // ReferenceError: digimon is not defined  
  
console.log(typeof digimon); // undefined
```

digimon 이라는 변수 자체가 선언되어 있지도 않은 상태에서 type을 찍어보면 **undefined** 라고 반환되어 브라우저가 오류 처리를 하지 않도록 도와준다.

-> 자바스크립트가 **undefined** 와 **undeclared** 를 모두 섞어버려 에러 메시지 (ReferenceError) 뿐만 아니라 typeof의 반환값도 전부 **undefined** 로 통일시켜버린다.

## 결론

1. `undefined` : 선언된 변수에 할당할 수 있는 값
2. `null` : 변수에 어떠한 값도 없다.
3. `undeclared` : 변수 자체가 선언된 적이 없다.