C#.NET Frequently Asked Questions

Concepts Included: Fundamentals, object-oriented concepts, collections, LINQ, exception handling, etc..

Basics of C# (1–15)

1. What is C#? How is it different from Java?
2. What are value types and reference types in C#?
3. What is the difference between `==` and `.Equals()` in C#?
4. Explain the difference between `const`, `readonly`, and `static` in C#.
5. What is the purpose of the `var` keyword?
6. What are nullable types in C#? How are they declared?
7. What are the different access modifiers in C#?
8. What is the difference between `break`, `continue`, and `return` statements?
9. What are string interpolation and verbatim strings in C#?
10. What is boxing and unboxing in C#?
11. Explain the difference between `Array` and `ArrayList`.
12. What is the difference between `ref` and `out` parameters?
13. What are implicit and explicit conversions in C#?
14. What is an enum in C# and how is it used?
15. What is a structure in C# and how is it declared?

Object-Oriented Programming in C# (16–30)

16. What is the difference between a class and an object in C#?
17. What is inheritance in C#? Does C# support multiple inheritance?
18. What is the difference between an abstract class and an interface?
19. What is method overloading and method overriding?
20. What is polymorphism in C#? Provide an example.
21. What is encapsulation? How is it achieved in C#?
22. Explain the `sealed` keyword in C#.
23. What is the purpose of the `base` keyword?
24. What is the use of `virtual`, `override`, and `new` keywords?
25. What are constructors? What is a static constructor?
26. What is destructor in C#? When is it called?
27. What is the difference between shallow copy and deep copy?
28. What is the purpose of the `this` keyword?
29. Can a class implement multiple interfaces? Explain.
30. What are extension methods?

tnrao.trainer@gmail.com

Collections and Generics (31–38)

31. What are generics in C#? Why are they useful?
32. What is the difference between `List<T>` and `ArrayList`?
33. What is a dictionary in C#? How do you use it?
34. What are the main differences between `IEnumerable`, `ICollection`, and `IList`?
35. How do you sort a list of custom objects in C#?
36. What are hashsets and what are they good for?
37. What are queues and stacks in C#? Give examples.
38. What is a `SortedList` and how is it different from a `Dictionary`?

LINQ (39–43)

39. What is LINQ? What are its benefits?
40. What is the difference between `Select()` and `SelectMany()`?
41. How does `Where()` work in LINQ?
42. What is deferred execution in LINQ?
43. What is the difference between `First()`, `FirstOrDefault()`, `Single()`, and `SingleOrDefault()`?

Exception Handling & Misc (44–50)

44. How is exception handling done in C#?
45. What is the difference between `throw` and `throw ex`?
46. What are custom exceptions and how do you create them?
47. What is the `using` statement in C#?
48. How does garbage collection work in C#?
49. What is asynchronous programming? Explain `async` and `await`.
50. What are delegates in C#?