Jupyter Lab on Virtual Environment

가상 환경(Virtual Environment)은 **독립된 Python 실행 환경**으로, 시스템의 전역 Python 환경과는 분리된 환경을 제공합니다. 각 가상 환경은 자신만의 Python 인터프리터, 패키지 관리자(pip), 패키지 설치 디렉토리를 가집니다. 이를 통해, 프로젝트마다 필요한 패키지와 버전을 독립적으로 설치하고 관리할 수 있습니다.

**가상 환경의 특징**

* **독립성**: 각 가상 환경은 다른 가상 환경과 독립되어 있으며, 패키지나 의존성에 영향을 주지 않습니다.
* **격리된 패키지 관리**: 프로젝트마다 독립된 패키지를 관리할 수 있습니다. 즉, 가상 환경 내부에 설치된 패키지는 그 가상 환경 내에서만 사용되며, 시스템 전역 패키지에 영향을 주지 않습니다.
* **Python 버전 관리**: 여러 Python 버전이 필요한 경우, 각각 다른 Python 버전으로 가상 환경을 생성할 수 있습니다.

전역 **Python과 가상 환경의 관계**

Python은 **기본적으로 전역 시스템 환경**에 설치되지만, 가상 환경을 생성하면 Python의 실행 파일과 패키지 설치 경로가 **가상 환경 내부로 복사**됩니다. 즉, 가상 환경 안에서 python이나 pip 명령을 실행하면, **가상 환경 내의 Python**과 패키지 관리 도구가 실행됩니다.

* **전역 Python 환경**: 시스템에 설치된 Python과 그에 설치된 패키지.
* **가상 환경**: 프로젝트별로 독립적으로 생성된 Python 환경으로, 전역 환경과는 별도로 패키지를 관리합니다.

**가상 환경 내 Python 실행**

* 가상 환경을 생성하면 해당 가상 환경은 **자체적으로 Python 인터프리터**를 갖습니다.
* 가상 환경을 활성화한 후에 python 명령을 실행하면, **시스템 전역의 Python이 아니라 가상 환경의 Python**이 실행됩니다.

1. 가상환경(virtual environment) 만들기

STEP 1. Python 설치 및 환경변수 등록

STEP 2. Project 디렉토리(F:\examination) 만들기

STEP 3. 가상환경(가상환경이름=vir-env.jupyter) 생성

PS F:\examination> python -m venv vir-env.jupyter

가상환경의 HOME Dir = F:\examination\vir-env.jupyter

STEP 4. (Windows의 PowerShell) Script 실행 정책 변경

**현재 PowerShell 세션**에서만 **스크립트 실행 제한을 우회**할 수 있게 설정합니다.

PS F:\examination> Set-ExecutionPolicy -Scope Process -ExecutionPolicy Bypass

STEP 5. 가상환경(vir-env.jupyter)의 Scripts 폴더로 이동

PS F:\examination> cd vir-env.jupyter

PS F:\examination\vir-env.jupyter> cd Scripts

PS F:\examination\vir-env.jupyter\Scripts> ls

디렉터리: F:\examination\vir-env.jupyter\Scripts

Mode LastWriteTime Length Name

---- ------------- ------ ----

-a---- 2024-10-04 오전 1:10 2074 activate

-a---- 2024-10-04 오전 1:10 1021 activate.bat

-a---- 2024-10-04 오전 1:10 26199 Activate.ps1

-a---- 2024-10-04 오전 1:10 393 deactivate.bat

-a---- 2024-10-04 오전 1:10 108408 pip.exe

-a---- 2024-10-04 오전 1:10 108408 pip3.12.exe

-a---- 2024-10-04 오전 1:10 108408 pip3.exe

-a---- 2024-10-04 오전 1:10 270104 python.exe

-a---- 2024-10-04 오전 1:10 258840 pythonw.exe

STEP 6. 가상환경(vir-env.jupyter)를 Active 하는 Script 실행후 확인

PS F:\examination\vir-env.jupyter\Scripts> ./Activate.ps1

(vir-env.jupyter) PS F:\examination\vir-env.jupyter\Scripts>

STEP 7. 가상환경에 맞는 독립적인 Jupyter 설정 파일을 관리할 수 있어 환경별로 설정을 쉽게 분리시키기 위해서 가상환경(vir-env.jupyter)의 jupyter 설정폴더 생성

PS F:\examination\vir-env.jupyter> mkdir .jupyter

STEP 8. 가상환경(vir-env.jupyter) 활성화시 Jupyter의 설정 파일 위치 지정

./Activate.bat Or ./Activate.ps1 파일에 설정 추가

set JUPYTER\_CONFIG\_DIR=%VIRTUAL\_ENV%\.jupyter

1. Jupyter Lab 설치

(vir-env.jupyter) PS F:\examination\vir-env.jupyter\Scripts> pip install

Jupyterlab

1. Kernel 설치

STEP 1. 설치된 커널 확인

(vir-env.jupyter) PS F:\examination\vir-env.jupyter> jupyter kernelspec list

STEP 2. 커널 설치

(vir-env.jupyter) PS F:\examination\vir-env.jupyter\Scripts> pip install ipykernel

STEP 3. 가상환경에 커널 등록

python -m ipykernel install --prefix="가상환경의 홈 디렉토리" --name "커널명" --display-name "Jupyter Notebook/Lab에서 표시할 커널명"

(vir-env.jupyter) PS F:\examination\vir-env.jupyter> python -m ipykernel install --prefix="F:\examination\vir-env.jupyter" --name "mykernel" --display-name "Python(mykernel4exam)"

1. workspace에서 Jupyter Lab 실행

(vir-env.jupyter) PS F:\examination\vir-env.jupyter\Scripts> cd ..

(vir-env.jupyter) PS F:\examination\vir-env.jupyter> cd ..

(vir-env.jupyter) PS F:\examination> cd workspace

(vir-env.jupyter) PS F:\examination\workspace> jupyter lab