System.Xml.XmlWriter Class

[ILASM]

```
.class public abstract XmlWriter extends System.Object
```

[C#]

```
public abstract class XmlWriter
```

Assembly Info:

- *Name:* System.Xml
- *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

Represents a writer that provides a non-cached, forward-only means of generating streams or files containing XML data.

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

The output of this class conforms to the W3C Extensible Markup Language (XML) 1.0 and the Namespaces in XML recommendations.

[*Note:* This class is **abstract** and is implemented in the **System.Xml.XmlTextWriter** class.]

XmlWriter() Constructor

```
[ILASM]
family specialname instance void .ctor()

[C#]
protected XmlWriter()
```

Summary

Constructs a new instance of the **System.Xml.XmlWriter** class.

XmlWriter.Close() Method

[ILASM]

```
.method public hidebysig virtual abstract void Close()
```

[C#]

```
public abstract void Close()
```

Summary

Closes the writer.

Behaviors

This method closes any remaining open elements or attributes.

This method calls the **System.Xml.XmlWriter.Flush** method to flush the underlying buffered stream and then closes the stream.

This method sets the **System.Xml.XmlWriter.WriteState** to **System.Xml.WriteState.Closed**.

How and When to Override

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

XmlWriter.Flush() Method

[ILASM]

```
.method public hidebysig virtual abstract void Flush()
```

[C#]

```
public abstract void Flush()
```

Summary

Clears all buffers and causes any buffered data to be written to the underlying stream.

Behaviors

As described above.

How and When to Override

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

XmlWriter.LookupPrefix(System.String) Method

[ILASM]

```
.method public hidebysig virtual abstract string
LookupPrefix(string ns)
```

[C#]

```
public abstract string LookupPrefix(string ns)
```

Summary

Retrieves the prefix defined in the current namespace scope for the specified namespace URI.

Parameters

Parameter	Description
ns	A **System.String** specifying the namespace URI.

Return Value

A **System.String** containing the corresponding prefix, or **System.String.Empty** if the prefix is not found and *ns* is the default namespace, or **null** if no matching namespace URI is found in the current scope.

Behaviors

As described above.

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.ArgumentException	*ns* is **null** or **System.String.Empty**.

XmlWriter.WriteAttributes(System.Xml.XmlReader, System.Boolean) Method

```
[ILASM]
.method public hidebysig virtual void WriteAttributes(class
System.Xml.XmlReader reader, bool defattr)

[C#]
public virtual void WriteAttributes(XmlReader reader, bool
defattr)
```

Summary

Writes the attributes found at the current position of a **System.Xml.XmlReader**.

Parameters

Parameter	Description
reader	A **System.Xml.XmlReader** from which to copy the attributes.
defattr	A **System.Boolean** where **true** specifies to copy the default attributes from *reader*; otherwise, **false**.

Behaviors

If the reader is positioned on an **Element** or **XmlDeclaration** node, this method writes all the contained attributes. If the reader is positioned on an **Attribute** node, this method writes the current attribute, then the rest of the attributes until the closing tag is reached. If the reader is positioned on any other node type, this method throws an exception.

Default

This method positions the reader by calling its **System.Xml.XmlReader.MoveToFirstAttribute** and **System.Xml.XmlReader.MoveToNextAttribute** methods, and retrieves the value of the attributes by calling the **System.Xml.XmlReader.ReadAttributeValue** method of the reader.

How and When to Override

Override this method to remove any content that would invalidate the document.

Usage

Use this method to write all the attributes found at the current position.

Exceptions

Exception	Condition
System.ArgumentNullException	*reader* is **null**.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.
System.Xml.XmlException	*reader* is not positioned on a node of **System.Xml.XmlNodeType Attribute**, **Element**, or **XmlDeclaration**.

XmlWriter.WriteAttributeString(System.String, System.String, System.String) Method

```
[ILASM]
.method public hidebysig instance void
WriteAttributeString(string localName, string ns, string
value)

[C#]
public void WriteAttributeString(string localName, string
ns, string value)
```

Summary

Writes a new attribute with the specified local name, namespace URI, and value.

Parameters

Parameter	Description
localName	A **System.String** specifying the local name of the attribute.
ns	A **System.String** specifying the namespace URI of the attribute.
value	A **System.String** specifying the value of the attribute.

Description

This method calls the following methods in order to write a complete attribute:

System.Xml.XmlWriter.WriteStartAttribute(**null**, *localName*, *ns*)

System.Xml.XmlWriter.WriteString(*value*)

System.Xml.XmlWriter.WriteEndAttribute()

Exceptions

Exception	Condition
System.ArgumentException	*localName* is **null** or **System.String.Empty**.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is not **System.Xml.WriteState.Element**.

XmlWriter.WriteAttributeString(System.String, System.String) Method

```
[ILASM]
.method public hidebysig instance void
WriteAttributeString(string localName, string value)

[C#]
public void WriteAttributeString(string localName, string
value)
```

Summary

Writes a new attribute with the specified local name and value.

Parameters

Parameter	Description
localName	A **System.String** specifying the local name of the attribute.
value	A **System.String** specifying the value of the attribute.

Description

This method calls the following methods in order to write a complete attribute:

System.Xml.XmlWriter.XmlWriter.WriteStartAttribute(**null**, *localName*, **null**)

System.Xml.XmlWriter.WriteString(*value*)

System.Xml.XmlWriter.WriteEndAttribute()

Exceptions

Exception	Condition
System.ArgumentException	*localName* is **null** or **System.String.Empty**.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is not **System.Xml.WriteState.Element**.

XmlWriter.WriteAttributeString(System.String, System.String, System.String, System.String) Method

```
[ILASM]
.method public hidebysig instance void
WriteAttributeString(string prefix, string localName,
string ns, string value)

[C#]
public void WriteAttributeString(string prefix, string
localName, string ns, string value)
```

Summary

Writes a new attribute with the specified prefix, local name, namespace URI, and value.

Parameters

Parameter	Description
prefix	A **System.String** specifying the namespace prefix of the attribute.
localName	A **System.String** specifying the local name of the attribute.
ns	A **System.String** specifying the namespace URI of the attribute.
value	A **System.String** specifying the value of the attribute.

Description

This method calls the following methods in order to write a complete attribute:

System.Xml.XmlWriter.WriteStartAttribute(*prefix*, *localName*, *ns*)

System.Xml.XmlWriter.WriteString(*value*)

System.Xml.XmlWriter.WriteEndAttribute()

Exceptions

Exception	Condition
System.ArgumentException	*localName* is **null** or **System.String.Empty**.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is not **System.Xml.WriteState.Element**.

XmlWriter.WriteBase64(System.Byte[], System.Int32, System.Int32) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteBase64(class System.Byte[] buffer, int32 index, int32
count)

[C#]
public abstract void WriteBase64(byte[] buffer, int index,
int count)
```

Summary

Encodes the specified binary bytes as Base64 and writes out the resulting text.

Parameters

Parameter	Description
buffer	A **System.Byte** array containing the bytes to encode.
index	A **System.Int32** specifying the position within the array of the first byte to encode.
count	A **System.Int32** specifying the number of bytes to encode.

Description

[*Note:* Base64 encoding represents byte sequences in a text form comprised of the 65 US-ASCII characters (A-Z, a-z, 0-9, +, /,=) where each character encodes 6 bits of the binary data. For more information on Base64 encoding, see RFC 2045 (http://www.ietf.org/rfc/rfc2045).]

Behaviors

As described above.

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	*buffer* is **null**.
System.ArgumentException	The buffer length minus *index* is less than *count*.
System.ArgumentOutOfRangeException	*index* or *count* is less than zero.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteBinHex(System.Byte[], System.Int32, System.Int32) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteBinHex(class System.Byte[] buffer, int32 index, int32
count)

[C#]
public abstract void WriteBinHex(byte[] buffer, int index,
int count)
```

Summary

Encodes the specified binary bytes as BinHex and writes the resulting text.

Parameters

Parameter	Description
buffer	A **System.Byte** array containing the bytes to encode.
index	A **System.Int32** specifying the position within the array of the first byte to encode.
count	A **System.Int32** specifying the number of bytes to encode.

Description

[*Note:* For information on BinHex encoding, see RFC 1741 (http://www.ietf.org/rfc/rfc1741).]

Behaviors

As described above.

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	*buffer* is **null**.
System.ArgumentException	The buffer length minus *index* is less

	than *count*.
System.ArgumentOutOfRangeException	*index* or *count* is less than zero.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteCData(System.String) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteCData(string text)

[C#]
public abstract void WriteCData(string text)
```

Summary

Writes out a CDATA block containing the specified text.

Parameters

Parameter	Description
text	A **System.String** specifying the text to place inside the CDATA block.

Behaviors

This method writes `<![CDATA[` *text*`]]>`.

If *text* is **null** or **System.String.Empty**, this method writes an empty CDATA block, `<![CDATA[]]>`.

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.ArgumentException	The text would result in a non-well formed XML document.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteCharEntity(System.Char) Method

[ILASM]

```
.method public hidebysig virtual abstract void
WriteCharEntity(valuetype System.Char ch)
```

[C#]

```
public abstract void WriteCharEntity(char ch)
```

Summary

Forces the generation of a character entity for the specified Unicode character value.

Parameters

Parameter	Description
ch	The **System.Char** for which to generate the entity.

Behaviors

This method forces the generation of a character entity for the specified Unicode character value and writes the Unicode character in hexadecimal character entity reference format.

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.ArgumentException	The character is in the surrogate pair character range, 0xd800 - 0xdfff, or the text would result in a non-well formed XML document.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteChars(System.Char[], System.Int32, System.Int32) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteChars(class System.Char[] buffer, int32 index, int32
count)

[C#]
public abstract void WriteChars(char[] buffer, int index,
int count)
```

Summary

Writes text a buffer at a time.

Parameters

Parameter	Description
buffer	A **System.Char** array containing the text to write.
index	A **System.Int32** specifying the position within the array of the start of the text to write.
count	A **System.Int32** specifying the number of characters to write.

Behaviors

As described above.

How and When to Override

When overriding this method, throw an exception if surrogate pair characters would be split across multiple buffer writes. This exception must be caught in order to continue writing the next surrogate pair characters. The XML specification defines the valid ranges for surrogate pairs.

Exceptions

Exception	Condition
System.ArgumentNullException	*buffer* is **null**.
System.ArgumentException	The buffer length minus *index* is less than *count*.

System.ArgumentOutOfRangeException	*index* or *count* is less than zero.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteComment(System.String) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteComment(string text)

[C#]
public abstract void WriteComment(string text)
```

Summary

Writes a comment containing the specified text.

Parameters

Parameter	Description
text	A **System.String** containing the text to place inside the comment.

Behaviors

This method writes <!-- *text* -->.

If *text* is **null** or **System.String.Empty**, this method writes a comment with no content, <!---->.

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.ArgumentException	The text would result in a non-well formed XML document.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteDocType(System.String, System.String, System.String, System.String) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteDocType(string name, string pubid, string sysid,
string subset)

[C#]
public abstract void WriteDocType(string name, string
pubid, string sysid, string subset)
```

Summary

Writes the document type declaration with the specified name and optional attributes.

Parameters

Parameter	Description
name	A **System.String** specifying the name of the document type.
pubid	A **System.String** specifying the public identifier, which is an alternative to the system identifier.
sysid	A **System.String** specifying the system identifier, which is the URI of the DTD (document type definition) for the document.
subset	A **System.String** specifying a URI that contains markup declarations.

Behaviors

This method writes the document type declaration with the specified name and optional attributes. The optional attributes, *pubid*, *sysid*, and *subset*, are not checked for invalid characters.

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.ArgumentException	*name* is **null** or **System.String.Empty**. -or- The value for *name* would result in invalid XML.
System.InvalidOperationException	This method was called outside the prolog (after the root element).

XmlWriter.WriteElementString(System.String, System.String) Method

```
[ILASM]
.method public hidebysig instance void
WriteElementString(string localName, string value)

[C#]
public void WriteElementString(string localName, string
value)
```

Summary

Writes an element with the specified local name and value.

Parameters

Parameter	Description
localName	A **System.String** specifying the local name of the element.
value	A **System.String** specifying the value of the element.

Description

This method is equivalent to **System.Xml.XmlWriter.WriteElementString**(*localName*, **null**, *value*).

Exceptions

Exception	Condition
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteElementString(System.String, System.String, System.String) Method

```
[ILASM]
.method public hidebysig instance void
WriteElementString(string localName, string ns, string
value)

[C#]
public void WriteElementString(string localName, string ns,
string value)
```

Summary

Writes an element with the specified local name, namespace URI, and value.

Parameters

Parameter	Description
localName	A **System.String** specifying the local name of the element.
ns	A **System.String** specifying the namespace URI to associate with the element.
value	A **System.String** specifying the value of the element.

Description

This method calls the following methods to write a complete element:

System.Xml.XmlWriter.WriteStartElement(*localName*, *ns*)

System.Xml.XmlWriter.WriteString(*value*) - this method is not called if *value* is either **null** or **System.String.Empty**

System.Xml.XmlWriter.WriteEndElement()

If any of the input parameters are **null** or **System.String.Empty**, the element is written with that parameter missing.

Exceptions

Exception	**Condition**
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteEndAttribute() Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteEndAttribute()

[C#]
public abstract void WriteEndAttribute()
```

Summary

Closes the attribute started with the **System.Xml.XmlWriter.WriteStartAttribute** method.

Behaviors

As described above.

How and When to Override

Override the **System.Xml.XmlWriter.WriteStartAttribute**, **System.Xml.XmlWriter.WriteAttributeString**, and **System.Xml.XmlWriter.WriteEndElement** methods so these methods also close any attributes that are open when they are called.

Exceptions

Exception	Condition
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteEndDocument() Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteEndDocument()

[C#]
public abstract void WriteEndDocument()
```

Summary

Closes open elements and attributes and sets the **System.Xml.XmlWriter.WriteState** back to the **System.Xml.WriteState.Start** state.

Behaviors

This method closes all elements and attributes created by the **System.Xml.XmlWriter.WriteStartElement** and **System.Xml.XmlWriter.WriteStartAttribute** methods, respectively, that are open when the **System.Xml.XmlWriter.WriteEndDocument** method is called.

How and When to Override

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

Usage

After calling this method, the current instance can be used to write a new XML document.

Exceptions

Exception	Condition
System.ArgumentException	The current instance is in the wrong **System.Xml.WriteState**, or the document does not have a root element.

XmlWriter.WriteEndElement() Method

[ILASM]

```
.method public hidebysig virtual abstract void
WriteEndElement()
```

[C#]

```
public abstract void WriteEndElement()
```

Summary

Closes an open element and pops the corresponding namespace scope.

Behaviors

This method closes one element and pops the corresponding namespace scope. If the open element does not contain content, it is closed as an empty element using " />"; otherwise an end element is written.

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.InvalidOperationException	No element was open, or the **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteEntityRef(System.String) Method

[ILASM]

```
.method public hidebysig virtual abstract void
WriteEntityRef(string name)
```

[C#]

```
public abstract void WriteEntityRef(string name)
```

Summary

Writes an entity reference with the specified name.

Parameters

Parameter	Description
name	A **System.String** specifying the name of the entity reference.

Behaviors

This method writes % *name*;.

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.ArgumentException	*name* is either **null** or **System.String.Empty**.

XmlWriter.WriteFullEndElement() Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteFullEndElement()

[C#]
public abstract void WriteFullEndElement()
```

Summary

Closes an open element and pops the corresponding namespace scope.

Behaviors

This method closes one element and pops the corresponding namespace scope. This method writes an end element regardless of whether there is any content in the element.

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.InvalidOperationException	No element was open, or the **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteName(System.String) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteName(string name)

[C#]
public abstract void WriteName(string name)
```

Summary

Writes the specified name, ensuring it is a valid name according to the W3C XML 1.0 recommendation (http://www.w3.org/TR/1998/REC-xml-19980210#NT-Name).

Parameters

Parameter	Description
name	A **System.String** specifying the name to write.

Behaviors

This method writes the specified name, ensuring it is a valid name according to the W3C XML 1.0 recommendation.

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.ArgumentException	*name* is **null** or **System.String.Empty**; or *name* is not a valid XML Name.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteNmToken(System.String) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteNmToken(string name)

[C#]
public abstract void WriteNmToken(string name)
```

Summary

Writes the specified name, ensuring it is a valid name token (Nmtoken) according to the W3C XML 1.0 recommendation (http://www.w3.org/TR/1998/REC-xml-19980210#NT-Name).

Parameters

Parameter	Description
name	A **System.String** specifying the name to write.

Behaviors

As described above.

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.ArgumentException	*name* is **null** or **System.String.Empty**; or *name* is not a valid XML Nmtoken.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteNode(System.Xml.XmlReader, System.Boolean) Method

```
[ILASM]
.method public hidebysig virtual void WriteNode(class
System.Xml.XmlReader reader, bool defattr)

[C#]
public virtual void WriteNode(XmlReader reader, bool
defattr)
```

Summary

Writes the node found at the current position of the specified **System.Xml.XmlReader**, and all sub-nodes.

Parameters

Parameter	**Description**
reader	A **System.Xml.XmlReader** from which to copy the attributes.
defattr	A **System.Boolean** where **true** specifies to copy the default attributes from *reader*; otherwise, **false**.

Behaviors

After the nodes are read, *reader* is moved to the next node at the same depth as the current node.

Default

If *reader* is in the **System.Xml.ReadState.Initial** **System.Xml.XmlReader.ReadState**, this method moves *reader* to the end of the stream. If *reader* is in the **System.Xml.ReadState.EndOfFile** **System.Xml.XmlReader.ReadState**, this method is non-operational.

How and When to Override

Override this method to customize the behavior of this method in types derived from the **System.Xml.XmlWriter** class.

Usage

Use this method to write the node found at the current position and all sub-nodes.

Exceptions

Exception	**Condition**
System.ArgumentNullException	*reader* is **null**.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

Example

The following example uses a **System.Xml.XmlTextReader** and a **System.Xml.XmlTextWriter** to copy an XML file, specified in the command line, to the console.

[C#]

```
using System;
using System.Xml;

public class Copier {

  public static void Main(string[] args) {

    XmlTextReader xtReader = new XmlTextReader(args[0]);
    XmlTextWriter xtWriter =
      new XmlTextWriter(Console.Out);
    xtWriter.WriteNode(xtReader, false);
    xtWriter.Close();
    xtReader.Close();
  }
}
```

XmlWriter.WriteProcessingInstruction(System.String, System.String) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteProcessingInstruction(string name, string text)

[C#]
public abstract void WriteProcessingInstruction(string
name, string text)
```

Summary

Writes a processing instruction with the specified name and text.

Parameters

Parameter	Description
name	A **System.String** specifying the name of the processing instruction.
text	A **System.String** specifying the text to include in the processing instruction.

Behaviors

This method writes <? *name text* ?>.

If *text* is **null** or **System.String.Empty**, this method writes a processing instruction with no text content, <? *name* ?>.

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.ArgumentException	The text would result in a non-well formed XML document. - or -

	name is **null** or **System.String.Empty**. - or - This method is being used to create an XML declaration after **System.Xml.XmlWriter.WriteStartDocument** has already been called.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteQualifiedName(System.String, System.String) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteQualifiedName(string localName, string ns)

[C#]
public abstract void WriteQualifiedName(string localName,
string ns)
```

Summary

Writes the qualified name.

Parameters

Parameter	Description
localName	A **System.String** specifying the local name to write.
ns	A **System.String** specifying the namespace URI to associate with *localname*.

Behaviors

This method writes the qualified name. If *ns* maps to the current default namespace, no prefix is generated. When writing attribute values, this method generates a prefix if *ns* is not found. When writing element content, it throws an exception if *ns* is not found.

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.ArgumentException	*localName* is **null** or **System.String.Empty**.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.
System.Xml.XmlException	*localName* is not a valid XML name.

XmlWriter.WriteRaw(System.String) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteRaw(string data)

[C#]
public abstract void WriteRaw(string data)
```

Summary

Writes raw text from a string.

Parameters

Parameter	Description
data	A **System.String** specifying the text to write.

Behaviors

This method writes raw text from a string. This method does not encode any characters. If *data* is **null**, **System.String.Empty** is written.

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.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteRaw(System.Char[], System.Int32, System.Int32) Method

[ILASM]

```
.method public hidebysig virtual abstract void
WriteRaw(class System.Char[] buffer, int32 index, int32
count)
```

[C#]

```
public abstract void WriteRaw(char[] buffer, int index, int
count)
```

Summary

Writes raw text from a character array.

Parameters

Parameter	Description
buffer	A **System.Char** array containing the text to write.
index	A **System.Int32** specifying the position within the array of the start of the text to write.
count	A **System.Int32** specifying the number of characters to write.

Behaviors

This method writes raw text from a character array. This method does not encode any characters.

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	*buffer* is **null**.
System.ArgumentException	The buffer length minus *index* is less than *count*.
System.ArgumentOutOfRangeException	*index* or *count* is less than zero.

System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteStartAttribute(System.String, System.String, System.String) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteStartAttribute(string prefix, string localName, string
ns)

[C#]
public abstract void WriteStartAttribute(string prefix,
string localName, string ns)
```

Summary

Writes the start of an attribute with the specified prefix and name, and associates the prefix with the specified namespace URI.

Parameters

Parameter	Description
prefix	A **System.String** specifying the namespace prefix of the attribute.
localName	A **System.String** specifying the local name of the attribute.
ns	A **System.String** specifying the namespace URI associated with the attribute.

Behaviors

If any of the input parameters are **null** or **System.String.Empty**, the start attribute is written with that parameter missing.

How and When to Override

When overriding this method, close any open attributes before writing the new attribute.

Exceptions

Exception	Condition
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteStartAttribute(System.String, System.String) Method

```
[ILASM]
.method public hidebysig instance void
WriteStartAttribute(string localName, string ns)

[C#]
public void WriteStartAttribute(string localName, string
ns)
```

Summary

Writes the start of an attribute.

Parameters

Parameter	Description
localName	A **System.String** specifying the local name of the attribute.
ns	A **System.String** specifying the namespace URI of the attribute.

Description

This method calls **System.Xml.XmlWriter.WriteStartAttribute**(**null**, *localName*, *ns*).

Exceptions

Exception	Condition
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteStartDocument(System.Boolean) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteStartDocument(bool standalone)

[C#]
public abstract void WriteStartDocument(bool standalone)
```

Summary

Writes the XML declaration with the version "1.0" and the standalone attribute.

Parameters

Parameter	Description
standalone	A **System.Boolean** where **true** indicates to write "standalone= yes"; **false** indicates to write "standalone=no".

Description

[*Note:* Character encoding is set when the class in instantiated.]

Behaviors

As described above.

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.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is not **System.Xml.WriteState.Start**.

XmlWriter.WriteStartDocument() Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteStartDocument()

[C#]
public abstract void WriteStartDocument()
```

Summary

Writes the XML declaration with the version "1.0" and no standalone attribute.

Description

[*Note:* Character encoding is set when the class in instantiated.]

Behaviors

As described above.

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.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is not **System.Xml.WriteState.Start**.

XmlWriter.WriteStartElement(System.String, System.String, System.String) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteStartElement(string prefix, string localName, string
ns)

[C#]
public abstract void WriteStartElement(string prefix,
string localName, string ns)
```

Summary

Writes a start element with the specified name, and associates it with the given namespace and prefix.

Parameters

Parameter	Description
prefix	A **System.String** specifying the namespace prefix of the element.
localName	A **System.String** specifying the local name of the element.
ns	A **System.String** specifying the namespace URI to associate with the element.

Behaviors

This method writes a start element and name, and associates it with a namespace and prefix. If the namespace is already in scope and has an associated prefix, that prefix is automatically written also.

If any of the input parameters are **null** or **System.String.Empty**, the start element is written with that parameter missing.

How and When to Override

When overriding this method, also override the **System.Xml.XmlWriter.WriteEndElement**, **System.Xml.XmlWriter.WriteString**, and **System.Xml.XmlWriter.WriteRaw** methods so they close any open start element.

Usage

Use this method to write a specified start element and name, and associate it with a given namespace and prefix. Write any attributes using the **System.Xml.XmlWriter.WriteStartAttribute**, **System.Xml.XmlWriter.WriteString**, and **System.Xml.XmlWriter.WriteEndAttribute** methods, then close the element using the **System.Xml.XmlWriter.WriteEndElement** method.

Exceptions

Exception	Condition
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

Example

See **System.Xml.XmlTextWriter.WriteStartElement** for an example demonstrating this method.

XmlWriter.WriteStartElement(System.String, System.String) Method

[ILASM]

```
.method public hidebysig instance void
WriteStartElement(string localName, string ns)
```

[C#]

```
public void WriteStartElement(string localName, string ns)
```

Summary

Writes a start element with the specified name, and associates it with the given namespace.

Parameters

Parameter	Description
localName	A **System.String** specifying the local name of the element.
ns	A **System.String** specifying the namespace URI to associate with the element.

Description

This method calls **System.Xml.XmlWriter.WriteStartElement**(**null**, *localName*, *ns*).

Exceptions

Exception	Condition
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteStartElement(System.String) Method

[ILASM]

```
.method public hidebysig instance void
WriteStartElement(string localName)
```

[C#]

```
public void WriteStartElement(string localName)
```

Summary

Writes a start element with the specified name.

Parameters

Parameter	Description
localName	A **System.String** specifying the local name of the element.

Description

This method calls **System.Xml.XmlWriter.WriteStartElement**(**null**, *localName*, **null**).

Exceptions

Exception	Condition
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteString(System.String) Method

[ILASM]

```
.method public hidebysig virtual abstract void
WriteString(string text)
```

[C#]

```
public abstract void WriteString(string text)
```

Summary

Writes the specified text.

Parameters

Parameter	Description
text	A **System.String** specifying the text to write.

Behaviors

This method performs the following conversions before writing the text:

- The characters '&', '<', and '>' are replaced with "&amp;", "&lt;", and "&gt;", respectively.
- Character values in the range 0x-0x1F (excluding the white space characters 0x9, 0x10, and 0x13) are replaced with numeric character entities ("&#0;" through "&#0x1F").
- If called in the context of an attribute value, double and single quotes are replaced with "&quot;" and "&apos;" respectively.

If *text* is **null** or **System.String.Empty**, this method writes a text node with no data content.

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.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteSurrogateCharEntity(System.Char, System.Char) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteSurrogateCharEntity(valuetype System.Char lowChar,
valuetype System.Char highChar)

[C#]
public abstract void WriteSurrogateCharEntity(char lowChar,
char highChar)
```

Summary

Generates and writes the surrogate character entity for the surrogate character pair.

Parameters

Parameter	Description
lowChar	A **System.Char** containing the low surrogate. This must be a value between 0xDC00 and 0xDFFF.
highChar	A **System.Char** containing the high surrogate. This must be a value between 0xD800 and 0xDBFF.

Description

[*Note:* Applications encode DOM strings using UTF-16. For both HTML and XML, the document character set (and therefore the notation of numeric character references) is based on UCS [ISO-10646]. A single numeric character reference in a source document may therefore in some cases correspond to two 16-bit units in a DOM string (a high surrogate and a low surrogate). These 16-bit units are referred to as a surrogate pair.

For more information regarding surrogates or characters, refer to section 3.7 of the Unicode 3.0/Unicode 2.0 standard located at http://www.unicode.org, or section 2.2 of the W3C XML 1.0 Recommendation located at http://www.w3.org/TR/REC-xml#charsets.]

Behaviors

This method generates and writes the surrogate character entity for a surrogate character pair. The surrogate character entity is written in hexadecimal format. The range for surrogate characters is #x10000 to #x10FFFF. The following formula is used to generate the surrogate character entity: (*highChar* `- 0xD800) * 0x400 + (`*lowChar* `- 0xDC00) + 0x10000`.

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.ArgumentException	An invalid surrogate character pair was passed.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteWhitespace(System.String) Method

```
[ILASM]
.method public hidebysig virtual abstract void
WriteWhitespace(string ws)

[C#]
public abstract void WriteWhitespace(string ws)
```

Summary

Writes the given white space.

Parameters

Parameter	Description
ws	A **System.String** containing the white space characters.

Behaviors

As described above.

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.ArgumentException	*ws* is **null** or **System.String.Empty** or contains non-white space characters.
System.InvalidOperationException	The **System.Xml.XmlWriter.WriteState** is **System.Xml.WriteState.Closed**.

XmlWriter.WriteState Property

[ILASM]

```
.property valuetype System.Xml.WriteState WriteState {
public hidebysig virtual abstract specialname valuetype
System.Xml.WriteState get_WriteState() }
```

[C#]

```
public abstract WriteState WriteState { get; }
```

Summary

Gets the write state of the writer.

Property Value

One of the members of the **System.Xml.WriteState** enumeration.

Behaviors

As described above.

This property is read-only.

How and When to Override

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

Usage

Use this property to query the current state, for example, whether the writer is newly initialized, writing specific XML structures, or closed.

XmlWriter.XmlLang Property

```
[ILASM]
.property string XmlLang { public hidebysig virtual
abstract specialname string get_XmlLang() }

[C#]
public abstract string XmlLang { get; }
```

Summary

Gets the language attribute, `xml:lang`, specifying the language in which the content and attribute values of the current element are written.

Property Value

A **System.String** containing the language attribute, or **null** if the language attribute is not specified for the element.

Behaviors

As described above.

This property is read-only.

How and When to Override

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

XmlWriter.XmlSpace Property

[ILASM]
```
.property valuetype System.Xml.XmlSpace XmlSpace { public
hidebysig virtual abstract specialname valuetype
System.Xml.XmlSpace get_XmlSpace() }
```

[C#]
```
public abstract XmlSpace XmlSpace { get; }
```

Summary

Gets the white space attribute, `xml:space`, specifying how white space is handled in the current element.

Property Value

One of the members of the **System.Xml.XmlSpace** enumeration, or **System.Xml.XmlSpace.None** if the white space attribute is not specified for the element.

Behaviors

As described above.

This property is read-only.

How and When to Override

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