

# System.AttributeTargets Enum

```
[ILAsm]
.class public sealed serializable AttributeTargets extends System.Enum

[C#]
public enum AttributeTargets
```

## 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:

- FlagsAttribute

## Summary

Enumerates the application elements to which it is valid to attach an attribute.

## Inherits From: System.Enum

## Library: BCL

## Description

`System.AttributeTargets` is used as a parameter for `System.AttributeUsageAttribute` to enable an attribute to be associated with one or more kinds of application elements.

## Example

The following example demonstrates how `System.AttributeTargets` can be used with `System.AttributeUsageAttribute` so that a user-defined attribute class, *Author*, can be applied to structures and classes. The *Author* attribute is then applied to a class. There is no output.

```
[C#]

using System;

[AttributeUsageAttribute(AttributeTargets.Class |
                        AttributeTargets.Struct)]
public class Author: Attribute {
    public Author(string Name) {
```

```
1
2     this.name = Name; }
3     string name;
4 }
5
6 [Author("John Q Public")]
7 class JohnsClass {
8
9     public static void Main() {
10    }
11 }
12
```

# AttributeTargets.All Field

```
[ILAsm]
.field public static literal valuetype System.AttributeTargets All =
Assembly | 0x2 | Class | Struct | Enum | Constructor | Method | Property |
Field | Event | Interface | Parameter | Delegate | ReturnValue |
GenericParameter

[C#]
All = Assembly | 0x2 | Class | Struct | Enum | Constructor | Method |
Property | Field | Event | Interface | Parameter | Delegate | ReturnValue
| GenericParameter
```

## Summary

Attribute can be applied to any element.

## 1 AttributeTargets.Assembly Field

```
2 [ILAsm]
3 .field public static literal valuetype System.AttributeTargets Assembly =
4 0x1
5
6 [C#]
7 Assembly = 0x1
```

## 7 Summary

8 Attribute can be applied to an assembly.

9

# 1    **AttributeTargets.Class Field**

```
2    [ILAsm]  
3    .field public static literal valuetype System.AttributeTargets Class = 0x4  
  
4    [C#]  
5    Class = 0x4
```

## 6    **Summary**

7        Attribute can be applied to a class.

8

# 1 AttributeTargets.Constructor Field

```
2 [ILAsm]  
3 .field public static literal valuetype System.AttributeTargets Constructor  
4 = 0x20  
  
5 [C#]  
6 Constructor = 0x20
```

## 7 Summary

8 Attribute can be applied to a constructor.

# AttributeTargets.Delegate Field

```
[ILAsm]  
.field public static literal valuetype System.AttributeTargets Delegate =  
0x1000  
  
[C#]  
Delegate = 0x1000
```

## Summary

Attribute can be applied to a delegate.

## 1    **AttributeTargets.Enum Field**

```
2    [ILAsm]  
3    .field public static literal valuetype System.AttributeTargets Enum = 0x10  
  
4    [C#]  
5    Enum = 0x10
```

## 6    **Summary**

7        Attribute can be applied to an enumeration.

8



# 1    **AttributeTargets.Event Field**

```
2    [ILAsm]  
3    .field public static literal valuetype System.AttributeTargets Event =  
4    0x200  
  
5    [C#]  
6    Event = 0x200
```

## 7    **Summary**

8        Attribute can be applied to an event.

9

# 1 AttributeTargets.Field Field

```
2 [ILAsm]  
3 .field public static literal valuetype System.AttributeTargets Field =  
4 0x100  
  
5 [C#]  
6 Field = 0x100
```

## 7 Summary

8 Attribute can be applied to a field.

# 1    **AttributeTargets.GenericParameter Field**

```
2    [ILAsm]  
3    .field public static literal valuetype System.AttributeTargets  
4    GenericParameter = 0x4000  
  
5    [C#]  
6    GenericParameter = 0x4000
```

## 7    **Summary**

8        Attribute can be applied to a generic Parameter.

## 1    **AttributeTargets.Interface Field**

```
2    [ILAsm]  
3    .field public static literal valuetype System.AttributeTargets Interface =  
4    0x400  
  
5    [C#]  
6    Interface = 0x400
```

## 7    **Summary**

8        Attribute can be applied to an interface.

9

# 1    **AttributeTargets.Method Field**

```
2    [ILAsm]  
3    .field public static literal valuetype System.AttributeTargets Method =  
4    0x40  
  
5    [C#]  
6    Method = 0x40
```

## 7    **Summary**

8        Attribute can be applied to a method.

9

**The following member must be implemented if the RuntimeInfrastructure library is present in the implementation.**

## AttributeTargets.Module Field

```
[ILAsm]  
.field public static literal valuetype System.AttributeTargets Module =  
0x2  
  
[C#]  
Module = 0x2
```

### Summary

Attribute can be applied to a module.

# AttributeTargets.Parameter Field

```
[ILAsm]  
.field public static literal valuetype System.AttributeTargets Parameter =  
0x800  
  
[C#]  
Parameter = 0x800
```

## Summary

Attribute can be applied to a parameter.

# 1    **AttributeTargets.Property Field**

```
2    [ILAsm]  
3    .field public static literal valuetype System.AttributeTargets Property =  
4    0x80  
  
5    [C#]  
6    Property = 0x80
```

## 7    **Summary**

8        Attribute can be applied to a property.

9



# 1    **AttributeTargets.ReturnValue Field**

```
2    [ILAsm]  
3    .field public static literal valuetype System.AttributeTargets ReturnValue  
4    = 0x2000  
  
5    [C#]  
6    ReturnValue = 0x2000
```

## 7    **Summary**

8        Attribute can be applied to a return value.

# 1    **AttributeTargets.Struct Field**

```
2    [ILAsm]  
3    .field public static literal valuetype System.AttributeTargets Struct =  
4    0x8  
  
5    [C#]  
6    Struct = 0x8
```

## 7    **Summary**

8        Attribute can be applied to a value type.

9