

<https://www.halvorsen.blog>

Installing Python Packages on Raspberry Pi

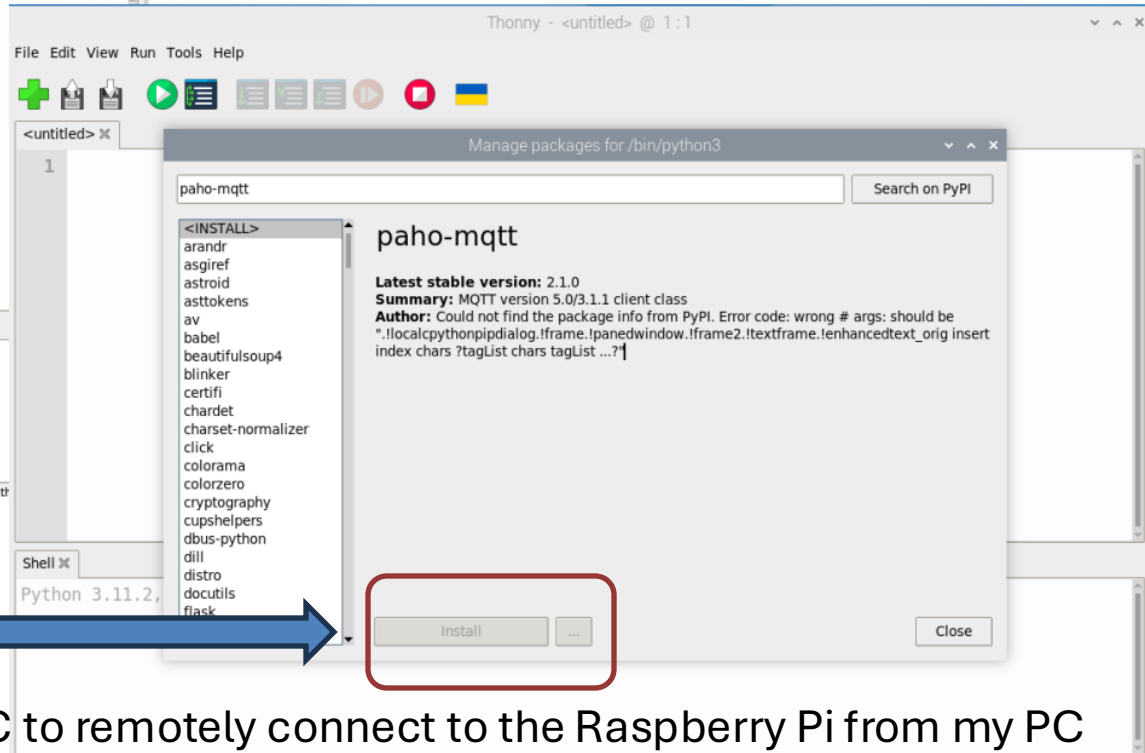
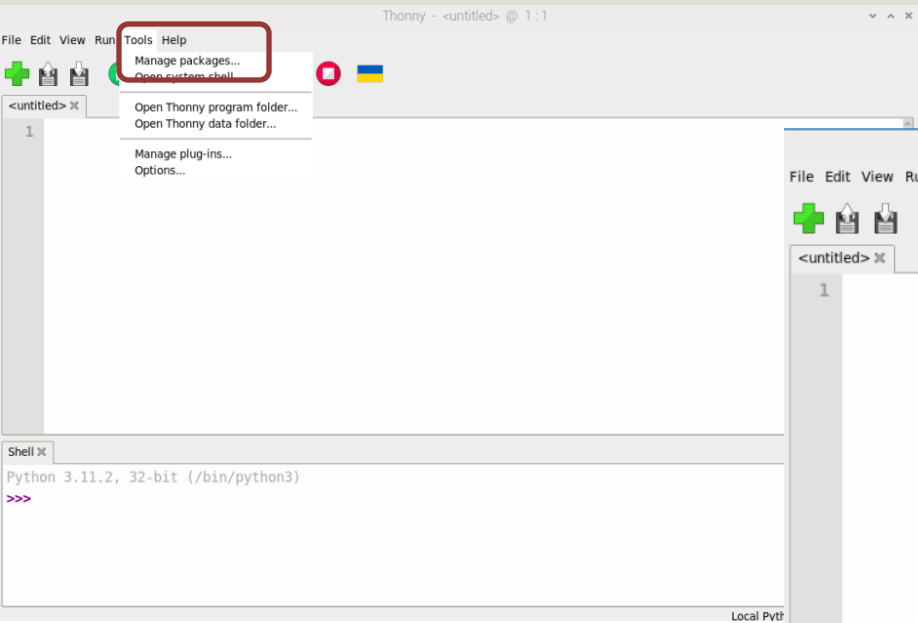
Hans-Petter Halvorsen



Installing Python Packages

- Python is installed by default as a "system-wide" installation as part of the Raspberry Pi OS
- By default, in the latest version of Raspberry Pi OS you cannot install Python packages in the global Python ("system-wide" installation) environment on the Raspberry Pi OS.
- This is due to security reasons and the fact that you can destroy this environment by installing new packages which can cause other programs on Raspberry Pi OS to no longer work.
- You have then several options when installing Python packages on Raspberry Pi which will be presented here.

Install Packages using Thonny



The Install button
is disabled



I am using VNC Viewer from RealVNC to remotely connect to the Raspberry Pi from my PC

Install Packages using Terminal

```
pihph@raspberrypi: ~  
File Edit Tabs Help  
pihph@raspberrypi:~$ pip install paho-mqtt  
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 b  
reaking your Python installation or OS, by passing --break-system-packages.  
hint: See PEP 668 for the detailed specification.  
pihph@raspberrypi:~$
```



You get this error message

Installing Python Packages

- Alt 1. You can run "`sudo apt-get install python3-packagename`" from the Terminal in Raspberry Pi OS to still install packages "system-wide".
- Alt 2. You can create a new virtual Python environment on the Raspberry Pi.
 - This is the recommended alternative as you do not destroy other programs or the system-wide Python environment.
 - This can be done in several ways, but the easiest way is to do it from the graphical environment in Thonny, alternatively from the Terminal.
- Alt 3. You can switch off the default setting which allows you to install Python packages "system-wide" in the usual way using pip/Terminal or from Thonny. Not recommended.

Alt 1 - Terminal

Run “**sudo apt-get install python3-packagename**” in the Terminal

```
pihph@raspberrypi: ~  
File Edit Tabs Help  
error: externally-managed-environment  
  
x 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 b  
reaking your Python installation or OS, by passing --break-system-packages.  
hint: See PEP 668 for the detailed specification.  
pihph@raspberrypi:~ $ sudo apt-get install python3-paho-mqtt  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following packages were automatically installed and are no longer required:  
 chromium-browser chromium-browser-l10n  
Use 'sudo apt autoremove' to remove them.  
The following NEW packages will be installed:  
 python3-paho-mqtt  
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.  
Need to get 56.1 kB of archives.  
After this operation, 302 kB of additional disk space will be used.  
Get:1 http://mirrors.dotsrc.org/raspbian/raspbian bookworm/main armhf python3-paho-mqtt all 1.6.1-1 [56.1 kB]  
Fetched 56.1 kB in 1s (61.1 kB/s)  
Selecting previously unselected package python3-paho-mqtt.  
(Reading database ... 163288 files and directories currently installed.)  
Preparing to unpack .../python3-paho-mqtt_1.6.1-1_all.deb ...  
Unpacking python3-paho-mqtt (1.6.1-1) ...  
Setting up python3-paho-mqtt (1.6.1-1) ...  
pihph@raspberrypi:~ $
```

Alt 2 – Virtual Environment

- A Python Virtual Environment is an isolated space where you can work on your Python projects, separately from your system-installed Python.
- You can set up your own libraries and dependencies without affecting the system Python.
- Why do we need a Virtual Environment?
 - Imagine a scenario where you are working on 2 different projects that need 2 different version of a specific Python package)
 - In such situations, we need to create a virtual environment in Python that can maintain the dependencies of both projects.
- We can, e.g., use the “venv” command in the Terminal to create a virtual environment in Python, or we can use Thonny.

Alt 2 - Thonny

Create a Virtual Python Environment using Thonny

The image shows the Thonny IDE interface. On the left, the 'Run' menu is open, with 'Configure interpreter...' highlighted. A red box is drawn around this menu item. Below the menu, a shell window shows 'Python 3.11.2, 32-bit (/bin/python3)' with a prompt '>>>'. In the center, the text 'Current Python Environment' has a blue arrow pointing to the 'Python executable' field in the 'Thonny options' dialog. The dialog is open to the 'Interpreter' tab, showing 'Local Python 3' selected in the dropdown. Below it, the 'Python executable' field contains '/bin/python3'. A red box is drawn around the 'New virtual environment' link at the bottom right of the dialog. The 'OK' and 'Cancel' buttons are at the bottom.

File Edit View Run Tools Help

Configure interpreter...

Run current script F5
Debug current script (nicer) Ctrl+F5
Debug current script (faster) Shift+F5

Visualize current script at Python Tutor
Debug current script (birdseye) Ctrl+Shift+B

Step over F6
Step into F7
Step out
Resume F8
Run to cursor Ctrl+F8
Step back Ctrl+B

Run current script in terminal Ctrl+T

Dock user windows
Pygame Zero mode

Stop/Restart backend Ctrl+F2
Interrupt execution Ctrl+C
Send EOF / Soft reboot Ctrl+D
Disconnect

Shell x

Python 3.11.2, 32-bit (/bin/python3)
>>>

Thonny options

General Interpreter Editor Theme & Font Run & Debug Terminal Shell Assistant

Which kind of interpreter should Thonny use for running your code?
Local Python 3

Details

Python executable
/bin/python3

NB! Thonny only supports Python 3.8 and later

You can activate an existing virtual environment also via the right-click context menu in the file navigation when selecting a virtual environment folder, or the 'pyveng.cfg' file inside.

[New virtual environment](#)

OK Cancel

Alt 2 - Thonny

The image shows a screenshot of the Thonny IDE interface. In the foreground, the 'Thonny options' dialog box is open, with the 'Interpreter' tab selected. It asks 'Which kind of interpreter should Thonny use for running your code?' and shows 'Local Python 3' as the selected option. The path for the Python executable is '/home/pihph/Documents/Development/bin/python3'. Below this, there is a note: 'NB! Thonny only supports Python 3.8 and later' and instructions on how to activate a virtual environment. A 'New virtual' link is visible at the bottom right of the dialog. In the background, another instance of the 'Manage packages for /home/pihph/Documents/Development/bin/python3' dialog is open, showing the search results for 'pymongo'. The search results include the package name, version (4.10.1), summary, author, and license. At the bottom of this dialog, there is an 'Install' button and an ellipsis button, both of which are highlighted with a red box. A blue arrow points from the text 'Now you can install Python Packages' to the 'Install' button. The main IDE window shows a code editor with a single line of code and a shell window with the prompt '>>>'.

Thonny - <untitled> @ 1:1

File Edit View Run Tools Help

Thonny options

General Interpreter Editor Theme & Font Run & Debug Terminal Shell Assistant

Which kind of interpreter should Thonny use for running your code?

Local Python 3

Details

Python executable

`/home/pihph/Documents/Development/bin/python3`

NB! Thonny only supports Python 3.8 and later

You can activate an existing virtual environment also via the right-click context menu in the file navigation when selecting a virtual environment folder or the 'pyveng.cfg' file inside.

New virtual

OK

Shell <x>

Python 3.11.2, 32-bit

>>>

Manage packages for /home/pihph/Documents/Development/bin/python3

pymongo Search on PyPI

<INSTALL>

pip
setuptools

Latest stable version: 4.10.1
Summary: Python driver for MongoDB <<http://www.mongodb.org>>
Author: The MongoDB Python Team
License: Apache License Version 2.0, January 2004 <http://www.apache.org/licenses/TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION> 1. Definitions. "License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document. "Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License. "Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity. "You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License. "Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files. "Object" form shall mean any form resulting from mechanical transformation or translation of a source form, including but not limited to compiled object code, generated documentation, and conversions to other media types. "Work"

Install ... Close

Local Python 3 • /home/pihph/Documents/Development/bin/python3

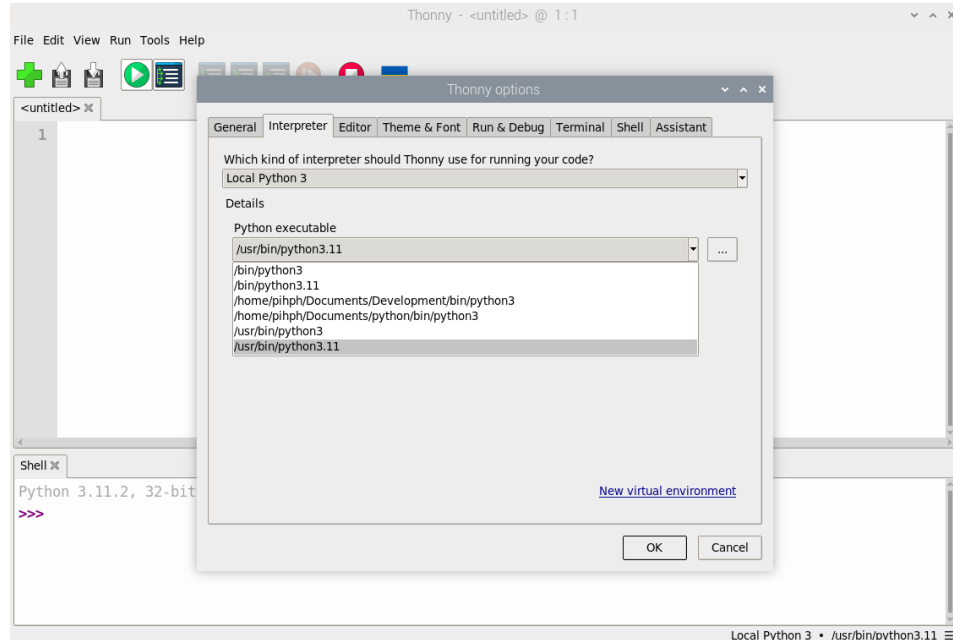
Now you can install Python Packages

Alt 2 - Terminal

Run the following in the Terminal:

- Step 1: Create Virtual Environment:
`python -m venv yourvenvname`
- Step 2: Activate the Virtual Environment:
`source yourvenvname/bin/activate`
- Step 3: Install Python Packages:
`pip install package`

It will also be available
from Thonny afterwards:



Hans-Petter Halvorsen

University of South-Eastern Norway

www.usn.no

E-mail: hans.p.halvorsen@usn.no

Web: <https://www.halvorsen.blog>

