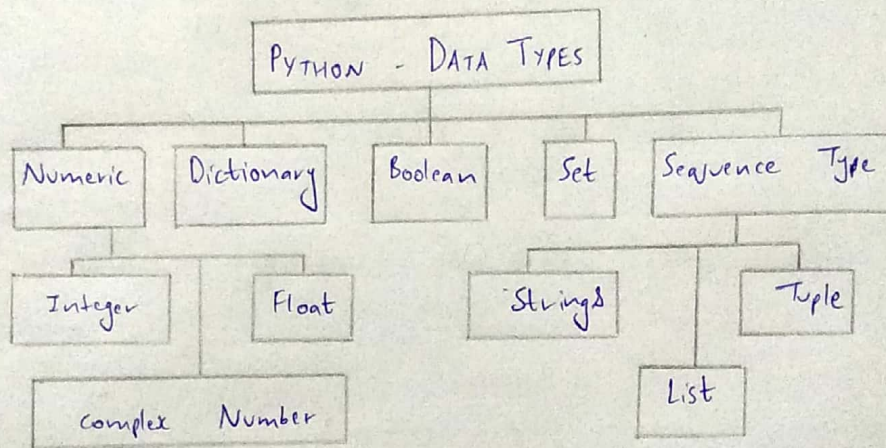


Python Assignment - 04

1. What are the datatypes in python? Explain.

Ans

Datatypes 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, datatypes are actually classes and variables are instance (object) of these classes.



Numeric

A numeric value is any representation of data which has a numeric value. Python identifies three types of numbers:

- Integer - Positive or negative whole numbers.
- Float - Any real number with a floating point representing in which a fractional component is denoted by a decimal symbol or scientific notation.
- Complex number - A number with a real and imaginary component represented as $x + yj$. x and y are floats and j is -1 (square root of -1 called an imaginary number).

Dictionary

A dictionary object is an unordered collection of data in key-value pair form. A collection of such pairs is enclosed in curly brackets.

Boolean

Data with one of two built-in values True or False. Notice that 'T' and 'F' are capital.

Set

Set is an unordered collection of datatypes that is iterable, mutable and has no duplicate items/elements. The major advantage of using a set is that it has a highly optimized method for checking whether a specific element is contained in the set.

Sequence type

A sequence is an ordered collection of similar or different data types. Python has the following built-in data types:

- String - A string value is a collection of one or more characters put in single, double or triple quotes.
- List - A list object is an ordered collection of one or more data items, not necessarily of the same type, put in square brackets.
- Tuple - A tuple object is an ordered collection of one or more data items, not necessarily of the same type, put in parentheses.

2. Briefly explain history of Python.

Ans

- Python laid its foundation in the late 1980's.
- The implementation of Python was started in December 1989 by Guido Van Rossum at CWI in Netherlands.
- In February 1991, Guido Van Rossum published the code (labelled 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 such as list comprehensions, garbage collection systems.
- On December 3, 2008, Python 3.0 (also called "Py3k") was released. It was designed to rectify the 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 the Amoeba Operating System.
- The following programming languages influence Python:
 - ABC language.
 - Modula-3.
- The latest Python version 3.8 was released on October 14, 2019.

3. Explain all the operators in Python.

Ans

Operators are used to perform operations on variables and values.

Python divides the operators in the following groups:

- Arithmetic operators
- Assignment operators
- Comparison operators
- Logical operators
- Identity operators
- Membership operators
- Bitwise operators

Python Arithmetic Operators

Arithmetic operators are used with numeric values to perform common mathematical operations.

Operator	Name	Example
+	Addition	$x + y$
-	Subtraction	$x - y$
*	Multiplication	$x * y$
/	Division	x / y
%	Modulus	$x \% y$
**	Exponentiation	$x ** y$
//	Floor division	$x // y$

Python Assignment operators

Assignment operators are used to assign values to variables.

Operator	Example	Same as
=	$x = 5$	$x = 5$
+=	$x += 5$	$x = x + 5$
-=	$x -= 3$	$x = x - 3$
*=	$x *= 3$	$x = x * 3$
/=	$x /= 3$	$x = x / 3$
%=	$x \% = 3$	$x = x \% 3$
//=	$x //= 3$	$x = x // 3$
**=	$x ** = 3$	$x = x ** 3$
&=	$x \& = 3$	$x = x \& 3$
=	$x = 3$	$x = x 3$
^=	$x ^= 3$	$x = x ^ 3$
>>=	$x >> = 3$	$x = x >> 3$
<<=	$x << = 3$	$x = x << 3$

Python Comparison Operators

Comparison operators are used to compare two values.

Operator	Name	Example
==	Equal	$x == y$
!=	Not equal	$x != y$
>	Greater than	$x > y$
<	Less than	$x < y$
>=	Greater than or equal to	$x \geq y$
<=	Lesser than or equal to	$x \leq y$

Python Logical Operators

Logical operators are used to combine conditional statements.

Operator	Description	Example
and	Returns True if both statements are true.	$x < 5$ and $x < 10$
or	Returns True if one of the statements is true.	$x < 5$ or $x < 4$
not	Reverse the result, returns False if the result is true.	not ($x < 5$ and $x < 10$)

Python Identity Operators

Identity operators are used to compare the objects, not if they are equal, but if they are actually the same object, with the same memory location.

Operator	Description	Example
is	Returns True if both variables are the same object.	$x \text{ is } y$
is not	Returns True if both variables are not the same object.	$x \text{ is not } y$

Python Membership Operators

Membership operators are used to test if a sequence is presented in an object.

Operator	Description	Example
in	Returns True if a sequence with the specified value is present in the object.	x in y
not in	Returns True if a sequence with the specified value is not present in the object.	x not in y

Python Bitwise Operators

Bitwise Operators are used to compare (binary) numbers.

Operator	Name	Description
&	AND	Sets each bit to 1 if both bits are 1.
	OR	Sets each bit to 1 if one of two bits is 1.
^	XOR	Sets each bit to 1 if only one of two bits is 1.
~	NOT	Inverts all the bits.
<<	Zero fill left shift	Shift left by pushing zeroes in from the right and let the leftmost bits fall off.
>>	Signed right shift	Shift right by pushing copies of the leftmost bit in from the left, and let the rightmost bits fall off.

4. Explain the features of Python.

Ans There are many features in Python, some of which are discussed below.

- Easy to code - Python is a high-level programming language. It has few keywords, simple structure and a clearly defined syntax. Easier compared to other programming languages. It is a developer-friendly language.
- Free and open source - Python language is available to download at official website. Source code is also available to the public.
- Object-oriented programming (OOP) language - Python supports OOP and concepts of classes, objects, encapsulation etc.
- GUI Programming support - Graphical User Interfaces can be made using a module such as PyQt5 (most popular), PyQt4, wxPython or Tk in Python.
- High-level language - When we write programs in Python, we need not remember the system architecture or manage the memory.
- Extensible feature - We can write up some Python code into C or C++ and also compile that code in C/C++ language.
- Portable language - Python code can be run on any platform.
- Interpreted language - Python code is executed line by line at a time making it easier to debug our code. This source code of Python is converted into an intermediate form called bytecode.

- Large Standard Library - Python has a large standard library which provides a rich set of module and function so you need not write your own code for every single thing.
 - Dynamically typed language - We need not specify the type of variable as the data type for a variable is decided at run time and not in advance.
-

5. Justify why python is interactive interpreted language.

Ans Python has 2 modes - Script and interactive mode. .py files are run in the python interpreter. Interactive mode is a command line shell which gives immediate feedback for each statements in active mem., while running previously fed statements in active memory. Helps in the debugging process.

Python is called an interpreted language because it goes through an interpreter, which turns code you write into the language understood by your computer's processor i.e source code is converted into byte code by the interpreter.
