System.Reflection.EventInfo Class

[ILASM]

```
.class public abstract EventInfo extends
System.Reflection.MemberInfo
```

[C#]

```
public abstract class EventInfo: MemberInfo
```

Assembly Info:

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

Summary

Provides access to event metadata.

Inherits From: System.Reflection.MemberInfo

Library: Reflection

Thread Safety: This type is safe for multithreaded operations.

Description

Events are handled by delegates. An event listener supplies an event-handler delegate that is invoked whenever the event is raised by an event source. In order to connect to the event source, the event listener adds this delegate to the invocation list of the source. When the event is raised, the event-handler delegate invokes the methods in its invocation list. The **System.Reflection.EventInfo.GetAddMethod**, **System.Reflection.EventInfo.AddEventHandler**, **System.Reflection.EventInfo.GetRemoveMethod**, and **System.Reflection.EventInfo.RemoveEventHandler** methods, and the delegate type of the event-handler associated with an event, are required to be marked in the metadata.

[*Note:* For information on delegates, see the **System.Delegate** class overview.]

[*Note:* For information on events, see Partitions I and II of the CLI specification.]

EventInfo() Constructor

[ILASM]

```
family rtspecialname specialname instance void .ctor()
```

[C#]

```
protected EventInfo()
```

Summary

Constructs a new instance of the **System.Reflection.EventInfo** class.

EventInfo.AddEventHandler(System.Object, System.Delegate) Method

```
[ILASM]
.method public hidebysig instance void
AddEventHandler(object target, class System.Delegate
handler)

[C#]
public void AddEventHandler(object target, Delegate
handler)
```

Summary

Adds the specified event handler delegate to the specified event source.

Parameters

Parameter	Description
target	An object that represents an event source.
handler	A **System.Delegate** instance to be added to *target* that references methods to be invoked when the event reflected by the current instance is raised by *target*.

Description

Each time the event reflected by the current instance is raised by *target*, the methods in the invocation list of *handler* are invoked.

Exceptions

Exception	Condition
System.ArgumentException	*handler* is not the same type **System.Type** as the event handler delegate declared for the event reflected by the current instance.
System.Reflection.TargetException	The event reflected by the current instance is non-static, and *obj* is **null** or is of a type that does not implement the event reflected by the current instance.

EventInfo.GetAddMethod(System.Boolean) Method

```
[ILASM]
.method public hidebysig virtual abstract class
System.Reflection.MethodInfo GetAddMethod(bool nonPublic)

[C#]
public abstract MethodInfo GetAddMethod(bool nonPublic)
```

Summary

Returns the method used to add an event handler delegate to an event source for the event reflected by the current instance, specifying whether or not to return non-public methods.

Parameters

Parameter	Description
nonPublic	A **System.Boolean** value that specifies whether non-public methods can be returned by this method. Specify **true** to return non-public methods; otherwise, specify **false**.

Return Value

A **System.Reflection.MethodInfo** instance that reflects the method used to add an event handler delegate to an event source for the event reflected by the current instance, if found; otherwise, returns **null**.

Description

[*Note:* The returned method is used to add an event-handler delegate to the invocation list of an event source. Typically, the method has the following signature format:

`add_<EventName>(<EventHandlerType> handler)`]

Behaviors

As described above.

Exceptions

Exception	Condition
System.MethodAccessException	*nonPublic* is **true**, the method used to add an event handler delegate is non-public, and the caller does not have permission to reflect on non-public methods.

Permissions

Permission	Description
System.Security.Permissions.ReflectionPermission	Requires permission to reflect non-public members of a type in loaded assemblies. See **System.Security.Permissions.ReflectionPermissionFlag.TypeInformation**.

EventInfo.GetAddMethod() Method

[ILASM]

```
.method public hidebysig instance class
System.Reflection.MethodInfo GetAddMethod()
```

[C#]

```
public MethodInfo GetAddMethod()
```

Summary

Returns the public method used to add an event handler delegate to an event source for the event reflected by the current instance.

Return Value

A **System.Reflection.MethodInfo** instance that reflects the public method used to add an event handler delegate to an event source for the event reflected by the current instance, if found; otherwise, returns **null**.

Description

This method is equivalent to **System.Reflection.EventInfo.GetAddMethod**(**false**).

[*Note:* The returned method is used to add an event-handler delegate to the invocation list of an event source. Typically, the method has the following signature format:

`add_<EventName>(<EventHandlerType> handler)`]

EventInfo.GetRaiseMethod(System.Boolean) Method

```
[ILASM]
.method public hidebysig virtual abstract class
System.Reflection.MethodInfo GetRaiseMethod(bool nonPublic)

[C#]
public abstract MethodInfo GetRaiseMethod(bool nonPublic)
```

Summary

Returns the method that is called when the event reflected by the current instance is raised, specifying whether the method to be returned is public or non-public.

Parameters

Parameter	Description
nonPublic	A **System.Boolean** value that specifies whether non-public methods can be returned by this method. Specify **true** to return non-public methods; otherwise, specify **false**.

Return Value

A **System.Reflection.MethodInfo** instance that reflects the method that is called when the event reflected by the current instance is raised, if found; otherwise, returns **null**.

Behaviors

As described above.

Exceptions

Exception	Condition
System.MethodAccessException	*nonPublic* is **true**, the method used to raise the event is non-public, and the caller does not have permission to reflect on non-public methods.

Permissions

Permission	Description

System.Security.Permissions.ReflectionPermission	Requires permission to reflect non-public members of a type in loaded assemblies. See **System.Security.Permissions.ReflectionPermissionFlag.TypeInformation**.

EventInfo.GetRaiseMethod() Method

[ILASM]

```
.method public hidebysig instance class
System.Reflection.MethodInfo GetRaiseMethod()
```

[C#]

```
public MethodInfo GetRaiseMethod()
```

Summary

Returns the public method that is called when the event reflected by the current instance is raised.

Return Value

A **System.Reflection.MethodInfo** instance that reflects the public method that is called when the event reflected by the current instance is raised, if found; otherwise, returns **null**.

EventInfo.GetRemoveMethod(System.Boolean) Method

[ILASM]

```
.method public hidebysig virtual abstract class
System.Reflection.MethodInfo GetRemoveMethod(bool
nonPublic)
```

[C#]

```
public abstract MethodInfo GetRemoveMethod(bool nonPublic)
```

Summary

Returns the method used to remove an event-handler delegate from the event reflected by the current instance, specifying whether or not to return non-public methods.

Parameters

Parameter	Description
nonPublic	A **System.Boolean** value that specifies whether non-public methods can be returned by this method. Specify **true** to return non-public methods; otherwise, specify **false**.

Return Value

A **System.Reflection.MethodInfo** instance that reflects the method used to remove an event handler delegate from the event reflected by the current instance, if found; otherwise, returns **null**.

Description

[*Note:* Typically, the method has the following signature format:

`remove_<EventName>(<EventHandlerType> handler)`]

Behaviors

As described above.

Exceptions

Exception	Condition
System.MethodAccessException	*nonPublic* is **true**, the method used to remove an event handler delegate is non-public, and the

	caller does not have permission to reflect on non-public methods.

Permissions

Permission	Description
System.Security.Permissions.ReflectionPermission	Requires permission to reflect non-public members of a type in loaded assemblies. See **System.Security.Permissions.ReflectionPermissionFlag.TypeInformation**.

EventInfo.GetRemoveMethod() Method

[ILASM]

```
.method public hidebysig instance class
System.Reflection.MethodInfo GetRemoveMethod()
```

[C#]

```
public MethodInfo GetRemoveMethod()
```

Summary

Returns the public method used to remove an event-handler delegate from the event reflected by the current instance.

Return Value

A **System.Reflection.MethodInfo** instance that reflects the public method used to remove an event handler delegate from the event reflected by the current instance, if found; otherwise, returns **null**.

Description

This method is equivalent to **System.Reflection.EventInfo.GetRemoveMethod**(**false**).

[*Note:* Typically, the method has the following signature format:

`remove_<EventName>(<EventHandlerType> handler)`]

EventInfo.RemoveEventHandler(System.Object, System.Delegate) Method

```
[ILASM]
.method public hidebysig instance void
RemoveEventHandler(object target, class System.Delegate
handler)

[C#]
public void RemoveEventHandler(object target, Delegate
handler)
```

Summary

Removes the specified event handler delegate from the specified event source.

Parameters

Parameter	Description
target	An object that represents an event source.
handler	A **System.Delegate** instance to be disassociated from the events reflected by the current instance that are raised by *target*.

Description

After this method is invoked, subsequent events reflected by the current instance that are raised by *target* will no longer cause *handler* to invoke its methods.

Exceptions

Exception	Condition
System.ArgumentException	*handler* is not the same type **System.Type** as the event handler delegate declared for the event reflected by the current instance.

EventInfo.Attributes Property

[ILASM]

```
.property valuetype System.Reflection.EventAttributes
Attributes { public hidebysig virtual abstract specialname
valuetype System.Reflection.EventAttributes
get_Attributes() }
```

[C#]

```
public abstract EventAttributes Attributes { get; }
```

Summary

Gets the attributes of the event reflected by the current instance.

Property Value

A **System.Reflection.EventAttributes** value that specifies the attributes in the metadata of the event reflected by the current instance.

EventInfo.EventHandlerType Property

[ILASM]

```
.property class System.Type EventHandlerType { public
hidebysig specialname instance class System.Type
get_EventHandlerType() }
```

[C#]

```
public Type EventHandlerType { get; }
```

Summary

Gets the **System.Type** of the event-handler **System.Delegate** associated with the event reflected by the current instance.

Property Value

A **System.Type** that represents the type of the event-handler **System.Delegate** associated with the event reflected by the current instance. Returns **null** if the method used to add a delegate to the event is not public and is in a loaded assembly, and the caller does not have the required permission.

Description

This property is read-only.

Permissions

Permission	Description
System.Security.Permissions. ReflectionPermission	Requires permission to reflect non-public members of a type in loaded assemblies. See **System.Security.Permissions. ReflectionPermissionFlag.TypeInformation**.