1.. What are the six combinations of access modifier keywords and what do they do?

public: can be accessed by anywhere.

protected: can be accessed only by code in the same class, or from a class that inherits it.

private: can be accessed only by code in the same classs

internal: can be accessed from code that is part of the same compilation.

protected internal: can be accessed from code that is part of the same compilation or from within a derived class in any assembly.

private protected: can be accessed from derived classes that are declared within its containing assembly or within the same class.

2.What is the difference between the static, const, and readonly keywords when applied to a type member?static keyword denotes a member of that belongs to the class rather than a particular object of the class. It can also be used to declare static classes

Const keyword makes a variable constant. The value is declared at compile time.

Readonly – readonly is similar to const but only applicable to fields and is declared at runtime, typically at the constructor.

3. What does a constructor do?A constructor instantiates an object of the class.

4. Why is the partial keyword useful?

you can divide the implementation of a class, interface, a struct, a method over two or more files using the partial keyword.

5. What is a tuple?Tuple allows to group multiple different data types together. A tuple may contain any number of elements.

6. What does the C# record keyword do?A record is an immutable data type that is read only. It can only be initialized in a constructor.

7. What does overloading and overriding mean?

**Overloading:** allowing methods to have the same name for reusability but for different parameters to allow different functionality

Overriding: providing a new implementation of a base class method in the derived class

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

Fields are normal variable members of a class.

Properties are an abstraction for accessing fields. They are also called accessors because they offer a way to change and retrieve a field.

9. How do you make a method parameter optional?

We can make a parameter optional by assigning it a default value.

10. What is an interface and how is it different from abstract class?Interface is a contract that gives a list of methods which must be implemented by the derived class. Abstract classes cannot be instantiated but concrete classes can. Even one single abstract method makes the class abstract. Interfaces can support multiple inheritance but abstract classes cannot

11. What accessibility level are members of an interface?

Public12. True/False. Polymorphism allows derived classes to provide different implementationsof the same method.- True

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

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

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 thebase class – False

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

21. True/False. A class that implements an interface is allowed to have other members thataren’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

What is meant by the terms managed resource and unmanaged resource in .NET

Unmanaged objects are created outside the control of .NET libraries. Garbage collector is unable to clear them automatically.

Managed objects are created and managed and under the scope .NET code. Garbaqge collector disposes them automatically

24. What's the purpose of Garbage Collector in .NET?

Garbage collector is automatic memory manager. It removes objects that are no longer being used, clears their memory, and keeps the memory available for future allocations. It has 3 generations called 0,1,2.