1. What are the six combinations of access modifier keywords and what do they do?
   1. public: Access is not restricted.
   2. protected: Access is limited to the containing class or types derived from the containing class.
   3. internal: Access is limited to the current assembly.
   4. protected internal: Access is limited to the current assembly or types derived from the containing class.
   5. private: Access is limited to the containing type.
   6. private protected: Access is limited to the containing class or types derived from the containing class within the current assembly.
2. What is the difference between the static, const, and readonly keywords when applied to a type member?
   1. static members are common to all the objects and they do not tie to a specific object. This keyword can be used with classes, fields, methods, properties, operators, events, and constructors, but it cannot be used with indexers, destructors, or types other than classes.
   2. A const field is a compile-time constant. A constant field or local variable can be initialized with a constant expression which must be fully evaluated at compile time.
   3. A readonly field can be initialized either at the time of declaration or within the constructor of the same class. Therefore, readonly fields can be used for run-time constants.
3. What does a constructor do?
   1. 1)is a special method which shares the same name of the class and doesn't have any return type, not even void
   2. 2) constructor is used to create an object of the class and initialize class members
   3. 3) if there is no constructor in the class, c# compiler will provide a default constructor
   4. 4) if we create any constructor ourselves, the default constructor will be replaced
   5. 5) constructor can be overloaded
   6. 6) constructor cannot be inherited so a constructor cannot be overridden
   7. 7) by default, the derived class constructor will make a call the the base class constructor
4. Why is the partial keyword useful?
   1. It provides a special ability to implement the functionality of a single class into multiple files and all these files are combined into a single class file when the application is compiled.
   2. This keyword is also useful to split the functionality of methods, interfaces, or structure into multiple files.
5. What is a tuple?
   1. the tuples feature provides concise syntax to group multiple data elements in a lightweight data structure.
6. What does the C# record keyword do?
   1. Beginning with C# 9, you use the record keyword to define a reference type that provides built-in functionality for encapsulating data. C# 10 allows the record class syntax as a synonym to clarify a reference type, and record struct to define a value type with similar functionality. You can create record types with immutable properties by using positional parameters or standard property syntax.
7. What does overloading and overriding mean?
   1. Overloading occurs when two or more methods in one class have the same method name but different parameters.
   2. Overriding occurs when two methods have the same method name and parameters.
8. What is the difference between a field and a property?
   1. A field is a variable of any type that is declared directly in a class.
   2. 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?
   1. . To implement the optional parameter first you need to add System.Runtime.InteropServices namespace in your program, then creates an optional parameter using the Optional keyword enclosed in square brackets before the definition of the parameter in the method. The default value of OptionalAttribut is zero.
10. What is an interface and how is it different from abstract class?
    1. abstract class provides a base class to its subclasses -- use when we have a clear class hierachy
    2. interface defines common behaviors / functionalities that can be implemented by any classes -- work as a contract
    3. one class can only inherit from one parent class, but one class can implement multiple interfaces
    4. methods in abstract class can be abstract methods or non-abstract methods, but methods in interfaces are by default abstract
11. What accessibility level are members of an interface?
    1. By default it is abstract.
    2. public
    3. protected
    4. internal
    5. private
    6. protected internal
    7. private protected
12. True/False. Polymorphism allows derived classes to provide different implementations of the same method.
    1. 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.
    1. 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.
    1. False
15. True/False. Abstract methods can be used in a normal (non-abstract) class.
    1. False
16. True/False. Normal (non-abstract) methods can be used in an abstract class.
    1. True
17. True/False. Derived classes can override methods that were virtual in the base class.
    1. True
18. True/False. Derived classes can override methods that were abstract in the base class.
    1. True
19. True/False. In a derived class, you can override a method that was neither virtual non abstract in the base class.
    1. 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.
    1. False
21. True/False. A class that implements an interface is allowed to have other members that aren’t defined in the interface.
    1. False
22. True/False. A class can have more than one base class.
    1. False
23. True/False. A class can implement more than one interface.
    1. True