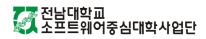


버전관리 Git, Github 이해와 활용

5. 실습(1) Python패키지 배포 및 활용

유재명(wesleyok@jnu.ac.kr)

목차



- 1. Git 원리
- 2. 개발환경 만들기
- 3. 로컬에서 Git 사용하기
- 4. 로컬저장소, 원격저장소 연계 및 Github 활용
- 5. 실습(1) Python 패키지 배포 및 활용
 - 1) Python 패키지 생성 원리
 - 2) Python 패키지 배포 및 활용 절차
 - 3) My Python 패키지 활용
 - 4) 부록) 파이썬 설치 및 jupyterlab 활용
- 6. 실습(2) Node.js 패키지 배포 및 활용
- 7. 브런치 이해 및 활용

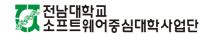
1) Python 패키지 생성원리(1)



- ★ 파이썬의 사용자정의 패키지란
 - 여러 개의 모듈(파이썬 파일)을 하나의 패키지로 묶어 재사용하는 것
 - 패키지의 기본 구조

```
setup.py
from setuptools import setup, find_packages
setup(
                패키지 이름
  name='mylib',
  version='0.0.2', 패키지 버전
  description='my functions', 패키지에 대한 간단한 설명
  url='github-my_library', 패키지의 웹사이트 또는 github URL
  author='Yoo Jae Myeong', <mark>패키지 저자</mark>
  author_email='yoojaemyeong@gmail.com', 저자의 이메일
  license='free', 패키지 라이선스(전남대학교)
  packages=find_packages(), 패키지 목록 자동 검색
  install_requires=['PyQt5','numpy'], 이 패키지가 의존하는 외부 목록
                                Numpy>=1.21.0
                                PvQt5>=5.0.0
     init _.py
 from .hi import *
 from .cal import *
             hi.py
                                      cal.py
def hi(name):
                                     #사칙연산
  if name:
                                     def add(a,b):
     print("안녕하세요 ", name, "님!!")
                                       return a+b
  else:
                                     def sub(a,b):
     print("안녕하세요")
                                       return a-b
```

1) Python 패키지 생성원리(2)



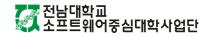
- ★ 파이썬의 사용자정의 패키지란
 - 여러 개의 모듈(파이썬 파일)을 하나의 패키지로 묶어 재사용하는 것
 - 패키지의 기본 구조를 이용하여 모듈 생성 및 사용

```
F:\PROJECTS\MY_LIBRARY
                                                               F:\Projects\my_library>python test.py
F:\Projects>tree /f my_library
                                           README.md
                                                               안녕하세요
                                                                           kim 님 !!
Data 볼륨에 대한 폴더 경로의 목록입니다.
                                           requirements.txt
                                                               안녕하세요
볼륨 일련 번호는 F032-03F9입니다.
                                           setup.py
                                                               4+5= 9
                                           test.py
                                                               4-7= -4
F:\PROJECTS\MY_LIBRARY | 패키지 프로젝트명
   README.md 패키지 설명(옵션)
                                          -mylib
                                                               F:\Projects\my_library>tree /f .
   requirements.txt 패키지 모듈 작성시 필요한
라이브러리 목록(옵션)
                                               cal.py
                                                               Data 볼륨에 대한 폴더 경로의 목록입니다
                                               hi.py
                                                                    일련 번호는 F032-03F9입니다.
   setup.py 패키지 환경 구성 파일
                                               __init__.py
                                                               F:\PROJECTS\MY_LIBRARY
                                                                  README.md
                                          from mylib import *
                                                                  requirements.txt
 mylib 패키지명, 패키지의 루트 디렉토리
                                                                  setup.py
        cal.py
                                          hi('kim')
                                                                  test.py
       hi.py
                                          hi(")
                                                                -mylib
                                          print('4+5=', add(4,5))
        __init__.py 패키지 초기화 파일
                                                                      cal.py
                                          print('4-7=', sub(3,7))
                                                                      hi.py
                                           메모장을 이용하여
                                                                      __init__.py
                                           test.py 작성
                                                                      _pycache__
                                                                          cal.cpython-312.pyc
                                                                          hi.cpython-312.pyc
                                                                          __init__.cpython-312.pyc
```



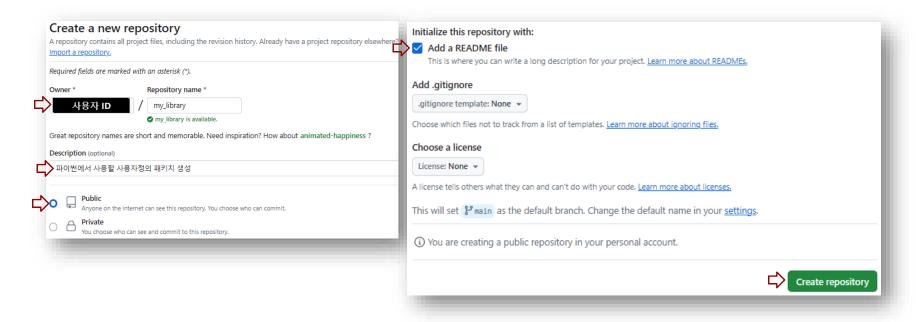
→ 파이썬 패키지 배포 및 활용 절차

- ① Github 원격저장소 만들기 <= 이곳이 파이썬 패키지 배포 저장소
 - Repository name: my_library
 - Description: "파이썬에서 사용할 사용자정의 패키지 생성"
 - Public 선택
 - Add a README file 체크 후 [Create repository] 클릭
- ② 내PC의 Projects 폴더에서 원격저장소 복제(git clone) F:\Projects>git clone https://github.com/[User ID]/my_library.git
- ③ Projects/my_library 폴더(Working Directory)로 이동하여 패키지 작업 진행
- ④ Working Directory에서 git Staging Area로 파일 추가(git add .) 및 커밋(git commit -m "메시지") 진행
- ⑤ 로컬저장소에서 원격저장소로 배포(git push)
- ⑥ 새로운 장소에서 배포된 원격저장소의 파이썬 패키지 설치(pip install)
- ⑦ 내가 만든 Python 패키지 활용
- ⑧ 내가 만든 Python 패키지 버전 업데이트



◆ 파이썬 패키지 배포 및 활용 절차

- ① Github 원격저장소 만들기 <= <mark>이곳이 파이썬 패키지 배포 저장소</mark>
 - Repository name: my_library
 - Description: "파이썬에서 사용할 사용자정의 패키지 생성"
 - Public 선택
 - Add a README file 체크 후 [Create repository] 클릭





- → 파이썬 패키지 배포 및 활용 절차
 - ② 내PC의 Projects 폴더에서 원격저장소 복제(git clone) F:\Projects>git clone https://github.com/[User ID]/my_library.git
 - ③ Projects/my_library 폴더(Working Directory)로 이동하여 패키지 작업 진행

```
my_library 黃더

Setup.py

from setuptools import setup, find_packages setup(
    name='mylib',
    version='0.0.2',
    description='my functions',
    url='github-my_library',
    author='Yoo Jae Myeong',
    author_email='yoojaemyeong@gmail.com',
    license='free',
    packages=find_packages(),
    install_requires=['PyQt5','numpy'],
)
```

```
my_library\mylib 폴더
     init _.py
 from .hi import *
 from .cal import *
             hi.py
                                     cal.py
def hi(name):
                                    #사칙연산
   if name:
                                    def add(a,b):
     print("안녕하세요 ", name, "님!!")
                                       return a+b
   else:
                                    def sub(a,b):
     print("안녕하세요")
                                       return a-b
```



- ◆ 파이썬 패키지 배포 및 활용 절차
 - ④ Working Directory에서 git Staging Area로 파일 추가(git add .) 및 커밋(git commit -m "메시지") 진행

```
my_library>git add .
```

```
,my_library>git commit -m "python 패키지 모듈 추가"
```

```
[main 0d7427b] add my packages
5 files changed, 26 insertions(+)
create mode 100644 mylib/__init__.py
create mode 100644 mylib/cal.py
create mode 100644 mylib/hi.py
create mode 100644 requirements.txt
create mode 100644 setup.py
```



- → 파이썬 패키지 배포 및 활용 절차
 - ⑤ 로컬저장소에서 원격저장소로 배포(git push)
 - 로컬저장소에서 원격저장소 별칭 확인(git remote -v)

```
F:\Projects\my_library>git remote -v
origin https://github.com/ 사용자ID /my_library.git (fetch)
origin https://github.com/ 사용자ID /my_library.git (push)
```

- 로컬저장소에서 원격저장소로 배포 F:\Projects\my_library>git push origin main

- 로컬저장소에서 git log 확인

```
F:\Projects\my_library>git log --oneline --all
9ab4294 (HEAD -> main, origin/main, origin/HEAD) Add my packages
fff491c Add my packages
437fefe add my packages
0d7427b add my packages
cfb97b0 Initial commit
```

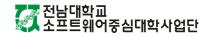


- ★ 파이썬 패키지 배포 및 활용 절차
 - ⑥ 새로운 장소에서 배포된 원격저장소의 파이썬 패키지 설치
 - 현재 python 모듈 중 mylib 존재 여부 확인
 F:\Projects\my_library_test2>pip list | findstr mylib
 - 패키지 설치(<mark>pip install git+https://github.com/[ID]/my_library.git</mark>)

```
F:\Projects\my_library_test2>pip install git+https://github.com/
                                                                                                                          /my_library.git
                                                                                                           사용사ID
Collecting git+nttps://gitnub.com/yoojaemyeong/my_library.git
Cloning https://github.com/yoojaemyeong/my_library.git to c:\users\master\appdata\local\temp\pip-req-build-0ddtbkie
  Running command qit clone --filter=blob:none --quiet https://qithub.com/yoojaemyeong/my_library.qit 'C:\Users\master\A
ppData\Local\Temp\pip-req-build-0ddtbkie'
  Resolved https://github.com/yoojaemyeong/my_library.git to commit 9ab4294ace5415455d7e149b8d077af6b9058dc1
  Preparing metadata (setup.py) ... done
Requirement already satisfied: PyOt5 in c:\python312\lib\site-packages (from mylib==0.0.2) (5.15.10)
Requirement already satisfied: numpy in c:\users\master\appdata\roaming\python\python312\site-packages (from mylib==0.0.
Requirement already satisfied: PyQt5-sip<13,>=12.13 in c:\python312\lib\site-packages (from PyQt5->mylib==0.0.2) (12.13.
Requirement already satisfied: PyQt5-Qt5>=5.15.2 in c:\python312\lib\site-packages (from PyQt5->mylib==0.0.2) (5.15.2)
Building wheels for collected packages: mylib
  Building wheel for mylib (setup.py) ... done
  Created wheel for mylib: filename=mylib-0.0.2-py3-none-any.whl size=1635 sha256=adf27956bb04581b44f5daf8ce10adba9c9f3c
57447e3438db67b990f51d0e10
  Stored in directory: C:\Users\master\AppData\Local\Temp\pip-ephem-wheel-cache-90kp7lkh\wheels\25\da\50\dc9202fdcf0d7a8
0a1dd5502fcc778738c5e3a7f4c573baea4
Successfully built mylib
```

- python 패키지 확인(pip list | findstr mylib)

```
F:\Projects\my_library_test2>pip list | findstr mylib mylib 0.0.2
```



◆ 파이썬 패키지 배포 및 활용 절차

⑦ 내가 만든 Python 패키지 활용

```
F:\Projects\my_library_test2>notepad test.py
F:\Projects\my_library_test2>python test.py
안녕하세요 Everyone 님 !!
3+4 = 7
10 - 3 = 7
```

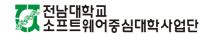
test.py

```
from mylib import *

hi('Everyone')
print('3+4 = ', add(3,4))
print('10 - 3 = ',sub(10,3))
```



- → 파이썬 패키지 재배포 및 활용 절차
 - ⑧ 내가 만든 Python 패키지 버전 업데이트
 - 기존 모듈
 - hi.py : hi()
 - cal.py : add(a,b), sub(a,b)
 - 추가 모듈
 - hi.py : hi(), bye()
 - cal.py : add(a,b), sub(a,b), multiply(a,b), divide(a,b)
 - 패키지 업데이트 작업순서
 - ③ 로컬저장소에서 각 파일에서 추가 모듈 작업 진행
 - · hi.py, cal.py, setup.py(버전: 0.0.2 -> 0.0.3)
 - ⑤ [working area] (git add)> [staging area] (git commit)> [로컬저장소]
 - © [로컬저장소] (git push)> [원격저장소]
 - ① 다른 장소에서 변경된 내용 확인



- ◆ 파이썬 패키지 배포 및 활용 절차
 - ⑧ 내가 만든 Python 패키지 버전 업데이트
 - 작업순서
 - ③ 로컬저장소에서 각 파일에서 추가 모듈 작업 진행
 - · hi.py, cal.py, setup.py(버전: 0.0.2 -> 0.0.3)

my_library>notepad setup.py

```
from setuptools import setup, find_packages

setup(
    name='mylib',
    version='0.0.3',
    description='my functions',
    url='github-my_library',
    author='wesleyok',
    author_email='yoojaemyeong@gmail.com',
    license='free',
    packages=find_packages(),
    install_requires=['numpy'],
```

my_library>notepad mylib\hi.py

```
def hi(name):
    if name:
        print("안녕하세요 ", name, "님 !! ")
    else:
        print("안녕하세요")

def bye():
    print("다음에 만나요 ~~~")
```

my_library>notepad mylib\cal.py

```
def add(a,b):
return a+b
def sub(a,b):
return a-b
def multiply(a,b):
return a*b
def divide(a,b):
return a/b
```

#사칙연산



- ◆ 파이썬 패키지 배포 및 활용 절차
 - ⑧ 내가 만든 Python 패키지 버전 업데이트
 - 작업순서
 - ⑤ [working area] (git add)> [staging area] (git commit)> [로컬저장소]

```
F:\Projects\my_library>git status

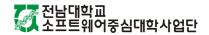
Changes not staged for commit:
   (use "git add <file>..." to update what will be committed)
   (use "git restore <file>..." to discard changes in working directory)
        modified: mylib/cal.py
        modified: mylib/hi.py
        modified: setup.py

no changes added to commit (use "git add" and/or "git commit -a")

F:\Projects\my_library>git add setup.py mylib/cal.py mylib/hi.py

F:\Projects\my_library>git commit -m "모듈 업데이트 0.0.3"

3 files changed, 11 insertions(+), 5 deletions(-)
```



★ 파이썬 패키지 배포 및 활용 절차

F:\Projects\my_library>git remote -v

- ⑧ 내가 만든 Python 패키지 버전 업데이트
 - 작업순서
 - © [로컬저장소] (git push)> [원격저장소]

```
origin https://github.com/ 사용자ID /my_library.git (fetch)
origin https://github.com/ 사용자ID /my_library.git (push)

F:\Projects\my_library>git push origin main
Enumerating objects: 16, done.
Counting objects: 100% (16/16), done.
Delta compression using up to 24 threads
Compressing objects: 100% (11/11), done.
Writing objects: 100% (11/11), 1.58 KiB | 1.58 MiB/s, done.
Total 11 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/ 사용자ロ J/my_library.git
9ab4294..5bcb633 main -> main
```



- **★ 파이썬 패키지 배포 및 활용 절차**
 - ⑧ 내가 만든 Python 패키지 버전 업데이트
 - 작업순서
 - ⑥ 다른 장소에서 변경된 내용 확인(새버전 설치: pip install git+https://...)

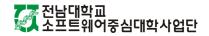
```
F:\Projects\my_library_test2>pip install git+https://github.com/ 자용자 ID /my_library.git
                                                                                  my_library_test2>notepad test.py
Collecting git+https://github.com/ ப் பார்க்கம் /my_library.git
                                                                                  from mylib import *
 Cloning https://github.com/ ার্মান /my_library.git to c:\users\master\appdata\local\te
 Running command git clone --filter=blob:none --quiet https://github.com/ ABAID /my_li
                                                                                   hi('Everyone')
ppData\Local\Temp\pip-reg-build-myvghdtl'
                                                                                   print('3+4 = ', add(3,4))
 Resolved https://github.com/ 사용사고 /my_library.git to commit 5bcb6339b9395d7f22218peb
                                                                                   print('10 - 3 = ',sub(10,3))
 Preparing metadata (setup.py) ... done
                                                                                   print("### 추가된 모듈 테스트 ###")
Requirement already satisfied: numpy in c:\users\master\appdata\roaming\python\python312\sit
                                                                                   bye()
3) (1.26.3)
                                                                                   print('3x4 = ', multiply(3,4))
Building wheels for collected packages: mylib
                                                                                   print('10 / 3 = ', divide(10,3))
 Building wheel for mylib (setup.py) ... done
                                                                                F:\Projects\my_library_test2>pip list | findstr mylib
 Created wheel for mylib: filename=mylib-0.0.3-py3-none-any.whl size=1686 sha256=e216387 d3
9bda6766cf548dbabe3918dbd2
                                                                                mylib
                                                                                                             0.0.3
 Stored in directory: C:\Users\master\AppData\Local\Temp\pip-ephem-wheel-cache-4szh3ogp\whe
0a1dd5502fcc778738c5e3a7f4c573baea4
                                                                                F:\Projects\my_library_test2>python test.py
Successfully built mylib
                                                                                안녕하세요 Everyone 님 !!
Installing collected packages: mylib
                                                                                3+4 = 7
 Attempting uninstall: mylib
                                                                                10 - 3 = 7
   Found existing installation: mylib 0.0.2
                                                                                 ### 추가된 모듈 테스트 ###
                                                                                다음에 만나요 ~~~
   Uninstalling mylib-0.0.2:
                                                                                3x4 = 12
     Successfully uninstalled mylib-0.0.2 -
                                                                                10 / 3 = 3.33333333333333333
Successfully installed mylib-0.0.3
```



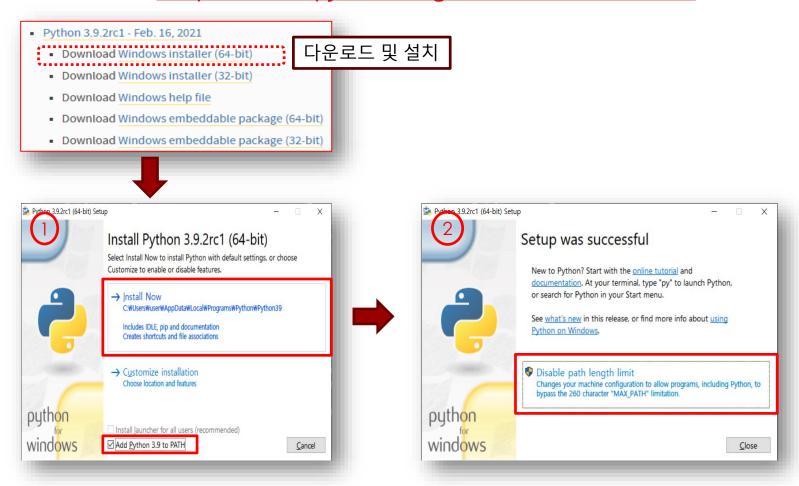
부록

파이썬 설치 및 Jupyterlab-git 활용

파이썬 설치

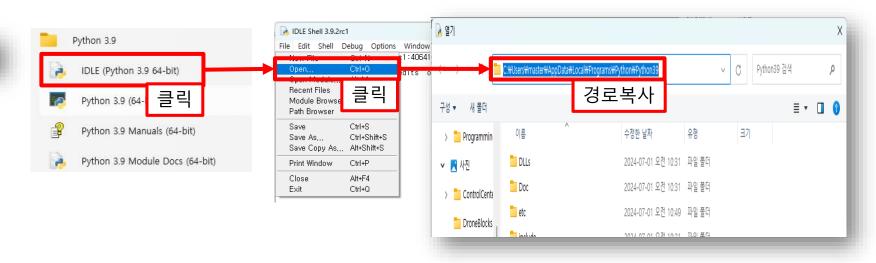


- → 버전: python-3.9.2rc1-amd64.exe
- ◆ 다운로드: https://www.python.org/downloads/windows/









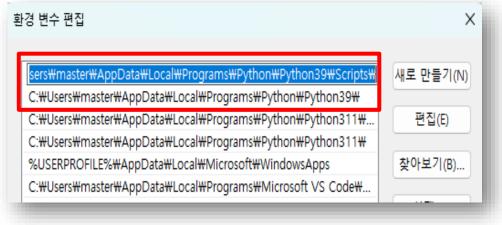
위에서 복사한 경로를 시스템 환경변수에 등록하는 방법

② 실행장(■+R)을 열고 control system 입력후 엔터 > 고급 시스템 설정 클릭 >

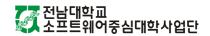
<mark>환경변수</mark> 클릭 > <mark>사용자 변수</mark>에서 <mark>Path 선택 후 편집 비튼</mark> 클릭 > 아래와 같이 새로 만들기로 등록







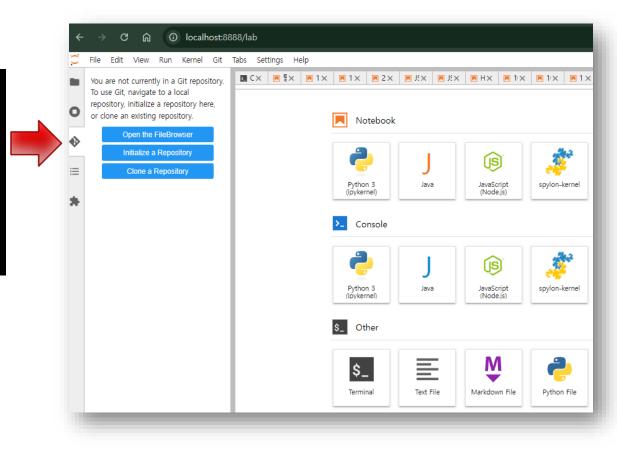
환경구성 및 주피터노트북 실행



- ◆ 탐색기를 열어(■+E) 작업폴더 만들기:
 - 바탕 화면에 "인공지능" 폴더 생성 후 클릭
 - 탐색기 주소창에 "cmd" 입력 후 엔터 > 명령창 생성

명령창에서 순서대로 실행

pip install tensorflow python -m pip install --upgrade pip pip install pandas pip install scikit-learn pip install scipy pip install statsmodels pip install jupyterlab jupyterlab-git pip install jupyterlab-visualpython jupyter lab





감사합니다.