03 Object-Oriented Programming

Test your knowledge

1. What are the six combinations of access modifier keywords and what do they do?
   private: Accessible only within the same class.
   public: Accessible from anywhere.
   protected: Accessible within the same class and derived classes.
   internal: Accessible within the same assembly.
   protected internal: Accessible within the same assembly and by derived classes even though they are in different aseembly.
   private protected: Accessible only within the same class and derived classes within the same assembly.

2. What is the difference between the static, const, and readonly keywords when applied to a type member?
   Static: Used for members that belong to the class, not objects. Access without creating an object.
   Const: Used for values that never change. Set at compile time and can't be changed.
   Readonly: Used for values that can only be set once, either when declared or in the constructor. Cannot change later.

3. What does a constructor do?
   Constructor allows to create object (instantiate). It is the first point of operation when the object is created.

4. Why is the partial keyword useful?
   It is useful when the class is large we can make parts of it using the partial keyword. The partial classes compile as one during compilation.

5. What is a tuple?
   Tuple is data type which stores different tyes in a single unit.

6. What does the C# record keyword do?
   A record in C# is a special type used to store data that can't be changed once it's created. It is good for data that needs to be compared by values, not by reference.

7. What does overloading and overriding mean?
   Overloading means to apply multiple parameters to a method with same name whereas overriding means override the statements in of the base method by a child class. Overloading is compile time whereas overriding is runtime

8. What is the difference between a field and a property?
   A field is a variable that holds an object's data, while a property provides controlled access to a field after instantiation.

9. How do you make a method parameter optional?

By giving the default value of parameter

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

Interface: It is a contract that defines only method signatures (no implementation). Any class that implements the interface must provide its own implementation of those methods. An interface cannot contain any method bodies, only definitions.

Abstract Class: It can contain both abstract methods (without body) and concrete methods (with body). A class that inherits from an abstract class must implement the abstract methods, but it can also inherit the concrete methods as is.

11. What accessibility level are members of an interface?

Public only

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.

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 by using override

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 that is the whole point of interface

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 only interfaces

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

True