

VSCODE Tour

* VSCode is an IDE (Integrated Development Environment) that is

free – no cost to download, free for private or commercial use

open source – source code available on github

multi language – lots including Python, Java, C variants, Arduino, PHP, Go, Lua,

multi platform – Windows, Mac, Ubuntu, Debian, Red Hat variants

fairly lightweight

tons of extensions and add-ons available

learning curve is not as steep as some other full-featured IDEs

* What the tour covers:

Installation

Setup and Configuration

Extensions

Features to help you code

Debugging (Breakpoints, Single Stepping, Viewing Variables, etc)

* Start with Raspberry Pi OS (I used August 2020 version)

* Git is installed by default

* Python 3 is installed by default

Install VSCode (code-oss)

* Code-OSS is available for 32 bit ARM processors

* Instructions available at code.headmelted.com

```
wget -O - https://code.headmelted.com/installers/apt.sh
```

* The .deb package can also be downloaded directly, choose armhf.deb for 32 bit version to work with Raspberry Pi

Start/Setup Code-OSS

* Code-OSS start icon available under Raspberry Pi icon (top left) under Programming

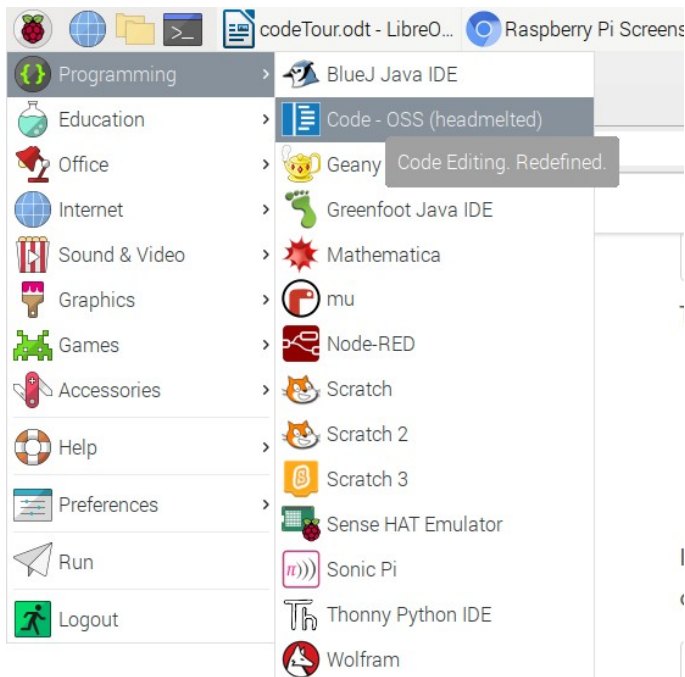


Figure 1: VSCode Start Menu icon

VSCode Starts

Some HouseKeeping

- * I personally don't like the Dark theme, so I change it
- * Select File → Preferences → Settings → Workbench → Appearance

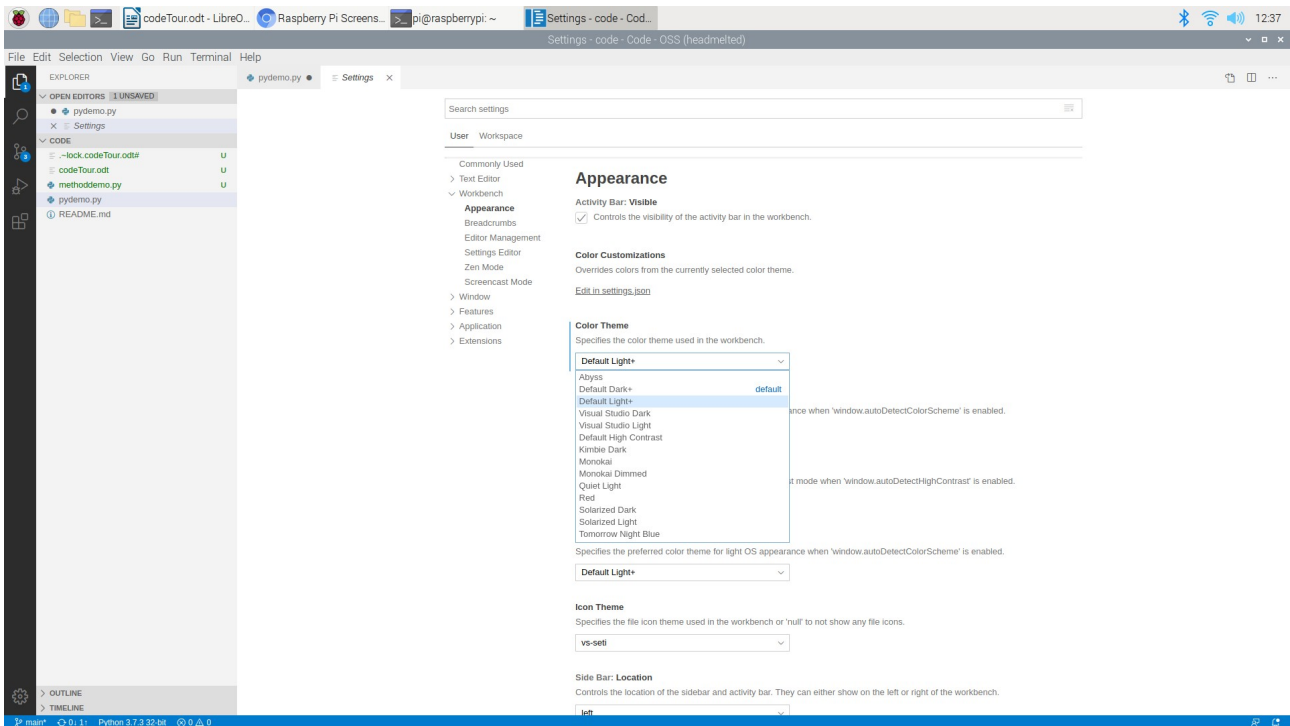


Figure 2: Configuration Settings

- * There are tons of other settings here (font, screen size, tab spacing, etc, etc)
- * I mentioned that Git is supported out of the box, but Python needs to be added
- * I'll start with the Python extension from Microsoft, which has everything I need for now.

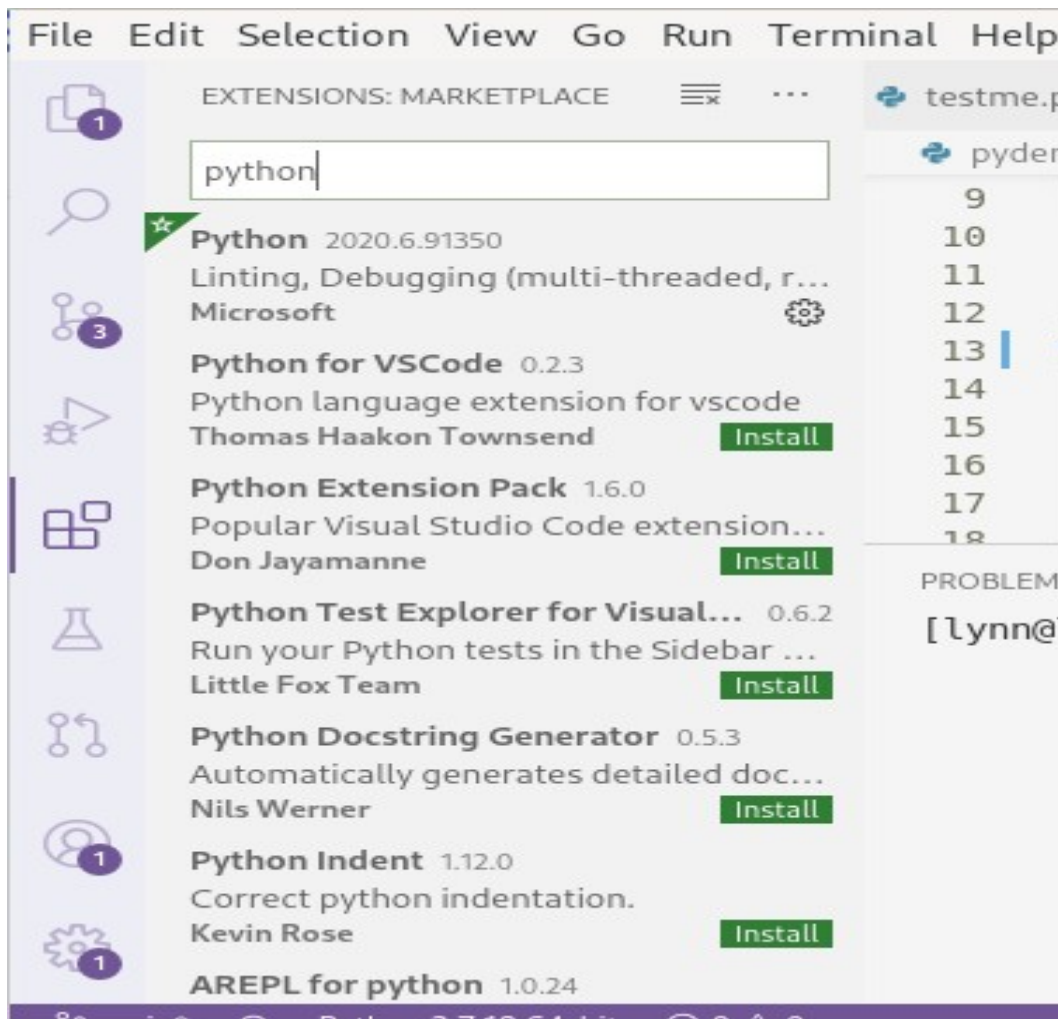


Figure 3: VSCode Extensions Installation

* The extensions icon is the one with four blocks. Click on it, and search for what you want. In this case, I search for Python. The one I want comes up first, I click “Install”.

* note that language and language features need to be installed in OS first

Start a Project, Start Coding

* close VSCode

* start a command prompt

* create directory you want your project in

* still on the command line, cd to project directory

```
code-oss .
```

That is, code dash oss space dot for Raspberry Pi OS

* VS Code starts

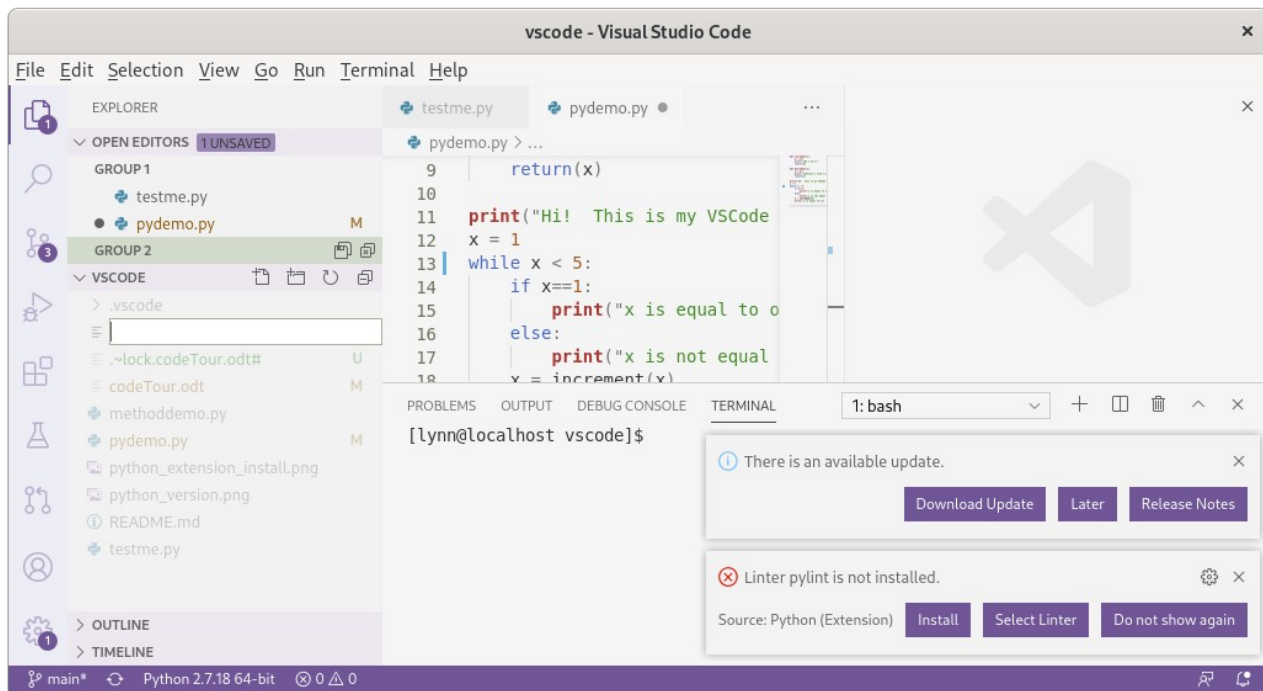


Figure 4: Add a File to a Workspace

Start a New File

* file icon is top left

* file plus icon starts a new file

* give it a name, enter

* start typing

Snippets, etc

* Python extension adds syntax highlighting

* snippets give syntax reminders/examples

if/else

try/catch

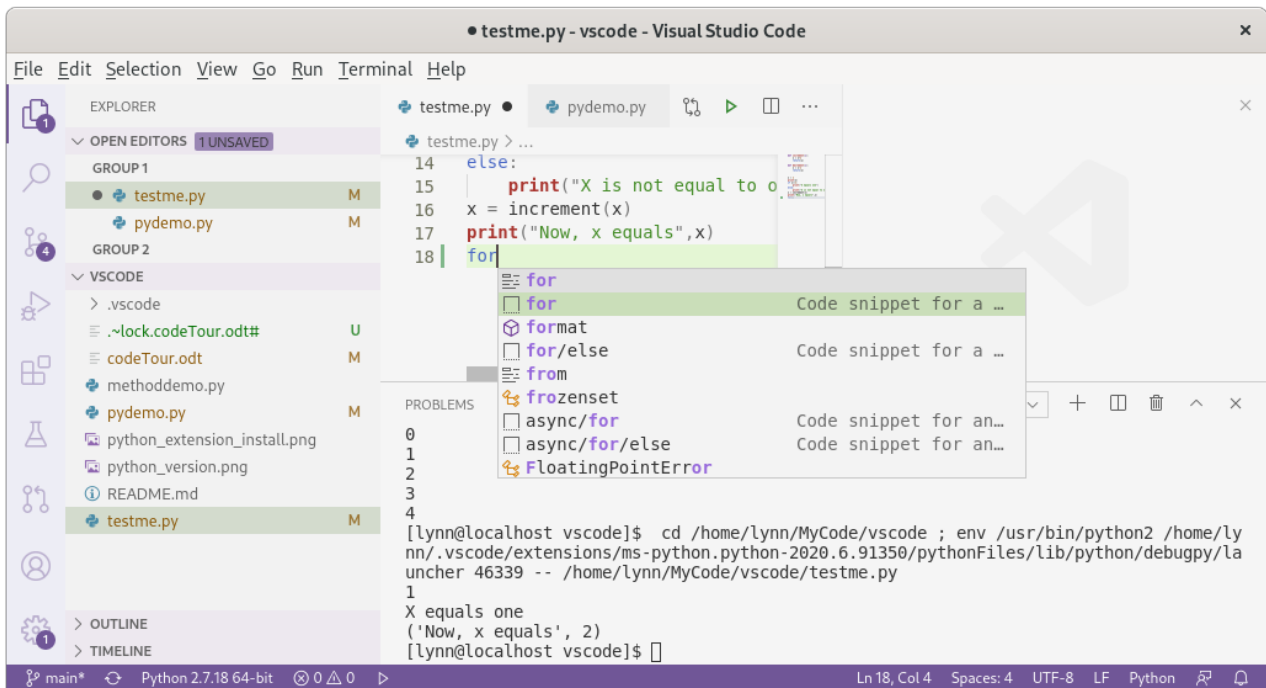


Figure 5: Example of For Loop Snippet

* Syntax highlighting is done by colour-coding

* Auto indent spacing is set up in Preferences

Debug Stepping

<https://stackoverflow.com/questions/52368009/what-is-the-difference-between-step-in-step-out-and-step-over>

Step in: means that if there is a function call, it goes inside the function and you can see how the function is executing line by line till it returns and you go back to the next line right after the function call.

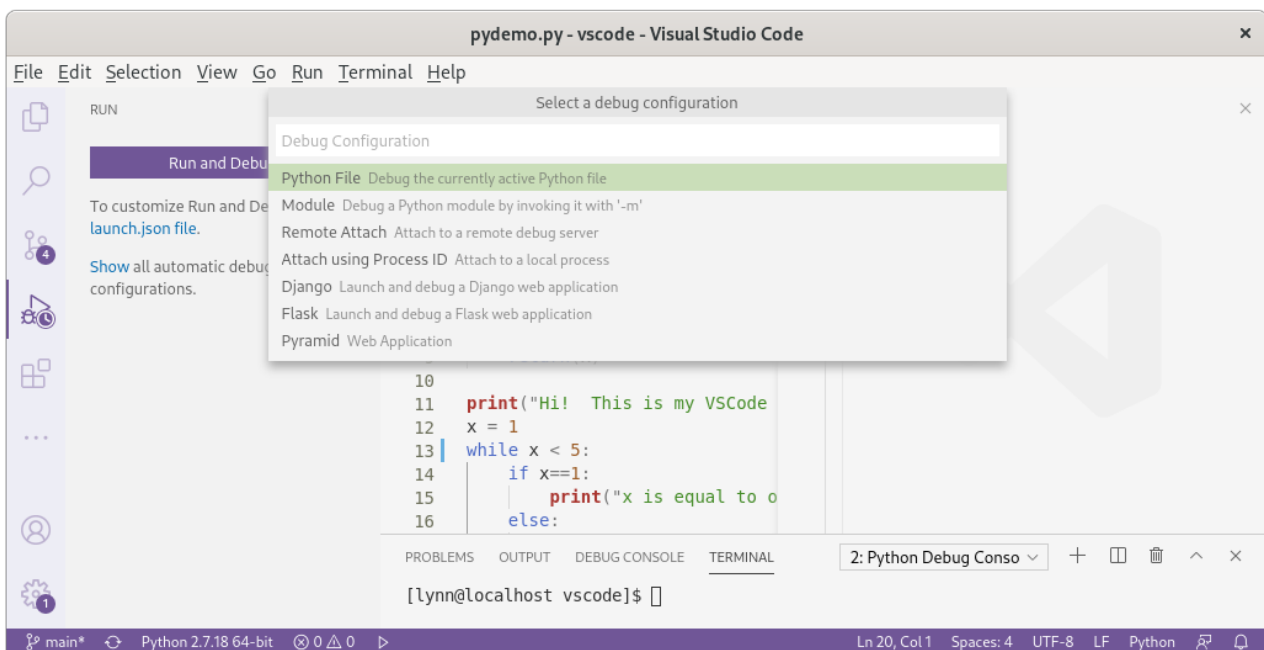
Step over: means that if there is a function call, it just executes it like a black box and returns the result, but you cannot see how the function was executed.

Step out: means that if you have **Stepped in** a function and now you want to skip seeing how the rest of the function is going to execute, you **Step out** and the function returns. Then, you go back to the next line, that is the line right after the function call.

* To set a breakpoint (or many), click to the left of the line number. A red dot appears where the breakpoint is set

* After breakpoint(s) set, set up debugging

View → Command Pallet → Python Debug Current File in Python Interactive Window

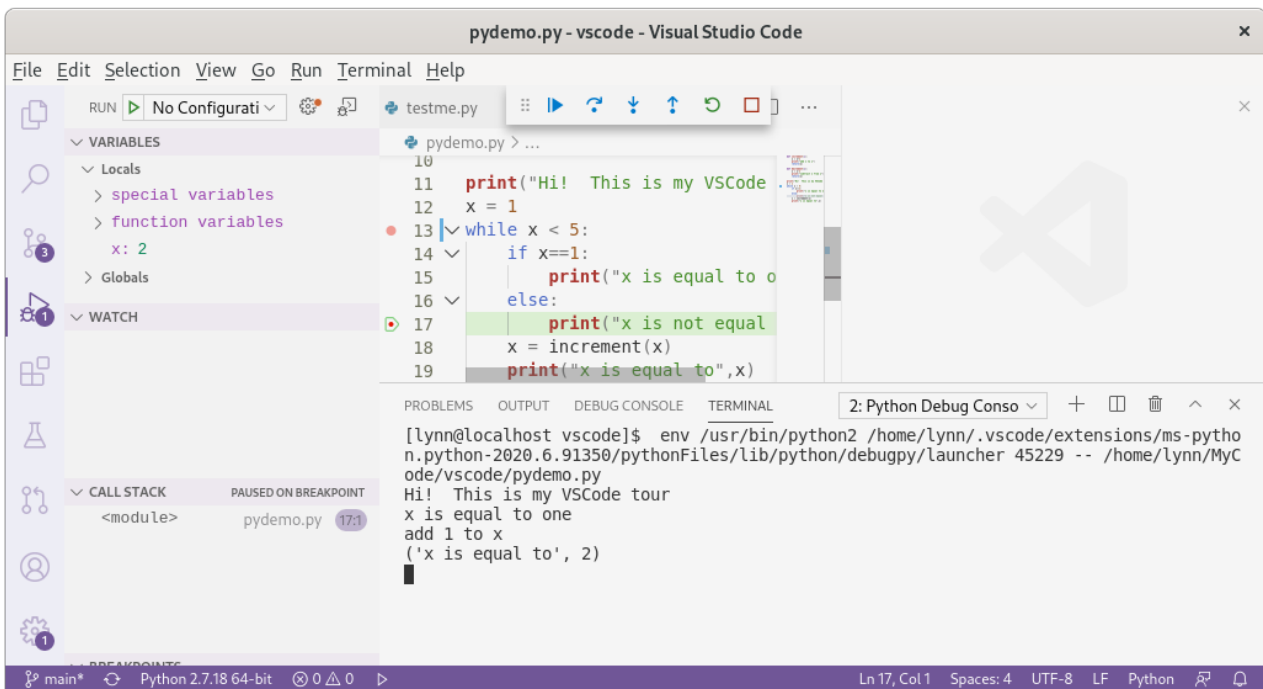


* Click on “Run & Debug” icon (left side)

* Click on “Run and Debug” in run window upper left

* In debug config, select “Python File”. Switches to debug mode.

Debug ToolBar



- * block of dots (left) moves pallet around
- * Continue (run the program until a breakpoint is encountered)
- * Step over
- * Step into
- * Step out
- * Restart
- * Stop

Debug Pane

* **VARIABLES**– top part of pane

* Variables change as code is run

* Arrow on line number and line is highlighted

* **WATCH** – keep an eye on a particular variable or item

* **CALL STACK**

in our example, <module> is the code we're debugging. Notice it shows the line number we're on. When branching to a function, the line number is where we will return to

when branching to a function, the function name will appear with the line number currently being executed

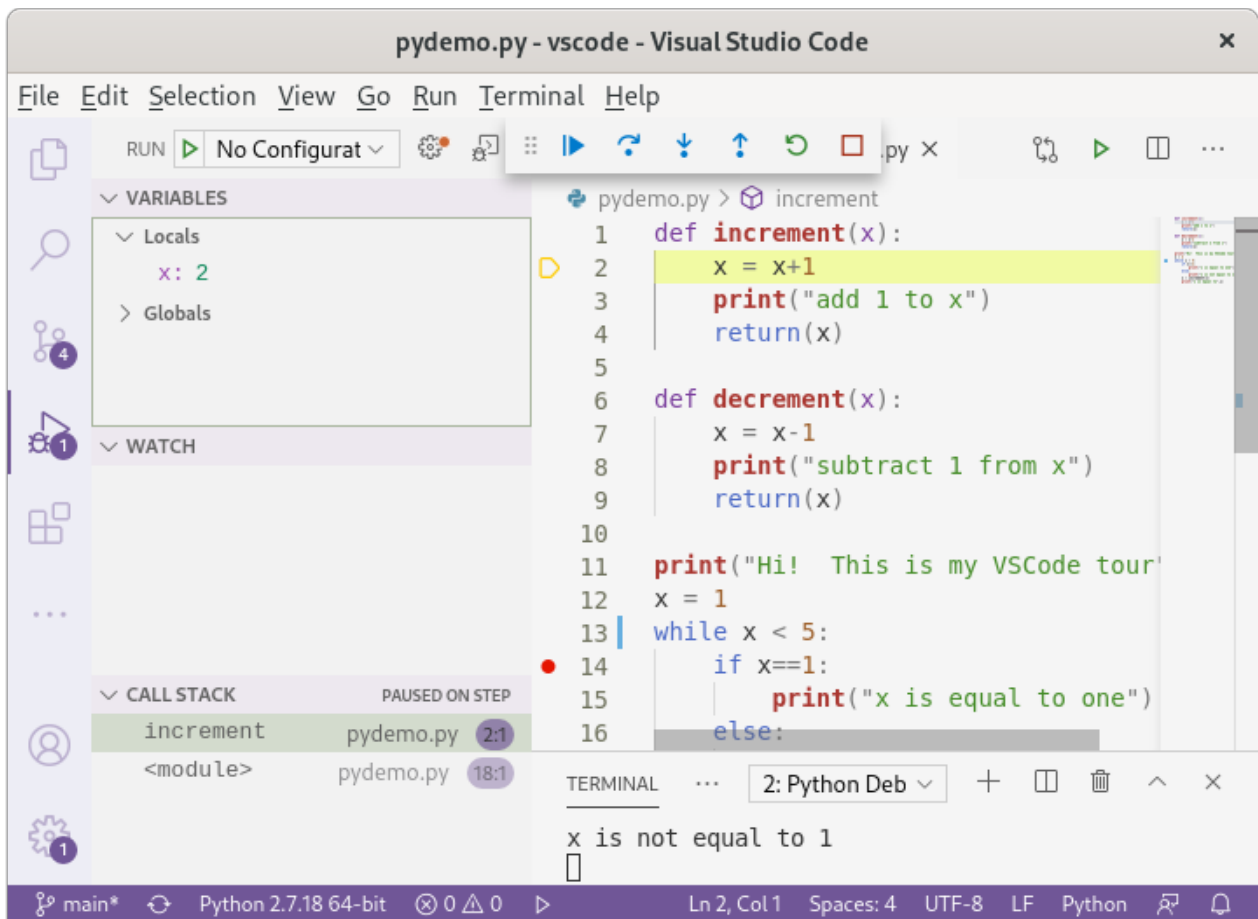


Figure 6: call stack once program branches to function call

To Recap

- * VSCode is a free, open source IDE that works for many different hardware/software combinations. Learn once, use everywhere :)
- * We covered installation and setup of VSCode on a Raspberry Pi.
- * We saw how extensions were added
- * VSCode has syntax highlighting and snippets
- * It's easy to debug by setting breakpoints and single-stepping while watching variable values change.