**PASS BY VALUE AND PASS BY REFERENCE IN C#**

**(REF AND OUT KEYWORDS)**

PASS BY VALUE

static void Main(string[] args)

{

int value = 5;

PassByValue(value); 15

console.writeline(value);

}

Static void PassByValue (int a)

{

a = a + 10;

console.writeline(“value is: ” + a);

}

5

5

15

abc

PASS BY REFERENCE

Static void PassByRef (ref int a)

{

a = a + 10;

console.writeline(“value is: ” + a);

}

static void Main(string[] args)

{

int value = 5;

PassByRef(ref value);15

console.writeline(value);

}

static void Main(string[] args)

{

int value;

PassByOut(out value);

console.writeline(value);

}

PASS BY OUT

Static void PassByOut (out int a)

{

a = 20;

console.writeline(“value is: ” + a);

}

**PASS BY REFERENCE (REF KEYWORD)**

* The ref keyword causes arguments to be passed in a method by reference.
* In call by reference, the called method changes the value of the parameters passed to it from the calling method.
* Any changes made to the parameters in the called method will be reflected in the parameters passed from the calling method when control passes back to the calling method.
* It is necessary that both the called method and the calling method must explicitly specify the **ref** keyword before the required parameters.
* The variables passed by reference from the calling method must be first initialized.

**OUT KEYWORD**

* The out keyword is similar to the ref keyword and causes arguments to be passed by reference.
* The only difference between the two is that the out keyword does not require the variables that are passed by reference to be initialized.
* Both the called method and the calling method must explicitly use the **out** keyword.