

Python virtual environment

Python virtual environment

Python virtual environment

Create a virtual environment

Activate virtual environment

Install the third line package

Exit the virtual environment

Thonny Editor

Change the virtual environment Python version

In previous versions of the operating system, the program could be installed using Python's pip tool:

```
$ pip install buildhat
```

But in newer versions of Raspberry Pi OS systems, this is not allowed; if you try to install a Python package system-wide, you will receive an error similar to the following:

```
error: externally-managed-environment

× This environment is externally managed
╰─> To install Python packages system-wide, try apt install
    python3-xyz, where xyz is the package you are trying to
    install.

    If you wish to install a non-Debian-packaged Python package,
    create a virtual environment using python3 -m venv path/to/venv.
    Then use path/to/venv/bin/python and path/to/venv/bin/pip. Make
    sure you have python3-full installed.

    For more information visit http://rptl.io/venv

note: If you believe this is a mistake, please contact your Python installation or OS distribution
provider. You can override this, at the risk of breaking your Python installation or OS, by passi
ng --break-system-packages.
hint: See PEP 668 for the detailed specification.
```

We used another method to eliminate this warning when building the jupyter lab environment, but here we only introduce how to build a python virtual environment.

Python virtual environment

To use a virtual environment, you need to create a container to store the environment.

Create a virtual environment

Create and enter the Demo_Project folder, then use Python's built-in module venv to create a virtual environment named env:

```
mkdir Demo_Project
cd Demo_Project
python -m venv env
```

```
pi@raspberrypi: ~/Demo_Project
File Edit Tabs Help
pi@raspberrypi:~$ mkdir Demo_Project
pi@raspberrypi:~$ cd Demo_Project
pi@raspberrypi:~/Demo_Project$ python -m venv env
pi@raspberrypi:~/Demo_Project$ ls
env
```

After the operation is successful, there will be an additional env folder under the folder.

Activate virtual environment

In this directory is a complete Python distribution; activate the virtual environment and make this Python version the currently used version:

```
source env/bin/activate
```

```
pi@raspberrypi:~/Demo_Project$ source env/bin/activate
(env) pi@raspberrypi:~/Demo_Project$ which python
/home/pi/Demo_Project/env/bin/python
```

At this point, we are no longer using the system Python, but the version of Python included in the virtual environment:

- Any changes made here will not cause problems for your system Python;
- Any new modules installed into the environment will also not affect system Python.

Install the third line package

```
pip install jupyterlab
```

```
pi@raspberrypi:~/Demo_Project$ source env/bin/activate
(env) pi@raspberrypi:~/Demo_Project$ which python
/home/pi/Demo_Project/env/bin/python
(env) pi@raspberrypi:~/Demo_Project$ pip install jupyterlab
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Collecting jupyterlab
  Downloading https://www.piwheels.org/simple/jupyterlab/jupyterlab-4.0.9-py3-none-any.whl (9.2 MB)
    9.2/9.2 MB 2.1 MB/s eta 0:00:00
Collecting async-lru>=1.0.0
  Downloading https://www.piwheels.org/simple/async-lru/async_lru-2.0.4-py3-none-any.whl (6.1 kB)
Collecting ipykernel
  Downloading https://www.piwheels.org/simple/ipykernel/ipykernel-6.27.1-py3-none-any.whl (114 kB)
    114.6/114.6 kB 175.9 kB/s eta 0:00:00
Collecting jinja2>=3.0.3
  Downloading https://www.piwheels.org/simple/jinja2/Jinja2-3.1.2-py3-none-any.whl (133 kB)
    133.1/133.1 kB 413.9 kB/s eta 0:00:00
Collecting jupyter-core
  Downloading https://www.piwheels.org/simple/jupyter-core/jupyter_core-5.5.1-py3-none-any.whl (28 kB)
Collecting jupyter-lsp>=2.0.0
  Downloading https://www.piwheels.org/simple/jupyter-lsp/jupyter_lsp-2.2.1-py3-none-any.whl (66 kB)
    66.0/66.0 kB 2.1 MB/s eta 0:00:00
Collecting jupyter-server<3,>=2.4.0
  Downloading https://www.piwheels.org/simple/jupyter-server/jupyter_server-2.12.1-py3-none-any.whl (380 kB)
    380.2/380.2 kB 378.5 kB/s eta 0:00:00
Collecting jupyterlab-server<3,>=2.19.0
  Downloading https://www.piwheels.org/simple/jupyterlab-server/jupyterlab_server-2.25.2-py3-none-any.whl (58 kB)
    58.9/58.9 kB 2.4 MB/s eta 0:00:00
Collecting notebook-shim>=0.2
  Downloading https://www.piwheels.org/simple/notebook-shim/notebook_shim-0.2.3-py3-none-any.whl (13 kB)
Collecting packaging
  Downloading https://www.piwheels.org/simple/packaging/packaging-23.2-py3-none-any.whl (53 kB)
    53.0/53.0 kB 433.9 kB/s eta 0:00:00
```

View the modules installed by the current Python version:

```
pip list
```

```
(env) pi@raspberrypi:~/Demo_Project $ pip list
Package                                Version
-----
anyio                                  4.2.0
argon2-cffi                           23.1.0
argon2-cffi-bindings                  21.2.0
arrow                                  1.3.0
asttokens                              2.4.1
asvnc-lru                              2.0.4
jupyter_client                         8.6.0
jupyter_core                           5.5.1
jupyter-events                         0.9.0
jupyter-lsp                            2.2.1
jupyter_server                         2.12.1
jupyter_server_terminals               0.5.0
jupyterlab                             4.0.9
jupyterlab_pygments                    0.3.0
jupyterlab_server                      2.25.2
```

Exit the virtual environment

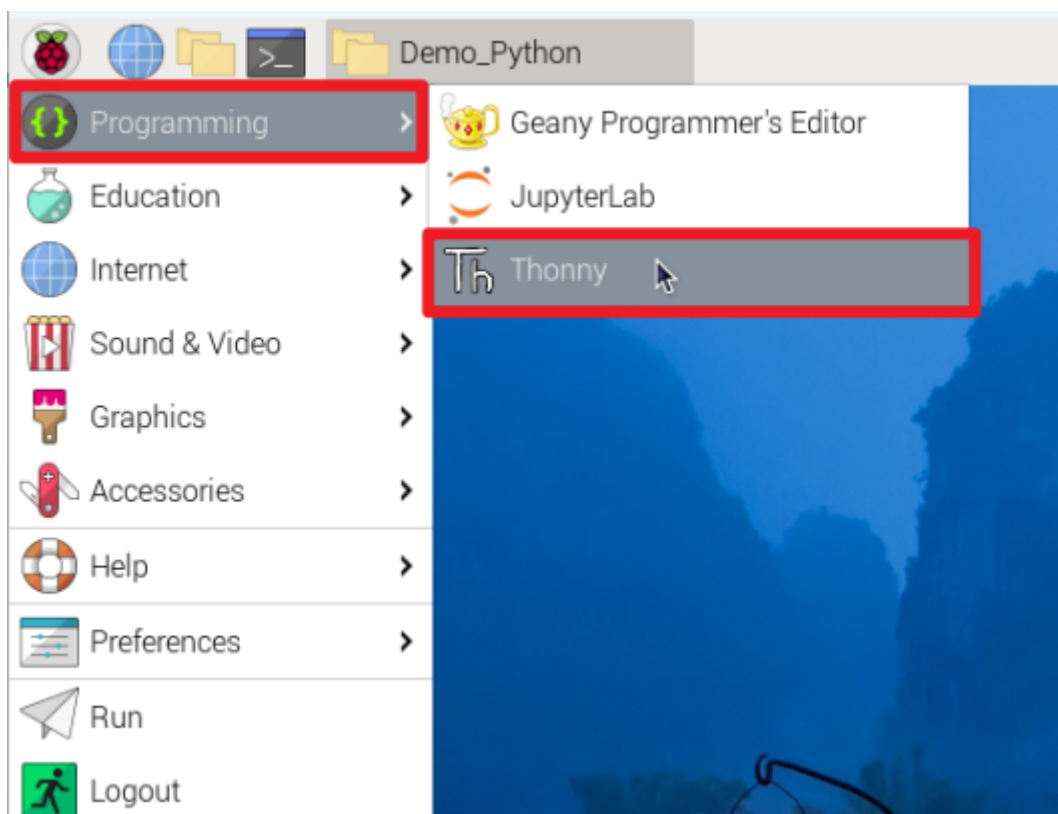
```
deactivate
```

```
(env) pi@raspberrypi:~/Demo_Project $ deactivate
pi@raspberrypi:~/Demo_Project $
```

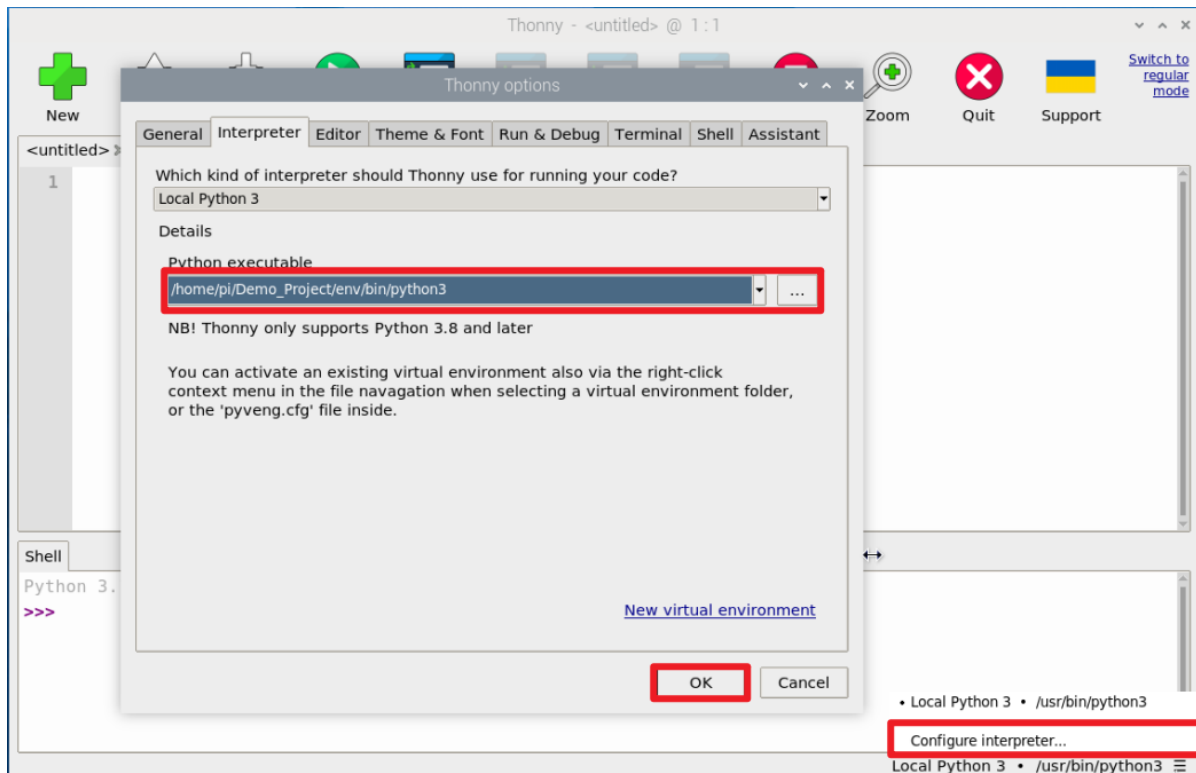
Thonny Editor

When using Python on the Raspberry Pi, Thonny is the system default editor and the interpreter uses the system Python.

Change the virtual environment Python version



Click the interpreter menu in the lower right corner of the Thonny interface to switch to the Python virtual environment: Add according to the virtual environment path



Configuration successful:

