66 Python Debugging interview questions to assess developers

Questions

1. How would you approach debugging a Python script that is running slower than expected?

2. What steps would you take if your Python program throws an unexpected exception?

3. Can you explain a situation where you had to debug a complex issue in a Python application?

4. What strategies do you use to prevent bugs in your Python code?

5. How do you handle a situation where someone else's code is causing issues in a project?

6. What tools and techniques do you find most useful for debugging Python applications?

7. Explain how you would debug a memory leak in a Python application.

8. How do you ensure your Python code is easy to debug and maintain?

9. What is a breakpoint and how do you use it in Python debugging?

10. How would you debug a recursive function that's causing a stack overflow?

11. Explain the difference between a syntax error and a runtime error in Python.

12. What is the purpose of the 'pdb' module in Python?

13. How do you use print statements effectively for debugging?

14. What is a logical error and how would you approach debugging it?

15. Explain how you would use logging instead of print statements for debugging.

16. How do you debug a Python script that's producing incorrect output?

17. What is the significance of the 'traceback' in Python error messages?

18. How would you debug a Python script that's hanging or not terminating?

19. Explain the concept of step-through debugging in Python.

20. How do you use assertions in Python for debugging?

21. What tools would you use to profile a Python script for performance issues?

22. How do you debug race conditions in multi-threaded Python programs?

23. Explain how you would use a debugger to inspect variable values during runtime.

24. What strategies do you use to debug import errors in Python?

25. How would you approach debugging a Python script that works locally but fails in production?

26. Explain the use of the 'try-except' block in debugging Python code.

27. How do you debug issues related to Python's indentation?

28. What techniques would you use to debug memory usage in a large Python application?

29. How would you approach debugging a Python script that works well on your machine but fails on the server?

30. Can you explain how you would debug an issue where a Python script is consuming too much memory?

31. What steps would you take to debug a Python application that crashes without any error messages?

32. How do you handle debugging a Python script that interacts with an API, but the API is returning unexpected results?

33. What methods do you use to debug permission issues in a Python script running on a different system?

34. Can you describe how you would debug a Python application that intermittently fails to connect to a database?

35. How would you debug a Python application that suddenly starts throwing a lot of exceptions?

36. What would you do if you suspect a third-party library is causing issues in your Python script?

37. How do you handle debugging a Python script that interacts with a large dataset and is running out of memory?

38. Can you explain how you would debug a Python script that is supposed to run as a scheduled task but never starts?

39. How would you debug a Python script that works fine in development but fails in production due to environment differences?

40. Explain your approach to debugging a Python application that's experiencing race conditions in a multi-threaded environment.

41. What techniques would you use to debug a Python script that's causing unexpected high CPU usage?

42. How would you approach debugging a Python application that's experiencing intermittent network-related errors?

43. Describe your strategy for debugging a Python script that's producing inconsistent results with large datasets.

44. How would you debug a Python application where a specific function is occasionally returning None unexpectedly?

45. What steps would you take to debug a Python script that's experiencing unexpected behavior due to circular imports?

46. Explain your approach to debugging a Python application where certain API calls are timing out sporadically.

47. How would you debug a Python script that's exhibiting different behavior on different operating systems?

48. What techniques would you use to debug a Python application that's experiencing subtle floating-point calculation errors?

49. How would you approach debugging a Python script that's failing silently without raising exceptions?

50. Describe your strategy for debugging a Python application where a specific module is occasionally not being imported correctly.

51. How would you debug a Python script that's producing inconsistent results with different input sizes?

52. Explain how you would debug a Python application where certain API calls are failing intermittently.

53. How would you approach debugging a Python script that's causing unexpected high CPU usage?

54. How would you debug a Python application where a specific function is occasionally returning None unexpectedly?

55. How would you debug a Python script that works fine in development but fails in production due to environment differences?

56. How would you approach debugging a Python application that's experiencing race conditions in a multi-threaded environment?

57. How would you debug a Python script that's failing silently without raising exceptions?

58. How would you debug a Python script that's producing different results on Windows and Linux systems?

59. You're working on a Python project and notice that a certain function occasionally returns unexpected results. How would you go about investigating this issue?

60. How would you debug a Python script that's running out of memory when processing large datasets?

61. You're debugging a Python application and suspect there might be a deadlock in a multi-threaded section. How would you investigate and resolve this issue?

62. How would you approach debugging a Python script that's exhibiting non-deterministic behavior?

63. You're working on a Python project where certain API calls are failing intermittently. How would you go about debugging this issue?

64. How would you debug a Python script that's producing floating-point calculation errors?