1. What are the six combinations of access modifier keywords and what do they do?
Ans: public, protected, internal, private, static, sealed
Public, protected, internal, private are used to limit the right of accessing class.
Sealed is used to make class can not be inherited, if use in method, it's made the method cannot be overridden.
Static members are shared across all instances of a class is used to make the class and method can be called by its class name without instantiation.
2.
What is the difference between the static, const, and readonly keywords when applied to a type member?

Ans: Static members are shared across all instances of a class, while const members are compile-time constants that are implicitly static and cannot be changed at runtime. readonly members can be initialized at runtime but cannot be changed thereafter, and they are instance-specific.

3. What does a constructor do?
It's a special function help us to initialize object.

4. Why is the partial keyword useful?
It's a keyword that can be used to modifier the class, struct, interface or method allowing them to be defined in multiple parts.

5. What is a tuple?

A tuple is a data structure that groups together a fixed set of values of different types into a single object. Tuples are useful for returning multiple values from a method, grouping and sorting data, and other scenarios where a collection of values of different types need to be grouped together.

6. What does the C# record keyword do?
Ans: The record keyword is used to declare a special type of class that is primarily intended for holding immutable data.

7. What does overloading and overriding mean?
Overloading is that we can define multiple methods with the same name in a class but need to use different method signatures that is we need to use different parameter lists.

Overriding is that we can rewrite the method inherited from the parent's class.

8. What is the difference between a field and a property?

A field is a variable of any type that is declared directly in a class. A property is a member that provides a flexible mechanism to read, write or compute the value of a private field.

9. How do you make a method parameter optional?

Use optional in parameters list.

10. What is an interface and how is it different from abstract class?

An interface contains a set of function without providing implementation.
An abstract class is a class that cannot be instantiated and may contain abstract methods, as well as non-abstract methods and fields.

11. What accessibility level are members of an interface?

Public , internal

12. True/False. Polymorphism allows derived classes to provide different implementations of the same method.

True

13. True/False. The override keyword is used to indicate that a method in a derived class is providing its own implementation of a method.

True

14. True/False. The new keyword is used to indicate that a method in a derived class is providing its own implementation of a method.

True

15. True/False. Abstract methods can be used in a normal (non-abstract) class.

False

16.True/False. Normal (non-abstract) methods can be used in an abstract class.

True

17. True/False. Derived classes can override methods that were virtual in the base class.

True

18. True/False. Derived classes can override methods that were abstract in the base class.

True

19. True/False. In a derived class, you can override a method that was neither virtual non abstract in the base class.

False

20. True/False. A class that implements an interface does not have to provide an implementation for all of the members of the interface.

False

21. True/False. A class that implements an interface is allowed to have other members that aren't defined in the interface.

True

22. True/False. A class can have more than one base class.

False

23. True/False. A class can implement more than one interface.

True