

Package

Created by 이윤준 (yoonyoon.lee@gmail.com)

May, 2019

Package?

패키지(Packages)는 도트(.)를 이용하여 파이썬 모듈을 계층적(디렉터리 구조)으로 관리할 수 있습니다. 예를 들어 모듈명이 A.B인 경우 A는 패키지명이고 B는 A 패키지의 B 모듈입니다.

```
game/  
  __init__.py  
  sound/  
    __init__.py  
    echo.py  
    wav.py  
  graphic/  
    __init__.py  
    screen.py  
    render.py  
  play/
```

Create a Package

1. `C:\Users\YoonJoon\Lectures\` 라는 디렉터리 밑에 `game` 및 기타 서브 디렉터리들을 생성하고 `.py` 파일들을 다음과 같이 만듭니다.

```
C:/Users/YoonJoon/Lectures/game/__init__.py
C:/Users/YoonJoon/Lectures/game/sound/__init__.py
C:/Users/YoonJoon/Lectures/game/sound/echo.py
C:/Users/YoonJoon/Lectures/game/graphic/__init__.py
C:/Users/YoonJoon/Lectures/game/graphic/render.py
```

2. 각 디렉터리에 `__init__.py` 파일을 만들어 놓기만 하고 내용은 일단 비워 둡니다.

3. [echo.py](#) 파일은 다음과 같이 만듭니다.

```
# echo.py
def echo_test():
    print ("echo")
```

4. [render.py](#) 파일은 다음과 같이 만듭니다.

```
# render.py
def render_test():
    print ("render")
```

5. 우리가 만든 game 패키지를 참조할 수 있도록 다음과 같이 명령 프롬프트 창에서 set 명령을 이용하여 PYTHONPATH 환경 변수에 C:\Users\YoonJoon\Lectures\ 디렉터리를 추가한다. 그리고 파이썬 인터프리터(Interactive shell)를 실행합니다.

```
> set PYTHONPATH=C:/doit
> python
Type "help", "copyright", "credits" or "license" for more
```

Run functions defined in Package

- echo 모듈을 import하여 실행하는 방법

```
>>> import game.sound.echo  
>>> game.sound.echo.echo_test()
```

- echo 모듈이 있는 디렉터리까지 from ... import하여 실행하는 방법

```
>>> from game.sound import echo  
>>> echo.echo_test()
```

- echo 모듈의 echo_test 함수를 직접 import하여 실행하는 방법

```
>>> from game.sound.echo import echo_test
>>> echo_test()
```

- 불가능

```
>>> import game
>>> game.sound.echo.echo_test()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: 'module' object has no attribute 'sound'
```

```
>>> import game.sound.echo.echo_test
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ImportError: No module named echo_test
```

Use of __init__.py

__init__.py 파일은 해당 디렉터리가 패키지의 일부임을 알려주는 역할을 합니다.

“ python3.3 버전부터는 __init__.py 파일 없이도 패키지로 인식이 된다(PEP 420). 하지만 하위 버전 호환을 위해 __init__.py 파일을 생성하는 것이 안전한 방법이다. ”

```
>>> from game.sound import *
>>> echo.echo_test()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'echo' is not defined
```


특정 디렉터리의 모듈을 *를 이용하여 import할 때에는 다음과 같이 해당 디렉터리의 __init__.py 파일에 __all__이라는 변수를 설정하고 import할 수 있는 모듈을 정의해 주어야 한다.

```
# C:/Users/YoonJoon/Lectures/game/sound/__init__.py
__all__ = ['echo']
```

__all__이 의미하는 것은 sound 디렉터리에서 * 기호를 이용하여 import할 경우 이곳에 정의된 echo 모듈만 import된다는 의미

“ from game.sound.echo import * 는 __all__과 상관없이 무조건 import된다. 이렇게 __all__과 상관없이 무조건 import되는 경우는 from a.b.c import * 에서 from의 마지막 항목인 c가 모듈인 경우

”

Relative Package

만약 graphic 디렉터리의 [render.py](#) 모듈이 sound 디렉터리의 [echo.py](#) 모듈을 사용하고 싶다면 어떻게 해야 할까요?

```
# render.py
from game.sound.echo import echo_test
def render_test():
    print ("render")
    echo_test()
```

```
# render.py
from ..sound.echo import echo_test

def render_test():
    print ("render")
    echo_test()
```