

# Abstract classes and Interfaces

- **Static members** - can access on Class only not its objects
  - properties
  - functions
  - Helper methods can be made static
- **Abstract Classes**
  - cannot be instantiated
  - **abstract** 'keyword'
  - we can make our constructor protected
  - child classes must implement abstract members both - properties and methods
  - we can only inherit from abstract classes
  - **overriding parent methods** - same function signature in child class
- **Interfaces**
  - Interfaces are contract that defines in advance, how an object or a class structure will look, when an instance is created by user.
  - **'interface'** keyword
  - when dealing with objects we can choose either custom types or interface
  - interface is preferred
  - **extends** - one interface having properties of other interfaces
  - **multiple inheritance**
  - we can use interface with generics to make them flexible
  - we can use interfaces with class using **'implements'** keyword
  - we can implement multiple interfaces on a class
  - we cannot have any access modifiers on interfaces
  - the class that implements interface, cannot have 'private' or 'protected' interface declared properties
    - however, we can have other non-interface implemented properties as 'private' or 'protected'
- Classes can inherit from only one parent class

- **Interfaces vs Types**

- **type** - only **type** work not **interfaces**

- Intersection Types (&)
    - Union Types ( | )
    - tuples

- interfaces - only **interfaces** not **types**

- interfaces with same name - properties get merged together
    - interfaces can extend each other
    - we can implement multiple interfaces

- **Interfaces vs Abstract Classes**

- interface do not contain any implementation of property or methods
  - multiple interfaces, but only one parent class
  - abstract classes may contain static methods, interfaces do not

- \*tip - when stuck look over [stackoverflow](#), or other open source projects to look how people have solved the similar issue