History of Python

- The Python programming language was conceived in the late 1980s and its implementation was started in December 1989 by Guido van Rossum at CWI in the Netherlands.
- Python 2.0 was released on October 16, 2000, with many major new features, including garbage collector for memory management and support for Unicode.
- Python 3.0, a major, backwards-incompatible release, was released on December 3, 2008.
- It has a large and comprehensive standard library.
- The syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++ or Java.
- Python interpreters are available for many operating systems, allowing Python code to run on a wide variety of systems.
- ABC is an imperative general-purpose programming language and programming environment developed at CWI, Netherlands by Leo Geurts, Lambert Meertens, and Steven Pemberton. It is interactive, structured, high-level, and intended to be used instead of BASIC, Pascal, or AWK.
- Python has come to stand as one of the most important and most popular open source programming languages being used today.
- Python was born out of the ABC language, a terminated project of the Dutch CWI research institute that van Rossum worked for, and the Amoeba distributed operating system. When Amoeba needed a scripting language, van Rossum created Python.

Compiler-

A compiler is a special program that processes statements written in a particular programming language and turns them into machine language or "code" that a computer's processor uses.

Python — Strings

Strings are the collection of characters.

Python strings can be declared in single quotes or double quotes.

Python does not support character type.

```
Declaration of python strings-

str1 = 'Hello World';

str2 = "Python Programming";
```

print() Function

A function in Python is a tool developer's use to save time and make their programs more efficient.

The print() function is used whenever you want to print text to the screen.

Ex-

print ("Hello World!")

Example of Python Codes Using Strings

```
Example-1 ->
```

```
str1 = 'Hello World'
str2 = "Python Programming"
print "var1[0]: ", str1[0]
print "var2[1:5]: ", str2[1:5]

Example-2 ->
var1 = 'Hello World'
print "Updated String :- ", var1[:6] + 'Python'
```

Data Types of Python –

Python supports four different numerical types –

int (signed integers):It is used to define integer values that can be positive or negative whole numbers with no decimal point.

$$Ex-x=100$$

b) long (long integers): They are used to store integers of unlimited size, written like integers.

The long data type is defined by using uppercase or lowercase L.

Ex-

y=212525242525L

3) **float** (**floating point real values**): They represent real numbers and are written with a decimal point dividing the integer and fractional parts. Floats may also be in scientific notation, with E or e.

Ex-

a = 120.25

b = 32.3e2

4) Complex (complex numbers): A complex number is a number that can be expressed in the form a + bi, where a and b are real numbers and i is the imaginary unit, satisfying the equation $i^2 = -1$. In this expression, a is called the real part and b is the imaginary part.

Ex-

a = 3e + 26J

Conjugate complex number

- The complex conjugate of a complex number is the number with equal real part and imaginary part equal in magnitude but opposite in sign.
- For example, the complex conjugate of 3 + 4i is 3 4i

Example-

z=2+3i

y=z.conjugate()

print(y);

Rules for declaring the python variable

- a) Must begin with a letter (a z, A B) or underscore (_)
- b) It is Case Sensitive
- c) It can be any (reasonable) length
- d) There are some reserved words which you cannot use as a variable name.

Declaration of variables

Python Assignment Statements

The assignment statement creates new variables and gives them values:

Syntax

<variable> = <expr>

Where the equal sign (=) is used to assign value (right side) to a variable name (left side).

Multiple Assignment

The multiple variables can be initialized once.

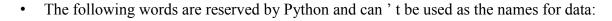
Syntax

var1=var2=var3...varn= = <expr>

Example:

x = y = z = 1

Reserved Keywords



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