1. DATA TYPES IN PYTHON :

## Data types in Python

Every value in Python has a datatype. Since everything is an object in Python programming, data types are actually classes and variables are instance (object) of these classes.

There are various data types in Python :

Integers, floating point numbers and complex numbers fall under [Python numbers](https://www.programiz.com/python-programming/numbers) category. They are defined as int, float and complex classes in Python.

We can use the type() function to know which class a variable or a value belongs to. Similarly, the isinstance() function is used to check if an object belongs to a particular class.

Integers can be of any length, it is only limited by the memory available.

A floating-point number is accurate up to 15 decimal places. Integer and floating points are separated by decimal points. 1 is an integer, 1.0 is a floating-point number.

Complex numbers are written in the form, x + yj, where x is the real part and y is the imaginary part.

## Python List

[List](https://www.programiz.com/python-programming/list) is an ordered sequence of items. It is one of the most used datatype in Python and is very flexible. All the items in a list do not need to be of the same type.

Declaring a list is pretty straight forward. Items separated by commas are enclosed within brackets [ ].

Lists are mutable, meaning, the value of elements of a list can be altered.

## Python Tuple

[Tuple](https://www.programiz.com/python-programming/tuple) is an ordered sequence of items same as a list. The only difference is that tuples are immutable. Tuples once created cannot be modified.

Tuples are used to write-protect data and are usually faster than lists as they cannot change dynamically.

It is defined within parentheses () where items are separated by commas.

## Python Strings

[String](https://www.programiz.com/python-programming/string) is sequence of Unicode characters. We can use single quotes or double quotes to represent strings. Multi-line strings can be denoted using triple quotes, ''' or """.

1. HISTORY OF PYTHON :

# History of Python

[Python](https://www.geeksforgeeks.org/python-programming-language/) is a widely used general-purpose, high-level programming language.

* Python laid its foundation in the late 1980s.
* The implementation of Python was started in the December 1989 by **Guido Van Rossum** at CWI in Netherland.
* In February 1991, van Rossum published the code (labeled 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 like: list comprehensions, garbage collection system.
* On December 3, 2008, Python 3.0 (also called "Py3K") was released. It was designed to rectify 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 Amoeba Operating System.
* Python is influenced by following programming languages:
  + ABC language.
  + Modula-3

Python developer, Rossum always wanted the name o

1. OPERATORS IN PYTHON :

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

ARITHMETIC OPERATIONS :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Operator** | **Name** | **Example** |  | **Try it** |
| + | Addition | x + y |  | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_oper_add) |
| - | Subtraction | x - y |  | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_oper_sub) |
| \* | Multiplication | x \* y |  | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_oper_mult) |
| / | Division | x / y |  | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_oper_div) |
| % | Modulus | x % y |  | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_oper_mod) |
| \*\* | Exponentiation | x \*\* y |  | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_oper_exp) |
| // | Floor division | x // y |  | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_oper_floordiv) |

ASSINGMENT OPERATORS :

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

CONDITIONAL OPERATRS :

|  |  |  |
| --- | --- | --- |
| **Operator** | **Name** |  |
| == | Equal |  |
| != | Not equal |  |
| > | Greater than |  |
| < | Less than |  |
| >= | Greater than or equal to |  |
| <= | Less than or equal to |  |

|  |  |
| --- | --- |
| **Operator** | **Description** |
| and | Returns True if both statements are true |
| or | Returns True if one of the statements is true |
| not | Reverse the result, returns False if the result is true |

LOGICAL OPERATORS :

# Python Features

### 1. Easy

When we say the word ‘easy’, we mean it in different contexts.

#### **a. Easy to Code**

As we have seen in earlier lessons, Python is very **easy to code**. Compared to other popular languages like Java and C++, it is easier to code in Python. Anyone can learn [***Python syntax***](https://data-flair.training/blogs/python-syntax-semantics/)in just a few hours. Though sure, mastering Python requires learning about all its advanced concepts and packages and modules. That takes time. Thus, it is programmer-friendly.

#### **b. Easy to Read**

Being a high-level language, Python code is quite like English. Looking at it, you can tell what the code is supposed to do. Also, since it is **dynamically-typed**, it mandates indentation. This aids readability.

### 2. Expressive

First, let’s learn about expressiveness. Suppose we have two languages A and B, and all programs that can be made in A can be made in B using local transformations. However, there are some programs that can be made in B, but not in A, using local transformations. Then, B is said to be more expressive than A. [**Python**](https://techvidvan.com/tutorials/python-tutorial/) provides us with a myriad of constructs that help us focus on the solution rather than on the syntax. This is one of the outstanding python features that tell you why you should learn Python.

### 3. Free and Open-Source

Firstly, Python is **freely available**. You can download it from the [Python Website](https://www.python.org/downloads/).

Secondly, it is **open-source**. This means that its source code is available to the public. You can download it, change it, use it, and distribute it. This is called **FLOSS(Free/Libre and Open Source Software)**. As the Python community, we’re all headed toward one goal- an ever-bettering Python.

### 4. High-Level

As we discussed in point 2b, it is a high-level language. This means that as programmers, we don’t need to remember the system architecture. Nor do we need to manage the memory. This makes it more **programmer-friendly** and is one of the key python features.

### 5. Portable

Let’s assume you’ve written a Python code for your Windows machine. Now, if you want to run it on a Mac, you don’t need to make changes to it for the same. In other words, you can take one code and run it on any machine, there is no need to write different code for different machines. This makes Python a **portable language**. However, you must avoid any system-dependent features in this case.

### 6. Interpreted

If you’re familiar with any languages like C++ or Java, you must first compile it, and then run it. But in Python, there is no need to compile it. Internally, its source code is converted into an immediate form called **bytecode**. So, all you need to do is to run your Python code without worrying about linking to libraries, and a few other things.

By interpreted, we mean the source code is executed line by line, and not all at once. Because of this, it is **easier to debug your code**. Also, interpreting makes it just slightly slower than Java, but that does not matter compared to the benefits it has to offer.

If you have any doubt in DataFlair’s features of python programming language article, drop a comment below and we will get back to you.

### 7. Object-Oriented

A programming language that can model the real world is said to be object-oriented. It focuses on objects and combines data and functions. Contrarily, a procedure-oriented language revolves around functions, which are code that can be reused. Python supports both **procedure-oriented** and **object-oriented programming** which is one of the key python features. It also supports multiple inheritances, unlike Java. A class is a blueprint for such an object. It is an abstract data type and holds no values.

### 8. Extensible

If needed, you can write some of your Python code in other languages like**C++**. This makes Python an extensible language, meaning that it can be extended to other languages.

[***240+ Python Tutorials***](https://data-flair.training/blogs/python-tutorials-home/)***– Master the Python Programming with practicals and real-time projects.***

### 9. Embeddable

We just saw that we can put code in other languages in our Python source code. However, it is also possible to put our Python code in a source code in a different language like C++. This allows us to integrate scripting capabilities into our program of the other language.

### 10. Large Standard Library

Python downloads with a large library that you can use so you don’t have to write your own code for every single thing. There are libraries for regular expressions, documentation-generation, unit-testing, web browsers, threading, databases, CGI, email, image manipulation, and a lot of other functionality.

### 11. GUI Programming

A software is not user-friendly until its GUI is made. A user can easily interact with the software with a GUI. Python offers various libraries for making Graphical user interface for your applications. For this, you can use Tkinter, wxPython or JPython. These toolkits allow you for easy and fast development of GUI.

### 12. Dynamically Typed

Python is dynamically-typed. This means that the type for a value is decided at runtime, not in advance. This is why we don’t need to specify the type of data while declaring it.

This is all about the features of python programming language tutorial.

## Summary

Now that you know what are the features of python, you know what makes it special. In this tutorial, we learned about various features of Python. We saw that it is interpreted, dynamically-typed, and object-oriented, among other python features. It is also portable, free, and easy. Now that’s some motivation to dive into the world of Python.

Hope you like the Features of Python Programming Tutorial. ***Now!! It’s time to get yourself acquainted with the***[***advantages and disadvantages of Python***](https://data-flair.training/blogs/advantages-and-disadvantages-of-python/)***.***

If you have any queries regarding Python features article, mention in the comment section. We will be happy to help you