1.Static

- **Definition**: Members declared as static belong to the class itself rather than to any specific instance of the class.

- **Usage**: Useful for defining class-level methods and fields.

+ Created only one in the memory

+ Accessed in the name of the class (AppSettings.Version)

+ Cant be accessed through objects (obj.Version) // not possible

+ Value can be Modified.

2. Const

- **Definition**: Constants (const) are immutable values which are known at compile-time and do not change for the life of the program.

- **Usage**: Ideal for defining constant values.

+ Created only once in the memory

+ Created only one in the memory

+ Accessed in the name of the class (AppSettings.Version)

+ Can't be accessed through objects (obj.Version) // not possible

+ Value can't be Modified.

3.Static ReadOnly

- **Definition**: Fields declared as static readonly can be assigned at runtime or in a static constructor. They are immutable after initialization.

- **Usage**: Useful for fields that need to be initialized at runtime and then remain constant.

- Created only one in the memory

- Accessed in the name of the class (AppSettings.Version)

- Can't be accessed through objects (obj.Version) // not possible

- Value can be modified only from a static Constructor

4. ReadOnly

- **Definition**: Fields declared as readonly can be assigned either at the time of declaration or in a constructor. They can only be changed during object construction.

- **Usage**: Ideal for instance-level fields that should remain constant after initialization.

- Created for each Objects

- Accessed in the name of the class (AppSettings.Version)

- Can't be accessed through objects (obj.Version) // not possible

- Value can be modified only from a Constructor