

1. What are the datatypes in Python? Explain?

In Python datatypes are divided into 5 types

1. Numeric
2. Sequence type
3. Boolean
4. Set
5. Dictionary

1. Numeric: Numeric datatype represents the data which has numeric value. Numeric value can be integer, float or complex numbers.

Integers - This value is represented by int class. It contains positive or negative whole numbers. In Python there is no limit on how long the integer value can be.

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

Complex number - It is represented by complex class. Specified as (real part) + (imaginary part).

2. Sequence type: Sequence is the ordered collection of similar or different data types. It allows to store multiple values in an organized and efficient fashion. Several sequence types are

String: strings are arrays of bytes representing unicode characters. A string is a collection of one or more characters put in a single quote, double quotes or triple quote. In Python there is no character data type, a character is a string of length one. Represented by str class. Strings are immutable.

List: lists are just like arrays declared in other languages.

List need not be homogeneous always which makes it the most powerful tool. A single list may contain data types like int, strings as well as objects. Lists are mutable and represented by list class. Indexing starts from 0. enclosed by [] braces.

`append()` → only one element can be added to the list

`insert()` → addition of element at desired position

`extend()` → add multiple elements at same time at the end of list

Tuple: It is an ordered collection of Python objects much like a list. The sequence of values stored in a tuple can be of any type, are indexed by integers. Tuples are immutable. Represented by tuple class.

3. Boolean: Data types with two built-in values, true or false. Denoted by bool class.

4. Set: Set is an unordered collection of data type that is iterable, mutable and has no duplicate elements.

5. Dictionary: It is an unorder collection of data values used to store values like map, which unlike other data types that hold only single value as an element, Dictionary holds key:value pair.

2. Explain history of Python?

Python was conceived in the late 1980s as a successor to the ABC language. Python 2.0 released in 2000, introduced features like comprehensions and a garbage collection system with reference counting.

Python 3.0, released in 2008, was a major revision of the language that is not completely backward-compatible, and much Python 2 code does not run unmodified on Python 3.

The Python 2 language was officially discontinued in 2020 and "Python 2.7.18" is the last python 0.7 release and therefore Python 2 release.

3 Explain all the operators in python

types of operators

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

Arithmetic operators:

operator	description	Example
+ addition	Add values on either side of the operator	$a = 10, b = 20$ $a + b = 30$
- subtraction	Subtracts right hand from left hand operand	$a - b = -10$
* multiplication	multiplies values on either side of operator	$a * b = 200$
/ division	divides left hand operand by right hand operand	$b / a = 2$
% Modulus	divides left hand operand by right hand operand and returns remainder	$b \% a = 0$
** Exponent	performs exponential calculation on operators	$a ** b \in 10^{20}$
// Floor division	Division of operands where the result is the quotient in which the digits after the decimal point are removed.	$9 // 2 = 4$

4. Explain features of Python?

Features of Python are

1. Easy to learn - Python has few keywords, simple structure and a clearly defined syntax.
2. Easy to read - It is more clearly defined and visible to the eyes.
3. Easy to maintain: source code is fairly easy to maintain.
4. Broad standard library: The bulk of library is very portable and class-platform compatible on UNIX, Windows.
5. Interactive mode: Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
6. Portable: Python can run on a wide variety of hardware platforms and has same interface on all platforms.
7. Extendable - You can add low-level modules to the interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
8. Database: Python provides interfaces to all major commercial database
9. GUI Programming: supports GUI applications that can be created and ported to many system calls, libraries and window systems, such as windows MFC, Matintosh.
10. Scalable - Python provides a better structure and support for large programs than shell scripting.

5. Justify why python is interactive interpreted language?
unlike c/c++ etc python is an interpreted object-oriented programming language. By interpreted it means that each time a program is run the interpreter checks through the code for errors and then interprets the instructions into machine readable.

python is interactive. When a python statement is entered and is followed by the return key if appropriate the result will be printed on screen, immediately on next line.