

1 System.Diagnostics.ConditionalAttribute

2 Class

```
3  
4  
5 [ILASM]  
6 .class public sealed serializable ConditionalAttribute  
7 extends System.Attribute
```

```
8 [C#]  
9 public sealed class ConditionalAttribute: Attribute
```

10 Assembly Info:

- 11 • Name: mscorlib
- 12 • Public Key: [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00]
- 13 • Version: 1.0.x.x
- 14 • Attributes:
 - 15 ○ CLSCompliantAttribute(true)

16 Type Attributes:

- 17 • AttributeUsageAttribute(AttributeTargets.Method, AllowMultiple=true,
18 Inherited=true)

19 Summary

20
21 Indicates to compilers that a method is callable if and only if a
22 specified pre-processing identifier has been defined on the method.

23 Inherits From: System.Attribute

24
25 Library: BCL

26
27 **Thread Safety:** All public static members of this type are safe for multithreaded
28 operations. No instance members are guaranteed to be thread safe.

29 Description

30
31 [Note: A **System.Diagnostics.ConditionalAttribute**, which has an
32 associated condition
33 **System.Diagnostics.ConditionalAttribute.ConditionString**, can be
34 attached to the definition of a method, creating a *conditional method*.
35 Thereafter, when a compiler encounters a call to that method, it may
36 choose to ignore the call unless a compilation variable is defined at the
37 site of the call, with a value that matches in a case-sensitive manner
38 the **System.Diagnostics.ConditionalAttribute.ConditionString**
39 supplied to the **System.Diagnostics.ConditionalAttribute**.
40

1 Note that compilers may provide several techniques to define such
2 compilation variables, such as:

- 3 • compiler command-line switches (for example, /define:DEBUG)
- 4 • environment variables in the operating system shell (for example, SET
5 DEBUG=1)
- 6 • as pragmas in the source code (for example, #define DEBUG, to define
7 the compilation variable, or #undef DEBUG to undefine it)

8 CLS-Compliant compilers are permitted to ignore uses of the
9 **System.Diagnostics.ConditionalAttribute.**]

10 Example

11

12 The following example demonstrates the use of
13 **System.Diagnostics.ConditionalAttribute** with a particular compiler
14 that supports the use of this attribute. The
15 **System.Diagnostics.ConditionalAttribute.ConditionString**
16 property of the current attribute is initialized as "DEBUG".

17

18

```
[C#]
```

19

```
using System;
```

20

```
using System.Diagnostics;
```

21

```
public class MyClass {
```

22

```
    [ConditionalAttribute("DEBUG")]
```

23

```
    public static void Display() {
```

24

25

```
        Console.WriteLine("Compiled with DEBUG");
```

26

```
    }
```

27

```
}
```

28

```
public class TestCondition {
```

29

30

```
    public static void Main() {
```

31

```
        Console.WriteLine("How was this compiled?");
```

32

```
        MyClass.Display();
```

33

```
        Console.WriteLine("<eop>");
```

34

```
    }
```

35

```
}
```

36

37

38

39

40

41

42 When this code is compiled with the compilation-variable DEBUG
43 defined at the callsite, the output when run is

1 How was this compiled?

2

3

4

Compiled with DEBUG

5

6

7

<eop>

8

9

10 When this code is compiled without the compilation-variable DEBUG
11 defined at the callsite, the output when run is

12

13

How was this compiled?

14

15

16

<eop>

17

18

1 ConditionalAttribute(System.String)

2 Constructor

```
3 [ILASM]  
4 public rtspecialname specialname instance void .ctor(string  
5 conditionString)  
  
6 [C#]  
7 public ConditionalAttribute(string conditionString)
```

8 Summary

9 Constructs and initializes a new instance of the
10 **System.Diagnostics.ConditionalAttribute** class.

11 Parameters

12
13

Parameter	Description
<i>conditionString</i>	A System.String that contains the pre-processing identifier that makes callable the target method of the current instance.

14

15 Description

16 This constructor initializes the
17 **System.Diagnostics.ConditionalAttribute.ConditionString**
18 property of the current instance using *conditionString*.

19

1 ConditionalAttribute.ConditionString

2 Property

```
3 [ILASM]  
4 .property string ConditionString { public hidebysig  
5 specialname instance string get_ConditionString() }  
6  
7 [C#]  
8 public string ConditionString { get; }
```

8 Summary

9 Gets the **System.String** that contains the pre-processing identifier
10 that makes callable the target method of the current instance.

11 Property Value

12

13 A **System.String** that contains the pre-processing identifier that
14 makes callable the target method of the current instance.

15 Description

16 This property is read-only.

17