64 .NET interview questions to ask your candidates

Questions

1. Can you explain what the .NET Framework is and its main components?

2. What is the difference between managed and unmanaged code in .NET?

3. How does the Common Language Runtime (CLR) work in .NET?

4. Can you describe the purpose of the Global Assembly Cache (GAC) in .NET?

5. What are assemblies in .NET, and how are they different from namespaces?

6. Can you explain what LINQ is and provide an example of its usage?

7. How does memory management work in .NET, particularly with garbage collection?

8. What are the main differences between .NET Core and .NET Framework?

9. Can you describe how exception handling is implemented in .NET?

10. What is ASP.NET, and how does it differ from standard .NET applications?

11. Can you explain the purpose of the Common Type System (CTS) in .NET?

12. What is the role of the .NET Standard?

13. Can you describe the Model-View-Controller (MVC) pattern and its importance in ASP.NET?

14. What is the difference between a Class and an Object in .NET?

15. What is the purpose of the Base Class Library (BCL) in .NET?

16. Can you explain the concept of Delegates in .NET and their use cases?

17. What is the role of the Just-In-Time (JIT) compiler in .NET?

18. How would you explain the concept of boxing and unboxing in .NET?

19. Can you explain the concept of asynchronous programming in .NET and provide a scenario where it's useful?

20. How do you implement dependency injection in .NET applications?

21. What is the difference between an abstract class and an interface in .NET?

22. How would you handle state management in ASP.NET applications?

23. Can you describe the process of creating and using custom attributes in .NET?

24. How do you optimize the performance of a .NET application?

25. What is reflection in .NET and how can it be used?

26. Can you explain the role of NuGet packages in .NET development?

27. How do you secure a .NET application, particularly an ASP.NET application?

28. What are action filters in ASP.NET MVC and how do you use them?

29. Can you describe the difference between synchronous and asynchronous methods in .NET?

30. How do you manage configuration settings in .NET applications?

31. What are extension methods in C#, and how are they useful?

32. How would you implement logging in a .NET application?

33. Can you explain the purpose and use of generics in .NET?

34. How do you manage versioning in .NET applications?

35. Can you describe how you would implement error logging in a .NET application?

36. How would you handle concurrency issues in a .NET application?

37. What strategies do you use to optimize the performance of a .NET application?

38. How do you ensure the security of a .NET application?

39. What is your approach to designing scalable .NET applications?

40. How do you manage dependencies in .NET projects?

41. Can you explain how you handle data access in .NET applications?

42. What are your strategies for ensuring code quality in a .NET project?

43. How do you use the Entity Framework in a .NET application?

44. Can you explain the concept of Middleware in ASP.NET Core?

45. What is the purpose of the IHttpContextAccessor interface in .NET?

46. How do you implement authentication and authorization in an ASP.NET Core application?

47. What are the different types of dependency injection lifetimes in .NET Core?

48. How do you use AutoMapper in a .NET application?

49. Can you explain the role of Kestrel in .NET Core applications?

50. What are Razor Pages in ASP.NET Core, and when would you use them?

51. How do you handle cross-origin requests (CORS) in .NET Core?

52. What is the purpose of the IServiceProvider interface in .NET Core?

53. How do you implement and use a custom middleware in ASP.NET Core?

54. What are the different types of filters in ASP.NET Core MVC, and how are they used?

55. Can you describe how you implement SOLID principles in your .NET projects?

56. What techniques do you use to ensure code maintainability in .NET applications?

57. How do you handle version control and source code management in .NET projects?

58. Can you explain how you use dependency injection to improve code quality in .NET?

59. What are some best practices for using asynchronous programming in .NET?

60. How do you manage and mitigate technical debt in .NET development?

61. What strategies do you employ to ensure scalability in .NET applications?

62. How do you approach unit testing and integration testing in .NET?

63. Can you describe the importance of using interfaces and abstractions in .NET?

64. What are some best practices for error handling and logging in .NET applications?