IDisposable Interface

Namespace: System

Assembly: System.Runtime.dll

Provides a mechanism for releasing unmanaged resources.

In this article

C# Copy

```
public interface IDisposable
```

Derived DbLinq.Util.BaseLock
Microsoft.Build.Debugging.DebuggerManager.IslandThread
Microsoft.Build.Evaluation.ProjectCollection
Microsoft.Extensions.Caching.Memory.ICacheEntry
Microsoft.Extensions.Caching.Memory.IMemoryCache
Microsoft.Extensions.Caching.Memory.MemoryCache
Microsoft.Extensions.Caching.Redis.RedisCache
Microsoft.Extensions.Caching.StackExchangeRedis.RedisCache
Microsoft.Extensions.DependencyInjection.IServiceScope
Microsoft.Extensions.DependencyInjection.ServiceProvider
Microsoft.Extensions.DependencyModel.DependencyContextJsonReader
Microsoft.Extensions.DependencyModel.IDependencyContextReader
Microsoft.Extensions.FileProviders.PhysicalFileProvider
Microsoft.Extensions.FileProviders.Physical.PhysicalFilesWatcher
Microsoft.Extensions.Hosting.BackgroundService
Microsoft.Extensions.Hosting.IHost
Microsoft.Extensions.Hosting.Systemd.SystemdLifetime

Microsoft.Extensions.Logging.ILoggerFactory
Microsoft.Extensions.Logging.ILoggerProvider
Microsoft.Extensions.Logging.LoggerFactory
Microsoft.Extensions.Logging.Abstractions.NullLoggerFactory
Microsoft.Extensions.Logging.Abstractions.NullLoggerProvider
Microsoft.Extensions.Logging.AzureAppServices.BatchingLoggerProvider
Microsoft.Extensions.Logging.Console.ConsoleLoggerProvider
Microsoft.Extensions.Logging.Debug.DebugLoggerProvider
Microsoft.Extensions.Logging.EventLog.EventLogLoggerProvider
Microsoft.Extensions.Logging.EventSource.EventSourceLoggerProvider
Microsoft.Extensions.Logging.TraceSource.TraceSourceLoggerProvider
Microsoft.Extensions.Logging.TraceSource.TraceSourceScope
Microsoft.Extensions.Primitives.StringTokenizer.Enumerator
Microsoft.Extensions.Primitives.StringValues.Enumerator
Microsoft.VisualBasic.Compatibility.VB6.BaseDataEnvironment
Microsoft.VisualBasic.Compatibility.VB6.MBinding
Microsoft.VisualBasic.Compatibility.VB6.MBindingCollection
Microsoft.VisualBasic.FileIO.TextFieldParser
Microsoft.VisualC.StlClr.DequeEnumerator<TValue>
Microsoft.VisualC.StlClr.HashEnumerator<TKey,TValue>
Microsoft.VisualC.StlClr.ListEnumerator<TValue>
Microsoft.VisualC.StlClr.TreeEnumerator<TKey,TValue>
Microsoft.VisualC.StlClr.VectorEnumerator<TValue>
Microsoft.Win32.RegistryKey
Mono.Security.Interface.IMonoSslStream
Mono.Security.Protocol.Ntlm.ChallengeResponse
System.ActivationContext
System.ArraySegment<T>.Enumerator
System.CharEnumerator
System.Activities.WorkflowDataContext
System.Activities.Debugger.StateManager
System.Activities.Presentation.EditingContext
System.Activities.Presentation.Model.ModelEditingScope
System.Activities.Statements.DurableTimerExtension
System.AddIn.Contract.IEnumeratorContract<T>
System.AddIn.Pipeline.ContractHandle
System.Buffers.IMemoryOwner<T>
System.Buffers.MemoryHandle
System.Buffers.MemoryManager<T>

System.Buffers.MemoryPool<T>
System.CodeDom.Compiler.TempFileCollection
System.Collections.Concurrent.BlockingCollection<T>
System.Collections.Generic.Dictionary<TKey,TValue>.Enumerator
System.Collections.Generic.Dictionary<TKey,TValue>.KeyCollection.Enumerator
System.Collections.Generic.Dictionary<TKey,TValue>.ValueCollection.Enumerator
System.Collections.Generic.HashSet<T>.Enumerator
System.Collections.Generic.IEnumerator<T>
System.Collections.Generic.LinkedList<T>.Enumerator
System.Collections.Generic.List<T>.Enumerator
System.Collections.Generic.Queue<T>.Enumerator
System.Collections.Generic.SortedDictionary<TKey,TValue>.Enumerator
System.Collections.Generic.SortedDictionary<TKey,TValue>.KeyCollection.
Enumerator
System.Collections.Generic.SortedDictionary<TKey,TValue>.ValueCollection.
Enumerator
System.Collections.Generic.SortedSet<T>.Enumerator
System.Collections.Generic.Stack<T>.Enumerator
System.Collections.Immutable.ImmutableDictionary<TKey,TValue>.Enumerator
System.Collections.Immutable.ImmutableHashSet<T>.Enumerator
System.Collections.Immutable.ImmutableList<T>.Enumerator
System.Collections.Immutable.ImmutableSorted
Dictionary<TKey,TValue>.Enumerator
System.Collections.Immutable.ImmutableSortedSet<T>.Enumerator
System.ComponentModel.Component
System.ComponentModel.Container
System.ComponentModel.EventHandlerList
System.ComponentModel.IComponent
System.ComponentModel.IContainer
System.ComponentModel.INestedContainer
System.ComponentModel.License
System.ComponentModel.MarshalByValueComponent
System.ComponentModel.NestedContainer
System.ComponentModel.Composition.ExportLifetimeContext<T>
System.ComponentModel.Composition.Hosting.AggregateExportProvider
System.ComponentModel.Composition.Hosting.AtomicComposition
System.ComponentModel.Composition.Hosting.CatalogExportProvider
System.ComponentModel.Composition.Hosting.ComposablePartExportProvider
System.ComponentModel.Composition.Hosting.CompositionContainer

System.ComponentModel.Composition.Hosting.CompositionService
System.ComponentModel.Composition.Hosting.ImportEngine
System.ComponentModel.Composition.Primitives.ComposablePartCatalog
System.ComponentModel.Design.ComponentDesigner
System.ComponentModel.Design.DesignerActionService
System.ComponentModel.Design.DesignerActionUIService
System.ComponentModel.Design.DesignerTransaction
System.ComponentModel.Design.DesignSurface
System.ComponentModel.Design.DesignSurfaceManager
System.ComponentModel.Design.IDesigner
System.ComponentModel.Design.InheritanceService
System.ComponentModel.Design.IRootDesigner
System.ComponentModel.Design.ITreeDesigner
System.ComponentModel.Design.LocalizationExtenderProvider
System.ComponentModel.Design.MenuCommandService
System.ComponentModel.Design.ServiceContainer
System.ComponentModel.Design.UndoEngine
System.ComponentModel.Design.Serialization.CodeDomLocalizationProvider
System.ComponentModel.Design.Serialization.SerializationStore
System.Composition.Export<T>
System.Composition.Hosting.CompositionHost
System.Composition.Hosting.Core.CompositionOperation
System.Composition.Hosting.Core.LifetimeContext
System.Data.IDataReader
System.Data.IDbCommand
System.Data.IDbConnection
System.Data.IDbTransaction
System.Data.Common.DbCommand
System.Data.Common.DbConnection
System.Data.Common.DbDataReader
System.Data.Common.DbTransaction
System.Data.Linq.DataContext
System.Data.Linq.IExecuteResult
System.Data.Linq.IMultipleResults
System.Data.Linq.ISingleResult<T>
System.Data.Linq.SqlClient.SqlProvider
System.Data.Metadata.Edm.ReadOnlyMetadataCollection<T>.Enumerator
System.Data.Objects.ObjectContext
System.Data.Objects.ObjectResult

System.Data.OleDb.OleDbCommand
System.Data.OleDb.OleDbConnection
System.Data.OracleClient.OracleBFile
System.Data.OracleClient.OracleLob
System.Data.Services.Client.DataServiceStreamResponse
System.Data.SqlClient.SqlBulkCopy
System.Data.SqlClient.SqlDataReader
System.Deployment.Application.InPlaceHostingManager
System.Device.Location.GeoCoordinateWatcher
System.Diagnostics.ActivityListener
System.Diagnostics.ActivitySource
System.Diagnostics.ActivityTagsCollection.Enumerator
System.Diagnostics.DiagnosticListener
System.Diagnostics.TraceListener
System.Diagnostics.Eventing.EventProvider
System.Diagnostics.Eventing.Reader.EventLogConfiguration
System.Diagnostics.Eventing.Reader.EventLogPropertySelector
System.Diagnostics.Eventing.Reader.EventLogReader
System.Diagnostics.Eventing.Reader.EventLogSession
System.Diagnostics.Eventing.Reader.EventLogWatcher
System.Diagnostics.Eventing.Reader.EventRecord
System.Diagnostics.Eventing.Reader.ProviderMetadata
System.Diagnostics.PerformanceData.CounterSet
System.Diagnostics.PerformanceData.CounterSetInstance
System.Diagnostics.PerformanceData.CounterSetInstanceCounterDataSet
System.Diagnostics.Tracing.DiagnosticCounter
System.Diagnostics.Tracing.EventListener
System.Diagnostics.Tracing.EventSource
System.DirectoryServices.SearchResultCollection
System.DirectoryServices.AccountManagement.Principal
System.DirectoryServices.AccountManagement.PrincipalContext
System.DirectoryServices.AccountManagement.PrincipalSearcher
System.DirectoryServices.AccountManagement.PrincipalSearchResult<T>
System.DirectoryServices.ActiveDirectory.ActiveDirectoryInterSiteTransport
System.DirectoryServices.ActiveDirectory.ActiveDirectoryPartition
System.DirectoryServices.ActiveDirectory.ActiveDirectorySchemaClass
System.DirectoryServices.ActiveDirectory.ActiveDirectorySchemaProperty
System.DirectoryServices.ActiveDirectory.ActiveDirectorySite
System.DirectoryServices.ActiveDirectory.ActiveDirectorySiteLink

System.DirectoryServices.ActiveDirectory.ActiveDirectorySiteLinkBridge
System.DirectoryServices.ActiveDirectory.ActiveDirectorySubnet
System.DirectoryServices.ActiveDirectory.DirectoryServer
System.DirectoryServices.ActiveDirectory.Forest
System.DirectoryServices.ActiveDirectory.ReplicationConnection
System.DirectoryServices.Protocols.LdapConnection
System.Drawing.Brush
System.Drawing.BufferedGraphics
System.Drawing.BufferedGraphicsContext
System.Drawing.Font
System.Drawing.FontConverter.FontNameConverter
System.Drawing.FontFamily
System.Drawing.Graphics
System.Drawing.Icon
System.Drawing.IDeviceContext
System.Drawing.Image
System.Drawing.Pen
System.Drawing.Region
System.Drawing.StringFormat
System.Drawing.Drawing2D.CustomLineCap
System.Drawing.Drawing2D.GraphicsPath
System.Drawing.Drawing2D.GraphicsPathIterator
System.Drawing.Drawing2D.Matrix
System.Drawing.Imaging.EncoderParameter
System.Drawing.Imaging.EncoderParameters
System.Drawing.Imaging.ImageAttributes
System.Drawing.Text.FontCollection
System.EnterpriseServices.ServicedComponent
System.Formats.Asn1.AsnWriter.Scope
System.IdentityModel.AsyncResult
System.IdentityModel.Claims.WindowsClaimSet
System.IdentityModel.Claims.X509CertificateClaimSet
System.IdentityModel.Selectors.X509SecurityTokenProvider
System.IdentityModel.Tokens.X509SecurityToken
System.IO.BinaryReader
System.IO.BinaryWriter
System.IO.Stream
System.IO.TextReader
System.IO.TextWriter

System.IO.UnmanagedMemoryAccessor
System.IO.Compression.BrotliDecoder
System.IO.Compression.BrotliEncoder
System.IO.Compression.ZipArchive
System.IO.Enumeration.FileSystemEnumerator<TResult>
System.IO.IsolatedStorage.IsolatedStorageFile
System.IO.Log.FileRecordSequence
System.IO.Log.IRecordSequence
System.IO.Log.LogRecord
System.IO.Log.LogRecordSequence
System.IO.Log.LogStore
System.IO.MemoryMappedFiles.MemoryMappedFile
System.IO.Packaging.EncryptedPackageEnvelope
System.IO.Packaging.Package
System.IO.Packaging.PackageProperties
System.Management.ManagementObjectCollection.ManagementObject
Enumerator
System.Management.ManagementObjectCollection
System.Messaging.Cursor
System.Messaging.MessageEnumerator
System.Messaging.MessageQueueEnumerator
System.Messaging.MessageQueueTransaction
System.Messaging.SecurityContext
System.Net.FtpWebResponse
System.Net.HttpListener
System.Net.HttpListenerResponse
System.Net.Http.HttpContent
System.Net.Http.HttpMessageHandler
System.Net.Http.HttpMessageInvoker
System.Net.Http.HttpRequestMessage
System.Net.Http.HttpResponseMessage
System.Net.Mail.AlternateViewCollection
System.Net.Mail.AttachmentBase
System.Net.Mail.AttachmentCollection
System.Net.Mail.LinkedResourceCollection
System.Net.Mail.MailMessage
System.Net.Mail.SmtpClient
System.Net.PeerToPeer.PeerNameRegistration
System.Net.PeerToPeer.Collaboration.ContactManager

System.Net.PeerToPeer.Collaboration.Peer
System.Net.PeerToPeer.Collaboration.PeerApplication
System.Net.PeerToPeer.Collaboration.PeerEndPoint
System.Net.PeerToPeer.Collaboration.PeerObject
System.Net.Sockets.Socket
System.Net.Sockets.SocketAsyncEventArgs
System.Net.Sockets.TcpClient
System.Net.Sockets.UdpAnySourceMulticastClient
System.Net.Sockets.UdpClient
System.Net.Sockets.UdpSingleSourceMulticastClient
System.Net.WebSockets.WebSocket
System.Printing.PrintJobInfoCollection
System.Printing.PrintQueueCollection
System.Printing.PrintSystemObject
System.Printing.PrintSystemObjectPropertiesChangedEventArgs
System.Printing.PrintSystemObjectPropertyChangedEventArgs
System.Printing.PrintSystemObjects
System.Printing.IndexedProperties.PrintProperty
System.Printing.IndexedProperties.PrintPropertyDictionary
System.Printing.Interop.PrintTicketConverter
System.Reflection.MetadataLoadContext
System.Reflection.Metadata.AssemblyFileHandleCollection.Enumerator
System.Reflection.Metadata.AssemblyReferenceHandleCollection.Enumerator
System.Reflection.Metadata.BlobBuilder.Blobs
System.Reflection.Metadata.CustomAttributeHandleCollection.Enumerator
System.Reflection.Metadata.CustomDebugInformationHandleCollection.
Enumerator
System.Reflection.Metadata.DeclarativeSecurityAttributeHandleCollection.
Enumerator
System.Reflection.Metadata.DocumentHandleCollection.Enumerator
System.Reflection.Metadata.EventDefinitionHandleCollection.Enumerator
System.Reflection.Metadata.ExportedTypeHandleCollection.Enumerator
System.Reflection.Metadata.FieldDefinitionHandleCollection.Enumerator
System.Reflection.Metadata.GenericParameterConstraintHandleCollection.
Enumerator
System.Reflection.Metadata.GenericParameterHandleCollection.Enumerator
System.Reflection.Metadata.ImportDefinitionCollection.Enumerator
System.Reflection.Metadata.ImportScopeCollection.Enumerator
System.Reflection.Metadata.InterfaceImplementationHandleCollection.Enumerator

System.Reflection.Metadata.LocalConstantHandleCollection.Enumerator
System.Reflection.Metadata.LocalScopeHandleCollection.ChildrenEnumerator
System.Reflection.Metadata.LocalScopeHandleCollection.Enumerator
System.Reflection.Metadata.LocalVariableHandleCollection.Enumerator
System.Reflection.Metadata.ManifestResourceHandleCollection.Enumerator
System.Reflection.Metadata.MemberReferenceHandleCollection.Enumerator
System.Reflection.Metadata.MetadataReaderProvider
System.Reflection.Metadata.MethodDebugInformationHandleCollection.
Enumerator
System.Reflection.Metadata.MethodDefinitionHandleCollection.Enumerator
System.Reflection.Metadata.MethodImplementationHandleCollection.Enumerator
System.Reflection.Metadata.ParameterHandleCollection.Enumerator
System.Reflection.Metadata.PropertyDefinitionHandleCollection.Enumerator
System.Reflection.Metadata.SequencePointCollection.Enumerator
System.Reflection.Metadata.TypeDefinitionHandleCollection.Enumerator
System.Reflection.Metadata.TypeReferenceHandleCollection.Enumerator
System.Reflection.PortableExecutable.PEReader
System.Resources.IResourceReader
System.Resources.IResourceWriter
System.Resources.ResourceReader
System.Resources.ResourceSet
System.Resources.ResourceWriter
System.Resources.ResXResourceReader
System.Resources.ResXResourceWriter
System.Resources.Extensions.DeserializingResourceReader
System.Resources.Extensions.PreserializedResourceWriter
System.Runtime.MemoryFailPoint
System.Runtime.Caching.ChangeMonitor
System.Runtime.Caching.MemoryCache
System.Runtime.InteropServices.CriticalHandle
System.Runtime.InteropServices.SafeHandle
System.Runtime.Loader.AssemblyLoadContext.ContextualReflectionScope
System.Security.SecureString
System.Security.SecurityContext
System.Security.Cryptography.AesCcm
System.Security.Cryptography.AesGcm
System.Security.Cryptography.AsymmetricAlgorithm
System.Security.Cryptography.CngKey
System.Security.Cryptography.CryptoAPITransform

System.Security.Cryptography.CryptoStream
System.Security.Cryptography.DeriveBytes
System.Security.Cryptography.ECDiffieHellmanPublicKey
System.Security.Cryptography.FromBase64Transform
System.Security.Cryptography.HashAlgorithm
System.Security.Cryptography.ICryptoTransform
System.Security.Cryptography.IncrementalHash
System.Security.Cryptography.RandomNumberGenerator
System.Security.Cryptography.RijndaelManagedTransform
System.Security.Cryptography.SymmetricAlgorithm
System.Security.Cryptography.ToBase64Transform
System.Security.Cryptography.X509Certificates.X509Certificate
System.Security.Cryptography.X509Certificates.X509Chain
System.Security.Cryptography.X509Certificates.X509Store
System.Security.Principal.WindowsIdentity
System.Security.RightsManagement.CryptoProvider
System.Security.RightsManagement.SecureEnvironment
System.ServiceModel.ChannelFactory
System.ServiceModel.ClientBase<TChannel>.ChannelBase<T>
System.ServiceModel.IClientChannel
System.ServiceModel.OperationContextScope
System.ServiceModel.ServiceHostBase
System.ServiceModel.Activities.SendMessageChannelCache
System.ServiceModel.Channels.Message
System.ServiceModel.Channels.MessageBuffer
System.ServiceModel.Channels.MessageProperties
System.ServiceModel.Channels.RequestContext
System.ServiceModel.Discovery.AnnouncementClient
System.ServiceModel.Discovery.DiscoveryClient
System.ServiceModel.Dispatcher.XPathResult
System.ServiceModel.Routing.RoutingService
System.ServiceModel.Security.SecurityMessageProperty
System.Speech.Recognition.SpeechRecognitionEngine
System.Speech.Recognition.SpeechRecognizer
System.Speech.Synthesis.SpeechSynthesizer
System.Text.StringRuneEnumerator
System.Text.Json.JsonDocument
System.Text.Json.JsonElement.ArrayEnumerator
System.Text.Json.JsonElement.ObjectEnumerator

System.Text.Json.Utf8JsonWriter
System.Threading.AsyncFlowControl
System.Threading.Barrier
System.Threading.CancellationTokenRegistration
System.Threading.CancellationTokenSource
System.Threading.CountdownEvent
System.Threading.ExecutionContext
System.Threading.HostExecutionContext
System.Threading.ManualResetEventSlim
System.Threading.PreAllocatedOverlapped
System.Threading.ReaderWriterLockSlim
System.Threading.SemaphoreSlim
System.Threading.ThreadLocal<T>
System.Threading.ThreadPoolBoundHandle
System.Threading.Timer
System.Threading.WaitHandle
System.Threading.Tasks.Task
System.Transactions.Transaction
System.Transactions.TransactionScope
System.Web.HttpApplication
System.Web.XmlSiteMapProvider
System.Web.Caching.CacheDependency
System.Web.Caching.CacheStoreProvider
System.Web.ClientServices.ClientFormsIdentity
System.Web.Compilation.ClientBuildManager
System.Web.Compilation.IAssemblyPostProcessor
System.Web.Hosting.AspNetMemoryMonitor
System.Web.Hosting.IApplicationMonitor
System.Web.Services.Description.BasicProfileViolationEnumerator
System.Web.UI.Control
System.Web.UI.DataVisualization.Charting.ChartElement
System.Web.UI.DataVisualization.Charting.ChartElementCollection<T>
System.Web.UI.DataVisualization.Charting.ChartElementOutline
System.Web.UI.DataVisualization.Charting.Title
System.Web.UI.Design.IDesignTimeResourceWriter
System.Web.UI.Design.ITemplateEditingFrame
System.Web.UI.Design.TemplateEditingService
System.Web.UI.Design.TemplateEditingVerb
System.Web.UI.Design.WebFormsRootDesigner

System.Web.UI.WebControls.WebParts.WebPartTracker
System.Windows.FreezableCollection<T>.Enumerator
System.Windows.TextDecorationCollection.Enumerator
System.Windows.Annotations.Storage.AnnotationStore
System.Windows.Controls.SoundPlayerAction
System.Windows.Controls.Primitives.DocumentPageView
System.Windows.Documents.DocumentPage
System.Windows.Forms.ApplicationContext
System.Windows.Forms.Control
System.Windows.Forms.Cursor
System.Windows.Forms.DataGridViewBand
System.Windows.Forms.DataGridViewCell
System.Windows.Forms.DataGridViewColumn
System.Windows.Forms.HtmlHistory
System.Windows.Forms.IBindableComponent
System.Windows.Forms.ImageListStreamer
System.Windows.Forms.PaintEventArgs
System.Windows.Forms.ScrollableControl
System.Windows.Forms.TaskDialogIcon
System.Windows.Forms.ToolStrip
System.Windows.Forms.ToolStripItem
System.Windows.Forms.ToolStripOverflow
System.Windows.Forms.ToolStripPanel
System.Windows.Forms.ToolStripPanelRow
System.Windows.Forms.WebBrowserSiteBase
System.Windows.Forms.WindowsFormsSynchronizationContext
System.Windows.Forms.DataVisualization.Charting.AxisScrollBar
System.Windows.Forms.DataVisualization.Charting.Chart
System.Windows.Forms.DataVisualization.Charting.ChartElement
System.Windows.Forms.DataVisualization.Charting.ChartElementCollection<T>
System.Windows.Forms.DataVisualization.Charting.ChartElementOutline
System.Windows.Forms.DataVisualization.Charting.Cursor
System.Windows.Forms.DataVisualization.Charting.PrintingManager
System.Windows.Forms.DataVisualization.Charting.Title
System.Windows.Forms.Design.ComponentDocumentDesigner
System.Windows.Forms.Design.DocumentDesigner
System.Windows.Forms.Design.Behavior.BehaviorService
System.Windows.Ink.GestureRecognizer
System.Windows.Input.Cursor

System.Windows.Interop.HwndHost
System.Windows.Interop.HwndSource
System.Windows.Markup.Primitives.MarkupWriter
System.Windows.Media.CompositionTarget
System.Windows.Media.DoubleCollection.Enumerator
System.Windows.Media.DrawingCollection.Enumerator
System.Windows.Media.DrawingContext
System.Windows.Media.GeneralTransformCollection.Enumerator
System.Windows.Media.GeometryCollection.Enumerator
System.Windows.Media.GradientStopCollection.Enumerator
System.Windows.Media.Int32Collection.Enumerator
System.Windows.Media.PathFigureCollection.Enumerator
System.Windows.Media.PathSegmentCollection.Enumerator
System.Windows.Media.PointCollection.Enumerator
System.Windows.Media.StreamGeometryContext
System.Windows.Media.TextEffectCollection.Enumerator
System.Windows.Media.TransformCollection.Enumerator
System.Windows.Media.VectorCollection.Enumerator
System.Windows.Media.Animation.TimelineCollection.Enumerator
System.Windows.Media.Effects.BitmapEffectCollection.Enumerator
System.Windows.Media.Media3D.GeneralTransform3DCollection.Enumerator
System.Windows.Media.Media3D.MaterialCollection.Enumerator
System.Windows.Media.Media3D.Model3DCollection.Enumerator
System.Windows.Media.Media3D.Point3DCollection.Enumerator
System.Windows.Media.Media3D.Transform3DCollection.Enumerator
System.Windows.Media.Media3D.Vector3DCollection.Enumerator
System.Windows.Media.Media3D.Visual3DCollection.Enumerator
System.Windows.Media.TextFormatting.TextFormatter
System.Windows.Media.TextFormatting.TextLine
System.Windows.Media.TextFormatting.TextLineBreak
System.Windows.Threading.DispatcherProcessingDisabled
System.Windows.Xps.Packaging.XpsDocument
System.Windows.Xps.Packaging.XpsResource
System.Windows.Xps.Serialization.BasePackagingPolicy
System.Windows.Xps.Serialization.PackageSerializationManager
System.Workflow.Activities.ActiveDirectoryRole
System.Workflow.ComponentModel.ActivityExecutionContext
System.Workflow.ComponentModel.DependencyObject
System.Workflow.ComponentModel.Compiler.TypeProvider

System.Workflow.ComponentModel.Design.ActivityDesigner
System.Workflow.ComponentModel.Design.Connector
System.Workflow.ComponentModel.Design.DesignerTheme
System.Workflow.ComponentModel.Design.IWorkflowRootDesigner
System.Workflow.ComponentModel.Design.WorkflowDesignerMessageFilter
System.Workflow.ComponentModel.Design.WorkflowTheme
System.Workflow.Runtime.WorkflowRuntime
System.Xaml.XamlReader
System.Xaml.XamlWriter
System.Xml.XmlNodeList
System.Xml.XmlReader
System.Xml.XmlWriter
System.Xml.Serialization.XmlSchemaEnumerator

Examples

The following example demonstrates how to create a resource class that implements the IDisposable interface.

C# Copy

```
using System;
using System.ComponentModel;

// The following example demonstrates how to create
// a resource class that implements the IDisposable interface
// and the IDisposable.Dispose method.

public class DisposeExample
{
    // A base class that implements IDisposable.
    // By implementing IDisposable, you are announcing that
    // instances of this type allocate scarce resources.
    public class MyResource: IDisposable
    {
        // Pointer to an external unmanaged resource.
        private IntPtr handle;
        // Other managed resource this class uses.
        private Component component = new Component();
        // Track whether Dispose has been called.
        private bool disposed = false;

        // The class constructor.
        public MyResource(IntPtr handle)
        {
```

```
            this.handle = handle;
        }

        // Implement IDisposable.
        // Do not make this method virtual.
        // A derived class should not be able to override this method.
        public void Dispose()
        {
            Dispose(true);
            // This object will be cleaned up by the Dispose method.
            // Therefore, you should call GC.SupressFinalize to
            // take this object off the finalization queue
            // and prevent finalization code for this object
            // from executing a second time.
            GC.SuppressFinalize(this);
        }

        // Dispose(bool disposing) executes in two distinct scenarios.
        // If disposing equals true, the method has been called di-
rectly
        // or indirectly by a user's code. Managed and unmanaged re-
sources
        // can be disposed.
        // If disposing equals false, the method has been called by the
        // runtime from inside the finalizer and you should not refer-
ence
        // other objects. Only unmanaged resources can be disposed.
        protected virtual void Dispose(bool disposing)
        {
            // Check to see if Dispose has already been called.
            if(!this.disposed)
            {
                // If disposing equals true, dispose all managed
                // and unmanaged resources.
                if(disposing)
                {
                    // Dispose managed resources.
                    component.Dispose();
                }

                // Call the appropriate methods to clean up
                // unmanaged resources here.
                // If disposing is false,
                // only the following code is executed.
                CloseHandle(handle);
                handle = IntPtr.Zero;

                // Note disposing has been done.
                disposed = true;
            }
        }
```

```
        // Use interop to call the method necessary
        // to clean up the unmanaged resource.
        [System.Runtime.InteropServices.DllImport("Kernel32")]
        private extern static Boolean CloseHandle(IntPtr handle);

        // Use C# destructor syntax for finalization code.
        // This destructor will run only if the Dispose method
        // does not get called.
        // It gives your base class the opportunity to finalize.
        // Do not provide destructors in types derived from this class.
        ~MyResource()
        {
            // Do not re-create Dispose clean-up code here.
            // Calling Dispose(false) is optimal in terms of
            // readability and maintainability.
            Dispose(false);
        }
    }
    public static void Main()
    {
        // Insert code here to create
        // and use the MyResource object.
    }
}
```

Remarks

The primary use of this interface is to release unmanaged resources. The garbage collector automatically releases the memory allocated to a managed object when that object is no longer used. However, it is not possible to predict when garbage collection will occur. Furthermore, the garbage collector has no knowledge of unmanaged resources such as window handles, or open files and streams.

Use the Dispose method of this interface to explicitly release unmanaged resources in conjunction with the garbage collector. The consumer of an object can call this method when the object is no longer needed.

> **⚠ Warning**
>
> It is a breaking change to add the **IDisposable** interface to an existing class. Because pre-existing consumers of your type cannot call **Dispose**, you cannot be certain that unmanaged resources held by your type will be released.

Because the IDisposable.Dispose implementation is called by the consumer of a type when the resources owned by an instance are no longer needed, you should either wrap the managed object in a SafeHandle (the recommended alternative), or you should override Object.Finalize to free unmanaged resources in the event that the consumer forgets to call Dispose.

> **Important**
>
> In the .NET Framework, the C++ compiler supports deterministic disposal of resources and does not allow direct implementation of the **Dispose** method.

For a detailed discussion about how this interface and the Object.Finalize method are used, see the Garbage Collection and Implementing a Dispose Method topics.

Using an object that implements IDisposable

If your app simply uses an object that implements the IDisposable interface, you should call the object's IDisposable.Dispose implementation when you are finished using it. Depending on your programming language, you can do this in one of two ways:

- By using a language construct such as the `using` statement in C# and Visual Basic.
- By wrapping the call to the IDisposable.Dispose implementation in a `try`/`finally` block.

> **Note**
>
> Documentation for types that implement **IDisposable** note that fact and include a reminder to call its **Dispose** implementation.

The C# and Visual Basic Using statement

If your language supports a construct such as the using statement in C# and the Using statement in Visual Basic, you can use it instead of explicitly calling IDisposable.Dispose yourself. The following example uses this approach in defining a `WordCount` class that preserves information about a file and the number of words in it.

C#	Copy

```
using System;
using System.IO;
using System.Text.RegularExpressions;

public class WordCount
{
   private String filename = String.Empty;
   private int nWords = 0;
   private String pattern = @"\b\w+\b";

   public WordCount(string filename)
   {
      if (! File.Exists(filename))
         throw new FileNotFoundException("The file does not exist.");

      this.filename = filename;
      string txt = String.Empty;
      using (StreamReader sr = new StreamReader(filename)) {
         txt = sr.ReadToEnd();
      }
      nWords = Regex.Matches(txt, pattern).Count;
   }

   public string FullName
   { get { return filename; } }

   public string Name
   { get { return Path.GetFileName(filename); } }

   public int Count
   { get { return nWords; } }
}
```

The `using` statement is actually a syntactic convenience. At compile time, the language compiler implements the intermediate language (IL) for a `try`/`finally` block.

For more information about the `using` statement, see the Using Statement or using Statement topics.

The Try/Finally block

If your programming language does not support a construct like the `using` statement in C# or Visual Basic, or if you prefer not to use it, you can call the IDisposable.Dispose implementation from the `finally` block of a `try`/`finally` statement. The following example replaces the `using` block in the previous example with a `try`/`finally` block.

C# Copy

```
using System;
using System.IO;
using System.Text.RegularExpressions;

public class WordCount
{
   private String filename = String.Empty;
   private int nWords = 0;
   private String pattern = @"\b\w+\b";

   public WordCount(string filename)
   {
      if (! File.Exists(filename))
         throw new FileNotFoundException("The file does not exist.");

      this.filename = filename;
      string txt = String.Empty;
      StreamReader sr = null;
      try {
         sr = new StreamReader(filename);
         txt = sr.ReadToEnd();
      }
      finally {
         if (sr != null) sr.Dispose();
      }
      nWords = Regex.Matches(txt, pattern).Count;
   }

   public string FullName
   { get { return filename; } }

   public string Name
   { get { return Path.GetFileName(filename); } }

   public int Count
   { get { return nWords; } }
}
```

For more information about the `try`/`finally` pattern, see Try...Catch...Finally Statement, try-finally, or try-finally Statement.

Implementing IDisposable

You should implement IDisposable only if your type uses unmanaged resources directly. The consumers of your type can call your IDisposable.Dispose implementation to free

resources when the instance is no longer needed. To handle cases in which they fail to call Dispose, you should either use a class derived from SafeHandle to wrap the unmanaged resources, or you should override the Object.Finalize method for a reference type. In either case, you use the Dispose method to perform whatever cleanup is necessary after using the unmanaged resources, such as freeing, releasing, or resetting the unmanaged resources.

> **Important**
>
> If you are defining a base class that uses unmanaged resources and that either has, or is likely to have, subclasses that should be disposed, you should implement the **IDisposable.Dispose** method and provide a second overload of `Dispose`, as discussed in the next section.

IDisposable and the inheritance hierarchy

A base class with subclasses that should be disposable must implement IDisposable as follows. You should use this pattern whenever you implement IDisposable on any type that isn't `sealed` (`NotInheritable` in Visual Basic).

- It should provide one public, non-virtual Dispose() method and a protected virtual `Dispose(Boolean disposing)` method.
- The Dispose() method must call `Dispose(true)` and should suppress finalization for performance.
- The base type should not include any finalizers.

The following code fragment reflects the dispose pattern for base classes. It assumes that your type does not override the Object.Finalize method.

C# Copy

```
using Microsoft.Win32.SafeHandles;
using System;
using System.Runtime.InteropServices;

class BaseClass : IDisposable
{
   // Flag: Has Dispose already been called?
   bool disposed = false;
   // Instantiate a SafeHandle instance.
   SafeHandle handle = new SafeFileHandle(IntPtr.Zero, true);
```

```
    // Public implementation of Dispose pattern callable by consumers.
    public void Dispose()
    {
        Dispose(true);
        GC.SuppressFinalize(this);
    }

    // Protected implementation of Dispose pattern.
    protected virtual void Dispose(bool disposing)
    {
        if (disposed)
            return;

        if (disposing) {
            handle.Dispose();
            // Free any other managed objects here.
            //
        }

        disposed = true;
    }
}
```

If you do override the Object.Finalize method, your class should implement the following pattern.

C# Copy

```
using System;

class BaseClass : IDisposable
{
    // Flag: Has Dispose already been called?
    bool disposed = false;

    // Public implementation of Dispose pattern callable by consumers.
    public void Dispose()
    {
        Dispose(true);
        GC.SuppressFinalize(this);
    }

    // Protected implementation of Dispose pattern.
    protected virtual void Dispose(bool disposing)
    {
        if (disposed)
            return;
```

```
        if (disposing) {
            // Free any other managed objects here.
            //
        }

        // Free any unmanaged objects here.
        //
        disposed = true;
    }

    ~BaseClass()
    {
        Dispose(false);
    }
}
```

Subclasses should implement the disposable pattern as follows:

- They must override `Dispose(Boolean)` and call the base class `Dispose(Boolean)` implementation.
- They can provide a finalizer if needed. The finalizer must call `Dispose(false)`.

Note that derived classes do not themselves implement the IDisposable interface and do not include a parameterless Dispose method. They only override the base class `Dispose(Boolean)` method.

The following code fragment reflects the dispose pattern for derived classes. It assumes that your type does not override the Object.Finalize method.

C# Copy

```
using Microsoft.Win32.SafeHandles;
using System;
using System.Runtime.InteropServices;

class DerivedClass : BaseClass
{
    // Flag: Has Dispose already been called?
    bool disposed = false;
    // Instantiate a SafeHandle instance.
    SafeHandle handle = new SafeFileHandle(IntPtr.Zero, true);

    // Protected implementation of Dispose pattern.
    protected override void Dispose(bool disposing)
    {
        if (disposed)
            return;
```

```
        if (disposing) {
            handle.Dispose();
            // Free any other managed objects here.
            //
        }

        // Free any unmanaged objects here.
        //

        disposed = true;
        // Call base class implementation.
        base.Dispose(disposing);
    }
}
```

Methods

Dispose()	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

Applies to

.NET

5.0 RC1

.NET Core

3.1, 3.0, 2.2, 2.1, 2.0, 1.1, 1.0

.NET Framework

4.8, 4.7.2, 4.7.1, 4.7, 4.6.2, 4.6.1, 4.6, 4.5.2, 4.5.1, 4.5, 4.0, 3.5, 3.0, 2.0, 1.1

.NET Standard

2.1, 2.0, 1.6, 1.5, 1.4, 1.3, 1.2, 1.1, 1.0

UWP

10.0

Xamarin.Android

7.1

Xamarin.iOS

10.8

Xamarin.Mac

3.0

See also

- SafeFileHandle
- Implementing a Dispose Method