

15. let, const

▼ TDZ란?

Temporal Dead Zone. 선언과 초기화 사이에 일시적으로 변수 값을 참조할 수 없는 구간.

Javascript 엔진의 경우

```
static InitializationFlag DefaultInitializationFlag(VariableMode mode) {
    DCHECK(IsDeclaredVariableMode(mode));
    return mode == VariableMode::kVar ? kCreatedInitialized
                                       : kNeedsInitialization
}
```

var로 선언한 경우와 let, const로 선언한 경우, 다르게 동작하는 것을 확인할 수 있다.

이 동작 방식으로 인해 hoisting은 발생하나, 초기화가 필요하다는 'Reference Error'가 발생하게 된다.

▼ var, let, const의 스코프 차이

var는 함수 스코프를 가지고 let, const는 블록 범위의 스코프를 가진다.

그래서 var로 선언한 변수의 경우, 다음과 같은 코드의 경우 에러가 발생할 수 있다.

```
var i = 4;

for (var i = 0; i < 5; i++) {
    console.log('test')
}

console.log(i) // 5
```

let, const의 경우 조건문안도 같은 블록 범위로 간주 하기 때문에 위 식에서 let을 사용하면 error를 방지할 수 있다.

현재 JS에서는 var 사용을 지양하고 const, let 사용을 권장한다.