

1 System.Xml.XmlParserContext Class

2
3

```
4 [ILASM]  
5 .class public XmlParserContext extends System.Object  
6 [C#]  
7 public class XmlParserContext
```

8 Assembly Info:

- 9 • Name: System.Xml
- 10 • Public Key: [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00]
- 11 • Version: 1.0.x.x
- 12 • Attributes:
 - 13 ○ CLSCompliantAttribute(true)

14 Summary

15

16 Provides all the context information required by instances of the
17 **System.Xml.XmlTextReader** class to parse an XML fragment.

18 Inherits From: System.Object

19

20 **Library:** XML

21

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

24

25

26

XmlParserContext(System.Xml.XmlNameT able, System.Xml.XmlNamespaceManager, System.String, System.Xml.XmlSpace) Constructor

```
[ILASM]
public rtspecialname specialname instance void .ctor(class
System.Xml.XmlNameTable nt, class
System.Xml.XmlNamespaceManager nsMgr, string xmlLang,
valuetype System.Xml.XmlSpace xmlSpace)

[C#]
public XmlParserContext(XmlNameTable nt,
XmlNamespaceManager nsMgr, string xmlLang, XmlSpace
xmlSpace)
```

Summary

Constructs and initializes a new instance of the **System.Xml.XmlParserContext** class with the specified values.

Parameters

Parameter	Description
<i>nt</i>	The System.Xml.XmlNameTable to use. If <i>nt</i> is null , this defaults to the System.Xml.XmlNameTable used to construct <i>nsMgr</i> .
<i>nsMgr</i>	The System.Xml.XmlNamespaceManager to use for looking up namespace information.
<i>xmlLang</i>	A System.String specifying the <code>xml:lang</code> scope.
<i>xmlSpace</i>	A System.Xml.XmlSpace value indicating the <code>xml:space</code> scope.

Description

This method is equivalent to **System.Xml.XmlParserContext**(*nt*, *nsMgr*, **null**, **null**, **null**, **System.String.Empty**, *xmlLang*, *xmlSpace*, **null**) constructor.

Exceptions

Exception	Condition
System.Xml.XmlException	<i>nt</i> is not the same System.Xml.XmlNameTable used to construct <i>nsMgr</i> .

1
2
3

XmlParserContext(System.Xml.XmlNameTable, System.Xml.XmlNamespaceManager, System.String, System.Xml.XmlSpace, System.Text.Encoding) Constructor

```
[ILASM]
public rtspecialname specialname instance void .ctor(class
System.Xml.XmlNameTable nt, class
System.Xml.XmlNamespaceManager nsMgr, string xmlLang,
valuetype System.Xml.XmlSpace xmlSpace, class
System.Text.Encoding enc)

[C#]
public XmlParserContext(XmlNameTable nt,
XmlNamespaceManager nsMgr, string xmlLang, XmlSpace
xmlSpace, Encoding enc)
```

Summary

Constructs and initializes a new instance of the **System.Xml.XmlParserContext** class with the specified values.

Parameters

Parameter	Description
<i>nt</i>	The System.Xml.XmlNameTable to use. If <i>nt</i> is null , this defaults to the System.Xml.XmlNameTable used to construct <i>nsMgr</i> .
<i>nsMgr</i>	The System.Xml.XmlNamespaceManager to use for looking up namespace information.
<i>xmlLang</i>	A System.String specifying the <code>xml:lang</code> scope.
<i>xmlSpace</i>	A System.Xml.XmlSpace value indicating the <code>xml:space</code> scope.
<i>enc</i>	An instance of a class derived from the System.Text.Encoding class indicating the encoding to use.

Description

This method is equivalent to **System.Xml.XmlParserContext**(*nt*, *nsMgr*, **null**, **null**, **null**, **System.String.Empty**, *xmlLang*, *xmlSpace*, *enc*).

Exceptions

Exception	Condition
-----------	-----------

1
2
3

System.Xml.XmlException	<i>nt</i> is not the same System.Xml.XmlNameTable used to construct <i>nsMgr</i> .
--------------------------------	-------------------------------------------------------------------------------------------

1 XmlParserContext(System.Xml.XmlNameT 2 able, System.Xml.XmlNamespaceManager, 3 System.String, System.String, 4 System.String, System.String, 5 System.String, System.String, 6 System.Xml.XmlSpace) Constructor

```
7 [ILASM]  
8 public rtspecialname specialname instance void .ctor(class  
9 System.Xml.XmlNameTable nt, class  
10 System.Xml.XmlNamespaceManager nsMgr, string docTypeName,  
11 string pubId, string sysId, string internalSubset, string  
12 baseURI, string xmlLang, valuetype System.Xml.XmlSpace  
13 xmlSpace)  
  
14 [C#]  
15 public XmlParserContext(XmlNameTable nt,  
16 XmlNamespaceManager nsMgr, string docTypeName, string  
17 pubId, string sysId, string internalSubset, string baseURI,  
18 string xmlLang, XmlSpace xmlSpace)
```

19 Summary

20 Constructs and initializes a new instance of the
21 **System.Xml.XmlParserContext** class with the specified values.

22 Parameters

23
24

Parameter	Description
<i>nt</i>	The System.Xml.XmlNameTable to use. If <i>nt</i> is null , this defaults to the System.Xml.XmlNameTable used to construct <i>nsMgr</i> .
<i>nsMgr</i>	The System.Xml.XmlNamespaceManager to use for looking up namespace information.
<i>docTypeName</i>	A System.String specifying the name of the document type declaration.
<i>pubId</i>	A System.String specifying the public identifier.
<i>sysId</i>	A System.String specifying the system identifier.
<i>internalSubset</i>	A System.String specifying the internal DTD subset.
<i>baseURI</i>	A System.String specifying the base URI for the XML fragment (the location from which the fragment was loaded).
<i>xmlLang</i>	A System.String containing the <code>xml:lang</code> scope.
<i>xmlSpace</i>	A System.Xml.XmlSpace value indicating the <code>xml:space</code> scope.

1

2 **Description**

3 This method is equivalent to **System.Xml.XmlParserContext** (*nt*,
4 *nsMgr*, *docTypeName*, *pubId*, *sysId*, *internalSubset*, *baseUri*, *xmlLang*,
5 *xmlSpace*, **null**).

6 **Exceptions**

7

8

Exception	Condition
System.Xml.XmlException	<i>nt</i> is not the same System.Xml.XmlNameTable used to construct <i>nsMgr</i> .

9

10

11

1 XmlParserContext(System.Xml.XmlNameT 2 able, System.Xml.XmlNamespaceManager, 3 System.String, System.String, 4 System.String, System.String, 5 System.String, System.String, 6 System.Xml.XmlSpace, 7 System.Text.Encoding) Constructor

```
8 [ILASM]  
9 public rtspecialname specialname instance void .ctor(class  
10 System.Xml.XmlNameTable nt, class  
11 System.Xml.XmlNamespaceManager nsMgr, string docTypeName,  
12 string pubId, string sysId, string internalSubset, string  
13 baseURI, string xmlLang, valuetype System.Xml.XmlSpace  
14 xmlSpace, class System.Text.Encoding enc)  
  
15 [C#]  
16 public XmlParserContext(XmlNameTable nt,  
17 XmlNamespaceManager nsMgr, string docTypeName, string  
18 pubId, string sysId, string internalSubset, string baseURI,  
19 string xmlLang, XmlSpace xmlSpace, Encoding enc)
```

20 Summary

21 Constructs and initializes a new instance of the
22 **System.Xml.XmlParserContext** class with the specified values.

23 Parameters

24
25

Parameter	Description
<i>nt</i>	The System.Xml.XmlNameTable to use. If <i>nt</i> is null , this defaults to the System.Xml.XmlNameTable used to construct <i>nsMgr</i> .
<i>nsMgr</i>	The System.Xml.XmlNamespaceManager to use for looking up namespace information.
<i>docTypeName</i>	A System.String specifying the name of the document type declaration.
<i>pubId</i>	A System.String specifying the public identifier.
<i>sysId</i>	A System.String specifying the system identifier.
<i>internalSubset</i>	A System.String specifying the internal DTD subset.
<i>baseURI</i>	A System.String specifying the base URI for the XML fragment (the location from which the fragment was loaded).
<i>xmlLang</i>	A System.String specifying the <code>xml:lang</code> scope.

<i>xmlSpace</i>	A System.Xml.XmlSpace value indicating the <code>xml:space</code> scope.
<i>enc</i>	The System.Text.Encoding to use.

1

2 **Description**

3 The constructor sets **System.Xml.XmlParserContext.BaseURI** to
 4 *baseURI*, **System.Xml.XmlParserContext.DocTypeName** to
 5 *docTypeName*, **System.Xml.XmlParserContext.InternalSubset** to
 6 *internalSubset*, **System.Xml.XmlParserContext.PublicId** to *pubId*,
 7 **System.Xml.XmlParserContext.SystemId** to *sysId*, and
 8 **System.Xml.XmlParserContext.XmlLang** to *xmlLang*. If **null** is
 9 passed for any of these parameters, the corresponding property is set
 10 to **System.String.Empty**.

11

12 [Note: The DocumentType (DTD) information stored in this constructor
 13 is ignored when an instance of the class is passed to a
 14 **System.Xml.XmlTextReader**.]

15 **Exceptions**

16

17

Exception	Condition
System.Xml.XmlException	<i>nt</i> is not the same System.Xml.XmlNameTable used to construct <i>nsMgr</i> .

18

19

20

1 XmlParserContext.BaseURI Property

```
2 [ILASM]
3 .property string BaseURI { public hidebysig specialname
4 instance string get_BaseURI() public hidebysig specialname
5 instance void set_BaseURI(string value) }
6
7 [C#]
8 public string BaseURI { get; set; }
```

8 Summary

9 Gets or sets the base URI.

10 Property Value

11

12 A **System.String** specifying the base URI to use for resolving the DTD
13 file.

14 Description

15 If an attempt is made to set this property to **null**, it is set to
16 **System.String.Empty**.

17

18 [*Note:* A networked XML document is comprised of chunks of data
19 aggregated using various W3C standard inclusion mechanisms and
20 therefore may contain nodes that come from different places. The
21 **System.Xml.XmlParserContext.BaseURI** property shows where
22 these nodes originated.]

23

1 XmlParserContext.DocTypeName Property

```
2 [ILASM]
3 .property string DocTypeName { public hidebysig specialname
4 instance string get_DocTypeName() public hidebysig
5 specialname instance void set_DocTypeName(string value) }
6
7 [C#]
8 public string DocTypeName { get; set; }
```

8 Summary

9 Gets or sets the name of the document type in a document type
10 declaration.

11 Property Value

12

13 A **System.String** specifying the name of the document type.

14 Description

15 If an attempt is made to set this property to **null**, it is set to
16 **System.String.Empty**.

17

18 [*Note:* A document type declaration is of the following form:

19

```
20 <!DOCTYPE DocTypeName PUBLIC "PublicId" "SystemId"  
21 [InternalSubset]>
```

22

23 This property, along with

24 **System.Xml.XmlParserContext.InternalSubset**,

25 **System.Xml.XmlParserContext.PublicId**, and

26 **System.Xml.XmlParserContext.SystemId** properties, provide all
27 the document type declaration information.]

28

1 XmlParserContext.Encoding Property

```
2 [ILASM]
3 .property class System.Text.Encoding Encoding { public
4 hidebysig specialname instance class System.Text.Encoding
5 get_Encoding() public hidebysig specialname instance void
6 set_Encoding(class System.Text.Encoding value) }
7
8 [C#]
9 public Encoding Encoding { get; set; }
```

9 Summary

10 Gets or sets the encoding type.

11 Property Value

12

13 A **System.Text.Encoding** indicating the encoding type.

14

1 XmlParserContext.InternalSubset

2 Property

```
3 [ILASM]
4 .property string InternalSubset { public hidebysig
5 specialname instance string get_InternalSubset() public
6 hidebysig specialname instance void
7 set_InternalSubset(string value) }

8 [C#]
9 public string InternalSubset { get; set; }
```

10 Summary

11 Gets or sets the internal subset in a document type declaration.

12 Property Value

13

14 A **System.String** specifying the internal subset.

15 Description

16 If an attempt is made to set this property to **null**, it is set to
17 **System.String.Empty**.

18

19 [*Note:* A document type declaration is of the following form:

20

```
21 <!DOCTYPE DocTypeName PUBLIC "PublicId" "SystemId"  
22 [InternalSubset]>
```

23

24 This property, along with

25 **System.Xml.XmlParserContext.DocTypeName**,

26 **System.Xml.XmlParserContext.PublicId**, and

27 **System.Xml.XmlParserContext.SystemId** properties, provide all
28 the document type declaration information.]

29

1 XmlParserContext.NamespaceManager 2 Property

```
3 [ILASM]  
4 .property class System.Xml.XmlNamespaceManager  
5 NamespaceManager { public hidebysig specialname instance  
6 class System.Xml.XmlNamespaceManager get_NamespaceManager()  
7 public hidebysig specialname instance void  
8 set_NamespaceManager(class System.Xml.XmlNamespaceManager  
9 value) }  
10 [C#]  
11 public XmlNamespaceManager NamespaceManager { get; set; }
```

12 Summary

13 Gets or sets the **System.Xml.XmlNamespaceManager** used by the
14 current instance.

15 Property Value

17 The **System.Xml.XmlNamespaceManager** used by the current
18 instance.

19 Description

20 [*Note:* A **System.Xml.XmlNamespaceManager** defines the current
21 namespace scope and provides methods for looking up namespace
22 information.]

23

1 XmlParserContext.NameTable Property

```
2 [ILASM]
3 .property class System.Xml.XmlNameTable NameTable { public
4 hidebysig specialname instance class
5 System.Xml.XmlNameTable get_NameTable() public hidebysig
6 specialname instance void set_NameTable(class
7 System.Xml.XmlNameTable value) }
8
9 [C#]
public XmlNameTable NameTable { get; set; }
```

10 Summary

11 Gets or sets the **System.Xml.XmlNameTable** used by the current
12 instance to look up prefixes and namespace URIs.

13 Property Value

14

15 The **System.Xml.XmlNameTable** used by the current instance.

16

1 XmlParserContext.PublicId Property

```
2 [ILASM]
3 .property string PublicId { public hidebysig specialname
4 instance string get_PublicId() public hidebysig specialname
5 instance void set_PublicId(string value) }
6
7 [C#]
8 public string PublicId { get; set; }
```

8 Summary

9 Gets or sets the public identifier in a document type declaration.

10 Property Value

11

12 A **System.String** specifying the public identifier.

13 Description

14 If an attempt is made to set this property to **null**, it is set to
15 **System.String.Empty**.

16

17 [*Note:* A document type declaration is of the following form:

18

19 <!DOCTYPE DocTypeName PUBLIC "*PublicId*" "*SystemId*"

20 [*InternalSubset*]>

21

22 This property, along with

23 **System.Xml.XmlParserContext.DocTypeName**,

24 **System.Xml.XmlParserContext.InternalSubset**, and

25 **System.Xml.XmlParserContext.SystemId** properties, provide all

26 the document type declaration information.]

27

1 XmlParserContext.SystemId Property

```
2 [ILASM  
3 .property string SystemId { public hidebysig specialname  
4 instance string get_SystemId() public hidebysig specialname  
5 instance void set_SystemId(string value) }  
6  
7 [C#  
8 public string SystemId { get; set; }
```

8 Summary

9 Gets or sets the system identifier in a document type declaration.

10 Property Value

11

12 A **System.String** specifying the system identifier.

13 Description

14 If an attempt is made to set this property to **null**, it is set to
15 **System.String.Empty**.

16

17 [*Note:* A document type declaration is of the following form:

18

```
19 <!DOCTYPE DocTypeName PUBLIC "PublicId" "SystemId"  
20 [InternalSubset]>
```

21

22 This property, along with

23

System.Xml.XmlParserContext.DocTypeName,

24

System.Xml.XmlParserContext.InternalSubset, and

25

System.Xml.XmlParserContext.PublicId properties, provide all the
26 document type declaration information.]

26

27

1 XmlParserContext.XmlLang Property

```
2 [ILASM]
3 .property string XmlLang { public hidebysig specialname
4 instance string get_XmlLang() public hidebysig specialname
5 instance void set_XmlLang(string value) }
6
7 [C#]
8 public string XmlLang { get; set; }
```

8 Summary

9 Gets or sets the current `xml:lang` scope.

10 Property Value

11

12 A **System.String** specifying the current `xml:lang` scope. If there is no
13 `xml:lang` in scope, **System.String.Empty** is returned.

14 Description

15 If an attempt is made to set this property to **null**, it is set to
16 **System.String.Empty**.

17
18 [*Note:* The language attribute, `xml:lang`, specifies the language in
19 which the content and attribute values of the current element are
20 written.

21
22 For details on valid `xml:lang` values, refer to section 2.12 of the W3C
23 Extensible Markup Language (XML) 1.0 recommendation.]

24

1 XmlParserContext.XmlSpace Property

```
2 [ILASM]
3 .property valuetype System.Xml.XmlSpace XmlSpace { public
4 hidebysig specialname instance valuetype
5 System.Xml.XmlSpace get_XmlSpace() public hidebysig
6 specialname instance void set_XmlSpace(valuetype
7 System.Xml.XmlSpace value) }
8
9 [C#]
10 public XmlSpace XmlSpace { get; set; }
```

10 Summary

11 Gets or sets the current `xml:space` scope.

12 Property Value

13

14 One of the members of the **System.Xml.XmlSpace** enumeration
15 indicating the `xml:space` scope.

16 Description

17 [*Note:* The white space attribute, `xml:space`, specifies how white
18 space is handled in the current element.]

19