Twitter Thread by Pratham Prasoon



Pratham Prasoon
@PrasoonPratham

This is a step-by-step guide on getting started with Python as a total beginner.

(includes free resources and everything else you need to get started)

■■

Before we begin, I want to congratulate you on your decision to learn how to code using Python.

I still remember how I wrote my first piece of code 6 years and all the amazing and cool things I've been able to do with it ever since.

(2 / 24)

Make sure to keep in mind that it is probably best for you to keep your expectations in check.

Don't expect to make AAA games or state of the art machine learning models in a week.

(3 / 24)

Programming is not something that you can learn in a single week, it takes consistent effort and dedication over time to get good at it.

With all that being said, let's dive straight in.

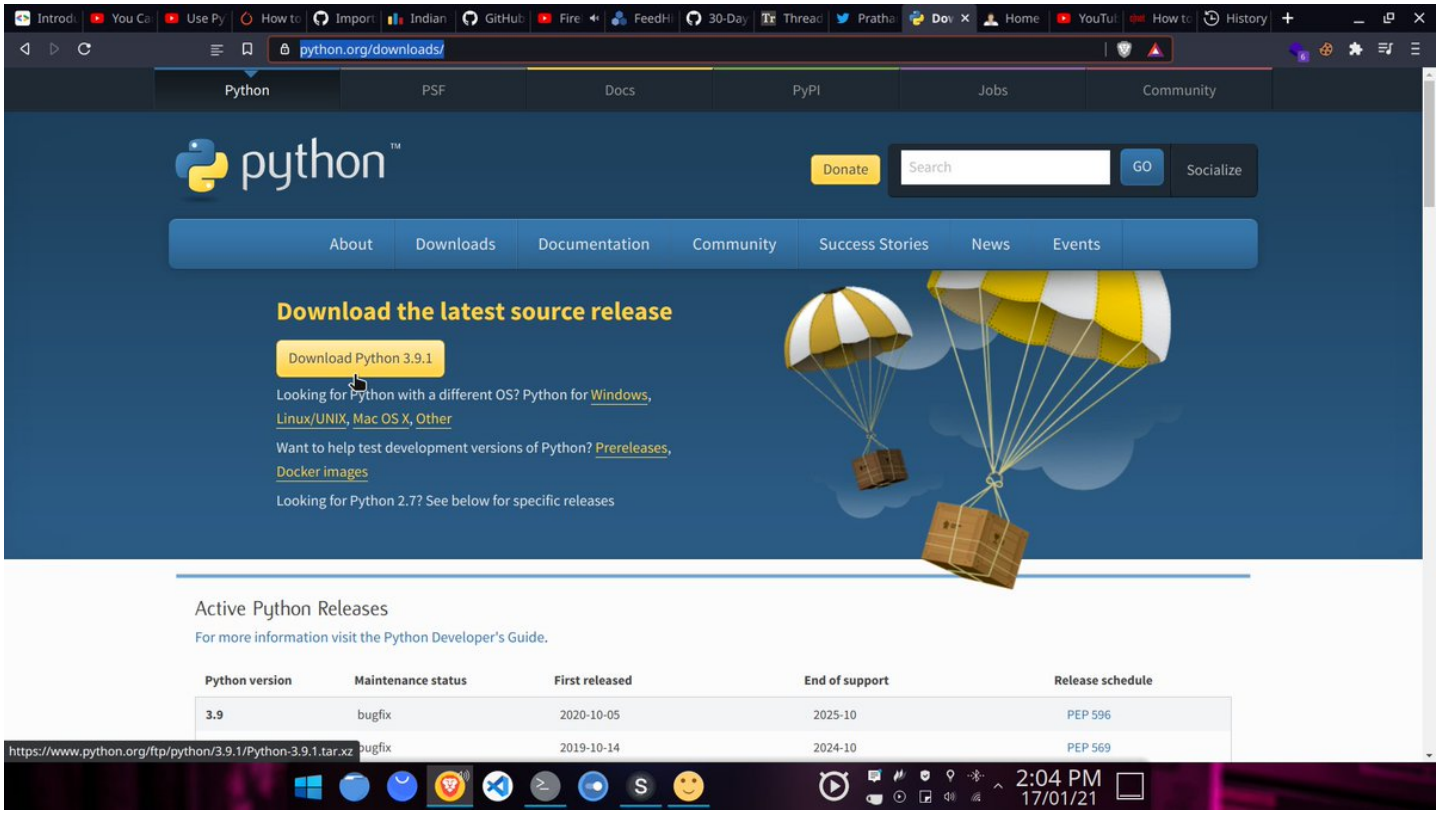
(4 / 24)

In order to write Python code, you'll need to install Python on your system.

Linux and macOS users can skip this step because they come pre-installed with Python.

Download link: https://t.co/KSZ4Qd6CNk

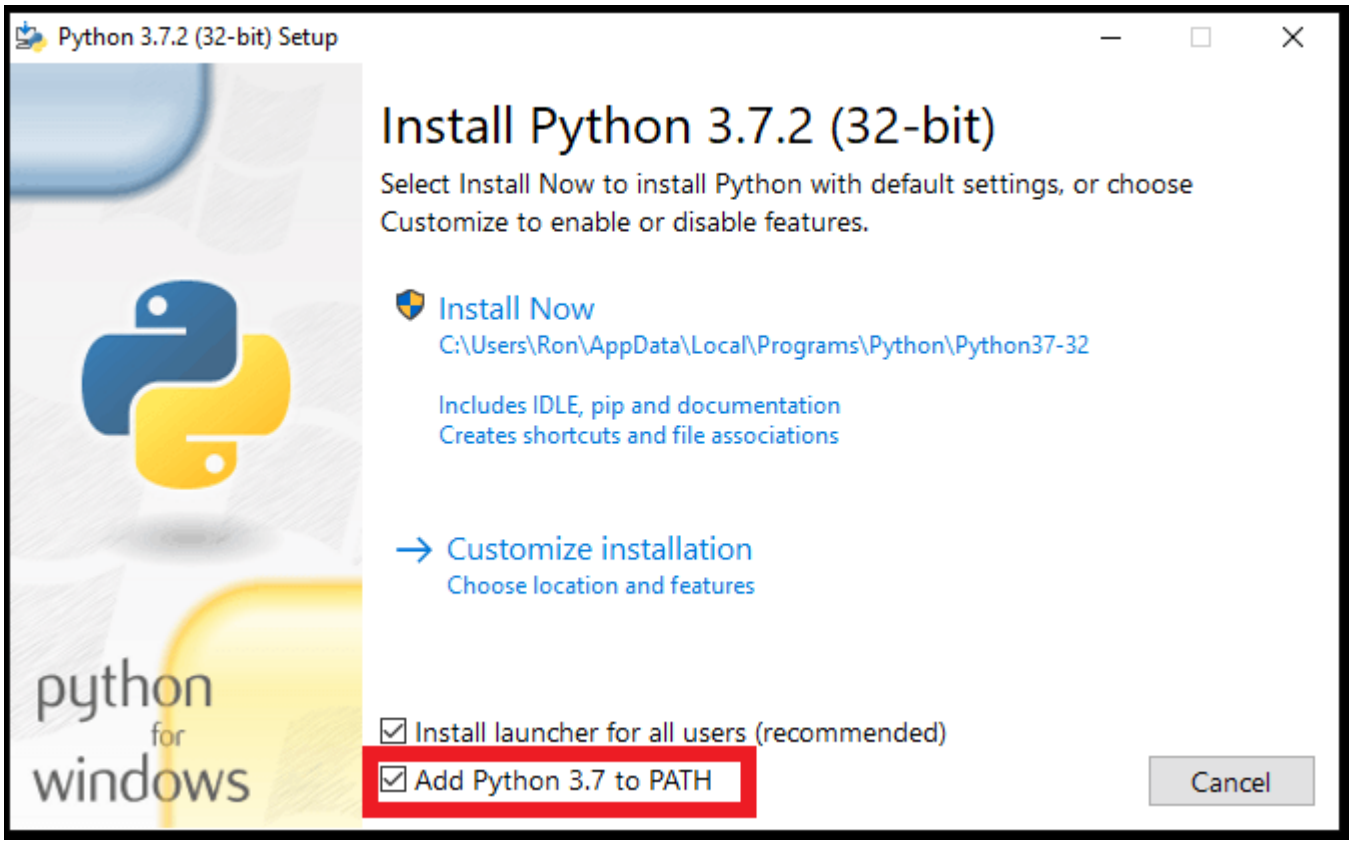
(5 / 24)



Click on the .exe file and follow the instructions.

Make sure to Add python to path by checking this option ■

(6 / 24)



Now you need to install a place where you can write your Python code, just like how you write your essays in Word or Google docs.

We'll be installing VS-Code, one of the best code editors out there and it's free!

(7 / 24)

Fortunately, Microsoft has this wonderful guide that'll help you out.

■https://t.co/0cN2JyM2di

(8 / 24)

In case you are unable to set up Python on your own system then I recommend using repl, a great way to write and run Python code without any hassle.

■https://t.co/eijcOcz42c

(8 / 24)

Here are the topics you should be focusing

- Printing statements
- Variables
- Operators
- Conditions
- Functions
- Loops

Let's take a closer look.

(9 / 24)

Printing statements

You can 'print' or output in Python using the print() function.

print('Hello World') will give you an output of `Hello World`

(10 / 24)

Variables

These store certain values that can change.

For example, You can declare a variable 'x' with a certain value.

x = 9 [Here 9 is assigned to varaiable x]
y = 'Hello World' ['Hello World' is assigned to y]

x = 8 [x has been updated to a new value which is 8]

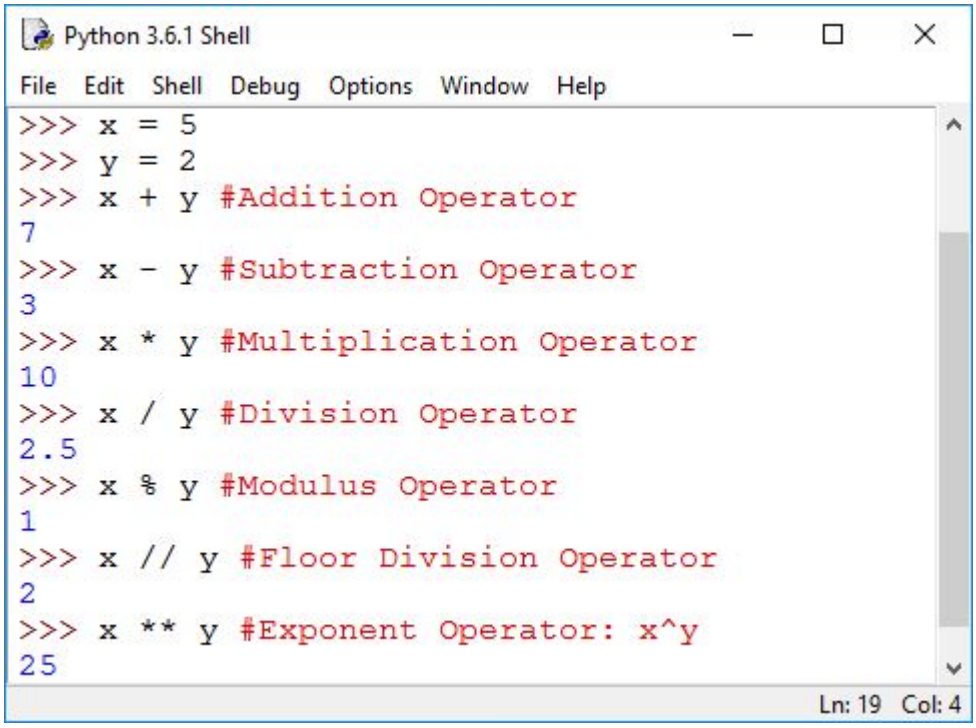
(11 / 24)

Operators

These carry out arithmetic operations in Python

+ : Addition
- : Subtraction
* : Multiplication
/ : Divide
% : Modulus (Remainder after dividing)
** : Exponents

(12 / 24)



Conditions

They look like this ■

a = 33
b = 200
if b > a:
print("b is greater than a")
else:
pass

Think of it as telling Python to do something based on conditions: if this is true, do this; else do something different.

(13 / 24)

Functions

These are basically blocks of code that can be run when you call them. This helps us write code more efficiently.

In Python, you can make functions using the def keyword.

```
def myFunction():
print("Hello")
myFunction()
```

(14 / 24)

Loops

These essentially run a piece of code over and over again until a certain condition is met.

There are 2 types of loops in Python:
- While loops
- For loops

Let's take a look at them.

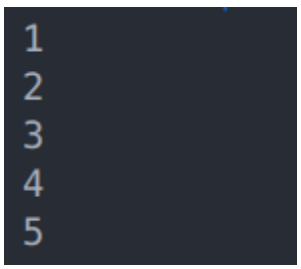
(15 / 24)

Here's what While Loops look like.

```
i = 1
while i < 6: //While i is less than 6
print(i) // do
i += 1 //this
```

The output is :

(16 / 24)

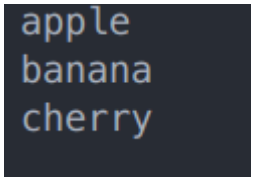


Here's a For loop

```
fruits = ["apple", "banana", "cherry"]
for fruit in fruits:
print(x)
```

The output:

(17 / 24)



The for loops can be a bit tricky to understand, let me try to break it down for you.

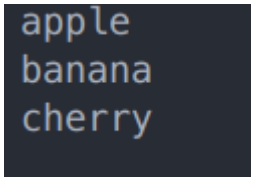
'fruit' is the above loop is a temporary variable. It gets a value from the list called 'fruits' starting from "apple" then to "banana" and then "cherry".

(18 / 24)

Each time it gets a value, it is printed.

This explains why we get this output ■

(19 / 24)



All that I explained before was to give you a taste of the basics of Python, having strong fundamentals at this stage is very important.

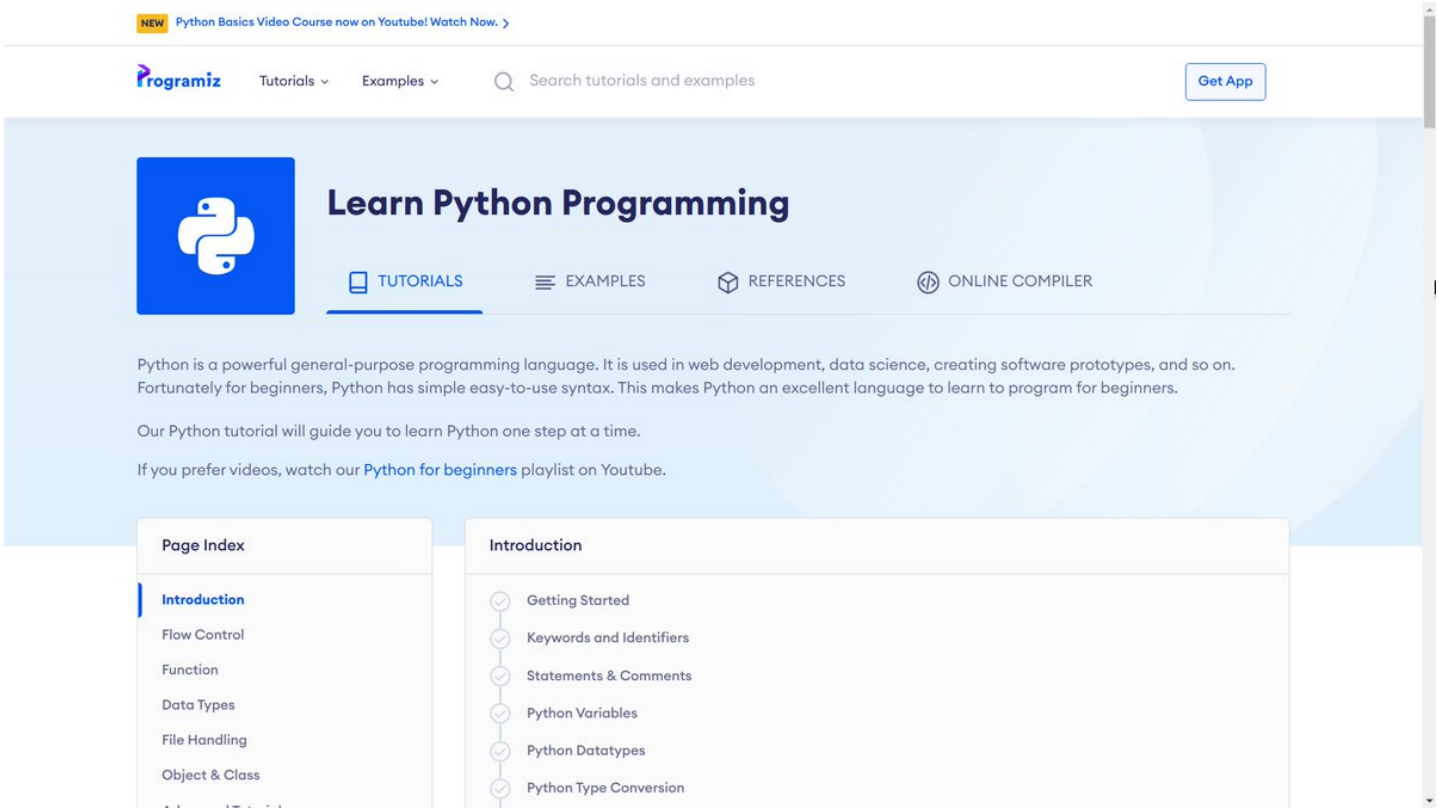
I have even more threads for further steps in Python coming up, stay tuned.

(20 / 24)

These are some of the resources I would recommend to you for further learning■

Incredible text-based tutorials: https://t.co/qRvZNItSM2

(21 / 24)



Objected Oriented Programming (advanced) : https://t.co/Bpxrjlv4MQ

This course on Traversy Media's channel is a great next step for taking your python skills to the next level!

(22 / 24)



Python Documentation: https://t.co/5DX75jwnXQ

This is probably the most underrated resource for learning python.

(23 / 24)

