An interface has the following properties:

* An interface is like an abstract base class. Any class or struct that implements the interface must implement all its members.
* An interface can't be instantiated directly. Its members are implemented by any class or struct that implements the interface.
* Interfaces can contain events, methods and properties.
* Interfaces contain no implementation of methods.
* A class or struct can implement multiple interfaces. A class can inherit a base class and also implement one or more interfaces.

To implement an interface member, the corresponding member of the implementing class must be public, non-static, and have the same name and signature as the interface member

Interfaces declared directly within a namespace can be declared as public or internal and, just like classes and structs, interfaces default to internal access. Interface members are always public because the purpose of an interface is to enable other types to access a class or struct. **No access modifiers can be applied to interface members.**

Explicit interfaces can also be used to hide the details of an interface that the class developer considers private.