System.Collections.Specialized.NameValueCollection Class

```
[ILASM]
.class public serializable NameValueCollection extends
System.Object

[C#]
public class NameValueCollection: Object
```

Assembly Info:

- *Name:* System
- *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)

Type Attributes:

- DefaultMemberAttribute("Item") [*Note:* This attribute requires the RuntimeInfrastructure library.]

Implements:

- **System.Collections.ICollection**
- **System.Collections.IEnumerable**

Summary

Represents a collection of associated **System.String** keys and **System.String** values.

Inherits From: System.Object

Library: Networking

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

Description

This class can be used for headers, query strings and form data. Each key in the collection is associated with one or more values. Multiple values for a particular key are contained in a single **System.String** are stored as a **System.Collections.ArrayList**.

The capacity is the number of key-and-value pairs that the **System.Collections.Specialized.NameValueCollection** can contain. The default initial capacity is zero. The capacity is automatically increased as required.

The hash code provider dispenses hash codes for keys in the **System.Collections.Specialized.NameValueCollection**.

The comparer determines whether two keys are equal.

NameValueCollection() Constructor

[ILASM]

```
public rtspecialname specialname instance void .ctor()
```

[C#]

```
public NameValueCollection()
```

Summary

Constructs and initializes a new instance of the **System.Collections.Specialized.NameValueCollection** class.

Description

The new instance is initialized with the default initial capacity, **System.Collections.IHashCodeProvider**, and **System.Collections.IComparer**.

NameValueCollection(System.Collections.Specialized.NameValueCollection) Constructor

```
[ILASM]
public rtspecialname specialname instance void .ctor(class
System.Collections.Specialized.NameValueCollection col)

[C#]
public NameValueCollection(NameValueCollection col)
```

Summary

Constructs and initializes a new instance of the **System.Collections.Specialized.NameValueCollection** class using the values of the specified **System.Collections.Specialized.NameValueCollection**.

Parameters

Parameter	Description
col	The **System.Collections.Specialized.NameValueCollection** used to initialize the new instance.

Description

The capacity, values, and order of values of the new instance are equal to the capacity and values of *col*. The **System.Collections.IHashCodeProvider** and **System.Collections.IComparer** of the new instance are the default instances.

The elements of the new **System.Collections.Specialized.NameValueCollection** are sorted in the same order as the source **System.Collections.Specialized.NameValueCollection**.

Exceptions

Exception	Condition
System.ArgumentNullException	*col* is **null**.

NameValueCollection(System.Collections.IHashCodeProvider, System.Collections.IComparer) Constructor

```
[ILASM]
public rtspecialname specialname instance void .ctor(class
System.Collections.IHashCodeProvider hashProvider, class
System.Collections.IComparer comparer)

[C#]
public NameValueCollection(IHashCodeProvider hashProvider,
IComparer comparer)
```

Summary

Constructs and initializes a new instance of the **System.Collections.Specialized.NameValueCollection** class with the specified **System.Collections.IHashCodeProvider** and the specified **System.Collections.IComparer**.

Parameters

Parameter	Description
hashProvider	The **System.Collections.IHashCodeProvider** that supplies the hash codes for all keys in the new instance; or, **null** to use the default hash code provider.
comparer	The **System.Collections.IComparer** to use to determine whether two keys are equal. Specify **null** to use the default comparer.

Description

The new instance is initialized with the default capacity of zero.

NameValueCollection(System.Int32) Constructor

```
[ILASM]
public rtspecialname specialname instance void .ctor(int32
capacity)

[C#]
public NameValueCollection(int capacity)
```

Summary

Constructs and initializes a new instance of the **System.Collections.Specialized.NameValueCollection** class with the specified initial capacity.

Parameters

Parameter	Description
capacity	A **System.Int32** containing the initial number of entries that the new instance can contain.

Description

The new instance is initialized with the default **System.Collections.IHashCodeProvider** and **System.Collections.IComparer**.

Exceptions

Exception	Condition
System.ArgumentOutOfRangeException	*capacity* < 0.

NameValueCollection(System.Int32, System.Collections.Specialized.NameValueCollection) Constructor

[ILASM]
```
public rtspecialname specialname instance void .ctor(int32
capacity, class
System.Collections.Specialized.NameValueCollection col)
```

[C#]
```
public NameValueCollection(int capacity,
NameValueCollection col)
```

Summary

Constructs and initializes new instance of the **System.Collections.Specialized.NameValueCollection** class that contains the same values as the specified **System.Collections.Specialized.NameValueCollection** and either the specified capacity or the capacity of the specified collection, whichever is greater.

Parameters

Parameter	Description
capacity	A **System.Int32** containing the initial number of entries that the new instance can contain.
col	The **System.Collections.Specialized.NameValueCollection** used to initialize the new instance.

Description

The new instance is initialized with the default **System.Collections.IHashCodeProvider** and **System.Collections.IComparer**.

Exceptions

Exception	Condition
System.ArgumentNullException	*col* is **null**.

NameValueCollection(System.Int32, System.Collections.IHashCodeProvider, System.Collections.IComparer) Constructor

```
[ILASM]
public rtspecialname specialname instance void .ctor(int32
capacity, class System.Collections.IHashCodeProvider
hashProvider, class System.Collections.IComparer comparer)

[C#]
public NameValueCollection(int capacity, IHashCodeProvider
hashProvider, IComparer comparer)
```

Summary

Constructs and initializes new instance of the **System.Collections.Specialized.NameValueCollection** class with the specified initial capacity, hash code provider, and comparer.

Parameters

Parameter	Description
capacity	A **System.Int32** containing the initial number of entries that the **System.Collections.Specialized.NameValueCollection** can contain.
hashProvider	The **System.Collections.IHashCodeProvider** that will supply the hash codes for all keys in the new instance.
comparer	The **System.Collections.IComparer** to use to determine whether two keys in the new instance are equal.

Exceptions

Exception	Condition
System.ArgumentOutOfRangeException	*capacity* < 0.

NameValueCollection.Add(System.String, System.String) Method

```
[ILASM]
.method public hidebysig virtual void Add(string name,
string value)

[C#]
public virtual void Add(string name, string value)
```

Summary

Adds an entry with the specified key and value to the current instance.

Parameters

Parameter	Description
name	A **System.String** that represents the key of the entry to add. Can be **null**.
value	A **System.String** that represents the value of the entry to add. Can be **null**.

Behaviors

As described above.

Default

If the specified key already exists in the current instance, the specified value is added to the existing comma-separated list of values associated with the same key.

If the same value already exists under the same key in the collection, the new value overwrites the old value.

How and When to Override

Override this method to customize the default behavior in a type derived from the current type.

Usage

Use this method to add an entry to the current instance.

Exceptions

Exception	Condition
System.NotSupportedException	The current instance is read-only.

NameValueCollection.Add(System.Collections.Specialized.NameValueCollection) Method

```
[ILASM]
.method public hidebysig instance void Add(class
System.Collections.Specialized.NameValueCollection c)

[C#]
public void Add(NameValueCollection c)
```

Summary

Copies the entries from the specified **System.Collections.Specialized.NameValueCollection** to the current instance.

Parameters

Parameter	Description
c	The **System.Collections.Specialized.NameValueCollection** to copy to the current instance.

Description

If a key in *c* already exists in the target **System.Collections.Specialized.NameValueCollection** instance, the associated value in *c* is added to the existing comma-separated list of values associated with the same key in the target **System.Collections.Specialized.NameValueCollection** instance.

Exceptions

Exception	Condition
System.NotSupportedException	The current instance is read-only.
System.ArgumentNullException	*c* is **null**.

NameValueCollection.Clear() Method

[ILASM]

```
.method public hidebysig instance void Clear()
```

[C#]

```
public void Clear()
```

Summary

Invalidates the cached arrays and removes all entries from the current instance.

Description

The value of each key and value in the current instance is set to **null**.

If the current instance is empty, it remains unchanged and no exception is thrown.

Exceptions

Exception	Condition
System.NotSupportedException	The current instance is read-only.

NameValueCollection.CopyTo(System.Array, System.Int32) Method

```
[ILASM]
.method public hidebysig instance void CopyTo(class
System.Array dest, int32 index)

[C#]
public void CopyTo(Array dest, int index)
```

Summary

Copies the elements from the current instance to the specified **System.Array**, starting at the specified index in that array.

Parameters

Parameter	Description
dest	A one-dimensional, zero-based **System.Array** that is the destination of the elements copied from the current instance.
index	A **System.Int32** containing the zero-based index in *dest* at which copying begins.

Description

This method uses **System.Array.Copy** to copy the elements.

[*Note:* This method is implemented to support the **System.Collections.ICollection** interface.]

Exceptions

Exception	Condition
System.ArgumentNullException	*dest* is **null**.
System.ArgumentOutOfRangeException	*index* < 0.
System.ArgumentException	*dest* has more than one dimension. *index* >= *dest*.Length. The number of elements in the current instance is greater than the available space from *index* to the end of the destination *dest*.

System.InvalidCastException	At least one element in the current instance is not assignment-compatible with the type of *dest*.

NameValueCollection.Get(System.Int32) Method

```
[ILASM]
.method public hidebysig virtual string Get(int32 index)

[C#]
public virtual string Get(int index)
```

Summary

Returns the values at the specified index of the current instance.

Parameters

Parameter	Description
index	A **System.Int32** that specifies the zero-based index of the entry that contains the values to get from the current instance.

Return Value

A **System.String** that contains a comma-separated list of the values at the specified index of the current instance, if found; otherwise, **null**.

Behaviors

As described above.

Exceptions

Exception	Condition
System.ArgumentOutOfRangeException	*index* is outside the valid range of indices for the current instance.

NameValueCollection.Get(System.String) Method

[ILASM]

```
.method public hidebysig virtual string Get(string name)
```

[C#]

```
public virtual string Get(string name)
```

Summary

Gets the values associated with the specified key from the current instance combined into one comma-separated list.

Parameters

Parameter	Description
name	A **System.String** that specified the key of the entry that contains the values to get.

Return Value

A **System.String** that contains a comma-separated list of the values associated with the specified key from the current instance, if found; otherwise, **null**.

Behaviors

As described above.

Default

If *name* is **null**, no exception is thrown and **null** is returned.

NameValueCollection.GetKey(System.Int32) Method

```
[ILASM]
.method public hidebysig virtual string GetKey(int32 index)

[C#]
public virtual string GetKey(int index)
```

Summary

Returns the key at the specified index of the current instance.

Parameters

Parameter	Description
index	A **System.Int32** that specifies the zero-based index of the key to get from the current instance.

Return Value

A **System.String** that contains the key at the specified index of the current instance, if found; otherwise, **null**.

Behaviors

As described above.

Exceptions

Exception	Condition
System.ArgumentOutOfRangeException	*index* is outside the valid range of indices for the current instance.

NameValueCollection.GetValues(System.Int32) Method

[ILASM]

```
.method public hidebysig virtual class System.String[]
GetValues(int32 index)
```

[C#]

```
public virtual string[] GetValues(int index)
```

Summary

Returns an array that contains the values at the specified index of the current instance.

Parameters

Parameter	Description
index	A **System.Int32** that specifies the zero-based index of the entry that contains the values to get from the current instance.

Return Value

A **System.String** array containing the values at the specified index of the current instance, if found; otherwise, **null**.

Behaviors

As described above.

Exceptions

Exception	Condition
System.ArgumentOutOfRangeException	*index* is outside the valid range of indices for the current instance.

NameValueCollection.GetValues(System.String) Method

```
[ILASM]
.method public hidebysig virtual class System.String[]
GetValues(string name)

[C#]
public virtual string[] GetValues(string name)
```

Summary

Gets the values associated with the specified key from the current instance.

Parameters

Parameter	Description
name	A **System.String** that specifies the key of the entry that contains the values to get.

Return Value

A **System.String** array containing the values associated with *name* from the current instance, if found; otherwise, **null**.

Behaviors

As described above.

Default

If *name* is **null**, no exception is thrown and **null** is returned.

NameValueCollection.HasKeys() Method

[ILASM]

```
.method public hidebysig instance bool HasKeys()
```

[C#]

```
public bool HasKeys()
```

Summary

Gets a **System.Boolean** value indicating whether the current instance contains keys that are not **null**.

Return Value

true if the current instance contains keys that are not **null**; otherwise, **false**.

NameValueCollection.InvalidateCachedArrays() Method

[ILASM]

```
.method family hidebysig instance void
InvalidateCachedArrays()
```

[C#]

```
protected void InvalidateCachedArrays()
```

Summary

Resets the cached arrays of the current instance to **null**.

Description

[*Note:* The array returned by **System.Collections.Specialized.NameValueCollection.AllKeys** is cached for better performance and is automatically refreshed when the collection changes. A derived class can invalidate the cached version by calling **System.Collections.Specialized.NameValueCollection.InvalidateCachedArrays**, thereby forcing the arrays to be recreated.]

NameValueCollection.Remove(System.String) Method

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

[C#]
public virtual void Remove(string name)
```

Summary

Removes the entry with the specified key from the current instance.

Parameters

Parameter	Description
name	A **System.String** containing the key of the entry to remove from the current instance.

Behaviors

If *name* is found, the key *name* and its associated value are set to **null**. Removing an element does not alter the capacity of a **System.Collections.Specialized.NameValueCollection**.

Default

This method uses the **System.Object.Equals** implementation of *name* to locate *name* in the current instance. If *name* is not found in the current instance or is **null**, no exception is thrown and the current instance is unchanged.

Exceptions

Exception	Condition
System.NotSupportedException	The current instance is read-only or has a fixed size.

NameValueCollection.Set(System.String, System.String) Method

```
[ILASM]
.method public hidebysig virtual void Set(string name,
string value)

[C#]
public virtual void Set(string name, string value)
```

Summary

Sets the value of the specified entry in the current instance to the specified value.

Parameters

Parameter	Description
name	A **System.String** containing the key of the entry to add the new value to.
value	A **System.String** containing the new value to add to the specified entry.

Behaviors

If the specified key already exists in the current instance, this method overwrites the existing values with the specified value. If the specified key does not exist in the current instance, this method creates a new entry using the specified key and the specified value.

Usage

Use the **System.Collections.Specialized.NameValueCollection.Add** method to add the new value to the existing list of values.

Exceptions

Exception	Condition
System.NotSupportedException	The current instance is read-only.

NameValueCollection.AllKeys Property

```
[ILASM]
.property class System.String[] AllKeys { public hidebysig
virtual specialname class System.String[] get_AllKeys() }

[C#]
public virtual string[] AllKeys { get; }
```

Summary

Gets all the keys in the current instance.

Property Value

A **System.String** array containing all the keys of the current instance. If the current instance is empty, the value of this property is an empty array.

Behaviors

This property is read-only.

Usage

The array returned by **System.Collections.Specialized.NameValueCollection.AllKeys** is cached for better performance and is automatically refreshed when the collection changes. A derived class can invalidate the cached version by calling **System.Collections.Specialized.NameValueCollection.InvalidateCachedArrays**, thereby forcing the array to be refreshed.

NameValueCollection.Item Property

```
[ILASM]
.property string Item[string name] { public hidebysig
specialname instance string get_Item(string name) public
hidebysig specialname instance void set_Item(string name,
string value) }

[C#]
public string this[string name] { get; set; }
```

Summary

Gets or sets the value in the current instance that is associated with the specified key.

Parameters

Parameter	Description
name	A **System.String** containing the key of the entry to locate.

Property Value

A **System.String** that contains the comma-separated list of values associated with the specified key. If *key* is not contained in the current instance, attempting to get it returns **null**, and attempting to set it creates a new entry using *key*.

Description

If the specified key already exists in the collection, setting this property overwrites the existing values with the specified value. If the specified key does not exist in the collection, setting this property creates a new entry using the specified key and the specified value.

[*Note:* This property provides the ability to access a specific element in the current instance using the following notation: myCollection[key].

To add the new value to the existing list of values, use the **System.Collections.Specialized.NameValueCollection.Add** method.]

Exceptions

Exception	Condition

System.ArgumentOutOfRangeException	*index* is outside the valid range of indexes for the collection.
System.NotSupportedException	The property is being set and the current instance is read-only. The property is being set, *key* is not contained in the current instance, and the current instance has a fixed size.

NameValueCollection.Item Property

[ILASM]

```
.property string Item[int32 index] { public hidebysig
specialname instance string get_Item(int32 index) }
```

[C#]

```
public string this[int index] { get; }
```

Summary

Gets the value in the current instance that is associated with the specified index.

Parameters

Parameter	Description
index	A **System.Int32** that specifies the zero-based index of the entry to locate in the current instance.

Property Value

A **System.String** that contains the comma-separated list of values at the specified index of the current instance.

Description

This property is read-only.

[*Note:* This property provides the ability to access a specific element in the collection by using the following syntax: `myCollection[index]`.

This property cannot be set. To set the value at a specified index, use `Item[GetKey(index)]`.]

Exceptions

Exception	Condition
System.ArgumentOutOfRangeException	*index* is outside the valid range of indices for the current instance.