

모듈(module) 만들어 사용하기

이번 장에서는 파이썬에서 내가 만든 모듈의 사용 방법에 대하여 공부합니다.



1. 모듈과 import
2. 하나의 프로그램을 여러 개의 파일로 만들기
3. Import할 모듈이 있는 위치
4. 메인 모듈과 하위 모듈
5. 패키지 만들기





1. 모듈과 import

□ 모듈 (module)

- 모듈은 파이썬 코드를 작성해 놓은 스크립트 파일
- 모듈 안에 함수, 변수, 클래스 등이 정의되어 있음
- 파이썬 설치 디렉토리를 보면, 확장자가 py인 파일이 수십개 있으며, 여기에 파이썬에서 제공하는 표준 모듈들이 있음
- 표준 모듈의 집합을 '표준 라이브러리(standard library)' 라고 부름
(<https://docs.python.org/3/library/index.html>)

□ import 명령

- 모듈을 가져와 사용하려면 import 명령을 사용함
- 현재 파일에 import된 모듈의 코드가 읽혀짐



2. 하나의 프로그램을 여러 개의 파일로 만들기

calculator.py

```
def plus(a, b):  
    return a+b
```

```
def minus(a, b):  
    return a-b
```

```
def multiply(a, b):  
    return a*b
```

```
def divide(a, b):  
    return a/b
```



2. 하나의 프로그램을 여러 개의 파일로 만들기

calc_tester.py

```
import calculator

print(calculator.plus(10, 5))
print(calculator.minus(10, 5))
print(calculator.multiply(10, 5))
print(calculator.divide(10, 5))
```





2. 하나의 프로그램을 여러 개의 파일로 만들기

calc_tester2.py

```
from calculator import *  
  
print(plus(10, 5))  
print(minus(10, 5))  
print(multiply(10, 5))  
print(divide(10, 5))
```



2. 하나의 프로그램을 여러 개의 파일로 만들기

calc_tester3.py

```
from calculator import plus  
from calculator import minus
```

```
print(plus(10, 5))
```

```
print(minus(10, 5))
```

```
#print(multiply(10, 5))
```

```
#print(divide(10, 5))
```

#을 제거하고 실행해보시오



3. Import할 모듈이 있는 위치

□ Import할 모듈 파일을 찾는 위치(폴더)가 어디인지 알아보려면

- sys.path에 정의되어 있는 폴더

```
>>> import sys  
>>> print(sys.path)
```

참고) 파이썬에 설치된 표준 모듈 알아보기

```
>>> import sys  
>>> print(sys.builtin_module_names)
```




4. 메인 모듈과 하위 모듈

- 메인 모듈 : 최상위 수준에서 실행되는 모듈, 즉 제일 먼저 실행되는 모듈
- 하위 모듈 : 메인 모듈에서 import 하여 사용되는 모듈
- 내장 전역 변수 `__name__`
 - 모듈이 최상위 수준에서 실행될 때 `__name__` 의 값이 “`__main__`”으로 지정됨
 - 다른 모듈에서 import하여 실행될 때는 `__name__` 의 값이 자신 이름으로 지정됨

```
print(__name__)
```



4. 메인 모듈과 하위 모듈

□ 예 1

calculator.py

```
print("이것은 calculator 모듈입니다")  
print(__name__)
```

calc_tester.py

```
import calculator  
  
print("이것은 calc_tester 모듈입니다")  
print(__name__)
```

- 1) calc_tester를 실행해 봅니다.
- 2) calculator를 실행해 봅니다.



4. 메인 모듈과 하위 모듈

□ 예 2

calculator.py

```
if __name__ == '__main__':  
    print("이것은 calculator 모듈입니다")  
    print(__name__)
```

calc_tester.py

```
import calculator  
  
print("이것은 calc_tester 모듈입니다")  
print(__name__)
```

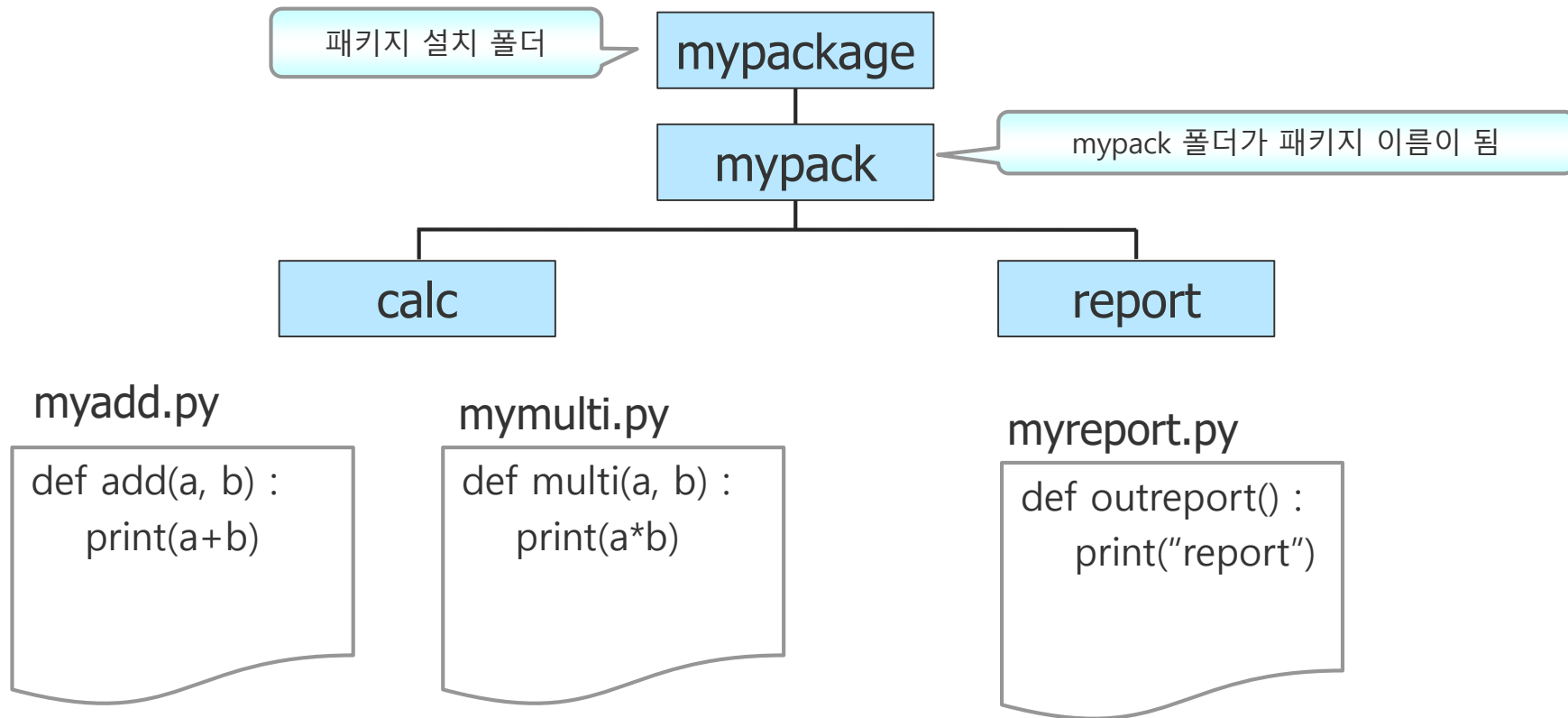
- 1) calc_tester를 실행해 봅니다.
- 2) calculator를 실행해 봅니다.



5. 패키지 만들기

□ 예

- C:\mypackage 폴더를 만들고 다음과 같이 폴더와 파일을 만들어보자





5. 패키지 만들기

□ 패키지를 사용하는 프로그램을 만들어보자

usepackage.py

```
import sys
sys.path.append("c:\\mypackage")

import mypack.calc.myadd
mypack.calc.myadd.add(1, 2)

import mypack.report.myreport
mypack.report.myreport.outreport()
```

`mypack.calc.myadd.add(1, 2)`

패키지 폴더 모듈 함수



5. 패키지 만들기

- 앞의 프로그램(usepackage.py)에서 import의 다른 방법

usepackage.py

```
import sys
sys.path.append( "c:\Wmypackage" )
```

```
from mypack.calc import myadd
myadd.add( 1, 2)
```

```
import sys
sys.path.append( "c:\Wmypackage" )
```

```
from mypack.calc.myadd import add
add( 1, 2)
```