

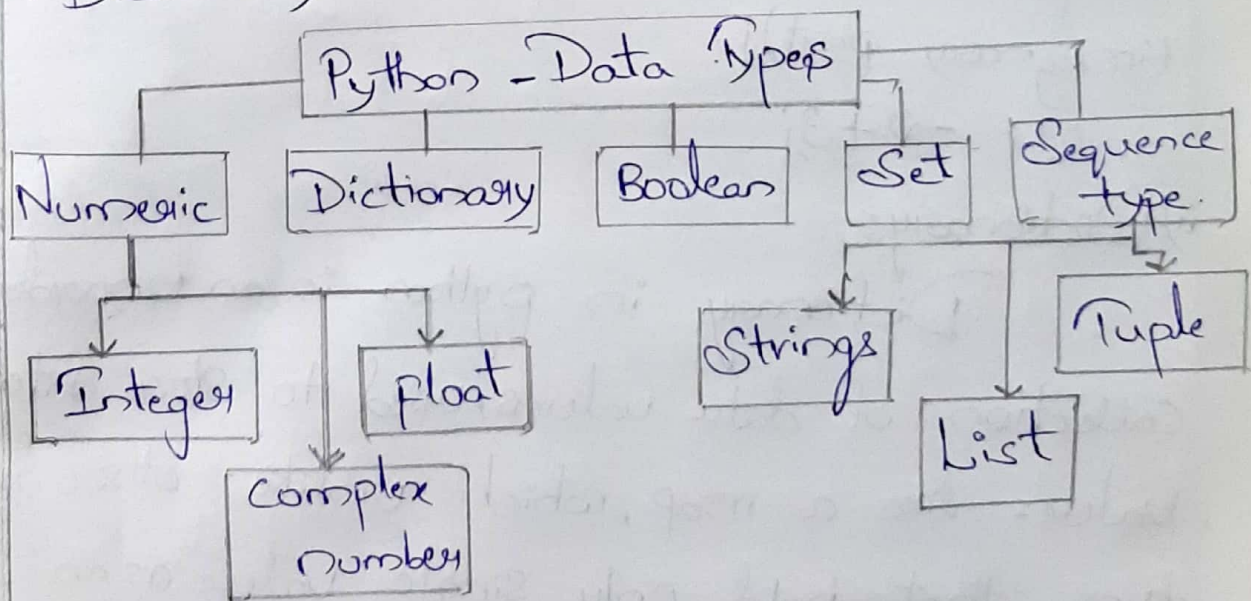
## Assignment - 2

1) What are the data types in python? Explain.

Python data types :- Data types are the classification or categorization of data items. It represents the kind of value that tells what operations can be performed on a particular data. Since everything is an object in python programming, data types are actually classes and variables are instance (object) of these classes.

The Standard or built-in data type of python:

- Numeric
- Sequence Type
- Boolean
- Set
- Dictionary.





## i) Numeric:-

Numeric data type represent the data which has numeric value. Numeric value can be integer, floating number or even complex numbers. These values are defined as int, float and complex class in python.

- **Integers:-** This value is represented by int class. It contains +ve or -ve whole numbers.

- **Float:-** It is represented by a float class. It is a real number when floating point representation. It is specified by a decimal point. Optionally the character e or E followed by a +ve or -ve integer.

- **Complex:-** Complex number is represented by complex class. It is specified as (real part) + (imaginary part)j.

ex:  $-2 + 3j$

## ii) Dictionary:-

Dictionary in python is an unordered collection of data values, used to store data values like a map, which unlike other data types that hold only single value as an element. It holds key: value pair. Key value is provided in it to make more optimized.



Each key-value pair in dictionary is separated by "colon"; whereas each key is separated by a "comma".

**Creating a Dictionary:-** It can be created by placing a sequence of elements within curly braces, separated by 'comma'. Values in it can be of any data-type and can be duplicated, the keys cannot be repeated and must be immutable.  
\* It also can be created by the built-in function `dict()`. An empty dictionary can be created by just placing a curly braces `{ }`.

iii) **Boolean:-**

Data type with one of the two built-in values, True or False. Boolean objects that are equal to True are true and those equal to False are false. But non-Boolean objects can be evaluated in boolean context as well and determined to be true or false. It is denoted by the class `bool`.

iv) **Set:-**

Set is an unordered collection of data type that is iterable, mutable and has no duplicate elements. The order of elements in a Set is undefined though it may consist



of various elements. The major advantage of using a Set, as opposed to a list, is that it has a highly optimized method for checking whether a specific element is contained in the set.

**Creating a Set :-** It can be created by using the built-in `set()` function with an iterable object or a sequence by placing the sequence inside curly braces, separated by 'Comma'. A set contains only unique elements but at the time of set creation, multiple duplicate values can be passed.

**V) Sequence type :-**

In python, Sequence is the ordered collection of similar or different data types. Sequences allow to store multiple values in an organised and efficient fashion. There are several sequence types in python.

- **String :-** Strings are arrays of bytes representing unicode characters. It is a collection of one or more characters put in a single quote, double-quote or triple quote. In python there is no character data type, a character is a string of length one. It is represented



by str class.

Creating a string :- It can be created by using double / single / triple quotes.

• List :- It is just like arrays declared in other languages. It need not be homogeneous always which makes it the most powerful tool in python. A single list may contain data types like integers, strings as well as objects. These are mutable and hence they can be altered even after their creation.

Creating a list :- Lists in python can be created by just placing the sequence inside the square brackets []. Unlike sets, lists don't need a built-in function for creation of list.

• Tuple :- It is an ordered collection of python objects much like a list. The sequence of values stored in tuple can be any of type and they are indexed by integers. The important difference between a list and a tuple is that tuples are immutable. Also, tuples are hashable whereas lists are not. It is represented by tuple class.

Creating a Tuple :- It is created by placing

Sequence of values separated by 'comma' with/ without the use of parentheses for grouping of data sequence. It may contain any no. of elements and any data type.

ex:- `type(1034)`

`<class 'int'>`

`type(55.50)`

`<class 'float'>`

`type(6+4i)`

`<class 'complex'>`

`type("hello")`

`<class 'str'>`

`type([1, 0, 3, 4])`

`<class 'list'>`

`type((1, 0, 3, 4))`

`<class 'tuple'>`

`type({1: "one", 0: "two", 3: "three"})`

`<class 'dict'>`

Q) Briefly explain history of python.

Python is a widely used general-purpose, high level programming language. It is initially designed by "Guido van Rossum" in 1991 and developed by Python Software Foundation. It was mainly developed for emphasis on code



readability, and its syntax allows programmer to express concepts in fewer lines of codes.

History :-

In the late 1980s, history was about to be written. It was that time when working on python started. Soon after that, Guido van Rossum began doing its application based work in December of 1989 by at Centrum Wiskunde and Informatie (CWI) which is situated in Netherlands. The programming language which Python is said to be ABC programming language, which had the interfacing with the Amoeba Operating System and had the feature of exception handling. He had taken the Syntax of ABC, and some of its good features. It came with a lot of complaints too. So he fixed some issues completely and had created a good scripting language which had removed all the flaws. The inspiration for the name came from BBC's TV show - 'Monty Python's Flying Circus', as he was a big fan of the TV show and also he wanted a short, unique and slightly mysterious name for his inven



tion and hence he named it python! He was the "Benevolent dictator for life" (BDFL) until he stepped down from the position as a leader on 12th July 2018.

The language was released in 1991. When it was released, it used a lot fewer codes to express the concepts, when we compare it with C, C++, Java. Its design philosophy was quite good too. Its main objective is to provide code readability and advanced developer productivity.

3) Explain all the operators in python.

i) Arithmetic operators:- These are used to perform mathematical expressions like addition, subtraction, multiplication and division.

Operator	Description	Syntax
+	To add two numbers	$a+b$
-	To subtract 2 numbers	$a-b$
*	To multiply 2 numbers	$a*b$
/	(float) To divide 2 numbers	$a/b$
//	(floor) To divide 1 <sup>st</sup> operand by 2 <sup>nd</sup>	$a//b$
%	Returns the remainder by 2 numbers	$a\%b$
**	Power:- Returns first raised to power second	$a**b$



ii) Relational Operators :- It compares two values. either it returns true/false according to the condition.

Operator	Description	Syntax
>	True if left operand is greater than right. (Greater than)	$x > y$
<	True if left operand is <sup>less</sup> greater than right (less than)	$x < y$
==	True when both operands are equal	$x == y$
!=	True where operands are unequal	$x != y$
>=	left operand is greater than or equal to the right	$x >= y$
<=	left operand is less than or equal to right	$x <= y$

iii) Logical Operators :- It performs logical operations like logical AND, Logical OR and logical NOT.

Operator	Description	Syntax
and (&&)	if both are true	$x \&\& y$
OR (  )	if either one is true	$x    y$
not	if operand is false	$\sim x$

Bitwise Operator	Description	Syntax
&	Bitwise AND	$x \& y$
	Bitwise OR	$x   y$
~	Bitwise NOT	$\sim x$



$\wedge$	Bitwise XOR	$x \wedge y$
$\gg$	Bitwise right shift	$x \gg$
$\ll$	Bitwise left shift	$x \ll$

### v) Assignment operators :-

Operator	Description	Syntax
$=$	equals to	$x = y + z$
$+=$	add right side operand to the left side & assign to left operand	$x += b$ $x = x + y$
$-=$	Subtract right operand to left and assign to left operand	$a -= b$ $a = a - b$
$*=$	multiply right with left & assign to left	$a *= b$ $a = a * b$
$/=$	divide left with right and assign to left	$a /= b$ $a = a / b$
$./=$	use left & right & assign to left.	$a ./= b$ $a = a ./ b$
$//=$	divide left operand and then assign the floor value to left operand.	$a //= b$ $a = a // b$
$**=$	using two operands calculate the power value using operands and assign value to left	$a **= b$ $a = a ** b$
$\&=$	Performs bitwise And on operands and assign value to left	$a \&= b$ $a = a \& b$



$  =$	Performs bitwise OR on operands and assign value to left	$a   = b$ $a = a   b$
$\wedge =$	Performs bitwise XOR on operands and assign to left	$a \wedge = b$ $a = a \wedge b$
$\ll =$	Bitwise left shift & assign to left	$a \ll = b$ $a = a \ll b$
$\gg =$	Bitwise right shift & assign to left	$a \gg = b$ $a = a \gg b$

vi) Special operators:- These are some special type of operations like "identity operators." is and is not are the identity operators both are used to check if two values are located on the same part of the memory. Two variables that are equal does not imply that they are identical.

4) Explain the features of python.

Python is a dynamic, high level, free open source and interpreted programming language. It supports object-oriented programming as well as procedural oriented programming. In python, we need not to declare the type of a variable, because it is a dynamic typed language.



## Features in python:-

- 1) Easy to code :- It is a high level programming language. Very easy to learn as compared to other language. It is very easy to code in Python language and anybody can learn easily. It is developer-friendly language.
- 2) Free and open source :- It is freely available at official website and it is open source, this means that source code is also available to the public. So you can download it, as well as share it.
- 3) Object-Oriented language :- One of the key feature of python is object-oriented programming. python supports object-oriented language and concepts of classes, objects encapsulation etc.
- 4) GUI programming support :- Graphical user interfaces can be made using a module such as PyQt5, PyQt4, wxPython or Tk in Python. PyQt5 is the most popular option for creating graphical apps with python.
- 5) High-level language :- When we write programs in python, we do not need to



remember the System architecture, nor do we need to manage memory.

6) Extensible feature :- we can write some Python code into c or c++ and also we can compile that code in c/c++ language.

7) Python is portable language :- if we have Python code for windows and if we want to run this code on other platform such as linux, Unix and mac then we do not need to change it, we can run this code on any platform.

8) Python is Integrated language :- We can easily integrate python with other language like c, c++ etc.

9) Interpreted language :- When python code is executed line by line at a time like other language c, c++, java etc there is no need to compile python code this makes it easier to debug our code. The source code immediately converts into byte code.

10) Large Standard Library :- which it provides rich set of module functions so you do not have to write your own code for



every single thing. There are many libraries present in python for such as regular expressions, unit-testing, web browsers etc.

11) Dynamically Typed language :- That means the type for a variable is decided at run time not in advance. because of this feature we don't need to specify the type of variable.

5) Justify why (ph) python is 'interactive interpreted language'.

Python is a 'interactive interpreted language'. interpreted it is meant that each time a program is run the interpreter checks through the code for errors and then interprets the instructions into machine readable bytecode.

An 'interpreter' is a translator in computer's language which translates the given code line by line in machine readable language (bytecodes). And when new lines are fed into the 'interpreter', the fed program is evaluated both in part and in whole.