Python Programming

1. How is memory managed in Python?

It involves a private heap containing all Python objects & data structures. A raw memory allocater ensures that there is enough room in private heap for storing data by interacting with memory manager of OS.

2. Explain what Flask is and it's benefits?

There are many modules or frameworks which allows to build webpages using python like Flask etc. It provides versatility & built in development server & fast debugger. Integrated support for unit testing. Restful request dispatching. Unicode based. Support for secure cookies.

3. Whenever Python exits, why isn't all the memory de-allocated?

4. What is the usage of help() and dir() function in Python?

* help() → provides help on any topic of python
* dir() → returns all names in current scope. else it returns attributes of given object passed as parameter

5. Name some of the features of Python.

- Easy to code - Free & open source - Object oriented - High-Level language - GUI Programming support - Extensible feature - Portable - Integrated.

6. What is the difference between tuples and lists in Python?

Tuples	List
* written within ()	* written within [ ]
* Immutable in nature	* Mutable in nature
* 33 available methods	* 46 available methods
* In dictionary, we can separate create keys using tuples	* We can't use list as keys in dictionary

Scanned with CamScanner

7. How will you convert a object to a regular expression in Python?

8. What is the purpose of // operator?

Floor division

9. How can you pick a random item from a list or tuple?

choice() e.g

```
import random as r
ls = [1, 2, 3, 4]
print(r.choice(ls))
```

10. How will you remove last object from a list?

pop()

11. Explain split(), sub(), subn() methods of "re" module in Python.

* split() → "s" used for creating space in string.
* sub() → used to replace sub strings.
* subn() → replace substring & returns new string & no. of replacements but

12. What is map function in Python?

used to apply a function on all elements of sepecified iterable & return map object - map object is an iterator

Scanned with CamScanner

13. What is the output of the following? try: if '1' != 1: raise

unexpected EOF while parsing or Invalid syntax

14. Differentiate between append() and extend() methods.?

* append() adds its argument as single element at the end of list whereas extend() iterates over its arguments (another list) & add each element to the list * append() increase length by +1 but extend() by no. of elements in argument. * append() time complexity is O(1) & extend() is O(K) where K is length of arguments

15. Write a Python program to count the number of lines in a text file.

```
file_path = r"C:\Path_to_file\file.txt"; line_count = 0
with open(file_path, 'r') as f: for i in f: line_count += 1
print("Total lines =", line_count)
```

16. What are the key features of Python?

- Dynamic - Portable - Integrated - Extensible
- Object oriented - Free & open source - Easy

17. How are compile-time and run-time code checking done in Python?

Python performs some compile time checkings, but most of the checks such as type, name etc are postponed until code execution. If a code references user defined function that does not exist, the code will compile successfully but the code will fail during runtime.

18. How is Python executed?

Py Code → [Syntax checker & Translator] → Byte code ↓

Input → [Python virtual Machine (PVM)] → output ↓

Scanned with CamScanner

19. What is a module and package in Python?

* Packages are namespaces which contains multiple packages & modules themselves * Modules in Python are simply python files with .py extension which can have set of functions, classes or variables defined & implemented.

20. Explain inheritance in python with an example.

Allows us to define a class that inherits all methods & properties of another class. e.g. base class as Person (fname, lname) and sub class as Student (Person) which inherits fname & lname for Person class

21. What type of a language is python? Interpreted or Compiled?

Interpreted - processed at runtime by interpreter. We don't need to compile our prog. before executing

22. What do you mean by python being an "interpreted language"? (Continues from previous question)

It converts source code to into intermediate language which is again translated into the native lang that is executed. Interpreter of Python executes code line by line

23. What is python's standard way of identifying a block of code?

Python used indentation to identify blocks of code. Whitespace is used for indentation. All code with same distance to the right belongs to same block of code. If a block has to be more deeply nested, it is simply intended further to right.

24. Please provide an example implementation of a function called "my_func" that returns the square of a given variable "x". (Continues from previous question)

```
def myFunc(x):
    return x**2
```


Scanned with CamScanner

25. Is python statically typed or dynamically typed?

Dynamically typed language

26. Is python strongly typed or weakly typed language?

Strongly typed language

27. Create a unicode string in python with the string "This is a test string"?

28. What is the python syntax for switch case statements?

Python doesn't support switch statement, we can use if else statement for this purpose.

29. What is a lambda statement? Provide an example.

It is a function that accepts n no. of arguments, but only one expression, which is evaluated & returned.

```
def lambda (x):
    x * x * x
```

30. What are the rules for local and global variables in Python?

Global → defined & declared outside function

local → declared inside function only & can't be used outside function

Scanned with CamScanner

31. What is the purpose of #!/usr/bin/python on the first line in the above code? Is there any advantage?

We specify exactly which interpreter will be used to run the script on a particular system. Advantage is, we can use a specific python version to run your code.

32. What does this list comprehension do:
[x ** 2 for x in range (10) if x % 2 == 0]

Ref. gives sq. of even no. between 0 to 9 b'coz range value (10-1) = 9. ∴ o/p : [0, 4, 16, 36, 64].

33. Do sets, dictionaries and tuples also support comprehensions?

Set & Dictionaries support comprehension but tuples are immutable & have generators but not comprehensions.

34. What are some mutable and immutable datatypes/Data/structures in python?

List & set are mutable.
Tuple & dictionary are immutable.
String (str) also immutable.

35. What are generators in Python?

Used to control iteration behaviour of a loop.

36. What can you use Python generator functions for?

It is defined like normal function, but whenever it needs to generate a value, it does so with the yield keyword rather than return. Generators function returns a generator object.

Scanned with CamScanner

37. When is it, not a good time to use python generators?

38. What's your preferred text editor?

Spyder or Jupyter

39. When should you use generator expressions vs list comprehensions in Python and viceversa?

List comprehensions allows us to create list using for loop with lesser code. generator expressions → instead of creating the list & keeping the whole sequence in memory, it generates next element in demand. (generator)

40. What is a negative index in Python?

Represents positions from end of the array. It means its beginning from the end, -1 refers to the last item, -2 refers to second last item

41. What is the difference between range and xrange functions?

* range() - returns range object (a type of iterable).
* xrange() - returns generator object that can be used to display no. only by looping.

42. What is PEP8?

It has emerged as a style guide that most projects add here to; it promotes a very readable & eye-pleasing code.

Scanned with CamScanner

43. How can I find methods or attributes of an object in Python?

getattr() → used to access attributes of object
hasattr() → used to check if attribute exist or not

44. What is the statement that can be used in Python if a statement is required syntactically but the program requires no action?

"pass" statement can be used

45. Do you know what is the difference between lists and tuples? Can you give me an example for their usage?

List	Tuple
* Mutable	* Immutable
* Time consuming iterations	* Faster iteration
* eg ls = [1, 2, 3, 4]	* eg tup = ("This", "is", "me")

46. What is the function of "self"?

It represents instance of a class. using which we can access the attributes & methods of the class.

47. What is __init__.py?

It is a constructor of a class. which is called when an object is created from a class & also allows to initialize attributes of the class.

48. Print contents of a file ensuring proper error handling?

```
try:
    with open('filename', 'r') as f:
        print(f.read())
except FileNotFoundError:
    print("No such file")
```


Scanned with CamScanner

49. How do we share global variables across modules in Python?

By creating a config module. Just import the config module in all modules of app; the module then becomes available as global name

50. Does Python support Multithreading?

51. How do I get a list of all files (and directories) in a given directory in Python?

```
import os
os.listdir('<path>')
```

52. How to append to a string in Python?

using '+' operator

53. How to convert a string to lowercase in Python?

str.lower(<string>)

54. How to convert a string to lowercase in Python?

str.lower(<string>)

Scanned with CamScanner

55. How to check if string A is substring of string B?

A.find(B) OR B in A

56. Find all occurrences of a substring in Python

A.count(B)

57. What is GIL? What does it do? Talk to me about the GIL. How does it impact concurrency in Python? What kinds of applications does it impact more than others?

58. Print the index of a specific item in a list?

<listname>.index(<element name>)

59. How do you iterate over a list and pull element indices at the same time?

list(enumerate(<list name>))

60. How does Python's list. Sort work at a high level? Is it stable? What is the runtime?

Scanned with CamScanner

61. What does the list comprehension do?

Used for creating new lists from other iterables. As list comprehension returns list, they consists of brackets containing expression, which is executed for each element along with for loop to iterate over each elemchr

62. How can we pass optional or keyword parameters from one function to another in Python?

63. Explain the role of repr function.

It returns a printable representation of the given object

64. How do you make a higher order function in Python?

To bind a function object to specific context, you can use either nested ~~loops~~ Scopes or callable objects i.e., def linear(a,b) which returns f(x) that computes value a*x+b. e

```
def linear(a,b):
    def result(x):
        return a*x+b
    return result
```

65. What is map?

map() function is used to apply a function on all elements of specified iterable & return map object. Python map object is an iterator, so we can iterate over its elements

66. Tell me a very simple solution to print every other element of this list?

```
ls = [1, 2, 3, 4, 5, 6, 7]
ls[1::2]
```


Scanned with CamScanner