

INTRODUCTION TO PYTHON

Python is a popular programming language. It was created by **Guido van Rossum**, and released in 1991.

It is used for:

- web development (server-side),
- software development,
- mathematics,
- system scripting.

FEATURES OF PYTHON

Python is a high-level language.

It is a free and open-source language

Python is case case-sensitive language.

Python is portable and platform-independent.

Python has a rich library of predefined functions.

Python use indentation for blocks and nested blocks.

Execution Modes of Python

There are two ways to use the python interpreter :-

Interactive mode: In the interactive mode we can simply type of python statement on the prompt>>> directly. As soon as the press enter ,the interpreter execute the statement and display the result(s). This mode is convenient for testing a single line code for instant execution. But in the interactive mode we cannot save the statement for future use and we have to retype the statement to run them again.

Script mode : In the script mode we can write a python program in a file, save it and then use the interpreter to execute it. Python file has an

extension “.py” to execute the python program in script mode click run → run module from a menu or press F5 from the keyboard.

Python Terminology

Keywords: Keywords are reserved words. Each keyword has a specific meaning to the Python interpreter, and we can use a keyword in our program only for the purpose for which it has been defined.

Identifiers: Identifiers are name used to identify a variable, function, or other entities in a program. The naming Conventions of identifiers in python are as follows

An identifier can not start with a digit.

Keywords can not be used as identifiers

We can not use special symbols like !, @, #, \$, % etc, in identifiers, (underscore can be used).

Example of valid identifiers

Num1

name

a_1

stu_marks

variables : Variables in python refers to an objects – an items or elements that is stored in the memory. Values of a variable can be strings (eg- ‘b’, sname), numeric (eg.,31) or any combination of alpha numeric characters (cd19) for eg

marks = 56

name='amit'

stu1=101

Rules to declare variable

- A variable can consist of upper- and lowercase letters, the digits 0-9 and the underscore character.
- The first character of a variable cannot be a digit.

- Keywords like if or the Boolean True are reserved and cannot be used as variable names.
- Variables are case-sensitive; therefore x is different from X.

Write a program to store the name of three student and display them.

```
name1='Aditya'  
name2='Ankit'  
name3='Sumit'  
print(name1)  
print(name2)  
print(name3)
```

output

Aditya

Ankit

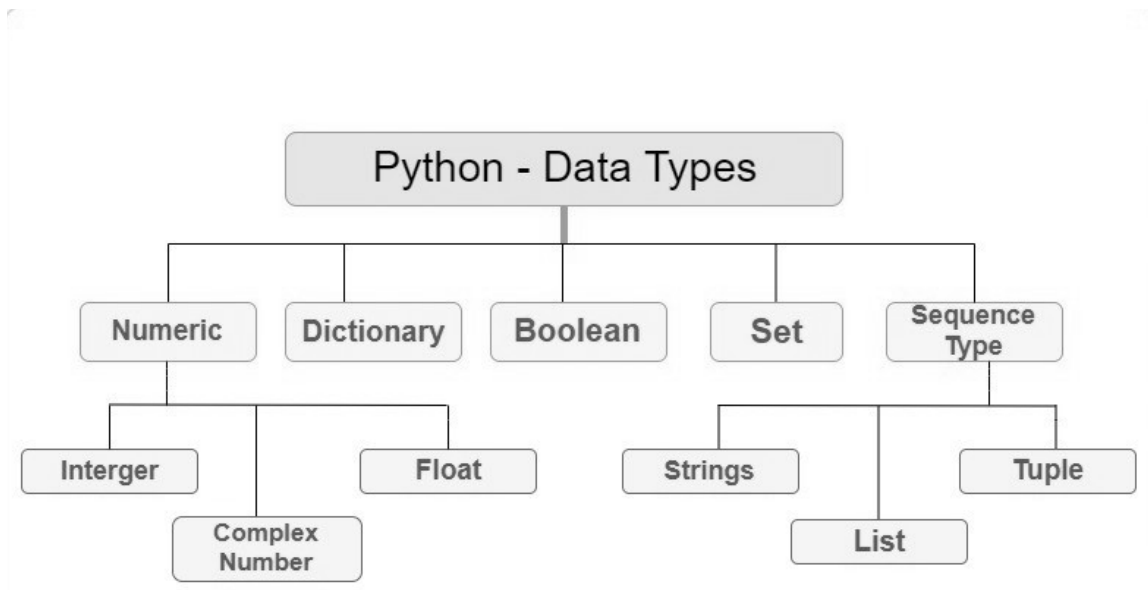
Sumit

Write a program to display the sum of two number

```
num1=25  
num2=36  
s=num1+num2  
print(s)
```

output: 61

Data types of python



1. Numeric Data Types in Python

The numeric data type in Python represents the data that has a numeric value. A numeric value can be an integer, a floating number, or even a complex number.

Int – this data type stores integers. Eg – 4,-5,-90

Float this data types stores floating points.Eg- 4.2,-23.45,3.14

Complex- This data type stores floating points number. Eg-1+2j, 21+5j

Boolean Data Type in Python

Python Data type with one of the two built-in values, True or False. Boolean objects that are equal to True are truthy (true), and those equal to False are falsy (false).

2. Sequence Data Types in Python

It is an ordered collection of items ,where each items is indexed by an integer.

1 string :- String is a collection of characters (like alphabets ,digits or special characters including spaces) . String are enclosed in single quotes (' ') or in double quotes (" ").

Eg - str1="anuj"

Str2='23454'

Additon of two string (string concatenation)

eg 1

fname ='abhishek'

lname='kumar'

S=fname+lname

Print(s)

Output- abhishekkumar

Eg 2

num1 ='102'

Num2='48'

S=num1+num2

Print(s)

Output- '10248'

List- list is a sequence of items enclosed in a square brackets [] and items are separated by commas .

List is mutable

For eg -

L1=[23, 'aman',123456789, 'kendriya vidyalaya']

L2=[2,4,6,8,10,12,14,16,18,20]

Tuple- tuple is a sequence of items seprated by commas and items are enclosed in parenthesis().

Tuple is immutable

Eg T1=(1,3,5,7,9)

Mapping- It is an unordered data type in python.

Dictionary

A dictionary in Python is an unordered collection of data values, used to store data values like a map, unlike other Python Data Types that hold only a single value as an element, a Dictionary holds a key: value pair. Key-value is provided in the dictionary to make it more optimized.

Eg -

D1={'name': 'ritik', 'age': 18, 'class': '9'}

Operators in python

Python language supports various types of operators, which are:

1. Arithmetic Operators
2. Comparison (Relational) Operators
3. Assignment Operators
4. Logical Operators
5. Bitwise Operators
6. Membership Operators
7. Identity Operators

Arithmetic Operators

Operator	Name	Example
+	Addition	10 + 20 = 30
-	Subtraction	20 - 10 = 10
*	Multiplication	10 * 20 = 200

/	Division	$20 / 10 = 2$
%	Modulus	$22 \% 10 = 2$
**	Exponent	$4^{**}2 = 16$
//	Floor Division	$9//2 = 4$

Operator	Name	Example
==	Equal	$4 == 5$ is not true.
!=	Not Equal	$4 != 5$ is true.
>	Greater Than	$4 > 5$ is not true
<	Less Than	$4 < 5$ is true
>=	Greater than or Equal to	$4 >= 5$ is not true.
<=	Less than or Equal to	$4 <= 5$ is true.

Comparison (Relational) Operators

Assignment Operators

Operator	Name	Example
=	Assignment Operator	$a = 10$
+=	Addition Assignment	$a += 5$ (Same as $a = a + 5$)
-=	Subtraction Assignment	$a -= 5$ (Same as $a = a - 5$)

<code>*</code>	Multiplication Assignment	<code>a *= 5</code> (Same as <code>a = a * 5</code>)
<code>/</code>	Division Assignment	<code>a /= 5</code> (Same as <code>a = a / 5</code>)
<code>%</code>	Remainder Assignment	<code>a %= 5</code> (Same as <code>a = a % 5</code>)
<code>**</code>	Exponent Assignment	<code>a **= 2</code> (Same as <code>a = a ** 2</code>)
<code>//</code>	Floor Division Assignment	<code>a //= 3</code> (Same as <code>a = a // 3</code>)

Logical Operators

Operator	Description	Example
and Logical AND	If both of the operands are true then the condition becomes true.	(a and b) is true.
or Logical OR	If any of the two operands is non-zero then the condition becomes true.	(a or b) is true.
not Logical NOT	Used to reverse the logical state of its operand	Not(a and b) is false.

AI BIAS

AI bias, also called machine learning bias or algorithm bias, refers to the occurrence of [biased results](#) due to human biases that skew the original training data or AI algorithm—leading to distorted outputs and potentially harmful outcomes.

AI ETHICS

Ethics is a set of moral principles which help us discern between right and wrong. AI ethics is a multidisciplinary field that studies how to optimize [AI's](#) beneficial impact while reducing risks and adverse outcomes.

Examples of AI ethics issues include data responsibility and privacy, fairness, explainability, robustness, transparency, environmental sustainability, inclusion, moral agency, value alignment, accountability, trust, and technology misuse.