

## local 변수

함수 안에서 사용하는 변수는 일반적으로 함수 내부에서만 유효하여 함수 밖에서는 쓸 수 없습니다.

```
1. def x_is_one():
2.     x = 1
3.
4. x_is_one()
5. print(x)
```

```
Traceback (most recent call last):
  File "untitled.py", line 5, in <module>
    print(x)
NameError : name 'x' is not defined
```

1. (4) 번 줄에서 `x_is_one()` 이라는 함수 호출로 인해 함수가 정의된 (1) 번 줄로 이동합니다.
2. (2) 번 줄에서 `x` 가 1로 지정됩니다.
3. 함수를 다 실행했기 때문에, (4) 번 줄 끝으로 이동합니다.
4. (5) 번 줄에 `print(x)` 가 있지만, 함수 바깥에서 `x` 가 지정된 적은 단 한 번도 없습니다. 따라서 **name 'x' is not defined** 라는 오류가 나옵니다. `x` 가 정의된 적이 없다는 것이죠. 함수 안에서 정의된 변수가 함수 밖에서 쓸 수 없다는 걸 확인할 수 있습니다.

이번에는 오류가 발생하지 않도록 함수 바깥에 `x` 를 지정해줍시다.

```
1. def x_is_one():
2.     x = 1
3.
4. x = 5
5. x_is_one()
6. print(x)
```

5

1. (4) 번 줄에서 `x` 가 5로 지정됩니다. 이는 함수 바깥에서 정의된 `x` 입니다.
2. (5) 번 줄에서 `x_is_one()` 함수가 호출되고, `x` 가 1로 지정됩니다. 하지만 이는 함수 안에서 새로 만들어진 local 변수 `x` 입니다. 두 `x` 는 전혀 관계가 없는거죠.
3. (6) 번 줄에 `print(x)` 는 함수 밖에 있기 때문에, 함수 바깥에 정의된 `x` 를 출력합니다. 따라서 함수 안의 1이 아닌 함수 바깥의 5가 출력됩니다.

## global 변수

그렇다면 함수 안의 `x` 와 함수 밖의 `x` 를 통일하기 위해서는 어떻게 해야 할까요?

첫 번째 변수를 다른 어떤 함수 안에서 사용하고자 한다면, 함수 내에서 변수가 선언된 변수라고 표시해

아래의 코드를 보면 여러 코드 줄에서 사용하고 있지만, 코드 내에서 변수가 global 변수라고 표시를 해줘야 합니다.

```
1. def x_is_one():
2.     global x
3.     x = 1
4.
5. x = 5
6. x_is_one()
7. print(x)
```

1

이전 예시와는 다르게, 콘솔에 1이 출력된 모습을 보실 수 있습니다. 왜 이런 결과가 나온 것일까요?

- (5)번 줄에서 `x`가 5로 지정됩니다. 이는 함수 밖에서 정의된 `x`입니다.
- 그 후 (6)번 줄의 `x_is_one()` 때문에 함수가 정의된 (1)번 줄로 이동합니다.
- (2)번 줄의 `global x`는 함수 안의 `x`와 함수 밖의 `x`를 통일시키라는 명령입니다.
- (3)번 줄에서 `x`는 1로 지정됩니다. (2)번 줄의 `global x` 때문에, 기존에 5의 값을 가졌던 `x`는 새롭게 1의 값을 갖게 됩니다.
- 함수가 끝나면, (6)번 줄 끝으로 이동한 후 (7)번 줄의 명령을 실행합니다. 현재 `x`의 값 1이 출력됩니다.

global 변수에 대해 잘 이해하셨나요? 앞에 있는 `return` 문처럼 헛갈릴 수 있는 개념이니까, 강의에서 다루지 않은 예시를 하나 더 들어드리도록 할게요.

```
1. def next_number():
2.     global x
3.     x = x + 1
4.
5. x = 0
6. next_number()
7. print(x)
```

1

- (5)번 줄에서 `x`가 0으로 지정됩니다. 이는 함수 밖에서 정의된 global 변수 `x`입니다.
- (6)번 줄에서 `next_number()` 때문에 함수가 정의된 (1)번 줄로 이동합니다.
- (2)번 줄의 `global x`는 함수 안의 `x`와 함수 밖의 `x`를 통일시켜줍니다.
- (3)번 줄에서 `x = x + 1`의 명령에 따라, 기존에 0의 값을 가졌던 `x`가, 새롭게 1의 값을 갖게 됩니다.
- 함수가 끝나면, (6)번 줄 끝으로 이동한 후 (7)번 줄의 명령을 실행합니다. 현재 `x`의 값 1이 출력됩니다.

## 주의

여러 함수에서 통일된 global 변수를 쓰면 한번의 실수로 프로그램 전체에 영향을 줄 수 있습니다. global 변수를 꼭 써야하는 상황인지 신중하게 생각해보고 쓰셔야 합니다!



수업을 완료하셨으면 체크해주세요.



수강생 Q&A 보기



(/questions?  
질문하기

assignment\_id=50&sort\_by=popular)  
(/questions/new?

assignment\_id=50&op1=%ED%94%84%EB%A1%9C%EA%B7%B



이전 강의

global변수와 local 변수

(/assignments/49)

다음 강의

함수 심화 퀴즈



(/assignments/136)