iNeuron


# Python

## Interview Questions

Q1. What are Python packages?

Ans: Python packages are namespaces containing multiple modules.

Q2.How can files be deleted in Python?

Ans: To delete a file in Python, you need to import the OS Module. After that, you need to use the os. remove() function.

Q3. What advantages do NumPy arrays offer over (nested) Python lists?

Ans:

1. Python's lists are efficient general-purpose containers. They support (fairly) efficient insertion, deletion, appending, and concatenation, and Python's list comprehensions make them easy to construct and manipulate.
2. They have certain limitations: they don't support "vectorized" operations like elementwise addition and multiplication, and the fact that they can contain objects of differing types mean that Python must store type information for every element, and must execute type dispatching code when operating on each element.
3. NumPy is not just more efficient; it is also more convenient. You get a lot of vector and matrix operations for free, which sometimes allow one to avoid unnecessary work. And they are also efficiently implemented.

4. NumPy array is faster and You get a lot built in with NumPy, FFTs, convolutions, fast searching, basic statistics, linear algebra, histograms, etc.

Q53. How to remove values to a python array?

Ans: Array elements can be removed using pop() or remove() method. The difference between these two functions is that the former returns the deleted value whereas the latter does not.

Q4. Does Python have OOps concepts?

Ans: Python is an object-oriented programming language. This means that any program can be solved in python by creating an object model. However, Python can be treated as a procedural as well as structural language.

Q5. What is the difference between deep and shallow copy?

Ans: Shallow copy is used when a new instance type gets created and it keeps the values that are copied in the new instance. Shallow copy is used to copy the reference pointers just like it copies the values. These references point to the original objects and the changes made

in any member of the class will also affect the original copy of it. Shallow copy allows faster execution of the program and it depends on the size of the data that is used.

Deep copy is used to store the values that are already copied. Deep copy doesn't copy the reference pointers to the objects. It makes the reference to an object and the new object that is pointed by some other object gets stored. The changes made in the original copy won't affect any other copy that uses the object. Deep copy makes execution of the program slower due to making certain copies for each object that is been called.

Q6. How is Multithreading achieved in Python?

Ans:

1. Python has a multi-threading package but if you want to multi-thread to speed your code up, then it's usually not a good idea to use it.
2. Python has a construct called the Global Interpreter Lock (GIL). The GIL makes sure that only one of your 'threads' can execute at any one time. A thread acquires the GIL, does a little work, then passes the GIL onto the next thread.
3. This happens very quickly so to the human eye it may seem like your threads are executing in parallel, but they are really just taking turns using the same CPU core.
4. All this GIL passing adds overhead to execution. This means that if you want to make your code run faster then using the threading package often isn't a good idea.

Q7. What is the process of compilation and linking in python?

Ans: The compiling and linking allow the new extensions to be compiled properly without any error and the linking can be done only when it passes the compiled procedure. If the dynamic loading is used then it depends on the style that is being provided with the system. The python interpreter can be used to provide the dynamic loading of the configuration setup files and will rebuild the interpreter.

The steps that are required in this as:

1. Create a file with any name and in any language that is supported by the compiler of your system. For example file.c or file.cpp
2. Place this file in the Modules/ directory of the distribution which is getting used.
3. Add a line in the file Setup. Local that is present in the Modules/ directory.
4. Run the file using spam file.o
5. After a successful run of this rebuild the interpreter by using the make command on the top-level directory.
6. If the file is changed then run rebuildMakefile by using the command as 'make Makefile'.

Q8. What are Python libraries? Name a few of them.

Ans- Python libraries are a collection of Python packages. Some of the majorly used python libraries are – Numpy, Pandas, Matplotlib, Scikit-learn and many more.

Q9. What is split used for?

Ans- The split() method is used to separate a given String in Python.

Q10. What is the difference between range & xrange?

Functions in Python, range() and xrange(), are used to iterate inside a for loop for a fixed number of times. Functionality-wise, both these functions are the same. The difference comes when talking about the Python version support for these functions and their return values.

Q11. What is pickling and unpickling?

The Pickle module accepts the Python object and converts it into a string representation and stores it into a file by using the dump function. This process is called pickling. On the other hand, the process of retrieving the original Python objects from the string representation is called unpickling.

Q12. What do you understand by the word Tkinter?

Ans- Tkinter is a built-in Python module that is used to create GUI applications and it is Python's standard toolkit for GUI development.

Tkinter comes pre-loaded with Python so there is no separate installation needed. You can start using it by importing it in your script.

Q13. Is Python fully object oriented?

Python does follow an object-oriented programming paradigm and has all the basic OOPs concepts such as inheritance, polymorphism, and more, with the exception of access specifiers. Python doesn't support strong encapsulation (adding a private keyword before data members). Although, it has a convention that can be used for data hiding, i.e., prefixing a data member with two underscores.

Q14. **What do file-related modules in Python do? Can you name some file-related modules in Python?**

Python comes with some file-related modules that have functions to manipulate text files and binary files in a file system. These modules can be used to create text or binary files, update content by carrying out operations like copy, delete, and more.

Some file-related modules are os, os.path, and shutil.os. The os.path module has functions to access the file system, while the shutil.os module can be used to copy or delete files.

Q15. Explain the use of the 'with' statement and its syntax?

In Python, using the ‘with’ statement, we can open a file and close it as soon as the block of code, where ‘with’ is used, exits. In this way, we can opt for not using the close() method.

Q16. What does *args and **kwargs mean in Python?

Ans- *args: It is used to pass multiple arguments in a function.

**kwargs: It is used to pass multiple keyworded arguments in a function in Python.

Q17. How will you remove duplicate elements from a list?

To remove duplicate elements from the list we use the set() function.

Consider the below example:

```
demo_list = [5, 4, 4, 6, 8, 12, 12, 1, 5]

unique_list = list(set(demo_list))

output = [1, 5, 6, 8, 12]
```

Q18. How can files be deleted in Python?

Ans- You need to import the OS Module and use os.remove() function for deleting a file in python.

consider the code below:

```
import os

os.remove("file_name.txt")
```

Q19. How will you read a random line in a file?

We can read a random line in a file using the random module.

For example:

```
import random

def read_random(fname):

lines = open(fname).read().splitlines()

return random.choice(lines)

print(read_random('hello.txt'))
```

Q20. Write a Python program to count the total number of lines in a text file?

Refer the code below to count the total number of lines in a text file-

```
def file_count(fname):
    with open(fname) as f:
        for i, _ in enumerate(f):
            pass
    return i + 1

print("Total number of lines in the text file:",
file_count("file.txt"))
```