Setting up virtual environments and dependencies in Python 3

Green background represents real, recognized code/commands

Making a python virtual environment:
python3 -m venv <your-environment-name>
Example:
python3 -m env test-env

Activating an environment:
Unix/MacOS: source <your-environment-name>/bin/activate
Windows: <your-environment-name>\Scripts\activate.bat
Example:
source test-env/bin/activate
or
test-env\Scripts\activate.bat

Exiting virtual environment:
deactivate or conda deactivate (depending on distribution/version installed)

Installing packages in/for an environment:
Use pip, etc. like you normally would
Example:
pip install numpy

Viewing installed packages (via pip):
pip list

Install many (required) packages at once:
pip install -r requirements.txt
Where requirements.txt contains a list of packages, with specific versions using ==
Example contents:
requests==2.7.0
numpy

Export installed packages to requirements.txt:
pip freeze > requirements.txt