

# Ask Django

모듈/팩키지

# Agenda

- import
- modules (모듈)
- packages (패키지)

# import

- 다른 파이썬 소스파일 내 함수/클래스등을 현재의 공간으로 가져오기
- import 시점에서 해당 코드가 실행됩니다.

```
module.py
pkg1
├── __init__.py
└── pkg2
    └── __init__.py
```

```
import module                # module.py
module.some_fn()             # module.py 내 some_fn
```

```
from module import some_fn   # module.py 내 some_fn
some_fn()
```

```
import pkg1                  # pkg1/__init__.py
import pkg1.pkg2             # pkg1/pkg2/__init__.py
pkg1.pkg2.some_fn()          # pkg1/pkg2/__init__.py 내 some_fn
```

```
from pkg1 import pkg2        # pkg1/pkg2/__init__.py
pkg2.some_fn()
```

```
import mymodule1
mymodule1.mysum(1, 2)
```

```
# mymodule1.py 내 사항을 가져옴
# mymodule1.py 내 mysum 함수
```

```
import pkg1.pkg2
pkg1.pkg2.mysum(1, 2)
```

```
# pkg1/pkg2/__init__.py 사항을 가져옴
# pkg1/pkg2/__init__.py 내 mysum 함수
```

```
import pkg1.pkg2.모듈
pkg1.pkg2.모듈.mysum(1, 2)
```

```
# pkg1/pkg2/모듈.py 내 사항을 가져옴
# pkg1/pkg2/모듈.py 내 mysum 함수
```

# modules (모듈)

- 다수의 함수/클래스들을 정의해둔 파이썬 소스코드 **파일**

```
# mymodule.py
def mysum(x, y):
    return x + y
```

```
mymultiply = lambda x, y: x * y
```

## 실행

```
>>> import mymodule                # 모듈 단위로 가져오기
>>> mymodule.mysum(1, 2)           # 필히 모듈을 통해 접근
3
>>> mymodule.mymultiply(1, 2)
2

>>> from mymodule import mysum, mymultiply # 특정 함수만 가져오기
>>> mysum(1, 2)
3
>>> mymultiply(1, 2)
2
```

# packages (팩키지)

- 파이썬 소스코드가 들어있는 **디렉토리**
- 해당 디렉토리에는 필히 **\_\_init\_\_** 파일이 있어야, 파이썬 팩키지로서 인식합니다. (파이썬 3.3 이상에서는 없어도 인식하긴 합니다.)
- 팩키지를 import할때에는 **\_\_init\_\_.py** 가 import 대상이 됩니다.

```
mylib
|- __init__.py
    |- mysum4 임포트된 함수
|- math.py
    |- mysum4 함수
```

```
# mylib/__init__.py
from .math import mysum4 # 현재 __init__.py 파일이 있는 디렉토리 내 math의 mysum4

# mylib/math.py
def mysum4(a, b, c, d):
    return a + b + c + d
```

## 셸에서 실행

```
# 가져와서 쓰기
>>> from mylib import mysum4 # mylib/__init__.py 내 임포트된 mysum4 함수
>>> mysum(1, 2, 3, 4)
10

>>> from mylib.math import mysum4 # mylib/math.py 내 오리지널 mysum4 함수
>>> mysum(1, 2, 3, 4)
10
```

# import 해서, 이름을 변경해서 쓰기

mylib.py

|--- mysum4 함수

mylib2.py

|--- mysum4 함수

```
>>> from mylib import mysum4
```

```
>>> from mylib2 import mysum4
```

```
>>> mysum4(1, 2, 3, 4)
```

```
10
```

```
>>> mysum4(1, 2, 3, 4)
```

```
10
```

# 앞서 import된 mysum4를 덮어씁니다.

# 둘 다 따로 쓰고 싶은 데, 늦게 import된 mylib2.mysum4만 사용됩니다.



import시에 **as**를 통해 원하는 이름으로 변경

```
>>> from mylib import mysum4 as mylib_mysum4
>>> from mylib2 import mysum4 as mylib2_mysum4
>>> mylib_mysum4(1, 2, 3, 4)
10
>>> mylib2_mysum4(1, 2, 3, 4)
10
```

# Relative Import

팩키지 내에서 다른 모듈/팩키지 가져오기

main.py

pkg1

- \_\_init\_\_.py
  - mysum 임포트된 함수

- math.py
  - mysum 함수

# pkg1/math.py

```
def mysum(x, y):  
    return x + y
```

# pkg1/\_\_init\_\_.py

```
from .math import mysum # mysum함수를 현재 이름공간(name)으로 가져옴.
```

# main.py : 아래 2가지 mysum을 모두 사용 가능

```
from pkg1 import mysum  
from pkg1.math import mysum
```

# import 경로

- import 를 수행할 때, **sys.path** 경로에서 모듈/패키지 탐색
  - 환경변수 **PATH** 개념과 유사
- sys.path 는 **list** 이기 때문에, 자유롭게 추가/수정/삭제 가능
  - 수정된 내용은 현재 프로세스에서만 유효
  - 관리성이 나빠지기 때문에 권장하는 방법은 아님
- 현재 디렉토리와 sys.path 경로에서 지정 모듈/패키지를 찾지 못했을 경우, **ImportError** 발생

# 파이썬 소스코드 내 `__file__`

- 해당 파이썬 소스코드 파일 경로
- `pkg1/helloworld.py`일 경우
  - `"/Users/askdjango/pkg1/helloworld.py"`
- 참고 : 장고프로젝트/settings.py 내 `BASE_DIR`

```
from os.path import abspath, dirname
BASE_DIR = dirname(dirname(abspath(__file__))) # 현 장고 프로젝트 ROOT 절대경로 계산
```

# 파이썬 소스코드 내 **\_\_name\_\_**

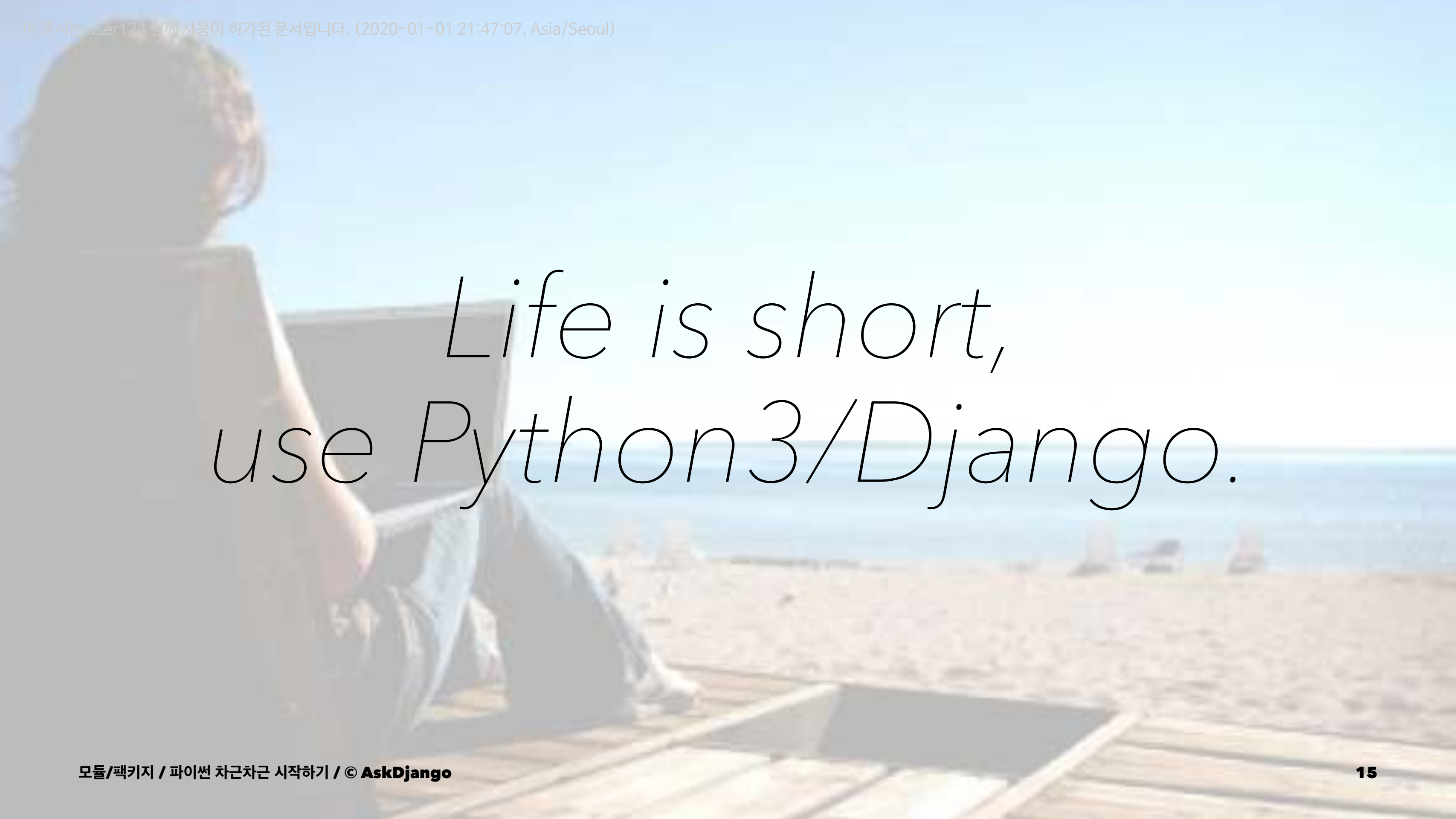
- 해당 파이썬 소스코드 파일명
  - pkg1/helloworld.py일 경우 : "helloworld"
- 비교
  - 최초 진입 소스코드일 경우 : "\_\_main\_\_"으로 변경되어 실행
  - import된 소스코드 : 본래 \_\_name\_\_이 유지된 채로 실행

- 이를 통해, import시에는 실행되지 않고, **최초 진입 시에만** 실행될 코드를 지정 가능

```
def main():  
    print('본 스크립트가 최초 진입 소스코드 경우에만 실행됩니다.')
```

```
if __name__ == '__main__':  
    main()
```

- 위 스크립트에서 `if __name__ == '__main__'` 블록은 다른 소스코드에 의해 import 될때에는 실행되지 않습니다.

A person with long hair is sitting on a wooden pier, looking out at the ocean. The scene is bright and sunny, with a clear blue sky and a calm sea. The person is wearing a light-colored top and jeans. The pier is made of wooden planks, and there are some small structures in the distance.

*Life is short,  
use Python3/Django.*