

Intro to Python — SMM692

Getting Started with Python

Simone Santoni

Bayes Business School

MSc Pre-Course Series

Outline

- 1 Installing Python
 - Options
 - Installation Procedure for Anaconda
- 2 How Python Runs Programs
- 3 How We Run Python Programs
 - Non-Interactive Approach
 - Interactive Approach

Outline

- 1 Installing Python
 - Options
 - Installation Procedure for Anaconda
- 2 How Python Runs Programs
- 3 How We Run Python Programs
 - Non-Interactive Approach
 - Interactive Approach

Option 1: Official Installer

Download the latest version for macOS

[Download Python 3.10.5](#)

Looking for Python with a different OS? Python for [Windows](#), [Linux/UNIX](#), [macOS](#), [Other](#)

Want to help test development versions of Python? [Prereleases](#), [Docker images](#)

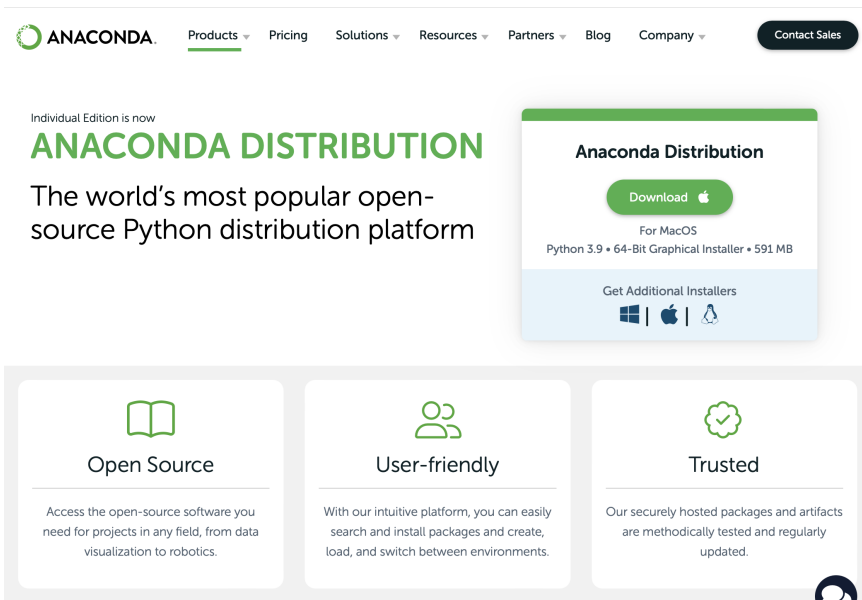
Looking for Python 2.7? See below for specific releases

Active Python Releases

For more information visit the [Python Developer's Guide](#).

Python version	Maintenance status	First released	End of support	Release schedule
3.10	bugfix	2021-10-04	2026-10	PEP 619
3.9	security	2020-10-05	2025-10	PEP 596

Option 2: Anaconda Distro (Preferred Way)



The screenshot shows the Anaconda website homepage. At the top is a navigation bar with the Anaconda logo, links for Products, Pricing, Solutions, Resources, Partners, Blog, and Company, and a Contact Sales button. The main content area features the text 'Individual Edition is now' followed by 'ANACONDA DISTRIBUTION' in large green letters. Below this is the tagline 'The world's most popular open-source Python distribution platform'. To the right is a download box for 'Anaconda Distribution' with a 'Download' button for macOS, specifying 'Python 3.9 • 64-Bit Graphical Installer • 591 MB'. Below the download box is a section 'Get Additional Installers' with icons for Windows, macOS, and Linux. At the bottom are three feature boxes: 'Open Source' (with an open book icon), 'User-friendly' (with a person icon), and 'Trusted' (with a checkmark icon). Each box contains a brief description of the feature.


ANACONDA. Products Pricing Solutions Resources Partners Blog Company [Contact Sales](#)

Individual Edition is now

ANACONDA DISTRIBUTION




The world's most popular open-source Python distribution platform


Anaconda Distribution

[Download](#) 

For MacOS
Python 3.9 • 64-Bit Graphical Installer • 591 MB


Get Additional Installers

 |  | 




Open Source

Access the open-source software you need for projects in any field, from data visualization to robotics.



User-friendly

With our intuitive platform, you can easily search and install packages and create, load, and switch between environments.



Trusted

Our securely hosted packages and artifacts are methodically tested and regularly updated.

Outline

- 1 Installing Python
 - Options
 - Installation Procedure for Anaconda
- 2 How Python Runs Programs
- 3 How We Run Python Programs
 - Non-Interactive Approach
 - Interactive Approach

Steps

- ❶ Download the installer for your operating system (unless you have a very old machine running Win, go for the 64-Bit version)
- ❷ Run the installer
 - For Linux: navigate to the folder where you have downloaded the installer as per step 1, open a shell session, then run `$ bash ./Anaconda3-XXXX.XX-Linux-x86-64.sh`
 - For Win and Mac OS: just run the graphical installer downloaded in step 1
- ❸ Accept the terms proposed by the Anaconda people to use their software, comprising Python, the conda package manager, and a bundle of modules for data science
- ❹ That's it!
 - For Linux users: if you accepted the default installation options, an environmental variable has been created either in your `.bashrc` or `.zshrc`. That means you can access the various pieces of software included in the Anaconda installation (e.g., Anaconda Navigator) from a shell session
 - For Win and Mac OS users: the various pieces of software included in the Anaconda installation are available from the menu of your system

Outline

- 1 Installing Python
 - Options
 - Installation Procedure for Anaconda
- 2 How Python Runs Programs
- 3 How We Run Python Programs
 - **Non-Interactive Approach**
 - Interactive Approach

Script Preparation → Script Run

Step 1: script preparation

The below displayed Python code achieves two things: i) it prints the string object “Bazinga!”, and ii) it prints the result of the algebraic operation $4 + 2$. Note that all lines starting with # are not considered Python code — instead, they are comments that illustrate/explain the logic of the script.

```
# This prints a string object  
print("Bazinga")  
# This prints the result of an algebraic operation  
print(2 + 4)
```

Step 2: script run



```
(base) → ~ python simple_script.py  
Bazinga  
6  
(base) → ~
```

Outline

- 1 Installing Python
 - Options
 - Installation Procedure for Anaconda
- 2 How Python Runs Programs
- 3 How We Run Python Programs
 - Non-Interactive Approach
 - Interactive Approach

Running a Python Shell in the Terminal

```
(base) → ~ python
Python 3.9.7 (default, Sep 16 2021, 08:50:36)
[Clang 10.0.0 ] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Bazinga")
Bazinga
>>> print(2 + 4)
6
>>> █
```

Running an IPython Shell in the Terminal

```
(base) → ~ ipython
Python 3.9.7 (default, Sep 16 2021, 08:50:36)
Type 'copyright', 'credits' or 'license' for more information
IPython 7.29.0 -- An enhanced Interactive Python. Type '?' for help.

In [1]: print("Bazinga")
Bazinga

In [2]: print(2 + 4)
6

In [3]:
```

Running a Python IDE

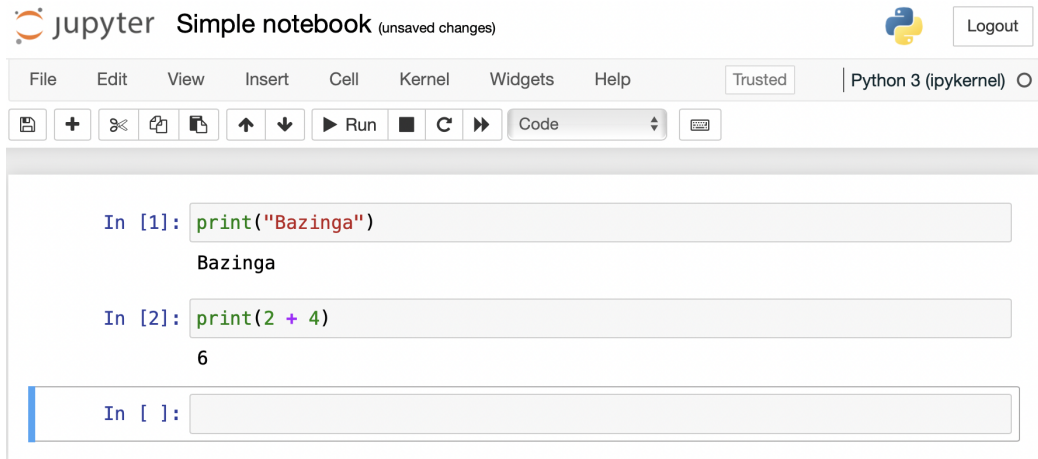
There are plenty of Python IDEs in the market, including:

- Colab (online)
- IDLE
- Datalore (online)
- Jupyter/Jupyterlab
- PyCharm
- Spyder
- Thonny
- Wing

By installing a couple of plugins, the following (advanced) text editors can turn into Python IDEs:

- Emacs
- Vim/Neovim
- VSCode

Interactive Python Coding with Jupyter



The screenshot shows the Jupyter Simple notebook interface. At the top, the Jupyter logo is followed by the text "Simple notebook (unsaved changes)". To the right is a Python logo and a "Logout" button. Below this is a menu bar with options: File, Edit, View, Insert, Cell, Kernel, Widgets, Help, Trusted, and Python 3 (ipykernel). Under the menu bar is a toolbar with icons for saving, adding, deleting, copying, pasting, undo, redo, and running code. The main area contains three code cells. The first cell has the input `In [1]: print("Bazinga")` and the output `Bazinga`. The second cell has the input `In [2]: print(2 + 4)` and the output `6`. The third cell is empty, showing `In []:`.

jupyter Simple notebook (unsaved changes)

Python 3 (ipykernel)

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

Save Add Delete Copy Paste Undo Redo Run Code

```
In [1]: print("Bazinga")
Bazinga
```

```
In [2]: print(2 + 4)
6
```

```
In [ ]:
```

Interactive Python Coding with VSCode

The screenshot shows the VS Code interface with a file named `simple_script.py` open. The editor contains the following code:

```
1 print("Bazinga")
2 print(2 + 4)
```

The right-hand side of the interface displays the interactive Python environment. It shows the command prompt for `tech_sci (Python 3.9.7)` and the output of the script:

```
Python 3.9.7 (default, Sep 16 2021, 08:50:36)
Type 'copyright', 'credits' or 'license' for more
information
IPython 7.29.0 -- An enhanced Interactive
Python. Type '?' for help.

✓ print("Bazinga") ...
... Bazinga

✓ print(2 + 4) ...
... 6
```

Interactive Python Coding in Jupyter with VSCode

The screenshot displays the VS Code Jupyter interface for a notebook titled "simple_script.ipynb — intro-to-Python-SMM692". The interface includes a sidebar on the left with icons for Explorer, Search, Source Control, Run and Debug, Testing, Extensions, Python, Remote Explorer, and Docker. The main editor area shows the notebook content. The top bar indicates the current kernel is "tech_sci (Python 3.9.7)".

The notebook contains two executed code cells:

- Cell [1]:

```
print("Bazinga")
```

Output: Bazinga
- Cell [2]:

```
print(2 + 4)
```

Output: 6

A third, empty code cell is currently selected at the bottom, with its input area containing a cursor and a small toolbar with icons for running, stepping through, and deleting code.