**Python Environments and Package Management**

**1. Creating and Using a Virtual Environment**

A **virtual environment (venv)** is an isolated Python workspace where you can install dependencies without affecting the global Python installation.

**Steps to Create and Use a Virtual Environment:**

1. **Create a Virtual Environment:**

python -m venv myenv

This creates a folder myenv containing the isolated environment.

1. **Activate the Virtual Environment:**
   * On **Windows (CMD/PowerShell):**

myenv\Scripts\activate

* + On **Mac/Linux:**

source myenv/bin/activate

1. After activation, the terminal prompt changes to indicate the active environment.
2. **Deactivate the Virtual Environment:**

deactivate

This exits the virtual environment and returns to the system Python.

**2. Packaging Your Python Application**

Packaging helps distribute Python applications as reusable modules.

* Create a Python package
* Define dependencies
* Package and install your application

**Step 1: Create Project Structure**

* mkdir mymodule
* touch mymodule/\_\_init\_\_.py
* touch mymodule/main.py
* touch setup.py
* touch README.md

**Step 2: Write a Simple Python Module**

Edit mymodule/main.py and add the following code:

def greet(name):

return f"Hello, {name}! Welcome to my package."

if \_\_name\_\_ == "\_\_main\_\_":

print(greet("User"))

**Step 3: Define setup.py**

Edit setup.py and add:

from setuptools import setup, find\_packages

setup(

name="mymodule",

version="0.1",

packages=find\_packages(),

install\_requires=[],

entry\_points={

"console\_scripts": [

"greet=mymodule.main:greet"

]

},

)

**Step 4: Create a Virtual Environment and Install Dependencies**

1. Create a virtual environment:

python -m venv venv

1. Activate the virtual environment:
   * **Windows:**

venv\Scripts\activate

* + **Mac/Linux:**

source venv/bin/activate

1. Install setuptools and wheel:

pip install setuptools wheel

**Step 5: Build the Package**

1. Run the following command to create the package:

python setup.py sdist bdist\_wheel

1. A dist/ directory will be created with .tar.gz and .whl files.

**Step 6: Install the Package Locally**

1. Install the package:

pip install dist/mymodule-0.1-py3-none-any.whl

1. Test the installed package in Python:

from mymodule.main import greet

print(greet("Tester"))

**3. Using requirements.txt and PIP to Manage Dependencies**

A **requirements.txt** file lists all dependencies needed for the project.

**Steps to Manage Dependencies:**

1. **Generate a requirements.txt file:**

Pandas

Numpy

This saves all installed packages and versions.

1. **Install Dependencies from a requirements.txt file:**

pip install -r requirements.txt

This installs all dependencies listed in the file.

1. **Uninstall All Installed Packages (if needed):**

pip uninstall -r requirements.txt -y