C# Questions & Answers

1. What is C#?

- C# is an object-oriented programming language developed by Microsoft.

2. What is the .NET framework?

- A software framework for building applications on Windows.

3. What are the key features of C#?

- Strong typing, object-oriented, garbage collection, and exception handling.

4. What is the difference between a struct and a class?

- A struct is a value type, while a class is a reference type.

5. What is an interface in C#?

- An interface is a contract that classes must implement.

6. What is polymorphism?

- The ability to have different implementations of the same method.

7. What is encapsulation?

- Encapsulation is bundling data and methods together.

8. What is inheritance?

- Inheritance allows a class to derive properties from another class.

9. What is an abstract class?

- A class that cannot be instantiated and may contain abstract methods.

10. What is the difference between `const`, `readonly`, and `static`?

- `const` is compile-time constant, `readonly` is runtime constant, and `static` belongs to the class.

11. What is LINQ?

- LINQ (Language Integrated Query) is used to query collections.

12. What is the difference between `ref` and `out`?

- `ref` requires initialization, while `out` does not.

13. What is the purpose of `async` and `await`?

- They enable asynchronous programming.

14. What is a delegate?

- A delegate is a type that represents references to methods.

15. What is an event in C#?

- An event is a delegate used to provide notifications.

16. What is garbage collection?

- Automatic memory management that removes unused objects.

17. What is the difference between `Dispose()` and `Finalize()`?

- `Dispose()` is manually called, while `Finalize()` is used by the garbage collector.

18. What is the difference between value types and reference types?

- Value types store data, while reference types store references to data.

19. What is the purpose of `using` statement?

- It ensures the disposal of resources.

20. What are extension methods?

- Methods that allow adding functionality to existing types.

21. What is dependency injection?

- A design pattern that provides dependencies instead of creating them.

22. What is `var`, `dynamic`, and `object`?

- `var` is inferred at compile time, `dynamic` at runtime, and `object` is the base type.

23. What is a sealed class?

- A sealed class cannot be inherited.

24. What is boxing and unboxing?

- Boxing converts a value type to an object, and unboxing extracts the value.

25. What is a nullable type?

- A value type that can also hold null (`int? x = null`).

26. What is the difference between `==` and `Equals()`?

- `==` compares references, while `Equals()` compares values.

27. What is a partial class?

- A class that can be split across multiple files.

28. What is the difference between `throw` and `throw ex`?

- `throw` preserves the stack trace, while `throw ex` resets it.

29. What is the difference between `Stack` and `Queue`?

- `Stack` follows LIFO, while `Queue` follows FIFO.

30. What is a tuple?

- A data structure that groups multiple values into a single unit.