Introduction to Python

Mahesh Mohan M R
Centre of Excellence in AI
Indian Institute of Technology Kharagpur

Python

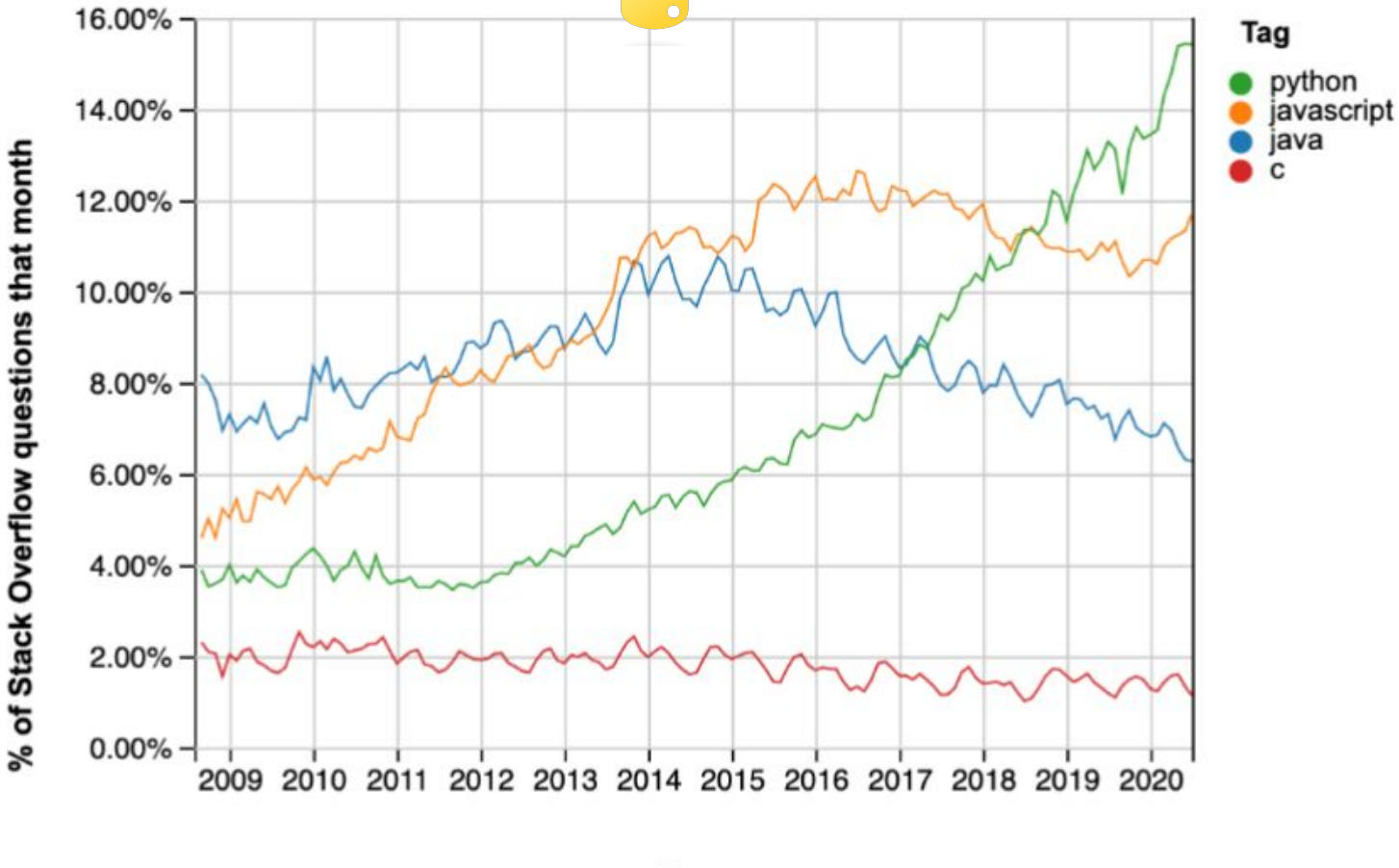
- Created in 1990 by Guido van Rossum

Python: Fun to Use

- Created in 1990 by Guido van Rossum
- Named after Monty Python

vs

Python: Fun to Use



Source: A practical intro to Python 3, 4th edition, Real python

Python: Fun to Use

1. An easy and intuitive language
2. Open source :)
3. Code that is as understandable as plain English
4. Suitability for all fields, and everyday tasks.

Python Highlights

- Easily extended with new functions and data types implemented in C or C++
- Available on Windows, Mac OS X, and Unix operating systems
- Allows you to split your program into modules that can be reused.
- Comes with a large collection of standard modules

Matrices/vectors and Mathematical functions

Data Manipulation and Analysis

Plotting Data and Results

Machine learning library

Deep learning library

Machine Learning Process

NumPy

Matrices/vectors and Mathematical functions

pandas

Data Manipulation and Analysis

matplotlib

Plotting Data and Results

scikit learn

Machine learning library

PyTorch

Deep learning library

Python 3.12.1 documentation

Welcome! This is the official documentation for Python 3.12.1.

Parts of the documentation:

What's new in Python 3.12?
or all "What's new" documents since 2.0

Tutorial
start here

Library Reference
keep this under your pillow

Language Reference
describes syntax and language elements

Python Setup and Usage
how to use Python on different platforms

Python HOWTOs
in-depth documents on specific topics

Installing Python Modules
installing from the Python Package Index & other sources

Distributing Python Modules
publishing modules for installation by others

Extending and Embedding
tutorial for C/C++ programmers

Python/C API
reference for C/C++ programmers

FAQs
frequently asked questions (with answers!)

https://docs.python.org/3/

Running Python

- Python is pre-installed on most Unix systems
- Download from http://python.org/downloads/ and google
- Python comes with a large library of standard modules
- There are several options for an IDE
  - Pycharm – works well with Windows, Linux and Mac
  - Shell with your favorite text editor
  - Jupyter notebook
  - ……. Lots and Lots …….

In the Lab tutorials, we use Google Collab

Google Collab

Installation and Run in Google Suite

Installation and Run in your Computer

Collaboration
Additional Hardware
Cloud Based

Running code,
Taking notes
Show multimedia
Interactivity

Read–eval–print loop Terminal,
Kernel with the frontend interfaces

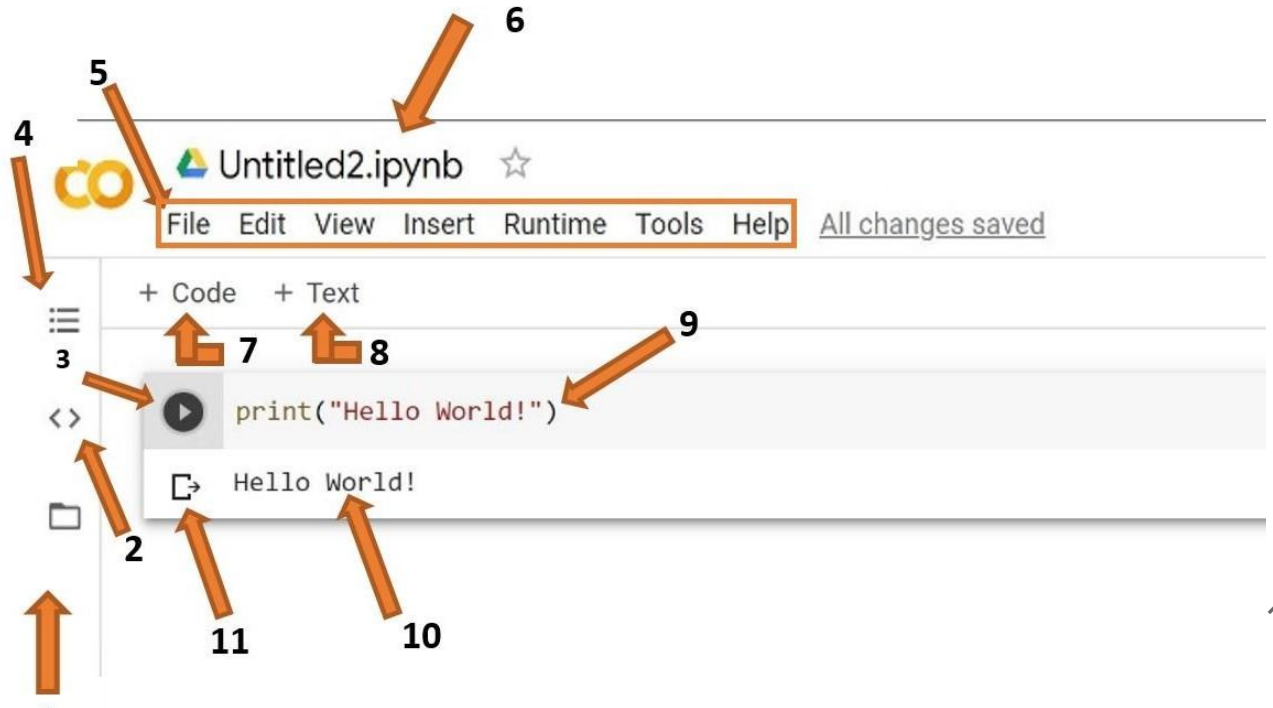
iPython

Jupyter Notebook

Google Colab

Quick Intro to Google Collab

- Access Google Colab at colab.research.google.com



1. Upload datasets
2. Adding libraries
3. Run Cell
4. Table of contents
5. Menu Bar (upload/open a notebook)
6. Fine name (.ipynb)
7. Insert code cell
8. Insert Text cell
9. Cell (write your code)
10. Output (including error)
11. Clear Output

Reference: https://runestone.academy/ns/books/published/httlads/PythonReview/google_colab.html

Quick Intro to Python

0s

```
my_variable = 5       # writing your comment
if my_variable < 0:
  print("negative")
elif my_variable == 0:
  print("null")
else: # my_variable > 0
  print("positive")
```

```
positive
```

Quick Intro to Python Functions

0s

```
def square(x):
  return x ** 2

def multiply(a, b):
  return a * b

# Functions can be composed.
square(multiply(3, 2))
```

```
36
```

0s

```
def func1(a, b):
    return a / b


func1(4,0)
```

```
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
<ipython-input-12-571502bfeb5c> in <cell line: 5>()
      3
      4
----> 5 func1(4,0)

<ipython-input-12-571502bfeb5c> in func1(a, b)
      1 def func1(a, b):
----> 2     return a / b
      3
      4
      5 func1(4,0)

ZeroDivisionError: division by zero
```

SEARCH STACK OVERFLOW

Machine Learning Process

pandas

Data Manipulation and Analysis

matplotlib

Plotting Data and Results

NumPy

Matrices/vectors and Mathematical functions

pandas

Data Manipulation and Analysis

matplotlib

Plotting Data and Results

Quick Intro to Numpy

- Popular library for storing arrays of numbers and performing computations on them.
- Run faster, since most NumPy routines are implemented in C for speed.

```
# To use NumPy in your program, import like this
import numpy as np

array_a = np.array([1, 2, 3])
array_b = np.array([4, 5, 6])
array_a + array_b
```

```
array([5, 7, 9])
```

Quick Intro to Numpy

- Popular library for storing arrays of numbers and performing computations on them.
- Run faster, since most NumPy routines are implemented in C for speed.

0s

```
# To use NumPy in your program, import like this
import numpy as np

array_a = np.array([1, 2, 3])
array_b = np.array([4, 5, 6])
print('Addition')
print(np.add(array_a, array_b))
print('Subtraction')
print(np.subtract(array_a, array_b))
```

```
Addition
[5 7 9]
Subtraction
[-3 -3 -3]
```

Quick Intro to pandas

```
import pandas as pd
df = pd.DataFrame(
    {
        "Depression": ["Y", "N", "Y","Y", "N", "Y","Y", "N", "Y", "Y", "N", "N"],
        "Paleness": np.random.randn(12),
        "Breath rate": np.random.randn(12),
        "Restlessness": np.random.randn(12)
    }
)
print(df)
df.describe()
```

Quick Intro to pandas

```python
import pandas as pd
df = pd.DataFrame(
    {
        "Depression": ["Y", "N", "Y","Y", "N", "Y","Y", "N", "Y", "Y", "N", "N"],
        "Paleness": np.random.randn(12),
        "Breath rate": np.random.randn(12),
        "Restlessness": np.random.randn(12)
    }
)
print(df)
df.describe()
```

	Depression	Paleness	Breath rate	Restlessness
0	Y	0.869699	-1.989896	1.234126
1	N	0.025260	2.115986	2.229885
2	Y	-0.423437	0.361045	0.282993
3	Y	-0.331191	-0.228197	0.121887
4	N	-1.392590	-0.092830	0.142579
5	Y	0.449738	0.113998	-0.131692
6	Y	-1.768175	1.046479	0.045753
7	N	-0.132764	0.715555	0.051864
8	Y	1.150582	-0.335435	0.261132
9	Y	-0.042879	0.223646	1.931582
10	N	1.580644	-1.997885	1.237896
11	N	0.092382	0.275313	0.111543

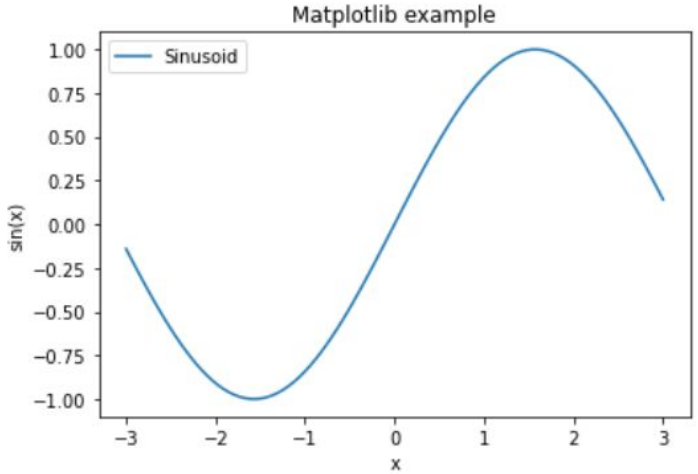
	Paleness	Breath rate	Restlessness
count	12.000000	12.000000	12.000000
mean	0.006439	0.017315	0.626629
std	0.961239	1.147113	0.812679
min	-1.768175	-1.997885	-0.131692
25%	-0.354252	-0.255007	0.096623
50%	-0.008809	0.168822	0.201855
75%	0.554728	0.449672	1.235068
max	1.580644	2.115986	2.229885

Quick Intro to matplotlib

```
[ ]  from matplotlib import pyplot as plt

     x_values = np.linspace(-3, 3, 100)

     plt.figure()
     plt.plot(x_values, np.sin(x_values), label="Sinusoid")
     plt.xlabel("x")
     plt.ylabel("sin(x)")
     plt.title("Matplotlib example")
     plt.legend(loc="upper left")
     plt.show()
```



Some examples of Figs generated using Matplotlib