

150 .NET Interview Questions

Level up your .NET career with this curated list of 150 essential interview questions covering C#, ASP.NET Core, EF Core. Each question is crafted to help you understand how real systems work — not just prepare for theory. Whether you're aiming for junior, middle, or senior roles, this collection gives you the clarity and confidence needed to pass interviews. Use it to strengthen your fundamentals, sharpen your thinking, and accelerate your growth as a .NET developer.

Hello there!

I'm Anton Martyniuk — a Microsoft MVP and a Technical Lead

I help .NET developers ship robust, scalable systems faster.

-  I have over 10 years of hands-on experience in .NET development and architecture.
-  I've dedicated my career to empowering developers to excel in building robust, scalable systems.
-  Over the years, I've mentored countless developers and shared actionable insights through my blog and LinkedIn
-   Microsoft MVP since 2025



by **Anton Martyniuk** | Microsoft MVP | Anton DevTips

Table of Contents

Contents

150 .NET Interview Questions	1
Table of Contents	2
ASP.NET Core	3
EF Core	6
C#	8

ASP.NET Core

1. Explain how routing works in ASP NET Core
2. What is middleware and in what order do they execute?
3. How can you stop other middlewares from executing?
4. What is the difference between MVC and Razor Pages?
5. Name 3 ways to create middleware
6. Explain how appsettings.json configuration layering works
7. What is the diff between Singleton, Scoped, Transient services?
8. How to use Scoped service inside a Singleton service?
9. How to execute code when application is starting and stopping?
10. What is Background Service?
11. Name a few ways to read data from appsettings.json configuration
12. What is the Options Pattern?
13. Name the use cases for using IOptionsSnapshot and IOptionsMonitor
14. How to validate configuration?
15. What is the difference between DataAnnotations and FluentValidation?
16. Explain action filters and how you might use them with controllers?
17. Why Minimal APIs are faster than Controllers?
18. How to add authorization to the project?
19. How to add authorization to all Controller's methods except one?
20. How would you implement Log-In functionality?
21. Explain how JWT tokens work
22. Explain Refresh tokens and how they work

23. How would you implement an application that allows access to certain resources if a user has specific permissions?
24. What is HostedService used for?
25. Explain the difference between PeriodicTimer and await Task.Delay()
26. What is HSTS?
27. How to return a file from API Endpoint?
28. How to accept the file by API Endpoint?
29. How to access query string parameters in API Endpoint?
30. How to get current logged in user information?
31. How to inject dependencies in Minimal APIs?
32. How would you structure your Minimal API endpoints?
33. What is Output Caching?
34. Difference between IMemoryCache and IDistributedCache?
35. Explain how HybridCache/FusionCache works?
36. What Caching patterns do you know?
37. What is Rate Limiting used for and what types do you know?
38. How to invalidate data in OutputCache?
39. How to implement API versioning?
40. How to add API versioning to the existing API if you're not allowed to change the API URLs?
41. What is Swagger used for?
42. How to add documentation of endpoints, models and fields in Swagger?
43. How to get a connection string from the configuration?
44. How can you deploy ASP NET Core Application?

45. How to configure logging in ASP NET Core?
46. What are the IHostApplicationLifetime events and how can you attach callbacks to each?
47. How can you stop the BackgroundService from execution?
48. What are the default return types you can use in controller actions?
49. What is model binding and how does it work in ASP.NET Core controllers?
50. Explain the difference between [FromBody], [FromQuery], [FromRoute], and [FromForm] attributes.
51. What is a custom model-binder class and how do you implement one?
52. How do you enable or disable a controller or action? Which attribute is used?
53. What is Method Injection in Controller and how does it differ from constructor injection?
54. How do you implement validation in controller actions? How do you handle invalid models?
55. Explain the lifecycle of a controller in ASP.NET Core. When is it instantiated and disposed?
56. What's the difference between ControllerBase and Controller? When would you use each?
57. How can you handle exceptions globally for all controllers?
58. What is structured logging and why should you use it?
59. How can you log request and response data for incoming HTTP requests?
60. How would you handle sensitive data in logs to ensure security and compliance (e.g., GDPR)?

EF Core

Junior Level (10 Questions)

1. What is a migration in EF Core?
2. How do you apply pending migrations to a database?
3. What is the difference between Eager Loading, Explicit Loading and Lazy loading?
4. What are navigation properties and why are they important?
5. What is the difference between value types and reference types in EF Core mappings?
6. How do you configure relationships (one-to-many, many-to-many) in EF Core models?
7. How do you configure one-to-many relationships in EF Core?
8. How does EF Core handle cascade delete behavior?
9. What is the difference between tracked and untracked entities?
10. What does AsNoTracking do and when should you use it?
11. What are shadow properties in EF Core?
12. How does EF Core handle concurrency conflicts?
13. How do you configure indexes using the Fluent API?
14. What is the difference between FirstOrDefault and SingleOrDefault in EF queries?
15. What is the purpose of AsSplitQuery?
16. How does EF Core translate LINQ expressions into SQL?
17. What are global query filters and when are they used?
18. What is the difference between SaveChanges and SaveChangesAsync?
19. How does EF Core handle enum properties in entities?
20. What are owned types in EF Core?
21. How do you seed data using EF Core migrations?
22. What is the difference between client-side and server-side evaluation?
23. How do you execute raw SQL queries in EF Core?
24. How do you design an aggregate root in a DDD-based EF Core model?
25. How do you optimize EF Core queries for high-traffic read workloads?

26. What strategies do you use to reduce N+1 queries in EF Core?
27. How do compiled queries improve performance in EF Core?
28. What problems arise from using AutoInclude and when should it be avoided?
29. How do you implement soft delete in EF Core?
30. What is the difference between Table-per-Hierarchy, Table-per-Type and Table-per-Concretetype-Type inheritance ?
31. How do you implement optimistic concurrency tokens in EF Core?
32. How does connection pooling behave in EF Core?
33. How do you use interceptors to customize EF Core behavior?
34. What is the difference between tracking graphs and attaching detached entities?
35. Do you use Repository pattern with EF Core?
36. What is the UnitOfWork pattern and how is it related to EF Core?
37. How to start a transaction in EF Core?
38. How do you handle transactional boundaries across multiple DbContexts?
39. How do you implement read/write separation using multiple DbContexts (CQRS)?
40. How do you apply migrations in production?

1. Difference between readonly vs const
2. What is a sealed keyword used for?
3. Name all the access modifiers for types
4. Difference between interface and abstract class
5. When is a static constructor called?
6. How to create an extension method?
7. Does C# support multiple class inheritance?
8. Explain boxing and unboxing
9. What is heap and stack?
10. Difference between string and StringBuilder
11. How to create a date with a specific timezone?
12. How to change current culture?
13. What is the difference between HashSet and Dictionary?
14. What is the purpose of the method ToLookup?
15. Does LINQ Cast<T> method create a new object?
16. Explain deferred execution in LINQ
17. How does the ImmutableList work?
18. What are the benefits of using Frozen collections?
19. Name thread safe collections
20. How to perform a lock for asynchronous code?
21. Name all the ways for creating a new thread
22. How to execute multiple async tasks at once?

23. Explain Inheritance vs Composition
24. Difference between class vs record vs struct
25. What are ref structs used for?
26. Name the two forms of records
27. What is "with" keyword used for?
28. What is the purpose of Primary Constructors?
29. Explain how Nullable Reference Types work
30. Do switch expressions have any return type limitations?
31. What is yield return used for?
32. How many generations does the Garbage Collector have?
33. What is Interlocked class user for?
34. What is the code generated by compiler for auto properties?
35. How is Polymorphism implemented in C# ?
36. How is Encapsulation implemented in C# ?
37. What is the difference between ref and out parameters?
38. How does the using statement work ?
39. What is delegate and how is it used?
40. Explain method overloading and overriding
41. Difference between IEnumerable and IQueryable
42. What are expression trees in LINQ?
43. How does exception handling work in C#?
44. Name all the ways to rethrow an exception
45. Explain the generics

- 46. What is the difference between Auto and Manual reset events?
- 47. What are Channels in C#?
- 48. Difference between volatile and Interlocked
- 49. Explain how Async Streams work
- 50. Difference between Task(.).Run and TaskFactory(.).StartNew