INTRODUCTION WITH PYTHON

Python History and Versions
- Python laid its foundation in the late 1980s.
- The implementation of Python was started in the December 1989 by Guido Van Rossum at CWI in Netherland.
- In February 1991, van Rossum published the code (labeled version 0.9.0) to alt.sources.
- In 1994, Python 1.0 was released with new features like: lambda, map, filter, and reduce.
- Python 2.0 added new features like: list comprehensions, garbage collection system.
- On December 3, 2008, Python 3.0 (also called "Py3K") was released. It was designed to rectify fundamental flaw of the language.
- ABC programming language is said to be the predecessor of Python language which was capable of Exception Handling and interfacing with Amoeba Operating System.
- Python is influenced by following programming languages:
- ABC language.
- Modula-3

Python Version List

Python programming language is being updated regularly with new features and supports. There are lots of updations in python versions, started from 1994 to current release.

A list of python versions with its released date is given below.

Python Version	Released Date
Python 1.0	January 1994
Python 1.5	December 31, 1997
Python 1.6	September 5, 2000
Python 2.0	October 16, 2000
Python 2.1	April 17, 2001
Python 2.2	December 21, 2001
Python 2.3	July 29, 2003
Python 2.4	November 30, 2004
Python 2.5	September 19, 2006
Python 2.6	October 1, 2008
Python 2.7	July 3, 2010
Python 3.0	December 3, 2008
Python 3.1	June 27, 2009
Python 3.2	February 20, 2011
Python 3.3	September 29, 2012

INTRODUCTION WITH PYTHON

Python 3.4	March 16, 2014
Python 3.5	September 13, 2015
Python 3.6	December 23, 2016
Python 3.7	June 27, 2018

Installation
- Visit the link https://www.python.org/downloads/ to download the latest release of Python.
- Double-click the executable file which is downloaded; the following window will open. Select Customize installation and proceed.
- Now, try to run python on the command prompt. Type the command python in case of python2 or python3 in case of python3.

Environment Variables
- To set the path of python, we need to the right click on "my computer" and go to Properties → Advanced → Environment Variables.
- Add the new path variable in the user variable section.
- Type PATH as the variable name and set the path to the installation directory of the python.
- Now, the path is set, we are ready to run python on our local system. Restart CMD, and type python again. It will open the python interpreter shell where we can execute the python statements.

Using python in CMD
Create a "*.py" file by using notepad which contains python code.
Move to the directory containing python file by using cmd.
Write the name of file with extension and press Return.
Your program will start executing.

IDLE(Integrated Development-Learning Environment)
IDLEs are additional software that get automatically installed with Python Shell.
It provides features of IDE as well as Python shell means I can compile and interpret the python codes.
To run this,
> Use Search bar and search for IDLE.
> Press Enter for the 1st result.

Editing Python files(*.py)
Move to your local directory where your python file is located.
Select the file and right click on the file and open it with Notepad.
Here you can modify your python codes as per your need.

Python Documentation
To make your program more readable we add comments in it.
To use comments **Hashtag(#)** is used. e.g., #This is my first program.
A string can also work as a comment(also Multiline comment) until it isn't assigned to any varriable.

Dynamic Types
In python variables can change its data type as per its recent assignment.
For example

```
a=5
print(type(a))
a="Hello"
print(type(a))
```

After Compilation

```
<class 'int'>
<class 'str>
```

Keywords(Also known as Reserved words)

In Python, keywords are case sensitive.
There are 33 keywords in Python 3.7. This number can vary slightly in the course of time.
All the keywords except True, False and None are in lowercase and they must be written as it is. The list of all the keywords is given below.

False	class	finally	is	return
None	continue	for	lambda	try
True	def	from	nonlocal	while
and	del	global	not	with
as	elif	if	or	yield
assert	else	import	pass	
break	except	in	raise	

Naming Conventions
1. Statement Termination- Python does not uses any specific character to terminate any statement.
2. Max. Line Length- Not more than 79 chars.
3. Lines & indentation- As like in C/C++ or java where curly braces were used here indentation is used.
   Example
   if true:
           print("Statement")
   else:
           print("Another statement")
4. Blank Lines- Add 2 blank lines to separate btw top level definition , one line btw method/function definition.
5. Avoid multiple statement on one line- By using semicolon(;) multiple statement can be put into one line only.
6. Case Sensitive - It is case sensitive language means a & A are two different variables.