

모듈, 패키지, 라이브러리

Python document : <https://docs.python.org/3/>

- 비슷한 종류의 일을 하는 함수들 끼리 따로 묶어서 '모듈' 이라는 묶음으로 제공
- 내장 함수 (Built-in Fuction) : 바로 사용 가능
- Built-in 자료형 관련 메소드 : 각각 자료형에 해당되는 메서드
- 모듈 : 외부함수/라이브러리 사용 전에 관련 모듈을 먼저 import 해야 함.

-
- 모듈(module) : 특정기능을 하나의 *.py 파일 안에 상수, 변수, 함수, 클래스 을 담고 있다.
 - 패키지(package) : 여러 모듈을 폴더로 묶어서 계층적으로 관리. 모듈을 주제별로 분리할때 주로 사용
 - 라이브러리(Library): 여러 모듈과 패키지를 묶음
 - 파이썬 표준 라이브러리 (Python Standard Library, PSL) : 파이썬에 기본으로 설치된 모듈과 패키지, 내장 함수 (예: time, sys, os, math, random, urllib 등)
 - python.org가 아닌 외부 3rd party에서 개발한 모듈과 패키지를 묶어서 외부 라이브러리(Third Party Library)라고 합니다. (예: Numpy, Scipy, requests, scrapy, webbrowser)

모듈 (module)

- Python 에서의 모듈은 함수나 변수, 클래스 등을 모아 놓은 파일
- 기능 단위로 만든 python 파일로서, 라이브러리 역할을 함
- 사용자가 만들어서 추가할 수도 있고, 외부에서 받은 파일도 사용할 수 있다.
- 기본 Python 의 기능을 확장하기 위하여 사용



- 파이썬의 세 가지 모듈 (import)

1) 표준 모듈 : 파이썬에서 제공하는 모듈

- random, datetime, math, string, turtle, tkinter, file 등 200여개 모듈 존재

- 파일과 디렉터리 접근 : sys, os
- 데이터 파일 저장 : pickle
- 수학 및 랜덤 : math, random
- 인터넷 액세스 : webbrowser, urllib
- 날짜와 시간 : time, datetime

2) 서드 파티(3rd Party) 모듈 : 파이썬이 아닌 외부 회사나 단체에서 제공하는 모듈

- 파이썬 표준 모듈이 모든 기능을 제공 않음
- 서드 파티 모듈 덕분에 파이썬에서도 고급 프로그래밍 가능
 - 파이썬으로 GUI 프로그래밍을 하려면 TkInter, wxPython1, PyQt2
 - 엑셀 파일 처리 : openpyxl, xlrd, xlwt
 - 게임 개발 기능 pyGame, 윈도우창을 제공 PyGTK, 데이터베이스 기능 SQLAlchemy
 - 데이터처리: Numpy, Pandas, Matplotlib, Tensorflow



3) 사용자 정의 모듈 : 직접 만들어서 사용하는 모듈

- import 모듈 가져오기 (import ~ / import ~ as ~ / from ~ import ~)

- ```
----- 모듈 -----
- import 모듈 / import 모듈1, 모듈2
- import 모듈 as 이름

----- (특정변수, 함수, 클래스를 import 할때) -----
- from 모듈 import 변수, 함수, 클래스
- from 모듈 import *
- from 모듈 import 변수(함수, 클래스) as 이름
- from 모듈 import 변수 as 이름1, 함수 as 이름2, 클래스 as 이름3
```

### - 표준모듈 사용하기

```
In []: # Math 모듈의 정수 관련 함수
import math

print('floor(7.7) : %d' % math.floor(7.7))
print('ceil(10.1) : %d' % math.ceil(10.1))
print('round(8.6) : %d' % round(8.6))

print('5의 팩토리얼 : %d' % math.factorial(5))
```

```
In []: # 삼각/거듭제곱/제곱근/로그 함수
import math as m

print('sin(pi/2) : %.2f' % m.sin(m.pi/2))
print('cos(pi) : %.2f' % m.cos(m.pi))
```

```
print('tan(pi*2) : %.2f' % m.tan(m.pi*2))

print('2의 4승 : %d' % m.pow(2,4))
print('49의 제곱근 : %d' % m.sqrt(49))
print('log10(100) : %.2f' % m.log10(100))
```

```
In []: # date 클래스: 날짜를 표현
time 클래스: 시간을 표현
datetime 클래스: 날짜와 시간을 표현

import datetime
now = datetime.datetime.now() # datetime 모듈에는 클래스가 정의 --> 객체생성후 사용
print(now)
print(type(now))
```

```
In []: from datetime import datetime

now = datetime.now()
now.year, now.month, now.day, now.hour
```

```
In []: import datetime

set_day = datetime.date(2019,3,3)

print(set_day)
print(type(set_day))
```

```
In []: # random() 함수

import random

for i in range(3) :
 print(random.random())
```

```
In []: import random

dice1 = random.randint(1,6) # 임의의 정수가 생성됨
dice2 = random.randint(1,6) # 임의의 정수가 생성됨

print('주사위 두 개의 숫자: {0}, {1}'.format(dice1, dice2))
```

```
In []: import random

random.sample([1, 2, 3, 4, 5], 2) # 모집단에서 두 개의 인자 선택
```

----- END -----