Top Python Theoritical Interview Questions and Answers

1. Explain Python?

> Python is a highly comprehensive, interactive, and object-oriented scriptwriting language. It was developed to make the content highly readable among net surfers. Python makes use of various English keywords other than just punctuations.

---

2. What are the distinct features of Python?

> 1. Structured and functional programmings are supported.
> 2. It can be compiled to byte-code for creating larger applications.
> 3. Develops high-level dynamic data types.
> 4. Supports checking of dynamic data types.
> 5. Applies automated garbage collection.
> 6. It could be used effectively along with Java, COBRA, C, C++, ActiveX, and COM.

---

3. What is Python path?

> A Python path tells the Python interpreter to locate the module files that can be imported into the program. It includes the Python source library directory and source code directory.

---

4. What are the supported standard data types in Python?

> The supported standard data types in Python include the following.
>
> 1. List
> 2. Number
> 3. String
> 4. Dictionary

> 5. Tuples

5. Define tuples in Python?

> Tuples is a sequence data type in Python. The number of values in tuples are separated by commas.

6. What are the positive and negative indices?

> In the positive indices are applied the search beings from left to the right. In the case of the negative indices, the search begins from right to left. For example, in the array list of size n the positive index, the first index is 0, then comes 1 and until the last index is n-1. However, in the negative index, the first index is -n, then -(n-1) until the last index will be -1

7. What can be the length of the identifier in Python?

> The length of the identifier in Python can be of any length. The longest identifier will violate from PEP – 8 and PEP – 20.

8. Define Pass statement in Python?

> A Pass statement in Python is used when we cannot decide what to do in our code, but we must type something for making syntactically correct.

9. What are the limitations of Python?

> There are certain limitations of Python, which include the following:

> 1. It has design restrictions.
> 2. It is slower when compared with C and C++ or Java.
> 3. It is inefficient in mobile computing.
> 4. It consists of an underdeveloped database access layer.

10. Do runtime errors exist in Python? Give an example?

> Yes, runtime errors exist in Python. For example, if you are duck typing and things look like a duck, then it is considered as a duck even if that is just a flag or stamp or any other thing. The code, in this case, would be A Run-time error. For example, Print "Hackr io", then the runtime error would be the missing parenthesis that is required by print ( ).

---

11. Why do we need a break in Python?

> Break helps in controlling the Python loop by breaking the current loop from execution and transfer the control to the next block.

---

12. Why do we need a continue in Python?

> A continue also helps in controlling the Python loop but by making jumps to the next iteration of the loop without exhausting it.

---

13. Can we use a break and continue together in Python? How?

> Break and continue can be used together in Python. The break will stop the current loop from execution, while jump will take to another loop.

---

14. Does Python support an intrinsic do-while loop?

> No Python does not support an intrinsic do-while loop.

---

15. How many ways can be applied for applying reverse string?

> There are five ways in which the reverse string can be applied which include the following.
>
> 1. Loop

> 2. Recursion
> 3. Stack
> 4. Extended Slice Syntax
> 5. Reversed

---

16. What are the different stages of the Life Cycle of a Thread?

> The different stages of the Life Cycle of a Thread can be stated as follows.
>
> Stage 1: Creating a class where we can override the run method of the Thread class.
>
> Stage 2: We make a call to start() on the new thread. The thread is taken forward for scheduling purposes.
>
> Stage 3: Execution takes place wherein the thread starts execution, and it reaches the running state.
>
> Stage 4: Thread wait until the calls to methods including join() and sleep() takes place.
>
> Stage 5: After the waiting or execution of the thread, the waiting thread is sent for scheduling.
>
> Stage 6: Running thread is done by executing the terminates and reaches the dead state.

---

17. What is the purpose of relational operators in Python?

> The purpose of relational operators in Python is to compare values.

---

18. What are assignment operators in Python?

> The assignment operators in Python can help in combining all the arithmetic operators with the assignment symbol.

---

19. Why do we need membership operators in Python?

> We need membership operators in Python with the purpose to confirm if the value is a member in another or not.

20. How are identity operators different than the membership operators?

> Unlike membership operators, the identity operators compare the values to find out if they have the same value or not.

21. Describe how multithreading is achieved in Python?

> Even though Python comes with a multi-threading package, if the motive behind multithreading is to speed the code then using the package is not the go-to option.
>
> The package has something called the GIL or Global Interpreter Lock, which is a construct. It ensures that one and only one of the threads execute at any given time. A thread acquires the GIL and then do some work before passing it to the next thread.
>
> This happens so fast that to a user it seems that threads are executing in parallel. Obviously, this is not the case as they are just taking turns while using the same CPU core. Moreover, GIL passing adds to the overall overhead to the execution.
>
> Hence, if you intend to use the threading package for speeding up the execution, using the package is not recommended.

22. What is Inheritance in Python?

> Inheritance enables a class to acquire all the members of another class. These members can be attributes, methods, or both. By providing reusability, inheritance makes it easier to create as well as maintain an application.
>
> The class which acquires is known as the child class or the derived class. The one that it acquires from is known as the superclass or base class or the parent class

23. What are the different types of inheritance?

> 1. Single Inheritance – A single derived class acquires from on single superclass.

> 2. Multi-Level Inheritance – At least 2 different derived classes acquire from two distinct base classes.
> 3. Hierarchical Inheritance – A number of child classes acquire from one superclass
> 4. Multiple Inheritance – A derived class acquires from several superclasses.

---

24. Explain memory managed in Python?

> Python private heap space takes place of memory management in Python. It contains all Python objects and data structures. The interpreter is responsible to take care of this private heap and the programmer does not have access to it. The Python memory manager is responsible for the allocation of Python heap space for Python objects. The programmer may access some tools for the code with the help of the core API. Python also provides an inbuilt garbage collector, which recycles all the unused memory and frees the memory and makes it available to heap space.

---

25. What are Python decorators?

> A specific change made in Python syntax to alter the functions easily are termed as Python decorators.

---

26. What do you understand by the process of compilation and linking in Python?

> In order to compile new extensions without any error, compiling and linking is used in Python. Linking initiates only and only when the compilation is complete.
>
> In the case of dynamic loading, the process of compilation and linking depends on the style that is provided with the concerned system. In order to provide dynamic loading of the configuration setup files and rebuilding the interpreter, the Python interpreter is used.

---

27. What is the map() function used for in Python?

> The map() function applies a given function to each item of an iterable. It then returns a list of the results. The value returned from the map() function can then be passed on to functions to the likes of the list() and set().

> Typically, the given function is the first argument and the iterable is available as the second argument to a map() function. Several tables are given if the function takes in more than one arguments.

---

28. How will you distinguish between NumPy and SciPy?

> Typically, NumPy contains nothing but the array data type and the most basic operations, such as basic element-wise functions, indexing, reshaping, and sorting. All the numerical code resides in SciPy.
>
> As one of NumPy's most important goals is compatibility, the library tries to retain all features supported by either of its predecessors. Hence, NumPy contains a few linear algebra functions despite the fact that these more appropriately belong to the SciPy library.
>
> SciPy contains fully-featured versions of the linear algebra modules available to NumPy in addition to several other numerical algorithms.

---

29. What is the lambda function?

> An anonymous function is known as a lambda function. This function can have only one statement but can have any number of parameters.

---

30. Differentiate between list and tuple?

> Tuple is not mutable it can be hashed eg. key for dictionaries. On the other hand, lists are mutable.

---

31. What is the // operator? What is its use?

> The // is a Floor Divisionoperator used for dividing two operands with the result as quotient displaying digits before the decimal point. For instance, 10//5 = 2 and 10.0//5.0 = 2.0.

---

32. What is monkey patching in Python?

> The dynamic modifications made to a class or module at runtime are termed as monkey patching in Python

---

33. What is the split function used for?

> The split function breaks the string into shorter strings using the defined separator. It returns the list of all the words present in the string.

---

34. Explain the Dogpile effect?

> The event when the cache expires and websites are hit by multiple requests made by the client at the same time. Using a semaphore lock prevents the Dogpile effect. In this system when value expires, the first process acquires the lock and starts generating new value.

---

35. What is a pass in Python?

> No-operation Python statement refers to pass. It is a place holder in the compound statement, where there should have a blank left or nothing written there.

---

36. Define slicing in Python.

> Slicing refers to the mechanism to select the range of items from sequence types like lists, tuples, strings

---

37. What are docstring?

> Docstring is a Python documentation string, it is a way of documenting Python functions, classes and modules.

---

38. What is [::-1] used for?

> [::-1] reverses the order of an array or a sequence. However, the original array or the list remains unchanged.

39. Define Python Iterators

> Group of elements, containers or objects that can be traversed.

40. How are comments written in Python?

> Comments in Python start with a # character, they can also be written within docstring(String within triple quotes)

41. How to capitalize the first letter of string?

> Capitalize() method capitalizes the first letter of the string, and if the letter is already capital it returns the original string

42. What is, not and in operators?

> Operators are functions that take two or more values and returns the corresponding result.
>
> 1. is: returns true when two operands are true
> 2. not: returns inverse of a boolean value
> 3. in: checks if some element is present in some sequence.

43. How are files deleted in Python?

> To delete a file in Python:
>
> 1. Import OS module
> 2. Use os.remove() function

44. Does Python supports multiple inheritances?

> Yes, in Python a class can be derived from more than one parent class.

45. What does the method object( ) do?

> The method returns a featureless object that is base for all classes. This method does not take any parameters.

---

46. What is pep 8?

> Python Enhancement Proposal or pep 8 is a set of rules that specify how to format Python code for maximum readability.

---

47. What is namespace in Python?

> A naming system used to make sure that names are unique to avoid naming conflicts refers to as Namespace.

---

48. Is indentation necessary in Python?

> Indentation is required in Python if not done properly the code is not executed properly and might throw errors. Indentation is usually done using four space characters.

Question: Define a function in Python
Answer: A block of code that is executed when it is called is defined as a function. Keyword def is used to define a Python function.

Question: Define self in Python
Answer: An instance of a class or an object is self in Python. It is included as the first parameter. It helps to differentiate between the methods and attributes of a class with local variables.