**C#**

1. Oops - Interface & abstract class, 2 Interface question, Abstraction & encapsulation, Virtual methods, Diffrence between Const & Readonly, pure abstract class, Protected & internal, Polymorphism
2. What is Boxing/Unboxing -> N
3. What is Managed or Unmanaged Code -> N
4. Can “this” be used within a static method -> N
5. What is extension method in c# and how to use them?
6. what is the difference between dispose and finalize methods in c#?
7. Var & dynamic -> N
8. IQuerable and IEnumerable -> N later to be discussed
9. Exception Program --N
10. Generics -> N
11. SOLID principles -> N
12. Dependency injection? Where to handle it? -> N
13. state management in asp.net
14. page life cycle
15. scaffolding
16. dependency injection
17. Code maintainance
18. What are delegates in C# and how will you use them in your projects?
19. What is the relation between delegates and events?
20. Do you know what is reflection in C#?Please explain with the help of an example?
21. Have you done any Muti-threading in your application?
22. Oops - Interface & abstract class, 2 Interface question, Abstraction & encapsulation, Virtual methods, Diffrence between Const & Readonly,
23. pure abstract class, Protected & internal, Polymorphism
24. What is Boxing/Unboxing -> N
25. Dependency injection? Where to handle it? -> N
26. access mofiers

|  |
| --- |
| 1. what is delegates |
| 1. relation between delegates and events |
| 1. reflection in C# give a example |
| 1. multithreading concept |

1. array list
2. encapsulation can be by using access modifiers
3. can we create constructor for abstract class
4. Is the ActionResult is abstract class
5. Suppose we can inherited 2 interface in one class and we have same method in both the interface then how the class will know which method we have to call
6. Can if loop have multiple catch statement

lazy loading and egar loading

https://www.c-sharpcorner.com/article/eager-loading-lazy-loading-and-explicit-loading-in-entity-framework/