

71 API Testing interview questions to ask candidates

Questions

1. How would you approach testing an API that lacks documentation?
2. What strategies would you use to test the performance of an API?
3. How would you ensure the security of an API during testing?
4. How do you approach testing API versioning?
5. What challenges have you faced when testing RESTful APIs, and how did you overcome them?
6. Can you explain what an API is in simple terms?
7. What is the difference between REST and SOAP APIs?
8. How do you validate the response of an API?
9. What tools do you prefer for API testing and why?
10. How would you handle an API that returns a 500 Internal Server Error?
11. Describe the process of writing test cases for an API.
12. What are the common status codes you might encounter in API testing?
13. How do you test the negative scenarios for an API?
14. What is JSON and how is it used in API testing?
15. Explain what a mock server is and how you would use it in testing.
16. How do you test API endpoints that require authentication?
17. What is the difference between PUT and POST methods in an API?
18. How do you handle rate limiting while testing APIs?
19. Can you explain what API throttling is?
20. How do you verify data integrity during API testing?
21. What is an API gateway and what role does it play?
22. How do you handle deprecated API endpoints in your tests?
23. What is the importance of idempotency in API testing?
24. How would you approach testing an API with inconsistent or unreliable responses?
25. Explain the concept of API contract testing and its benefits.
26. How would you test API pagination functionality?
27. What strategies would you use to test API rate limiting?
28. How would you approach testing an API that uses OAuth 2.0 for authentication?
29. Explain the concept of idempotency in API testing and why it's important.
30. How would you test an API's response time and throughput?
31. What approach would you take to test API versioning?
32. How would you test error handling and edge cases in an API?
33. Explain the concept of API mocking and how you would use it in testing.
34. How would you design a scalable API testing framework for a microservices architecture?
35. Explain your approach to testing GraphQL APIs and how it differs from REST API testing.
36. Describe a situation where you had to optimize API test execution time. What strategies did you employ?
37. How would you test an event-driven API that uses webhooks?
38. What techniques would you use to test API race conditions and concurrency issues?
39. How do you approach testing API backward compatibility when new versions are released?
40. Explain your strategy for testing an API that interacts with a third-party payment gateway.
41. How would you test an API that uses gRPC for communication?
42. Describe your approach to testing API caching mechanisms and their effectiveness.
43. How would you design tests for an API that needs to handle high volumes of real-time data?
44. What strategies would you employ to test an API's resilience to network failures and latency?
45. How do you approach testing APIs that use JWT (JSON Web Tokens) for authentication?
46. Explain your method for testing API rate limiting and throttling mechanisms.
47. How would you test an API that uses server-sent events for real-time updates?
48. Describe your approach to testing an API's ability to handle large payloads efficiently.
49. How would you design test cases for an API with multiple interdependent endpoints?
50. What strategies would you use to create test cases for API parameter combinations?
51. How do you approach creating test cases for API error handling and edge cases?
52. How would you design test cases to verify the consistency of an API across different versions?
53. What considerations would you keep in mind when designing test cases for API performance?
54. How would you approach designing test cases for an API that integrates with third-party services?
55. How do you approach testing an API when you expect it to fail?
56. What steps do you take to test an API's error responses?
57. How would you handle and test an API that returns inconsistent error messages?
58. How do you ensure that an API handles rate limiting errors correctly?
59. What methods do you use to test an API's response to unexpected inputs?
60. How would you verify that an API's error handling meets specified requirements?
61. How do you approach testing an API's response to timeouts?
62. What strategies do you use to test the resilience of an API to network failures?
63. How would you handle a situation where the API documentation is incomplete or ambiguous?
64. Describe a scenario where you had to test an API under high load. How did you approach it?
65. How do you test an API's compatibility with different environments, such as staging and production?
66. What approach would you take to test an API's integration with a third-party service?
67. How would you handle testing an API that has frequent updates and changes?