OpcName Members

Namespace: Opc.UaFx
Assemblies: Opc.UaFx.Advanced.dll, Opc.UaFx.Advanced.dll
The OpcName type exposes the following members.

Constructors

OpcName(String)

Initializes a new instance of the OpcName class using the `value` specified.

C#

```
public OpcName(string value)
```

Parameters

`value` String

The String value of the name to represent. A null reference (Nothing in Visual Basic) or a value equals Empty results into a null name (see IsNull).

OpcName(String, Int32)

Initializes a new instance of the OpcName class using the `value` and `namespaceIndex` specified.

C#

```
public OpcName(string value, int namespaceIndex)
```

Parameters

`value` String

The String value of the name to represent. A null reference (Nothing in Visual Basic) or a value equals Empty results into a null name (see IsNull).

`namespaceIndex` Int32

The index of the namespace that this name should refer to.

OpcName(String, OpcNamespace)

Initializes a new instance of the OpcName class using the `value` and `nameNamespace` specified.

C#

```
public OpcName(string value, OpcNamespace nameNamespace)
```

Parameters

`value` String

The String value of the name to represent. A null reference (Nothing in Visual Basic) or a value equals Empty results into a null name (see IsNull).

`nameNamespace` OpcNamespace

The namespace that this name should refer to.

Properties

IsNull

Gets a value indicating whether the name is a null name.

C#

```
public bool IsNull { get; }
```

Property Value

Boolean

The value true if the Value is a null reference (Nothing in Visual Basic) or equals to Empty; otherwise the value false.

Namespace

Gets the OpcNamespace used by the OpcName to describe the namespace to that the name belongs.

C#

```
public OpcNamespace Namespace { get; }
```

Property Value

OpcNamespace

An instance of the OpcNamespace with the known information about the namespace to that the OpcName belongs.

NamespaceIndex

Gets the index of the namespace that this name belongs.

C#

```
public int NamespaceIndex { get; }
```

Property Value

Int32

The index value which refers to the namespace in the namespace array of the server to that the name belong.

Remarks

The available namespaces of the server can be retrieved through reading the value of the namespaces-node using the node identifier 'OpcObjectTypes.Server.Namespaces'.

NamespaceUri

Gets the uniform resource identifier (URI) of the namespace referred by the NamespaceIndex.

C#

```
public Uri NamespaceUri { get; }
```

Property Value

Uri

The Uri to that the OpcName refers using the NamespaceIndex.

Null

Gets the default null name.

C#

```
public static OpcName Null { get; }
```

Property Value

OpcName

An instance of the OpcName which can be used for general purpose in cases there a null name is enough.

Value

Gets the name without any namespace information.

C#

```
public string Value { get; }
```

Property Value

String

The (unqualified) value of the name.

Methods

CompareTo(Object)

Compares the current OpcName with the `other`.

C#

```
public int CompareTo(object other)
```

Parameters

`other` Object

The OpcName to compare with this OpcName.

Returns

Int32

A 32-bit signed integer that indicates the relative order of the objects being compared (CompareTo(Object)).

CompareTo(OpcName)

Compares the current OpcName with another OpcName.

C#

```
public int CompareTo(OpcName other)
```

Parameters

`other` OpcName

The OpcName to compare with this OpcName.

Returns

Int32

A 32-bit signed integer that indicates the relative order of the objects being compared (CompareTo()).

Equals(Object)

Determines whether the specified `other` is equal to this OpcName.

C#

```
public override bool Equals(object other)
```

Parameters

`other` Object

The OpcName to compare to the current OpcName.

Returns

Boolean

The value true if the specified OpcName is equal to the current OpcName; otherwise the value false.

Equals(OpcName)

Determines whether the specified `other` is equal to this OpcName.

C#

```
public bool Equals(OpcName other)
```

Parameters

`other` OpcName

The OpcName to compare to the current OpcName.

Returns

Boolean

The value true if the specified OpcName is equal to the current OpcName; otherwise the value false.

GetHashCode()

Retrieves a hash code for this OpcName.

C#

```
public override int GetHashCode()
```

Returns

Int32

An Int32 that contains the hash code for the OpcName.

IsNullOrEmpty(OpcName)

Indicates whether the specified OpcName is null or its Value is null or an empty string (“”).

C#

```
public static bool IsNullOrEmpty(OpcName name)
```

Parameters

`name` OpcName

The OpcName to test.

Returns

Boolean

The value true if the `name` parameter or its Value is null or an empty string (“”); otherwise the value false.

Remarks

This method ignores the circumstance whether the Namespace of the `name` might be specified anyway.

Parse(String)

Converts a name string to a OpcName instance.

C#

```
public static OpcName Parse(string value)
```

Parameters

`value` String

A string that contains a name.

Returns

OpcName

An instance of the OpcName class.

Exceptions

FormatException

The `value` is not a valid name.

ToString()

Returns a string representing the name.

C#

```
public override string ToString()
```

Returns

String

A string formatted with the Value and Namespace of the name.

TryParse(String, out OpcName)

Determines whether a string is a valid name.

C#

```
public static bool TryParse(string value, out OpcName name)
```

Parameters

`value` String

The string to validate.

`name` OpcName

The OpcName version of the string.

Returns

Boolean

The value true, if `value` is a valid name; otherwise the value false.

Operators

Equality(OpcName, OpcName)

Returns a value indicating whether two instance of OpcName are equal.

C#

```
public static bool operator ==(OpcName left, OpcName right)
```

Explicit(OpcName to QualifiedName)

Converts a OpcName to an QualifiedName object.

C#

```
[CLSCompliant(false)]
public static explicit operator QualifiedName(OpcName value)
```

GreaterThan(OpcName, OpcName)

Determines whether the first specified OpcName object is greater than the second specified OpcName object.

C#

```
public static bool operator>(OpcName left, OpcName right)
```

GreaterThanOrEqual(OpcName, OpcName)

Determines whether the first specified OpcName object is greater than or equal to the second specified OpcName object.

C#

```
public static bool operator >=(OpcName left, OpcName right)
```

Implicit(QualifiedName to OpcName)

Converts a QualifiedName to an OpcName object.

C#

```
[CLSCompliant(false)]
public static implicit operator OpcName(QualifiedName value)
```

Implicit(String to OpcName)

Converts a String to an OpcName object.

C#

```
public static implicit operator OpcName(string value)
```

Inequality(OpcName, OpcName)

Returns a value indicating whether two instances of OpcName are not equal.

C#

```
public static bool operator !=(OpcName left, OpcName right)
```

LessThan(OpcName, OpcName)

Determines whether the first specified OpcName object is less than the second specified OpcName object.

C#

```
public static bool operator <(OpcName left, OpcName right)
```

Exceptions

ArgumentNullException

The `left` is a null reference (Nothing in Visual Basic).

LessThanOrEqual(OpcName, OpcName)

Determines whether the first specified OpcName object is less than or equal to the second OpcName object.

C#

```
public static bool operator <=(OpcName left, OpcName right)
```

Exceptions

ArgumentNullException

The `left` is a null reference (Nothing in Visual Basic).

Table of Contents