03 Object-Oriented Programming

Test your knowledge

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

* Public: access is not restricted.
* Protected: limit to the containing class and derived from containing class.
* Private: within contain class only
* Internal: current assembly.
* Protected internal: in current assembly or derived from containing class
* Private protected: in current 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?

* Static: belongs to the type itself rather than to specific object. Don’t need instantiated.
* Const: use for constant value which not change. Don’t need instantiated
* Readonly: can only occur as part of declaration of the class opr constructor.

3. What does a constructor do?

* Is a special method which share same name of the class, and don’t have return type, not even void; use for initialize class members and create instance object of class.
* If no contructor, compiler will provide default one.
* If we created constructor by ourselves, then default constructors no more exist.
* Constructors can be overloaded
* Constructor cannot be inherited so constructor cannot be overridden.
* By default, derived class contructor makes call to the base class parameterless contructor.

4. Why is the partial keyword useful?

* The partial keyword indicates that other parts of the class, struct, or interface can be defined in the namespace. All the parts must use the partial keyword. All the parts must be available at compile time to form the final type

5. What is a tuple?

* A tuple is data structure can contains sequence of different types of elements

6. What does the C# record keyword do?

* To clarify that it’s a reference type

7. What does overloading and overriding mean?

* Overloading: multiple methods in same class, same name but must have different numbers of parameters or different return types.
* Overriding: happen when there is base class and child class, must be same name, same parameters, and same return type. Child class provide different implementation.

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

|  |  |
| --- | --- |
| Field | Property |
| A field is a variable of any type that is declared directly in the class | Is a member that provide flexible read or write, or compute the value of private fields |
| Explain characteristic of object or a class | Set and receive value of a field. |

9. How do you make a method parameter optional?

* Default value will be assigned to the original parameter.

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

|  |  |
| --- | --- |
| Abstract class | Interface |
| Abstract class provides a base class to its subclasses, use when we have a clear class hierarchy | Interface defines common behaviors or functionalities that can be implemented by any class |
| One class can only inherit from one base class | One class can implement multiple interface |
| Method in abstract class can be abstract or non-abstract methods | By default abstract and public. |

11. What accessibility level are members of an interface?

* public

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

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

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

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

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

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

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

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

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

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

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

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