Python Interview Questions 📘

Beginner to Advanced — 100 Qs to level up your prep

Core Python basics explained simply

Modules, functions & everyday coding tricks

Advanced topics
— GIL, async, metaclasses & more

Perfect for learners & interview prep

Save, revise & crack your next Python interview! 🔥

100 Python Interview Questions

Beginner Level

1. What are Python's key features?

2. What are Python's built-in data types?

3. Difference between `list` , `tuple` , `set` , and `dict` ?

4. What is Python's indentation used for?

5. What is PEP 8 and why is it important?

6. What is the difference between mutable and immutable types?

7. What are Python's numeric types?

8. What are Python's string operations?

9. How are comments written in Python?

10. What is the difference between single, double, and triple quotes?

11. What are Python keywords?

12. Explain Python variables and naming conventions.

13. What are Python operators?

14. Explain operator precedence in Python.

15. What are Python's logical operators?

16. What are Python's membership and identity operators?

17. Explain Python slicing with examples.

18. What are Python functions?

19. What are arguments in Python functions?

20. Explain default and keyword arguments.

21. What are variable-length arguments (`*args` , `**kwargs`)?

22. What are Python `docstrings` ?

23. Difference between `return` and `yield` ?

24. What is recursion in Python?

25. What is Python's input and output function?

26. Explain Python's `print()` formatting.

27. What are `f-strings` in Python?

28. Explain Python's exception handling.

29. What is `try` , `except` , `finally` ?

30. What are Python's built-in exceptions?

Intermediate Level

31. What is a Python module?

32. What is a Python package?

33. Difference between module and package?

34. What is `__init__.py` file?

35. How to import modules in Python?

36. Explain relative vs absolute imports.

37. What is Python's `sys.path` ?

38. What are Python's standard libraries?

39. Explain Python's OS module.

40. Explain Python's `shutil` module.

41. What is Python's JSON module used for?

42. Difference between JSON and Pickle?

43. What is Python's Pickle module?

44. Explain Python's `datetime` module.

45. Explain Python's `math` module.

46. What is Python's `random` module?

47. How does Python handle files?

48. What are Python file modes (`r, w, a, rb, wb`)?

49. Difference between `read()` , `readline()` , and `readlines()` ?

50. What is the `with` statement in Python?

51. What is a context manager?

52. Difference between shallow copy and deep copy?

53. What are Python's `copy` and `deepcopy` ?

54. What are Python's iterators?

55. What are Python's generators?

56. What is the difference between iterator and iterable?

57. What are comprehensions in Python?

58. Explain list comprehension with example.

59. Explain dictionary comprehension with example.

60. Explain set comprehension with example.

61. What is Python's `enumerate()` ?

62. What is Python's `zip()` ?

63. What is Python's `map()` ?

64. What is Python's `filter()` ?

65. What is Python's `reduce()` ?

66. What are Python's `lambda` functions?

67. What is Python's `sorted()` function?

68. Difference between `sort()` and `sorted()` ?

69. What is Python's `any()` and `all()` ?

70. What is Python's `min()` and `max()` ?

Advanced Level

71. What is Python's GIL (Global Interpreter Lock)?

72. Explain Python multithreading.

73. Difference between multithreading and multiprocessing?

74. What is Python's threading module?

75. What is Python's multiprocessing module?

76. What is Python's `asyncio` ?

77. Difference between concurrency and parallelism?

78. What is Python's `async` and `await` ?

79. What are Python coroutines?

80. What are Python futures and tasks?

81. What is Python's memory management?

82. What is reference counting in Python?

83. What is garbage collection in Python?

84. What are weak references in Python?

85. Explain Python's `__slots__` .

86. What is Python's `__new__` and `__init__` ?

87. What is the difference between class and instance variables?

88. What are Python's properties (`@property`)?

89. Difference between `@staticmethod` and `@classmethod` ?

90. What are metaclasses in Python?

91. What are Python descriptors?

92. What are abstract classes in Python?

93. What is multiple inheritance in Python?

94. What is MRO (Method Resolution Order)?

95. What is monkey patching in Python?

96. What are Python data classes?

97. What is Python's typing module?

98. What are type hints in Python?

99. What are Python memoryviews?

100. What are Python's best practices (PEP 8, PEP 20)?