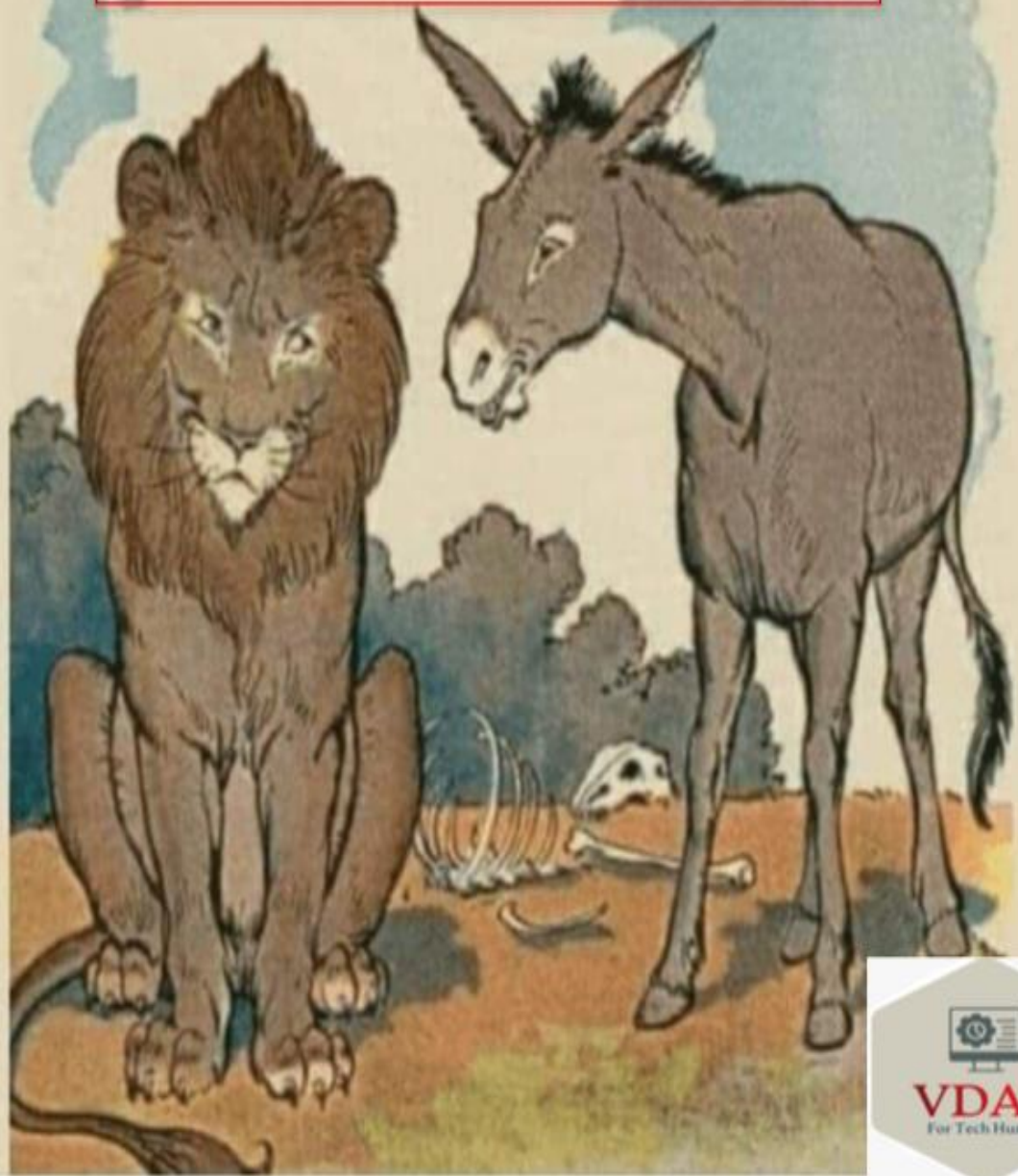


PYTHON – INTERVIEW GUIDE



VDAC
For Tech Hunter

1. Which version of python are you using?
2. What are the coding standards you follow while you are writing python code?
3. Which python IDE you are using to develop python project and what other IDEs are you familiar with?
4. What is the difference between list and tuple?
5. What do you mean by mutable and Immutable objects?
6. What are the data types can we classify as mutable objects?
7. What are the data types can we classify as immutable objects?
8. are numbers mutable or immutable objects?
9. How can python memory mapping be done?
10. What is the difference between append () and extend () functions?
11. What is shallow copying and deep copying?
12. What does the dir() function do ?
13. When to use lists and when to use tuples and which one is more efficient?
14. How to create mapping objects in python?
15. What are the sequential (data type) objects in python and what are the operations we can perform on it?
16. What are dictionaries and how can they be helpful?
17. What are the properties of dictionaries?
18. What is the difference between the operator is and the operator == ?
19. What is operator precedence and operator associativity?
20. What are the conditional statements & write down the syntax?
21. What are the iterative statements?
22. What are the loop constraints?
23. Without if - block can we write else block in python?
24. The else - statement with iterative statement when it will execute.
25. When to use for loop and when to use while loop in python?
26. What does the range () function does?
27. What was the best use case of range () function?
28. What is an iterator object?
29. What will split () and join () functions do?
30. How will you read the data from the large data file?
31. What does enumerate () function do?
32. What is the zip () function can be used for?
33. What are the default arguments and non-default arguments?
34. What are the actual arguments and formal arguments?
35. What is the variable number of arguments?
36. What are positional arguments?
37. What is the difference between local and global variables?
38. Is it possible to change global variables inside a function?
39. Why are we not able to access local variables outside of the function definition?
40. Why do we need to return the values from the function?
41. Is this mandatory to return a value from the function? If not, what will happen?
42. How to collect in case of a function returning more than one value?
43. What is the method of overloading?
44. What is call by value and call by reference?

45. Does python follow call by value or call by reference?
46. What are recursive functions and what is the limit of recursion?
47. What are nested functions?
48. What are the closures?
49. What is a decorator?
50. What is a generator?
51. What is a predicate function?
52. What is an anonymous function? And how can we define anonymous functions?
53. What is the difference between map () and filter() functions?
54. What do you mean by higher order function?
55. What os.walk() function can do ?
56. How to take command line arguments?
57. What is pickling and unpickling?
58. How will you generate random numbers?
59. How can we wait time in a program?
60. What is a python module?
61. What is the difference between python module and package?
62. What happened when the module was imported into another python file.
63. What are built-in modules, third-party modules, and user defined modules?
64. Which location of your system the build modules are available?
65. Which location of your system the third-part modules are available?
66. How to install third-party modules in your system?
67. What are the dunder attributes?
68. What is the significance of __name__?
69. What is doc string?
70. What is function prototype?
71. What is a constructor and how will we define a constructor in python?
72. What is the advantage of __init__ () method.
73. What is the significance of self-parameters?
74. What is a tostring method?
75. The difference between __str__ and __repr__?
76. Difference between class variables and instance variables?
77. Difference between instance method, static method and class method?
78. What is property?
79. What is object-oriented principles?
80. What is inheritance and the type of inheritance?
81. Does python support multiple inheritance?
82. What is MRO?
83. Is there any limitation for level of inheritance?
84. What are the access specifiers?
85. What is encapsulation?
86. How to define a setter () and getter () methods?
87. What is abstraction?
88. How to define private methods and private variables in python.
89. What is a destructor in python?
90. What is garbage collector?