

# System.ObsoleteAttribute Class

```
[ILAsm]
.class public sealed serializable ObsoleteAttribute extends
System.Attribute

[C#]
public sealed class ObsoleteAttribute: Attribute
```

## Assembly Info:

- *Name:* mscorlib
- *Public Key:* [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00 00]
- *Version:* 2.0.x.x
- *Attributes:*
  - CLSCompliantAttribute(true)

## Type Attributes:

- AttributeUsageAttribute(AttributeTargets.Class | AttributeTargets.Struct | AttributeTargets.Enum | AttributeTargets.Constructor | AttributeTargets.Method | AttributeTargets.Property | AttributeTargets.Field | AttributeTargets.Event | AttributeTargets.Interface | AttributeTargets.Delegate, AllowMultiple=false, Inherited=false)

## Summary

Indicates that the target of the current attribute will be removed in future versions of the assembly in which the target is contained.

## Inherits From: System.Attribute

## Library: BCL

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

## Description

[*Note:* Marking an item as obsolete provides consumers of that item the information that the item will not be available in future versions of the assembly in which it is contained. A `System.ObsoleteAttribute` has a `System.ObsoleteAttribute.Message` property that can be used to suggest alternative ways of obtaining the functionality provided by the item, i.e. a workaround. This class also has a `System.ObsoleteAttribute.IsError` property that designates whether a compiler will treat usage of the obsolete item as an error. If this property is `false`, the compiler will issue a warning if the obsolete item is used and the compiler supports the generation of

```
1      such warnings.
2
3      This attribute can be applied to any valid attribute target except assemblies,
4      parameters, and return values. For a complete list of valid attribute targets, see
5      System.AttributeTargets.
6
7      ]
```

## 8 **Example**

9

10 The following example demonstrates the usage of `System.ObsoleteAttribute` to  
11 generate a compile-time warning.

```
12
13 [C#]
14
15
16 using System;
17
18 public class ObsoleteAttributeExample {
19
20     [ObsoleteAttribute("OldMethod is being removed: use NewMethod in future
21 versions.")]
22     public static void OldMethod() {
23
24         //Execute some code here
25     }
26
27     public static void Main() {
28
29         OldMethod();
30     }
31 }
```

32 An example compile-time result is

```
33
34 ObsoleteAttributeExample.cs(8,4): warning CS0618:
35 'ObsoleteAttributeExample.OldMethod()' is obsolete: 'OldMethod is being
36 removed: use NewMethod in future versions.'
```

37

# ObsoleteAttribute() Constructor

```
[ILAsm]  
public rtspecialname specialname instance void .ctor()  
  
[C#]  
public ObsoleteAttribute()
```

## Summary

Constructs and initializes a new instance of the `System.ObsoleteAttribute` class.

## Description

This constructor is equivalent to `System.ObsoleteAttribute(null, false)`. The compiler does not treat an item with this attribute as an error.

# ObsoleteAttribute(System.String) Constructor

```
[ILAsm]  
public rtspecialname specialname instance void .ctor(string message)  
  
[C#]  
public ObsoleteAttribute(string message)
```

## Summary

Constructs and initializes a new instance of the `System.ObsoleteAttribute` class with the specified `System.String` that contains suggested workarounds.

## Parameters

Parameter	Description
<i>message</i>	The <code>System.String</code> that contains suggested workarounds.

## Description

This constructor is equivalent to `System.ObsoleteAttribute (message, false )`. The compiler does not treat an item with this attribute as an error.

# ObsoleteAttribute(System.String, System.Boolean) Constructor

```
[ILAsm]
public rtspecialname specialname instance void .ctor(string message, bool
error)

[C#]
public ObsoleteAttribute(string message, bool error)
```

## Summary

Constructs and initializes a new instance of the `System.ObsoleteAttribute` class with a `System.String` that contains suggested workarounds and a `System.Boolean` that indicates whether the compiler treats usage of the target of the current instance as an error.

## Parameters

Parameter	Description
<i>message</i>	A <code>System.String</code> that contains suggested workarounds.
<i>error</i>	A <code>System.Boolean</code> that indicates whether the compiler treats usage of the target of the current instance as an error.

## Description

Respectively, the `System.ObsoleteAttribute.Message` property and the `System.ObsoleteAttribute.IsError` property of the new instance are initialized as *message* and *error*.

# ObsoleteAttribute.IsError Property

```
[ILAsm]  
.property bool IsError { public hidebysig specialname instance bool  
get_IsError() }  
  
[C#]  
public bool IsError { get; }
```

## Summary

Gets a `System.Boolean` that indicates whether the compiler treats usage of the target of the current instance as an error.

## Property Value

true if the compiler treats usage of the target of the current instance as an error; otherwise, false.

## Description

This property is read-only.

The default value of this property is `false`.

# ObsoleteAttribute.Message Property

```
[ILAsm]
.property string Message { public hidebysig specialname instance string
get_Message() }

[C#]
public string Message { get; }
```

## Summary

Gets a `System.String` that contains suggested workarounds for the target of the current instance.

## Property Value

A `System.String` that contains suggested workarounds for the target of the current instance.

## Description

This property is read-only.

The current instance contains a suggested workaround message if and only if such a message was specified when the current instance was constructed. If no workaround was specified for the current instance, the value of this property is `null`.