OpcNamePath Members

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

Constructors

OpcNamePath(IEnumerable<OpcName>)

Initializes a new instance of the OpcNamePath class using the `pathElements` specified.

C#

```
public OpcNamePath(IEnumerable<OpcName> pathElements)
```

Parameters

`pathElements` IEnumerable<OpcName>

The elements of the new OpcNamePath.

Exceptions

ArgumentNullException

One of the items in `pathElements` is a null reference (Nothing in Visual Basic).

OpcNamePath(IEnumerable<String>)

Initializes a new instance of the OpcNamePath class using the `pathElements` specified.

C#

```
public OpcNamePath(IEnumerable<string> pathElements)
```

Parameters

`pathElements` IEnumerable<String>

The elements of the new OpcNamePath which are converted to OpcName instances used by the new instance.

Exceptions

ArgumentException

One of the items in `pathElements` is an empty string.

ArgumentNullException

One of the items in `pathElements` is a null reference (Nothing in Visual Basic).

OpcNamePath(OpcName[])

Initializes a new instance of the OpcNamePath class using the `pathElements` specified.

C#

```
public OpcNamePath(params OpcName[] pathElements)
```

Parameters

`pathElements` OpcName[]

The elements of the new OpcNamePath.

Exceptions

ArgumentNullException

One of the items in `pathElements` is a null reference (Nothing in Visual Basic).

OpcNamePath(String[])

Initializes a new instance of the OpcNamePath class using the `pathElements` specified.

C#

```
public OpcNamePath(params string[] pathElements)
```

Parameters

`pathElements` String[]

The elements of the new OpcNamePath which are converted to OpcName instances used by the new instance.

Exceptions

ArgumentException

One of the items in `pathElements` is an empty string.

ArgumentNullException

One of the items in `pathElements` is a null reference (Nothing in Visual Basic).

Properties

Empty

Gets the default empty path.

C#

```
public static OpcNamePath Empty { get; }
```

Property Value

OpcNamePath

An instance of the OpcNamePath which can be used for general purpose in cases there an empty path is enough.

IsEmpty

Gets a value indicating whether the path is empty.

C#

```
public bool IsEmpty { get; }
```

Property Value

Boolean

The value true if there exist no element in the path; otherwise the value false.

Item[Int32]

Gets the OpcName at the specified index.

C#

```
public OpcName this[int index] { get; }
```

Property Value

OpcName

The OpcName at the specified index.

Exceptions

ArgumentOutOfRangeException

The `index` is not valid index in the OpcNamePath.

Length

Gets the number of elements contained in the OpcNamePath.

C#

```
public int Length { get; }
```

Property Value

Int32

The zero-based number of elements.

Methods

Combine(OpcNamePath[])

Combines the `paths` specified into one path there its sequence of elements is created from the consecutive elements of the `paths` passed.

C#

```
public static OpcNamePath Combine(params OpcNamePath[] paths)
```

Parameters

`paths` OpcNamePath[]

The array of OpcNamePath instances its elements have to be combined to a new OpcNamePath.

Returns

OpcNamePath

A new instance of the OpcNamePath class with the elements of the first path followed by the elements of the second path and so on for all other `paths` specified.

Exceptions

ArgumentNullException

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

Remarks

In case there in `paths` is a null reference (Nothing in Visual Basic) it will be ignored.

CompareTo(Object)

Compares the current OpcNamePath with the `other`.

C#

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

Parameters

`other` Object

The OpcNamePath to compare with this OpcNamePath.

Returns

Int32

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

CompareTo(OpcNamePath)

Compares the current OpcNamePath with another OpcNamePath.

C#

```
public int CompareTo(OpcNamePath other)
```

Parameters

`other` OpcNamePath

The OpcNamePath to compare with this OpcNamePath.

Returns

Int32

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

Contains(OpcNamePath)

Returns a value indicating whether a specified sub-path occurs within this OpcNamePath.

C#

```
public bool Contains(OpcNamePath path)
```

Parameters

`path` OpcNamePath

The OpcNamePath to seek.

Returns

Boolean

The value true if `path` is a sub-path of this OpcNamePath; otherwise the value false.

Exceptions

ArgumentNullException

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

Equals(Object)

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

C#

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

Parameters

`other` Object

The OpcNamePath to compare to the current OpcNamePath.

Returns

Boolean

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

Equals(OpcNamePath)

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

C#

```
public bool Equals(OpcNamePath other)
```

Parameters

`other` OpcNamePath

The OpcNamePath to compare to the current OpcNamePath.

Returns

Boolean

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

GetHashCode()

Retrieves a hash code for this OpcNamePath.

C#

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

Returns

Int32

An Int32 that contains the hash code for the OpcNamePath.

Parse(String)

Converts a path string to a OpcNamePath instance.

C#

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

Parameters

`value` String

A string that contains a path.

Returns

OpcNamePath

An instance of the OpcNamePath class.

Exceptions

FormatException

The `value` is not a valid path.

ToArray()

Copies the elements of the path to a new array of OpcName instances.

C#

```
public OpcName[] ToArray()
```

Returns

OpcName[]

An array of OpcName instances representing the elements of the path.

ToString()

Returns a string representing the path.

C#

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

Returns

String

A string formatted with the elements of the path.

TryParse(String, out OpcNamePath)

Determines whether a string is a valid path.

C#

```
public static bool TryParse(string value, out OpcNamePath path)
```

Parameters

`value` String

The string to validate.

`path` OpcNamePath

The OpcNamePath version of the string.

Returns

Boolean

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

Operators

Equality(OpcNamePath, OpcNamePath)

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

C#

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

GreaterThan(OpcNamePath, OpcNamePath)

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

C#

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

GreaterThanOrEqual(OpcNamePath, OpcNamePath)

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

C#

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

Implicit(OpcName to OpcNamePath)

Creates a new instance of the OpcNamePath from the `value` specified.

C#

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

Implicit(String to OpcNamePath)

Converts a String to an OpcNamePath object.

C#

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

Inequality(OpcNamePath, OpcNamePath)

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

C#

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

LessThan(OpcNamePath, OpcNamePath)

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

C#

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

Exceptions

ArgumentNullException

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

LessThanOrEqual(OpcNamePath, OpcNamePath)

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

C#

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

Exceptions

ArgumentNullException

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

Table of Contents