

## 47 .NET Core interview questions and answers to hire top developers

### Questions

---

1. What are the main benefits of using .NET Core compared to the .NET Framework?
2. Can you explain what dependency injection is and why it is useful in .NET Core?
3. How does .NET Core handle memory management?
4. What is Kestrel, and why is it used in .NET Core?
5. Can you explain what middleware is in .NET Core and its role in request processing?
6. What is the role of the Startup class in a .NET Core application?
7. What are some common uses for configuration files in .NET Core?
8. How do you handle exceptions in .NET Core?
9. What is the difference between a .NET Core application and a .NET Framework application in terms of cross-platform compatibility?
10. Can you explain the purpose of the appsettings.json file in a .NET Core project?
11. How do you set up dependency injection in .NET Core applications?
12. What is the significance of the 'ConfigureServices' method in the Startup class?
13. Can you describe the concept of routing in ASP.NET Core?
14. What are Razor Pages, and how do they differ from MVC in .NET Core?
15. How does .NET Core support logging, and what are some common logging providers you can use?
16. What is the purpose of Entity Framework Core in .NET applications, and how does it differ from traditional ADO.NET?
17. Can you explain what a NuGet package is and how to manage them in a .NET Core project?
18. What is the role of the 'LaunchSettings.json' file in a .NET Core application?
19. Can you explain the role of the 'ServiceProvider' in .NET Core dependency injection?
20. What is the significance of the 'Configure' method in the Startup class?
21. How does .NET Core handle cross-platform development?
22. What are the different types of service lifetimes in .NET Core?
23. Can you explain what 'self-contained deployment' means in .NET Core?
24. How do you implement logging in a .NET Core application?
25. What is 'Health Checks' in .NET Core, and how do you set it up?
26. Can you explain the purpose of 'Hosting Environment' in .NET Core?
27. What are 'Tag Helpers' in ASP.NET Core, and how do they enhance development?
28. How do you handle configuration settings in a .NET Core application?
29. Can you explain the process of creating and managing background tasks in .NET Core?
30. How do you implement asynchronous programming in .NET Core, and what are its benefits?
31. What is the role of the Task Parallel Library (TPL) in .NET Core applications?
32. Can you describe how to use the IHostedService interface for background services in .NET Core?
33. What approaches can you take for handling long-running tasks in a .NET Core application?
34. How do you manage cancellation tokens in .NET Core, and why are they important?
35. Can you explain the concept of task scheduling in .NET Core?
36. How do you handle thread safety when working with shared resources in .NET Core?
37. What are the differences between async/await and traditional threading in .NET Core?
38. How can you implement a retry mechanism for failed tasks in a .NET Core application?
39. Can you explain what 'Just-In-Time' compilation means in the context of .NET Core?
40. What is the Common Language Runtime (CLR) in .NET Core and what role does it play?
41. Can you describe what 'boxing' and 'unboxing' mean in .NET Core?
42. What is the Global Assembly Cache (GAC) and how is it different in .NET Core compared to .NET Framework?
43. What is the purpose of the 'IDisposable' interface in .NET Core?
44. Can you explain what a 'delegate' is in .NET Core and provide an example of when you might use one?
45. What is the difference between 'ref' and 'out' parameters in .NET Core?
46. Can you explain what 'reflection' means in .NET Core and provide an example of when it might be useful?
47. What is the difference between 'const' and 'readonly' in .NET Core?