

# ByteString Members

**Namespace:** Opc.UaFx

**Assemblies:** Opc.UaFx.Advanced.dll

The [ByteString](#) struct defines the following members.

## Constructors

### ByteString(Byte[])

Initializes a new [ByteString](#) from the [value](#) specified.

**C#**

```
public ByteString(params byte[] value)
```

#### Parameters

[value](#) [Byte](#)[]

The array of [Byte](#) values to represent or a null reference (Nothing in Visual Basic).

#### Remarks

If [value](#) is a null reference (Nothing in Visual Basic) an empty array of [Byte](#) values is used.

## Properties

### Item[Int32]

Gets the [Byte](#) at the specified index.

**C#**

```
public byte this[int index] { get; set; }
```

#### Property Value

[Byte](#)

The [Byte](#) at the specified index.

#### Exceptions

[ArgumentOutOfRangeException](#)

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

## Length

Gets the total number of elements in all the dimensions of the `ByteString`.

**C#**

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

### Property Value

`Int32`

The total number of elements in all the dimensions of the `ByteString`; zero if there are no elements in the array.

## Methods

### CompareTo(ByteString)

Compares the current `ByteString` with another `ByteString`.

**C#**

```
public int CompareTo(ByteString other)
```

### Parameters

`other` `ByteString`

The `ByteString` to compare with this `ByteString`.

### Returns

`Int32`

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

### CompareTo(Object)

Compares the current `ByteString` with the `other`.

**C#**

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

### Parameters

**other** [Object](#)

The [ByteString](#) to compare with this [ByteString](#).

### Returns

[Int32](#)

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

## Equals(ByteString)

Determines whether the specified **other** is equal to this [ByteString](#).

**C#**

```
public bool Equals(ByteString other)
```

### Parameters

**other** [ByteString](#)

The [ByteString](#) to compare to the current [ByteString](#).

### Returns

[Boolean](#)

The value true if the specified [ByteString](#) is equal to the current [ByteString](#); otherwise the value false.

## Equals(Object)

Determines whether the specified **other** is equal to this [ByteString](#).

**C#**

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

### Parameters

**other** [Object](#)

The [ByteString](#) to compare to the current [ByteString](#).

### Returns

[Boolean](#)

The value true if the specified [ByteString](#) is equal to the current [ByteString](#); otherwise the value false.

# GetHashCode()

Retrieves a hash code for this [ByteString](#).

## C#

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

## Returns

[Int32](#)

An [Int32](#) that contains the hash code for the [ByteString](#).

# Parse(String)

Converts the specified hex encoded byte array to its [ByteString](#) equivalent.

## C#

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

## Parameters

[value](#) [String](#)

A [String](#) containing the hex encoded byte array to convert.

## Returns

[ByteString](#)

A [ByteString](#) which represents the hex encoded byte array of the [value](#) specified.

## Exceptions

[ArgumentNullException](#)

The [value](#) is a null reference (Nothing in Visual Basic).

[FormatException](#)

The [value](#) is not a valid hex encoded byte array.

# ToArray()

## C#

```
public byte[] ToArray()
```

## Returns

Byte[]

## ToString()

Converts the value of this instance to its equivalent hex encoded string representation.

**C#**

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

### Returns

String

The hex encoded string representation.

## TryParse(String, out ByteString)

Tries to convert the specified hex encoded byte array to its [ByteString](#) equivalent. A return value indicates whether the conversion succeeded or failed.

**C#**

```
public static bool TryParse(string value, out ByteString result)
```

### Parameters

value String

A [String](#) containing the hex encoded byte array to convert.

result [ByteString](#)

When this method returns, if the conversion succeeded, the [ByteString](#) if **value** is a valid hex encoded byte array. If the conversion failed, contains a default [ByteString](#). The conversion fails if **value** is a null reference (Nothing in Visual Basic) or is not a valid hex encoded byte array.

### Returns

Boolean

The value true if **value** was converted successfully; otherwise the value false.

## Operators

### Equality(ByteString, ByteString)

Returns a value indicating whether two values of [ByteString](#) are equal.

**C#**

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

## Explicit(ByteString to Byte[])

Converts a [ByteString](#) to a [Byte](#) array.

**C#**

```
public static explicit operator byte[](ByteString value)
```

## GreaterThan(ByteString, ByteString)

Determines whether the first specified [ByteString](#) value is greater than the second specified [ByteString](#) value.

**C#**

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

## GreaterThanOrEqual(ByteString, ByteString)

Determines whether the first specified [ByteString](#) value is greater than or equal to the second specified [ByteString](#) value.

**C#**

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

## Implicit(Byte to ByteString)

Converts a [Byte](#) to a [ByteString](#) value.

**C#**

```
public static implicit operator ByteString(byte value)
```

## Implicit(Byte[] to ByteString)

Converts a [Byte](#) array to a [ByteString](#) value.

**C#**

```
public static implicit operator ByteString(byte[] value)
```

## Inequality(ByteString, ByteString)

Returns a value indicating whether two values of [ByteString](#) are not equal.

**C#**

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

## LessThan(ByteString, ByteString)

Determines whether the first specified [ByteString](#) value is less than the second specified [ByteString](#) value.

**C#**

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

## LessThanOrEqual(ByteString, ByteString)

Determines whether the first specified [ByteString](#) value is less than or equal to the second [ByteString](#) value.

**C#**

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



# Table of Contents

|                                            |   |
|--------------------------------------------|---|
| <b>Constructors</b>                        | 1 |
| ByteString(Byte[])                         | 1 |
| <b>Properties</b>                          | 1 |
| Item[Int32]                                | 1 |
| Length                                     | 2 |
| <b>Methods</b>                             | 2 |
| CompareTo(ByteString)                      | 2 |
| CompareTo(Object)                          | 2 |
| Equals(ByteString)                         | 3 |
| Equals(Object)                             | 3 |
| GetHashCode()                              | 4 |
| Parse(String)                              | 4 |
| ToArray()                                  | 4 |
| ToString()                                 | 5 |
| TryParse(String, out ByteString)           | 5 |
| <b>Operators</b>                           | 5 |
| Equality(ByteString, ByteString)           | 5 |
| Explicit(ByteString to Byte[])             | 6 |
| GreaterThan(ByteString, ByteString)        | 6 |
| GreaterThanOrEqual(ByteString, ByteString) | 6 |
| Implicit(Byte to ByteString)               | 6 |
| Implicit(Byte[] to ByteString)             | 6 |
| Inequality(ByteString, ByteString)         | 6 |
| LessThan(ByteString, ByteString)           | 7 |
| LessThanOrEqual(ByteString, ByteString)    | 7 |