**Difference Between Const, Read only and Static**

1. Const is nothing but "constant", a variable of which the value is constant but at compile time.

- And it's mandatory to assign a value to it.

- By default a const is static and we cannot change the value of a const variable throughout the entire program.

Eg.

class Calendar1

{

public const int Months = 12;

}

Here I have created a class named Variables and defined all three variables, so first let's play with const.

Eg.

class Calendar3

{

public const int Months = 12;

public const int Weeks = 52;

public const int Days = 365;

public const double DaysPerWeek = (double) Days / (double) Weeks;

public const double DaysPerMonth = (double) Days / (double) Months;

}

Constants can be marked as public, private, protected, internal, protected internal or private protected. These access modifiers define how users of the class can access the constant

2.

Readonly is the keyword whose value we can change during runtime or we can assign it at run time but only through the non-static constructor.

Not even a method.

- When the variable is initialized in the declaration, for example

Eg.

public readonly int y = 5;

In this example, the value of the field year can't be changed in the method ChangeYear, even though it's assigned a value in the class constructor:

class Age

{

private readonly int \_year;

Age(int year)

{

\_year = year;

}

void ChangeYear()

{

//\_year = 1967; // Compile error if uncommented.

}

}

3. The static keyword in C# can be used on a variable, a method, or an object. Note that a static member of a class belongs to the type of the object rather than to the instance of the type. In other words, static members are accessed with the name of the class, not the name of an instance.

Eg.

public static class Calculator

{

private static int \_resultStorage = 0;

public static string Type = "Arithmetic";

public static int Sum(int num1, int num2)

{

return num1 + num2;

}

public static void Store(int result)

{

\_resultStorage = result;

}

}

Above, the Calculator class is a static. All the members of it are also static

**A rule for const, readonly, and static**

Here is the rule of the thumb you can follow when working with the const, readonly, and static keywords.

Use the const keyword when the value contained in a variable will never change during the lifetime of the application. Use the readonly keyword when you are not sure whether the value of a variable of an object needs to change but you want to prevent other classes from changing the value. Use the static keyword when you want the member of a class to belong to the type rather than to the instance of the type.