# 基本语法

# C#中out和ref之间的区别

ref（C# 参考）

ref 关键字使参数按引用传递。其效果是，当控制权传递回调用方法时，在方法中对参数的任何更改都将反映在该变量中。若要使用 ref 参数，则方法定义和调用方法都必须显式使用 ref 关键字。

例如：

class RefExample

{

static void Method(ref int i)

{

i = 44;

}

static void Main()

{

int val = 0;

Method(ref val);

// val is now 44

}

}

传递到 ref 参数的参数必须最先初始化。这与 out 不同，后者的参数在传递之前不需要显式初始化。

尽管 ref 和 out 在运行时的处理方式不同，但在编译时的处理方式相同。因此，如果一个方法采用 ref 参数，而另一个方法采用 out 参数，则无法重载这两个方法。例如，从编译的角度来看，以下代码中的两个方法是完全相同的，因此将不会编译以下代码：

class CS0663\_Example

{

// Compiler error CS0663: "cannot define overloaded

// methods that differ only on ref and out".

public void SampleMethod(ref int i) { }

public void SampleMethod(out int i) { }

}

但是，如果一个方法采用 ref 或 out 参数，而另一个方法不采用这两个参数，则可以进行重载，如下例所示：

class RefOutOverloadExample

{

public void SampleMethod(int i) { }

public void SampleMethod(ref int i) { }

}

out（C# 参考）

out 关键字会导致参数通过引用来传递。这与 ref 关键字类似，不同之处在于 ref 要求变量必须在传递之前进行初始化。若要使用 out 参数，方法定义和调用方法都必须显式使用out 关键字。

class OutExample

{

static void Method(out int i)

{

i = 44;

}

static void Main()

{

int value;

Method(out value);

// value is now 44

}

}

尽管作为 out 参数传递的变量不必在传递之前进行初始化，但需要调用方法以便在方法返回之前赋值。

区别：

ref和out的区别在C# 中，既可以通过值也可以通过引用传递参数。通过引用传递参数允许函数成员更改参数的值，并保持该更改。若要通过引用传递参数， 可使用ref或out关键字。ref和out这两个关键字都能够提供相似的功效，其作用也很像C中的指针变量。它们的区别是：

1、使用ref型参数时，传入的参数必须先被初始化。对out而言，必须在方法中对其完成初始化。

2、使用ref和out时，在方法的参数和执行方法时，都要加Ref或Out关键字。以满足匹配。

3、out适合用在需要retrun多个返回值的地方，而ref则用在需要被调用的方法修改调用者的引用的时候。

out

方法参数上的 out 方法参数关键字使方法引用传递到方法的同一个变量。当控制传递回调用方法时，在方法中对参数所做的任何更改都将反映在该变量中。

当希望方法返回多个值时，声明 out 方法非常有用。使用 out 参数的方法仍然可以返回一个值。一个方法可以有一个以上的 out 参数。

若要使用 out 参数，必须将参数作为 out 参数显式传递到方法。out 参数的值不会传递到 out 参数。

不必初始化作为 out 参数传递的变量。然而，必须在方法返回之前为 out 参数赋值。

属性不是变量，不能作为 out 参数传递。

ref是有进有出，而out是只出不进。  
（之前看过忘了，转载于https://www.cnblogs.com/sunliyuan/p/5999045.html）