System.Xml.XmlNameTable Class

```
[ILAsm]
.class public abstract XmlNameTable extends System.Object

[C#]
public abstract class XmlNameTable
```

Assembly Info:

- *Name:* System.Xml
- *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)

Summary

Creates a table that stores unique instances of `System.String` objects.

Inherits From: System.Object

Library: XML

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

Description

Only a single instance of any given string is stored even if the string is added multiple times to the table.

Using this class provides an efficient means for an XML parser to use the same `System.String` object for all repeated element and attribute names in an XML document. If the same object is used for all repeated names, the efficiency of name comparisons is increased by allowing the names to be compared using object comparisons rather than string comparisons.

[*Note:* This class is `abstract` and is implemented in the `System.Xml.NameTable` class.

]

Example

See the `System.Xml.NameTable` class for an example using this class.

XmlNameTable() Constructor

```
[ILAsm]
family rtspecialname specialname instance void .ctor()


[C#]
protected XmlNameTable()
```

Summary

Constructs a new instance of the `System.Xml.XmlNameTable` class.

XmlNameTable.Add(System.String) Method

```
[ILAsm]
.method public hidebysig virtual abstract string Add(string array)


[C#]
public abstract string Add(string array)
```

Summary

Adds the specified System.String to the table if a System.String instance with the same value does not already exist in the table.

Parameters

Parameter	Description
array	The System.String to add.

Return Value

array, if it did not exist in the table at the time of the call, or the System.String instance previously stored in the table with a value equal to *array*.

Behaviors

Only a single instance of any given System.String is stored in the table. If the value of *array* is already stored in the table, the System.String instance with that value is returned.

How and When to Override

This method must be overridden in order to provide the functionality described above, as there is no default implementation.

Exceptions

Exception	Condition
System.ArgumentNullException	*array* is null.

XmlNameTable.Add(System.Char[], System.Int32, System.Int32) Method

```
[ILAsm]
.method public hidebysig virtual abstract string Add(char[] array,
int32 offset, int32 length)


[C#]
public abstract string Add(char[] array, int offset, int length)
```

Summary

Adds the `System.String` equivalent of a specified subset of a `System.Char` array to the table if the string equivalent does not already exist in the table.

Parameters

Parameter	Description
array	A `System.Char` array containing the string to add.
offset	A `System.Int32` specifying the zero-based index into the array of the first character of the string.
length	A `System.Int32` containing the number of characters in the string.

Return Value

The `System.String` equivalent of the specified subset of the `System.Char` array that is stored in the table, or `System.String.Empty` if *length* is zero.

Behaviors

Only a single instance of any given `System.String` is stored in the table. Calling this method with the same subset (containing the same characters) of any `System.Char` array, returns the same instance of the `System.String` equivalent.

How and When to Override

This method must be overridden in order to provide the functionality described above, as there is no default implementation.

Exceptions

Exception	Condition
System.IndexOutOfRangeException	*offset* < 0. - or - *offset* >= *array*.Length. - or - *length* > *array*.Length - *offset*. The above conditions do not cause an exception to be thrown if *length* = 0.
System.ArgumentOutOfRangeException	*length* < 0.

XmlNameTable.Get(System.String) Method

```
[ILAsm]
.method public hidebysig virtual abstract string Get(string array)

[C#]
public abstract string Get(string array)
```

Summary

Looks up the value of the specified System.String in the table.

Parameters

Parameter	Description
array	The System.String to look up.

Return Value

The System.String instance previously stored in the table with a value equal to *array*, or null if it does not exist.

Behaviors

Only a single instance of any given System.String is stored in the table. If the value of *array* is already stored in the table, the System.String instance with that value is returned.

How and When to Override

This method must be overridden in order to provide the functionality described above, as there is no default implementation.

Exceptions

Exception	Condition
System.ArgumentNullException	*array* is null.

XmlNameTable.Get(System.Char[], System.Int32, System.Int32) Method

```
[ILAsm]
.method public hidebysig virtual abstract string Get(char[] array,
int32 offset, int32 length)


[C#]
public abstract string Get(char[] array, int offset, int length)
```

Summary

Looks up the System.String equivalent of a specified subset of a System.Char array in the table.

Parameters

Parameter	Description
array	A System.Char array containing the string to look up.
offset	A System.Int32 specifying the zero-based index into the array of the first character of the string.
length	A System.Int32 containing the number of characters in the string.

Return Value

The System.String equivalent of the specified subset of the System.Char array that is stored in the table, or null if the equivalent System.String is not in the table.

Behaviors

Only a single instance of any given System.String is stored in the table. Calling this method with the same subset (containing the same characters) of any System.Char array, returns the same instance of the System.String equivalent, if it exists.

How and When to Override

This method must be overridden in order to provide the functionality described above, as there is no default implementation.

Exceptions

Exception	Condition
System.IndexOutOfRangeException	*offset* < 0. - or - *offset* >= *array*.Length. - or - *length* > *array*.Length - *offset*. The above conditions do not cause an exception to be thrown if *length* = 0.
System.ArgumentOutOfRangeException	*length* < 0.