

# System.Boolean Structure

```
[ILAsm]
.class public sequential sealed serializable Boolean extends
System.ValueType implements System.IComparable,
System.IComparable`1<bool>, System.IEquatable`1<bool>

[C#]
public struct Boolean: IComparable, IComparable<Boolean>,
IEquatable<Boolean>
```

## 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)

## Implements:

- **System.IComparable**
- **System.IComparable<System.Boolean>**
- **System.IEquatable<System.Boolean>**

## Summary

Represents a Boolean value.

## Inherits From: System.ValueType

**Library:** BCL

**Thread Safety:** This type is safe for multithreaded operations.

## Description

The `System.Boolean` value type represents the logical values `true` and `false`. The size of this type is 8 bits, the representation of `false` is all-bits-zero, and the representation of `true` is unspecified except that it shall have at least one bit set.

# Boolean.FalseString Field

```
[ILAsm]  
.field public static initOnly string FalseString
```

```
[C#]  
public static readonly string FalseString
```

## Summary

Contains a `System.String` representation of the logical value `false`.

## Description

This field is read-only.

The value of this `System.String` is "False".

# Boolean.TrueString Field

```
[ILAsm]  
.field public static initOnly string TrueString
```

```
[C#]  
public static readonly string TrueString
```

## Summary

Contains a `System.String` representation of the logical value `true`.

## Description

This field is read-only.

The value of this `System.String` is "True".

# Boolean.CompareTo(System.Boolean) Method

```
[ILAsm]  
.method public final hidebysig virtual int32 CompareTo(bool value)  
  
[C#]  
public int CompareTo(bool value)
```

## Summary

Returns the sort order of the current instance compared to the specified `System.Boolean`.

## Parameters

Parameter	Description
<i>value</i>	A <code>System.Boolean</code> to compare to the current instance.

## Return Value

The return value is a negative number, zero, or a positive number reflecting the sort order of the current instance as compared to *value*. For non-zero return values, the exact value returned by this method is unspecified. The following table defines the return value:

Return Value	Condition
A negative number	Current instance is false and <i>value</i> is true.
Zero	Current instance == <i>value</i> .
Any positive number	Current instance is true and <i>value</i> is false.

## Description

[*Note:* This method is implemented to support the `System.IComparable<Boolean>` interface.]

# Boolean.CompareTo(System.Object) Method

```
[ILAsm]  
.method public final hidebysig virtual int32 CompareTo(object obj)  
  
[C#]  
public int CompareTo(object obj)
```

## Summary

Returns the sort order of the current instance compared to the specified `System.Object`.

## Parameters

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

## Return Value

The return value is a negative number, zero, or a positive number reflecting the sort order of the current instance as compared to *obj*. For non-zero return values, the exact value returned by this method is unspecified. The following table defines the return value:

Return Value	Condition
A negative number	Current instance is false and <i>obj</i> is true.
Zero	Current instance == <i>obj</i> .
Any positive number	Current instance is true and <i>obj</i> is false, or <i>obj</i> is a null reference.

## Description

[*Note:* This method is implemented to support the `System.IComparable` interface.]

## Exceptions

Exception	Condition
<b>System.ArgumentException</b>	<i>obj</i> is not a <code>System.Boolean</code> and is not a null reference.

# Boolean.Equals(System.Boolean) Method

```
[ILAsm]  
.method public hidebysig virtual bool Equals(bool obj)
```

```
[C#]  
public override bool Equals(bool obj)
```

## Summary

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

## Parameters

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

## Return Value

true if *obj* has the same value as the current instance; otherwise, false.

[*Note:* This method is implemented to support the `System.IEquatable<Boolean>` interface.]

# Boolean.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 the specified `System.Object` represent the same type and value.

## Parameters

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

## Return Value

true if *obj* is a `System.Boolean` with the same value as the current instance. If *obj* is a null reference or is not an instance of `System.Boolean`, returns false.

## Description

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



# Boolean.GetHashCode() Method

```
[ILAsm]  
.method public hidebysig virtual int32 GetHashCode()  
  
[C#]  
public override int GetHashCode()
```

## Summary

Generates a hash code for the current instance.

## Return Value

A `System.Int32` value containing a hash code for the current instance.

## Description

The algorithm used to generate the hash code is unspecified.

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

# Boolean.Parse(System.String) Method

```
[ILAsm]  
.method public hidebysig static bool Parse(string value)
```

```
[C#]  
public static bool Parse(string value)
```

## Summary

Returns the specified `System.String` converted to a `System.Boolean` value.

## Parameters

Parameter	Description
<i>value</i>	A <code>System.String</code> containing the value to convert. The string is equivalent to either <code>System.Boolean.TrueString</code> or <code>System.Boolean.FalseString</code> , can contain leading and/or trailing whitespace, and is parsed in a case-insensitive manner.

## Return Value

true if *value* is equivalent to `System.Boolean.TrueString`; otherwise, false.

## Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	<i>value</i> is a null reference.
<b>System.FormatException</b>	<i>value</i> is not equivalent to either <code>System.Boolean.TrueString</code> or <code>System.Boolean.FalseString</code> .

## Example

The following example demonstrates the `System.Boolean.Parse` method.

```
[C#]  
  
using System;  
public class BoolParse {  
    public static void Main() {  
        Boolean b = Boolean.Parse(" true ");  
        Console.WriteLine("\n true  \n parses to \"{0}\".", b);  
    }  
}
```

The output is

```
" true " parses to "True".
```

# Boolean.ToString(System.IFormatProvider) Method

```
[ILAsm]  
.method public final hidebysig virtual string ToString(class  
System.IFormatProvider provider)  
  
[C#]  
public string ToString(IFormatProvider provider)
```

## Summary

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

## Parameters

Parameter	Description
<i>provider</i>	(Reserved) A System.IFormatProvider object.

## Return Value

System.Boolean.TrueString if the value of this instance is true, or  
System.Boolean.FalseString if the value of this instance is false.

## Description

The *provider* parameter is reserved. It does not participate in the execution of this method.

# Boolean.ToString() Method

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

## Summary

Returns a `System.String` representation of the value of the current instance.

## Return Value

`System.Boolean.FalseString` if the value of the current instance is false;  
otherwise, `System.Boolean.TrueString`.

## Description

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