

PYTHON

Name derived from famous BBC comedy show namely Monty Python's flying circus

Introduction

- ❖ Python was created by Guido Van Rossum
- ❖ The language was released in February 1991
- ❖ Python got its name from a BBC comedy series from seventies-
“Monty Python”s Flying Circus”
- ❖ Python can be used to follow both Procedural approach and
Object Oriented approach of programming
- ❖ It is free to use
- ❖ Python is based on or influenced with two programming
languages:
 - ❖ ABC language [replacement of BASIC]
 - ❖ Modula-3

Features which make Python so popular

- ❖ Easy to use Object oriented language
- ❖ Expressive language
- ❖ Interpreted Language
- ❖ Its completeness
- ❖ Cross-platform Language
- ❖ Free and Open source
- ❖ Variety of Usage / Applications

Easy to Use

- Python is compact and very easy to use object oriented language with very simple syntax rules. It is very high level language and thus very-very programmer -friendly

Expressive Language

- More capable to express code's purpose than many othr language like – swapping of two numbers:

In C++	In Python
<pre>int a=2, b=3, tmp; tmp=a; a=b; b=tmp;</pre>	<pre>a,b=2,3 a,b=b,a</pre>

Interpreted

- It is interpreted not compiled, thus executes code line by line and it makes python easy-to-debug and suitable for beginners and advanced users

Completeness

- When you install Python, you get everything i.e. you don't need to install additional libraries. All functionality is available with Python additional library. Features like web-pages, database connectivity, GUI features are available in Python standard library. Python follows – “Battery included” philosophy

Cross-Platform

- Python can run equally well on variety of platforms – Windows, Linux/UNIX, Macintosh, supercomputers, smart phones. Python is portable language.

Free and Open Source

- It is freely available i.e. without any cost can be downloaded from www.python.org.
- It is also open source i.e. you can modify, improve/extend an open-source software

Variety of Usage/Applications

- Python is being used in many diverse fields/applications, some of which are:
 - Scripting
 - Web Applications
 - Game Development
 - System Administrations
 - Rapid Prototyping
 - GUI Programs
 - Database Applications

Limitations (Minus) of Python

- ❖ Not the fastest language
- ❖ Lesser Libraries than C, Java, Perl
- ❖ Not Strong on Type-binding
- ❖ Not Easily convertible

Not the fastest language

- Because of its interpreted nature Python is not fast as compare to compiled language. Python is first compiled into an internal byte-code which is then executed by Python interpreter.

Lesser Libraries than C, Java, Perl

- Python offers library support for almost all computing programs, but its library is still not competent with languages like C, java as they have larger collection available.

Not Strong on Type-binding

- Python is not very strong on data type checking i.e. “Type-mismatch” issues. For example if you declare variable as integer but later store a string value in it, Python will not raise any error.

Not Easily Convertible

- Because of its lack of syntax, Python is easy to program but it becomes disadvantage when it comes to translate python program to another language.

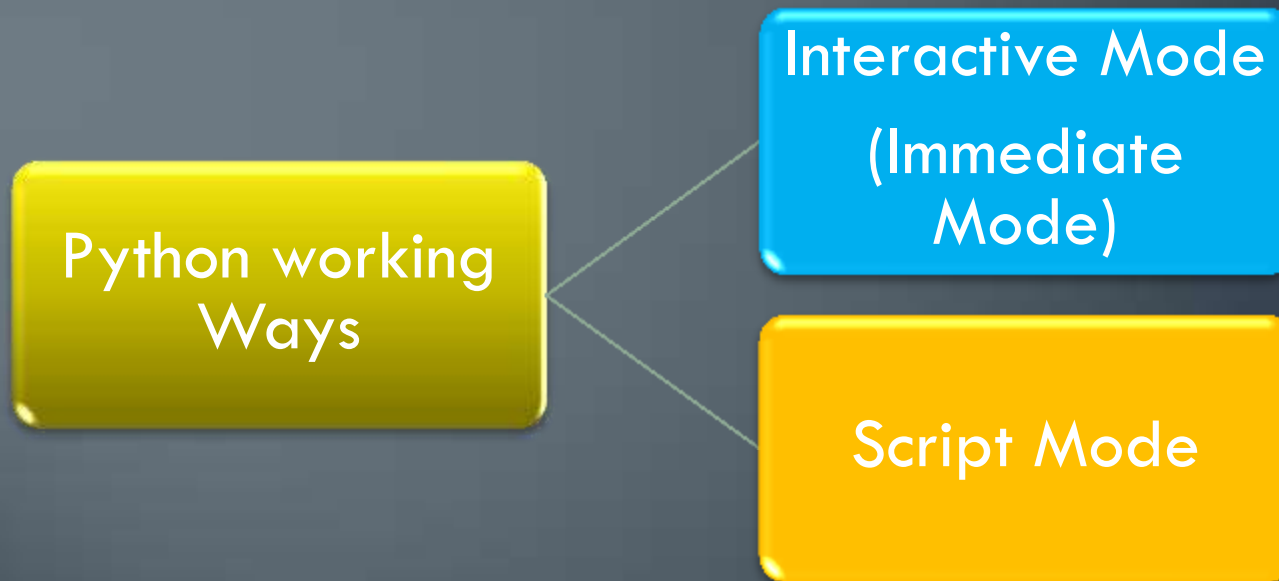
Working In Python

Before we start working on Python we need to install Python in our computer. There are multiple distributions available today:

- ❖ Default Installation available from www.python.org is called Cpython installation and comes with python interpreter, Python IDLE(Python GUI) and Pip(package installer)
- ❖ ANACONDA Python distribution is one such highly recommended distribution that comes with preloaded many packages and libraries(NumPy, SciPy, Panda etc)
- ❖ Other Popular IDEs like Sypder, PytCharm, etc. Spyder IDE is available as a part of ANACONDA.

Working Modes in Python

After Python installation we can start working with python. In Python we can work in 2 ways:



Working in Default CPython Distribution

The default installation of CPython comes with Python Interpreter, Python IDLE and pip(package installer). To work in either interactive mode or script we need to open **Python IDLE**

To work in interactive mode, follow the process given below:

(i) Click start button → All Programs → Python 3.6.x → IDLE

Or

Click start button → All Programs → Python 3.6.x → Python
(command line)

Working in Interactive Mode(Python IDLE)

- ❖ Interactive modes – one command at a time
- ❖ Python executes the given command and gives the output.
- ❖ In interactive mode we type command at IDLE prompt (`>>>`)
- ❖ For e.g if you type `20 + 30` in from of IDLE prompt
`>>> 20 + 30` (command give by user)
`50` (**output given by python**)
- ❖ From the above example you can see than at `>>>` we have to just give the command to execute and python we execute it if it is error free otherwise gives an error.

```
Type "copyright", "credits" or "li
>>> print "hello world!"
hello world!
>>> print (10+20)
30
>>> print (20-10)
10
>>> print (10 * 20)
200
>>> print (20/10)
2
>>> a=30
>>> b=40
>>> c=a+b
>>> print (c )
70
>>> print ("Hello")
```

Working in Script Mode(Python IDLE)

- ❖ Script Mode – multiple commands can be saved in a file as a program and then we can execute the entire program
- ❖ we type Python program in a file and then use the interpreter to execute the content from the file.
- ❖ Working in interactive mode is convenient for beginners and for testing small pieces of code, as we can test them immediately. But for coding more than few lines, we should always save our code so that we may modify and reuse the code

Creating Script/Module/Program File

❖ In Python IDLE :

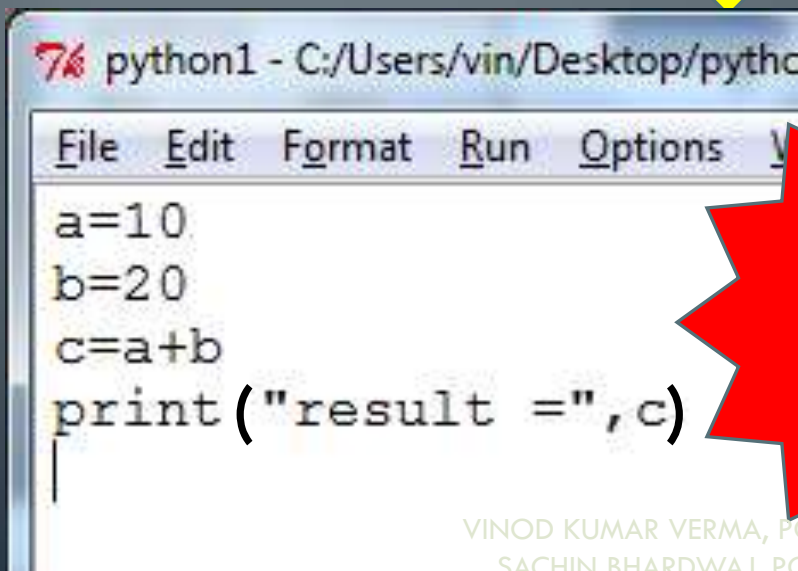
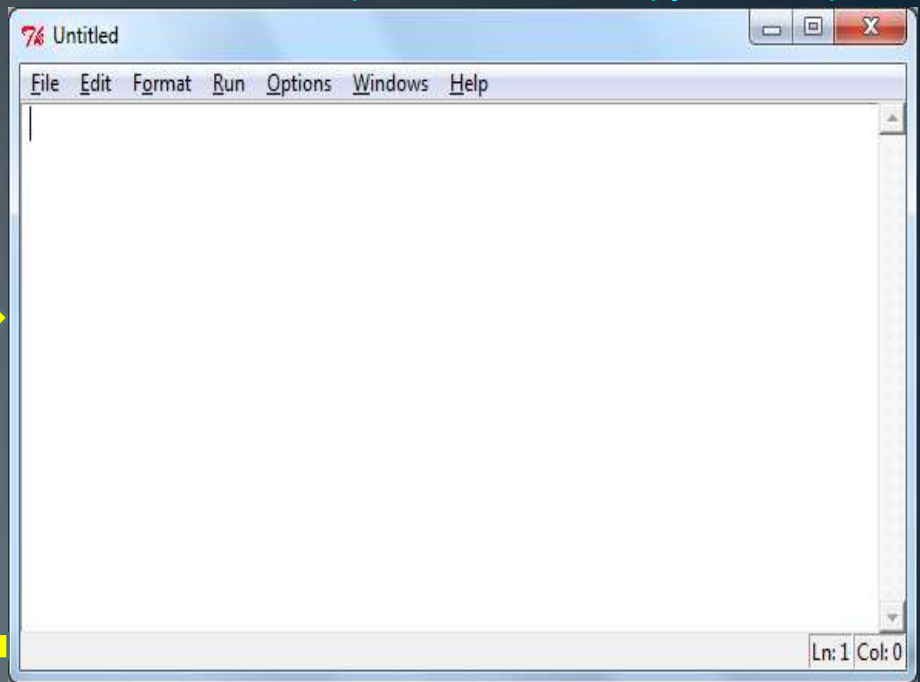
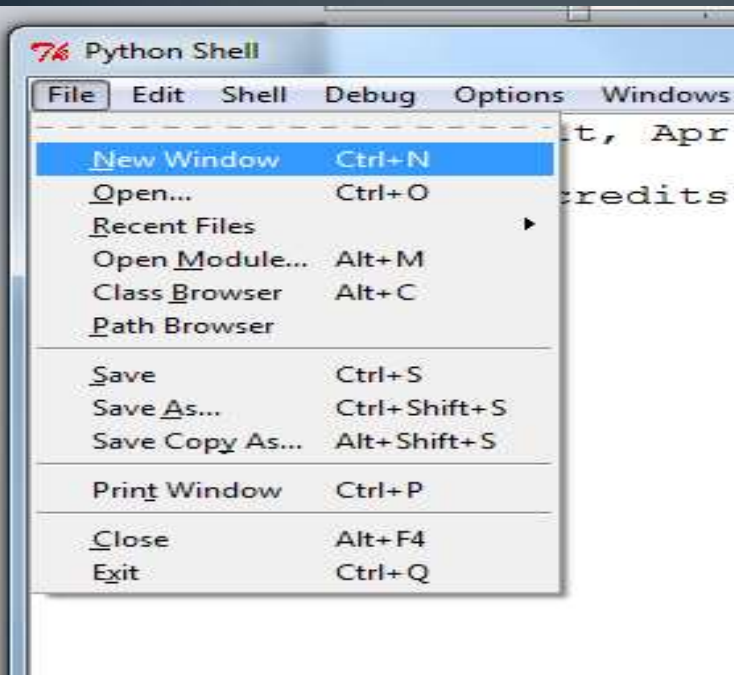
❖ Click File → New

❖ In new window type the commands you want to save in program

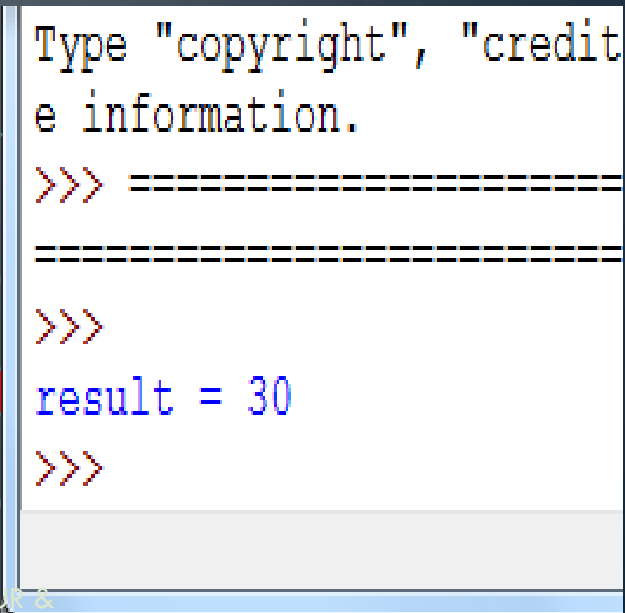
❖ For example:

❖ `print("Hello World!")`

❖ File → Save to save file. Give extension **.py** to execute it as python script



Save the program and press F5





Sign in to Anaconda Cloud

Home

Environments

Learning

Community

Documentation

Developer Blog

Feedback



Applications on base (root)

Channels

Refresh



jupyterlab

0.32.1

An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Launch



notebook

5.5.0

Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Launch



qtconsole

4.3.1

PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

Launch

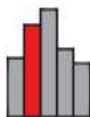


spyder

3.2.8

Scientific PYTHON Development Environment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

Launch



glueviz

0.13.3

Multidimensional data visualization across files. Explore relationships within and among related datasets.



orange3

3.13.0

Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows



rstudio

1.1.423

A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.



vscode

1.25.1

Streamlined code editor with support for development operations like debugging, task running and version controls

Go to Settings to activate Windows.

Spyder (Python 3.6)

File Edit Search Source Run Debug Consoles Projects Tools View Help

Editor - C:\Users\Lab Admin\spyder-py3\temp.py

```
1 # -*- coding: utf-8 -*-
2 """
3 Spyder Editor
4 This is a temporary script file.
5 """
6
7
8
```

Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our [tutorial](#)

Variable explorer File explorer Help

IPython console

Console 1/A

Python 3.6.5 |Anaconda, Inc.| (default, Mar 29 2018, 13:23:52) [MSC v.1900 32 bit (Intel)]
Type "copyright", "credits" or "license" for more information.

IPython 6.4.0 -- An enhanced Interactive Python.

In [1]:

Activate Windows
Go to Settings to activate Windows.

IPython console History log

Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 8 Column: 1 Memory: 80%

VINOD KUMAR VERMA, PGT(CS), KV OEF KANPUR &
SACHIN BHARDWAJ, PGT(CS), KV NG. 1 TEZPUR

Type here to search

Desktop 11:31 AM 16-Jul-18

Analyzing Script and output

Type the below code in script file:

```
#My First Program  
print("Hello World!")
```

When you execute the script file output will be only
Hello World

Note: the first line starts from # which makes line as a comment i.e. only for programmer information not for interpreter. Comments are ignored by interpreter

Understanding print()

To print or display output Python 3.x provides print() function.

Examples:-

```
print("Welcome to Python")
```

it will print Welcome to Python

```
print('Python rocks!')
```

it will print Python rocks!

Note: Python allows both single quotation and double quotation to enclose string. In any case both opening and closing must be same.

```
print("Look carefully")
```

it will give error because both opening and closing quotes are different.

Points to remember...

Note:

- 1) In Python 2.x print is a statement, not a function**
- 2) Python is case-sensitive i.e. it will treat small letter and capital letter as two different entities like m & M will be treated differently in Python.**
- 3) We can use up arrow (↑) and down arrow (↓) to select previous command and press enter to recall and execute.**
- 4) print statement is required only while working with script mode otherwise in interactive mode values can be printed without print statement**

What will be the output of following code:

```
#This is my first program
#sample of basic statements
#print("Watch out")
print("Johny Johny Yes Papa")
print("No Papa")
```

(a)
This is my first program sample of basic
statements Watch out
Johny Johny Yes Papa
No Papa

(b)
Watch out
Johny Johny Yes Papa
No Papa

(c)
Johny Johny Yes Papa
No Papa

(d)
This is my first program
Johny Johny Yes Papa
No Papa

Which of the following are not valid strings in Python:

(a) `“Sayonara”`

(b) `‘Sayonara’`

(c) `“Sayonara’`

(d) `‘Sayonara”`

(e) `{Sayonara}`

Just a Minute...

- Who developed Python Programming Language?
- Is Python Object Oriented Language?
- Python is a high level interpreted language. What does it means
- What does cross-platform means?
- Python is free and open source. What do you understand by this feature?
- What is the difference in interactive mode and Script mode in Python?