

1 System.IntPtr Structure

```
2 [ILAsm]  
3 .class public sequential sealed serializable IntPtr extends  
4 System.ValueType  
5 [C#]  
6 public struct IntPtr
```

7 Assembly Info:

- 8 • *Name:* mscorlib
- 9 • *Public Key:* [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00]
- 10 • *Version:* 2.0.x.x
- 11 • *Attributes:*
 - 12 ○ CLSCompliantAttribute(true)

13 Summary

14 An implementation-specific type that is used to represent a pointer or a handle.

15 Inherits From: System.ValueType

16

17 **Library:** RuntimeInfrastructure

18

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

20

21 Description

22 The `System.IntPtr` type is designed to be an implementation-sized pointer. An instance
23 of this type is expected to be the size of a native `int` for the current implementation.

24

25 For more information on the native `int` type, see Partition II of the CLI Specification.

26

27 [*Note:* The `System.IntPtr` type provides CLS-compliant pointer functionality.

28

29 `System.IntPtr` instances can also be used to hold handles.

30

31 The `System.IntPtr` type is CLS-compliant while the `System.UIntPtr` type is not. The
32 `System.UIntPtr` type is provided mostly to maintain architectural symmetry with the
33 `System.IntPtr` type.

34

35]

36

1 IntPtr(System.Int32) Constructor

```
2 [ILAsm]  
3 public rtspecialname specialname instance void .ctor(int32 value)  
4 [C#]  
5 public IntPtr(int value)
```

6 Summary

7 Constructs a new `System.IntPtr` structure using the specified `System.Int32` containing
8 a pointer or a handle.

9 Parameters

Parameter	Description
<i>value</i>	A <code>System.Int32</code> containing a pointer or a handle.

10

11

1 IntPtr(System.Int64) Constructor

```
2 [ILAsm]  
3 public rtspecialname specialname instance void .ctor(int64 value)  
4 [C#]  
5 public IntPtr(long value)
```

6 Summary

7 Constructs a new `System.IntPtr` structure using the specified `System.Int64` containing
8 a pointer or a handle.

9 Parameters

Parameter	Description
<i>value</i>	A <code>System.Int64</code> containing a pointer or a handle.

10

11 Exceptions

Exception	Condition
System.OverflowException	The current platform is a 32-bit platform and the value of the current instance is greater than <code>System.Int32.MaxValue</code> or less than <code>System.Int32.MinValue</code> .

12

13

1 IntPtr.Zero Field

```
2 [ILAsm]  
3 .field public static initOnly valuetype System.IntPtr Zero  
4 [C#]  
5 public static readonly IntPtr Zero
```

6 Summary

7 Represents a pointer or handle that has been initialized as zero.

8 Description

9 *[Note:* The value of this field is not `null`, but is instead a pointer which has been
10 assigned the value zero. Use this field to efficiently determine whether an instance of
11 `System.IntPtr` has been set to a value other than zero. For example, if *ip* is a
12 `System.IntPtr` instance, using *ip* `!= IntPtr.Zero` is more efficient than *ip* `!= new`
13 *IntPtr(0)* to test if *ip* has been set to a value other than zero.
14
15]

16

1 IntPtr.Equals(System.Object) Method

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

6 Summary

7 Determines whether the current instance and the specified `System.Object` represent the
8 same type and value.

9 Parameters

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

10

11 Return Value

12 `true` if *obj* is a `System.IntPtr` instance and has the same value as the current instance.
13 If *obj* is a null reference or is not an instance of `System.IntPtr`, returns `false`.

14 Description

15 [Note: The method overrides `System.Object.Equals`.]
16
17

18

1 IntPtr.GetHashCode() Method

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

6 Summary

7 Generates a hash code for the current instance.

8 Return Value

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

10 Description

11 [*Note:* The algorithm used to generate the hash code is unspecified.]
12
13

14
15 [*Note:* This method overrides `System.Object.GetHashCode.`]
16
17

18

1 IntPtr.op_Equality(System.IntPtr, 2 System.IntPtr) Method

```
3 [ILAsm]  
4 .method public hidebysig static specialname bool op_Equality(valuetype  
5 System.IntPtr value1, valuetype System.IntPtr value2)  
  
6 [C#]  
7 public static bool operator ==(IntPtr value1, IntPtr value2)
```

8 Summary

9 Determines whether the two specified instances of `System.IntPtr` represent the same
10 value.

11 Parameters

Parameter	Description
<i>value1</i>	The first <code>System.IntPtr</code> to compare for equality.
<i>value2</i>	The second <code>System.IntPtr</code> to compare for equality.

12 13 Return Value

14 true if *value1* represents the same value as *value2*; otherwise, false.

15

1 IntPtr.op_Inequality(System.IntPtr, 2 System.IntPtr) Method

```
3 [ILAsm]  
4 .method public hidebysig static specialname bool op_Inequality(valuetype  
5 System.IntPtr value1, valuetype System.IntPtr value2)  
  
6 [C#]  
7 public static bool operator !=(IntPtr value1, IntPtr value2)
```

8 Summary

9 Determines whether the two specified instances of `System.IntPtr` represent different
10 values.

11 Parameters

Parameter	Description
<i>value1</i>	The first <code>System.IntPtr</code> to compare for inequality.
<i>value2</i>	The second <code>System.IntPtr</code> to compare for inequality.

12 13 Return Value

14 true if *value1* represents a different value than *value2*; otherwise, false.

15

1 IntPtr.ToInt32() Method

```
2 [ILAsm]  
3 .method public hidebysig instance int32 ToInt32()  
4 [C#]  
5 public int ToInt32()
```

6 Summary

7 Converts the value of the current instance to a `System.Int32`.

8 Return Value

9 A `System.Int32` containing the same value as the current instance.

10 Exceptions

Exception	Condition
System.OverflowException	The current platform is not a 32-bit platform and the value of the current instance is greater than <code>System.Int32.MaxValue</code> or less than <code>System.Int32.MinValue</code> .

11

12

1 IntPtr.ToInt64() Method

```
2 [ILAsm]  
3 .method public hidebysig instance int64 ToInt64()  
4 [C#]  
5 public long ToInt64()
```

6 Summary

7 Converts the value of the current instance to a System.Int64.

8 Return Value

9 A System.Int64 containing the same value as the current instance.

10

1 IntPtr.ToPointer() Method

```
2 [ILAsm]  
3 .method public hidebysig instance class System.Void* ToPointer()  
4 [C#]  
5 unsafe public void* ToPointer()
```

6 Summary

7 Converts the value of the current instance to a pointer to `void`.

8 Type Attributes:

- 9 • CLSCompliantAttribute(false)

10 Return Value

11 A pointer to `void`.

12 Description

13 This member is not CLS-compliant. For a CLS-compliant alternative, use
14 `System.IntPtr.ToInt32`.

15
16 [*Note:* A pointer to `void` points to memory containing data of an unspecified type.]
17
18

19

1 IntPtr.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 value of the current instance.

8 Return Value

9 A `System.String` representation of the current instance.

10 Description

11 [*Note:* If `System.IntPtr.Size` for the current instance is 4, `System.IntPtr.ToString`
12 is equivalent to `System.Int32.ToString()`; otherwise, this method is equivalent to
13 `System.Int64.ToString()`.

14 This method overrides `System.Object.ToString`.

15]
16]
17]

18

1 IntPtr.Size Property

```
2 [ILAsm]  
3 .property int32 Size { public hidebysig static specialname int32  
4 get_Size() }  
5 [C#]  
6 public static int Size { get; }
```

7 Summary

8 Gets the size in bytes of a pointer or a handle for the current implementation.

9 Property Value

10 A `System.Int32` containing the number of bytes of a pointer or handle for the current
11 implementation. The value of this property is equal to the number of bytes contained by
12 the `native int` type in the current implementation.

13 Description

14 This property is read-only.

15
16 For more information on the `native int` type, see Partition II of the CLI Specification.

17