

System.ValueType Class

```
[ILAsm]
.class public abstract serializable ValueType extends System.Object

[C#]
public abstract class ValueType
```

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:*
 - CLSCompliantAttribute(true)

Summary

Provides support for value types. This class is the base class for all value types.

Inherits From: System.Object

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

[*Note:* Data types are separated into value types and reference types. Value types are either stack-allocated or allocated inline in a structure. Reference types are heap-allocated. Both reference and value types are derived from the ultimate base class `System.Object`. In cases where a value type needs to act like an object, a wrapper that makes the value type look like a reference object is allocated on the heap, and the value type's value is copied into it. The wrapper is marked so that the system knows that it contains a value type. This process is known as boxing, and the reverse process is known as unboxing. Boxing and unboxing allow any type to be treated as an object.

]

Example

In the following example, the number 3 is boxed as a `System.Int32`, and `System.Int32.ToString()` is called.

```
[C#]

using System;
class Boxer {
    public static void Main() {
```

```
1     Console.WriteLine("Value is {0}.", 3.ToString());
2     }
3 }
4 The output is
5
6 Value is 3.
7
8
```

ValueType() Constructor

```
[ILAsm]  
family rtspecialname specialname instance void .ctor()  
  
[C#]  
protected ValueType()
```

Summary

Constructs a new instance of the `System.ValueType` class.

ValueType.Equals(System.Object) Method

```
[ILAsm]  
.method public hidebysig virtual bool Equals(object obj)  
  
[C#]  
public override bool Equals(object obj)
```

Summary

Determines whether the current instance and a specified `System.Object` represent the same value.

Parameters

Parameter	Description
<i>obj</i>	The <code>System.Object</code> to compare the current instance to.

Return Value

`true` if *obj* and the current instance are of the same type and represent the same value; otherwise, `false`.

Description

[*Note:* This method overrides `System.Object.Equals`.]

1

2
3
4
5

6

7

8

9

10

11

12
13

14

15

16

1 **ValueType.ToString() Method**

```
2    [ILAsm]  
3    .method public hidebysig virtual string ToString()  
  
4    [C#]  
5    public override string ToString()
```

6 **Summary**

7 Returns a `System.String` representation of the fully-qualified name of the type of the
8 current instance.

9 **Return Value**

10 A `System.String` representation of the fully-qualified name of the type of the current
11 instance.

12 **Description**

13 [*Note:* This method overrides `System.Object.ToString`.
14
15 This method returns the `System.Type.FullName` property.
16
17]