

Assignment - 2

1) What are the data types of python? Explain the standard types of python

- * Numeric
- * Sequence type
- * Boolean
- * Set
- * dictionary

Numeric :- In python numeric data type represents the data which has numeric value. Numeric value can be integer, floating number (or) Even Complex numbers

These are divided into

- * Integers
- * Float
- * Complex numbers

Integer :- It is represented by int class. It contains positive (or) negative whole numbers

Float :- It is represented by float class. It is a real number with floating point representation. It is specified by decimal point.

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

Sequence type :- Sequence is the ordered collection of similar (or) different datatypes. Sequences allow to store multiple values in an organized and efficient fashion.

There are several sequence types in python

* String

* List

* Tuple

⇒ String :- Strings are arrays of bytes representing unicode characters. It is represented by str class.

⇒ List :- Lists are just like the arrays declared in other languages. It is represented by list class.

⇒ Tuple :- Tuples are created by placing sequence of values separated by 'comma' with (or) without the use of parenthesis for grouping of data sequence.

⇒ Boolean :- Data type with one of the two built-in values, True (or) False. In python True and False should be capital 'T' and 'F'. It is terminated as bool.

Set :- Set is an unordered collection of datatype that is iterable, mutable and has no duplicate elements. The major advantage of using a set is as opposed to a list.

Dictionary :- Dictionary can be created by placing a sequence of elements within curly braces, separated by 'comma' and the other corresponding pair elements being key: value. Whereas keys can't be repeated and must be immutable.

2) Briefly explain history of python

Python was conceived in the late 1980's by GUIDO VAN ROSSUM at centrum wiskunde & informatica (CWI) in the Netherlands as to the ABC language (itself inspired by SCL) capable of exception handling and interfacing with the amoeba operating system.

⇒ python 2.0, released in 2000, introduced features like list comprehensions and a garbage collection system with reference counting

⇒ python is a multi-paradigm programming language object oriented programming and structured programming

3) Explain all the operators in python

* Arithmetic operator :- It is used to perform mathematical operations like addition, subtraction, multiplication and division

<u>operator</u>	<u>meaning</u>	<u>Example</u>
+	add 2 operands	$x + y$
-	subtracts 2 operands	$x - y$
*	multiplies 2 operands	$x * y$
/	divides 1 st operand	x / y
//	floor division	$x // y$
**	left operand raised to the power of right	$x ** y$

Comparison operators :- comparison operators are used to compare values. It returns either 'True' (or) False according to the condition

<u>operator</u>	<u>meaning</u>	<u>Example</u>
>	Greater than	$x > y$
<	less than	$x < y$
==	Equal to	$x == y$
!=	Not Equal to	$x != y$

$>=$	Greater than or Equal to	$x >= y$
$<=$	less than or Equal to	$x <= y$

Logical operators :- Logical operators are the and or, not operators

<u>operators</u>	<u>meaning</u>	<u>Example</u>
and	True if both the operands are true	$x \text{ and } y$
or	True if either of the operands is True	$x \text{ or } y$
not	True if operand is false (complements the operand)	not x

Bitwise operators :- Bitwise operators act on operands as if they were strings of binary digits. They operate bit by bit

<u>operator</u>	<u>meaning</u>	<u>Example</u>
$\&$	Bitwise AND	$x \& y$
$ $	Bitwise OR	$x y$
\sim	Bitwise Not	$\sim x$
\wedge	Bitwise XOR	$x \wedge y$
$>>$	Bitwise right shift	$x <<$

Assignment operators :- Assignment operators are used in python to assign values to variable

<u>operator</u>	<u>meaning</u>	<u>Example</u>
$+=$	Add AND	$x += y$
$-=$	Subtract AND	$x = x + y$ $x -= y$
$*=$	Multiply AND	$x = x - y$ $x *= y$
$/=$	Division AND	$x /= y$ $y = x / y$
$\%=$	modulus AND	$x \% = y$ $x = x \% y$
$//=$	floor AND	$x // = y$ $x // y$
$**=$	Exponent AND	$x ** = y$ $x = x ** y$
$\wedge =$	Bitwise OR	$y = x \wedge y$
$\wedge =$	Bitwise XOR	$x \wedge = y$ $x = x \wedge y$

membership operators:-

in and not are the membership operators in python

<u>operator</u>	<u>meaning</u>	<u>Example</u>
in	True if value is found in the sequence	$5 \text{ in } y$

not in

True if value
is not found in
sequence

5 not in x

4) Explain the features of python

- * Easy to code
- * Free and open source
- * Object oriented language
- * Extensible
- * Large standard library
- * GUI programming support
- * Integrated and Interpreted language
- * Portable language
- * High level language
- * Dynamically Typed language

5) Justify why python is interactive interpreted language

- * python program runs directly from the source code
- * python converts source code written by the programmer into intermediate language which is again translated into the native language / machine language that is executed so python is interpreted language

* python processed at runtime by the interpreter program need to be compiled before it's execution

— X —