

---

# Windows Presentation Foundation (WPF) with DocFx, Microsoft Logging and Zip Support Solution

## Quick Start Notes:

The Place holder for API Cover Page. Modify this markdown page to change API Cover Page. File is named cover.md located in the VS Project API folder.

### Solution Requirements

DocFx.Console version 2.56.5 or higher.

[Download DocFx Nuget package version 2.56.5](#)

Microsoft.Extensions.Logging version 5.0.0 or higher.

[Download Microsoft.Extensions.Logging Nuget package version 5.0.0](#)

Net Core 50 or higher.

[Download the Net Core 5.0](#)

Visual Studio Enterprise 2019 version 16.8.0 or higher.

[Download the Visual Studio Enterprise 2019](#)

# Table of Contents

WPFAppWithDocFxLoggingAndZip2

App

Extensions

FileLoggerExtensions

FileLoggerProvider

MainWindow

ZipMethods

---

# Namespace WPFAppWithDocFxLoggingAndZip2

## Classes

### [App](#)

Interaction logic for App.xaml

### [Extensions](#)

Class Extensions

### [FileLoggerExtensions](#)

Class FileLoggerExtensions

### [FileLoggerProvider](#)

Class FileLoggerProvider

### [MainWindow](#)

Class MainWindow

### [ZipMethods](#)

Class ZipMethods

# Class App

Interaction logic for App.xaml

Inheritance

System.Object

App

Namespace: [WPFAppWithDocFxLoggingAndZip2](#)

Assembly: WPFAppWithDocFxLoggingAndZip2.dll

Syntax

```
public class App : Application
```

Remarks

## History:

	<b>Code Changed by:</b>  Danny C. McNaught  <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b>  SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b>  Wednesday, December 9, 2020 13:08  <b>Code Changes:</b>  Created XML Comment

Constructors

App()

Declaration

```
public App()
```

Methods

InitializeComponent()

InitializeComponent

Declaration

```
public void InitializeComponent()
```

Main()

Application Entry Point.

Declaration

```
public static void Main()
```

# Class Extensions

## Class Extensions

### Inheritance

System.Object

### Extensions

Namespace: WPFAppWithDocFxLoggingAndZip2

Assembly: WPFAppWithDocFxLoggingAndZip2.dll

### Syntax

```
public static class Extensions : Object
```

### Remarks

### History:

	<b>Code Changed by:</b>  Danny C. McNaught  <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b>  SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b>  Wednesday, December 9, 2020 13:09  <b>Code Changes:</b>  Created XML Comment

### Methods

Add<T>(ICollection<T>, IEnumerable<T>)

Adds the specified collection.

### Declaration

```
public static void Add<T>(this ICollection<T> collection, IEnumerable<T> range)
```

### Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.ICollection<T>	collection	The collection.
System.Collections.Generic.IEnumerable<T>	range	The range.

### Type Parameters

NAME	DESCRIPTION
T	

### Remarks

### History:

--

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

Add<T>(IEnumerable<T>, T)

Adds the specified collection.

Declaration

```
public static IEnumerable<T> Add<T>(this IEnumerable<T> collection, T item)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.IEnumerable<T>	collection	The collection.
T	item	The item.

Returns

TYPE	DESCRIPTION
System.Collections.Generic.IEnumerable<T>	System.Collections.Generic.IEnumerable<T>.

Type Parameters

NAME	DESCRIPTION
T	

Remarks

**History:**

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE

	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment
--	---

AddRange<T>(ICollection<T>, ICollection<T>)

Adds the range.

Declaration

```
public static void AddRange<T>(this ICollection<T> collection, ICollection<T> range)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.ICollection<T>	collection	The collection.
System.Collections.Generic.ICollection<T>	range	The range.

Type Parameters

NAME	DESCRIPTION
T	

Remarks

History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

AddRange<T>(ObservableCollection<T>, IEnumerable<T>)

Adds the range.

Declaration

```
public static void AddRange<T>(this ObservableCollection<T> oc, IEnumerable<T> collection)
```

Parameters

TYPE	NAME	DESCRIPTION

TYPE	NAME	DESCRIPTION
System.Collections.ObjectModel.ObservableCollection<T>	oc	The oc.
System.Collections.Generic.IEnumerable<T>	collection	The collection.

Type Parameters

NAME	DESCRIPTION
T	

Remarks

History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

Exceptions

TYPE	CONDITION
System.ArgumentNullException	collection

AddUniqueIfNotEmpty(ObservableCollection<String>, String)

Adds the unique if not empty.

Declaration

```
public static void AddUniqueIfNotEmpty(this ObservableCollection<string> collection, string text)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.ObjectModel.ObservableCollection<System.String>	collection	The collection.
System.String	text	The text.

Remarks

History:

--



	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

## AppendUniqueOnNewLineIfNotEmpty(StringBuilder, String)

Appends the unique on new line if not empty.

Declaration

```
public static void AppendUniqueOnNewLineIfNotEmpty(this StringBuilder stringBuilder, string text)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Text.StringBuilder	stringBuilder	The string builder.
System.String	text	The text.

Remarks

## History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

## Compare<T>(IEnumerable<T>, IEnumerable<T>)

Compares the specified first.

Declaration

```
public static int Compare<T>(this IEnumerable<T> first, IEnumerable<T> second)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.IEnumerable<T>	first	The first.
System.Collections.Generic.IEnumerable<T>	second	The second.

Returns

TYPE	DESCRIPTION
System.Int32	int.

Type Parameters

NAME	DESCRIPTION
T	

Remarks

History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

Count(IEnumerable)

Counts the specified collection.

Declaration

```
public static int Count(this IEnumerable collection)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.IEnumerable	collection	The collection.

Returns

TYPE	DESCRIPTION
System.Int32	int.

## History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

DecryptStringFromBytes(Byte[], Byte[], Byte[])

Decrypts the string from bytes.

### Declaration

```
public static string DecryptStringFromBytes(this byte[] cipherText, byte[] Key, byte[] IV)
```

### Parameters

TYPE	NAME	DESCRIPTION
System.Byte[]	cipherText	The cipher text.
System.Byte[]	Key	The key.
System.Byte[]	IV	The iv.

### Returns

TYPE	DESCRIPTION
System.String	string.

### Remarks

## History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE

	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment
--	---

Exceptions

TYPE	CONDITION
System.ArgumentNullException	cipherText

DistinctBy<TSource, TKey>(IEnumerable<TSource>, Func<TSource, TKey>)

Distincts the by.

Declaration

```
public static IEnumerable<TSource> DistinctBy<TSource, TKey>(this IEnumerable<TSource> source, Func<TSource, TKey>
keySelector)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.IEnumerable<TSource>	source	The source.
System.Func<TSource, TKey>	keySelector	The key selector.

Returns

TYPE	DESCRIPTION
System.Collections.Generic.IEnumerable<TSource>	System.Collections.Generic.IEnumerable<TSource> .

Type Parameters

NAME	DESCRIPTION
TSource	The type of the t source.
TKey	The type of the t key.

Remarks

History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE

	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment
--	---

EncryptStringToBytes(String, Byte[], Byte[])

Encrypts the string to bytes.

Declaration

<b>public static byte[]</b> EncryptStringToBytes( <b>this string</b> plainText, <b>byte[]</b> Key, <b>byte[]</b> IV)
--

Parameters

TYPE	NAME	DESCRIPTION
System.String	plainText	The plain text.
System.Byte[]	Key	The key.
System.Byte[]	IV	The iv.

Returns

TYPE	DESCRIPTION
System.Byte[]	byte[].

Remarks

**History:**

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

FillWithMembers<T>(ICollection<T>)

Declaration

<b>public static void</b> FillWithMembers<T>( <b>this</b> ICollection<T> collection)
--

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.ICollection<T>	collection	

#### Type Parameters

NAME	DESCRIPTION
T	

### FindAllChildren(DependencyObject)

Finds all children.

#### Declaration

```
public static IEnumerable<DependencyObject> FindAllChildren(this DependencyObject obj)
```

#### Parameters

TYPE	NAME	DESCRIPTION
System.Windows.DependencyObject	obj	The object.

#### Returns

TYPE	DESCRIPTION
System.Collections.Generic.IEnumerable<System.Windows.DependencyObject>	System.Collections.Generic.IEnumerable<System.Windows.DependencyObject>.

#### Remarks

#### History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

### FindChild<T>(DependencyObject, String)

Finds the child.

#### Declaration

```
public static T FindChild<T>(DependencyObject parent, string childName)
    where T : DependencyObject
```

#### Parameters

TYPE	NAME	DESCRIPTION
System.Windows.DependencyObject	parent	The parent.
System.String	childName	Name of the child.

Returns

TYPE	DESCRIPTION
T	T.

Type Parameters

NAME	DESCRIPTION
T	

Remarks

History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

FindChildControl<T>(DependencyObject)

Finds the child control.

Declaration

```
public static DependencyObject FindChildControl<T>(DependencyObject control)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Windows.DependencyObject	control	The control.

Returns

TYPE	DESCRIPTION
System.Windows.DependencyObject	System.Windows.DependencyObject.

Type Parameters

NAME	DESCRIPTION
T	

Remarks

History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

FindChildren(DependencyObject)

Finds the children.

Declaration

```
public static IEnumerable<DependencyObject> FindChildren(this DependencyObject obj)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Windows.DependencyObject	obj	The object.

Returns

TYPE	DESCRIPTION
System.Collections.Generic.IEnumerable<System.Windows.DependencyObject>	System.Collections.Generic.IEnumerable<System.Windows.DependencyObject>.

Remarks

History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE



	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment
--	---

FindParentControl<T>(DependencyObject)

Finds the parent control.

Declaration

```
public static DependencyObject FindParentControl<T>(DependencyObject control)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Windows.DependencyObject	control	The control.

Returns

TYPE	DESCRIPTION
System.Windows.DependencyObject	System.Windows.DependencyObject.

Type Parameters

NAME	DESCRIPTION
T	

Remarks

History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

FindVisualParent<T>(Visual)

Finds the visual parent.

Declaration

```
public static T FindVisualParent<T>(Visual element)  
    where T : Visual
```

Parameters

TYPE	NAME	DESCRIPTION
System.Windows.Media.Visual	element	The element.

Returns

TYPE	DESCRIPTION
T	T.

Type Parameters

NAME	DESCRIPTION
T	

Remarks

#### History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

#### GetDescription(Enum)

Gets the description.

Declaration

<code>public static string GetDescription(this Enum value)</code>
---

Parameters

TYPE	NAME	DESCRIPTION
System.Enum	value	The value.

Returns

TYPE	DESCRIPTION
System.String	string.

Remarks

#### History:

--

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

## GetXDocument(XmlDocument)

Gets the x document.

Declaration

```
public static XDocument GetXDocument(this XmlDocument document)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Xml.XmlDocument	document	The document.

Returns

TYPE	DESCRIPTION
System.Xml.Linq.XDocument	System.Xml.Linq.XDocument.

Remarks

## History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

## GetXElement(XmlNode)

Gets the x element.

Declaration

```
public static XElement GetXElement(this XmlNode node)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Xml.XmlNode	node	The node.

Returns

TYPE	DESCRIPTION
System.Xml.Linq.XElement	System.Xml.Linq.XElement.

Remarks

History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

GetXmlDocument(XDocument)

Gets the XML document.

Declaration

```
public static XmlDocument GetXmlDocument(this XDocument document)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Xml.Linq.XDocument	document	The document.

Returns

TYPE	DESCRIPTION
System.Xml.XmlDocument	System.Xml.XmlDocument.

Remarks

History:

--

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

GetXmlNode(XElement)

Gets the XML node.

Declaration

```
public static XmlNode GetXmlNode(this XElement element)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Xml.Linq.XElement	element	The element.

Returns

TYPE	DESCRIPTION
System.Xml.XmlNode	System.Xml.XmlNode.

Remarks

History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

IsAny<T>(IEnumerable<T>)

Determines whether the specified data is any.

Declaration

```
public static bool IsAny<T>(this IEnumerable<T> data)
```

#### Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.IEnumerable<T>	data	The data.

#### Returns

TYPE	DESCRIPTION
System.Boolean	

#### Type Parameters

NAME	DESCRIPTION
T	

#### Remarks

#### History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

#### IsLetter(Char)

Determines whether the specified c is letter.

#### Declaration

```
public static bool IsLetter(this char c)
```

#### Parameters

TYPE	NAME	DESCRIPTION
System.Char	c	The c.

#### Returns

TYPE	DESCRIPTION
System.Boolean	

#### Remarks

#### History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

IsLetterOrSeparator(Char)

Determines whether [is letter or separator] [the specified c].

Declaration

```
public static bool IsLetterOrSeparator(this char c)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Char	c	The c.

Returns

TYPE	DESCRIPTION
System.Boolean	

Remarks

History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

IsNullOrEmpty(String)

Determines whether [is null or empty] [the specified s].

Declaration

```
public static bool IsNullOrEmpty(this string s)
```

## Parameters

TYPE	NAME	DESCRIPTION
System.String	s	The s.

## Returns

TYPE	DESCRIPTION
System.Boolean	

## Remarks

## History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> DESKTOP-ACLF30
	<b>Code Change Date and Time:</b> Monday, November 23, 2020 12:41 PM  <b>Code Changes:</b> Created XML Comment

IsNullOrEmpty<T>(IEnumerable<T>)

Determines whether [is null or empty] [the specified list].

## Declaration

<code>public static bool IsNullOrEmpty&lt;T&gt;(this IEnumerable&lt;T&gt; list)</code>
--

## Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.IEnumerable<T>	list	The list.

## Returns

TYPE	DESCRIPTION
System.Boolean	

## Type Parameters

NAME	DESCRIPTION
T	

## Remarks

## History:

--



	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

NotNullAny<T>(IEnumerable<T>, Func<T, Boolean>)

Nots the null any.

Declaration

```
public static bool NotNullAny<T>(this IEnumerable<T> enumerable, Func<T, bool> predicate)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.IEnumerable<T>	enumerable	The enumerable.
System.Func<T, System.Boolean>	predicate	The predicate.

Returns

TYPE	DESCRIPTION
System.Boolean	bool.

Type Parameters

NAME	DESCRIPTION
T	

Remarks

**History:**

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE

	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment
--	---

PerformClick(Button)

Performs the click.

Declaration

```
public static void PerformClick(this Button btn)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Windows.Controls.Button	btn	The BTN.

Remarks

History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

Refresh(UIElement)

Refreshes the specified UI element.

Declaration

```
public static void Refresh(this UIElement uiElement)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Windows.UIElement	uiElement	The UI element.

Remarks

History:

--

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

Remove<T>(ICollection<T>, IEnumerable<T>)

Removes the specified collection.

Declaration

```
public static void Remove<T>(this ICollection<T> collection, IEnumerable<T> range)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.ICollection<T>	collection	The collection.
System.Collections.Generic.IEnumerable<T>	range	The range.

Type Parameters

NAME	DESCRIPTION
T	

Remarks

History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

SelectAllRecursively<T>(IEnumerable<T>, Func<T, IEnumerable<T>>)

Selects all recursively.

#### Declaration

```
public static IEnumerable<T> SelectAllRecursively<T>(this IEnumerable<T> items, Func<T, IEnumerable<T>> func)
```

#### Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.IEnumerable<T>	items	The items.
System.Func<T, System.Collections.Generic.IEnumerable<T>>	func	The function.

#### Returns

TYPE	DESCRIPTION
System.Collections.Generic.IEnumerable<T>	System.Collections.Generic.IEnumerable<T>.

#### Type Parameters

NAME	DESCRIPTION
T	

#### Remarks

#### History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

#### SkipExceptions<T>(IEnumerable<T>)

Skips the exceptions.

#### Declaration

```
public static IEnumerable<T> SkipExceptions<T>(this IEnumerable<T> values)
```

#### Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.IEnumerable<T>	values	The values.

#### Returns

TYPE	DESCRIPTION
System.Collections.Generic.IEnumerable<T>	System.Collections.Generic.IEnumerable<T>.

Type Parameters

NAME	DESCRIPTION
T	

Remarks

History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

Sort<TSource, TKey>(ObservableCollection<TSource>, Func<TSource, TKey>)

Sorts the specified collection.

Declaration

```
public static void Sort<TSource, TKey>(this ObservableCollection<TSource> collection, Func<TSource, TKey> keySelector)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.ObjectModel.ObservableCollection<TSource>	collection	The collection.
System.Func<TSource, TKey>	keySelector	The key selector.

Type Parameters

NAME	DESCRIPTION
TSource	The type of the t source.
TKey	The type of the t key.

Remarks

History:

--

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

ToObservableCollection<T>(IEnumerable<T>)

To the observable collection.

Declaration

```
public static ObservableCollection<T> ToObservableCollection<T>(this IEnumerable<T> inputCollection)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.IEnumerable<T>	inputCollection	The input collection.

Returns

TYPE	DESCRIPTION
System.Collections.ObjectModel.ObservableCollection<T>	System.Collections.ObjectModel.ObservableCollection<T>.

Type Parameters

NAME	DESCRIPTION
T	

Remarks

**History:**

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

# Class FileLoggerExtensions

Class FileLoggerExtensions

Inheritance

System.Object

FileLoggerExtensions

Namespace: [WPFAppWithDocFxLoggingAndZip2](#)

Assembly: WPFAppWithDocFxLoggingAndZip2.dll

Syntax

```
public static class FileLoggerExtensions : Object
```

Remarks

## History:

	<b>Code Changed by:</b>  Danny C. McNaught  <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b>  SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b>  Wednesday, December 9, 2020 13:09  <b>Code Changes:</b>  Created XML Comment

## Methods

AddFile(ILoggerFactory, String, String)

Adds the file.

Declaration

```
public static ILoggerFactory AddFile(this ILoggerFactory factory, string name, string logFolder)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Extensions.Logging.ILoggerFactory	factory	The factory.
System.String	name	The name.
System.String	logFolder	The log folder.

## Returns

TYPE	DESCRIPTION
Microsoft.Extensions.Logging.ILoggerFactory	ILoggerFactory.

## Remarks

## History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

## Log(ILogger, LogLevel, Exception, String, Boolean, Object[])

Logs the specified log level.

## Declaration

```
public static void Log(this ILogger logger, LogLevel logLevel, Exception exception, string message, bool includeStackTrace = false, params object[] args)
```

## Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Extensions.Logging.ILogger	logger	The logger.
Microsoft.Extensions.Logging.LogLevel	logLevel	The log level.
System.Exception	exception	The exception.
System.String	message	The message.
System.Boolean	includeStackTrace	if set to <code>true</code> [include stack trace].



TYPE	NAME	DESCRIPTION
System.Object[]	args	The arguments.

Remarks

History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

Log(ILogger, LogLevel, String, Object[])

Logs the specified log level.

Declaration

```
public static void Log(this ILogger logger, LogLevel logLevel, string message, params object[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Extensions.Logging.ILogger	logger	The logger.
Microsoft.Extensions.Logging.LogLevel	logLevel	The log level.
System.String	message	The message.
System.Object[]	args	The arguments.

Remarks

History:

--

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

LogDebug(ILogger, String, Boolean, String, String, Int32)

Logs the debug.

Declaration

```
public static void LogDebug(this ILogger logger, string message, bool includeLineInfo, string callerMemberName = "", string callerFilePath = "", int callerLineNumber = 0)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Extensions.Logging.ILogger	logger	The logger.
System.String	message	The message.
System.Boolean	includeLineInfo	if set to <code>true</code> [include line information].
System.String	callerMemberName	Name of the caller member.
System.String	callerFilePath	The caller file path.
System.Int32	callerLineNumber	The caller line number.

Remarks

**History:**

--

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

LogError(ILogger, Exception, String, Boolean, Boolean, String, String, Int32)

Logs the error.

Declaration

```
public static void LogError(this ILogger logger, Exception exception, string message, bool includeLineInfo, bool includeStackTrace, string callerMemberName = "", string callerFilePath = "", int callerLineNumber = 0)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Extensions.Logging.ILogger	logger	The logger.
System.Exception	exception	The exception.
System.String	message	The message.
System.Boolean	includeLineInfo	if set to <code>true</code> [include line information].
System.Boolean	includeStackTrace	if set to <code>true</code> [include stack trace].
System.String	callerMemberName	Name of the caller member.
System.String	callerFilePath	The caller file path.
System.Int32	callerLineNumber	The caller line number.

Remarks

## History:

	<b>Code Changed by:</b>  Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b>  SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b>  Wednesday, December 9, 2020 13:09  <b>Code Changes:</b>  Created XML Comment

LogInformation(ILogger, String, Boolean, String, String, Int32)

Logs the information.

Declaration

```
public static void LogInformation(this ILogger logger, string message, bool includeLineInfo, string callerMemberName = "", string callerFilePath = "", int callerLineNumber = 0)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Extensions.Logging.ILogger	logger	The logger.
System.String	message	The message.
System.Boolean	includeLineInfo	if set to <code>true</code> [include line information].
System.String	callerMemberName	Name of the caller member.
System.String	callerFilePath	The caller file path.
System.Int32	callerLineNumber	The caller line number.

Remarks

## History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

LogTrace(ILogger, String, Boolean, String, String, Int32)

Logs the trace.

Declaration

```
public static void LogTrace(this ILogger logger, string message, bool includeLineInfo, string callerMemberName = "", string callerFilePath = "", int callerLineNumber = 0)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Extensions.Logging.ILogger	logger	The logger.
System.String	message	The message.
System.Boolean	includeLineInfo	if set to <code>true</code> [include line information].
System.String	callerMemberName	Name of the caller member.
System.String	callerFilePath	The caller file path.
System.Int32	callerLineNumber	The caller line number.

Remarks

**History:**

--

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

LogWarning(ILogger, String, Boolean, String, String, Int32)

Logs the warning.

Declaration

```
public static void LogWarning(this ILogger logger, string message, bool includeLineInfo, string callerMemberName = "", string callerFilePath = "", int callerLineNumber = 0)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Extensions.Logging.ILogger	logger	The logger.
System.String	message	The message.
System.Boolean	includeLineInfo	if set to <code>true</code> [include line information].
System.String	callerMemberName	Name of the caller member.
System.String	callerFilePath	The caller file path.
System.Int32	callerLineNumber	The caller line number.

Remarks

**History:**

--

	<b>Code Changed by:</b>  Danny C. McNaught  <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b>  SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b>  Wednesday, December 9, 2020 13:09  <b>Code Changes:</b>  Created XML Comment

# Class FileLoggerProvider

Class FileLoggerProvider

Inheritance

System.Object  
FileLoggerProvider

Implements

Microsoft.Extensions.Logging.ILoggerProvider  
System.IDisposable

Namespace: [WPFAppWithDocFxLoggingAndZip2](#)  
Assembly: WPFAppWithDocFxLoggingAndZip2.dll

Syntax

```
public sealed class FileLoggerProvider : Object, ILoggerProvider, IDisposable
```

Remarks

History:

	<b>Code Changed by:</b>  Danny C. McNaught  <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b>  SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b>  Wednesday, December 9, 2020 13:09  <b>Code Changes:</b>  Created XML Comment

Constructors

FileLoggerProvider()

Initializes a new instance of the System.Object class.

Declaration

```
public FileLoggerProvider()
```

Remarks

History:

	<b>Code Changed by:</b>  Danny C. McNaught  <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>



	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

## FileLoggerProvider(StreamWriter)

Initializes a new instance of the [FileLoggerProvider](#) class.

Declaration

```
public FileLoggerProvider(StreamWriter writer)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IO.StreamWriter	writer	The writer.

Remarks

## History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

## FileLoggerProvider(String, String)

Initializes a new instance of the [FileLoggerProvider](#) class.

Declaration

```
public FileLoggerProvider(string name = null, string logFolder = null)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	name	The name.
System.String	logFolder	The log folder.

Remarks

#### History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

#### Methods

##### AppName()

Applications the name.

Declaration

```
public static string AppName()
```

Returns

TYPE	DESCRIPTION
System.String	System.String.

Remarks

#### History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>

	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

## AppPath()

Applications the path.

Declaration

```
public static string AppPath()
```

Returns

TYPE	DESCRIPTION
System.String	System.String.

Remarks

## History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

## CreateLogger(String)

Creates a new Microsoft.Extensions.Logging.ILogger instance.

Declaration

```
public ILogger CreateLogger(string categoryName)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	categoryName	The category name for messages produced by the logger.

#### Returns

TYPE	DESCRIPTION
Microsoft.Extensions.Logging.ILogger	The instance of Microsoft.Extensions.Logging.ILogger that was created.

#### Remarks

#### History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

#### Dispose()

Disposes this instance.

#### Declaration

```
public void Dispose()
```

#### Remarks

#### History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE

**Code Change Date and Time:**

Wednesday, December 9, 2020 13:09

**Code Changes:**

Created XML Comment

GetLogFileName(String, String)

Gets the name of the log file.

Declaration

```
public static string GetLogFileName(string name, string logFolder)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	name	The name.
System.String	logFolder	The log folder.

Returns

TYPE	DESCRIPTION
System.String	System.String.

Remarks

**History:****Code Changed by:**

Danny C. McNaught

[danny.mcnaught@dannymcnaught.com](mailto:danny.mcnaught@dannymcnaught.com)

**Code changed on Visual Studio Host Machine:**

SURFACE-PRO-LTE

**Code Change Date and Time:**

Wednesday, December 9, 2020 13:09

**Code Changes:**

Created XML Comment

Implements

Microsoft.Extensions.Logging.ILoggerProvider

System.IDisposable

Implements

# Class MainWindow

Class MainWindow

Inheritance

System.Object  
MainWindow

Implements

System.Windows.Markup.IComponentConnector

Namespace: [WPFAppWithDocFxLoggingAndZip2](#)

Assembly: WPFAppWithDocFxLoggingAndZip2.dll

Syntax

```
public class MainWindow : Window, IComponentConnector
```

Remarks

History:

	<b>Code Changed by:</b>  Danny C. McNaught  <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b>  SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b>  Wednesday, December 9, 2020 13:09  <b>Code Changes:</b>  Created XML Comment

Constructors

MainWindow()

Initializes a new instance of the System.Windows.Window class.

Declaration

```
public MainWindow()
```

Remarks

History:

	<b>Code Changed by:</b>  Danny C. McNaught  <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>

	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

## Properties

### Factory

Gets or sets the factory.

#### Declaration

```
public static ILoggerFactory Factory { get; set; }
```

#### Property Value

TYPE	DESCRIPTION
Microsoft.Extensions.Logging.ILoggerFactory	The factory.

#### Remarks

### History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

## Logger

Gets or sets the logger.

#### Declaration

```
public static ILogger Logger { get; set; }
```

#### Property Value

## Properties

TYPE	DESCRIPTION
Microsoft.Extensions.Logging.ILogger	The logger.

Remarks

### History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

### Methods

#### AppName()

Applications the name.

Declaration

<code>public static string AppName()</code>
---

Returns

TYPE	DESCRIPTION
System.String	System.String.

Remarks

### History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE



	<b>Code Change Date and Time:</b>  Wednesday, December 9, 2020 13:09  <b>Code Changes:</b>  Created XML Comment
--	---

AppPath()

Applications the path.

Declaration

```
public static string AppPath()
```

Returns

TYPE	DESCRIPTION
System.String	System.String.

Remarks

History:

	<b>Code Changed by:</b>  Danny C. McNaught  <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b>  SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b>  Wednesday, December 9, 2020 13:09  <b>Code Changes:</b>  Created XML Comment

InitializeComponent()

InitializeComponent

Declaration

```
public void InitializeComponent()
```

Implements

System.Windows.Markup.IComponentConnector

# Class ZipMethods

Class ZipMethods

Inheritance

System.Object

ZipMethods

Namespace: [WPFAppWithDocFxLoggingAndZip2](#)

Assembly: WPFAppWithDocFxLoggingAndZip2.dll

Syntax

```
public static class ZipMethods : Object
```

Remarks

## History:

	<b>Code Changed by:</b>  Danny C. McNaught  <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b>  SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b>  Wednesday, December 9, 2020 13:09  <b>Code Changes:</b>  Created XML Comment

## Methods

AddFileToZip(String, String, Boolean, Boolean, CompressionLevel)

Adds the file to zip.

Declaration

```
public static void AddFileToZip(string zipFilePath, string fileToAddFullName, bool promptZipExist = true, bool promptFileExist = true, CompressionLevel compressionLevel)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	zipFilePath	The zip file path.
System.String	fileToAddFullName	Full name of the file to add.

TYPE	NAME	DESCRIPTION
System.Boolean	promptZipExist	Prompt if zip file exist.
System.Boolean	promptFileExist	Prompt if file exist.
System.IO.Compression.CompressionLevel	compressionLevel	The compression level.

Remarks

### History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

CreateNewFileInZip(String, String, String[], Boolean, CompressionLevel)

Creates the new file in zip.

Declaration

```
public static void CreateNewFileInZip(string zipFilePath, string fileNameToCreate, string[]
linesToAddToNewFile, bool promptFileExist = true, CompressionLevel compressionLevel)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	zipFilePath	The zip file path.
System.String	fileNameToCreate	The file name of new file to create in zip.
System.String[]	linesToAddToNewFile	The lines to add to new file in zip.

TYPE	NAME	DESCRIPTION
System.Boolean	promptFileExist	Prompt if file exist.
System.IO.Compression.CompressionLevel	compressionLevel	The compression level.

Remarks

#### History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

#### CreateNewZipFile(String, String, Boolean, CompressionLevel)

Creates the new zip file.

Declaration

```
public static void CreateNewZipFile(string zipFilePath, string fileToAddFullName, bool promptZipExist = true,
CompressionLevel compressionLevel)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	zipFilePath	The zip file path.
System.String	fileToAddFullName	Full name of the file to add.
System.Boolean	promptZipExist	Prompt if zip file exist.
System.IO.Compression.CompressionLevel	compressionLevel	The compression level.

Remarks

## History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09  <b>Code Changes:</b> Created XML Comment

## CreateZipFromDir(String, String, Boolean, CompressionLevel)

Creates the zip from dir.

### Declaration

```
public static void CreateZipFromDir(string DirToZip, string zipFilePath, bool promptFileExist = true,
CompressionLevel compressionLevel)
```

### Parameters

TYPE	NAME	DESCRIPTION
System.String	DirToZip	The existing directory to zip.
System.String	zipFilePath	The zip file path.
System.Boolean	promptFileExist	Prompt if file exist.
System.IO.Compression.CompressionLevel	compressionLevel	The compression level.

### Remarks

## History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>

	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

DeleteFileInZip(String, String, Boolean)

Deletes the file in zip.

Declaration

```
public static void DeleteFileInZip(string zipFilePath, string fileNameToDelete, bool promptFileExist = true)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	zipFilePath	The zip file path.
System.String	fileNameToDelete	The file name in the zip file to delete.
System.Boolean	promptFileExist	Prompt if file exist.

Remarks

**History:**

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment

FileExistInZip(String, String, Boolean)

Files the exist in zip.

## Declaration

```
public static bool FileExistInZip(string zipFilePath, string fileNameToFind, bool promptFileExist = true)
```

## Parameters

TYPE	NAME	DESCRIPTION
System.String	zipFilePath	The zip file path.
System.String	fileNameToFind	The file name to find.
System.Boolean	promptFileExist	The prompt file exist.

## Returns

TYPE	DESCRIPTION
System.Boolean	bool.

## Remarks

## History:

	<b>Code Changed by:</b> Danny C. McNaught <a href="mailto:danny.mcnaught@dannymcnaught.com">danny.mcnaught@dannymcnaught.com</a>
	<b>Code changed on Visual Studio Host Machine:</b> SURFACE-PRO-LTE
	<b>Code Change Date and Time:</b> Wednesday, December 9, 2020 13:09 <b>Code Changes:</b> Created XML Comment