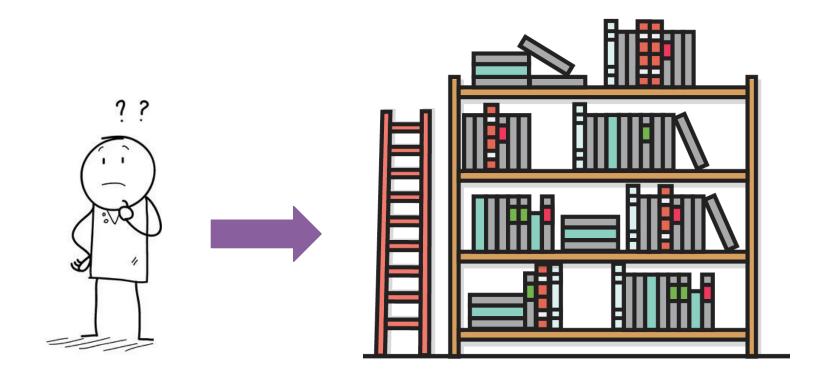


# Library

Library > Package > module







## Library, Module, Package

• **Class** : 객체(Object)를 생성하는 틀(설계도)

• Function : : 특정 작업을 수행하는 코드 블록

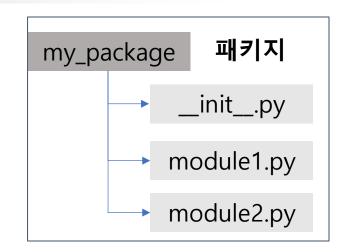
 Module: method(function) + variable + class 일반적으로 파일 1개로 구성 ex. ) moduel1.py

• Package : 특정 기능과 관련된 <u>여러 모듈</u>이 모인 폴더(디렉토리)

• Library : 특정 기능을 수행하기 위해 미리 작성된 코드 모음 (<u>패키지 모음</u>)

### Library > Package > module

1 import numpy as np 2 import pandas as pd



## Library, Module, Package

Function, class:기능구현

```
def greet(name):
   return f"Hello, {name}!"
# 함수 호출
print(greet("Alice")) # 출력: Hello, Alice!
class Person:
   def init (self, name, age):
       self.name = name
       self.age = age
   def greet(self):
       return f"Hello, my name is {self.name} and I am {self.age} years old."
# 클래스 인스턴스 생성
person1 = Person("Alice", 30)
# 메서드 호출
print(person1.greet()) # 출력: Hello, my name is Alice and I am 30 years old.
```

## **Standard Library**

Python 과 함께 배포되는 표준 라이브러리

## Standard Library

- 파이썬과 함께 배포되는 표준 라이브러리
- Python 공식 문서 설명: https://docs.python.org/ko/3/library/index.html

```
· TWO BETTER AND
               · 121年 国際報告機 第210 000 00
               * and of not
             • 숙제 함 — int, finat, complet
• 이라마다의
• 서면스 항 — List, tuple, namps
             는 에스트 NEC NE - NE PROPERTY NEW PARTY NEW PART

    STACE (IPSA 18)
    Type Association Types — Generic Atlas, Union
    TITL 45/19

               * 50 HERVE

    Latti Mitti
    Eacuption control

               · Emercina from trulificia excuptions
                  * WS WE 7.5
             * difflib - 현단 제산을 제한 도반이

* textwrap - 제소를 제되고 세우기

* unicodedate - 프라크로 안타하였다.
               * ATTIMODERS - THE BARN AND
                  * readtine — SAU readine (*1875)
* ricompleter — SAU readine (*1875)
               * ATTEMED - 1885 WINDOWS GLOSSIE HIS SENSE MAN
                    · codeca — 394 科丹在原刊的 特別在 實現在
               * detection — 718 graps sin to
               * rollertions, abr - 5000 his; $50 0000 mean

    houng 一当 并言2758
    hisset 一個首の目標音2758
    array 一直無可数 文本報告

    **** *** 보존 *** *** 보존 *

    * enum - 발개하지만
    * enum - 발개하지만
    * qraphith - 그래요와 무시한 구조하지만하는 개능
    * QRB 수의 말통
    * enumers - 오직 추신 해비스 함께스

                  * court - Wood an on to
             * decinal — UP JM APR 1 PR APR US

* fractions — RNA

* random — SM NA BM
                  * statuation - ON BUILDO
     * operator — BORMS DR 95%

    * aperator — 항우리에의 요한 인간이
    * 변설과 되면수의 역세소
    * path\sb — 대체 시한 위원 시스템 전문
```

· Fallegrout - Wittel America 68 correct NO \* stat - statt) ggrasum; tempfile — 등이 바랍기 다하다는 등점
 glob — 하나스 스타일 강조의 하는 환경
 frantch — 하나스 타일에 하는 일이 Linecache — 역소의 출연 (3만 하하의 역년소)
 shari 1. - 2/수문 6월 (2만)
 180만 되유성)
 pickis — 위성한 격세 개절함 · copyrep — pickle 지원 함수 등록 · sfelse — 파이턴 에서 다음성 · marshal — 대부 베이턴 에서 처럼의 · sqlits1 - SQLW OFFEMOLS DD-AR 2.0 YERRID. 1000 TA TAN
 110 - gpip R RATE TA • grip — grip 유럽 지원 • 8:2 — brip2 합유 지원 • Lana — LZNA 중요하용된 사용한 법유 \* viprile - Zipostina sei · harfs/e — ar 아이지나 매일 당기와 쓰기 可算 發利
 6 mx — CSV 可复 計7的 处力 \* configurate - 구현 매일 구분 분석기

\* mare - road 매일 시선

\* plintlp - 매월 .plint 대한 영상과 구분 본석

• 명조와 사이스 \* tanhist — 보면 해서한 매시지 모두 \* tank — 매시에 면접을 하면 가 해성 • becrets — 바면 면접을 하면 안전한 나수 성성 · WITH HIS MAN WHILE \* 21 - 721 ER MM TRIMELS time — 사는 액세스를 만든
 arsparae — 편안할 옵션, 인지한 무속 명안을 되면 되시. gernpt — 공연 중 옵션은 C 스타일 구인 분석기
 topgsap — 유대변 문장 시설 \* topging.costig - ## 26 \* ELFERS. INSTEAD — ELFAN ELFER 함께 하는 함께 위해 \* ELFERS. BEELL — ASCH 문제용 유행보다 \* ELFAN ELFAN ELFAN 함께 하는 ELFAN 함께 하는 ELFAN ELFAN 함께 ELFAN 함께 ELFAN 함께 ELFAN • 보시 선명 • 보기 선명 • 보기 선명 • 보기 선명 • 보기 선명 \* threating — AME TH WE NO · multigrocessing - REMA THIS STREET multiprocessing, charact\_memory — BEMA 전 역한 역사소를 하는 중앙 제도에를 제공합니다.
 concurrent 해당기 . concurrent futures - 800 for own it subprocess — 서남 프루셔스 근건
 sched — 역원을 스러워크 \* mess - errors o mac \* asyncio — 明後7110 \* socket — 阿存在 短期期分享明明日点 salt — 55 PRB TLSSSL (B)
 salact — 30 PE R)
 salactors — 302 NO NSB SEAT OF SHIP WELL WAVE HELD IN VIOLENCE

· minutypes - NOW MAKE ORNAME . torold - Sparts Sparts Sparts Sparts Sparts with the con-. hinhes - named Will William Clark · quopri - MAVE questo preside dictel (125) of 1125 구축하면 하고입 사이 도구
 호크리 — 아이버텍스의 바고입 문에 직원
 호크리 \_ 아이버텍스의 바고입 문에 직원
 호크리 \_ 라이버텍스의 바고입 문에 직원 . Brist, artition - Hither ply Writin Bill \* AND ANY MAN.

\* AND ANY MAN. \* mil dom.mixidam — 최소 DOM 구함 \* mil dam.palldom — 편호 DOM 표리 구축 지원 \* mil max — SAX2 구원 등석기 제원 \* mil max — SAX2 구원 등석기 제원 · ref. car. variet to - SAK REPORT · unl. parmers.expat - ExpatS NAT WE MA. TH WH · OTERS HARM THE VIEW \* webbrower — Coverent visit-brower co

\* weptref — WSGI #19 Fill #10 - 78 \* write-united th · artish-request — LRL은 경기 4차 학원 기능한 라이브라이 · artish-request — LRL은 경기 4차 학원 기능한 라이브라이 · artish-parse — 보면요 구선 교수도 구선 문제 A WALL OF MATERIAL WARRANCE THE STREET, MAY WHEN urlighted order with company and profile
 urlighted order or command the Built
 hap - MITP and
 haptallant - MITP aum & Builting
 raplab - FIP menum macage \* poplis - POPS NAME BROOMS \* immplib - SAAPA NAME BROOMS \* amplib - SMIP NAME BROOMS \* upid - RFC 4122 III FIF UUD IRH \* accessorer - WENG AME HE LANGE: \* Attp.server - HITP AME · http://pokiss - HTTP the per \* Attp. cookisjar — HTTP @HOTPEB SIN 441 MAI \* salipe — MALSPE HAD BRITISE CH · antercalizant - XML-RPC menters was mitroc.server — 718 XXX.-RDC Me
 tpaddress — PykiPys. 251 Hopens
 traction ANA. · wave - www rest std Ad-· Incate - MRE ARC. 프로그램 조선인되고
 turtio — 다른 그센터
 turtio — 다른 그센터
 tul — 은 지장 영영 인터보석의 자원 · shing - zitro oig gre \* Tell 사용한 그램의 사용적 전략되었는 \* tkinter — TelTk 대자를 인턴되었는 · tkinter\_colorchopser - 450 MM can wat \* thinter.fast - Thomas ## fire \* tkinter\_messagebox -- Tkinter MAN ABBE \* tkinter.szrolledtext - 土田書 Tim Taul ATW \* tkinter.dod - 五田王 田田 正都 本語 \* tkinter.ttk - Tk 明年 和明 . tkinter, tis - Estamica widges for Th • 제발 도구 • 1 5일대학 — 등 전혀 지원 • 9일대다 — 설명이 생산의 존리한 도움별 시스템 • 레이턴 개별 교도 • 레이턴 개별 교도의 효과

\* unittent.mack — 모기 때로 바다하다! \* unittent.mack — 시하하기 \* 203 - 바다전 2에서 마다만 10의 지동 2도 변환 \* test — স্থার ভার আন্ত ক্ষাম • test.support — স্থার আনত নমাত্র সমুদ্রার • test.support.sucket\_helper — একু আনত্তর সমুদ্রার . test, support, script below - word officials assess \* 1sst.support.bytecode\_iw\ger - #WE HHE IS SS NAME HH IN ST \* test\_support\_threading\_telper — Utilias for threading leads test, support, on\_belper — Utiling force tests
 test, support, inport belper — Utiling for important · CHEST SERVE · HALDIME B \* bob — First Singlist \* Faulthandler — Note Binston 可是 \* pob — 하하면 다하다 대이전 프로마일러
 timest — 하는 그리고 프리크 설명 시간 유명
 trace — 마이안 본장 설명 추억 企工机构的 用作品 世 報告 annumph — pap 설립 모드기 함께
 annumph — pap 설립 모드기면 무료스트립
 senu — 기장 환경 설립
 c 라마데라 — 함께 가는은 목대인 그와 티바이의 관리
 in이는 함께나면 서비스
 syn — 시스템 회장 레바디에의 용우 elyacostig — Golffor Pic Sylvin S(FA))
 builtime — HS SN
 huis — Top-level code environment varnings — 打工 NN
 dataclasses — NV NN 意味在
 contextile — with 表 전域企業 NN HUNN
 abc — 多於 NNA 意味在 \* atexat - Bill Millia · Industry - Motor New Biol \* codens - Wilet III. BRIS \* import lib - amort 2 75 · importion.metadata 4855 · syntable - Name of the sale tokes — 하이턴 구분 분석 ESM 사용되는 설수
 keyeard — 하이턴 하시고 설수
 tokesuze — 하이턴 소스템 하면 EDM(이미 · taboarry - URB BRES BR · dia - strip wide up of the state · MS ESP SEAMS · winner - ESSP REPORT WHO

\* unittest -- DRINGSE ENGELS

• •

- 1. 운영 체제 인터페이스 os 모듈
  - 운영 체제와 상호작용할 수 있는 다양한 기능 제공
  - 파일 시스템 작업(파일 및 디렉터리 생성, 삭제, 이동 등)과 환경 변수 작업 등 수행

```
import os
# 현재 작업 디렉터리 출력
current directory = os.getcwd()
print("Current Directory:", current directory)
# 새로운 디렉터리 생성
os.mkdir("new folder")
print("Created 'new folder'")
# 디렉터리 존재 여부 확인
print("Does 'new_folder' exist?", os.path.exists("new_folder"))
# 디렉터리 이름 변경
os.rename("new folder", "renamed folder")
print("Renamed 'new folder' to 'renamed folder'")
# 디렉터리 삭제
os.rmdir("renamed folder")
print("Deleted 'renamed folder'")
```

#### 2. 파일 탐색 - glob

- 파일 경로와 일치하는 모든 파일 이름을 찾는 데 사용
- 파일 이름의 패턴 매칭 수행, 와일드카드(\*, ?, [])를 사용하여 특정 패턴에 맞는 파일 서치

```
import glob
# 현재 디렉터리에서 .py 파일을 모두 찾기
python_files = glob.glob("*.py")
print("Python files in the current directory:", python_files)
```

#### 3. 수학 모듈 - math

- 수학적인 계산에 필요한 다양한 함수와 상수 제공
- math 모듈을 사용하여 기본적인 산술 연산부터 고급 수학 함수(삼각 함수, 로그 함수 등)까지
   다양한 수학적 작업 수행

```
import math
# 수학 상수
print("PI:", math.pi)
print("Euler's number (e):", math.e)
# 기본 수학 함수
print("Square root of 16:", math.sqrt(16))
print("3 raised to the power of 4:", math.pow(3, 4))
print("Factorial of 5:", math.factorial(5))
print("GCD of 60 and 48:", math.gcd(60, 48))
# 삼각 함수
angle rad = math.pi / 4 # 45도(라디안)
print("sin(45 degrees):", math.sin(angle rad))
print("cos(45 degrees):", math.cos(angle rad))
# 로그 함수
print("Natural log of 10:", math.log(10))
print("Log base 10 of 100:", math.log10(100))
# 올림/내림 함수
print("Ceiling of 4.3:", math.ceil(4.3))
print("Floor of 4.7:", math.floor(4.7))
```

- 4. 인터넷 액세스 request
  - HTTP 요청을 보낼 수 있게 해주는 라이브러리
  - GET, POST 등의 HTTP 메서드를 사용하여 웹 서버와 상호작용

```
# !pip install requests
import requests
# GET 요청 사용, 웹페이지 내용 가져오기
response = requests.get("https://api.github.com")
# 상태 코드 출력
print("Status Code:", response.status code)
# 응답 내용 출력 (텍스트 형태)
print("Response Text:", response.text)
# JSON 데이터로 변환하여 출력
data = response.json()
print("JSON Response:", data)
```

#### 5. 날짜와 시간 – datetime

- 날짜와 시간을 처리하는 데 사용되는 표준 라이브러리
- 현재 날짜와 시간, 특정 날짜 및 시간 계산, 날짜 형식 변환 등 수행

```
# 날짜 계산 (100일 후의 날짜)
future_date = today + datetime.timedelta(days=100)
print("Date 100 days from today:", future_date)

# 문자열을 날짜로 변환
date_str = "2024-08-25"
date_obj = datetime.datetime.strptime(date_str, "%Y-%m-%d")
print("Converted Date from String:", date_obj)

# 날짜를 문자열로 변환
date_to_str = now.strftime("%Y-%m-%d %H:%M:%S")
print("Formatted Date to String:", date_to_str)
```

#### 5. 날짜와 시간 - datetime

- 날짜와 시간을 처리하는 데 사용되는 표준 라이브러리
- 현재 날짜와 시간, 특정 날짜 및 시간 계산, 날짜 형식 변환 등 수행

```
import datetime

# 현재 날짜와 시간 가져오기
now = datetime.datetime.now()
print("Current Date and Time:", now)

# 특정 날짜와 시간 생성
specific_date = datetime.datetime(2023, 12, 25, 10, 30, 0)
print("Specific Date and Time:", specific_date)

# 날짜 부분만 가져오기
today = datetime.date.today()
print("Today's Date:", today)
```



## Standard Library

• 실습

10.1.3. Standard Library. ipynb

## 🤚 기타 유용한 Library

- 데이터 처리 : <u>Pandas</u>, <u>Numpy</u>, NetworkX
- 고속화 : Numba, Cython
- 딥러닝: Pytorch, <u>Tensorflow</u>, <u>Keras</u>, Caffe, Theano, ScikitLearn, Scipy, Xgboost
- 이미지 및 영상 : Pillow, OpenCV, MoviePy
- 시각화: <u>Matplotlib</u>, Plotly, <u>Seaborn</u>, Bokeh, ggplot2
- 크롤링 : Beautifulsoup, Selenium, Scrapy
- 백엔드, 웹 등 : Django, Requests, Bottle
- 게임: PyGame
- Gui 등 : Tkinter, PyQt, Pyinstaller

#### 1. Visual Studio code

```
# Will it be clear tomorrow at this time in Milan (Italy) ?
       forecast = mgr.forecast_at_place('Milan,IT', 'daily')
       answer = forecast.will_be_clear_at(timestamps.tomorrow())
       # ----- PAID API KEY example -----
       config dict = config.get default config for subscription type('professional')
       owm = OWM('your paid OWM API key', config_dict)
                                                                                  출력 디버그 콘솔
                              JUPYTER
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
새로운 크로스 플랫폼 PowerShell 사용 https://aka.ms/pscore6
                                                    pip install "library 이름" (python version 2)
                  \Desktop\SW 작업> pip install pyowm
PS C:\Users\
                                                    pip3 install "library 이름" (python version 3)
 현재 작업 중인 위치가 맞는지 확인!
                                                                                     த் Jupyter Server: local 👂
```



## How to install library

2. colab



!로시작 : command line interface (CLI)

• 파이썬 표준 라이브러리 <a href="https://docs.python.org/ko/3/library/index.html">https://docs.python.org/ko/3/library/index.html</a>

# THANK YOU