System.IO.StreamWriter Class

```
[ILASM]
.class public serializable StreamWriter extends
System.IO.TextWriter

[C#]
public class StreamWriter: TextWriter
```

Assembly Info:

- *Name:* mscorlib
- *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)

Implements:

- **System.IDisposable**

Summary

Implements a **System.IO.Stream** wrapper that writes characters to a stream in a particular encoding.

Inherits From: System.IO.TextWriter

Library: BCL

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 **System.IO.StreamWriter** class is designed for character output in a particular **System.Text.Encoding**, whereas subclasses of **System.IO.Stream** are designed for byte input and output.

System.IO.StreamWriter defaults to using an instance of **System.Text.UTF8Encoding** unless specified otherwise. This instance of **System.Text.UTF8Encoding** is constructed such that the **System.Text.Encoding.GetPreamble** method returns the Unicode byte order mark written in UTF-8. The preamble of the encoding is added to a stream when you are not appending to an existing stream. This means any text file you create with **System.IO.StreamWriter** has three byte order marks at its beginning. UTF-8 handles all Unicode characters correctly and gives consistent results on localized versions of the operating system.

[*Note:* By default, **System.IO.StreamWriter** is not thread safe. For a thread-safe wrapper, see **System.IO.TextWriter.Synchronized**.]

StreamWriter(System.IO.Stream) Constructor

[ILASM]
```
public rtspecialname specialname instance void .ctor(class
System.IO.Stream stream)
```

[C#]
```
public StreamWriter(Stream stream)
```

Summary

Constructs and initializes a new instance of the **System.IO.StreamWriter** class for the specified stream, using the default encoding and buffer size.

Parameters

Parameter	Description
stream	The **System.IO.Stream** to write to.

Description

This constructor initializes the **System.IO.StreamWriter.Encoding** property to a **System.Text.UTF8Encoding** whose **System.Text.Encoding.GetPreamble** method returns an empty byte array. For additional information, see **System.IO.TextWriter.Encoding**. The **System.IO.StreamWriter.BaseStream** property is initialized using *stream*.

[*Note:* The default buffer size may typically be around 4 KB.]

Exceptions

Exception	Condition
System.ArgumentException	*stream* does not support writing.
System.ArgumentNullException	*stream* is **null**.

StreamWriter(System.IO.Stream, System.Text.Encoding) Constructor

[ILASM]

```
public rtspecialname specialname instance void .ctor(class
System.IO.Stream stream, class System.Text.Encoding
encoding)
```

[C#]

```
public StreamWriter(Stream stream, Encoding encoding)
```

Summary

Constructs and initializes a new instance of the **System.IO.StreamWriter** class for the specified stream, using the specified encoding and the default buffer size.

Parameters

Parameter	Description
stream	The **System.IO.Stream** to write to.
encoding	A **System.Text.Encoding** that specifies the character encoding to use.

Description

This constructor initializes the **System.IO.StreamWriter.Encoding** property using *encoding*, and the **System.IO.StreamWriter.BaseStream** property using *stream*. For additional information, see **System.IO.TextWriter.Encoding**.

[*Note:* The default buffer size may typically be around 4 KB.]

Exceptions

Exception	Condition
System.ArgumentNullException	*stream* or *encoding* is **null**.
System.ArgumentException	*stream* does not support writing.

StreamWriter(System.IO.Stream, System.Text.Encoding, System.Int32) Constructor

```
[ILASM]
public rtspecialname specialname instance void .ctor(class
System.IO.Stream stream, class System.Text.Encoding
encoding, int32 bufferSize)

[C#]
public StreamWriter(Stream stream, Encoding encoding, int
bufferSize)
```

Summary

Constructs and initializes a new instance of the **System.IO.StreamWriter** class for the specified stream, using the specified encoding and buffer size.

Parameters

Parameter	Description
stream	The **System.IO.Stream** to write to.
encoding	A **System.Text.Encoding** that specifies the character encoding to use.
bufferSize	A **System.Int32** that specifies the buffer size.

Description

This constructor initializes the **System.IO.StreamWriter.Encoding** property using *encoding*, and the **System.IO.StreamWriter.BaseStream** property using *stream*. For additional information, see **System.IO.TextWriter.Encoding**.

Exceptions

Exception	Condition
System.ArgumentNullException	*stream* or *encoding* is **null**.
System.ArgumentOutOfRangeException	*bufferSize* is negative.
System.ArgumentException	*stream* does not support writing.

StreamWriter(System.String) Constructor

```
[ILASM]
public rtspecialname specialname instance void .ctor(string
path)

[C#]
public StreamWriter(string path)
```

Summary

Constructs and initializes a new instance of the **System.IO.StreamWriter** class for the specified file on the specified path, using the default encoding and buffer size.

Parameters

Parameter	Description
path	A **System.String** that specifies the complete file path to write to.

Description

This constructor initializes the **System.IO.StreamWriter.Encoding** property to a **System.Text.UTF8Encoding** whose **System.Text.Encoding.GetPreamble** method returns an empty byte array. For additional information, see **System.IO.TextWriter.Encoding**.

[*Note: path* is not required to be a file stored on disk; it can be any part of a system that supports access via streams. For example, depending on the system, this class may be able to access a physical device.

For information on the valid format and characters for path strings, see **System.IO.Path**.

The default buffer size may typically be around 4 KB.]

Exceptions

Exception	Condition
System.IO.IOException	*path* is in an invalid format or contains invalid characters.
System.IO.DirectoryNotFoundException	The directory information specified in *path* was not found.

System.UnauthorizedAccessException	Access to *path* is denied.
System.ArgumentException	*path* is an empty string ("").
System.ArgumentNullException	*path* is **null**.
System.Security.SecurityException	The caller does not have the required permission.

Permissions

Permission	**Description**
System.Security.Permissions.FileIOPermission	Requires permission for reading and writing files. See **System.Security.Permissions.FileIOPermissionAcces** **Read**, **System.Security.Permissions.FileIOPermissionAcces** **Write**

StreamWriter(System.String, System.Boolean) Constructor

```
[ILASM]
public rtspecialname specialname instance void .ctor(string
path, bool append)

[C#]
public StreamWriter(string path, bool append)
```

Summary

Constructs and initializes a new instance of the **System.IO.StreamWriter** class for the specified file on the specified path, using the default encoding and buffer size.

Parameters

Parameter	Description
path	A **System.String** that specifies the complete file path to write to.
append	A **System.Boolean** value that determines whether data is to be appended to the file. If the file exists and *append* is **false**, the file is overwritten. If the file exists and *append* is **true**, the data is appended to the file. Otherwise, a new file is created.

Description

This constructor initializes the **System.IO.StreamWriter.Encoding** property to **System.Text.UTF8Encoding** whose **System.Text.Encoding.GetPreamble** method returns an empty byte array. For additional information, see **System.IO.TextWriter.Encoding**.

If the specified file exists, it can be either overwritten or appended to. If the file does not exist, this constructor creates a new file.

[*Note: path* is not required to be a file stored on disk; it can be any part of a system that supports access via streams. For example, depending on the system, this class may be able to access a physical device.

For information on the valid format and characters for path strings, see **System.IO.Path**.

The default buffer size may typically be around 4 KB.]

Exceptions

Exception	Condition
System.IO.IOException	*path* is in an invalid format or contains invalid characters.
System.IO.DirectoryNotFoundException	The directory information specified in *path* was not found.
System.UnauthorizedAccessException	Access to *path* is denied.
System.ArgumentException	*path* is an empty string ("").
System.ArgumentNullException	*path* is **null**.
System.Security.SecurityException	The caller does not have the required permission.

Permissions

Permission	Description
System.Security.Permissions.FileIOPermission	Requires permission for reading and writing files. See **System.Security.Permissions.FileIOPermissionAcces** **Read**, **System.Security.Permissions.FileIOPermissionAcces** **Write**

StreamWriter(System.String, System.Boolean, System.Text.Encoding) Constructor

```
[ILASM]
public rtspecialname specialname instance void .ctor(string
path, bool append, class System.Text.Encoding encoding)

[C#]
public StreamWriter(string path, bool append, Encoding
encoding)
```

Summary

Constructs and initializes a new instance of the **System.IO.StreamWriter** class for the specified file on the specified path, using the specified encoding and default buffer size.

Parameters

Parameter	Description
path	A **System.String** that specifies the complete file path to write to.
append	A **System.Boolean** value that determines whether data is to be appended to the file. If the file exists and *append* is **false**, the file is overwritten. If the file exists and *append* is **true**, the data is appended to the file. Otherwise, a new file is created.
encoding	A **System.Text.Encoding** that specifies the character encoding to use.

Description

If the specified file exists, it can be either overwritten or appended to. If the file does not exist, this constructor creates a new file.

This constructor initializes the **System.IO.StreamWriter.Encoding** property using *encoding*. For additional information, see **System.IO.TextWriter.Encoding**.

[*Note: path* is not required to be a file stored on disk; it can be any part of a system that supports access via streams. For example, depending on the system, this class may be able to access a physical device.

For information on the valid format and characters for path strings, see **System.IO.Path**.

The default buffer size may typically be around 4 KB.]

Exceptions

Exception	Condition
System.IO.IOException	*path* is in an invalid format or contains invalid characters.
System.IO.DirectoryNotFoundException	The directory information specified in *path* was not found.
System.UnauthorizedAccessException	Access to *path* is denied.
System.ArgumentException	*path* is an empty string ("").
System.ArgumentNullException	*path* is **null**.
System.Security.SecurityException	The caller does not have the required permission.

Permissions

Permission	Description
System.Security.Permissions.FileIOPermission	Requires permission for reading and writing files. See **System.Security.Permissions.FileIOPermissionAcces** **Read**, **System.Security.Permissions.FileIOPermissionAcces** **Write**

StreamWriter(System.String, System.Boolean, System.Text.Encoding, System.Int32) Constructor

[ILASM]

```
public rtspecialname specialname instance void .ctor(string
path, bool append, class System.Text.Encoding encoding,
int32 bufferSize)
```

[C#]

```
public StreamWriter(string path, bool append, Encoding
encoding, int bufferSize)
```

Summary

Constructs and initializes a new instance of the **System.IO.StreamWriter** class for the specified file on the specified path, using the specified encoding and buffer size.

Parameters

Parameter	Description
path	A **System.String** that specifies the complete file path to write to.
append	A **System.Boolean** value that determines whether data is to be appended to the file. If the file exists and *append* is **false**, the file is overwritten. If the file exists and *append* is **true**, the data is appended to the file. Otherwise, a new file is created.
encoding	A **System.Text.Encoding** that specifies the character encoding to use.
bufferSize	A **System.Int32** that specifies the buffer size.

Description

If the specified file exists, it can be either overwritten or appended to. If the file does not exist, this constructor creates a new file.

This constructor initializes the **System.IO.StreamWriter.Encoding** property using *encoding*. For additional information, see **System.IO.TextWriter.Encoding**.

[*Note: path* is not required to be a file stored on disk; it can be any part of a system that supports access via streams. For example, depending on the system, this class may be able to access a physical device.

For information on the valid format and characters for path strings, see **System.IO.Path**.]

Exceptions

Exception	Condition
System.IO.IOException	*path* is in an invalid format or contains invalid characters.
System.IO.DirectoryNotFoundException	The directory information specified in *path* was not found.
System.ArgumentException	*path* is an empty string ("").
System.ArgumentNullException	*path* or *encoding* is **null**.
System.ArgumentOutOfRangeException	*bufferSize* is negative.
System.Security.SecurityException	The caller does not have the required permission.
System.UnauthorizedAccessException	Access to *path* is denied.

Permissions

Permission	Description
System.Security.Permissions.FileIOPermission	Requires permission for reading and writing files. See **System.Security.Permissions.FileIOPermissionAccess Read**, **System.Security.Permissions.FileIOPermissionAccess Write**

StreamWriter.Close() Method

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

[C#]
```
public override void Close()
```

Summary

Closes the current **System.IO.StreamWriter** and the underlying stream.

Description

This method calls **System.IO.StreamWriter.Flush**, writing buffered data to the underlying stream. Following a call to **System.IO.StreamWriter.Close**, any operations on the current instance might raise exceptions.

[*Note:* This version of **System.IO.StreamWriter.Close** is equivalent to **System.IO.StreamWriter.Dispose**(**true**).

This method overrides **System.IO.Stream.Close**.]

StreamWriter.Dispose(System.Boolean) Method

[ILASM]

```
.method family hidebysig virtual void Dispose(bool disposing)
```

[C#]

```
protected override void Dispose(bool disposing)
```

Summary

Releases the unmanaged resources used by the **System.IO.StreamWriter** and optionally releases the managed resources.

Parameters

Parameter	Description
disposing	**true** to release both managed and unmanaged resources; **false** to release only unmanaged resources.

Description

When the *disposing* parameter is **true**, this method releases all resources held by any managed objects that this **System.IO.StreamWriter** references. This method invokes the **Dispose()** method of each referenced object.

[*Note:* **System.IO.StreamWriter.Dispose** may be called multiple times by other objects. When overriding **System.IO.StreamWriter.Dispose**(**System.Boolean**), be careful not to reference objects that have been previously disposed in an earlier call to **System.IO.StreamWriter.Dispose**.

This method calls the dispose method of the base class, **System.IO.TextWriter.Dispose (***disposing*).]

StreamWriter.Finalize() Method

[ILASM]
```
.method family hidebysig virtual void Finalize()
```

[C#]
```
~StreamWriter()
```

Summary

Releases resources held by the current instance.

Description

[*Note:* Application code does not call this method; it is automatically invoked by during garbage collection unless finalization by the garbage collector has been disabled. For more information, see **System.GC.SuppressFinalize**, and **System.Object.Finalize**.

This method overrides **System.Object.Finalize**.]

StreamWriter.Flush() Method

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

[C#]
```
public override void Flush()
```

Summary

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

Description

[*Note:* This method overrides **System.IO.TextWriter.Flush**.]

Exceptions

Exception	Condition
System.ObjectDisposedException	The current writer is closed.
System.IO.IOException	An I/O error occurred.

StreamWriter.Write(System.String) Method

```
[ILASM]
.method public hidebysig virtual void Write(string value)

[C#]
public override void Write(string value)
```

Summary

Writes a string to the stream.

Parameters

Parameter	Description
value	The **System.String** to write to the stream. If *value* is **null**, nothing is written.

Description

The specified **System.String** is written to the underlying stream unless the end of the stream is reached prematurely.

If **System.IO.StreamWriter.AutoFlush** is **true**, **System.IO.StreamWriter.Flush** is invoked automatically.

[*Note:* This method overrides **System.IO.TextWriter.Write**.]

Exceptions

Exception	Condition
System.ObjectDisposedException	**System.IO.StreamWriter.AutoFlush** is **true** or the **System.IO.StreamWriter** buffer is full, and current writer is closed.
System.NotSupportedException	**System.IO.StreamWriter.AutoFlush** is **true** or the **System.IO.StreamWriter** buffer is full, and the contents of the buffer cannot be written to the underlying fixed size stream because the **System.IO.StreamWriter** is at the end the stream.
System.IO.IOException	An I/O error occurred.

StreamWriter.Write(System.Char[], System.Int32, System.Int32) Method

[ILASM]

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

[C#]

```
public override void Write(char[] buffer, int index, int
count)
```

Summary

Writes a sub-array of characters to the underlying stream.

Parameters

Parameter	Description
buffer	A **System.Char** array containing the data to write.
index	A **System.Int32** that specifies the index into *buffer* at which to begin writing.
count	A **System.Int32** that specifies the number of characters to read from *buffer*.

Description

The specified characters are written to the underlying stream unless the end of the stream is reached prematurely.

If **System.IO.StreamWriter.AutoFlush** is **true**, **System.IO.StreamWriter.Flush** is invoked automatically.

[*Note:* This method overrides **System.IO.TextWriter.Write**.]

Exceptions

Exception	Condition
System.ArgumentNullException	*buffer* is **null**.
System.ArgumentException	buffer.Length - *index* < *count*.
System.ArgumentOutOfRangeException	*index* or *count* is negative.
System.ObjectDisposedException	**System.IO.StreamWriter.AutoFlush** is **true** or the **System.IO.StreamWriter** buffer is full, and current writer is closed.

System.NotSupportedException	**System.IO.StreamWriter.AutoFlush** is **true** or the **System.IO.StreamWriter** buffer is full, and the contents of the buffer cannot be written to the underlying fixed size stream because the **System.IO.StreamWriter** is at the end the stream.
System.IO.IOException	An I/O error occurred.

StreamWriter.Write(System.Char[]) Method

[ILASM]

```
.method public hidebysig virtual void Write(class
System.Char[] buffer)
```

[C#]

```
public override void Write(char[] buffer)
```

Summary

Writes a character array to the underlying stream.

Parameters

Parameter	Description
buffer	A **System.Char** array containing the data to write. If *buffer* is **null**, nothing is written.

Description

The specified characters are written to the underlying stream unless the end of the stream is reached prematurely.

If **System.IO.StreamWriter.AutoFlush** is **true**, **System.IO.StreamWriter.Flush** is invoked automatically.

[*Note:* This method overrides **System.IO.TextWriter.Write**.]

Exceptions

Exception	Condition
System.ObjectDisposedException	**System.IO.StreamWriter.AutoFlush** is **true** or the **System.IO.StreamWriter** buffer is full, and current writer is closed.
System.NotSupportedException	**System.IO.StreamWriter.AutoFlush** is **true** or the **System.IO.StreamWriter** buffer is full, and the contents of the buffer cannot be written to the underlying fixed size stream because the **System.IO.StreamWriter** is at the end the stream.
System.IO.IOException	An I/O error occurred.

StreamWriter.Write(System.Char) Method

```
[ILASM]
.method public hidebysig virtual void Write(valuetype
System.Char value)

[C#]
public override void Write(char value)
```

Summary

Writes a character to the stream.

Parameters

Parameter	Description
value	The **System.Char** to write to the underlying stream.

Description

The specified character is written to the underlying stream unless the end of the stream is reached prematurely.

If **System.IO.StreamWriter.AutoFlush** is **true**, **System.IO.StreamWriter.Flush** is invoked automatically.

[*Note:* This method overrides **System.IO.TextWriter.Write**.]

Exceptions

Exception	Condition
System.ObjectDisposedException	**System.IO.StreamWriter.AutoFlush** is **true** or the **System.IO.StreamWriter** buffer is full, and current writer is closed.
System.NotSupportedException	**System.IO.StreamWriter.AutoFlush** is **true** or the **System.IO.StreamWriter** buffer is full, and the contents of the buffer cannot be written to the underlying fixed size stream because the **System.IO.StreamWriter** is at the end the stream.
System.IO.IOException	An I/O error occurred.

StreamWriter.AutoFlush Property

[ILASM]

```
.property bool AutoFlush { public hidebysig virtual
specialname bool get_AutoFlush() public hidebysig virtual
specialname void set_AutoFlush(bool value) }
```

[C#]

```
public virtual bool AutoFlush { get; set; }
```

Summary

Gets or sets a **System.Boolean** value indicating whether the current **System.IO.StreamWriter** will flush its buffer to the underlying stream after every call to **System.IO.StreamWriter.Write**.

Property Value

true to force **System.IO.StreamWriter** to flush its buffer; otherwise, **false**.

Description

The **System.IO.StreamWriter** will do a limited amount of buffering, both internally and potentially in the encoder from the encoding you passed in. If **System.IO.StreamWriter.AutoFlush** is set to **false**, the data will be flushed into the underlying stream only when the buffer is full, or when **System.IO.StreamWriter.Dispose**(**true**) or **System.IO.StreamWriter.Close** is called.

Setting **System.IO.StreamWriter.AutoFlush** to **true** forces **System.IO.StreamWriter** to flush the buffered data out of the encoder and call **System.IO.StreamWriter.Flush** on the stream every time **System.IO.StreamWriter.Write** is called.

Behaviors

As described above.

StreamWriter.BaseStream Property

[ILASM]

```
.property class System.IO.Stream BaseStream { public
hidebysig virtual specialname class System.IO.Stream
get_BaseStream() }
```

[C#]

```
public virtual Stream BaseStream { get; }
```

Summary

Gets the underlying stream.

Property Value

The **System.IO.Stream** the current **System.IO.StreamWriter** instance is writing to.

Behaviors

As described above.

StreamWriter.Encoding Property

[ILASM]

```
.property class System.Text.Encoding Encoding { public
hidebysig virtual specialname class System.Text.Encoding
get_Encoding() }
```

[C#]

```
public override Encoding Encoding { get; }
```

Summary

Gets the **System.Text.Encoding** in which the output is written.

Property Value

The **System.Text.Encoding** specified in the constructor for the current instance, or **System.Text.UTF8Encoding** if an encoding was not specified.

Description

[*Note:* This property overrides the **System.IO.TextWriter.Encoding** property.]

Behaviors

As described above.

Usage

This property is required in some XML scenarios where a header must be written containing the encoding used by the **System.IO.StreamWriter**. This allows XML code to consume an arbitrary **System.IO.StreamWriter** and generate a correct XML header.