C# Knowledge Test - Answers

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

- `public`: Accessible from any other class.  
- `private`: Accessible only within the containing class.  
- `protected`: Accessible within the containing class and its derived classes.  
- `internal`: Accessible only within the same assembly.  
- `protected internal`: Accessible within the same assembly or from derived classes.  
- `private protected`: Accessible only within the containing class or types derived from the containing class within the same 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 instances of the type.  
- `const`: A compile-time constant; must be initialized where declared.  
- `readonly`: A runtime constant; can be assigned in the declaration or in the constructor.

## 3. What does a constructor do?

Initializes a new instance of a class.

## 4. Why is the partial keyword useful?

Allows a class, struct, or interface to be split into multiple files, useful for separating auto-generated and user-written code.

## 5. What is a tuple?

A data structure that holds a finite ordered list of elements, possibly of different types.

## 6. What does the C# record keyword do?

Defines a reference type that provides built-in functionality for encapsulating data and value-based equality.

## 7. What does overloading and overriding mean?

- Overloading: Defining multiple methods with the same name but different signatures.  
- Overriding: Providing a new implementation for a method defined in a base class.

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

- Field: A variable that holds data.  
- Property: Provides controlled access to a field through get and set accessors.

## 9. How do you make a method parameter optional?

Provide a default value in the method signature.

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

- Interface: Defines a contract of methods/properties without implementation.  
- Abstract class: Can provide both fully implemented and abstract members.

## 11. What accessibility level are members of an interface?

Always public.

## 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 nor 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