

System.Reflection.Module Class

```
[ILASM]
.class public serializable Module extends System.Object

[C#]
public class Module
```

Assembly Info:

- *Name:* mscorlib
- *Public Key:* [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00]
- *Version:* 1.0.x.x
- *Attributes:*
 - CLSCompliantAttribute(true)

Summary

Provides access to module metadata.

Inherits From: System.Object

Library: Reflection

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

Description

A module is a single portable executable (PE) file.

[*Note:* One or more modules deployed as a unit composes an assembly.]

For more information on modules, see Partition II of the CLI Specification.]

Module.GetField(System.String) Method

```
[ILASM]
.method public hidebysig instance class
System.Reflection.FieldInfo GetField(string name)

[C#]
public FieldInfo GetField(string name)
```

Summary

Returns a **System.Reflection.FieldInfo** instance that reflects the global public field that has the specified name, and is a member of the module reflected by the current instance.

Parameters

Parameter	Description
<i>name</i>	A System.String that specifies the name of the field to be returned.

Return Value

Returns a **System.Reflection.FieldInfo** instance that reflects the global public field that has the name *name*, and is a member of the module reflected by the current instance, if found; otherwise, returns **null**.

Description

[Note: A global public field is a static field with a module-level scope.]

Exceptions

Exception	Condition
System.ArgumentNullException	<i>name</i> is null .

Module.GetField(System.String, System.Reflection.BindingFlags) Method

```
[ILASM]
.method public hidebysig instance class
System.Reflection.FieldInfo GetField(string name, valuetype
System.Reflection.BindingFlags bindingAttr)

[C#]
public FieldInfo GetField(string name, BindingFlags
bindingAttr)
```

Summary

Returns a **System.Reflection.FieldInfo** instance that reflects the global field that has the specified name and have the specified binding attributes, and is a member of the module reflected by the current instance.

Parameters

Parameter	Description
<i>name</i>	A System.String that specifies the name of the field to be returned.
<i>bindingAttr</i>	A bitwise combination of System.Reflection.BindingFlags value that control the binding process. [Note: Specify System.Reflection.BindingFlags.Public or System.Reflection.BindingFlags.NonPublic , or System.Reflection.BindingFlags.Static ; otherwise, this method will return null .]

Return Value

Returns a **System.Reflection.FieldInfo** instance that reflects the global field that has the name *name* and characteristics specified by *bindingAttr*, and is a member of the module reflected by the current instance, if found; otherwise, returns **null**.

Description

[Note: A global field is a field with a module-level scope.]

Exceptions

1
2
3

Exception	Condition
System.ArgumentNullException	<i>name</i> is null .

Module.GetFields(System.Reflection.BindingFlags) Method

```
[ILASM]
.method public hidebysig instance class
System.Reflection.FieldInfo[] GetFields(valuetype
System.Reflection.BindingFlags bindingAttr)

[C#]
public FieldInfo[] GetFields(BindingFlags bindingAttr)
```

Summary

Returns an array whose elements reflect the global fields that have the specified binding attributes, and are members of the module reflected by the current instance.

Parameters

Parameter	Description
<i>bindingAttr</i>	A bitwise combination of System.Reflection.BindingFlags values that control the binding process. [Note: Specify System.Reflection.BindingFlags.Public or System.Reflection.BindingFlags.NonPublic , and System.Reflection.BindingFlags.Static ; otherwise, this method will return null .]

Return Value

An array **System.Reflection.FieldInfo** objects that reflect the global fields that have the specified binding attributes, and are members of the module reflected by the current instance. If no global fields are contained in the module reflected by the current instance, returns an array with zero elements.

Description

[Note: A global public field is a static field with a module-level scope.]

1 Module.GetFields() Method

```
2 [ILASM]  
3 .method public hidebysig instance class  
4 System.Reflection.FieldInfo[] GetFields()  
  
5 [C#]  
6 public FieldInfo[] GetFields()
```

7 Summary

8 Returns an array whose elements reflect the global public fields that
9 are members of the module reflected by the current instance.

10 Return Value

11
12 An array **System.Reflection.FieldInfo** objects that reflect the global
13 public fields that are members of the module reflected by the current
14 instance. If no global public fields are contained in the module
15 reflected by the current instance, returns an array with zero elements.

16 Description

17 [Note: A global public field is a static field with a module-level scope.]
18

Module.GetMethod(System.String, System.Type[]) Method

```
[ILASM]
.method public hidebysig instance class
System.Reflection.MethodInfo GetMethod(string name, class
System.Type[] types)

[C#]
public MethodInfo GetMethod(string name, Type[] types)
```

Summary

Returns a **System.Reflection.MethodInfo** instance that reflects the global public method that has the specified name and parameter types, and is a member of the module reflected by the current instance.

Parameters

Parameter	Description
<i>name</i>	A System.String that specifies the name of the method to be returned.
<i>types</i>	An array of System.Type objects that contain the parameter types to search for.

Return Value

A **System.Reflection.MethodInfo** instance that reflects the global public method that has the name *name* and parameter types *types*, and is a member of the module reflected by the current instance, if found; otherwise, returns **null**.

Description

[Note: A global public method is a static method with a module-level scope.]

Exceptions

Exception	Condition
System.ArgumentNullException	<i>name</i> is null .
	<i>types</i> is null .

1
2
3

	At least one element of <i>types</i> is null .
--	---

Module.GetMethod(System.String) Method

```
[ILASM]
.method public hidebysig instance class
System.Reflection.MethodInfo GetMethod(string name)

[C#]
public MethodInfo GetMethod(string name)
```

Summary

Returns a **System.Reflection.MethodInfo** instance that reflects the global public method that has the specified name, and is a member of the module reflected by the current instance.

Parameters

Parameter	Description
<i>name</i>	A System.String that specifies the name of the method to be returned.

Return Value

A **System.Reflection.MethodInfo** instance that reflects the global public method that has the name *name*, and is a member of the module reflected by the current instance, if found; otherwise, returns **null**.

Description

[Note: A global public method is a static method with a module-level scope.]

Exceptions

Exception	Condition
System.ArgumentNullException	<i>name</i> is null .

Module.GetMethods(System.Reflection.BindingFlags) Method

```
[ILASM]
.method public hidebysig instance class
System.Reflection.MethodInfo[] GetMethods(valuetype
System.Reflection.BindingFlags bindingAttr)

[C#]
public MethodInfo[] GetMethods(BindingFlags bindingAttr)
```

Summary

Returns an array whose elements reflect the global methods that have the specified binding attributes, and are members of the module reflected by the current instance.

Parameters

Parameter	Description
<i>bindingAttr</i>	A bitwise combination of System.Reflection.BindingFlags values that control the binding process. [Note: Specify System.Reflection.BindingFlags.Public or System.Reflection.BindingFlags.NonPublic , and System.Reflection.BindingFlags.Static ; otherwise, this method will return null .]

Return Value

An array of **System.Reflection.MethodInfo** objects that reflect the global methods that have characteristics specified by *bindingAttr*, and are members of the module reflected by the current instance, if found; otherwise, returns **null**.

Description

[Note: A global method is a method with a module-level scope.]

1 Module.GetMethods() Method

```
2 [ILASM]  
3 .method public hidebysig instance class  
4 System.Reflection.MethodInfo[] GetMethods()  
  
5 [C#]  
6 public MethodInfo[] GetMethods()
```

7 Summary

8 Returns an array whose elements reflect the global public methods
9 that are members of the module reflected by the current instance.

10 Return Value

11

12 An array of **System.Reflection.MethodInfo** objects that reflect the
13 global public methods that are members of the module reflected by
14 the current instance.

15 Description

16 [Note: A global public method is a static method with a module-level
17 scope.]

18

1 Module.ToString() Method

```
2 [ILASM]  
3 .method public hidebysig virtual string ToString()  
4  
5 [C#]  
6 public override string ToString()
```

6 Summary

7 Returns a string representation of the name of the module reflected by
8 the current instance.

9 Return Value

10

11 A **System.String** representation of the name of the module reflected
12 by the current instance.

13 Description

14 [Note: This method overrides **System.Object.ToString.**]
15

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

Module.Assembly Property

```
[ILASM]  
.property class System.Reflection.Assembly Assembly {  
public hidebysig specialname instance class  
System.Reflection.Assembly get_Assembly() }
```

```
[C#]  
public Assembly Assembly { get; }
```

Summary

Gets the appropriate assembly for the module reflected by the current instance.

Property Value

A **System.Reflection.Assembly** instance that reflects the assembly that contains the module reflected by the current instance.

Description

This property is read-only.

Module.FullyQualifiedName Property

```
[ILASM]
.property string FullyQualifiedName { public hideby sig
virtual specialname string get_FullyQualifiedName() }

[C#]
public virtual string FullyQualifiedName { get; }
```

Summary

Gets a string that represents the full path of the module reflected by the current instance.

Property Value

A **System.String** that represents the full path of the module reflected by the current instance. If the assembly that contains the module reflected by the current instance was loaded from a **System.Byte** array, the value of this string is "<Unknown>".

Description

This property is read-only.

Behaviors

The case-sensitivity of the module name is implementation-specific.

How and When to Override

Override this property to customize the content of the **System.String** returned by this property in types derived from **System.Reflection.Module**.

Usage

To obtain the name of the module without path information, use the **System.Reflection.Module.Name** property.

Permissions

Permission	Description
System.Security.Permissions.FileIOPermission	Requires permission to access path information. See System.Security.Permissions.FileIOPermission and System.Security.Permissions.FileIOPermissionAccess .

1
2
3

	PathDiscovery.
--	-----------------------

1 Module.Name Property

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

7 Summary

8 Gets a string containing the name of the module reflected by the
9 current instance, with the path component removed.

10 Property Value

11

12 A **System.String** containing the name of the module reflected by the
13 current instance, with the path component removed. If the assembly
14 that contains the module reflected by the current instance was loaded
15 from a **System.Byte** array, the value of this string is "<Unknown>".

16 Description

17 This property is read-only.

18

19 [Note: The value of this property is equivalent to the value of the
20 string returned by the **System.Reflection.Module.ToString** method.

21

22 Use **System.Reflection.Module.FullyQualifiedName** to get the
23 name and path of the module reflected by the current instance.]

24