

OperatingSystem Class

Reference

Definition

Namespace: [System](#)

Assembly: System.Runtime.dll

Source: [OperatingSystem.cs](#) [↗](#)

Represents information about an operating system, such as the version and platform identifier. This class cannot be inherited.

C#

```
public sealed class OperatingSystem : ICloneable,  
System.Runtime.Serialization.ISerializable
```

Inheritance [Object](#) → [OperatingSystem](#)

Implements [ICloneable](#) , [ISerializable](#)

Examples

The following code example uses the [OperatingSystem](#) object to display information about the runtime operating system.

C#

```
using System;  
  
public class Example  
{  
    public static void Main()  
    {  
        var os = Environment.OSVersion;  
        Console.WriteLine("Current OS Information:\n");  
        Console.WriteLine("Platform: {0:G}", os.Platform);  
        Console.WriteLine("Version String: {0}", os.VersionString);  
        Console.WriteLine("Version Information:");  
        Console.WriteLine("    Major: {0}", os.Version.Major);  
        Console.WriteLine("    Minor: {0}", os.Version.Minor);  
        Console.WriteLine("Service Pack: '{0}'", os.ServicePack);  
    }  
}
```

```

}
// If run on a Windows 8.1 system, the example displays output like
// the following:
//      Current OS Information:
//
//      Platform: Win32NT
//      Version String: Microsoft Windows NT 6.2.9200.0
//      Version Information:
//          Major: 6
//          Minor: 2
//      Service Pack: ''
// If run on a Windows 7 system, the example displays output like the
// following:
//      Current OS Information:
//
//      Platform: Win32NT
//      Version String: Microsoft Windows NT 6.1.7601 Service Pack 1
//      Version Information:
//          Major: 6
//          Minor: 1
//      Service Pack: 'Service Pack 1'

```

Remarks

The [OperatingSystem](#) class contains information about an operating system.

For information about the current runtime operating system, retrieve the [OperatingSystem](#) object returned by the [Environment.OSVersion](#) property. For a list of Windows operating system versions and their corresponding version numbers returned by the [Version](#) and [VersionString](#) properties and the [ToString](#) method, see [Operating System Version](#).

By design, the [OperatingSystem](#) class is not a general purpose means of describing an operating system, and you cannot derive a more inclusive type from the [OperatingSystem](#) class. If you need a type to contain other information about an operating system, create your own type, then include a field of type [OperatingSystem](#) and any additional fields, properties, or methods that you require.

Constructors

[Expand table](#)

OperatingSystem(PlatformID, Version)	Initializes a new instance of the OperatingSystem class, using the specified platform identifier value and version object.
--	--

Properties

 Expand table

Platform	Gets a PlatformID enumeration value that identifies the operating system platform.
Service Pack	Gets the service pack version represented by this OperatingSystem object.
Version	Gets a Version object that identifies the operating system.
Version String	Gets the concatenated string representation of the platform identifier, version, and service pack that are currently installed on the operating system.

Methods

 Expand table

Clone()	Creates an OperatingSystem object that is identical to this instance.
Equals(Object)	Determines whether the specified object is equal to the current object. (Inherited from Object)
GetHashCode()	Serves as the default hash function. (Inherited from Object)
GetObjectData(Serialization Info, StreamingContext)	Obsolete. Populates a SerializationInfo object with the data necessary to deserialize this instance.
GetType()	Gets the Type of the current instance. (Inherited from Object)
IsAndroid()	Indicates whether the current application is running on Android.
IsAndroidVersion AtLeast(Int32, Int32, Int32, Int32)	Checks if the Android version (returned by the Linux command <code>uname</code>) is greater than or equal to the specified version. This method can be used to guard APIs that were added in the specified version.
IsBrowser()	Indicates whether the current application is running as WASM in a browser.
IsFreeBSD()	Indicates whether the current application is running on FreeBSD.

<code>IsFreeBSDVersionAtLeast(Int32, Int32, Int32, Int32)</code>	Checks if the FreeBSD version (returned by the Linux command <code>uname</code>) is greater than or equal to the specified version. This method can be used to guard APIs that were added in the specified version.
<code>IsIOS()</code>	Indicates whether the current application is running on iOS or MacCatalyst.
<code>IsIOSVersionAtLeast(Int32, Int32, Int32)</code>	Checks if the iOS/MacCatalyst version (returned by <code>libobjc.get_operatingSystemVersion</code>) is greater than or equal to the specified version. This method can be used to guard APIs that were added in the specified iOS version.
<code>IsLinux()</code>	Indicates whether the current application is running on Linux.
<code>IsMacCatalyst()</code>	Indicates whether the current application is running on Mac Catalyst.
<code>IsMacCatalystVersionAtLeast(Int32, Int32, Int32)</code>	Check for the Mac Catalyst version (iOS version as presented in Apple documentation) with a \leq version comparison. Used to guard APIs that were added in the given Mac Catalyst release.
<code>IsMacOS()</code>	Indicates whether the current application is running on macOS.
<code>IsMacOSVersionAtLeast(Int32, Int32, Int32)</code>	Checks if the macOS version (returned by <code>libobjc.get_operatingSystemVersion</code>) is greater than or equal to the specified version. This method can be used to guard APIs that were added in the specified macOS version.
<code>IsOSPlatform(String)</code>	Indicates whether the current application is running on the specified platform.
<code>IsOSPlatformVersionAtLeast(String, Int32, Int32, Int32, Int32)</code>	Checks if the operating system version is greater than or equal to the specified platform version. This method can be used to guard APIs that were added in the specified OS version.
<code>IsTvOS()</code>	Indicates whether the current application is running on tvOS.
<code>IsTvOSVersionAtLeast(Int32, Int32, Int32)</code>	Checks if the tvOS version (returned by <code>libobjc.get_operatingSystemVersion</code>) is greater than or equal to the specified version. This method can be used to guard APIs that were added in the specified tvOS version.
<code>IsWasi()</code>	Indicates whether the current application is running as WASI.
<code>IsWatchOS()</code>	Indicates whether the current application is running on watchOS.
<code>IsWatchOSVersionAtLeast(Int32, Int32, Int32)</code>	Checks if the watchOS version (returned by <code>libobjc.get_operatingSystemVersion</code>) is greater than or equal to the specified version. This method can be used to guard APIs that were added in the specified watchOS version.

IsWindows()	Indicates whether the current application is running on Windows.
IsWindowsVersionAtLeast(Int32, Int32, Int32, Int32)	Checks if the Windows version (returned by <code>RtlGetVersion</code>) is greater than or equal to the specified version. This method can be used to guard APIs that were added in the specified Windows version.
MemberwiseClone()	Creates a shallow copy of the current Object . (Inherited from Object)
ToString()	Converts the value of this OperatingSystem object to its equivalent string representation.

Applies to

Product	Versions
.NET	Core 2.0, Core 2.1, Core 2.2, Core 3.0, Core 3.1, 5, 6, 7, 8, 9
.NET Framework	1.1, 2.0, 3.0, 3.5, 4.0, 4.5, 4.5.1, 4.5.2, 4.6, 4.6.1, 4.6.2, 4.7, 4.7.1, 4.7.2, 4.8, 4.8.1
.NET Standard	2.0, 2.1

Collaborate with us on GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).

.NET

.NET feedback

.NET is an open source project.
Select a link to provide feedback:

 [Open a documentation issue](#)

 [Provide product feedback](#)