System.Net.Sockets Library Reference

January 5, 2016

Contents

Description ii

Namespaces iii

Description

Provides support for TCP sockets.

Copyrights

Namespaces

Namespace	Description
Global	Global namespace contains C functions for the implementation.
System.Net.Sockets	Provides support for TCP sockets.

1 Usage

1.0.1 Referencing the System.Net.Sockets library

Right-click a project node in IDE | Project References... | Add System Extension Library Reference... | enable *System.Net.Sockets* check box

or add following line to your project's .cmp file:

```
reference <ext/System.Net.Sockets/System.Net.Sockets.cml>;
```

2 Global Namespace

Global namespace contains C functions for the implementation.

2.1 Concepts

Concept	Description

2.2 Functions

Function	Description

2.3 Enumerations

Enumeration	Description
ShutdownMode	Mode for the shut down operation.

2.3.0.1 ShutdownMode Enumeration

Mode for the shut down operation.

Enumeration Constants

Constant	Value	Description
both	2	Shuts down both receiving and sending.
receive	0	Shuts down receiving from a socket.
send	1	Shuts down sending to a socket.

3 System.Net.Sockets Namespace

Provides support for TCP sockets.

3.4 Classes

Class	Description
NetworkBuffer	A handle to a dynamically allocated memory.
SocketByteStream	Represent stream of bytes connected to a TcpSocket.
SocketError	An exception class throw when a socket operation fails.
SocketLibrary	Represents the socket library initializer implemented as a singleton.
SocketLibraryException	Exception class thrown when the initialization of the socket library fails.
TcpSocket	Represents a TCP socket.

3.4.1 NetworkBuffer Class

A handle to a dynamically allocated memory.

Syntax

```
public class NetworkBuffer;
```

3.4.1.1 Member Functions

Member Function	Description
NetworkBuffer(System.Net.Sockets.NetworkBuffer&&)	Move constructor.
operator=(System.Net.Sockets.NetworkBuffer&&)	Move assignment.
Mem() const	Returns a pointer to the allocated memory block.
NetworkBuffer(int)	Constructor. Allocates specified number of bytes from the system.
Size() const	Returns the size of the allocated memory block.
∼NetworkBuffer()	Destructor. Frees the allocated memory back to the system.

NetworkBuffer(System.Net.Sockets.NetworkBuffer&&) Member Function

Move constructor.

Syntax

```
public NetworkBuffer(System.Net.Sockets.NetworkBuffer&& that);
```

Parameters

Name	Type	Description
that	System.Net.Sockets.NetworkBuffer&&	

operator=(System.Net.Sockets.NetworkBuffer&&) Member Function

Move assignment.

Syntax

```
public void operator=(System.Net.Sockets.NetworkBuffer&& __parameter0);
```

Parameters

Name	Type	Description
__parameter0	System.Net.Sockets.NetworkBuffer&&	

Mem() const Member Function

Returns a pointer to the allocated memory block.

Syntax

```
public void* Mem() const;
```

Returns

void*

Returns a pointer to the allocated memory block.

NetworkBuffer(int) Member Function

Constructor. Allocates specified number of bytes from the system.

Syntax

```
public NetworkBuffer(int size_);
```

Parameters

Name	Type	Description
size_	int	The number of bytes to allocate.

Size() const Member Function

Returns the size of the allocated memory block.

Syntax

```
public int Size() const;
```

Returns

int

Returns the size of the allocated memory block.

∼NetworkBuffer() Member Function

Destructor. Frees the allocated memory back to the system.

Syntax

```
public ∼NetworkBuffer();
```

3.4.2 SocketByteStream Class

Represent stream of bytes connected to a TcpSocket.

Syntax

```
public class SocketByteStream;
```

Base Class

System.IO.ByteStream

3.4.2.1 Member Functions

Member Function	Description
SocketByteStream()	Default constructor.
SocketByteStream(System.Net.Sockets.-SocketByteStream&&)	Move constructor.
operator=(System.Net.Sockets.-SocketByteStream&&)	Move assignment.
Read(byte*, int)	Reads at most given number of bytes from the TcpSocket to the given buffer. Returns the number of bytes read.
ReadByte()	Reads one byte of data from the TcpSocket and returns it. Return value of -1 indicates peer has shut down the connection (end of stream).
SocketByteStream(System.Net.Sockets.-TcpSocket&&)	Constructor. Initializes the socket byte stream with the given TcpSocket.
SocketByteStream(const System.String&, const System.String&)	Constructor. Initializes the socket byte stream by creating a TCP socket and connecting it to the specified node and service.
Write(byte)	Writes given byte to the socket byte stream.
Write(byte*, int)	Writes given number of bytes from the given buffer to the socket byte stream.
~SocketByteStream()	Destructor. Closes the socket.

SocketByteStream() Member Function

Default constructor.

Syntax

```
public SocketByteStream();
```

SocketByteStream(System.Net.Sockets.SocketByteStream&&) Member Function

Move constructor.

Syntax

```
public SocketByteStream(System.Net.Sockets.SocketByteStream&& __parameter0);
```

Parameters

Name	Type	Description
__parameter0	System.Net.Sockets.SocketByteStream&&	

operator=(System.Net.Sockets.SocketByteStream&&) Member Function

Move assignment.

Syntax

```
public void operator=(System.Net.Sockets.SocketByteStream&& __parameter0);
```

Parameters

Name	Type	Description
__parameter0	System.Net.Sockets.SocketByteStream&&	

Read(byte*, int) Member Function

Reads at most given number of bytes from the TcpSocket to the given buffer. Returns the number of bytes read.

Syntax

```
public int Read(byte* buf, int count);
```

Parameters

Name	Type	Description
buf	byte*	A buffer to read to.
count	int	Maximum number of bytes to read.

Returns

int

Returns number of bytes read. Return value of 0 indicates peer has shut down the connection (end of stream).

Remarks

Throws SocketError if reading fails.

ReadByte() Member Function

Reads one byte of data from the TcpSocket and returns it. Return value of -1 indicates peer has shut down the connection (end of stream).

Syntax

```
public int ReadByte();
```

Returns

int

Returns the byte read, or -1 if end of stream encountered.

Remarks

Throws SocketError if reading fails.

SocketByteStream(System.Net.Sockets.TcpSocket&&) Member Function

Constructor. Initializes the socket byte stream with the given TcpSocket.

Syntax

```
public SocketByteStream(System.Net.Sockets.TcpSocket&& socket_);
```

Parameters

Name	Type	Description
socket_	System.Net.Sockets.TcpSocket&&	A TcpSocket.

SocketByteStream(const System.String&, const System.String&) Member Function

Constructor. Initializes the socket byte stream by creating a TCP socket and connecting it to the specified node and service.

Syntax

```
public SocketByteStream(const System.String& node, const System.String& service);
```

Parameters

Name	Type	Description
node	const System.String&	A host name or an IP address to connect.
service	const System.String&	A protocol name or port number to connect.

Remarks

Throws SocketError if connecting fails.

Write(byte) Member Function

Writes given byte to the socket byte stream.

Syntax

```
public void Write(byte x);
```

Parameters

Name	Type	Description
x	byte	A byte to write.

Remarks

Throws SocketError if writing fails.

Write(byte*, int) Member Function

Writes given number of bytes from the given buffer to the socket byte stream.

Syntax

```
public void Write(byte* buf, int count);
```

Parameters

Name	Type	Description
buf	byte*	A buffer of data to write.
count	int	Number of bytes to write.

Remarks

Throws SocketError if writing fails.

∼SocketByteStream() Member Function

Destructor. Closes the socket.

Syntax

```
public ∼SocketByteStream();
```

3.4.3 SocketError Class

An exception class throw when a socket operation fails.

Syntax

```
public class SocketError;
```

Base Class

System.Exception

3.4.3.1 Member Functions

Member Function	Description
SocketError()	Default constructor.
SocketError(const System.Net.Sockets.SocketError&)	Copy constructor.
operator=(const System.Net.Sockets.SocketError&)	Copy assignment.
SocketError(System.Net.Sockets.SocketError&&)	Move constructor.
operator=(System.Net.Sockets.SocketError&&)	Move assignment.
ErrorCode() const	Returns the error code.
SocketError(const System.String&, const System.String&, int)	Constructor. Initializes the socket error with the specified operation text, error description text and error code.
SocketError(const System.String&, int)	Constructor. Initializes the socket error with the specified operation text, retrieved error description and the specified error code.

SocketError() Member Function

Default constructor.

Syntax

```
public SocketError();
```

SocketError(const System.Net.Sockets.SocketError&) Member Function

Copy constructor.

Syntax

```
public SocketError(const System.Net.Sockets.SocketError& __parameter0);
```

Parameters

Name	Type	Description
__parameter0	const System.Net.Sockets.SocketError&	

operator=(const System.Net.Sockets.SocketError&) Member Function

Copy assignment.

Syntax

```
public void operator=(const System.Net.Sockets.SocketError& that);
```

Parameters

Name	Type	Description
that	const System.Net.Sockets.SocketError&	Argument to assign.

SocketError(System.Net.Sockets.SocketError&&) Member Function

Move constructor.

Syntax

```
public SocketError(System.Net.Sockets.SocketError&& that);
```

Parameters

Name	Type	Description
that	System.Net.Sockets.SocketError&&	Argument to move from.

operator=(System.Net.Sockets.SocketError&&) Member Function

Move assignment.

Syntax

```
public void operator=(System.Net.Sockets.SocketError&& that);
```

Parameters

Name	Type	Description
that	System.Net.Sockets.SocketError&&	Argument to assign from.

ErrorCode() const Member Function

Returns the error code.

Syntax

```
public int ErrorCode() const;
```

Returns

int

Returns the error code.

SocketError(const System.String&, const System.String&, int) Member Function

Constructor. Initializes the socket error with the specified operation text, error description text and error code.

Syntax

```
public SocketError(const System.String& operation, const System.String& errorMessage,
int errorCode_);
```

Parameters

Name	Type	Description
operation	const System.String&	Description of the failed operation.
errorMessage	const System.String&	Description of the error.
errorCode_	int	Error code.

SocketError(const System.String&, int) Member Function

Constructor. Initializes the socket error with the specified operation text, retrieved error description and the specified error code.

Syntax

```
public SocketError(const System.String& operation, int errorCode_);
```

Parameters

Name	Type	Description
operation	const System.String&	Description of failed operation.
errorCode_	int	Error code.

3.4.4 SocketLibrary Class

Represents the socket library initializer implemented as a singleton.

Syntax

```
public class SocketLibrary;
```

3.4.4.1 Member Functions

Member Function	Description
Init()	Initializes the socket library.
Instance()	Returns a reference to the socket library singleton instance.
~SocketLibrary()	Destructor. Uninitializes the socket library.

Init() Member Function

Initializes the socket library.

Syntax

```
public void Init();
```

Instance() Member Function

Returns a reference to the socket library singleton instance.

Syntax

```
public static System.Net.Sockets.SocketLibrary& Instance();
```

Returns

System.Net.Sockets.SocketLibrary&

Returns a reference to the socket library singleton instance.

∼SocketLibrary() Member Function

Destructor. Uninitializes the socket library.

Syntax

```
public ∼SocketLibrary();
```

3.4.5 SocketLibraryException Class

Exception class thrown when the initialization of the socket library fails.

Syntax

```
public class SocketLibraryException;
```

Base Class

System.Exception

3.4.5.1 Member Functions

Member Function	Description
SocketLibraryException()	Default constructor.
SocketLibraryException(const System.Net.Sockets.SocketLibraryException&)	Copy constructor.
operator=(const System.Net.Sockets.SocketLibraryException&)	Copy assignment.
SocketLibraryException(System.Net.Sockets.SocketLibraryException&&)	Move constructor.
operator=(System.Net.Sockets.SocketLibraryException&&)	Move assignment.
SocketLibraryException(const System.String&)	Constructor. Initializes the socket library exception with the specified error message.

SocketLibraryException() Member Function

Default constructor.

Syntax

```
public SocketLibraryException();
```

SocketLibraryException(const System.Net.Sockets.SocketLibraryException&) Member Function

Copy constructor.

Syntax

```
public SocketLibraryException(const System.Net.Sockets.SocketLibraryException& __parameter0);
```

Parameters

Name	Type	Description
__parameter0	const System.Net.Sockets.SocketLibraryException&	

operator=(const System.Net.Sockets.SocketLibraryException&) Member Function

Copy assignment.

Syntax

```
public void operator=(const System.Net.Sockets.SocketLibraryException& that);
```

Parameters

Name	Type	Description
that	const System.Net.Sockets.SocketLibraryException&	Argument to assign.

SocketLibraryException(System.Net.Sockets.SocketLibraryException&&) Member Function

Move constructor.

Syntax

```
public SocketLibraryException(System.Net.Sockets.SocketLibraryException&& that);
```

Parameters

Name	Type	Description
that	System.Net.Sockets.SocketLibraryException&&	Argument to move from.

operator=(System.Net.Sockets.SocketLibraryException&&) Member Function

Move assignment.

Syntax

```
public void operator=(System.Net.Sockets.SocketLibraryException&& that);
```

Parameters

Name	Type	Description
that	System.Net.Sockets.SocketLibraryException&&	Argument to assign from.

SocketLibraryException(const System.String&) Member Function

Constructor. Initializes the socket library exception with the specified error message.

Syntax

```
public SocketLibraryException(const System.String& message_);
```

Parameters

Name	Type	Description
message_	const System.String&	An error message.

3.4.6 TcpSocket Class

Represents a TCP socket.

Syntax

```
public class TcpSocket;
```

3.4.6.1 Member Functions

Member Function	Description
TcpSocket()	Default constructor. Creates an unbound TCP socket.
TcpSocket(System.Net.Sockets.TcpSocket&&)	Move constructor.
operator=(System.Net.Sockets.TcpSocket&&)	Move assignment.
Accept()	Accepts a connection to a bound socket and returns a new connected TCP socket that represents the connection.
Bind(int)	Binds the socket to a port.
Close()	Closes the socket.
GetSocketHandle() const	Returns the socket handle.
Listen(int)	Begins listening connections to a bound TCP socket.
Receive(void*, int)	Receives data from a connected socket.
ReceiveAll()	Receives rest of data from a connected socket. That is: receives data until the peer shuts down its sending side of the connection.
Send(const System.String&)	Sends a string of data to a connected socket.
Send(const void*, int)	Sends data to a connected socket.
Shutdown(ShutdownMode)	Shuts down a connected socket.
TcpSocket(const System.String&, const System.String&)	Constructor. Creates a TCP socket and connects it to the specified node and service.
TcpSocket(int)	Constructor. Initializes a TCP socket with an existing socket handle.

~TcpSocket()	Destructor. Closes the socket if it is bound or connected.

TcpSocket() Member Function

Default constructor. Creates an unbound TCP socket.

Syntax

```
public TcpSocket();
```

TcpSocket(System.Net.Sockets.TcpSocket&&) Member Function

Move constructor.

Syntax

```
public TcpSocket(System.Net.Sockets.TcpSocket&& that);
```

Parameters

Name	Type	Description
that	System.Net.Sockets.TcpSocket&&	

operator=(System.Net.Sockets.TcpSocket&&) Member Function

Move assignment.

Syntax

```
public void operator=(System.Net.Sockets.TcpSocket&& that);
```

Parameters

Name	Type	Description
that	System.Net.Sockets.TcpSocket&&	

Accept() Member Function

Accepts a connection to a bound socket and returns a new connected TCP socket that represents the connection.

Syntax

```
public System.Net.Sockets.TcpSocket Accept();
```

Returns

System.Net.Sockets.TcpSocket

Returns a connected TCP socket that represents the connection.

Bind(int) Member Function

Binds the socket to a port.

Syntax

```
public void Bind(int port);
```

Parameters

Name	Type	Description
port	int	A port number to which to bind.

Close() Member Function

Closes the socket.

Syntax

```
public void Close();
```

GetSocketHandle() const Member Function

Returns the socket handle.

Syntax

```
public int GetSocketHandle() const;
```

Returns

int

Returns the socket handle.

Listen(int) Member Function

Begins listening connections to a bound TCP socket.

Syntax

```
public void Listen(int backlog);
```

Parameters

Name	Type	Description
backlog	int	The number of pending connections.

Receive(void*, int) Member Function

Receives data from a connected socket.

Syntax

```
public int Receive(void* buf, int len);
```

Parameters

Name	Type	Description
buf	void*	A buffer.
len	int	Maximum number of bytes to receive.

Returns

int

Returns the number of bytes received. This might be less than the number of bytes requested. Return value of 0 indicates that the peer has shut down the connection.

Remarks

Throws SocketError if reading fails.

ReceiveAll() Member Function

Receives rest of data from a connected socket. That is: receives data until the peer shuts down its sending side of the connection.

Syntax

```
public System.String ReceiveAll();
```

Returns

System.String

Returns the received data as a string.

Remarks

Throws SocketError if reading fails.

Send(const System.String&) Member Function

Sends a string of data to a connected socket.

Syntax

```
public void Send(const System.String& s);
```

Parameters

Name	Type	Description
s	const System.String&	A string to send.

Remarks

Throws SocketError if sending fails.

Send(const void*, int) Member Function

Sends data to a connected socket.

Syntax

```
public int Send(const void* buf, int len);
```

Parameters

Name	Type	Description
buf	const void*	A buffer of data to send.
len	int	Maximum number of bytes to send.

Returns

int

Returns the number of bytes sent. This might be less than the number of bytes requested.

Remarks

Throws SocketError if sending fails.

Shutdown(ShutdownMode) Member Function

Shuts down a connected socket.

Syntax

```
public void Shutdown(ShutdownMode mode);
```

Parameters

Name	Type	Description
mode	ShutdownMode	Shut down mode.

TcpSocket(const System.String&, const System.String&) Member Function

Constructor. Creates a TCP socket and connects it to the specified node and service.

Syntax

```
public TcpSocket(const System.String& node, const System.String& service);
```

Parameters

Name	Type	Description
node	const System.String&	A host name or an IP address to connect.
service	const System.String&	A protocol name or port number to connect.

TcpSocket(int) Member Function

Constructor. Initializes a TCP socket with an existing socket handle.

Syntax

```
public TcpSocket(int socket_);
```

Parameters

Name	Type	Description
socket_	int	A handle of an existing TCP socket.

∼TcpSocket() Member Function

Destructor. Closes the socket if it is bound or connected.

Syntax

```
public ∼TcpSocket();
```

3.5 Constants

Constant	Type	Value	Description
invalidSocketHandle	int	-1	Represents invalid socket handle.