

programming with

# PYTHON

**Abhay Mane**

Project Engineer

Big Data Analytics & Machine Learning Team

C-DAC Bengaluru



abhaym@cdac.in



abhaymane93



# OUTLINE

0

Basics

3

Loops

6

Object Oriented  
Programming

1

Data Types

4

Functions

7

Exception  
Handling

2

Conditional  
Statements

5

File I/O

8

Exploratory  
Data Analysis



# MODULE EVALUATION

CCEE

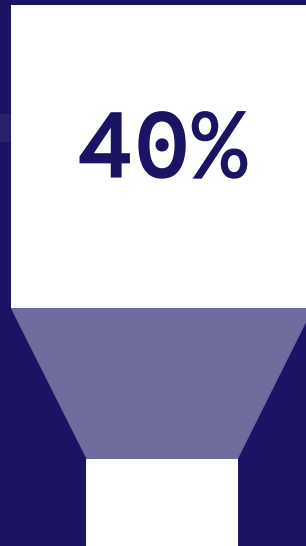
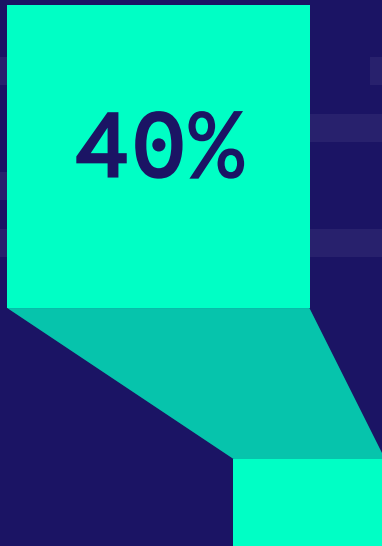
40%

Lab Exam

40%

Internal  
Exam

20%





# Introduction



# INTRODUCTION

Python is **general purpose**,  
**high level**, **interpreted**  
**language** with **easy** syntax  
and **dynamic** semantics.

Created by  
**Guido Van Rossum** in  
**1989**

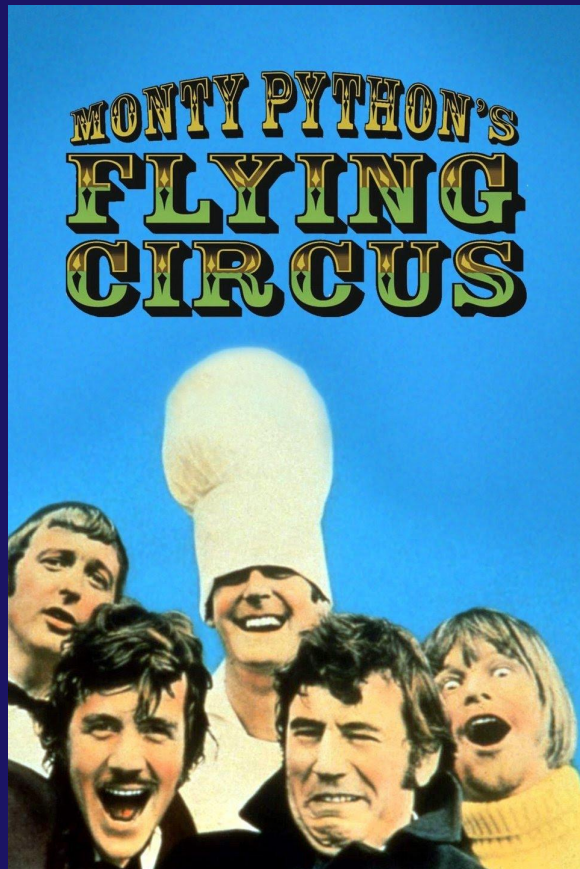




## WHY THE NAME?

Python gets its name from the 1970s British TV comedy series, *Monty Python's Flying Circus*.

According to Python folklore, Guido van Rossum was watching reruns of the show at about the same time he needed a name for a new language he was developing.





# FEATURES OF PYTHON

1

Simplicity

2

Open source

3

Embeddable &  
extensible

4

Interpreted

5

Object  
Orientation

6

Huge Libraries

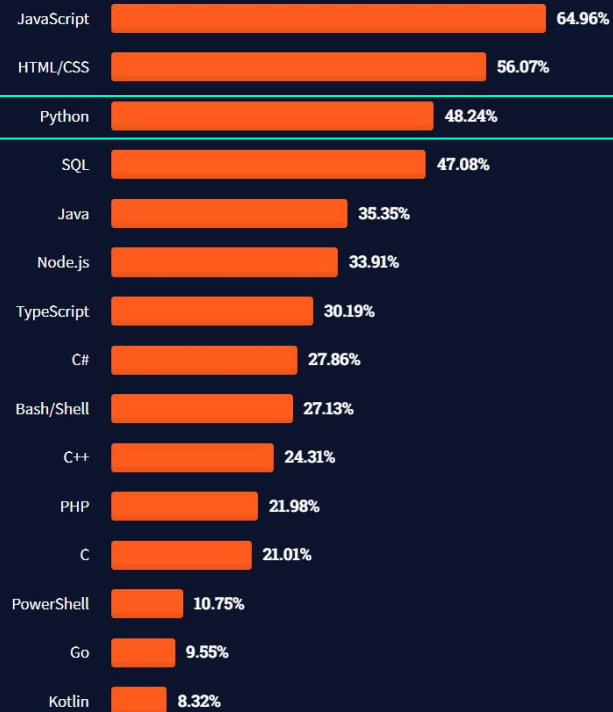


# Popularity



# 2021 Developer Survey

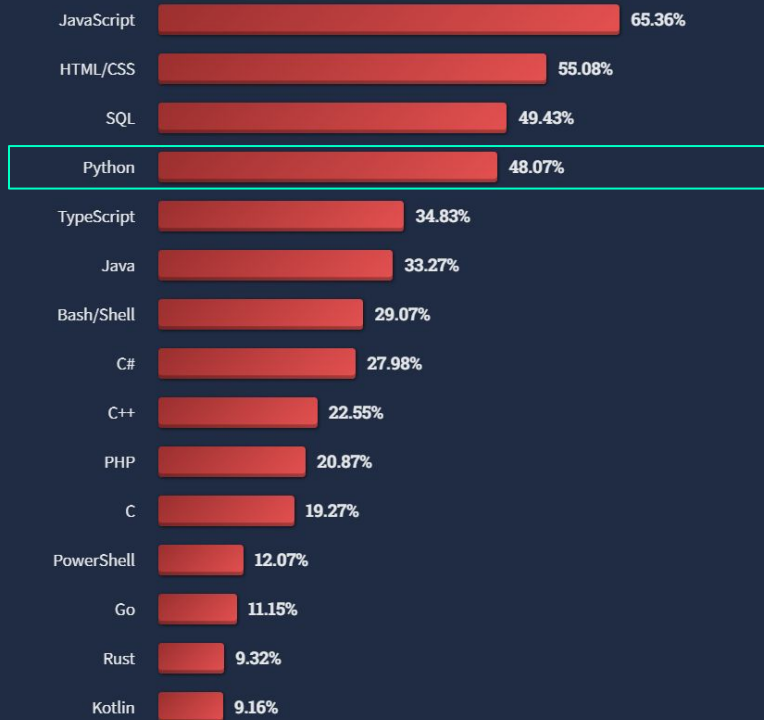
83,052 responses



71,547 responses



## 2022 Developer Survey





# Install Python



# PYTHON VERSION

		Maintenance status	First released	End of support
	3.10	bugfix	2021-Oct-04	2026-Oct
✓	3.9	security	2020-Oct-05	2025-Oct
	3.8	security	2019-Oct-14	2024-Oct
	3.7	security	2018-June-27	2023-Jun-27
	2.7	end-of-life	2010-July-03	2020-Jan-01

\* As on 2022-Oct-17

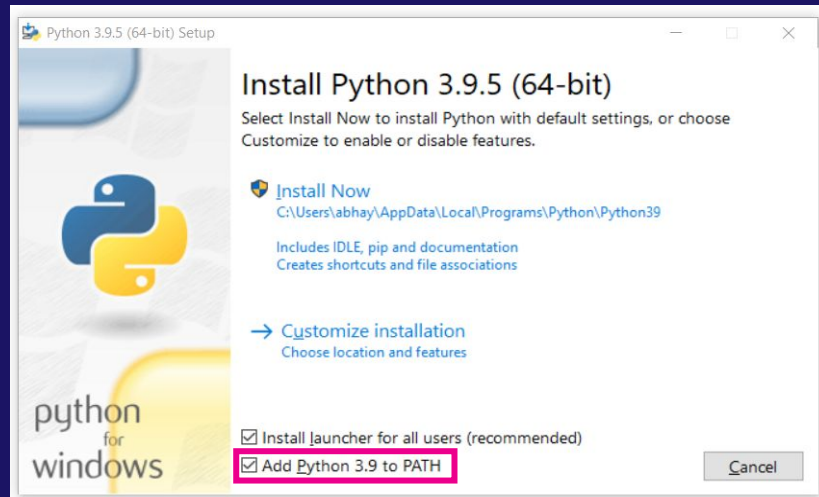


# INSTALL PYTHON

## Windows

Download installer from:

<https://www.python.org/downloads/release/python-390/>



Select the checkbox to add Python to PATH



# INSTALL PYTHON

## Linux

# Ubuntu

```
$ sudo apt-get update
```

```
$ sudo apt-get install python3.9
```

\* root permissions required

The Ubuntu logo, featuring the word "ubuntu" in white lowercase letters next to a white circular icon with a gear-like pattern, is centered on an orange rectangular background.



# INSTALL PYTHON

## Linux

```
# Open SUSE
```

```
# install open-ssl
```

```
# install python
```

\* root permissions required





# INSTALL PYTHON

## Anaconda

Quickly get started with a easy-to-use  
Python package and environment manager  
for educational and research use.







# INSTALL PYTHON

## Miniconda

Miniconda is a free minimal installer for conda. It is a small, bootstrap version of Anaconda that includes only conda, Python, the packages they depend on, and a small number of other useful packages





# INSTALL PYTHON

## Miniconda

# Installing Miniconda on Linux

# Download miniconda distribution

[https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86\\_64.sh](https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh)





# INSTALL PYTHON

## Miniconda

# After downloading

# Install using following command on terminal

```
$ bash Miniconda3-latest-Linux-x86_64.sh -b
```

```
$ echo 'export PATH=$HOME/miniconda3/bin:$PATH' >> ~/.bashrc
```

```
$ source ~/.bashrc
```





# INSTALL PYTHON

## Miniconda

# Check after installing

\$ which python

# should show \$HOME/miniconda3/bin/python

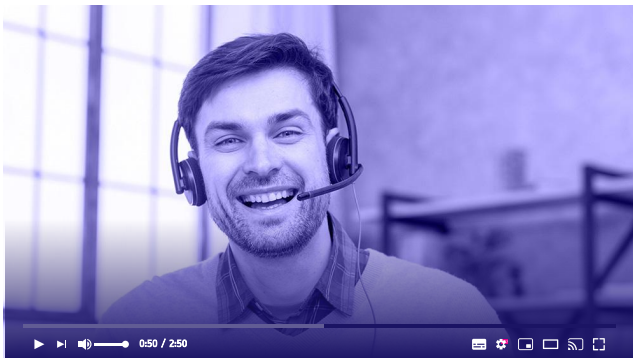
\$ python -V

# should show python 3.9.x





# Your First Python Program



HELLO WORLD!



```
#include <stdio.h>
int main()
{
    printf("Hello World!");
    return 0;
}
```



```
class HelloWorld
{
    public static void main(String[] args)
    {
        System.out.println("Hello World!");
    }
}
```



```
print("Hello World!")
```



# YOUR FIRST PYTHON PROGRAM

## Hello World

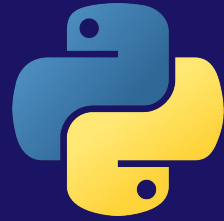
Q. Write a python script to display Hello World on the screen.

# Use vi editor for creating python script named hello.py

```
$ vi hello.py
```

# Run it using following command

```
$ python hello.py
```



# Virtual Environment





# VIRTUAL ENVIRONMENT

At its core, the main purpose of Python virtual environments is to create an isolated environment for Python projects.

This means that each project can have its own dependencies, regardless of what dependencies every other project has.

**Recommended** to create virtual environment and install required packages in it, whenever starting a new project.



✓ Lab

Linux

# VIRTUAL ENVIRONMENT

Always create a virtual environment in your project directory using following command:

```
$ python -m venv <name_for_venv>
```

A directory with name of the environment will be created.

To start virtual environment, run following command:

```
$ source <name_for_venv>/bin/activate
```

The name of virtual environment will appear at the beginning of the terminal prompt.

Check python using *which* command; it should show python from your virtual environment directory

Deactivate virtual environment:

```
$ deactivate
```



## Windows

# VIRTUAL ENVIRONMENT

Always create a virtual environment in your project directory using following command:

```
$ python -m venv <name_for_venv>
```

A directory with name of the environment will be created.

To start virtual environment, run following command:

```
$ <name_for_venv>\Scripts\activate
```

The name of virtual environment will appear at the beginning of the cmd prompt.

Deactivate virtual environment:

```
$ deactivate
```



# VIRTUAL ENVIRONMENT

After activating virtual environment install individual packages using:

```
$ pip install <package_name>
```

Install multiple packages using requirements file:

```
$ pip install -r requirements.txt
```



IDE

# IDE



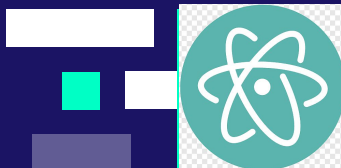
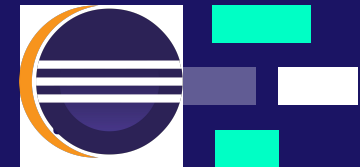
PyCharm

Spyder



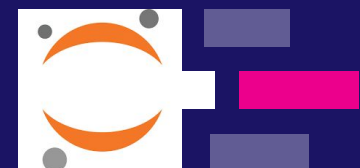
Visual Studio  
Code

Eclipse



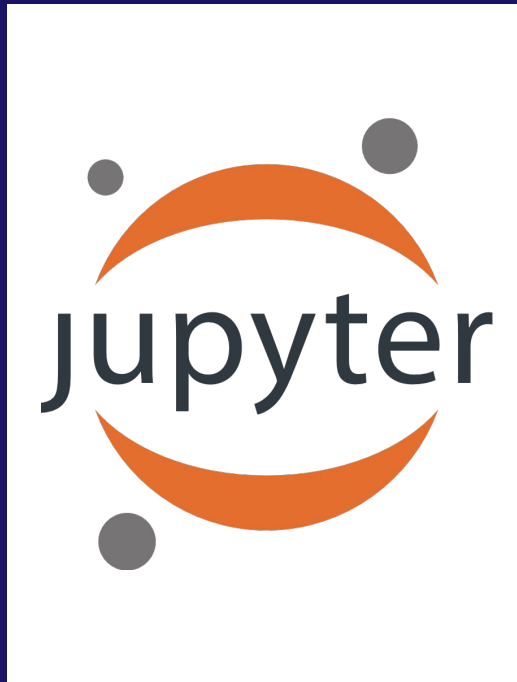
Atom

✓ Jupyter Lab





# Jupyter Lab



<https://jupyter.org/>

JupyterLab is a web-based interactive development environment for Jupyter notebooks, code, and data.

JupyterLab is flexible: configure and arrange the user interface to support a wide range of workflows in data science, scientific computing, and machine learning.



✓ Lab

# Install Jupyter Lab



Activate the virtual environment

Then install jupyter lab:

```
$ pip install jupyterlab
```





# Jupyter Notebook Keyboard Shortcuts



Key	Description
Enter	Enter edit mode
Esc	Leave edit mode and enter command mode
Shift + Enter	Run Cell and move control to next cell
Ctrl + Enter	Run cell
Ctrl + S	Save Notebook
A	Create new cell above current cell
B	Create new cell below current cell
DD	Delete Cell
Shift + M	Merge Cells
Shift + Ctrl + -	Split Cell
M	Change cell to markdown
Y	Change cell to code



# LAB

Steps to follow in lab  
session every day

## ○ Activate Virtual Environment

\$ source <venv\_name>/bin/activate

## ○ Start Jupyter Lab

\$ jupyter-lab

## ○ Code in Jupyter Lab in browser

Earth is where we live on

## ○ Shutdown Jupyter Lab

Using shutdown option in the File menu

## ○ Deactivate Virtual Environment

\$ deactivate