

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

\* There are 5 types of data types in python.

1. Numbers
2. Strings
3. Lists
4. Tuples
5. Dictionary

1. Numbers: Number data types store numeric values. Number objects are [created] created when you assign a value to them.

2. Strings: Strings in python are identified as a contiguous set of characters represented in the quotation marks. python allows either pair of single or double quotes.

3. Lists: Lists are the most versatile of python's compound data types. A list contains items separated by commas and enclosed within square brackets [].

4. Tuples: A Tuple is another sequence data type that is similar to the list. A tuple consists of a number of values separated by commas. Unlike lists, however, tuples are enclosed within parentheses.

5. Dictionary: python's dictionaries are kind of hash-table type. They work like associative arrays or hashes found in perl and consist of key-value pairs. A dictionary key can be almost any python type, but an

usually numbers or strings. values, on the other hand can be any arbitrary python object. Dictionaries are enclosed within curly braces.

2. Briefly explain history of python?

### History of Python

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 & Informatica (CWI) which is situated in Netherlands. It was started firstly as a hobby project because he was looking for an interesting project to keep him occupied during Christmas.

The programming language which python is said to have succeeded is ABC Programming language, which had the interfacing with the Amos Operating System and had the feature of exception handling. He had already helped to create ABC earlier in his career and he had seen some issues with ABC but liked most of the features. After that what he did as really very clever he had taken the syntax of ABC, and some of its good features. It came with a lot of complaints too, so he fixed those 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 invention and hence he named it python! He was the "Benevolent dictator for life" (BDFL) until he stepped down from the position as the leader on 12<sup>th</sup> July 2018. For quite some time he used to work for Google, but currently, he is working at Dropbox. The language was finally released in 1991.

When it was released, it used a lot fewer codes to express the concepts when we compare it with Java, C++ & C. Its design philosophy was quite good too. Its main objective is to provide code readability and advanced developer productivity. When it was released it had more than enough capability to provide classes with inheritance, several core data types, exception handling and functions.

3. Explain all the operators in python?

### Python Operators :

1. Arithmetic operators : Arithmetic operators are used to perform mathematical operations like addition, subtraction, multiplication and division.

Eg: +, -, \*, /, //, %, \*\*

2. Relational Operators : Relational Operators Compare the values. It either returns True or False according to the Condition.

eg:-  $>$ ,  $<$ ,  $==$ ,  $!=$ ,  $>=$ ,  $<=$

3. Logical Operators : Logical operators are used primarily in expression evaluation to make a decision.

Python Supports and, or, not logical Operators

4. Comparison Operator :

Comparison Operators are used to (perform) Compare the value of the two Operands and returns boolean True or False accordingly.

5. Assignment Operators : These are used to assign the value of the right expression to the left operand.

Ex:-  $=$ ,  $+=$ ,  $-=$ ,  $*=$ ,  $/=$ ,  $**=$ ,  $//=$

6. Bitwise Operators:-

The Bitwise Operators perform bit by bit Operation on the values of two Operands.

Binary and (&)

Binary or (|)

Binary xor (^)

Negation (~)

Left Shift (<<)

Right Shift (>>)



## Membership Operators :

These are used to check the membership of value inside a python. If the value is present in data structure, then the resulting value is true otherwise it returns false.

⇒ in and not in are membership Operators.

## vii) Identify Operators :

is - It is evaluated to be true if the reference present at both side point to the same object

is not - It is evaluated to be true if the reference present at both side do not point to the same object.

## 4. Explain the features of python?

### 1. Easy to learn and use

Python is easy to learn and use. It is developer-friendly and high level programming language.

### 2. Expression Expressive language : It means that is more understandable and readable

### 3. Interpreted language : Interpreter executes the code line by line at a time. This makes debugging easy and thus suitable for beginners.

#### 4. Cross-platform

It can run equally on different platform such as windows, linux, Unix, etc. So we can say python is a portable language.

#### 5. Free and Open Source:

It is freely available at official web address. Source code is also available, it is open source.

#### 6. Object-Oriented language:

It supports object oriented language and concept of classes and objects can come into existence.

#### 7. Extensible:

It implies that other languages such as C/C++ can be used to compile the code and thus it can be used further in our python code.

#### 8. Large Standard Library:

python has large and broad library and provide rich set of module and functions for rapid application development.

#### 9. GUI programming Support:

Graphical user interfaces can be developed using python.

#### 10. Integrated: It can be easily integrated with language like C, C++, Java, etc.



5. Justify why python is interactive interpreted language?

Python is an interacted interpreted language because unlike c/c++ programming language by 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 byte code.

We can easily integrated python with other language like c, c++ etc. There is no need to compile python code - this ~~marks~~ makes it easier to debug our code. This source code of python is converted into an immediate form called byte code.