

76 Swift interview questions to ask your applicants

Questions

1. Can you explain what optional types in Swift are and why they are useful?
2. What is the difference between 'let' and 'var' in Swift?
3. How would you explain the concept of 'protocols' in Swift?
4. Can you describe how error handling works in Swift?
5. What are closures in Swift and how are they used?
6. How do you manage memory in Swift?
7. What is the role of a delegate in Swift?
8. How does Swift handle concurrency, and can you give an example?
9. What are some key differences between Swift and Objective-C?
10. Explain the concept of 'type inference' in Swift.
11. What are extensions in Swift, and how do you use them?
12. Describe what 'lazy properties' are and when you might use them.
13. How do you handle JSON data in Swift?
14. Can you explain what 'guard' statements do in Swift?
15. What is the role of 'didSet' and 'willSet' in Swift?
16. How would you implement a singleton pattern in Swift?
17. Describe the model-view-controller (MVC) pattern and its use in iOS development.
18. What are generics in Swift? Can you provide an example?
19. How do you achieve thread safety in Swift?
20. Explain what 'enum' is in Swift and provide an example use case.
21. What is the difference between a class and a struct in Swift?
22. Can you explain the concept of 'mutability' in Swift?
23. What are higher-order functions in Swift? Can you give an example?
24. How would you debug a Swift application?
25. What is the purpose of 'defer' in Swift?
26. Explain what 'weak' and 'unowned' references are and when you would use them.
27. What are Swift Playgrounds, and how can they be useful?
28. What are tuples in Swift and how would you use them?
29. How does Swift ensure type safety?
30. What is the difference between weak and strong references in Swift?
31. What is the purpose of using 'lazy' properties in Swift?
32. How does Swift handle version compatibility when updating to a new version?
33. What are optionals in Swift and how do they enhance safety in your code?
34. How do you handle asynchronous tasks in Swift?
35. What is the significance of 'Protocol-Oriented Programming' in Swift?
36. How do you go about designing a responsive and efficient user interface in Swift?
37. How does Swift's memory management work, and what are some common pitfalls?
38. Can you explain the concept of 'value types' and 'reference types' in Swift?
39. What is the difference between a 'strong' and 'weak' reference cycle?
40. How do you implement custom operators in Swift?
41. Can you demonstrate how to use 'associated types' in Swift protocols?
42. How would you create a custom collection in Swift?
43. What are property wrappers in Swift and when would you use them?
44. How do you handle multi-threading in Swift using GCD (Grand Central Dispatch)?
45. Can you explain the difference between synchronous and asynchronous tasks in Swift?
46. Describe how you would handle dependency injection in a Swift application.
47. How do you implement and use result types in Swift?
48. What are context-sensitive keywords in Swift and how are they used?
49. Can you explain the purpose and use of 'dynamic' in Swift?
50. How would you optimize a Swift application for performance?
51. Describe the process of handling large data sets in Swift using Core Data.
52. How do you manage code modularity and organization in Swift projects?
53. How would you explain ARC (Automatic Reference Counting) in Swift?
54. What are strong, weak, and unowned references in Swift, and when would you use each?
55. Can you describe what a retain cycle is and how to avoid it in Swift?
56. What is a memory leak, and how can you detect and fix it in a Swift application?
57. How does Swift handle memory management for value types vs. reference types?
58. Can you explain the concept of 'deinit' in Swift and when it should be used?
59. What is the difference between 'copy-on-write' and manual copying, and how does Swift implement 'copy-on-write'?
60. How do you manage large data sets in memory within a Swift application?
61. Can you explain what a protocol is in Swift and provide an example scenario where it might be used?
62. How do delegates work in Swift, and why are they useful?
63. What is the difference between a delegate and a closure in Swift?
64. How would you implement a protocol with optional methods in Swift?
65. What is protocol inheritance and how does it work in Swift?
66. How can protocols be used to achieve polymorphism in Swift?
67. Can you explain how protocol extensions work in Swift?
68. What is the role of associated types in Swift protocols?
69. How do you conform to multiple protocols in Swift, and what are the benefits?
70. How would you handle a situation where a new feature you implemented caused the app to crash unexpectedly?
71. How would you prioritize tasks when working on multiple features simultaneously?
72. How do you handle user feedback that suggests a feature is not intuitive?
73. Describe a situation where you had to refactor a large portion of your codebase. How did you approach it?
74. How do you stay updated with the latest Swift developments and best practices?
75. How do you handle performance issues in a Swift application?
76. Describe a time when you had to work with a difficult team member on a Swift project. How did you handle it?