C# Sealed

C# sealed keyword applies restrictions on the class and method. If you create a sealed class, it cannot be derived. If you create a sealed method, it cannot be overridden.

> *Note: Structs are implicitly sealed therefore they can't be inherited.*

C# Sealed class

C# sealed class cannot be derived by any class. Let's see an example of sealed class in C#.

```
1.  using System;
2.  sealed public class Animal{
3.      public void eat() { Console.WriteLine("eating..."); }
4.  }
5.  public class Dog: Animal
6.  {
7.      public void bark() { Console.WriteLine("barking..."); }
8.  }
9.  public class TestSealed
10. {
11.     public static void Main()
12.     {
13.         Dog d = new Dog();
14.         d.eat();
15.         d.bark();
16.
17.
18.     }
19. }
```

Output:

```
Compile Time Error: 'Dog': cannot derive from sealed type 'Animal'
```

C# Sealed method

The sealed method in C# cannot be overridden further. It must be used with override keyword in method.

Let's see an example of sealed method in C#.

```
1.  using System;
2.  public class Animal{
3.      public virtual void eat() { Console.WriteLine("eating..."); }
4.      public virtual void run() { Console.WriteLine("running..."); }
5.
```

```
 6. }
 7. public class Dog: Animal
 8. {
 9.     public override void eat() { Console.WriteLine("eating bread..."); }
10.     public sealed override void run() {
11.     Console.WriteLine("running very fast...");
12.     }
13. }
14. public class BabyDog : Dog
15. {
16.     public override void eat() { Console.WriteLine("eating biscuits..."); }
17.     public override void run() { Console.WriteLine("running slowly..."); }
18. }
19. public class TestSealed
20. {
21.     public static void Main()
22.     {
23.         BabyDog d = new BabyDog();
24.         d.eat();
25.         d.run();
26.     }
27. }
```

Output:

```
Compile  Time  Error:  'BabyDog.run()':  cannot  override  inherited  member  'Dog.run()'
because it is sealed
```

Note: Local variables can't be sealed.

```
 1. using System;
 2. public class TestSealed
 3. {
 4.     public static void Main()
 5.     {
 6.         sealed int x = 10;
 7.         x++;
 8.         Console.WriteLine(x);
 9.     }
10. }
```

Output:

```
Compile Time Error: Invalid expression term 'sealed'
```