

# Other Details & Info

April 29, 2020 2:29 PM

Download Python

<https://www.python.org/downloads/>

Code Examples

<https://wiki.python.org/moin/BeginnersGuide/Examples>

Step 1: The tutorial

<https://docs.python.org/3/tutorial/>

Tutorial recommended by Tiago

[https://cs231n.github.io/python-numpy-tutorial/?fbclid=IwAR3fv4dFqhT-K916116XQ6G6w5mifajwK0ebFag\\_u970WE-VMkNT7PZz0GU](https://cs231n.github.io/python-numpy-tutorial/?fbclid=IwAR3fv4dFqhT-K916116XQ6G6w5mifajwK0ebFag_u970WE-VMkNT7PZz0GU)

# Scripting Environments (IDE's)

April 29, 2020 2:28 PM

This official Python page includes many.

<https://wiki.python.org/moin/IntegratedDevelopmentEnvironments>

Tiago recommended using Anaconda and VS Code:

Anaconda:

<https://www.anaconda.com/products/individual>

<https://get.anaconda.com/distribution/tutorial/>

Visual Studio Code:

<https://code.visualstudio.com/docs/languages/python>

Name	Where it Runs	Details
Google Collaboratory	In browser	<ul style="list-style-type: none"><li>• Installation free</li><li>• Cloud computing</li><li>• "Like a Jupyter notebook stored on Google Drive"</li><li>• Code is entered and run in cells<ul style="list-style-type: none"><li>• Text cells provide a place to comments that can have rich formatting</li></ul></li><li>• Can search built-in library of code snippets</li><li>• <a href="http://research.google.com/seedbank">http://research.google.com/seedbank</a> -&gt; A bank of many workbooks for a wide variety of applications which you can view, run</li><li>• Can choose what to run on: GPU's or TPU's</li></ul>

# Syntax

April 29, 2020 2:28 PM

The complete syntax guide:

<https://docs.python.org/3/reference/>

The language is interpreted as opposed to compiled. That means that the code gets compiled at run time. With a compiled language, the code that \*you\* write is compiled once, into machine code - all bits. That machine code itself is distributed to customers etc. as .exe or whatever format. Whereas an interpreted language gets interpreted every time it is run. That means that you ship out the code, with an interpreter that is specialized for the hardware or OS. Then each computer that runs it will compile it to machine code, (0's and 1's) before running it, every time.

Uses lambda functions

# Data Types

April 29, 2020 2:28 PM

Idea	Details
Dynamically Typed	<ul style="list-style-type: none"><li>• No variable declarations!</li><li>• Ties into the fact that its interpreted. If it were being compiled, the program would need to contain the declarations to ensure enough memory was stored for running. But because it is interpreted line by line, it doesn't need to do that. It just sticks a value in memory when the time comes, then labels it</li></ul>

# Data Structures

April 29, 2020 2:28 PM

Refer to the Standard Library for full list of non-essential built-in object types, functions, and modules

# Libraries

April 29, 2020 2:29 PM

**A website for storing code snippets** - There will be a code example for basically anything you want to do

<http://code.activestate.com/recipes/langs/python/>

## Keep Code Compatible

In Anaconda, you can define 'environments' -so code you write will always be run with the packages that it was coded on. That environment will not take on package updates, ensuring code remains functional and doesn't break. Within the environments tab simply select which packages and versions to use.

## DEFINITIONS

*Module*- Python code contained in a file (any file....)

*Package*- Python codes contained in a directory of file(s)

*Library*- A collection of modules and packages

Colloquially/in practical usage, can think of libraries/packages as collections of custom code that are distributed for a particular purpose (for example, a data analysis package containing all the custom functions etc. that do data analysis)

Library	Application	Features	Documents
Pandas	Data Analysis	<ul style="list-style-type: none"><li>• Data Frame object for data manipulation with integrated indexing</li><li>• Tools for reading and writing data from memory to different formats like CSV, text, Excel, SQL, etc.</li><li>• Means for handling messy data</li><li>• Flexible pivoting of data sets</li><li>• Aggregating or transforming data with split-apply-combine operations</li></ul>	<a href="https://pandas.pydata.org/docs/">https://pandas.pydata.org/docs/</a>

<https://gym.openai.com>

"A toolkit for developing and comparing reinforcement learning algorithms"

-Tiago used/recommends it