



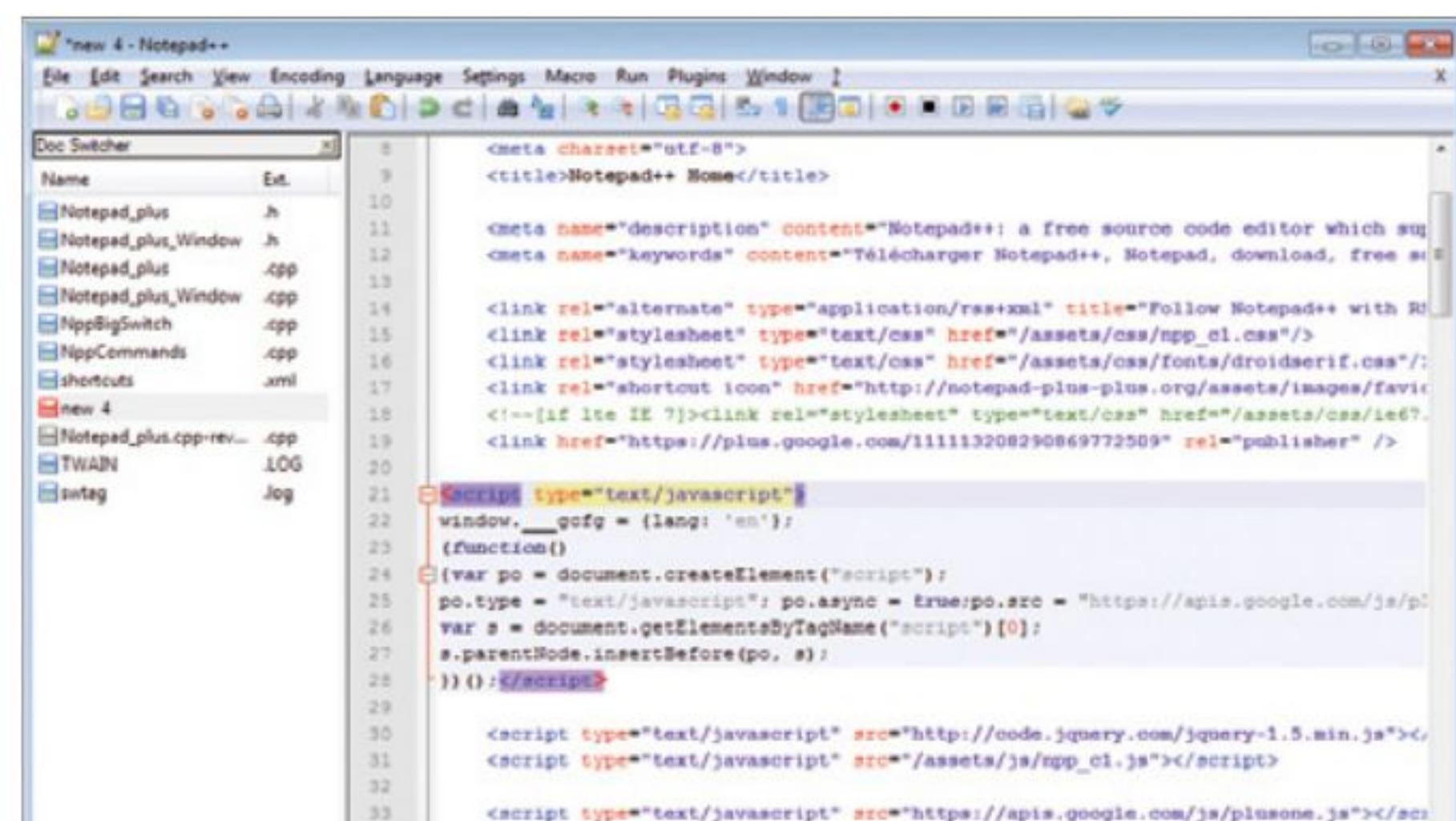
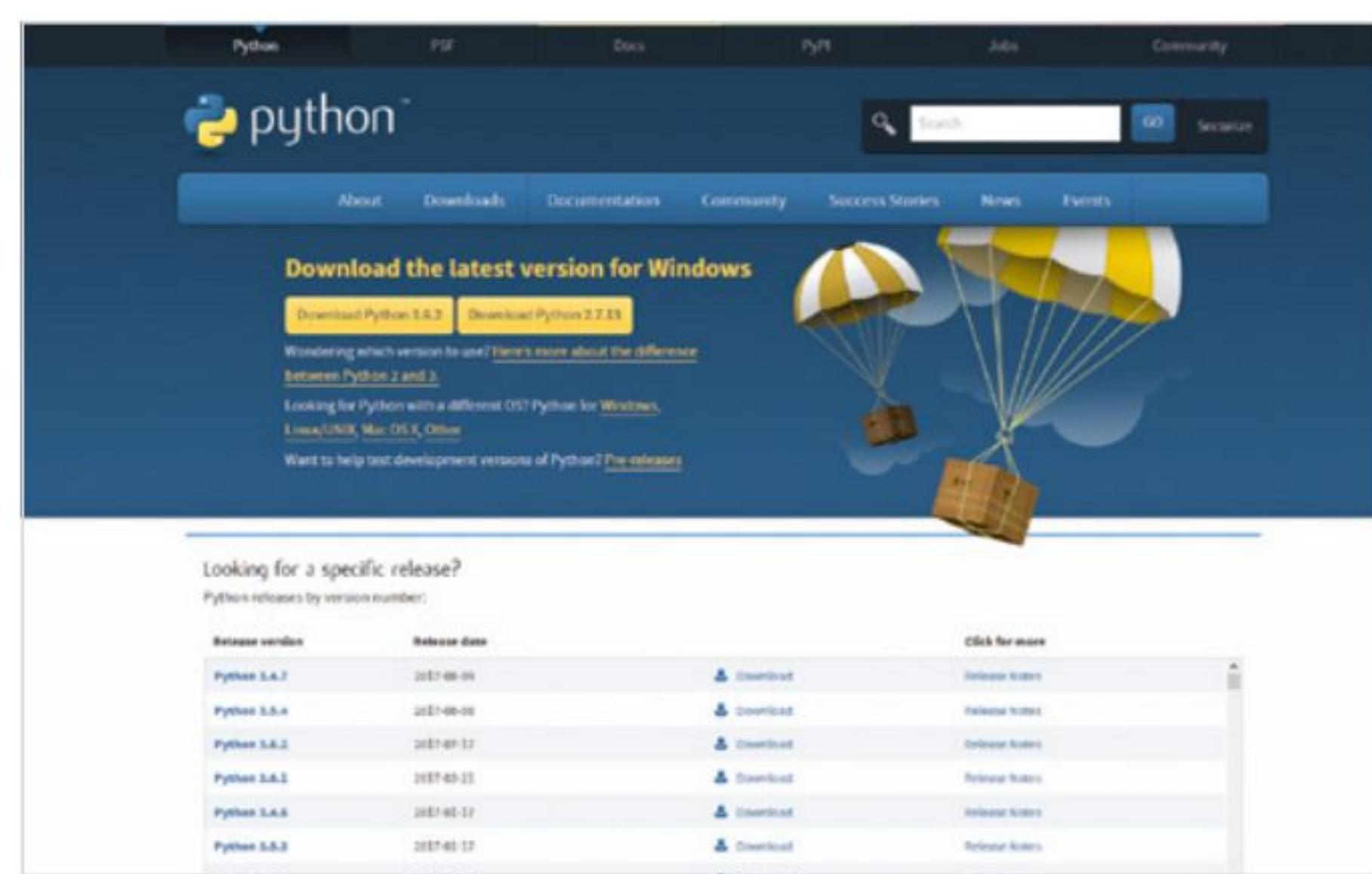
Equipment You Will Need



You can learn Python with very little hardware or initial financial investment. You don't need an incredibly powerful computer and any software that's required is freely available.

WHAT WE'RE USING

Thankfully, Python is a multi-platform programming language available for Windows, macOS, Linux, Raspberry Pi and more. If you have one of those systems, then you can easily start using Python.



COMPUTER

Obviously you're going to need a computer in order to learn how to program in Python and to test your code. You can use Windows (from XP onward) on either a 32 or 64-bit processor, an Apple Mac or Linux installed PC.

AN IDE

An IDE (Integrated Developer Environment) is used to enter and execute Python code. It enables you to inspect your program code and the values within the code, as well as offering advanced features. There are many different IDEs available, so find the one that works for you and gives the best results.

PYTHON SOFTWARE

macOS and Linux already come with Python preinstalled as part of the operating system, as does the Raspberry Pi. However, you need to ensure that you're running the latest version of Python. Windows users need to download and install Python, which we'll cover shortly.

TEXT EDITOR

Whilst a text editor is an ideal environment to enter code into, it's not an absolute necessity. You can enter and execute code directly from the IDLE but a text editor, such as Sublime Text or Notepad++, offers more advanced features and colour coding when entering code.

INTERNET ACCESS

Python is an ever evolving environment and as such new versions often introduce new concepts or change existing commands and code structure to make it a more efficient language. Having access to the Internet will keep you up-to-date, help you out when you get stuck and give access to Python's immense number of modules.

TIME AND PATIENCE

Despite what other books may lead you to believe, you won't become a programmer in 24-hours. Learning to code in Python takes time, and patience. You may become stuck at times and other times the code will flow like water. Understand you're learning something entirely new, and you will get there.

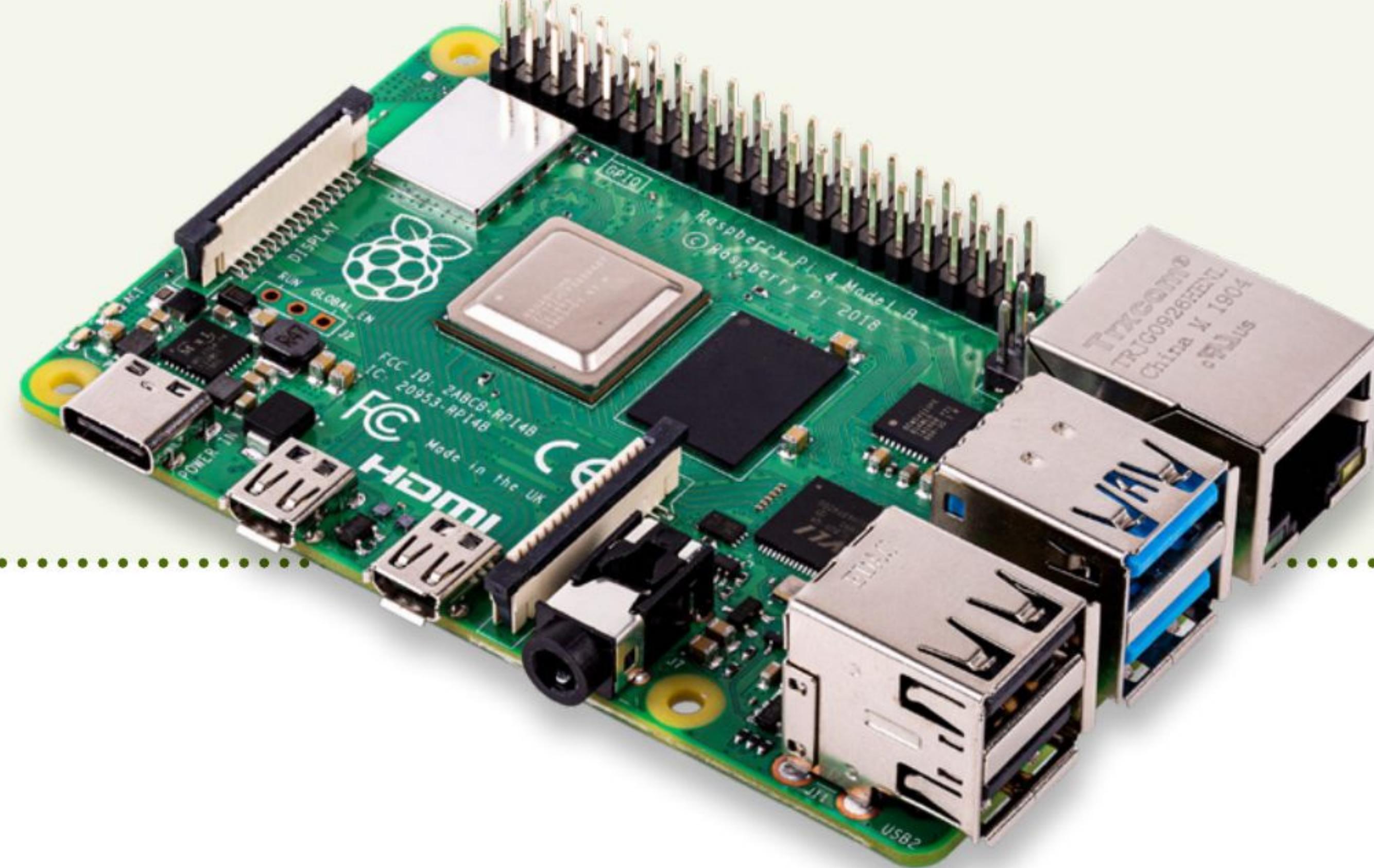


THE RASPBERRY PI

Why use a Raspberry Pi? The Raspberry Pi is a tiny computer that's very cheap to purchase, but offers the user a fantastic learning platform. Its main operating system, Raspbian, comes preinstalled with the latest Python along with many modules and extras.

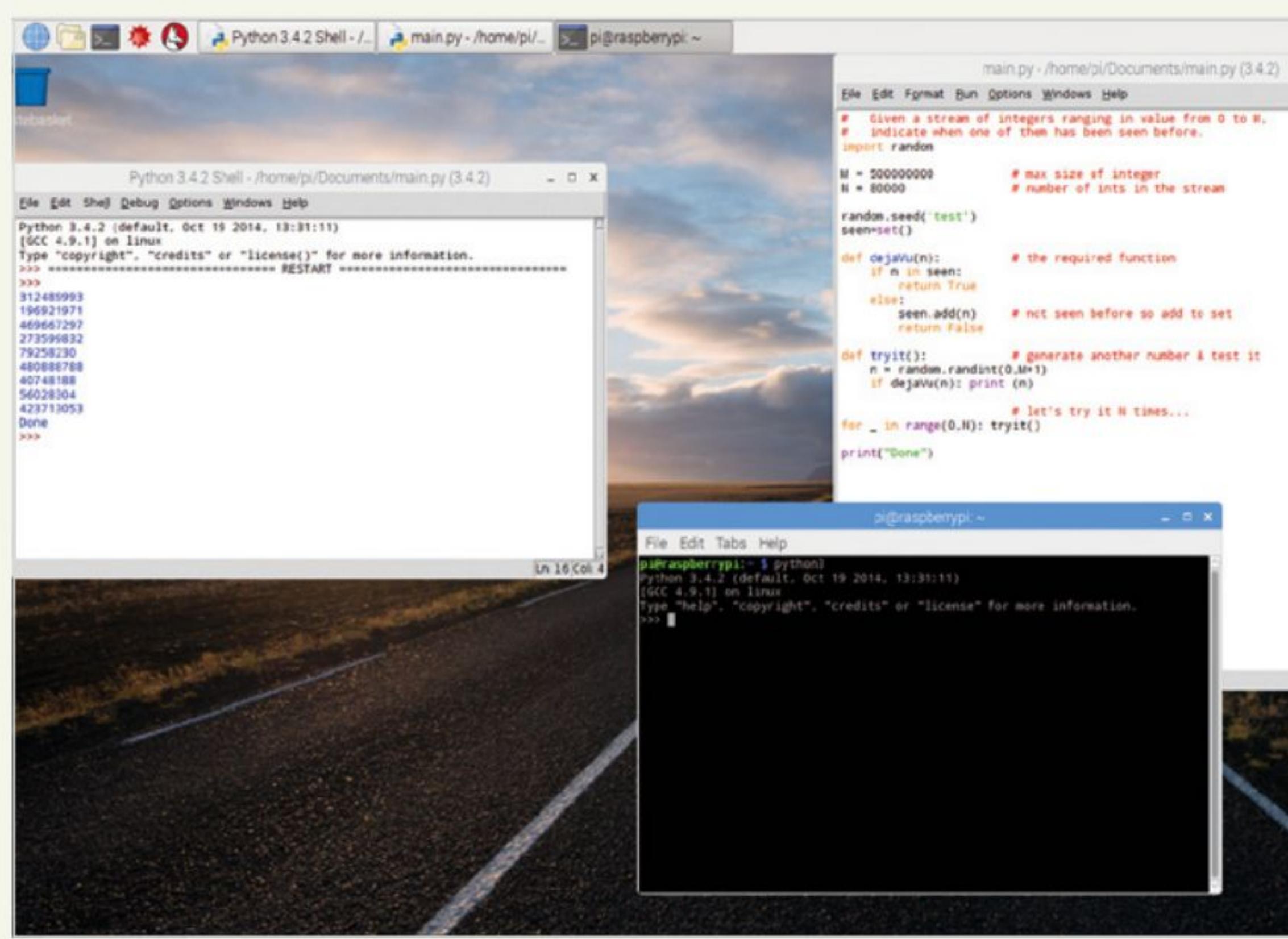
RASPBERRY PI

The Raspberry Pi 4 Model B is the latest version, incorporating a more powerful CPU, a choice of 1GB, 2GB or 4GB memory versions and Wi-Fi and Bluetooth support. You can pick up a Pi from around £33, increasing up to £54 for the 4GB memory version, or as a part of kit for £50+, depending on the kit you're interested in.



RASPBIAN

The Raspberry Pi's main operating system is a Debian-based Linux distribution that comes with everything you need in a simple to use package. It's streamlined for the Pi and is an ideal platform for hardware and software projects, Python programming and even as a desktop computer.



FUZE PROJECT

The FUZE is a learning environment built on the latest model of the Raspberry Pi. You can purchase the workstations that come with an electronics kit and even a robot arm for you to build and program. You can find more information on the FUZE at www.fuze.co.uk.

BOOKS

We have several great Raspberry Pi titles available via www.bdmpublications.com. Our Pi books cover how to buy your first Raspberry Pi, set it up and use it; there are some great step-by-step project examples and guides to get the most from the Raspberry Pi too.





Getting to Know Python

Python is one of the most used and popular programming languages. It's a great language on which to learn how to code, but it's also powerful enough to be used by companies when trawling through petabytes of raw data.

PYTHON POWER

Python is a high-level, general-purpose programming language that was designed by Guido van Rossum in the late '80s, as a successor to the ABC Programming Language, and became available to use in 1990.



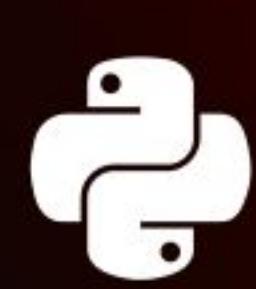
The Python Software Foundation, a non-profit organisation devoted to both furthering and improving Python, currently develops the language. The foundation's purpose is to "promote, protect, and advance the Python programming language, and to support and facilitate the growth of a diverse and international community of Python programmers."

The structure of Python code has been designed to flow easily, allowing those who are just beginning to code to follow it without too much difficulty. Yet, despite its ease of use, Python is regularly used throughout a number of industries in real-world scenarios.

Big Data and AI are the two fastest growing Python-backed technologies at the moment. Big Data is simply a modern term

used to describe huge amounts of data, such as sequences of numbers collated during a day's worth of trading on the Stock Exchange. Python code is used to dig into the voluminous collection of numbers, and then feed back with reports on the highs and lows, what's trending, and so on.

```
170 global_scene_setting = None
171
172 # name="Scene"
173 # attr=0x01, flags=0x00000000
174
175
176 def execute(self, context):
177     # get the folder
178     folder_path = os.path.dirname(self.filepath)
179
180     # get objects selected in the viewport
181     viewport_selection = bpy.context.selected_objects
182
183     # get export objects
184     obj_export_list = viewport_selection
185     if self.use_selection_setting == False:
186         obj_export_list = [i for i in bpy.context.scene.objects]
187
188     # deselect all objects
189     bpy.ops.object.select_all(action='DESELECT')
190
191     for item in obj_export_list:
192         item.select = True
193         if item.type == 'MESH':
194             file_path = os.path.join(folder_path, "{}.obj".format(item.name))
195             file_path = file_path.replace("\\", "/")
196             bpy.ops.export_scene.obj(filepath=file_path, use_selection=True,
197                                     axis_forward='Y',
198                                     axis_up='Z',
199                                     use_smooth_groups=True,
200                                     use_smooth_groups_bitflags=True,
201                                     use_maintain_all_welds=True,
202                                     use_maintain_order=True,
203                                     use_normals=True,
204                                     use_uvs=True,
205                                     use_materials=True,
206                                     use_triangles=True,
207                                     use_subdivision=False,
208                                     use_cyclic_loops=False,
209                                     use_welded=True,
210                                     use_infraUVs=False,
211                                     use_zremap=False,
212                                     use_vgroup_bitmask=False,
213                                     use_vcolor_bitmask=False,
214                                     use_tangent_bitmask=False)
```



Made up from over 5 Petabytes of data, spread across a ton of hard drives, Python helped science to unveil the first image of a black hole.

As a side note, it's not just the likes of the Stock Exchange that use Python to study large quantities of data; in April 2019 the first image of a black hole was released, the supermassive black hole in the galaxy called M87, located roughly 55 million light years away. Thanks to the collaboration of over 200 scientists, using an array of telescopes spanning the world (called the Event Horizon Telescope Project), the combined power of the telescopes formed an impressive five petabytes of data, spread across tens of hard drives weighing in at nearly one ton. Five petabytes, by the way, equates to around 5,000 years' worth of MP3 files. Once all those hard drives were gathered together and shipped to a central supercomputer cluster, the team then used Python to painstakingly stitch together all the fragments of data from the five petabytes to finally form the most talked about astronomic event of the decade.

AI, if you're not familiar with the term, stands for Artificial Intelligence. Although we're still a long way off from the visionary stories of Arthur C. Clarke, AI is fast becoming one of the most influential technologies of our modern age. Rather than controlling robots, the AI that Python drives is designed to create autonomous ways of interacting with people online. For example, when you search for something on the Amazon website you will usually notice that similar products start to appear, whether within Amazon itself,

PYTHON 3 VS PYTHON 2

In a typical computing scenario, Python is complicated somewhat by the existence of two active versions of the language: Python 2 and Python 3.



 Python 3 is the best option to download, or update to.

Python 3 is the newest release of the programming language. However, if you dig a little deeper into the Python site and investigate Python code online, you will undoubtedly come across Python 2. Crucially, although you can run Python 3 and Python 2 alongside each other, it's not recommended. Always opt for the latest stable release, as posted by the Python website. You will find, when using macOS or Linux, that Python 2 is already installed. This is because both these operating systems utilise elements of code necessary to the core functionality of the OS. Linux users tend to be better off, as most distributions package the latest version of Python 3 out-of-the-box, whereas macOS often has Python 3, it's usually an older version. Microsoft doesn't use any Python code for its core Windows systems, which is why you won't find Python inherent to Windows and therefore need to install it from scratch.

or from a search engine, or even Facebook. The code behind these targeted snippets is Python, and it's using a form of AI to help determine what it is you would likely search for.

Despite the fact that many people find the targeting of advertising intrusive, or even an invasion of privacy, you have to admit that the code technology behind it all is rather impressive. With some very clever techniques, a Python developer is able to create a machine thinking script that not only displays items, news stories, books, other websites and ideas relating to what you've searched for, but it can also predict what you may be interested in looking for in the future. Another element to consider, with regards to AI, is that Python code can be used to help a computer learn how to do something more efficiently. In the case of neural networks in AI, the Python code is designed to return a result, then, as the code is run over and over again, the AI portion will begin to learn how to obtain a more accurate result, or do the maths behind the code quicker. It all depends on what the developer wants from their AI Python code.

 An example of Python AI code, using a feedforward neural network.

```
import random
import numpy as np
class Network(object):
    def __init__(self, sizes):
        self.num_layers = len(sizes)
        self.sizes = sizes
        self.biases = [np.random.randn(y, 1) for y in sizes[1:]]
        self.weights = [np.random.randn(y, x)
                       for x, y in zip(sizes[:-1], sizes[1:])]
    def feedforward(self, a):
        for b, w in zip(self.biases, self.weights):
            a = sigmoid(np.dot(w, a)+b)
        return a
    def SGD(self, training_data, epochs, mini_batch_size, eta,
           test_data=None):
        if test_data: n_test = len(test_data)
        n = len(training_data)
```

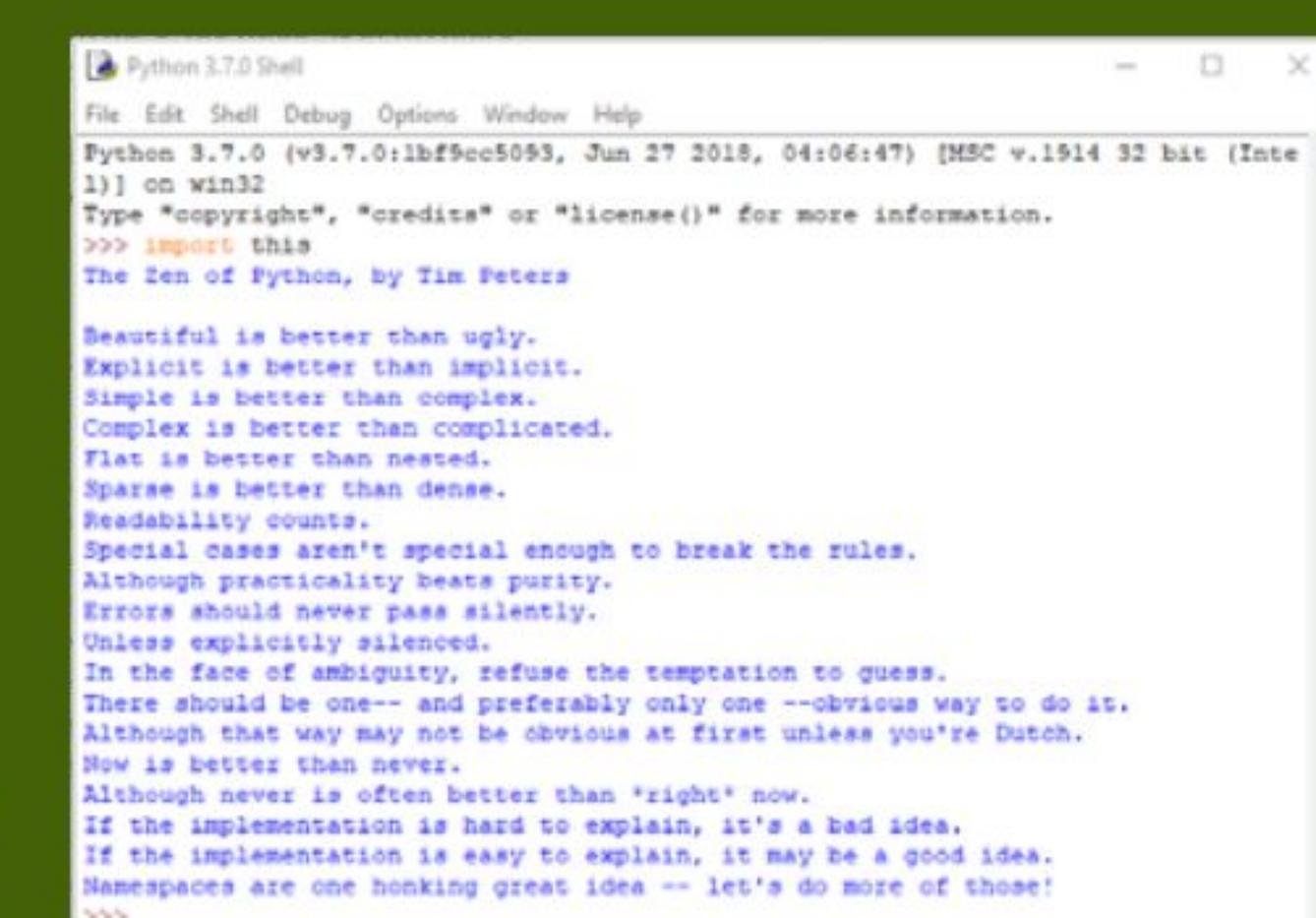
You need to be careful when you look up Python code online, although there are countless websites that offer quick tutorials, code snippets and support – and 99% of these sites are a great help to those starting out with Python – a lot of the sites haven't been updated for some time, and as such use Python 2. If you enter Python 2 code into the Python 3 IDLE, the chances are it won't work due to incompatibilities between the older version and the newer. Python 2 is good, but Python 3 is better. You can obviously spend time converting the Python 2 code into version 3, but, to begin with, it's best to make sure that the code you're looking at is for the Python 3 libraries. Don't worry, though, all the code in this book is designed for Python 3, and that includes all the sub versions from Python 3.1 to the latest 3.x.

Python 3's growing popularity has meant that it's now prudent to start learning to develop with the new features and begin to phase out the previous version. Many development companies, such as SpaceX and NASA, use Python 3 for snippets of important code.

However, support for Python 2 is set to end on January 1st 2020, but this doesn't mean it'll be the last you see of it. Many Linux distros use Python 2 libraries, as does macOS, and to be fair, for the developers to transfer the existing Python 2 content to Python 3 may take more time than they have available, i.e. before the start of 2020. It's likely then, that we will still be seeing Python 2 long after it has had the final nail hammered into its coffin – in fact, expect to see that cut-off date extend further into the future.

ZEN OF PYTHON

Python lets you access all the power of a computer in a language that humans can understand. Behind all this is an ethos called "The Zen of Python". This is a collection of 20 software principles that influences the design of the language. Principles include "Beautiful is better than ugly" and "Simple is better than complex." Type `import this`



 The Zen of Python, as seen when entering: `import this`, into the Python IDLE.

As you will discover over the coming pages, Python is a fantastic language to learn. Get to grips with the basics, and before long, you'll be creating your own Python code for games, tools, and maybe even something in AI. The only limit with Python is your own imagination.



How to Set Up Python in Windows

Windows users can easily install the latest version of Python via the main Python Downloads page. Whilst most seasoned Python developers may shun Windows as the platform of choice for building their code, it's still an ideal starting point for beginners.

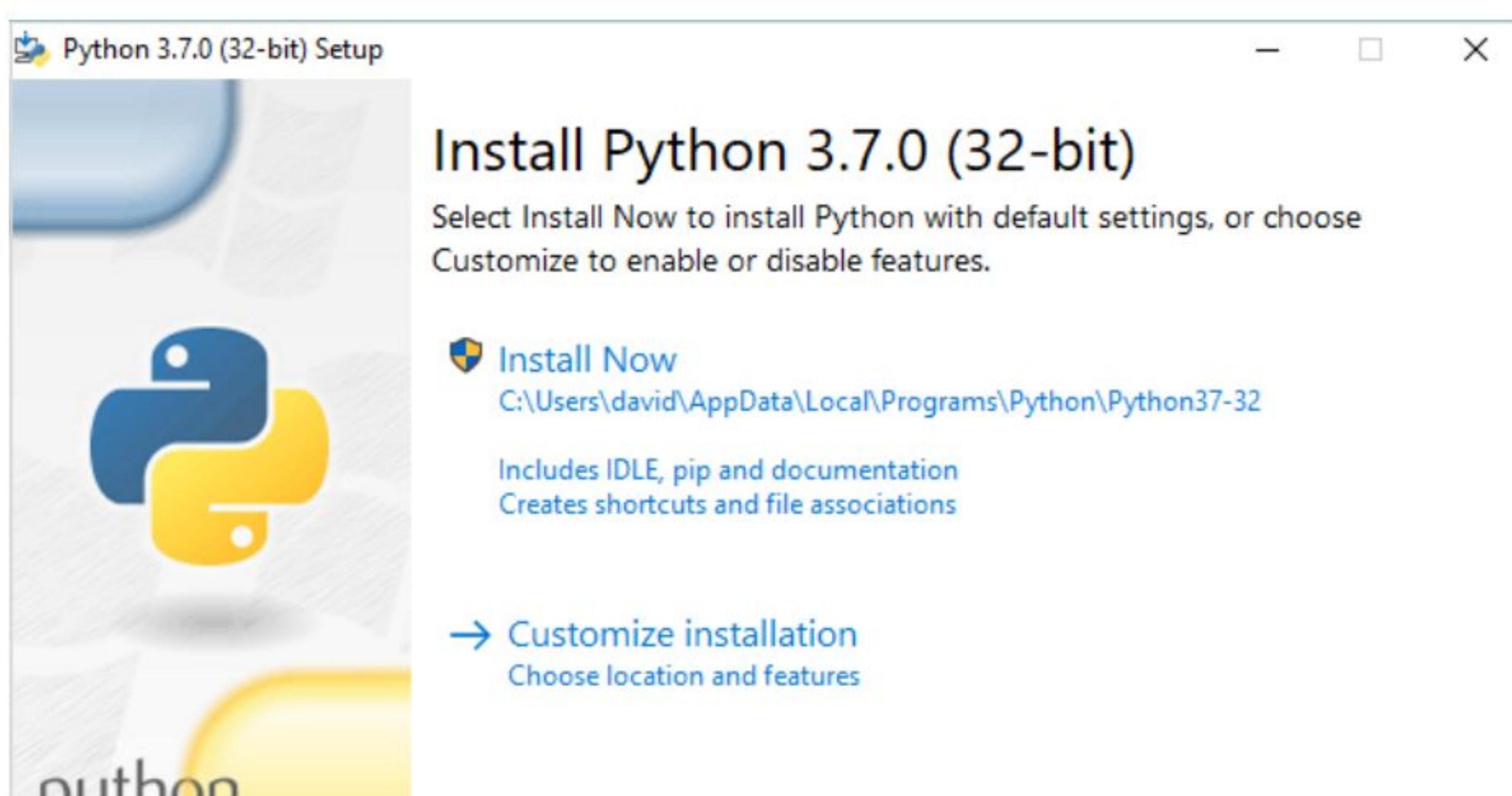
INSTALLING PYTHON 3.X

Microsoft Windows doesn't come with Python preinstalled as standard, so it will be necessary to install it yourself manually. Thankfully, it's an easy process to follow.

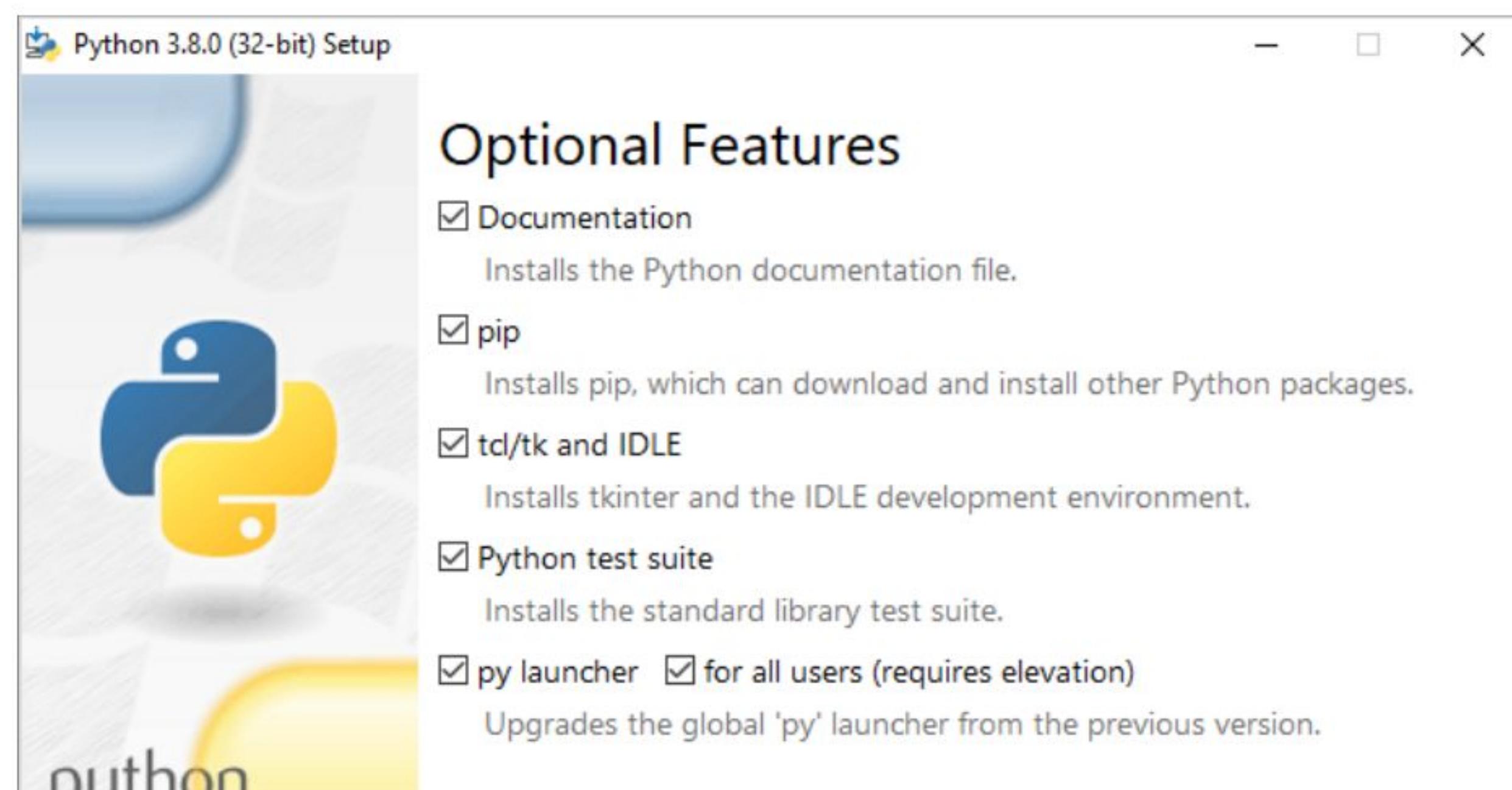
- STEP 1** Start by opening your web browser to www.python.org/downloads/. Look for the button detailing the Download link for Python 3.x. Python is regularly updated, changing the last digit for each bug fix and update. Therefore, don't worry if you see Python 3.8, or more, as long as it's Python 3, the code in this book will work fine.



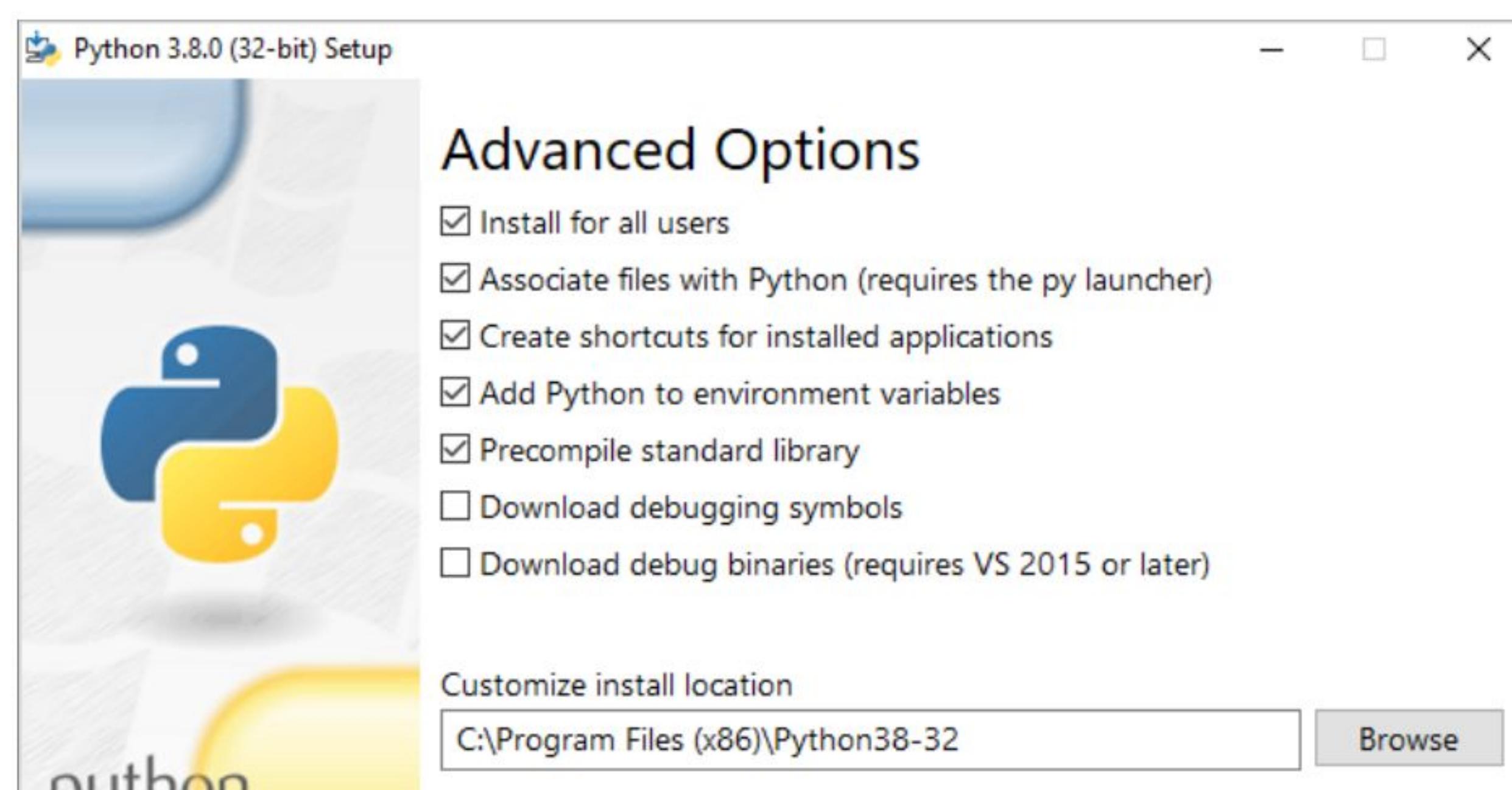
- STEP 2** Click the Download button for version 3.x and save the file to your Downloads folder. When the file is downloaded, double-click the executable and the Python installation wizard will launch. From here, you have two choices: Install Now and Customise Installation. We recommend opting for the Customise Installation link.



- STEP 3** Choosing the Customise option allows you to specify certain parameters, and whilst you may stay with the defaults, it's a good habit to adopt as, sometimes (not with Python, thankfully), installers can include unwanted additional features. On the first screen available, ensure all boxes are ticked and click the Next button.

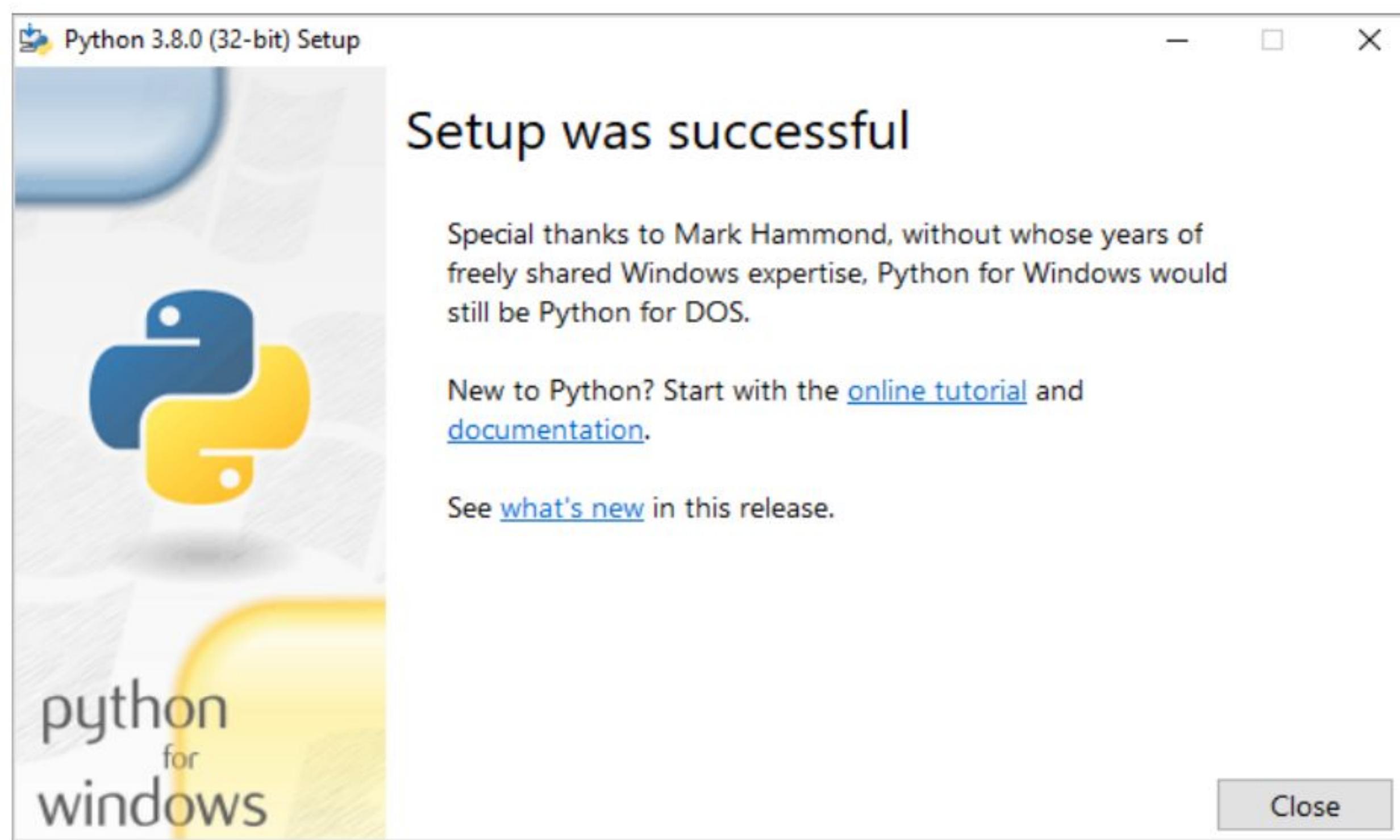


- STEP 4** The next page of options include some interesting additions to Python. Ensure the Associate file with Python, Create Shortcuts, Add Python to Environment Variables, Precompile Standard Library and Install for All Users options are ticked. These make using Python later much easier. Click Install when you're ready to continue.

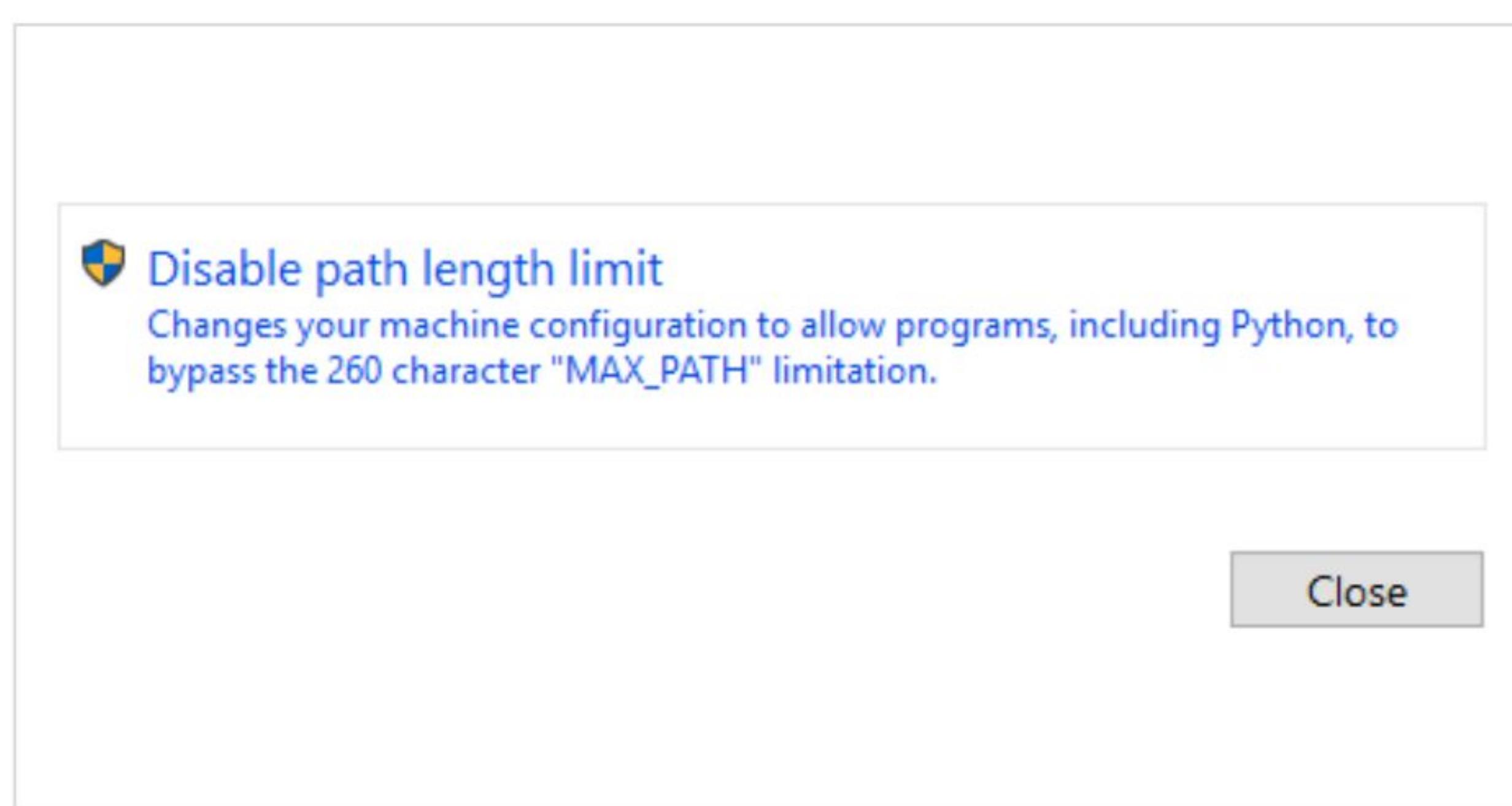


**STEP 5**

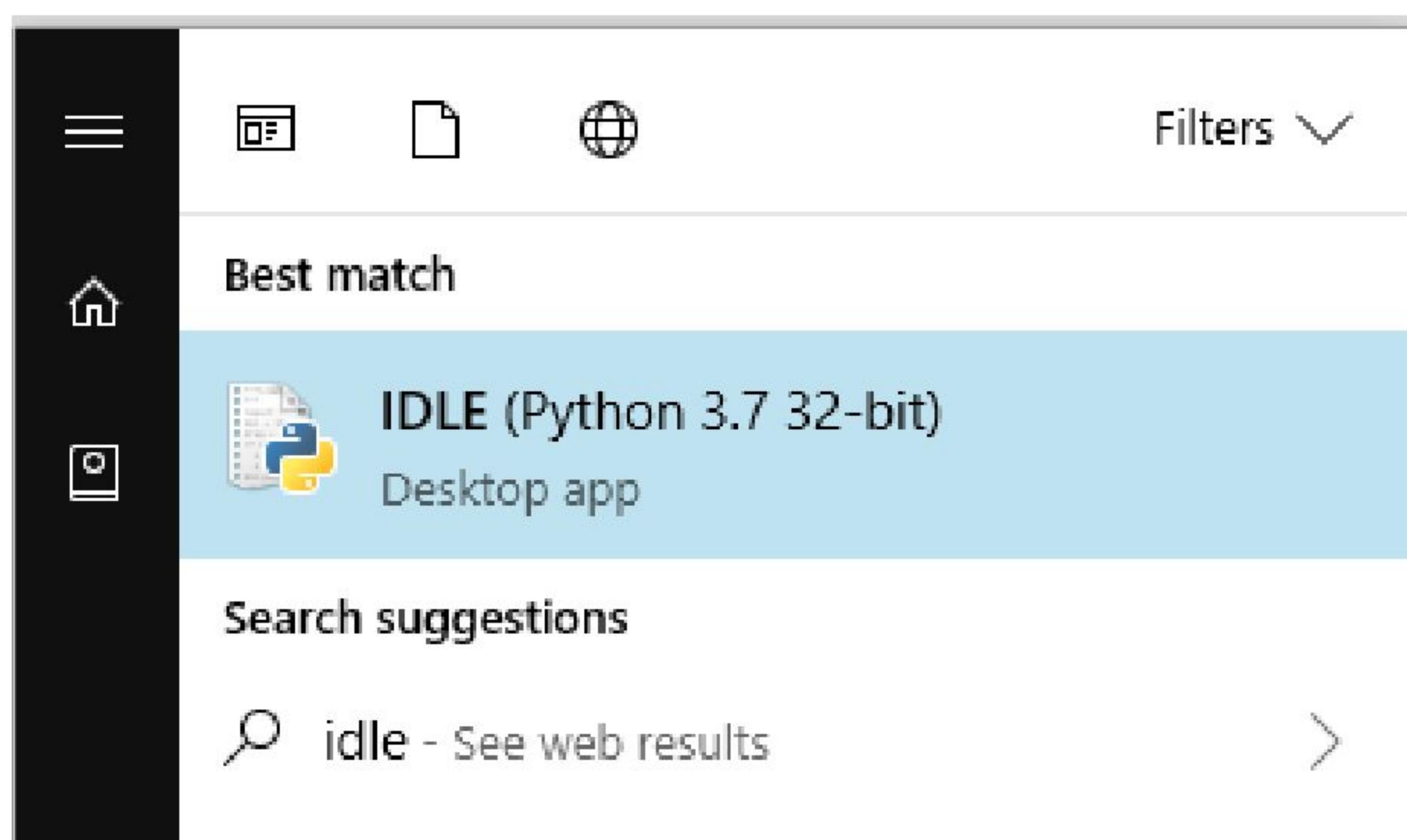
You may need to confirm the installation with the Windows authentication notification. Simply click Yes and Python will begin to install. Once the installation is complete, the final Python wizard page will allow you to view the latest release notes and follow some online tutorials.

**STEP 6**

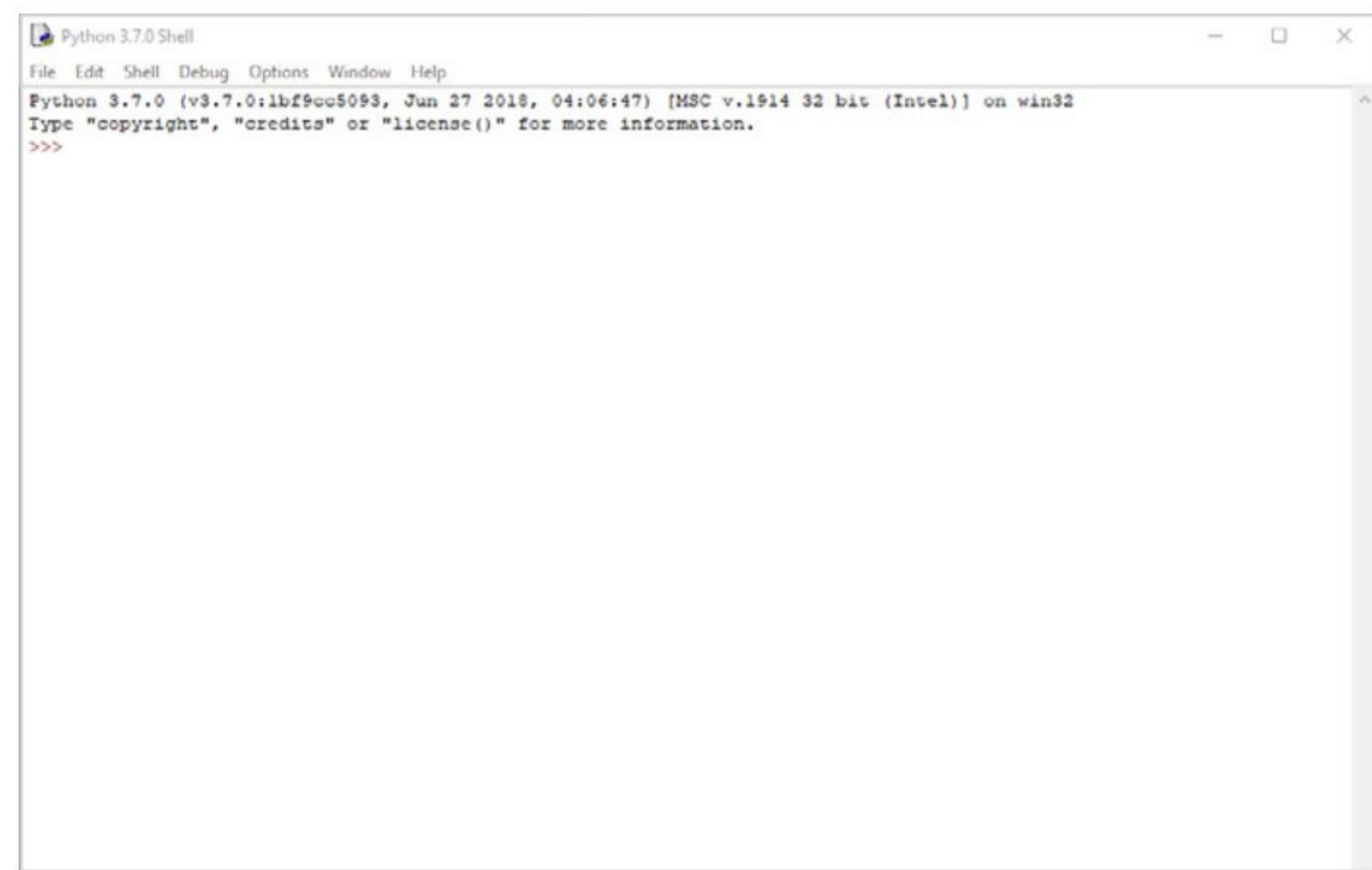
Before you close the install wizard window however, it's best to click on the link next to the shield detailed Disable Path Length Limit. This will allow Python to bypass the Windows 260 character limitation, enabling you to execute Python programs stored in deep folders arrangements. Click Yes again, to authenticate the process, then you can Close the installation window.

**STEP 7**

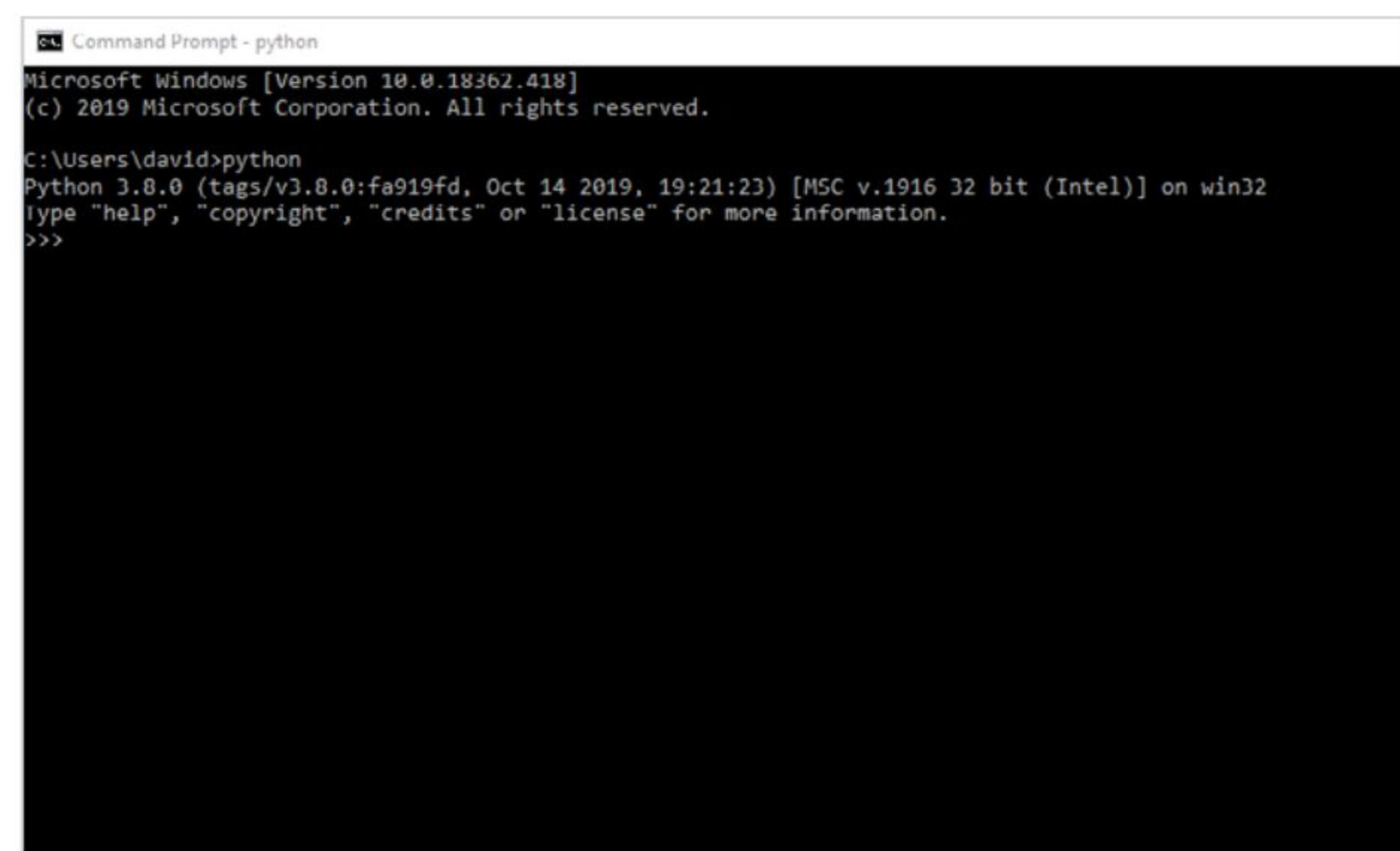
Windows 10 users can now find the installed Python 3.x within the Start button Recently Added section. The first link, Python 3.x (32-bit) will launch the command line version of Python when clicked (more on that in a moment). To open the IDLE, type IDLE into Windows start.

**STEP 8**

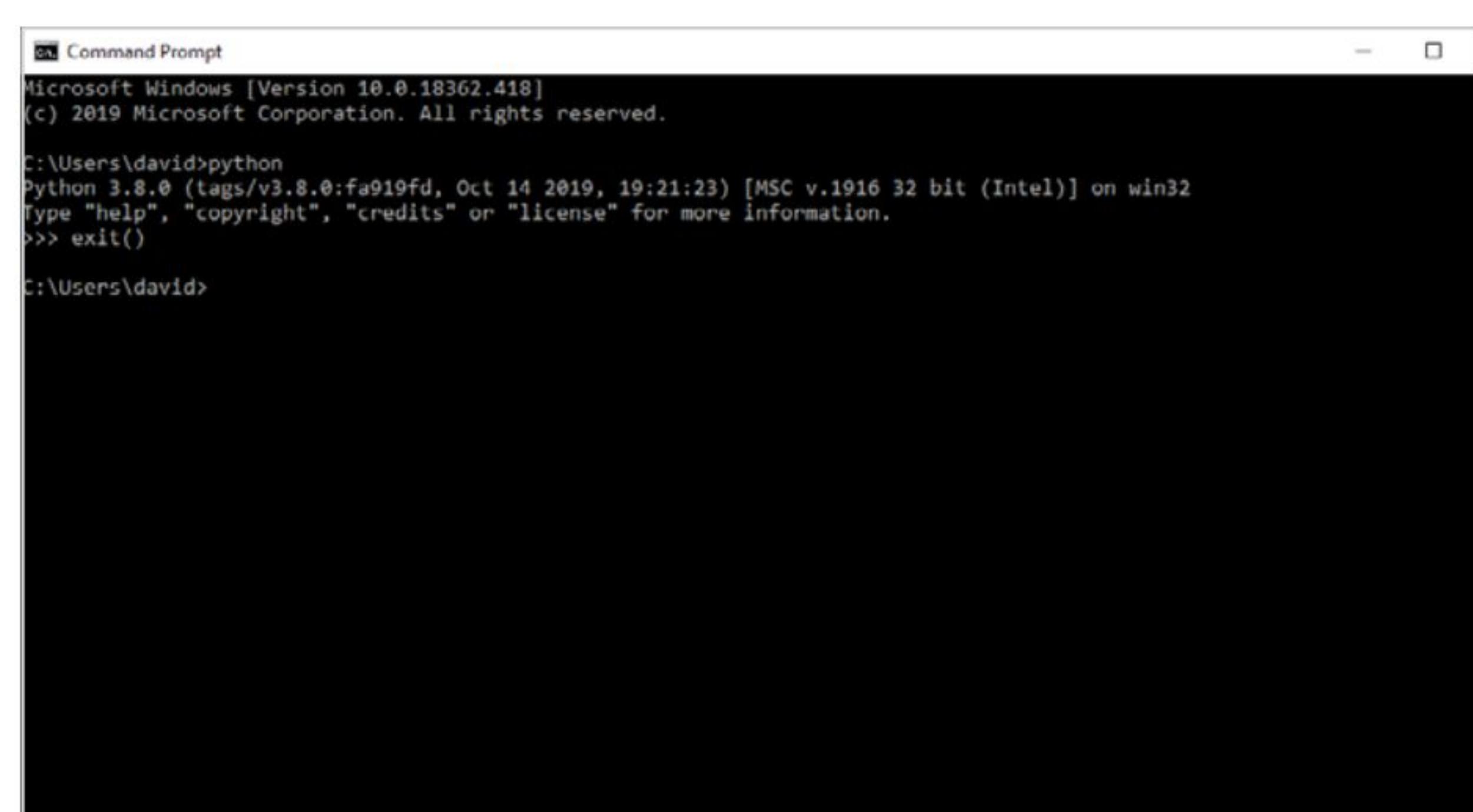
Clicking on the IDLE (Python 3.x 32-bit) link will launch the Python Shell, where you can begin your Python programming journey. Don't worry if your version is newer, as long as it's Python 3.x our code works inside your Python 3 interface.

**STEP 9**

If you now click on the Windows Start button again, and this time type: CMD, you'll be presented with the Command Prompt link. Click it to get to the Windows command line environment. To enter Python within the command line, you need to type: **python** and press Enter.

**STEP 10**

The command line version of Python works in much the same way as the Shell you opened in Step 8; note the three left-facing arrows (>>>). Whilst it's a perfectly fine environment, it's not too user-friendly, so leave the command line for now. Enter: **exit()** to leave and close the Command Prompt window.





How to Set Up Python in Linux

While the Raspberry Pi's operating system contains the latest, stable version of Python, other Linux distros don't come with Python 3 pre-installed. If you're not going down the Pi route, then here's how to check and install Python for Linux.

PYTHON PENGUIN

Linux is such a versatile operating system that it's often difficult to nail down just one way of doing something. Different distributions go about installing software in different ways, so for this particular tutorial we will stick to Linux Mint.

STEP 1 First you need to ascertain which version of Python is currently installed in your Linux system. To begin with, drop into a Terminal session from your distro's menu, or hit the Ctrl+Alt+T keys.

```
david@david-Mint:~\nFile Edit View Search Terminal Help\n[david@david-Mint:~$ ]
```

STEP 2 Next, enter: `python --version` into the Terminal screen. You should have the output relating to version 2.x of Python in the display. Most Linux distro come with both Python 2 and 3 by default, as there's plenty of code out there still available for Python 2. Now enter: `python3 --version`.

```
david@david-Mint:~\nFile Edit View Search Terminal Help\n[david@david-Mint:~$ python --version\nPython 2.7.15rc1\n[david@david-Mint:~$ python3 --version\nPython 3.6.7\n[david@david-Mint:~$ ]
```

STEP 3 In our case we have both Python 2 and 3 installed. As long as Python 3.x.x is installed, then the code in our tutorials will work. It's always worth checking to see if the distro has been updated with the latest versions, enter: `sudo apt-get update && sudo apt-get upgrade` to update the system.

```
david@david-Mint:~\nFile Edit View Search Terminal Help\n[david@david-Mint:~$ python --version\nPython 2.7.15rc1\n[david@david-Mint:~$ python3 --version\nPython 3.6.7\n[david@david-Mint:~$ sudo apt-get update && sudo apt-get upgrade\n[sudo] password for david: ]
```

STEP 4 Once the update and upgrade completes, enter: `python3 --version` again to see if Python 3.x is updated, or even installed. As long as you have Python 3.x, you're running the most recent major version, the numbers after the 3. indicate patches plus further updates. Often they're unnecessary, but they can contain vital new elements.

```
david@david-Mint:~\nFile Edit View Search Terminal Help\nNeed to get 1,409 kB of archives.\nAfter this operation, 23.6 kB of additional disk space will be used.\nDo you want to continue? [Y/n] y\nGet:1 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 libasound2 amd64 1:1.3-5ubuntu0.2 [359 kB]\nGet:2 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 libasound2-data all 1:1.3-5ubuntu0.2 [36.5 kB]\nGet:3 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 linux-libc-dev amd64 4.15.0-44.47 [1,013 kB]\nFetched 1,409 kB in 0s (3,023 kB/s)\n(Reading database ... 290768 files and directories currently installed.)\nPreparing to unpack .../libasound2_1.3-5ubuntu0.2_amd64.deb ...\nUnpacking libasound2:amd64 (1.3-5ubuntu0.2) over (1.3-5ubuntu0.1) ...\nPreparing to unpack .../libasound2-data_1.3-5ubuntu0.2_all.deb ...\nUnpacking libasound2-data (1.3-5ubuntu0.2) over (1.3-5ubuntu0.1) ...\nPreparing to unpack .../linux-libc-dev_4.15.0-44.47_amd64.deb ...\nUnpacking linux-libc-dev:amd64 (4.15.0-44.47) over (4.15.0-43.46) ...\nSetting up libasound2-data (1.3-5ubuntu0.2) ...\nSetting up linux-libc-dev:amd64 (4.15.0-44.47) ...
```

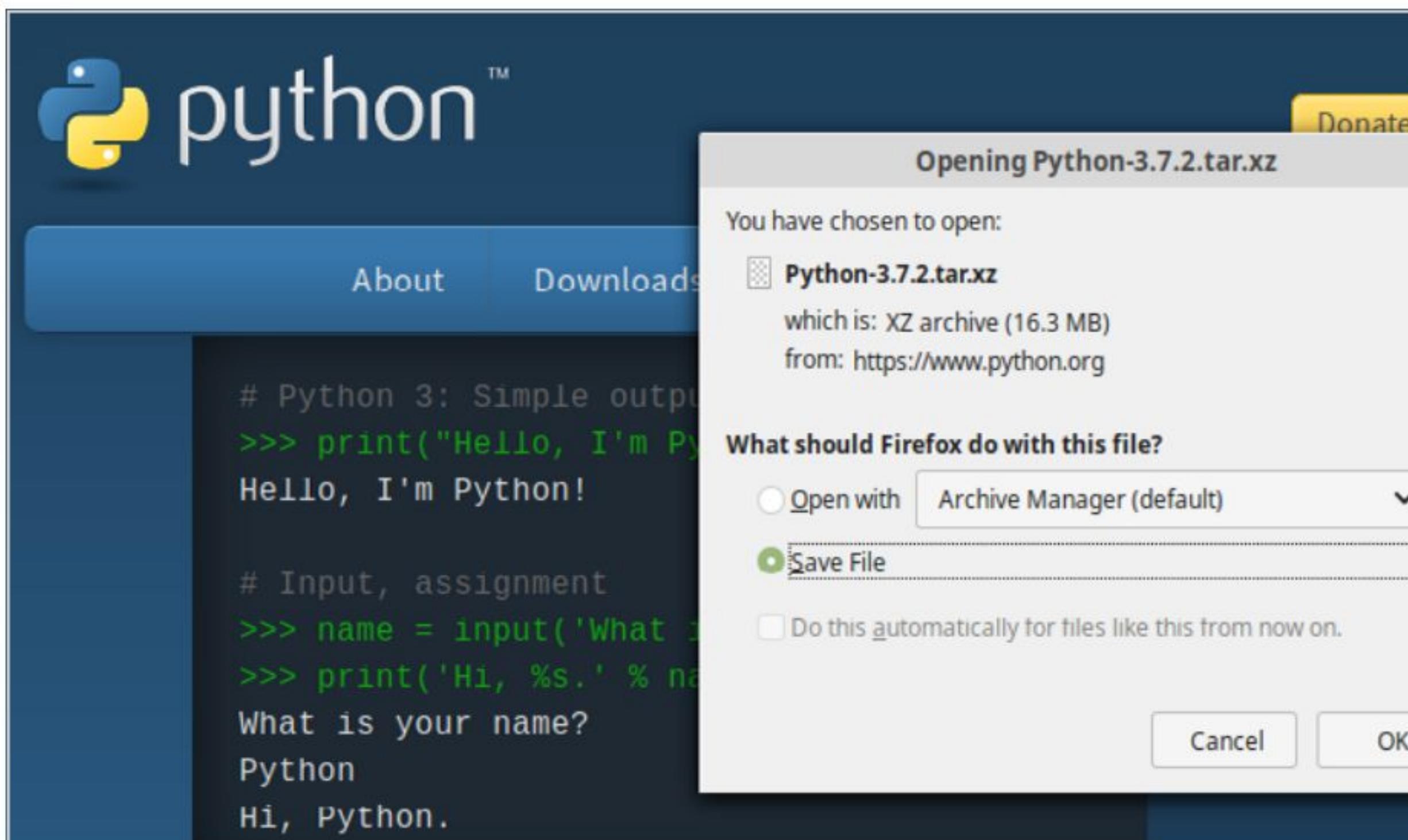
STEP 5 However, if you want the latest, cutting edge version, you'll need to build Python from source. Start by entering these commands into the Terminal:

```
sudo apt-get install build-essential checkinstall\nsudo apt-get install libreadline-gplv2-dev\nlibncursesw5-dev libssl-dev libsqlite3-dev tk-dev\nlibgdbm-dev libc6-dev libbz2-dev
```

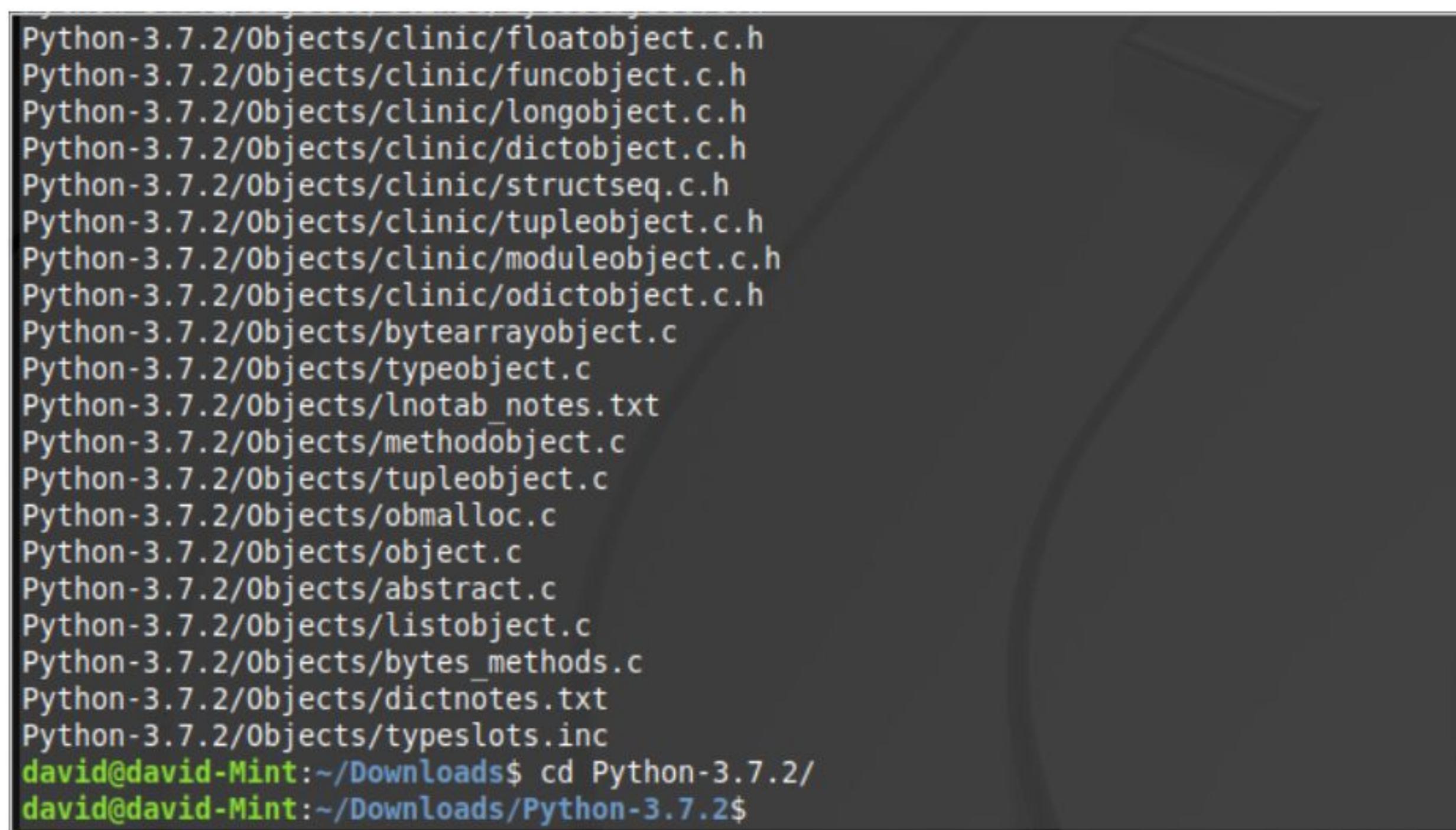
```
david@david-Mint:~\nFile Edit View Search Terminal Help\n[david@david-Mint:~$ sudo apt-get install build-essential checkinstall\nReading package lists... Done\nBuilding dependency tree\nReading state information... Done\nbuild-essential is already the newest version (12.4ubuntu1).\nThe following NEW packages will be installed\ncheckinstall\n0 to upgrade, 1 to newly install, 0 to remove and 3 not to upgrade.\nNeed to get 97.1 kB of archives.\nAfter this operation, 438 kB of additional disk space will be used.\nDo you want to continue? [Y/n] y
```



STEP 6 Open up your Linux web browser and go to the Python download page: <https://www.python.org/downloads>. Click on the Downloads, followed by the button under the Python Source window. This opens a download dialogue box, choose a location, then start the download process.



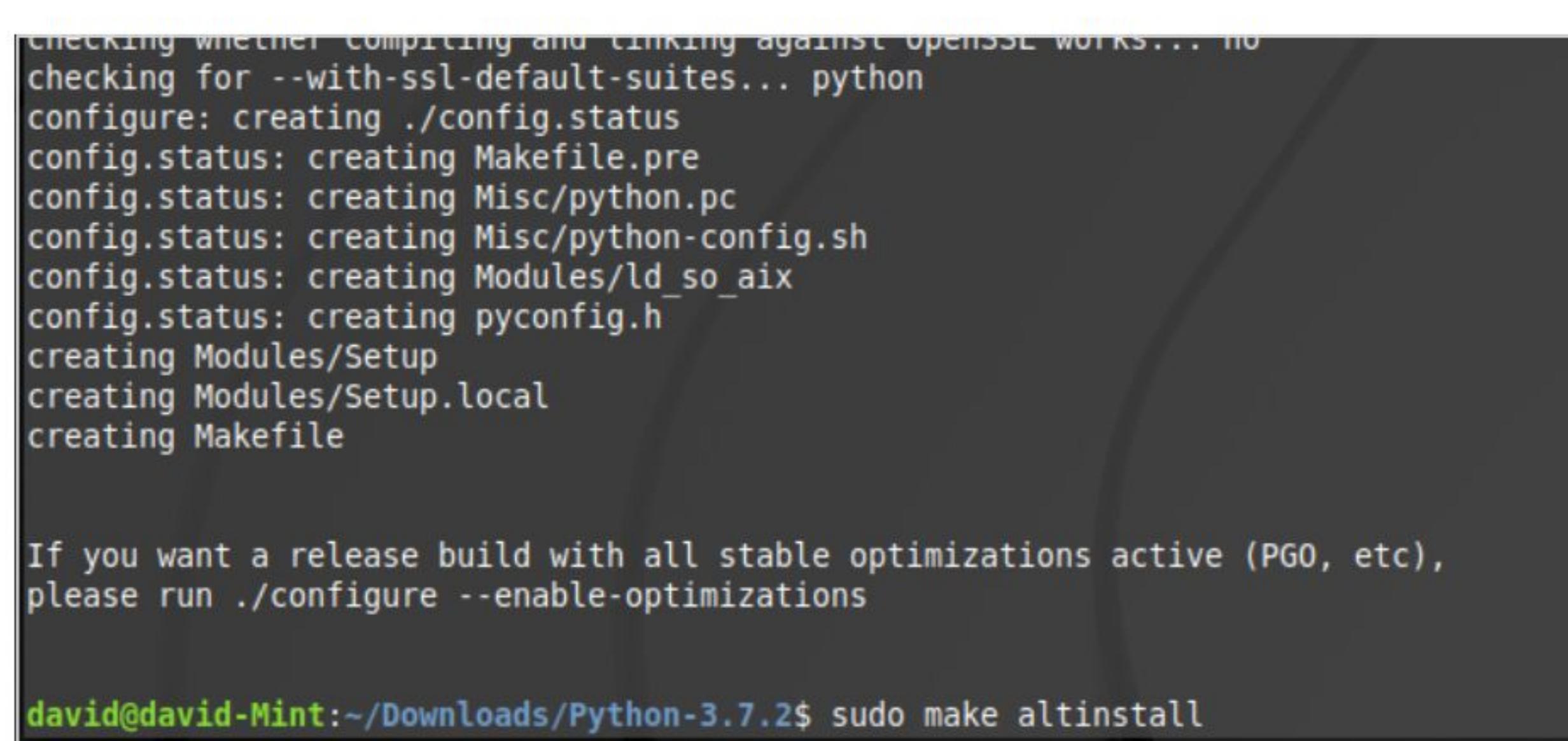
STEP 7 In the Terminal, go to the Downloads folder by entering: `cd Downloads/`. Then unzip the contents of the downloaded Python source code with: `tar -xvf Python-3.Y.Y.tar.xz` (replace the Y's with the version numbers you've downloaded). Now enter the newly unzipped folder with: `cd Python-3.Y.Y/`.



STEP 8 Within the Python folder, enter:

```
./configure  
sudo make altinstall
```

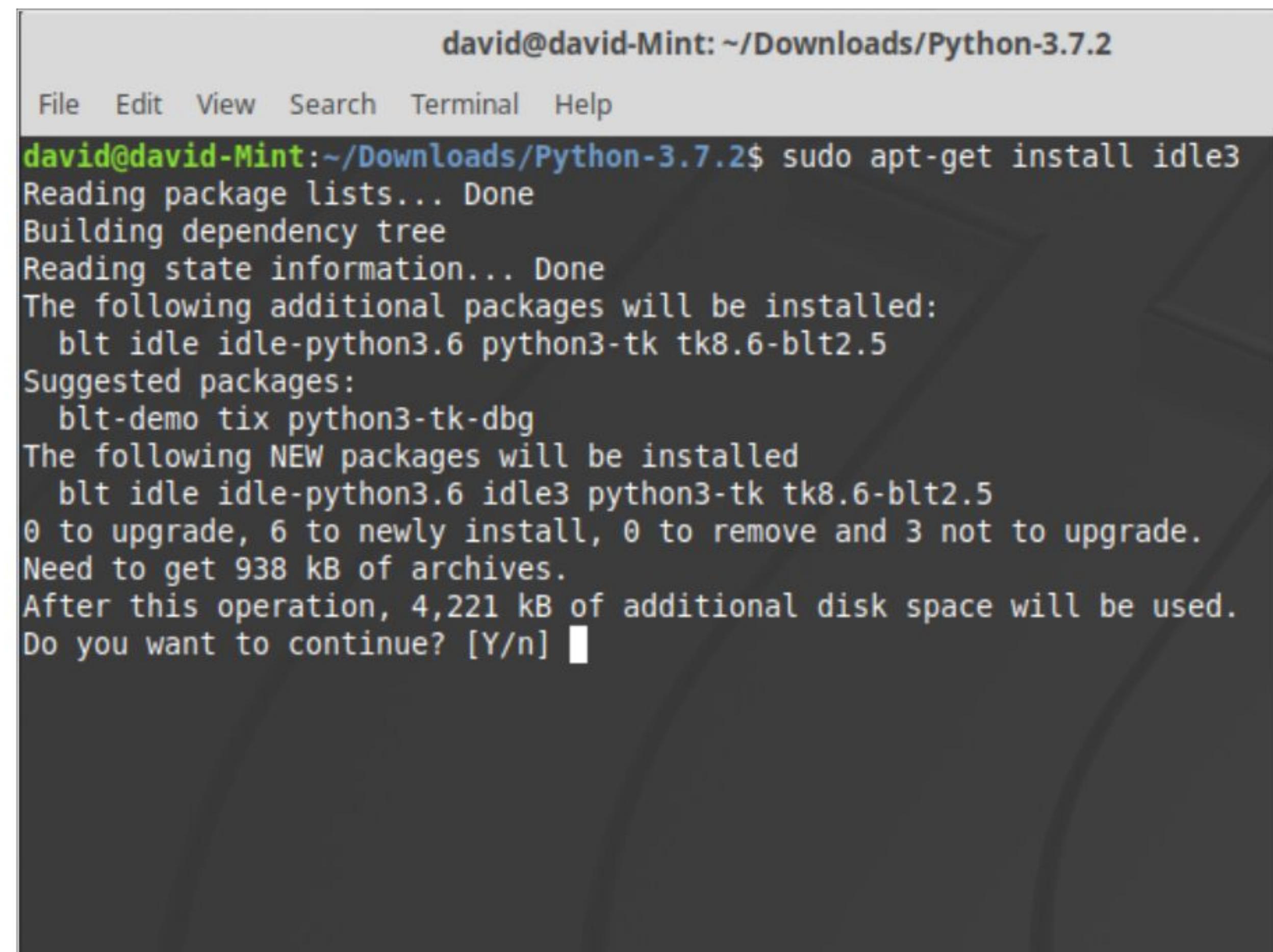
This could take a while, depending on the speed of your computer. Once finished, enter: `python3.7 --version` to check the latest installed version. You now have Python 3.7 installed, alongside older Python 3.x.x and Python 2.



STEP 9 For the GUI IDLE, you'll need to enter the following command into the Terminal:

```
sudo apt-get install idle3
```

The IDLE can then be started with the command: `idle3`. Note, that IDLE runs a different version to the one you installed from source.



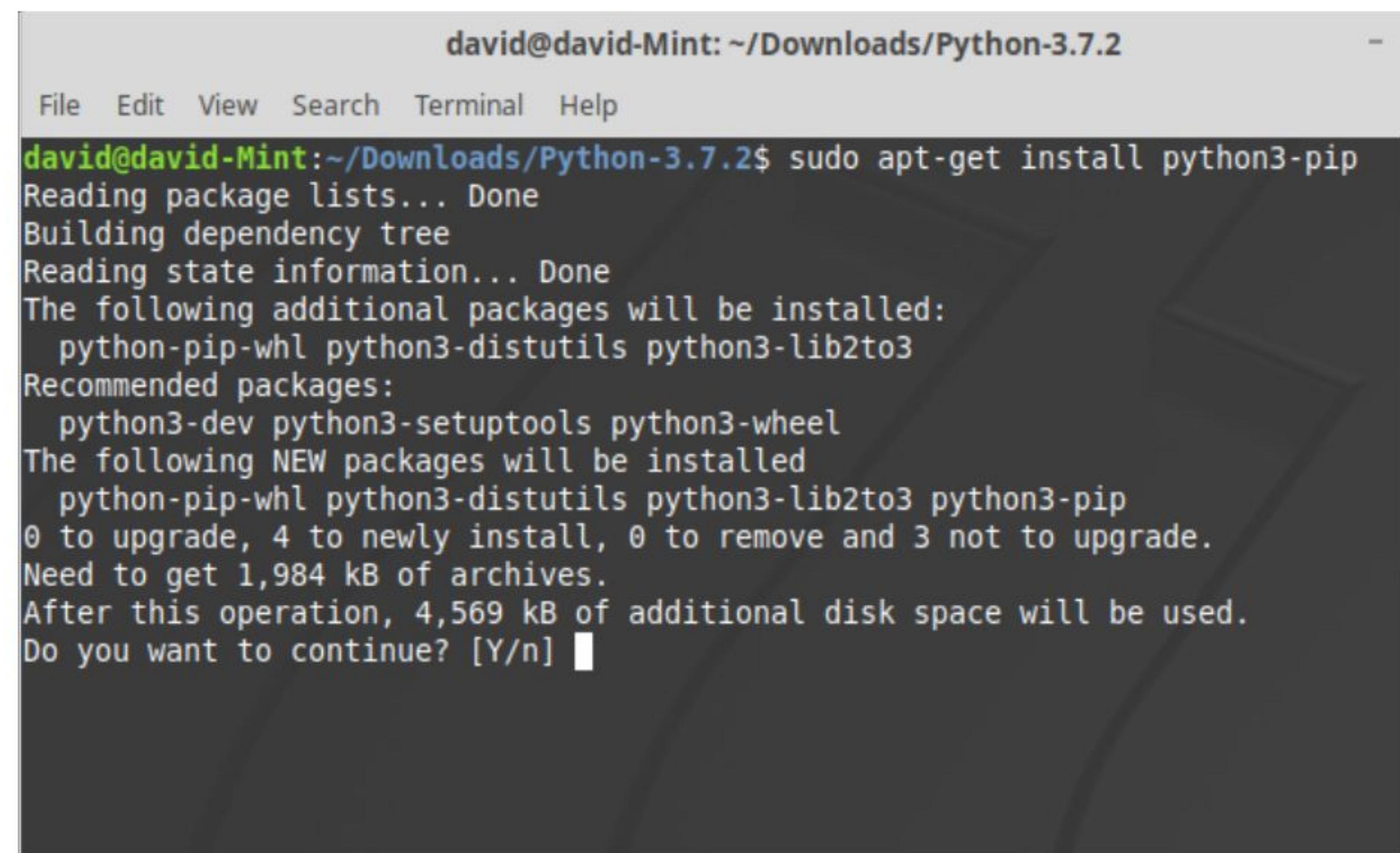
STEP 10 You'll also need PIP (Pip Installs Packages), which is a tool to help you install more modules and extras.

Enter: `sudo apt-get install python3-pip`

Once PIP is installed, check for the latest update with:

```
pip3 install --upgrade pip
```

When complete, close the Terminal and Python 3.x will be available via the Programming section in your distro's menu.



PYTHON ON macOS

Installation of Python on macOS can be done in much the same way as the Windows installation. Simply go to the Python webpage, hover your mouse pointer over the Downloads link and select Mac OS X from the options. You will then be guided to the Python releases for Mac versions, along with the necessary installers for macOS 64-bit for OS X 10.9 and later.