**What are the differences between Python and Java ?**

|  |  |  |
| --- | --- | --- |
| **Python Vs Java** | | |
| **Comparison** | **Python** | **Java** |
| Performance Speed | Fast | Not as much as Python |
| Indentation | Must be followed | Using proper flower braces is enough |
| Typing | Dynamically typed | Static typed |
| Accessability | Simple and compact | Not as much as Python |
| Platforms | Not compatible to many | Platform independent |
| Database Access | Weak compared to JAVA | Strong (JDBC) |

### Differences between Compilers and Interpreters

#### Compiler

Definition: A compiler is a computer program that transforms (translates) source code of a programming language (the source language) into another computer language (the target language). In most cases compilers are used to transform source code into executable program, i.e. they translate code from high-level programming languages into low (or lower) level languages, mostly assembly or machine code.

#### Interpreter

Definition: An interpreter is a computer program that executes instructions written in a programming language. It can either

* execute the source code directly or
* translates the source code in a first step into a more efficient representation and executes this code

**How is Python executed?**

**Whenever a python program is called, python will check, if there exists a compiled version with the .pyc suffix.**

**If such a file exists, python will load the byte code, which will speed up the startup time of the script.**

**If there exists no byte code version, python will create the byte code before it starts the execution of the program.**

**Execution of a python program means execution of the byte code on the python virtual machine (PVM).**

**What is the difference between .py and .pyc files?**

.py files are Python source files. .pyc files are the compiled bvtecode files that is generated by the Python compiler

**How do you invoke the Python interpreter for interactive use?**

**Python interpreter can be invoked just by typing “python” in the command prompt.**

**How are python blocks defined?**

**Block is a group of statements in a program or script.**

**It may contain one statement.**

**It may contain blocks called as nested blocks.**

**How does a function return values?**

Functions return values using the return statement.

**What happens when a function doesn’t have a return statement? Is this valid?**

Yes, this is valid. The function will then return a None object.

### **What is Python?**

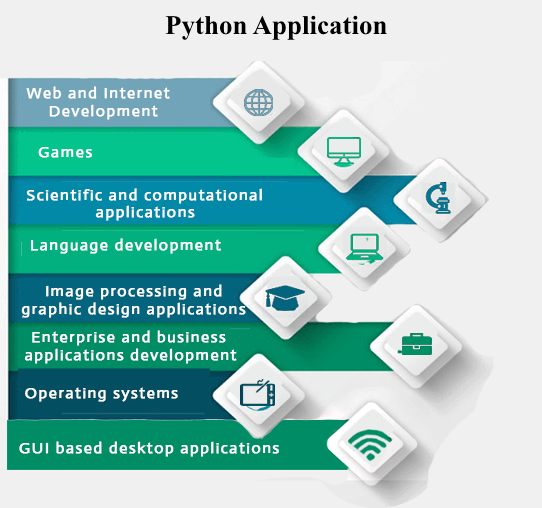
Python is a general-purpose computer programming language. It is a high-level, object-oriented language which can run equally on different platforms such as Windows, Linux, UNIX, and Macintosh. It is widely used in data science, machine learning and artificial intelligence domain.

It is easy to learn and require less code to develop the applications.

### **What are the applications of Python?**

Python is used in various software domains some application areas are given below.

* Web and Internet Development
* Games
* Scientific and computational applications
* Language development
* Image processing and graphic design applications
* Enterprise and business applications development
* Operating systems
* GUI based desktop applications



Python provides various web frameworks to develop web applications. The popular python web frameworks are **Django, Pyramid, Flask**.

Python's standard library supports for E-mail processing, FTP, IMAP, and other Internet protocols.

Python's **SciPy** and **NumPy** helps in scientific and computational application development.

Python's **Tkinter** library supports to create a desktop based GUI applications.

### **What are the features of Python?**

* Interpreted
* Free and open source
* Extensible
* Object-oriented
* Built-in data structure
* Readability
* High-Level Language
* Cross-platform  
  Interpreted: Python is an interpreted language. It does not require prior compilation of code and executes instructions directly.
* Free and open source: It is an open source project which is publicly available to reuse. It can be downloaded free of cost.
* Portable: Python programs can run on cross platforms without affecting its performance.
* Extensible: It is very flexible and extensible with any module.
* Object-oriented: Python allows to implement the Object Oriented concepts to build application solution.
* Built-in data structure: Tuple, List, and Dictionary are useful integrated data structures provided by the language.

### **What is PEP 8?**

PEP 8 is a coding convention which specifies a set of guidelines, so as to improve the readability of python code.

### **What do you mean by Python literals?**

Literals can be defined as a data which is given in a variable or constant. Python supports the following literals:

**String Literals**

String literals are formed by enclosing text in the single or double quotes. For example, string literals are string values.

**E.g.:**

"Aman", '12345'.

**Numeric Literals**

Python supports three types of numeric literals integer, float and complex. See the examples.

1. # Integer literal
2. a = 10
3. #Float Literal
4. b = 12.3
5. #Complex Literal
6. x = 3.14j

**Boolean Literals**

Boolean literals are used to denote boolean values. It contains either True or False.

1. # Boolean literal
2. isboolean = True

What is zip() function in python?

The python zip() function is used to transform multiple lists, i.e. list1, list2 and many more into a single list of tuples.

list1 **=** [**'A'**, **'B'**, **'C'**]  
list2 **=** [10, 20, 30]  
x **=** zip(list1, list2)  
print(list(x)) *# [('A', 10), ('B', 20), ('C', 30)]*

#### **Note: If the given lists are of different lengths, zip stops generating tuples when the first list ends. It means two lists are having 3, and 5 lengths will create a 3-tuple.**

### **How to overload constructors or methods in Python?**

Python's constructor: \_init\_\_ () is the first method of a class. Whenever we try to instantiate an object \_\_init\_\_() is automatically invoked by python to initialize members of an object. We can't overload constructors or methods in Python. It shows an error if we try to overload.

**class Student:  
 def \_\_init\_\_**(self, *name*)**:** self.name **=** *name* **def \_\_init\_\_**(self, *name*, *email*)**:** self.name **=** *name* self.email **=** *email*s **=** Student(**"saisk"**, **"sk@gmail.com"**)  
print(s.name)

we will get error for

s **=** Student(**"saisk"**)

we can’t overload the constructors.