System.Collections.Generic.Dictionary<TKey,TValue>.ValueCollection Class

```
[ILAsm]
.class sealed nested public serializable
System.Collections.Generic.Dictionary`2/ValueCollection<TKey,TValue>
extends System.Object implements
System.Collections.Generic.ICollection`1<!0>,
System.Collections.Generic.IEnumerable`1<!0>,
System.Collections.ICollection, System.Collections.IEnumerable

[C#]
public sealed class Dictionary<TKey,TValue>.ValueCollection:
ICollection<TKey>, IEnumerable<TKey>, ICollection, IEnumerable
```

Assembly Info:

- *Name:* mscorlib
- *Public Key:* [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00 00]
- *Version:* 2.0.x.x
- *Attributes:*
    - o CLSCompliantAttribute(true)

Implements:

- **System.Collections.ICollection**
- **System.Collections.IEnumerable**
- **System.Collections.Generic.ICollection<TKey>**
- **System.Collections.Generic.IEnumerable<TKey>**

Summary

Represents a read-only collection of values in a dictionary.

Inherits From: System.Object

Library: BCL

Thread Safety: Static members of this type are thread-safe. Any instance members are not guaranteed to be thread-safe.

Description

The `System.Collections.Generic.Dictionary<TKey,TValue>.Values` property returns an instance of this type, containing all the values in that dictionary. The order of the values in the value collection is unspecified, but it is the same order as the associated values in the key collection returned by the `System.Collections.Generic.Dictionary<TKey,TValue>.Keys` property.

If the underlying dictionary is modified, or the value of any key in that dictionary is modified, the behavior of the value collection is unspecified.

Dictionary<TKey,TValue>.ValueCollection (System.Collections.Generic.Dictionary<T Key,TValue>) Constructor

```
[ILAsm]
public rtspecialname specialname instance void .ctor(class
System.Collections.Generic.Dictionary`2<!2,!3> dictionary)


[C#]
public ValueCollection(Dictionary<TKey,TValue> dictionary)
```

Summary

Initializes a new value collection to reflect the values from the specified dictionary.

Parameters

Parameter	Description
dictionary	The dictionary whose values are to be reflected in the value collection.

Exceptions

Exception	Condition
System.ArgumentNullException	*dictionary* is `null`.

Dictionary<TKey,TValue>.ValueCollection. CopyTo(TKey[], System.Int32) Method

```
[ILAsm]
.method public hidebysig void CopyTo(!2[] array, int32 arrayIndex)


[C#]
public void CopyTo(TValue[] array, int arrayIndex)
```

Summary

Copies the elements of the value collection to a `System.Array`, starting at a particular index.

Parameters

Parameter	Description
array	A one-dimensional, zero-based `System.Array` that is the destination of the elements copied from the value collection.
arrayIndex	The zero-based index in *array* at which copying begins.

Description

The elements are copied into the array in the same order in which the enumerator iterates through the value collection.

Exceptions

Exception	Condition
System.ArgumentException	*array* is multidimensional. -or- *array* does not have zero-based indexing. -or- *arrayIndex* is greater than the length of *array*. -or- The number of elements in the value

	collection is greater than the available space from *arrayIndex* to the end of the destination *array*. -or- Some element of the value collection is not assignable to the element type of the array.
System.ArgumentNullException	*array* is `null`.
System.ArgumentOutOfRangeException	*arrayIndex* < 0.

Dictionary<TKey,TValue>.ValueCollection. GetEnumerator() Method

```
[ILAsm]
.method public hidebysig virtual class
System.Collections.Generic.Dictionary`2/ValueCollection/Enumerator<!
0,!1> GetEnumerator()


[C#]
public Dictionary<TKey,TValue>.ValueCollection.Enumerator
GetEnumerator()
```

Summary

Returns an enumerator that can be used to iterate over the value collection.

Return Value

An enumerator for the value collection.

Description

If the underlying dictionary is modified, or the value of any key in that dictionary is modified, the behavior of the enumerator is unspecified.

Usage

For a detailed description regarding the use of an enumerator, see
System.Collections.Generic.IEnumerator<TKey>.

Dictionary<TKey,TValue>.ValueCollection. System.Collections.Generic.ICollection<T Value>.Add(TValue) Method

```
[ILAsm]
.method private hidebysig virtual final void
System.Collections.Generic.ICollection`1<TValue>.Add(!1 item)


[C#]
void ICollection<TValue>.Add(TValue item)
```

Summary

This method is implemented to support the
`System.Collections.Generic.ICollection<TValue>` interface. This method
throws an exception of type `System.NotSupportedException`.

Dictionary<TKey,TValue>.ValueCollection. System.Collections.Generic.ICollection<T Value>.Clear() Method

```
[ILAsm]
.method private hidebysig virtual final void
System.Collections.Generic.ICollection`1<TValue>.Clear()


[C#]
void ICollection<TValue>.Clear()
```

Summary

This method is implemented to support the
`System.Collections.Generic.ICollection<TValue>` interface. This method
throws an exception of type `System.NotSupportedException`.

Dictionary<TKey,TValue>.ValueCollection. System.Collections.Generic.ICollection<T Value>.Contains(TValue) Method

```
[ILAsm]
.method private hidebysig virtual final bool
System.Collections.Generic.ICollection`1<TValue>.Contains(!1 item)


[C#]
bool ICollection<TValue>.Contains(TValue item)
```

Summary

This method is implemented to support the
System.Collections.Generic.ICollection<TValue> interface.

Dictionary<TKey,TValue>.ValueCollection. System.Collections.Generic.ICollection<T Value>.Remove(TValue) Method

```
[ILAsm]
.method private hidebysig virtual final bool
System.Collections.Generic.ICollection`1<TValue>.Remove(!1 item)


[C#]
bool ICollection<TValue>.Remove(TValue item)
```

Summary

This method is implemented to support the
System.Collections.Generic.ICollection<TValue> interface. This method
throws an exception of type System.NotSupportedException.

Dictionary<TKey,TValue>.ValueCollection. System.Collections.Generic.IEnumerable< TValue>.GetEnumerator() Method

```
[ILAsm]
.method private hidebysig virtual final class
System.Collections.Generic.IEnumerator`1<TValue>
System.Collections.Generic.IEnumerable`1<TValue>.GetEnumerator()


[C#]
IEnumerator<TValue> IEnumerable<TValue>.GetEnumerator()
```

Summary

This method is implemented to support the
`System.Collections.Generic.IEnumerable<TValue>` interface.

Description

If the underlying dictionary is modified, or the value of any key in that dictionary is modified, the behavior of the enumerator is unspecified.

Dictionary<TKey,TValue>.ValueCollection. System.Collections.ICollection.CopyTo(System.Array, System.Int32) Method

```
[ILAsm]
.method private hidebysig virtual final void
System.Collections.ICollection.CopyTo(class System.Array array,
int32 index)


[C#]
void ICollection.CopyTo(Array array, int index)
```

Summary

This method is implemented to support the `System.Collections.ICollection` interface.

Dictionary<TKey,TValue>.ValueCollection. System.Collections.IEnumerable.GetEnum erator() Method

```
[ILAsm]
.method private hidebysig virtual final class
System.Collections.IEnumerator
System.Collections.IEnumerable.GetEnumerator()


[C#]
IEnumerator IEnumerable.GetEnumerator()
```

Summary

This method is implemented to support the `System.Collections.IEnumerable` interface.

Description

If the underlying dictionary is modified, or the value of any key in that dictionary is modified, the behavior of the enumerator is unspecified.

Dictionary<TKey,TValue>.ValueCollection. Count Property

```
[ILAsm]
.property int32 Count { public hidebysig specialname instance int32
get_Count () }


[C#]
public int Count { get; }
```

Summary

Gets the number of elements contained in the value collection.

Property Value

The number of elements in the value collection.

Description

This property is read-only.

Dictionary<TKey,TValue>.ValueCollection. System.Collections.Generic.ICollection<T Value>.IsReadOnly Property

```
[ILAsm]
.property bool
System.Collections.Generic.ICollection`1<TValue>.IsReadOnly {
private hidebysig virtual final specialname bool get_IsReadOnly() }


[C#]
bool ICollection<TValue>.IsReadOnly { get; }
```

Summary

This read-only property is implemented to support the
System.Collections.Generic.ICollection<TValue> interface. This property
returns true.

Dictionary<TKey,TValue>.ValueCollection.System.Collections.ICollection.IsSynchronized Property

```
[ILAsm]
.property bool System.Collections.ICollection.IsSynchronized {
private hidebysig virtual final specialname bool
get_IsSynchronized() }

[C#]
bool ICollection.IsSynchronized { get; }
```

Summary

This read-only property is implemented to support the
`System.Collections.ICollection` interface. This property returns false.

Dictionary<TKey,TValue>.ValueCollection. System.Collections.ICollection.SyncRoot Property

```
[ILAsm]
.property object System.Collections.ICollection.SyncRoot { private
hidebysig virtual final specialname object get_SyncRoot() }


[C#]
object ICollection.SyncRoot { get; }
```

Summary

This property is implemented to support the `System.Collections.ICollection` interface.

Description

This read-only property returns the `SyncRoot` property of the underlying dictionary.