

Namespace AkiraNetwork.VirtualStorage Library

▼ Filter by title

Classes

- **AkiraNetwork.VirtualStorage**

Library

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html)

ActionNodeDelegate

(AkiraNetwork.VirtualStorageLibrary.

VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html)

Represents a virtual directory. A virtual directory functions as a container for other nodes. It can contain Virtual

Item (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html), VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html), and VirtualSymbolicLink

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html), providing a centralized way to manage these

entities.

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary.

This class supports operations such as adding, removing, updating, checking the existence of, and enumerating

PatternMatch

nodes within the directory. It serves as a fundamental component for efficiently managing the structure of the

virtual storage.

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary.

Additional nodes are generated according to the specified display conditions.

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary.

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter-1.html)

1.html)

VirtualException

VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualException.html)

(AkiraNetwork.VirtualStorageLibrary.

The base class for exceptions that occur within the virtual storage library.

(AkiraNetwork.VirtualStorageLibrary.

2.html)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition-2.html)

(AkiraNetwork.VirtualStorageLibrary.

Represents the conditions for grouping data, holding the property to group by and the order (ascending or

descending).

(AkiraNetwork.VirtualStorageLibrary.

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.

VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)

(AkiraNetwork.VirtualStorageLibrary.

Represents a virtual item. This class serves as the base class for the VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) type and is used solely for type checking of VirtualItem

(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html).

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.

VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter-1.html)

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeExtensions

VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html)

Represents a virtual item. A virtual item is an object used to manage item data within the virtual storage. It holds item data based on the user-defined type T, facilitating the management and manipulation of various types of content.

Virtual items ensure that the actual data is distinctly present. If the user-defined type T implements the IVirtualDeepCloneable<T> (AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html) interface, a deep clone of the item data is created, ensuring separate instances. If this interface is not implemented, shallow copies may result in multiple virtual items referencing the same instance of the item data.

- AkiraNetwork.VirtualStorageLibrary

VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.VirtualNode.html)

Represents a node delegate class for nodes.

(AkiraNetwork.VirtualStorageLibrary

IVirtualDeepCloneable<T>

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.VirtualNodeContext.html)

A VirtualNodeContext contains information for a node. It is returned during or after path traversal, providing information about VirtualStorageLibrary, parent directory, depth, index, resolved path, and symbolic link.

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.VirtualNodeExtensions.html)

Provides extension methods for operations on virtual nodes in the Virtual Storage Library.

(AkiraNetwork.VirtualStorageLibrary

VirtualDirectory

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.VirtualNodeName.html)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.VirtualNodeNotFoundException.html)

The specified VirtualStorageLibrary node is not found in the virtual storage.

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary

VirtualPath

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.VirtualPath.html)

VirtualID

(AkiraNetwork.VirtualStorageLibrary

VirtualSortCondition<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.VirtualSortCondition-1.html)

Represents the conditions for sorting data, holding the property to sort by and the order (ascending or descending).

(AkiraNetwork.VirtualStorageLibrary

1.html)

VirtualItem<T>

VirtualStorageExtensions

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.VirtualStorageExtensions.html)

Provides extension methods for various operations on collections in the Virtual Storage Library.

(AkiraNetwork.VirtualStorageLibrary

VirtualNodeContext

VirtualStorageSettings

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.VirtualStorageSettings.html)

(AkiraNetwork.VirtualStorageLibrary

VirtualStorageState ([AkiraNetwork.VirtualStorageLibrary.VirtualStorageState.html](#))

VirtualStorage<T> ([AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html](#))

VirtualSymbolicLink

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)

- **AkiraNetwork.VirtualStorageLibrary.VirtualStorage** A virtual symbolic link provides a virtual reference to other nodes (directories, items, or other symbolic links). This class represents symbolic links within the virtual storage library and manages paths to other nodes.

(AkiraNetwork.VirtualStorage)

The virtual symbolic link supports the concept of NULL links. When the TargetPath is null, the link resolution is not performed during path traversal. This feature can be useful when the target is undefined or dynamically determined.

VirtualDeepCloneable<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualDeepCloneable.html)

VirtualSymbolicLinkAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLinkAdapter-1.html)

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLinkAdapter)

NotifyNodeDelegate

Structs

(AkiraNetwork.VirtualStorageLibrary.VirtualStorage)

PatternMatch

(AkiraNetwork.VirtualStorageLibrary.VirtualStorage)

VirtualID ([AkiraNetwork.VirtualStorageLibrary.VirtualID.html](#))

(AkiraNetwork.VirtualStorageLibrary.VirtualID)

A **AkiraNetwork.VirtualStorageLibrary.VirtualID** identifier within the virtual storage library. This identifier is based on a randomly generated UUID version 4 (UUIDv4). UUIDv4 is a 128-bit value that ensures high uniqueness and has strong cryptographic properties.

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html)

Represents **AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions** a list of virtual nodes.

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition)

Interfaces

(AkiraNetwork.VirtualStorageLibrary.Interfaces)

VirtualID

(AkiraNetwork.VirtualStorageLibrary.VirtualID)

IVirtualDeepCloneable<T>

(AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html)

(AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable)

Provides functionality to create deep clones of type T.

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter)

IVirtualWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher.html)

(AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher)

Provides functionality for wildcard matching.

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.VirtualNode)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext)

VirtualNodeExtensions

Enums

VirtualNodeType ([AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html](#))

Specifies the types of virtual nodes.

▼

VirtualNodeTypeFilter

([AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html](#))

Specifies the filters for virtual node types.

(AkiraNetwork.VirtualStorageLibrary)

ActionNodeDelegate

([AkiraNetwork.VirtualStorageLibrary.ActionNodeDelegate.html](#))

IVirtualWildcardMatcher

A delegate used to perform an action on a specific node during node traversal.

([AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher.html](#))

NotifyNodeDelegate

([AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate.html](#))

IVirtualCycleDetector

A delegate used to notify the status of a specific node during node traversal.

([AkiraNetwork.VirtualStorageLibrary.IVirtualCycleDetector.html](#))

PatternMatch

([AkiraNetwork.VirtualStorageLibrary.PatternMatch.html](#))

IVirtualNameMatcher

A delegate used to determine if a node name matches a pattern.

([AkiraNetwork.VirtualStorageLibrary.IVirtualNameMatcher.html](#))

VirtualException

([AkiraNetwork.VirtualStorageLibrary.VirtualException.html](#))

VirtualGroupCondition<T, TKey>

([AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition.html](#))

VirtualID

([AkiraNetwork.VirtualStorageLibrary.VirtualID.html](#))

VirtualItem

([AkiraNetwork.VirtualStorageLibrary.VirtualItem.html](#))

VirtualItemAdapter<T>

([AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter.html](#))

VirtualNode

([AkiraNetwork.VirtualStorageLibrary.VirtualNode.html](#))

VirtualNodeContext

([AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html](#))

VirtualNodeExtensions

Delegate ActionNodeDelegate

A delegate used to perform an action on a specific node during node traversal.

Filter by title

Namespace: AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)

Assemblies: AkiraNetwork.dll

Symbols

(AkiraNetwork.VirtualStorage)

```
public delegate bool ActionNodeDelegate(VirtualDirectory parentDirectory, VirtualNodeName nodeName, VirtualPath nodePath)
```

IVirtualDeepCloneable<T>

Parameters

| Type | Name | Description |
|------------------------|-----------------|--|
| VirtualWildcardMatcher | | |
| VirtualDirectory | parentDirectory | The parent directory of the target node. |
| VirtualNodeDelegate | | |
| VirtualDirectoryName | nodeName | The name of the target node. |
| VirtualPath | nodePath | The path of the target node. |
| VirtualCycleDetector | | |
| VirtualPath | | |
| VirtualDirectory | | |

Returns

| Type | Description |
|----------------------------|---|
| VirtualDirectoryAdapter<T> | Returns true to continue node traversal, or false to stop node traversal. |

Extension Methods

| | |
|--|--|
| VirtualTextFormatter.GenerateSingleTableDebugText<T>(T) | |
| VirtualItem | |
| (AkiraNetwork.VirtualStorageLibrary.Uses.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilities_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_) | |
| VirtualItemAdapter<T> | |
| (AkiraNetwork.VirtualStorageLibrary.Uses.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilities_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_) | |
| VirtualItem<T> | |
| (AkiraNetwork.VirtualStorageLibrary.Uses.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilities_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_) | |
| VirtualNode | |
| (AkiraNetwork.VirtualStorageLibrary.Uses.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilities_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_) | |
| VirtualNodeContext | |
| (AkiraNetwork.VirtualStorageLibrary.Uses.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilities_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_) | |
| VirtualNodeExtensions | |

Interface IVirtualDeepCloneable<T>

Provides functionality to create deep clones of type T.

Filter by title

Namespace: AkiraNetwork ([AkiraNetwork.html](#)).VirtualStorageLibrary ([AkiraNetwork.VirtualStorageLibrary.html](#))

Assembly: VirtualStorageLibrary.dll

Syntax ([AkiraNetwork.VirtualStorageLibrary.html](#))

```
public interface IVirtualDeepCloneable<T>  
    where T : class
```

IVirtualDeepCloneable<T>

Type Parameters ([1.html](#))

| Name | Description |
|------|---|
| T | (AkiraNetwork.VirtualStorageLibrary) The type of the object to clone. |

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary)

PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

Methods

DeepClone(bool)

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

Creates a deep clone of the object.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualException

T (AkiraNetwork.VirtualStorageLibrary) false

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

Parameters

2.html)

| Type | virtualID | Name | Description |
|------|-----------|-----------|--|
| bool | | recursive | Indicates whether to recursively clone nested objects. If true, a recursive clone is performed; if false, only the object itself is cloned. This parameter is only relevant for VirtualDirectory, and is ignored for VirtualItem{T} and VirtualSymbolicLink. |

1.html)

Returns

VirtualItem<T>

| Type | Description |
|------|--------------------|
| T | The cloned object. |

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

Extension Methods

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)



- **AkiraNetwork.VirtualStorage**
Library
(AkiraNetwork.VirtualStorageL

ActionNodeDelegate
(AkiraNetwork.VirtualStorageLibrar
IVirtualDeepCloneable<T>
(AkiraNetwork.VirtualStorageLibrar
1.html)
IVirtualWildcardMatcher
(AkiraNetwork.VirtualStorageLibrar
NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrar
PatternMatch
(AkiraNetwork.VirtualStorageLibrar
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrar
VirtualDirectory
(AkiraNetwork.VirtualStorageLibrar
VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrar
1.html)
VirtualException
(AkiraNetwork.VirtualStorageLibrar
VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrar
2.html)
VirtualID
(AkiraNetwork.VirtualStorageLibrar
VirtualItem
(AkiraNetwork.VirtualStorageLibrar
VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrar
1.html)
VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrar
1.html)
VirtualNode
(AkiraNetwork.VirtualStorageLibrar
VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrar
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrar

Interface IVirtualWildcardMatcher

Provides functionality for wildcard matching.

Filter by title

Namespace: AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)

Library

Assembly: VirtualStorageLibrary.dll

(AkiraNetwork.VirtualStorageLibrary)

Syntax

```
ActionNodeDelegate  
  (AkiraNetwork.VirtualStorageLibrary  
  public interface IVirtualWildcardMatcher  
  VirtualDeepCloneable<IVirtualWildcardMatcher>
```

```
  (AkiraNetwork.VirtualStorageLibrary  
  1.html)  
  IVirtualWildcardMatcher
```

```
  (AkiraNetwork.VirtualStorageLibrary  
  NotifyNodeDelegate
```

```
  (AkiraNetwork.VirtualStorageLibrary  
  PatternMatch
```

```
  (AkiraNetwork.VirtualStorageLibrary  
  VirtualDirectory
```

Gets [VirtualNodeDelegate](#)s.

```
  (AkiraNetwork.VirtualStorageLibrary  
  Declaration
```

```
  VirtualDirectory
```

```
  (AkiraNetwork.VirtualStorageLibrary  
  int Count { get; }
```

```
  (AkiraNetwork.VirtualStorageLibrary  
  1.html)
```

Property Value

```
VirtualException
```

| Type | Description |
|---|--|
| int VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary https://learn.microsoft.com/dotnet/api/system.int32 2.html) | An integer representing the number of wildcards available. |

```
VirtualID
```

```
  (AkiraNetwork.VirtualStorageLibrary
```

Patterns

```
  (AkiraNetwork.VirtualStorageLibrary  
Gets a collection of available patterns.
```

```
  VirtualItemAdapter<T>
```

Declaration

```
  (AkiraNetwork.VirtualStorageLibrary  
  1.html)
```

```
  VirtualItem<T>  
  IEnumerable<string> Patterns { get; }
```

```
  (AkiraNetwork.VirtualStorageLibrary  
  1.html)
```

Property Value

```
  (AkiraNetwork.VirtualStorageLibrary
```

```
  VirtualNodeContext
```

```
  (AkiraNetwork.VirtualStorageLibrary
```

```
  VirtualNodeExtensions
```

```
  (AkiraNetwork.VirtualStorageLibrary
```

```
  VirtualNodeListConditions
```

| Type | Description |
|--|--|
| IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) <string (https://learn.microsoft.com/dotnet/api/system.string)> | An enumerable collection of strings representing the available patterns. |

Library

WildcardDictionaryVirtualStorage

ActionNodeDelegate
Gets a dictionary of wildcards and their corresponding regex patterns.
(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
1.html
    readonly Dictionary<string, string> WildcardDictionary { get; }
    IVirtualWildcardMatcher
        (AkiraNetwork.VirtualStorageLibrary)
```

Property Value

NotifyNodeDelegate

| Type | Description |
|--|---|
| AkiraNetwork.VirtualStorageLibrary PatternMatch ReadOnlyDictionary (AkiraNetwork.VirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.collections.objectmodel.VirtualCycleDetector.readonlydictionary) .readonlydictionary?) <string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)> | A read-only dictionary containing wildcards as keys and their corresponding regex patterns as values. |

Wildcards

VirtualException

Gets (AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

Declaration

```
1.html
virtual and
    IEquatable<string> Wildcards { get; }
```

VirtualItem

VirtualItemAdapter<T>

IEnumerable

1.html
(<https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1>)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

| Type | Description |
|--|---|
| IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) <string (https://learn.microsoft.com/dotnet/api/system.string)> | An enumerable collection of strings representing the available wildcards. |

Methods

| PatternMatcher(string, string) |
|--------------------------------|
|--------------------------------|

Determines whether the specified node name matches the specified pattern.

Declaration

```
bool PatternMatcher(string nodeName, string pattern)
```



Parameters

| Library | Name | Description |
|--|----------|-------------------------|
| Type (AkiraNetwork.VirtualStorageLibrary) | | |
| string (https://learn.microsoft.com/dotnet/api/system.string) ActionNodeDelegate | nodeName | The node name to check. |

| | | |
|--|---------|---------------------------------|
| Type (AkiraNetwork.VirtualStorageLibrary) | pattern | The pattern to compare against. |
|--|---------|---------------------------------|

| |
|-----------------------------------|
| Returns 1.html |
|-----------------------------------|

| Type | Description |
|--|--|
| VirtualWildcardMatcher | True if the node name matches the pattern; otherwise, false. |

Extension Methods

[VirtualDirectory.GenerateSingleTableDebugText<T>\(T\)](#)

([AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilsilites_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_](#))

[\(AkiraNetwork.VirtualStorageLibrary\)](#)
1.html

[VirtualException](#)

[\(AkiraNetwork.VirtualStorageLibrary\)](#)
VirtualGroupCondition<T, TKey>

[\(AkiraNetwork.VirtualStorageLibrary\)](#)
2.html

[VirtualID](#)

[\(AkiraNetwork.VirtualStorageLibrary\)](#)

[VirtualItem](#)

[\(AkiraNetwork.VirtualStorageLibrary\)](#)

[VirtualItemAdapter<T>](#)

[\(AkiraNetwork.VirtualStorageLibrary\)](#)
1.html

[VirtualItem<T>](#)

[\(AkiraNetwork.VirtualStorageLibrary\)](#)
1.html

[VirtualNode](#)

[\(AkiraNetwork.VirtualStorageLibrary\)](#)

[VirtualNodeContext](#)

[\(AkiraNetwork.VirtualStorageLibrary\)](#)

[VirtualNodeExtensions](#)

[\(AkiraNetwork.VirtualStorageLibrary\)](#)

[VirtualNodeListConditions](#)

Delegate NotifyNodeDelegate

A delegate used to notify the status of a specific node during node traversal.

Filter by title

Namespace: AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)
(AkiraNetwork.VirtualStorageLibrary)

Assembly: VirtualStorageLibrary.dll

ActionNodeDelegate

Syntax

```
(AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable<T>
public delegate void NotifyNodeDelegate(VirtualPath path, VirtualNode? node)
(AkiraNetwork.VirtualStorageLibrary.1.html)
```

IVirtualWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|--------------------|------|--|
| NotifyNodeDelegate | | |
| VirtualPath | path | The virtual path of the node being notified. |
| VirtualNode | node | The instance of the node being notified. |

Delegate PatternMatch

A delegate used to determine if a node name matches a pattern.

Filter by title

Namespace: AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)

Assembly: VirtualStorageLibrary.dll

(AkiraNetwork.VirtualStorageLibrary)

Syntax:

```
VirtualDeepCloneable<T>
(AkiraNetwork.VirtualStorageLibrary)
public delegate bool PatternMatch(string nodeName, string pattern)
```

IVirtualWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary)

Parameters

NotifyNodeDelegate

Type (AkiraNetwork.VirtualStorageLibrary)

Name

Description

PatternMatch

string (AkiraNetwork.VirtualStorageLibrary)

nodeName

The name of the node to be matched.

VirtualCycleDetector

string (AkiraNetwork.VirtualStorageLibrary)

pattern

The pattern to match against.

VirtualDirectory

Return VirtualDirectoryAdapter<T>

Type (AkiraNetwork.VirtualStorageLibrary)

Description

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

True if the node name matches the pattern; otherwise, false.

) VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

2.html)

Extension Methods

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary)

VirtualTextFormatter.GenerateSingleTableDebugText_1__0_()

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

Class VirtualCycleDetector

Inheritance

↳ [Object](#) (<https://learn.microsoft.com/dotnet/api/system.object>)

↳ [VirtualCycleDetector](#)
([AkiraNetwork.VirtualStorageLibrary](#))

Inherited Members

[VirtualDeepCloneable<T>](#)

[Object.Equals](#) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-object\)html](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-object)html))

[Object.Equals\(Object\)](#) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-object\)html](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-object)html))

[Object.GetHashCode](#) (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)

[Object.GetType](#) (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)

[Object.MemberwiseClone](#) () (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)

[Object.ReferenceEquals](#) ([VirtualCycleDetector](#)) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)

[Object.ToString](#) () (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)

([AkiraNetwork.VirtualStorageLibrary](#))

Namespace: [AkiraNetwork](#) ([AkiraNetwork.html](#)).[VirtualStorageLibrary](#) ([AkiraNetwork.VirtualStorageLibrary.html](#))

VirtualDirectory

Assembly: [VirtualStorageLibrary.dll](#)
([AkiraNetwork.VirtualStorageLibrary](#))

Syntax

```
VirtualDirectoryAdapter<T>
  (AkiraNetwork.VirtualStorageLibrary)
public class VirtualCycleDetector
  VirtualException
  (AkiraNetwork.VirtualStorageLibrary)
  VirtualGroupCondition<T, TKey>
  (AkiraNetwork.VirtualStorageLibrary)
  2.html)
```

Constructors

VirtualCycleDetector()

([AkiraNetwork.VirtualStorageLibrary](#))

Declaration

```
VirtualItem
  (AkiraNetwork.VirtualStorageLibrary)
VirtualItemAdapter<T>
  public VirtualCycleDetector()
  (AkiraNetwork.VirtualStorageLibrary)
  1.html)
```

```
VirtualItem<T>
  (AkiraNetwork.VirtualStorageLibrary)
```

Properties

VirtualNode

([AkiraNetwork.VirtualStorageLibrary](#))

Count

[VirtualNodeContext](#)

([AkiraNetwork.VirtualStorageLibrary](#))

Declaration

```
VirtualNodeExtensions
  (AkiraNetwork.VirtualStorageLibrary)
public int Count { get; }
```

```
VirtualNodeListConditions
```

([AkiraNetwork.VirtualStorageLibrary](#))

PropertyName

([AkiraNetwork.VirtualStorageLibrary](#))

| Type | Description |
|---|-------------|
| int (https://learn.microsoft.com/dotnet/api/system.int32) | |

CycleDictionary

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    IVirtualDeepCloneable<T>
    public Dictionary<VirtualID, VirtualSymbolicLink> CycleDictionary { get; }
(AkiraNetwork.VirtualStorageLibrary)
1.html)
```

Properties

| Type | Description |
|--|---|
| NotifyNodeDelegate | |
| Dictionary<VirtualID, VirtualSymbolicLink> | https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2 < |
| VirtualPattern | https://learn.microsoft.com/akiranetwork/virtualstorage.library.virtualid.html , |

VirtualSymbolicLink (<https://learn.microsoft.com/akiranetwork/virtualstorage.library.virtualsymboliclink.html>)>

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Clear()

VirtualException

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

2.html)

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

IsNodeInCycle(VirtualSymbolicLink)

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

public bool IsNodeInCycle(VirtualSymbolicLink link)

1.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

Parameters

1.html)

Type

VirtualNode

| Name | Description |
|------|-------------|
| link | |

(AkiraNetwork.VirtualStorageLibrary)

VirtualSymbolicLink (<https://learn.microsoft.com/akiranetwork/virtualstorage.library.virtualsymboliclink.html>)

link

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

Returns

VirtualNodeExtensions

Type

AkiraNetwork.VirtualStorageLibrary

| Description |
|-------------|
| |

VirtualNodeListConditions

bool (<https://learn.microsoft.com/dotnet/api/system.boolean>)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

Extension Methods

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

▼

(AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable<T>
(AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrary.PatternMatch
(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualID
(AkiraNetwork.VirtualStorageLibrary.VirtualItem
(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualNode
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary.VirtualnodeName
(AkiraNetwork.VirtualStorageLibrary.

Class VirtualDirectory

Represents a virtual directory. A virtual directory functions as a container for other nodes. It can contain VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html), VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html), and VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html), providing a centralized way to manage these entities (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html).

This class supports operations such as adding, removing, updating, checking the existence of, and enumerating nodes within the directory. It serves as a fundamental component for efficiently managing the structure of the virtual storage.

NotifyNodeDelegate

Adds (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) according to the specified display conditions.

PatternMatch

Inheritance

VirtualCycleDetector

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

↳ VirtualDirectory

↳ VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)

Implementation

IVirtualDeepCloneable (AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html) <VirtualNode

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)>

IVirtualException

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)>

IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1)<VirtualNode

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)>

IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.ienumerable)

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)

Inherited Members

2.html)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Name)

VirtualNodeCreatedDate

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_CreatedDate)

VirtualItemAdapter<T>

VirtualNodeUpdatedDate

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_UpdatedDate)

VirtualItem<T>

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_VID)

VirtualNodeIsReferencedInStorage

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_IsReferencedInStorage)

VirtualNodeContext

object (https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object))

VirtualNodeExtensions

object (https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-object))

VirtualNodeExtensions

object (https://learn.microsoft.com/dotnet/api/system.object.gethashcode#system-object-gethashcode)

VirtualNodeGetType

(https://learn.microsoft.com/dotnet/api/system.object.gettype#system-object-gettype)

VirtualNodeMemberwiseClone

(https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone#system-object-memberwiseclone)

VirtualNodeNotFoundException

object.ReferenceEquals(object, object) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)

Namespace: AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)

Assembly: VirtualStorageLibrary.dll

Syntax

```
public class VirtualDirectory : VirtualNode, IVirtualDeepCloneable<VirtualNode>, IEnumerable<VirtualNode>, IEnumerable<IVirtualDeepCloneable<T>>
```

Constructors

VirtualDirectoryMatcher

(AkiraNetwork.VirtualStorageLibrary.html)

NotifyNodeDelegate

VirtualDirectory()

(AkiraNetwork.VirtualStorageLibrary.html)

Initializes a new instance of the VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) class.

The node name is automatically generated by the GenerateNodeName(string)

(AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html#AkiraNetwork_VirtualStorageLibrary_VirtualnodeName_GeneratenodeName_System_String_.method).

VirtualDirectory

Declaration

(AkiraNetwork.VirtualStorageLibrary.html)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.html)

1.html)

VirtualException

(AkiraNetwork.VirtualStorageLibrary.html)

VirtualDirectory(VirtualNodeName)

(AkiraNetwork.VirtualStorageLibrary.html)

Initializes a new instance of the VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) class with the specified name.

VirtualName

Declaration

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.html)

```
public VirtualDirectory(VirtualNodeName name)
```

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.html)

Parameters

| Type | Name | Description |
|---|------|----------------------------|
| VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.html) | name | The name of the directory. |

VirtualDirectory(VirtualNodeName, DateTime, DateTime)

VirtualNodeExtensions

Initializes a new instance of the VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) class with the specified name, creation date, and update date.

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.html)

Declaration

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.html)

VirtualNodeNotFoundException

```
public VirtualDirectory(VirtualNodeName name, DateTime createdDate, DateTime updatedDate)
```

Parameters

| Type | | Name | Description |
|--|--|-------------|----------------------------|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | | name | The name of the directory. |
| DateTime (https://learn.microsoft.com/dotnet/api/system.datetime 1.html) | | createdDate | The creation date. |
| DateTime (https://learn.microsoft.com/dotnet/api/system.datetime 1.html) | | updatedDate | The update date. |

Properties

Count
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T>.Count.html)

VirtualCycleDetector

Count
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T>.Count.html)

VirtualDirectory

Gets the total number of nodes within the directory.

VirtualException

Declaration
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T>.Count.html)

public int Count { get; }

VirtualException

Property Value
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T>.Count.html)

Type
(VirtualItem.html)

Description

int
(https://learn.microsoft.com/dotnet/api/system.int32)

(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)

DirectoryCount
(VirtualItemAdapter<T>.DirectoryCount.html)

(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>.DirectoryCount.html)

Gets the number of directories within the directory.
(VirtualItemAdapter<T>.DirectoryCount.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>.DirectoryCount.html)

public int DirectoryCount { get; }

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

Property Value
(VirtualNodeContext.html)

Type
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html)

Description

int
(https://learn.microsoft.com/dotnet/api/system.int32)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html)

VirtualNodeNotFoundException

DirectoryViewCount
(VirtualNodeNotFoundException.html)

| Type | Description |
|---|-------------|
| int (https://learn.microsoft.com/dotnet/api/system.int32) | |

ItemCount

Gets the number of items based on the current display conditions.

IVirtualDeepCloneable<T>

Declaration

[AkiraNetwork.VirtualStorageLibrary](#)

1.html)

public int ItemCount { get; }

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

Property Value

(AkiraNetwork.VirtualStorageLibrary)

Type

PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

int (<https://learn.microsoft.com/dotnet/api/system.int32>)

Description

| Type | Description |
|--------------|-------------|
| PatternMatch | |

VirtualWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

Enum([AkiraNetwork.VirtualStorageLibrary](#))

VirtualDirectoryName

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

IEnumerable (<https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1>)<

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

Gets the type of the node. This class always returns Directory.

1.html)

Declaration

[VirtualNode](#)

(AkiraNetwork.VirtualStorageLibrary)

public override VirtualNodeType NodeType { get; }

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

Property Value

(AkiraNetwork.VirtualStorageLibrary)

Type

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType ([AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html](#))

(AkiraNetwork.VirtualStorageLibrary)

Overrides

VirtualNodeNotFoundException

Description

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Type | Description |
| --- | --- |

<tbl_r cells="2" ix="2" maxcspan

VirtualNode.NodeType
 (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_NodeType)

Nodes

Enumerates all nodes within the directory.

Declaration
 IVirtualDeepCloneable<T>
 (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_IVirtualDeepCloneable_T)
 1.html
 public IEnumerable<VirtualNode> Nodes { get; }
 VirtualWildcardMatcher
 (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_VirtualWildcardMatcher)

Property Value

| Type | Description |
|---------------------------------------|--|
| PatternMatch.IEnumerable<VirtualNode> | (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_IEnumerable_VirtualNode) |

NodesView

Gets a view of nodes based on the current display conditions.
 1.html

Declaration

VirtualGroupCondition<VirtualNode> NodesView { get; }
 (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_VirtualGroupCondition_VirtualNode_NodesView)
 2.html

Property Value

| Type | Description |
|---------------------------------|---|
| VirtualItemAdapter<VirtualNode> | (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_VirtualItemAdapter_VirtualNode) |

NodesViewCount

Gets the number of nodes based on the current display conditions.
 1.html

Declaration

VirtualNodeContext NodesViewCount { get; }
 (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeContext_NodesViewCount)
 VirtualNodeExtensions
 (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeExtensions)

Property Value

| Type | Description |
|-----------------|---|
| VirtualNodeName | int (https://learn.microsoft.com/dotnet/api/system.int32) |

SymbolicLinkCount

Gets the number of symbolic links within the directory.

Declaration

```
public int SymbolicLinkCount { get; }
```

| Property Value | Type |
|---|---|
| IVirtualDeepCloneable<T> (AkiraNetwork.VirtualStorageLibrary.VirtualDeepCloneable<T>.html) | int (https://learn.microsoft.com/dotnet/api/system.int32) (AkiraNetwork.VirtualStorageLibrary.VirtualInt32.html) |

| Property Value | Type | Description |
|---|---|-------------|
| IVirtualWildcardMatcher (AkiraNetwork.VirtualStorageLibrary.VirtualWildcardMatcher.html) | int (https://learn.microsoft.com/dotnet/api/system.int32) (AkiraNetwork.VirtualStorageLibrary.VirtualInt32.html) | |

SymbolicLinkViewCount

Gets the number of symbolic links based on the current display conditions.

| Declaration |
|---|
| AkiraNetwork.VirtualStorageLibrary.VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) |
| public int SymbolicLinkViewCount { get; } |

| Property Value | Type | Description |
|--|---|-------------|
| VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualException.html) | int (https://learn.microsoft.com/dotnet/api/system.int32) (AkiraNetwork.VirtualStorageLibrary.VirtualInt32.html) | |

| Method |
|--|
| VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) |
| VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualID.html) |

| Add(VirtualNode, bool) |
|---|
| AkiraNetwork.VirtualStorageLibrary.VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) |

| Declaration |
|--|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) |

| Parameters |
|--|
| VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html) |

| Type | Name | Description |
|--|------|------------------|
| VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html) | node | The node to add. |

| Type | Name | Description |
|---|-----------------------|--|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | <i>allowOverwrite</i> | If true, allows overwriting an existing node with the same name. |

Returns

| Type | Description |
|---|-----------------|
| <code>VirtualDeepCloneable<T></code> (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | The added node. |

Exceptions
IVirtualWildcardMatcher
([AkiraNetwork.VirtualStorageLibrary](#))

| Type | Condition |
|--|---|
| <code>NotifyNodeDelegate</code> (AkiraNetwork.VirtualStorageLibrary) <code>ArgumentException</code> (PatternMatch (https://learn.microsoft.com/dotnet/api/system.argumentexception) (AkiraNetwork.VirtualStorageLibrary) | Thrown if an invalid node name is specified. |
| <code>InvalidOperationException</code> (VirtualDirectory (https://learn.microsoft.com/dotnet/api/system.invalidoperationexception) <code>VirtualDirectory</code> (AkiraNetwork.VirtualStorageLibrary) <code>VirtualDirectoryAdapter<T></code> (AkiraNetwork.VirtualStorageLibrary) | Thrown if a node with the same name already exists. |

AddDirectory(VirtualNodeName, bool)

`VirtualException`
Adds a directory with the specified name.
([AkiraNetwork.VirtualStorageLibrary](#))

Declaration
`GroupCondition<T, TKey>`

`public VirtualDirectory AddDirectory(VirtualNodeName name, bool allowOverwrite = false)`
VirtualID
([AkiraNetwork.VirtualStorageLibrary](#))

Parameters

| Type | Name | Description |
|---|-----------------------|--|
| <code>AkiraNetwork.VirtualStorageLibrary</code> <code>VirtualItemAdapter<T></code> <code>VirtualNodeName</code> (AkiraNetwork.VirtualStorageLibrary) (AkiraNetwork.VirtualStorageLibrary) (VirtualNodeName.html) <code>VirtualItem<T></code> | <i>name</i> | The name of the directory. |
| <code>bool</code> (https://learn.microsoft.com/dotnet/api/system.boolean) <code>VirtualNode</code> (AkiraNetwork.VirtualStorageLibrary) <code>VirtualNodeContext</code> (AkiraNetwork.VirtualStorageLibrary) | <i>allowOverwrite</i> | If true, allows overwriting an existing node with the same name. |

Returns

| Type | Description |
|---|----------------------|
| <code>VirtualNodeExtensions</code> (AkiraNetwork.VirtualStorageLibrary) <code>VirtualDirectory</code> (AkiraNetwork.VirtualStorageLibrary) <code>VirtualNodeListConditions</code> (AkiraNetwork.VirtualStorageLibrary) <code>VirtualNodeName</code> (AkiraNetwork.VirtualStorageLibrary) | The added directory. |

AddItem<T>(VirtualNodeName, T?, bool)

`VirtualNodeNotFoundException`

Adds an item with the specified name.

Declaration

```
public VirtualItem<T> AddItem<T>(VirtualNodeName name, T? itemData = default, bool allowOverwrite = false)
```



Parameters

| VirtualDeepCloneable<T> | |
|---|-----------------------------|
| Type <code>AkiraNetwork.VirtualStorageLibrary</code> | Name |
| <code>VirtualNodeName</code> <code>VirtualWildcardMatcher</code> (<code>AkiraNetwork.VirtualStorageLibrary</code> . <code>VirtualNodeName</code>) <code>VirtualNodeDelegate</code> | <code>name</code> |
| <code>T</code> (<code>AkiraNetwork.VirtualStorageLibrary</code>) <code>PatternMatch</code> <code>bool</code> (<code>AkiraNetwork.VirtualStorageLibrary</code>) (https://learn.microsoft.com/dotnet/api/system.boolean) <code>VirtualCycleDetector</code> <code>VirtualStorageLibrary</code> | <code>itemData</code> |
| | <code>allowOverwrite</code> |

Returns

`AkiraNetwork.VirtualStorageLibrary`

| VirtualDirectoryAdapter<T> | |
|--|-----------------|
| <code>VirtualStorageLibrary</code> <code>VirtualItem</code> (<code>AkiraNetwork.VirtualStorageLibrary</code> . <code>VirtualItem-1.html</code>) <T> | The added item. |

Exceptions

`AkiraNetwork.VirtualStorageLibrary`

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

Return Value

`AkiraNetwork.VirtualStorageLibrary`

`VirtualItem` (`AkiraNetwork.VirtualStorageLibrary`)

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| VirtualGroupCondition<T, TKey> | |
|------------------------------------|----------------------------|
| <code>T</code> <code>2.html</code> | The data type of the item. |

| Name | Description |
| --- | --- |

<tbl_r cells="1" ix="1" maxcspan="2" maxrspan="1" usedcols="


```
public bool DirectoryExists(VirtualnodeName name)
```

Parameters

| Type | | Name | Description |
|--|--|------|-------------------------------------|
| VirtualnodeName (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) | | name | The name of the directory to check. |

Returns `bool`

| Type | IVirtualWildcardMatcher (AkiraNetwork.VirtualStorageLibrary.VirtualWildcardMatcher.html) | Description |
|---|---|---|
| bool (http://dotnet.developersoftware.com/dotnet/api/system.boolean) | | True if the directory exists; otherwise, false. |

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter.html)

PatternMatch

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter.PatternMatch.html)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html)

Gets the node with the specified name.

VirtualDirectory

Declaration

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html)

VirtualDirectoryAdapter<T>

```
public VirtualDirectory<T> Get(VirtualnodeName name, bool exceptionEnabled = true)
```

1.html

VirtualException

Parameters

(AkiraNetwork.VirtualStorageLibrary.VirtualException.html)

| Type | VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition.html) | Name | Description |
|--|---|------------------|--|
| VirtualnodeName (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) | | name | The name of the node. |
| bool (http://dotnet.developersoftware.com/dotnet/api/system.boolean) | VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter.html) | exceptionEnabled | If true, throws an exception if the node is not found. |

Returns

VirtualItem<T>

| Type | AkiraNetwork.VirtualStorageLibrary.VirtualItem<T> 1.html | Description |
|--|--|---|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html) | The node with the specified name, or null if the node does not exist. |

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html)

GetDirectory(VirtualnodeName)

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html)

Gets the directory with the specified name.

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html)

Declaration

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html)

VirtualNodeNotFoundException

```
public VirtualDirectory GetDirectory(VirtualNodeName name)
```

Parameters

| Type | Name | Description |
|---|------|----------------------------|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | name | The name of the directory. |

Returns

| Type | Description |
|---|--|
| IVirtualWildcardMatcher (AkiraNetwork.VirtualStorageLibrary.VirtualWildcardMatcher.html) | The directory with the specified name. |

Exceptions

| Type | Condition |
|--|--|
| VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html) | |
| VirtualNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNotFoundException.html) | Thrown if the node is not found. |
| VirtualDirectoryAdapter<T> InvalidOperationException (AkiraNetwork.VirtualStorageLibrary.InvalidOperationException.html) | Thrown if the specified node is not a directory. |

GetEnumerator()

Returns

Gets an enumerator for nodes based on the current display conditions.

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
```

VirtualItem

```
public System.IEnumerable<VirtualNode> GetEnumerator()
```

VirtualItemAdapter<T>

```
(AkiraNetwork.VirtualStorageLibrary)
```

Returns

1.html

Type

VirtualItem<T>

```
(AkiraNetwork.VirtualStorageLibrary)
```

1.html

IEnumerator

1.html

(https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerator-1)

<VirtualNode>(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

```
(AkiraNetwork.VirtualStorageLibrary)
```

VirtualNodeContext

```
(AkiraNetwork.VirtualStorageLibrary)
```

GetItemByName(VirtualNodeName)

```
(AkiraNetwork.VirtualStorageLibrary)
```

Gets the item with the specified name.

VirtualNodeListConditions

```
(AkiraNetwork.VirtualStorageLibrary)
```

VirtualNodeName

```
(AkiraNetwork.VirtualStorageLibrary)
```

VirtualNodeNotFoundException

```
public VirtualItem<T> GetItem<T>(VirtualNodeName name)
```

Parameters

| Type | Name | Description |
|---|------|-----------------------|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) IVirtualDeepCloneable<T> | name | The name of the item. |

Return Value

| Type | Description |
|--|-----------------------------------|
| Virt (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) <T> | The item with the specified name. |

Type Parameters

| Type | Parameters | Description |
|------|----------------------|--|
| | PatternMatch | (AkiraNetwork.VirtualStorageLibrary) |
| T | VirtualCycleDetector | The data type of the item. (AkiraNetwork.VirtualStorageLibrary) |

Exceptions

| Type | Condition |
|--|---|
| VirtualDirectoryAdapter<T> | Thrown if the node is not found. (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNotFoundException 1.html (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html) VirtualException | Thrown if the specified node is not an item. (https://learn.microsoft.com/dotnet/api/system.invalidoperationexception) (AkiraNetwork.VirtualStorageLibrary) |

GetNodesView()

VirtualItem

Gets a view of nodes based on the current display conditions.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

```
public IEnumerable<VirtualNode> GetNodesView()
```

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

Returns

T.html

| Type | Description |
|-------------|--|
| VirtualNode | An enumeration of nodes based on the display conditions. (AkiraNetwork.VirtualStorageLibrary) IEnumerable VirtualNodeContext (https://learn.microsoft.com/dotnet/api/system.collections.generic.IEnumerable-1) VirtualNodeExtensions <VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)> (AkiraNetwork.VirtualStorageLibrary) |

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

GetSymbolicLink(VirtualNodeName)

(AkiraNetwork.VirtualStorageLibrary)

Gets the symbolic link with the specified name.

VirtualNodeNotFoundException

Declaration

```
public VirtualSymbolicLink GetSymbolicLink(VirtualNodeName name)
```

Parameters

| Type | Name | Description |
|---|------|--------------------------------|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) 1.html) | name | The name of the symbolic link. |

Returns

| Type | Description |
|---|--|
| VirtualWildcardMatcher (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate) | The symbolic link with the specified name. |

Exceptions

| Type | Condition |
|--|--|
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException) | Thrown if the node is not found. |
| VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualGroupConditionAdapter<T>) | Thrown if the specified node is not a symbolic link. |

ItemExists(VirtualNodeName)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions)

Checks whether an item with the specified name exists.

(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>)

```
public bool ItemExists(VirtualNodeName name)
```

(AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions)

Parameters

| Type | Name | Description |
|--|------|--------------------------------|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext) | name | The name of the item to check. |

Returns

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions)

| Type | Description |
|---|--|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | True if the item exists; otherwise, false. |

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException)

NodeExists(VirtualNodeName)

Checks whether a node with the specified name exists.

Declaration

```
    public bool NodeExists(VirtualnodeName name)
```

| Parameters | <code>VirtualDeepCloneable<I></code> (AkiraNetwork.VirtualStorageLibrary.html) | | |
|------------|---|-------------|--------------------------------|
| Type | | Name | Description |
| | <code>IVirtualWildcardMatcher</code> (AkiraNetwork.VirtualStorageLibrary.html) | | |
| | <code>VirtualNodeName</code> (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | <i>name</i> | The name of the node to check. |
| | <code>NotifyNodeDelegate</code> | | |

| Returns | (AkiraNetwork.VirtualStorageLibrary.PatternMatch) |
|----------------|---|
| Type | AkiraNetwork.VirtualStorageLibrary.PatternMatch |

Remove Virtual Node

(AkiraNetwork.VirtualStorageLibrary)

RemoveNode(*node*)

VirtualException Declaration

(AkiraNetwork VirtualStorage) library

```
VirtualGroupCondition<T, TKey>
public void Remove(VirtualNode node)
(AkiraNetwork VirtualStorage) library
```

3.html)

| Parameter ID | Type | Name | Description |
|--|------|------|---------------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualItem) | | | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNode) | | node | The node to remove. |

| Type | Condition |
|--|----------------------------------|
| VirtualItem<T> | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html) | Thrown if the node is not found. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html) | |

SymbolicLinkExists(VirtualNodeName)

VirtualNodeExtensions
Checks whether a symbolic link with the specified name exists.
([libNetwork/virtualStorageLibrary](#))

```
Declaration NodeListConditions  
    (AkiraNetwork.VirtualStorageLibrary)  
  
VirtualNodeExists  
    pVirtualNodeSymbolicLinkExists(VirtualNodeName name)  
    (AkiraNetwork.VirtualStorageLibrary)  
  
VirtualNodeNotFoundException
```

Parameters

| Type | Name | Description |
|--|------|---|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | name | The name of the symbolic link to check. |

Returns

| Type | Description |
|---|---|
| IVirtualDeepCloneable<T> (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | True if the symbolic link exists; otherwise, false. |

IVirtualWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary.VirtualNode)

>Returns a string representation of the directory.
PatternMatch

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector  
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory  
public override string ToString()  
VirtualDirectory  
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T>
```

Returns

| Type | Description |
|--|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T>.html) | A string representation of the directory. |

Overrides

```
object.ToString() (https://learn.microsoft.com/dotnet/api/system.object.tostring)  
2.html
```

VirtualID

Update(VirtualNode)

(AkiraNetwork.VirtualStorageLibrary.VirtualItem)

Updates the current directory with the data from the specified node.

(AkiraNetwork.VirtualStorageLibrary.VirtualNode)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>  
1.html)  
public override void Update(VirtualNode node)  
VirtualItem<T>  
(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>)
```

Parameters

| Type | Name | Description |
|--|------|-------------------------------|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | node | The node used for the update. |

Overrides

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.UpdateVirtualNode)

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Update_AkiraNetwork_VirtualStorageLibrary_VirtualNode_)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName)

VirtualNodeNotFoundException

Exceptions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException)

| Type | Condition |
|--|---|
| ArgumentException (https://learn.microsoft.com/dotnet/api/system.argumentexception) | Thrown if the specified node is not a VirtualDirectory. |

1

Operators

VirtualDeep

IVirtualDeepCloneable<T>

(AkiraNetwork.VirtualStorageLibrary

1.html) + VirtualDir

operator + (VirtualDirectory, VirtualNode)
IVirtualWildcardMatcher
Overlaid the additional VirtualStrategy added a node to the directory.

NotifyNodeDelegate

Declaration Akira Network Virtual Storage library

(AKTIONEN)NETZWERK

```
public static VirtualDirectory operator +(VirtualDirectory directory, VirtualNode node)
```

VirtualCycleDetector

VirtualCycleDetector

parameters

| Type | Name | Description |
|--|------------------|---|
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary) | | |
| VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary) | <i>directory</i> | The directory to which the node is added. |
| VirtualNode (AkiraNetwork.VirtualStorageLibrary) | <i>node</i> | The node to add. |

VirtualGroupCondition<T, TKey>

| Type | Description |
|-----------|------------------------------------|
| VirtualID | The directory with the added node. |

implicit operator VirtualDirectory(VirtualDirectoryName)

1.html) Performs an implicit conversion from VirtualNodeName

(AkiryaItemToVirtualStorageLibraryVirtualNodeName.html) to VirtualDirectory.

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html
(AkiraNetwork.VirtualStorageLibrary.html).
1.html)

Declaration Node

(AkiraNetwork VirtualStorage) library

```
nVirtualNodeContexticit operator VirtualDirectory(VirtualNodeName nodeName)
```

(AkiraNetwork VirtualStorage) library

VirtuAlNodeExtensions

Parameters `VirtualNetworkVirtualInterfaceLibrary`

| Type | Name | Description |
|--|-----------------|---------------------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions) | | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName) | <i>nodeName</i> | The node name to convert. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | | |
| (AkiraNetwork.VirtualStorageLibrary) | | |

VirtualNodeNotFoundException

Returns

| Type | Description |
|---|-------------|
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) | |

operator -(VirtualDirectory, VirtualNode)

|VirtualDeepCloneable<T>
Overloads the subtraction operator to remove a node from the directory.
(AkiraNetwork.VirtualStorageLibrary)

Declaration

IVirtualWildcardMatcher
public static VirtualDirectory operator -(VirtualDirectory directory, VirtualNode node)

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary)

Parameters

PatternMatch

| Type | Name | Description |
|--|-----------|---|
| AkiraNetwork.VirtualStorageLibrary VirtualCycleDetector | directory | The directory from which the node is removed. |
| AkiraNetwork.VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) | node | The node to remove. |

Returns

VirtualException

| Type | Description |
|--|--------------------------------------|
| AkiraNetwork.VirtualStorageLibrary VirtualGroupCondition<T, TKey> VirtualDirectory (AkiraNetwork.VirtualStorageLibrary) (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) | The directory with the removed node. |

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

Implements

| | |
|--------------------------|---|
| IVirtualDeepCloneable<T> | (AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html) |
| IEnumerable<T> | (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) |
| IEnumerable | (https://learn.microsoft.com/dotnet/api/system.collections.ienumerable) |
| VirtualItem<T> | |

| |
|---|
| VirtualTextFormatter.GenerateSingleTableDebugText<T>(T) |
| VirtualNode |

| |
|---|
| (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilsilites_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_) |
| VirtualNodeContext |

| |
|---|
| VirtualNodeExtensions.ResolveNodeType(VirtualNode) |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Extensions_ResolveNodeType_AkiraNetwork_VirtualStorageLibrary_VirtualNode_) |

| |
|--|
| VirtualTextFormatter.GenerateTableDebugText<T>(IEnumerable<T>) |
| VirtualNodeListConditions |

| |
|--|
| (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilsilites_VirtualTextFormatter_GenerateTableDebugText_1_System_Collections_Generic_IEnumerable__0_) |
| VirtualNodeName |

| |
|---|
| VirtualStorageExtensions.ApplySortConditions<T>(IEnumerable<T>, List<VirtualSortCondition<T>>?) |
| (AkiraNetwork.VirtualStorageLibrary) |

| |
|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualStorageExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualStorageExtensions_VirtualNodeNotFoundException) |
|---|

```
rageExtensions_ApplySortConditions_1_System_Collections_Generic_IEnumerator_0_System_Collections_Generi
c_List_AkiraNetwork_VirtualStorageLibrary_VirtualSortCondition_0__)
VirtualStorageExtensions.GroupAndSort<T>(IEnumerable<T>, VirtualGroupCondition<T, object>?, List<Virtual
SortCondition<T>>?)
(AkiraNetwork.VirtualStorageLibrary.VirtualStorageExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualSto
rageExtensions_GroupAndSort_1_System_Collections_Generic_IEnumerator_0_AkiraNetwork_VirtualStorageLibr
ary_VirtualGroupCondition_0_System_Object_System_Collections_Generic_List_AkiraNetwork_VirtualStorageLibr
ary_VirtualSortCondition_0__)
    IVirtualDeepCloneable<T>
        (AkiraNetwork.VirtualStorageLibrar
        1.html)
    IVirtualWildcardMatcher
        (AkiraNetwork.VirtualStorageLibrar
    NotifyNodeDelegate
        (AkiraNetwork.VirtualStorageLibrar
    PatternMatch
        (AkiraNetwork.VirtualStorageLibrar
    VirtualCycleDetector
        (AkiraNetwork.VirtualStorageLibrar
    VirtualDirectory
        (AkiraNetwork.VirtualStorageLibrar
    VirtualDirectoryAdapter<T>
        (AkiraNetwork.VirtualStorageLibrar
        1.html)
    VirtualException
        (AkiraNetwork.VirtualStorageLibrar
    VirtualGroupCondition<T, TKey>
        (AkiraNetwork.VirtualStorageLibrar
        2.html)
    VirtualID
        (AkiraNetwork.VirtualStorageLibrar
    VirtualItem
        (AkiraNetwork.VirtualStorageLibrar
    VirtualItemAdapter<T>
        (AkiraNetwork.VirtualStorageLibrar
        1.html)
    VirtualItem<T>
        (AkiraNetwork.VirtualStorageLibrar
        1.html)
    VirtualNode
        (AkiraNetwork.VirtualStorageLibrar
    VirtualNodeContext
        (AkiraNetwork.VirtualStorageLibrar
    VirtualNodeExtensions
        (AkiraNetwork.VirtualStorageLibrar
    VirtualNodeListConditions
        (AkiraNetwork.VirtualStorageLibrar
    VirtualnodeName
        (AkiraNetwork.VirtualStorageLibrar
    VirtualNodeNotFoundException
```

Class VirtualDirectoryAdapter<T>

Inheritance

Filter by title ↴ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary)

Inherited Members 1.html)

[`object.Equals\(object, object\)`](https://docs.microsoft.com/en-us/dotnet/api/system.object.equals#system-object-equals(system-object,-system-object))

`object.Equals(object)` (<https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals>)

`object.GetHashCode()` (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)

object (Akavai Network Virtual Storage Library <http://www.akavai.com/dotnet/api/system.object.gettype>)

`object.MemberwiseClone()` (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)

[Object.ReferenceEquals](https://learn.microsoft.com/dotnet/api/system.object.referenceequals)(**array1**, **array2**)

object.ToString() (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)

(AkiraNetwork-VirtualStorageLibrary) Page 10 of 10 (Akira Network Virtual Storage Library)

Namespace: AkiraNetwork ([AkiraNetwork.html](#)).VirtualStorageLibrary ([AkiraNetwork.VirtualStorageLibrary.html](#))

Syntax

VirtualException

`public class NetworkVirtualStorageAdapter<T>`

VirtualGroupCondition<T, TKey>

Type Parameters

| Name | Description |
|-----------|--|
| VirtualID | (AkiraNetwork.VirtualStorageLibrary.VirtualItem) |

VirtualItemAdapt Constructors

CU(AkriaNetworkVirtualStorageLibrary
1.html)

Virtually

Virtual Director
(Akira Network)

VirtualDirectoryAdapter(VirtualStorage<T>)

Declarative HTML

Declaration

VirtualNode

```
public VirtualDirectoryAdapter<VirtualStorage<T> storage>
    VirtualNodeContext
```

(AkiraNetwork.VirtualStorageLibrary)

Parameters

| Type | Name | Description |
|---|---------|-------------|
| (AkiraNetwork.VirtualStorageLibrary.Virtual NodeListConditions | | |
| VirtualStorage (AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html) <T> | storage | |
| VirtualNodeName | | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException | | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotReachableException | | |

Properties

this[VirtualPath, bool]

Declaration

```
public VirtualDirectory this[VirtualPath path, bool followLinks = true] { get; set; }  
(AkiraNetwork.VirtualStorageLibrary  
1.html)
```

Parameters

| Type | Description |
|---|-------------|
| AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>path</i> |

bool (https://learn.microsoft.com/dotnet/api/system.boolean)

| Type | Description |
|--|-------------|
| VirtualDirectory | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) | |

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary
1.html)

Extension Methods

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)
(AkiraNetwork.VirtualStorageLibrary.Uses.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilities_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

2.html

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

Class VirtualException

The base class for exceptions that occur within the virtual storage library.

Filter by title

Inheritance

1.html

↳ object (<https://learn.microsoft.com/dotnet/api/system.object>)
↳ VirtualWildcardMatcher
↳ Exception (<https://learn.microsoft.com/dotnet/api/system.exception>)
 (AkiraNetwork.VirtualStorageLibrary)
 ↳ VirtualException
 ↳ NotifyNodeDelegate
 ↳ VirtualNodeNotFoundException
 (AkiraNetwork.VirtualStorageLibrary)
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html)
 ↳ PatternMatch

Implements

ISerializable (<https://learn.microsoft.com/dotnet/api/system.runtime.serialization.iserializable>)
 (AkiraNetwork.VirtualStorageLibrary)

Inherited Members

VirtualDirectory

Exception (GetBaseException) (<https://learn.microsoft.com/dotnet/api/system.exception.getbaseexception>)
Exception (GetType) (<https://learn.microsoft.com/dotnet/api/system.exception.gettype>)
Exception (ToString) (<https://learn.microsoft.com/dotnet/api/system.exception.tostring>)
Exception (Data) (<https://learn.microsoft.com/dotnet/api/system.exception.data>)
Exception (HelpLink) (<https://learn.microsoft.com/dotnet/api/system.exception.helplink>)
Exception (HRESULT) (<https://learn.microsoft.com/dotnet/api/system.exception.hresult>)
Exception (InnerException) (<https://learn.microsoft.com/dotnet/api/system.exception.innerexception>)
Exception (Message) (<https://learn.microsoft.com/dotnet/api/system.exception.message>)
Exception (Source) (<https://learn.microsoft.com/dotnet/api/system.exception.source>)
Exception (StackTrace) (<https://learn.microsoft.com/dotnet/api/system.exception.stacktrace>)
Exception (TargetSite) (<https://learn.microsoft.com/dotnet/api/system.exception.targetsite>)
Exception (SerializeObjectState) (<https://learn.microsoft.com/dotnet/api/system.exception.serializeobjectstate>)
object (Equals) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object)))
object (VirtualItemAdapter<T>) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object-t\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object-t)))
object (GetHashCode) (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)
object (MemberwiseClone) (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)
object (ReferenceEquals) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)

Namespace: AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)
 (AkiraNetwork.VirtualStorageLibrary)

Assembly: VirtualStorageLibrary.dll

VirtualNodeContext

Syntax

(AkiraNetwork.VirtualStorageLibrary)

 VirtualNodeExtensions

public abstract class VirtualException : Exception, ISerializable

 VirtualNodeListConditions

 (AkiraNetwork.VirtualStorageLibrary)

 VirtualNodeName

 (AkiraNetwork.VirtualStorageLibrary)

 VirtualNodeNotFoundException

 (AkiraNetwork.VirtualStorageLibrary)

VirtualException()

 VirtualNodeType

Constructors

Initializes a new instance of the VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualException.html) class.

Declaration

```
public VirtualException()
```



VirtualException(string)

IVirtualWildcardMatcher

Initializes a new instance of the VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualException.html) class with a specified error message.

NotifyNodeDelegate

Declaration

PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

```
public VirtualException(string message)
```

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

Parameters

VirtualDirectory

| Type | Name | Description |
|--|------|-------------|
| (AkiraNetwork.VirtualStorageLibrary) VirtualDirectoryAdapter<T> | | |

string (https://learn.microsoft.com/dotnet/api/system.string)

message

The error message.

1.html)

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualException(string, Exception)

VirtualGroupCondition<T, Key>

Initializes a new instance of the VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualException.html) class with a specified error message and a reference to the inner exception that is the cause of this exception.

VirtualID

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

```
public VirtualException(string message, Exception innerException)
```

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

Parameters

1.html)

| Type | Name | Description |
|--|------|-------------|
| (AkiraNetwork.VirtualStorageLibrary) VirtualItem<T> | | |

string

(AkiraNetwork.VirtualStorageLibrary)

message

The error message.

1.html)

(https://learn.microsoft.com/dotnet/api/system.string)

VirtualNode

Exception

(AkiraNetwork.VirtualStorageLibrary)

(https://learn.microsoft.com/dotnet/api/system.exception)

(AkiraNetwork.VirtualStorageLibrary)

innerException

The exception that is the cause of this exception.

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

ISerializable

(System.Runtime.Serialization.ISerializable)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

Extension Methods

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)



1.html)
IVirtualWildcardMatcher
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
PatternMatch
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
1.html)
VirtualException
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
2.html)
VirtualID
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualItem
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
1.html)
VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
1.html)
VirtualNode
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualnodeName
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualNodeType

Class VirtualGroupCondition<T, TKey>

Represents the conditions for grouping data, holding the property to group by and the order (ascending or descending) by title

Inheritance

(AkiraNetwork.VirtualStorageLibrary.VirtualWildcardMatcher)
↳ object (https://learn.microsoft.com/dotnet/api/system.object)
 ↳ NotifyNodeDelegate
 ↳ VirtualGroupCondition<T, TKey>
 (AkiraNetwork.VirtualStorageLibrary)

Inherited Members

(AkiraNetwork.VirtualStorageLibrary.object.Equals(object))
object.Equals(object, object) (https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object))
(AkiraNetwork.VirtualStorageLibrary.object.Equals(object, object))
object.Equals(object, object) (https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object))
(AkiraNetwork.VirtualStorageLibrary.object.GetHashCode())
object.GetHashCode() (https://learn.microsoft.com/dotnet/api/system.object.gethashcode)
VirtualDirectoryAdapter<T>
object.GetType() (https://learn.microsoft.com/dotnet/api/system.object.gettype)
(AkiraNetwork.VirtualStorageLibrary.object.MemberwiseClone())
object.MemberwiseClone() (https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone)
1.html
object.ReferenceEquals(object, object) (https://learn.microsoft.com/dotnet/api/system.object.referenceequals)
VirtualException
object.ToString() (https://learn.microsoft.com/dotnet/api/system.object.tostring)
(AkiraNetwork.VirtualStorageLibrary)

Namespace: AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)

Assembly: AkiraNetwork.VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.dll)

Syntax

```
VirtualID  
(AkiraNetwork.VirtualStorageLibrary.  
public class VirtualGroupCondition<T, TKey>  
VirtualItem  
(AkiraNetwork.VirtualStorageLibrary)
```

Type Parameters

| Name | Description |
|------|---|
| T | VirtualItem<T> The type of the entity to be grouped. |
| TKey | (AkiraNetwork.VirtualStorageLibrary) The type of the key used for grouping. |

Constructors

VirtualGroupCondition<Expression<Func<T, TKey>>, bool>
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<Expression<Func<T, TKey>>, bool>
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions
Initializes a new instance of the VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary) class with the specified grouping property and order:
VirtualNodeName
(AkiraNetwork.VirtualStorageLibrary)

Declaration
VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary)

```
public VirtualGroupCondition(Expression<Func<T, TKey>> groupBy, bool ascending = true)
```

Parameters

| Type | Name | Description |
|--|-----------|---|
| Expression<Func<T, TKey>> (https://learn.microsoft.com/dotnet/api/system.linq.expressions.expression) Network.VirtualStorageLibrary<Func<T, TKey>> (https://learn.microsoft.com/dotnet/api/system.func-2)<T, TKey> (AkiraNetwork.VirtualStorageLibrary) | groupBy | The property to group by. |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary) | ascending | A value indicating whether the grouping order is ascending. |

Properties

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Ascending

(AkiraNetwork.VirtualStorageLibrary)

Gets or sets a value indicating whether the grouping order is ascending. True if the order is ascending; otherwise, false.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualID

```
public bool Ascending { get; set; }
```

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

Property Value

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

bool

VirtualItem<T>
(<https://learn.microsoft.com/dotnet/api/system.boolean>)

(AkiraNetwork.VirtualStorageLibrary)

lean)

1.html)

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

GroupBy

(AkiraNetwork.VirtualStorageLibrary)

Gets the `VirtualNode` expression used for grouping.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

```
public Expression<Func<T, TKey>> GroupBy { get; set; }
```

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

Property Value

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

Description

A boolean value indicating whether the grouping order is ascending.

| Type | Description |
|--|--|
| Expression (https://learn.microsoft.com/dotnet/api/system.linq.expressions.expression-1) <Func (https://learn.microsoft.com/dotnet/api/system.func-2)<T, TKey>> | An expression that specifies the property to use for grouping. |

Extension Methods

| | |
|---|--------------------------------------|
| IVirtualWildcardMatcher | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeDelegateGenerateSingleTableDebugText<T>(T) | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualTextFormatter<T>(T) | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualTextFormatter.GenerateSingleTableDebugText_1__0_() | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualCycleDetector | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualDirectory | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualDirectoryAdapter<T> | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualException | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualGroupCondition<T, TKey> | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualID | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualItem | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualItemAdapter<T> | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualItem<T> | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNode | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeContext | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeExtensions | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeListConditions | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualnodeName | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeNotFoundException | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeType | (AkiraNetwork.VirtualStorageLibrary) |

Struct VirtualID

A structure representing a unique identifier within the virtual storage library. This identifier is based on a randomly generated UUID version 4 (UUIDv4). UUIDv4 is a 128-bit value that ensures high uniqueness and has strong cryptographic properties.

(AkiraNetwork.VirtualStorageLibrary)

Notifies node delegate

Implements

(AkiraNetwork.VirtualStorageLibrary)

IEquatable<VirtualID>

(AkiraNetwork.VirtualStorageLibrary)

PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary)

Inherited Members

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

ValueObject<object> (https://learn.microsoft.com/dotnet/api/system.valuetype.equals)

ValueType.GetHashCode() (https://learn.microsoft.com/dotnet/api/system.valuetype.gethashcode)

object.Equals(object) (https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-

equals(system.Object, system.Object))

object.GetType() (https://learn.microsoft.com/dotnet/api/system.object.gettype)

object.ReferenceEquals(object, object) (https://learn.microsoft.com/dotnet/api/system.object.referenceequals)

VirtualException

Namespace: AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)

(AkiraNetwork.VirtualStorageLibrary)

Assembly: VirtualStorageLibrary.dll

VirtualGroupCondition<T, TKey>

Syntax

(AkiraNetwork.VirtualStorageLibrary)

2.html)

public readonly record struct VirtualID : IEquatable<VirtualID>

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualID0

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

Initializes a new instance of the VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualID.html) structure, generating

a unique identifier.

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

Name

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

Gets the unique identifier.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

(AkiraNetwork.NodeTypeFilter)

NodeFilter

</

```
public Guid ID { get; }
```

Property Value

| Type | Description |
|--|--|
| Guid (https://learn.microsoft.com/dotnet/api/system.guid) AkiraNetwork.VirtualStorageLibrary /system/guid VirtualNodeDelegate | A Guid (https://learn.microsoft.com/dotnet/api/system.guid) value representing the unique identifier. |

Methods

`ToString()`

Returns a string representation of this instance's unique identifier.
(<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualID.ToString>)

1.html)

Declaration

VirtualException

```
(AkiraNetwork.VirtualStorageLibrary  
public override string ToString()  
VirtualGroupCondition<T, TKey>)
```

(AkiraNetwork.VirtualStorageLibrary
VirtualID)

2.html)

Returns

VirtualID

(AkiraNetwork.VirtualStorageLibrary
VirtualID)

Description

string
(<https://learn.microsoft.com/dotnet/api/system.string>)
(AkiraNetwork.VirtualStorageLibrary
VirtualID)

A string that represents the unique identifier.

Overrides

(AkiraNetwork.VirtualStorageLibrary
ValueType.ToString()) (<https://learn.microsoft.com/dotnet/api/system.valuetype.tostring>)
1.html)

VirtualItem<T>

Implements

(AkiraNetwork.VirtualStorageLibrary
VirtualNode)

1.html)

IEquatable<T> (<https://learn.microsoft.com/dotnet/api/system.iequatable-1>)

VirtualNode

(AkiraNetwork.VirtualStorageLibrary
VirtualNode)

Extension Methods

(AkiraNetwork.VirtualStorageLibrary
VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary
VirtualNodeName
(AkiraNetwork.VirtualStorageLibrary
VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary
VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary
VirtualNodeTypeFilter

Class VirtualItem

Represents a virtual item. This class serves as the base class for the `VirtualItem<T>`

([AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html](#)) type and is used solely for type checking of `VirtualItem` ([AkiraNetwork.VirtualStorageLibrary.VirtualItem.html](#)).

([AkiraNetwork.VirtualStorageLibrary](#))

Inheritance

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `object` (<https://learn.microsoft.com/dotnet/api/system.object>)

↳ `VirtualNode` ([AkiraNetwork.VirtualStorageLibrary.VirtualNode.html](#))

↳ `VirtualItem`

↳ `VirtualCycledDetector`

↳ `VirtualItem` ([AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html](#)) <`T`>

Implements

([AkiraNetwork.VirtualStorageLibrary](#))

`IVirtualDeepCloneable` ([AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html](#)) <`VirtualNode`

↳ `VirtualDirectoryAdapter<T>`

([AkiraNetwork.VirtualStorageLibrary.VirtualNode.html](#)) >

Inherited Members

1.html)

`VirtualException`

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualGroupCondition<T, TKey>`

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualNode` ([VirtualNode.html](#)) #[AkiraNetwork_VirtualStorageLibrary_VirtualNode_CreatedDate](#) 2.html)

`VirtualUpdatedDate`

([AkiraNetwork.VirtualStorageLibrary](#))

`DateVirtualItem`

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualNodeAdapter<T>`

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualNode` ([VirtualNode.html](#)) #[AkiraNetwork_VirtualStorageLibrary_VirtualNode_NodeType](#)

e) ([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualNode.VID`

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualUpdateVirtualStorage`

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualNode` ([VirtualNode.html](#)) #[AkiraNetwork_VirtualStorageLibrary_VirtualNode_Update_A](#)

kiraNetwork_VirtualStorageLibrary_VirtualNode_

`VirtualNodeReferenceVirtualStorage`

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualNode` ([VirtualNode.html](#)) #[AkiraNetwork_VirtualStorageLibrary_VirtualNode_IsReferen](#)

ciedInStorage

`VirtualNodeExtensions` ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object)))

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualNodeObject` ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object)))

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualHashCode` (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualGetType` (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualMemberClone` (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualReferenceEquals` (<https://learn.microsoft.com/dotnet/api/system.object.referenceeequals>)

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualToString` (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)

([AkiraNetwork.VirtualStorageLibrary](#))

Namespace: [AkiraNetwork](#) ([AkiraNetwork.html](#)).VirtualStorageLibrary ([AkiraNetwork.VirtualStorageLibrary.html](#))

VirtualNodeTypeFilter

Syntax

```
public abstract class VirtualItem : VirtualNode, IVirtualDeepCloneable<VirtualNode>
```



Constructors

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

VirtualItem(VirtualNodeName)

PatternMatch

Initializes a new instance of the VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) class.

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

```
protected VirtualItem(VirtualNodeName name)
```

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html

Type

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

2.html

VirtualItem(VirtualNodeName, DateTime)

(AkiraNetwork.VirtualStorageLibrary)

Initializes a new instance of the VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) class.

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

```
protected VirtualItem(VirtualNodeName name, DateTime createdDate)
```

1.html

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

Type

1.html

VirtualNode

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

DateTime (<https://learn.microsoft.com/dotnet/api/system.datetime>)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

VirtualItem(VirtualNodeName, DateTime, DateTime)

(AkiraNetwork.VirtualStorageLibrary)

Initializes a new instance of the VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) class.

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Name | Description |
|------------------|------|-----------------------|
| VirtualException | name | The name of the item. |

| Type | Name | Description |
|-----------------|-------------|--------------------------------|
| VirtualNode | name | The name of the item. |
| VirtualNodeName | createdDate | The creation date of the item. |

Implements

IVirtualDeepCloneable<T> (AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html)

Extension Methods

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilsilie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
NotifyNodeDelegate
VirtualNodeExtensions.ResolveNodeType(VirtualNode)
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode
Extensions_ResolveNodeType_AkiraNetwork_VirtualStorageLibrary_VirtualNode_)
(AkiraNetwork.VirtualStorageLibrar
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html#AkiraNetwork_VirtualStorageLibrary_VirtualCyc
eDetector)
VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html#AkiraNetwork_VirtualStorageLibrary_VirtualDir
ectory)
VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter.html#AkiraNetwork_VirtualStorageLibrary_Virt
ualDirectoryAdapter_1.html)
VirtualException
(AkiraNetwork.VirtualStorageLibrary.VirtualException.html#AkiraNetwork_VirtualStorageLibrary_Virt
ualException)
VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition.html#AkiraNetwork_VirtualStorageLibrary_Virt
ualGroupCondition_2.html)
VirtualID
(AkiraNetwork.VirtualStorageLibrary.VirtualID.html#AkiraNetwork_VirtualStorageLibrary_Virt
ualID)
VirtualItem
(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html#AkiraNetwork_VirtualStorageLibrary_Virt
ualItem)
VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter.html#AkiraNetwork_VirtualStorageLibrary_Virt
ualItemAdapter_1.html)
VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualItemT.html#AkiraNetwork_VirtualStorageLibrary_Virt
ualItemT_1.html)
VirtualNode
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_Virt
ualNode)
VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html#AkiraNetwork_VirtualStorageLibrary_Virt
ualNodeContext)
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html#AkiraNetwork_VirtualStorageLibrary_Virt
ualNodeExtensions)
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html#AkiraNetwork_VirtualStorageLibrary_V
irtualNodeListConditions)
VirtualnodeName
(AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html#AkiraNetwork_VirtualStorageLibrary_Virt
ualnodeName)
VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html#AkiraNetwork_VirtualStorageLibrary_V
irtualNodeNotFoundException)
VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html#AkiraNetwork_VirtualStorageLibrary_Virt
ualNodeType)
VirtualNodeTypeFilter

Class VirtualItemAdapter<T>

Inheritance

↳ [Object](#) (<https://learn.microsoft.com/dotnet/api/system.object>)
↳ [VirtualItemAdapter<T>](#)
 (AkiraNetwork.VirtualStorageLibrary)

Inherited Members

NotifyNodeDelegate

object.Equals(object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object)))
PatternMatch
(AkiraNetwork.VirtualStorageLibrary)
object.Equals(object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object)))
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary)
object.GetHashCode() (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)
object.GetType() (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)
object.MemberwiseClone()
(AkiraNetwork.VirtualStorageLibrary)
VirtualDirectoryAdapter<T>
object.ReferenceEquals(object1, object2) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)
object.ToString() (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)
1.html)

Namespace: AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)
VirtualException

Assembly: VirtualStorageLibrary.dll

Syntax: VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

public class VirtualItemAdapter<T>

 VirtualID

(AkiraNetwork.VirtualStorageLibrary)

Type Parameters:

VirtualItem

| Name | Description |
|-----------------------|--------------------------------------|
| VirtualItemAdapter<T> | (AkiraNetwork.VirtualStorageLibrary) |
| T | (AkiraNetwork.VirtualStorageLibrary) |

1.html)

VirtualItem<T>

Constructors

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualNode

VirtualItemAdapter(VirtualStorage<T>)

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

public VirtualItemAdapter(VirtualStorage<T> storage)
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|--|---------|-------------|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary) | | |
| VirtualNodeNameFilter (AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html)<T> (AkiraNetwork.VirtualStorageLibrary) | storage | |

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Properties

this[VirtualPath, bool]

Declaration

```
public VirtualItem<T> this[VirtualPath path, bool followLinks = true] { get; set; }
```

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

Parameters

(AkiraNetwork.VirtualStorageLibrary)

Type

PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

VirtualPath

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

bool

(https://learn.microsoft.com/dotnet/api/system.boolean)

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

Property Value

VirtualDirectoryAdapter<T>

Type

AkiraNetwork.VirtualStorageLibrary

1.html

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html)<T>

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

VirtualID

(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilsilie

s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Extension Methods

Class VirtualItem<T>

Represents a virtual item. A virtual item is an object used to manage item data within the virtual storage. It holds item data based on the user-defined type T, facilitating the management and manipulation of various types of content.

(AkiraNetwork.VirtualStorageLibrary)
Virtual items ensure that the actual data is distinctly present. If the user-defined type T implements the IVirtualNotifyNodeDelegate
DeepCloneable<T> (AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html) interface, a deep clone of
the item data is created, ensuring separate instances. If this interface is not implemented, shallow copies may
result in multiple virtual items referencing the same instance of the item data.
(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

Inheritance (AkiraNetwork.VirtualStorageLibrary)

↳ object (<https://learn.microsoft.com/dotnet/api/system.object>)
↳ VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)
↳ VirtualDirectoryAdapter<T>
↳ VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)
↳ VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary)
1.html)

Implements

VirtualException
IVirtualDeepCloneable (AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html) <VirtualNode
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)>
IDisposable (<https://learn.microsoft.com/dotnet/api/system.idisposable>)

Inherited Members

VirtualID
VirtualNode.Name
(AkiraNetwork.VirtualStorageLibrary)
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Name)
VirtualItem
VirtualNode.CreatedDate
(AkiraNetwork.VirtualStorageLibrary)
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_CreatedD
ate)
VirtualNode.UpdatedDate
VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Updated
Date)
VirtualNode.VID
VirtualNode
VirtualNode.IsReferencedInStorage
VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary)
object.Equals(object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object)))
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary)
object.Equals(object, object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object)))
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary)
object.GetHashCode() (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)
VirtualNodeName
VirtualNode
object.MemberwiseClone() (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)
VirtualNodeNotFoundException
object.ReferenceEquals(object, object) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)
(AkiraNetwork.VirtualStorageLibrary)

Namespace AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)

Assembly AkiraNetwork.VirtualStorageLibrary

VirtualNodeTypeFilter

Syntax

```
public class VirtualItem<T> : VirtualItem, IVirtualDeepCloneable<VirtualNode>, IDisposable
```

Type Parameters

| Name | Description |
|------|---|
| T | (AkiraNetwork.VirtualStorageLibrary) The user-defined type library can be freely specified by the user of this library. |

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary)

PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

Initializes a new instance of the VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) class. The node name is automatically generated by the GenerateNodeName(string)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeName_GenerateNodeName_System_String_) method.

VirtualException

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

public VirtualItem()

(AkiraNetwork.VirtualStorageLibrary)

2.html)

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem(VirtualNodeName)

VirtualItem

Initializes a new instance of the VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) class.

VirtualItemAdapter<T>

Declaration

(AkiraNetwork.VirtualStorageLibrary)

1.html)

public VirtualItem(VirtualNodeName name)

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Parameters

VirtualNode

Type

(AkiraNetwork.VirtualStorageLibrary)

Name

Description

| | | |
|--------------------|------|-----------------------|
| VirtualNodeContext | name | The name of the item. |
|--------------------|------|-----------------------|

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem(VirtualNodeName, T?)

(AkiraNetwork.VirtualStorageLibrary)

Initializes a new instance of the VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) class.

VirtualNodeName

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

public VirtualItem(VirtualNodeName name, T? itemData)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

Parameters

VirtualNodeTypeFilter

| Type | Name | Description |
|--|-----------------|---|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | <i>name</i> | The name of the item. |
| T ▼ | <i>itemData</i> | The data associated with this item. It is user-defined and can hold any type of data. |

VirtualItem(VirtualNodeName, T?, DateTime)

PatternMatch

Initializes a new instance of the VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) class.

VirtualCycleDetector

Declaration

```
public VirtualItem(VirtualNodeName name, T? itemData, DateTime createdDate)
    : base(name, itemData, createdDate)
```

Parameters

1.html)

| Type | Name | Description |
|---|-------------|-----------------------|
| VirtualException | | |
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | <i>name</i> | The name of the item. |
| VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | | |

T VirtualID

(AkiraNetwork.VirtualStorageLibrary.VirtualID.html)

VirtualItem

DateTime
(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)

(https://learn.microsoft.com/dotnet/api/system.datetime)
VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter.html)

1.html)

VirtualItem<T>

VirtualItem(VirtualNodeName, T?, DateTime, DateTime)

1.html)

Initializes a new instance of the VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) class.

VirtualNode

Declaration

```
public VirtualItem(VirtualNodeName name, T? itemData, DateTime createdDate, DateTime updatedDate)
    : base(name, itemData, createdDate, updatedDate)
```

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html)

VirtualNodeListConditions

Parameters

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html)

| Type | Name | Description |
|--|-------------|-----------------------|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | <i>name</i> | The name of the item. |
| VirtualNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNotFoundException.html) | | |

VirtualNodeNameType
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNameType.html)

VirtualNodeTypeFilter
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html)

Methods

DeepClone(bool)

Creates a deep clone of this item.

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.  
VirtualNodeDelegate VirtualNode DeepClone(bool recursive = false))
```

Parameters

| Type | Name | Description |
|----------------------|-----------|--|
| virtualCycleDetector | | |
| bool | recursive | If true, perform a deep clone recursively. |

VirtualDirectoryAdapter<T>

Returns
(AkiraNetwork.VirtualStorageLibrary.

| Type | Description |
|------------------|--|
| VirtualException | A deep clone of the current item as a VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html). |

Overrides

VirtualDeepClone(bool)
(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html#AkiraNetwork_VirtualStorageLibrary_VirtualItem_DeepClone_1.html)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.

Dispose()

VirtualItem<T>
Releases the resources used by the VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html).
(AkiraNetwork.VirtualStorageLibrary.

Declaration

```
VirtualNode  
public void Dispose()  
VirtualNodeContext  
(AkiraNetwork.VirtualStorageLibrary.  
VirtualNodeExtensions
```

Dispose(bool)

VirtualNodeListConditions
Releases the unmanaged resources used by the VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) and optionally releases the managed resources.

(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeNotFoundException

```
(AkiraNetwork.VirtualStorageLibrary.  
VirtualNodeTypeFilter  
protected virtual void Dispose(bool disposing))
```

(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeTypeFilter

Parameters

| Type | Name | Description |
|---|------------------|---|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | <i>disposing</i> | true to release both managed and unmanaged resources; false to release only unmanaged resources. |

▼

~VirtualItem()

NotifyNodeDelegate
Finalizer for the VirtualItem<T> ([AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html](#)) class.
([AkiraNetwork.VirtualStorageLibrary](#))

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
VirtualCycleDetector
protected VirtualItem()
(AkiraNetwork.VirtualStorageLibrary)
VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary)
VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary)
```

Tostring()

VirtualException
Declaration
([AkiraNetwork.VirtualStorageLibrary](#))

VirtualGroupCondition<T, TKey>
public override string ToString()
(AkiraNetwork.VirtualStorageLibrary)
2.html)

Returns

VirtualID
([AkiraNetwork.VirtualStorageLibrary](#))
Type
VirtualItem
([AkiraNetwork.VirtualStorageLibrary](#))

Description

A string that represents the current object.

Overrides

object.ToString() (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)
VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)
1.html)

Update(VirtualNode)

VirtualNode

Updates the specified node's data.

VirtualNodeContext
Declaration
([AkiraNetwork.VirtualStorageLibrary](#))

```
VirtualNodeExtensions
public override void Update(VirtualNode node)
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeListConditions
```

Parameters

VirtualNodeName
Type
([AkiraNetwork.VirtualStorageLibrary](#))

Name

Description

VirtualNodeNotFoundException
VirtualNode
([AkiraNetwork.VirtualStorageLibrary](#))

node

The node containing the data to update.

VirtualNodeType
Overridden
VirtualNodeTypeFilter
([AkiraNetwork.VirtualStorageLibrary](#))

VirtualNode.Update(VirtualNode)
 (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Update_AkiraNetwork_VirtualStorageLibrary_VirtualNode_)

Exceptions

| Type | Condition |
|---|---|
| ArgumentException https://learn.microsoft.com/dotnet/api/system.argumentexception NotifyNodeDelegate https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate | Thrown if the specified node is not of the same type as the current item. |

Operators

| |
|---|
| VirtualCycleDetector https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector |
| (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector) |

implicit operator VirtualItem<T>(VirtualNodeName)

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryNameAdapter<T>
 Implicitly converts the specified node name to a VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html)
<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryNameAdapter>)

Declaration

```
VirtualException  

(AkiraNetwork.VirtualStorageLibrary.VirtualException)
public static implicit operator VirtualItem<T>(VirtualNodeName name)  

  VirtualGroupCondition<T, TKey>
  (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition)
```

Parameters

| Type | Name | Description |
|--|------|-----------------------|
| VirtualID https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualID | | |
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualNodeName | name | The name of the node. |

Returns

| Type | Description |
|---|--|
| VirtVirtualItem<T> https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualItem | A VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) initialized with the specified node name. |

implicit operator VirtualItem<T>((VirtualNodeName nodeName, T? itemData))

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions
 Converts a tuple (nodeName, itemData) to a VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html)
<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions>)

Declaration

```
VirtualNotFoundException  

  (AkiraNetwork.VirtualStorageLibrary.VirtualNotFoundException)
VirtualItem<T>((VirtualNodeName nodeName, T? itemData) tuple  

  e(AkiraNetwork.VirtualStorageLibrary.VirtualNodeFilter)
  VirtualNodeType
  (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType)
  VirtualNodeTypeFilter)
```

| Type | Name | Description |
|--|-------|---|
| (VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) nodeName (https://learn.microsoft.com/dotnet/api/system.valuetuple-akiranetwork.virtualstoragelibrary.virtualnodename,-0-.nodename) , T itemData (https://learn.microsoft.com/dotnet/api/system.valuetuple-akiranetwork.virtualstoragelibrary.virtualnodename,-0-.itemdata)) (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate) | tuple | A tuple containing the node name and item data. |

Returns
PatternMatch

| Type | Description |
|--|---|
| AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory Item-1.html) <T> (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) | A VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) initialized with the specified values. |

implicit operator VirtualItem<T>(T?)

(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey>
Implicitly converts the specified data of type T to a VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html). The node name is automatically generated by the
GenerateNodeName(string
2.html)
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeName_GenerateNodeName_System_String_) method.
(AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html)

Declaration
(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>
public static implicit operator VirtualItem<T>(T? itemData)
(AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html)

Parameters

| Type | Name | Description |
|------|-----------------|--|
| T | VirtualNodeData | The data associated with this item. It is user-defined and can hold any type of data. (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) |

Returns

| Type | Description |
|--|---|
| VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext VirtualNodeExtensions <T>VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) <T>VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | A VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) initialized with the specified data. |

Implements
IVirtualNode (AkiraNetwork.VirtualStorageLibrary.IVirtualNode-1.html)
IDisposable (https://learn.microsoft.com/dotnet/api/system.idisposable)

Extension Methods

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualNodeExtensions.ResolveNodeType(VirtualNode)
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode
Extensions_ResolveNodeType_AkiraNetwork_VirtualStorageLibrary_VirtualNode_)
(AkiraNetwork.VirtualStorageLibrary.
NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrary.
PatternMatch
(AkiraNetwork.VirtualStorageLibrary.
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary.
VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary.
VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.
1.html)
VirtualException
(AkiraNetwork.VirtualStorageLibrary.
VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary.
2.html)
VirtualID
(AkiraNetwork.VirtualStorageLibrary.
VirtualItem
(AkiraNetwork.VirtualStorageLibrary.
VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.
1.html)
VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary.
1.html)
VirtualNode
(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary.
VirtualnodeName
(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeTypeFilter

Class VirtualNode

Represents an abstract class for nodes.

Filter by title

Inheritance

(AkiraNetwork.VirtualStorageLibrary)

↳ object (<https://learn.microsoft.com/dotnet/api/system.object>)

↳ NotifyNodeDelegate

↳ VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

↳ VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html)

↳ VirtualItem

(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)

↳ VirtualSymbolicLink

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)

↳ VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

↳ VirtualDirectory

IVirtualDeepCloneable ([AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html](#)) <VirtualNode>

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

↳ VirtualDirectoryAdapter<T>

Inherited Members

(AkiraNetwork.VirtualStorageLibrary)

object.Equals(object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object)))

VirtualException

object.Equals(object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object)))

VirtualGroupCondition<TKey>

object.GetHashCode() (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)

object.GetType() (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)

object.MemberwiseClone() (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)

object.ReferenceEquals(object, object) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)

object.ToString() (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)

(AkiraNetwork.VirtualStorageLibrary)

Namespace: AkiraNetwork ([AkiraNetwork.html](#)).VirtualStorageLibrary ([AkiraNetwork.VirtualStorageLibrary.html](#))

VirtualItemAdapter<T>

Assembly: VirtualStorageLibrary.dll

(AkiraNetwork.VirtualStorageLibrary)

Syntax

.html)

VirtualItem<T>

public class VirtualNode : IVirtualDeepCloneable<VirtualNode>

1.html)

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNode(VirtualNodeName)

(AkiraNetwork.VirtualStorageLibrary)

Initializes a new instance of the VirtualNode ([AkiraNetwork.VirtualStorageLibrary.VirtualNode.html](#)) class.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

protected NodeNotFoundException VirtualnodeName name)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Name | Description |
|---|------|-------------------|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | name | The name of node. |

VirtualNode(VirtualNodeName, DateTime)

Initializes a new instance of the VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) class.
 (AkiraNetwork.VirtualStorageLibrary)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
protected VirtualNode(VirtualNodeName name, DateTime createdDate)
(AkiraNetwork.VirtualStorageLibrary)
  VirtualCycleDetector
```

Parameters

| | | | |
|----------------------|---|-------------|---------------------------|
| Type | VirtualDirectoryName (AkiraNetwork.VirtualStorageLibrary) | Name | Description |
| VirtualDirectoryName | VirtualDirectoryNameAdapter<T> (AkiraNetwork.VirtualStorageLibrary) | name | The name of node. |
| DateTime | DateTime (https://learn.microsoft.com/dotnet/api/system.datetime) VirtualException (AkiraNetwork.VirtualStorageLibrary) | createdDate | The created date of node. |

| | |
|--------------------------------|--------------------------------------|
| VirtualGroupCondition<T, TKey> | (AkiraNetwork.VirtualStorageLibrary) |
|--------------------------------|--------------------------------------|

VirtualNode(VirtualNodeName, DateTime, DateTime)

VirtualID
 Initializes a new instance of the VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) class.
 (AkiraNetwork.VirtualStorageLibrary)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
VirtualItemAdapter<T>
protected VirtualNode(VirtualNodeName name, DateTime createdDate, DateTime updatedDate)
(AkiraNetwork.VirtualStorageLibrary)
  1.html)
```

Parameters

| | | | |
|-----------------------|---|-------------|---------------------------|
| Type | VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary) | Name | Description |
| VirtualDirectoryName | VirtualDirectoryNameAdapter<T> (AkiraNetwork.VirtualStorageLibrary) | name | The name of node. |
| VirtualNodeContext | DateTime (https://learn.microsoft.com/dotnet/api/system.datetime) (AkiraNetwork.VirtualStorageLibrary) | createdDate | The created date of node. |
| VirtualNodeExtensions | DateTime (https://learn.microsoft.com/dotnet/api/system.datetime) (AkiraNetwork.VirtualStorageLibrary) | updatedDate | The updated date of node. |

Properties

| | |
|------------------------------|--|
| CreatedDate | VirtualNodeName (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeName | VirtualNodeNameFilter (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeNotFoundException | VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeType | VirtualNodeType (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeTypeFilter | VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary) |

Gets the created date of the node. This date represents when the node was first created.

Declaration

```
public DateTime CreatedDate { get; }
```



Property Value

| Type | Description |
|--|-------------|
| (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate DateTime (https://learn.microsoft.com/dotnet/api/system.datetime) (AkiraNetwork.VirtualStorageLibrary.PatternMatch (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualNode VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions VirtualnodeName (https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType VirtualNodeTypeFilter | |

IsReferencedInStorage

(AkiraNetwork.VirtualStorageLibrary)

Gets [VirtualDirectory](#) indicating whether referenced in storage. If this property is true, the node is referenced from storage.

Other ([AkiraNetwork.VirtualStorageLibrary](#))

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

```
public bool IsReferencedInStorage { get; }
```

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

Property Value

| Type | Description |
|--|-------------|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary) | |

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

Name

(VirtualItemAdapter<T>)

(AkiraNetwork.VirtualStorageLibrary)

Gets the name of node.

1.html)

Declaration

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

```
1.html) public VirtualNodeName Name { get; }
```

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

Property Value

| Type | Description |
|--|-------------|
| VirtualNodeName (https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType VirtualNodeTypeFilter | |

NodeType

(AkiraNetwork.VirtualStorageLibrary)

Gets [VirtualNodeNotFoundException](#).

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

```
public abstract VirtualNodeType NodeType { get; }
```

Property Value

| Type | Description |
|---|-------------|
| VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html) (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary.PatternMatch (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector Gets the updated date of the node. This date represents the last time the node was modified. It is set to the current date and time at the moment of instantiation or cloning. (AkiraNetwork.VirtualStorageLibrary.Directory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter.UpdatedDate { get; } (AkiraNetwork.VirtualStorageLibrary.1.html) VirtualException | |

UpdatedDate

| Type | Description |
|--|-------------|
| AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey> DateTime (https://learn.microsoft.com/dotnet/api/system.datetime) (AkiraNetwork.VirtualStorageLibrary.VirtualItem 2.html) VirtualID | |

VID

| Declaration |
|--|
| (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T> 1.html) public VirtualID VID { get; } VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem 1.html) |

Property Value

| Type | Description |
|---|-------------|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode VirtualData (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.VirtualID.html) (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType Creates a deep clone of the VirtualStorageLibrary. However, the CreatedDate and UpdatedDate should not be cloned as they are set to VirtualNode date and time at the time of cloning. (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter VirtualNodeType | |

Methods

DeepClone(bool)

| | |
|--|--|
| Creates a deep clone of the VirtualStorageLibrary. However, the CreatedDate and UpdatedDate should not be cloned as they are set to VirtualNode date and time at the time of cloning. (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter | |
|--|--|

```
public abstract VirtualNode DeepClone(bool recursive = false)
```

Parameters

| Type | Name | Description |
|------|-----------|---|
| bool | recursive | When true, all child nodes are also cloned, creating a deep copy of the entire system. The default is false. The CreatedDate and UpdatedDate properties are not preserved. They are set to the current date and time at the moment of instantiation or cloning. |

PatternMatch

Returns [AkiraNetwork.VirtualStorageLibrary](#)

| Type | Description |
|--|--|
| VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary) | |
| VirtualNode (AkiraNetwork.VirtualStorageLibrary) | Cloned |
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary) | VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) |

1.html)

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

Update(VirtualNode)

Updates the [VirtualNode](#) ([AkiraNetwork.VirtualStorageLibrary.VirtualNode.html](#)).
2.html)

Declaration

(AkiraNetwork.VirtualStorageLibrary)

[VirtualItem](#)

public abstract void Update(VirtualNode node)

(AkiraNetwork.VirtualStorageLibrary)

[VirtualItemAdapter](#)<T>

Parameters

| Type | Name | Description |
|-----------------------------|------|-------------|
| VirtualItem | | |

1.html)

[VirtualNode](#)

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

node

Value to update

Implements

(AkiraNetwork.VirtualStorageLibrary)

[VirtualNodeContext](#)

[IVirtualDeepCloneable](#)<T> ([AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html](#))

(AkiraNetwork.VirtualStorageLibrary)

[VirtualNodeExtensions](#)

(AkiraNetwork.VirtualStorageLibrary)

[VirtualNodeListConditions](#)

[VirtualTextFormatter](#).GenerateSingleTableDebugText<T>(T)

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilities_VirtualNodeName)

s_VirtualTextFormatter.GenerateSingleTableDebugText_1__0_

(AkiraNetwork.VirtualStorageLibrary)

[VirtualNodeExtensions](#).ResolveNodeType(VirtualNode)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Extensions_ResolveNodeType_AkiraNetwork_VirtualStorageLibrary_VirtualNode_)

VIRTUALNODETYPE

(AkiraNetwork.VirtualStorageLibrary)

[VirtualNodeTypeFilter](#)



(AkiraNetwork.VirtualStorageLibrary)
NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrary)
PatternMatch
(AkiraNetwork.VirtualStorageLibrary)
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary)
VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary)
VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary)
1.html)
VirtualException
(AkiraNetwork.VirtualStorageLibrary)
VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary)
2.html)
VirtualID
(AkiraNetwork.VirtualStorageLibrary)
VirtualItem
(AkiraNetwork.VirtualStorageLibrary)
VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary)
1.html)
VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary)
1.html)
VirtualNode
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary)
VirtualnodeName
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeTypeFilter

Class VirtualNodeContext

A class that holds context information for a node. It is returned during or after path traversal, providing information about the node, path, parent directory, depth, index, resolved path, and symbolic link.

Inheritance

NotifyNodeDelegate

↳ object (<https://learn.microsoft.com/dotnet/api/system.object>)

↳ VirtualNodeContext

Inherited Members

VirtualCycleDetector

object.Equals(object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object)))

VirtualDirectory

object.Equals(object, object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object)))

VirtualDirectoryAdapter<T>

object.GetHashCode() (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)

object.GetType() (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)

object.MemberwiseClone() (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)

object.ReferenceEquals(object, object) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)

NameSpace AkiraNetwork(AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)

Assembly

(AkiraNetwork.VirtualStorageLibrary.dll)

Syntax

VirtualID

(AkiraNetwork.VirtualStorageLibrary)
public class VirtualNodeContext

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualItem<T>

VirtualNodeContext(VirtualNode?, VirtualPath, VirtualDirectory?, int, int, VirtualPath?, bool, VirtualSymbolicLink?)

VirtualNode
Initializes a new instance of the VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html)

(AkiraNetwork.VirtualStorageLibrary)
class VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)
public VirtualNodeContext(VirtualNode? node, VirtualPath traversalPath, VirtualDirectory? pa
rentNode = null, int depth = 0, int index = 0, VirtualPath? resolvedPath = null, bool resolv
ed = false, VirtualSymbolicLink? resolvedLink = null)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Name | Description |
|--|----------------------|---|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | <i>node</i> | The node being traversed. |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryPatternMatch VirtualDirectory.html) (AkiraNetwork.VirtualStorageLibrary.int VirtualCycleDetector (https://learn.microsoft.com/dotnet/api/system.int32?view=dotnet-plat-ext-2.2) VirtualDirectory int (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T> (https://learn.microsoft.com/dotnet/api/system.int32?view=dotnet-plat-ext-2.2) (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) | <i>traversalPath</i> | The path used for node traversal. |
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) | <i>parentNode</i> | The parent directory of the node. |
| int VirtualCycleDetector (https://learn.microsoft.com/dotnet/api/system.int32?view=dotnet-plat-ext-2.2) VirtualDirectory int (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T> (https://learn.microsoft.com/dotnet/api/system.int32?view=dotnet-plat-ext-2.2) (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) | <i>depth</i> | The depth of the node. The root is considered 0. |
| VirtualPathException (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition.html) | <i>index</i> | The index of the node. Corresponds to the enumeration order within the directory. |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean?view=dotnet-plat-ext-2.2) VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualID.html) | <i>resolved</i> | Indicates whether the symbolic link has been resolved. |
| VirtualItemLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter.html) | <i>resolvedLink</i> | The resolved symbolic link. |
| VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) | | |

Properties

Depth

Gets the depth of the node. Can only be set within the assembly. The root is considered 0.

VirtualNodeExtensions

Declaration
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

public int Depth { get; set; }
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

Property Value
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

Type
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType
int (https://learn.microsoft.com/dotnet/api/system.int32)
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Description

The depth of the node.

Index

Gets the index of the node and can only be set within the assembly. This is the index corresponding to the enumeration order within the directory.

Declaration

```
▼  
public int Index { get; }  
(AkiraNetwork.VirtualStorageLibrary  
NotifyNodeDelegate  
Property Value  
TypeatternMatch  
int (https://learn.microsoft.com/dotnet/api/system.int32)  
VirtualCycleDetector  
(AkiraNetwork.VirtualStorageLibrary  
VirtualDirectory  
Node  
AkiraNetwork.VirtualStorageLibrary  
VirtualDirectoryAdapter<T>  
Gets the node and can only be set within the assembly.  
(AkiraNetwork.VirtualStorageLibrary  
1.html)
```

Declaration

```
VirtualException  
(AkiraNetwork.VirtualStorageLibrary  
public VirtualNode? Node { get; }  
VirtualGroupCondition<T, TKey>  
(AkiraNetwork.VirtualStorageLibrary  
1.html)
```

Property Value

| Type | Description |
|--|-------------------|
| VirtualID (AkiraNetwork.VirtualStorageLibrary VirtualNode) (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | The current node. |

ParentDirectory

Gets the parent directory of the node and can only be set within the assembly.
(AkiraNetwork.VirtualStorageLibrary
1.html)

Declaration

```
VirtualNode  
p(AkiraNetwork.VirtualStorageLibrary  
VirtualDirectory { get; }  
VirtualNodeContext  
(AkiraNetwork.VirtualStorageLibrary  
1.html)
```

Property Value

| Type | Description |
|--|---|
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) | The instance of the parent directory. (AkiraNetwork.VirtualStorageLibrary 1.html) |

Resolved

Gets a value indicating whether the symbolic link has been resolved and can only be set within the assembly.
VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary
1.html)

Declaration

```
VirtualNodeTypeFilter  
(AkiraNetwork.VirtualStorageLibrary  
1.html)
```


Property Value

| Type | Description |
|---|-----------------------------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | The path used for node traversal. |



Methods

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary)

ToString()

PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

Returns a string representation of this instance's information.

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

Declaration

AkiraNetwork.VirtualStorageLibrary

VirtualDirectory

public string ToString()

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

Returns

1.html

| Type | Description |
|------------------|--------------------------------------|
| VirtualException | (AkiraNetwork.VirtualStorageLibrary) |

string (<https://learn.microsoft.com/dotnet/api/system.string>)

VirtualGroupCondition<T, TKey>

A string representing this instance's information.

(AkiraNetwork.VirtualStorageLibrary)

Overrides

2.html

object.ToString() (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

Extension Methods

(AkiraNetwork.VirtualStorageLibrary)

VirtualTextFormatterAdapter.GenerateSingleTableDebugText<T>(T)

(AkiraNetwork.VirtualStorageLibrary) ([AkiraNetwork_VirtualStorageLibrary_utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilsilities_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_](#))

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Class VirtualNodeExtensions

Provides extension methods for operations on virtual nodes in the Virtual Storage Library.

Filter by title

Inheritance

(AkiraNetwork.VirtualStorageLibrary)
↳ object (<https://learn.microsoft.com/dotnet/api/system.object>)
 ↳ NotifyNodeDelegate
 ↳ VirtualNodeExtensions
 ↳ AkiraNetwork.VirtualStorageLibrary

Inherited Members

(AkiraNetwork.VirtualStorageLibrary)
object.Equals(object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object)))
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary)
object.Equals(object, object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object)))
VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary)
object.GetHashCode() (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)
VirtualDirectoryAdapter<T>
object.GetType() (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)
(AkiraNetwork.VirtualStorageLibrary)
object.MemberwiseClone() (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)
1.html
object.ReferenceEquals(object, object) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)
VirtualException
object.ToString() (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)
(AkiraNetwork.VirtualStorageLibrary)

Name space: [AkiraNetwork](#) ([AkiraNetwork.html](#)).VirtualStorageLibrary ([AkiraNetwork.VirtualStorageLibrary.html](#))

Assembly: [AkiraNetwork.VirtualStorageLibrary.dll](#) ([AkiraNetwork.VirtualStorageLibrary.dll.html](#))

Syntax

VirtualID
(AkiraNetwork.VirtualStorageLibrary)
public static class VirtualNodeExtensions

VirtualItem
(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

Methods

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

ResolveNodeType(VirtualNode)

VirtualNode

Resolves the type of the specified virtual node. If the node is a symbolic link, returns the type of the target node.

(AkiraNetwork.VirtualStorageLibrary)

Otherwise, returns the type of the node itself.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

public static VirtualNodeType ResolveNodeType(this VirtualNode node)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

Parameters

VirtualNodeName
(AkiraNetwork.VirtualStorageLibrary)

Type
VirtualNodeNotFoundException

Name Description

Virt
(AkiraNetwork.VirtualStorageLibrary)

node

The virtual node to resolve the type for.

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Returns

| Type | Description |
|---|---|
| VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html) ▼ | The type of the node or, if the node is a symbolic link, the type of the target node. |

(AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrary.PatternMatch
(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualException
(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary.VirtualID
(AkiraNetwork.VirtualStorageLibrary.VirtualItem
(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualNode
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary.VirtualnodeName
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter

Struct VirtualNodeListConditions

Represents the conditions for creating a list of virtual nodes.

Filter by title

Inherited Members

(AkiraNetwork.VirtualStorageLibrary)
Value.Equals(object) (https://learn.microsoft.com/dotnet/api/system.valuetype.equals)
NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrary)
Value.GetHashCode() (https://learn.microsoft.com/dotnet/api/system.valuetype.gethashcode)
Value.ToString() (https://learn.microsoft.com/dotnet/api/system.valuetype.tostring)
object.Equals(object, object) (https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object))
VirtualCycledDetector
object.GetType() (https://learn.microsoft.com/dotnet/api/system.object.gettype)
object.ReferenceEquals(object, object) (https://learn.microsoft.com/dotnet/api/system.object.referenceequals)
(AkiraNetwork.VirtualStorageLibrary)

NameSpace AkiraNetwork.VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)

Assembly! VirtualStorageLibrary.dll

(AkiraNetwork.VirtualStorageLibrary)
Syntax
1.html)

VirtualException
public struct VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary)
VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary)

Constructors

VirtualNodeListConditions()
(AkiraNetwork.VirtualStorageLibrary)
VirtualID
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions()

(AkiraNetwork.VirtualStorageLibrary)
Initializes a new instance of the VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary.VIRTUALNODELISTCONDITIONS)

1.html)
Declaration
VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary)
public VirtualNodeListConditions()
1.html)

VirtualNode
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions(VirtualNodeTypeFilter,
(AkiraNetwork.VirtualStorageLibrary)
VirtualGroupCondition<VirtualNode, object>?,
VirtualNodeExtensions
List<VirtualSortCondition<VirtualNode>>?)
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions
Initializes a New instance of the VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary.VIRTUALNODELISTCONDITIONS.html) class with the specified conditions.
VirtualNodeName

Declaration
(AkiraNetwork.VirtualStorageLibrary)
VirtualNotFoundException
(AkiraNetwork.VirtualStorageLibrary)VIRTUALNODELISTCONDITIONS(VirtualNodeTypeFilter filter, VirtualGroupCondition<Virtual
Node, object> groupCondition, List<VirtualSortCondition<VirtualNode>>? sortConditions)
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeTypeFilter

Parameters

| Type | Name | Description |
|--|-----------------------|---|
| VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | <i>filter</i> | The filter criteria for the virtual nodes. |
| VirtualGroupCondition (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition-2.html) <VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html), object (https://learn.microsoft.com/dotnet/api/system.object)> not my node delegate | <i>groupCondition</i> | The grouping condition for the virtual nodes. |
| List (AkiraNetwork.VirtualStorageLibrary.PatternMatch (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1) (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector < VirtualSortCondition (AkiraNetwork.VirtualStorageLibrary.VirtualSortCondition-1.html) (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory <VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)>> VirtualDirectoryAdapter (T) | <i>sortConditions</i> | The sorting conditions for the virtual nodes. |

VirtualException Properties

VirtualGroupCondition<T, TKey>

(AkiraNetwork VirtualStorage) ijr

[\[er.html\]](#)

- 2.htm

VirtualId
Specifies the filter criteria, determining which types of virtual nodes to include.
(AkiraNetwork.VirtualStorageLibrary)

DeclarativeItem

(AkiraNetw)

Virtueller Adapter (T)

```
public virtual ModelTypeFilter Filter { readonly get; set; }
```

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Property Value
VirtualItem<T>

Type: AkiraNetwork.VirtualStorageLibrary

| Type | Description |
|--|--|
| 1.html) | |
| VirtualNodeTypeFilter | The filter specifying the types of virtual nodes. The default value is |
| VirtualNode | All |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html#AkiraNetwork.VirtualStorageLibrary_VirtualNodeTypeFilter_All) |
| VirtualNodeContext | . |
| (AkiraNetwork.VirtualStorageLibrary. | |
| VirtualNodeExtensions | . |
| (AkiraNetwork.VirtualStorageLibrary. | |
| VirtualNodeListConditions | . |
| (AkiraNetwork.VirtualStorageLibrary. | |
| VirtualNodeName | |
| (AkiraNetwork.VirtualStorageLibrary. | |
| GroupCondition | Specifying the group VirtualStorageLibrary determining how to group the virtual nodes. |
| VirtualNodeNotFoundException | |
| Declaration | (AkiraNetwork.VirtualStorageLibrary. |
| VirtualNodeType | |
| public VirtualGroupCondition<VirtualNode, object>? GroupCondition { readonly get; set; } | (AkiraNetwork.VirtualStorageLibrary. |
| VirtualNodeTypeFilter | |

Property Value

| Type | Description |
|--|--|
| VirtualGroupCondition (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition-2.html) < VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) , object (https://learn.microsoft.com/dotnet/api/system.object)> (AkiraNetwork.VirtualStorageLibrary. PatternMatch (AkiraNetwork.VirtualStorageLibrary. FileDetector | The condition for grouping virtual nodes. The default value is <code>null</code> . You can specify <code>null</code> if no grouping condition is needed. |

SortConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory)
Specifies the sorting conditions, determining the order in which the virtual nodes are arranged.

Declaration

```
VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualSortCondition<VirtualNode>>? SortConditions { readonly get; set; }  

1.html)
```

VirtualException

Property Value

| Type | Description |
|--|---|
| VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.List 2.html) (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1) list-1(AkiraNetwork.VirtualStorageLibrary.< VirtualItem VirtualSortCondition (AkiraNetwork.VirtualStorageLibrary.VirtualSortCondition-1.html) VirtualItemAdapter<T> < (AkiraNetwork.VirtualStorageLibrary.VirtualNode) VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) >> (AkiraNetwork.VirtualStorageLibrary. 1.html) | A list of conditions for sorting the virtual nodes. The default value is <code>null</code> . You can specify <code>null</code> if no sorting condition is needed. |

Extension Methods

VirtualNodeContext
VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilities_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter)

Class VirtualnodeName

Inheritance

↳ [Object](#) (<https://learn.microsoft.com/dotnet/api/system.object>)

↳ [VirtualnodeName](#)
([AkiraNetwork.VirtualStorageLibrary](#))

Implements

[INotifyNodeDelegate](#)
([AkiraNetwork.VirtualStorageLibrary](#))

[IEquatable<VirtualnodeName>](#)
([AkiraNetwork.VirtualStorageLibrary](#))

[IComparable<VirtualnodeName>](#)
([AkiraNetwork.VirtualStorageLibrary](#))

[INodeDetector](#)
([AkiraNetwork.VirtualStorageLibrary](#))

[IComparable](#) (<https://learn.microsoft.com/dotnet/api/system.icomparable>)

Inherited Members

[VirtualDirectory](#)
([AkiraNetwork.VirtualStorageLibrary](#))

[object.Equals\(object, object\)](#) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object)))

[object.GetType\(\)](#) (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)

[object.MemberwiseClone\(\)](#) (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)

[object.ReferenceEquals\(object, object\)](#) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)

[Namespace](#) ([AkiraNetwork](#)) ([AkiraNetwork.Key.html](#)).VirtualStorageLibrary ([AkiraNetwork.VirtualStorageLibrary.html](#))

[Assembly](#) ([AkiraNetwork](#)) ([VirtualStorageLibrary.dll](#))

Syntax

[VirtualID](#)
([AkiraNetwork.VirtualStorageLibrary](#))

```
public class VirtualnodeName : IEquatable<VirtualnodeName>, IComparable<VirtualnodeName>, IComparable
```

[VirtualItem](#)
([AkiraNetwork.VirtualStorageLibrary](#))

[VirtualItemAdapter<T>](#)
([AkiraNetwork.VirtualStorageLibrary](#))

Constructors

[VirtualItem<T>](#)

([AkiraNetwork.VirtualStorageLibrary](#))

VirtualnodeName(string)

[VirtualNode](#)

[Declaration](#)
([AkiraNetwork.VirtualStorageLibrary](#))

[VirtualNodeContext](#)

[public VirtualnodeName\(string name\)](#)

[VirtualNodeExtensions](#)

([AkiraNetwork.VirtualStorageLibrary](#))

[Parameters](#)

[VirtualNodeListConditions](#)

[Type](#)
([AkiraNetwork.VirtualStorageLibrary](#))

| Type | Name | Description |
|---|-------------|-------------|
| VirtualnodeName string (https://learn.microsoft.com/dotnet/api/system.string) | <i>name</i> | |

([AkiraNetwork.VirtualStorageLibrary](#))

[VirtualNodeNotFoundException](#)

([AkiraNetwork.VirtualStorageLibrary](#))

[VirtualNodeType](#)

([AkiraNetwork.VirtualStorageLibrary](#))

[VirtualNodeTypeFilter](#)

Properties

IsRoot

Declaration

```
public bool IsRoot { get; }
```



Property Value

| Type | Description |
|---|-------------|
| NotifyNodeDelegate<bool> (https://learn.microsoft.com/dotnet/api/system.boolean) | |

Name

VirtualDirectoryName
(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
Directory  
(AkiraNetwork.VirtualStorageLibrary)  
virtual string Name { get; }  
(AkiraNetwork.VirtualStorageLibrary)
```

Property Value

| Type | Description |
|--|-------------|
| VirtualGroupCondition<T, TKey> string (https://learn.microsoft.com/dotnet/api/system.string) | |

2.html

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

Methods

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

CompareTo(VirtualNodeName?)

1.html

Compares the current instance with another object of the same type and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
VirtualNode  
(AkiraNetwork.VirtualStorageLibrary)  
public int CompareTo(VirtualNodeName? other)
```

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

Parameters

VirtualNodeExtensions

| Type | Name | Description |
|--|-------|--|
| VirtualNodeListConditions VirtualNodeName (AkiraNetwork.VirtualStorageLibrary) | other | An object to compare with this instance. |

(AkiraNetwork.VirtualStorageLibrary)

Return

VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Description | | | | | | | | |
|--|---|-------|---------|----------------|--|------|---|-------------------|---|
| int https://learn.microsoft.com/dotnet/api/system.int32 | A value that indicates the relative order of the objects being compared. The return value has these meanings: <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>Less than zero</td> <td>This instance precedes <code>other</code> in the sort order.</td> </tr> <tr> <td>Zero</td> <td>This instance occurs in the same position in the sort order as <code>other</code>.</td> </tr> <tr> <td>Greater than zero</td> <td>This instance follows <code>other</code> in the sort order.</td> </tr> </tbody> </table> | Value | Meaning | Less than zero | This instance precedes <code>other</code> in the sort order. | Zero | This instance occurs in the same position in the sort order as <code>other</code> . | Greater than zero | This instance follows <code>other</code> in the sort order. |
| Value | Meaning | | | | | | | | |
| Less than zero | This instance precedes <code>other</code> in the sort order. | | | | | | | | |
| Zero | This instance occurs in the same position in the sort order as <code>other</code> . | | | | | | | | |
| Greater than zero | This instance follows <code>other</code> in the sort order. | | | | | | | | |

CompareTo(object?)

[\(AkiraNetwork.VirtualStorageLibrary\)](#)

Compares the current instance with another object of the same type and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object.
[1.html](#)

Declaration:

```
(AkiraNetwork.VirtualStorageLibrary)
public int CompareTo(object? obj)
    {
        if (obj == null)
            throw new ArgumentNullException("obj");
        if (obj.GetType() != this.GetType())
            throw new ArgumentException("obj must be of the same type as the current instance.");
        VirtualItemAdapter<T> adapter = obj as VirtualItemAdapter<T>;
        if (adapter != null)
            return this.CompareTo(adapter);
        VirtualGroupComparison<TKey> comparison = obj as VirtualGroupComparison<TKey>;
        if (comparison != null)
            return this.CompareTo(comparison);
        return this.CompareTo((VirtualItem) obj);
    }
```

Parameters:

VirtualID

| Type | Name | Description |
|--|------------------|--|
| AkiraNetwork.VirtualStorageLibrary | <code>obj</code> | An object to compare with this instance. |

VirtualItemAdapter<T>

[\(AkiraNetwork.VirtualStorageLibrary\)](#)

| Type | Description | | | | | | | | |
|--------------------------------------|---|-------|---------|----------------|--|------|---|-------------------|---|
| VirtualItem<T> | A value that indicates the relative order of the objects being compared. The return value has these meanings: <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>Less than zero</td> <td>This instance precedes <code>obj</code> in the sort order.</td> </tr> <tr> <td>Zero</td> <td>This instance occurs in the same position in the sort order as <code>obj</code>.</td> </tr> <tr> <td>Greater than zero</td> <td>This instance follows <code>obj</code> in the sort order.</td> </tr> </tbody> </table> | Value | Meaning | Less than zero | This instance precedes <code>obj</code> in the sort order. | Zero | This instance occurs in the same position in the sort order as <code>obj</code> . | Greater than zero | This instance follows <code>obj</code> in the sort order. |
| Value | Meaning | | | | | | | | |
| Less than zero | This instance precedes <code>obj</code> in the sort order. | | | | | | | | |
| Zero | This instance occurs in the same position in the sort order as <code>obj</code> . | | | | | | | | |
| Greater than zero | This instance follows <code>obj</code> in the sort order. | | | | | | | | |

Exceptions:

[VirtualNodeNotFoundException](#)

| Type | Condition |
|---------------------------------------|---|
| VirtualStorageLibrary | <code>obj</code> is not the same type as this instance. |

Equals(VirtualNodeName?)

Indicates whether the current object is equal to another object of the same type.

Declaration

```
public bool Equals(VirtualNodeName? other)
```

Parameters (AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

Type (AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

Returns

VirtualDirectory

Type (AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

bool true (<https://learn.microsoft.com/dotnet/csharp/language-reference/builtin-types/bool>) if the

(AkiraNetwork.VirtualStorageLibrary)

(<https://learn.microsoft.com/dotnet/api/system.boolean>) current object is equal to the other parameter; otherwise,

[crosoft.com/dotnet/csharp/language-reference/builtin-types/bool](https://learn.microsoft.com/dotnet/csharp/language-reference/builtin-types/bool).

VirtualException

[net/api/system.boolean](https://learn.microsoft.com/dotnet/api/system.boolean)

(AkiraNetwork.VirtualStorageLibrary)

boolean

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

2.html)

Equals(object?)

(AkiraNetwork.VirtualStorageLibrary)

Determines whether the specified object is equal to the current object.

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

public override bool Equals(object? obj)

1.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Type VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

(<https://learn.microsoft.com/dotnet/api/system.object>)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

Type VirtualNodeList<Conditions>

(AkiraNetwork.VirtualStorageLibrary)

bool true (<https://learn.microsoft.com/dotnet/csharp/language-reference/builtin-types/bool>) if the

(<https://learn.microsoft.com/dotnet/api/system.boolean>) specified object is equal to the current object; otherwise,

[crosoft.com/dotnet/csharp/language-reference/builtin-types/bool](https://learn.microsoft.com/dotnet/csharp/language-reference/builtin-types/bool).

VirtualNodeNotFoundException

[net/api/system.exception](https://learn.microsoft.com/dotnet/api/system.exception)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

Overrides VirtualNodeTypeFilter

object.Equals(object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object)))

GenerateNodeName(string)

Declaration

```
public string GenerateNodeName(string prefix)
```

 NotifyNodeDelegate

Parameters
 PatternMatch

Type AkiraNetwork.VirtualStorageLibrary.

 VirtualCycleDetector
 string (<https://learn.microsoft.com/dotnet/api/system.string>)
 (AkiraNetwork.VirtualStorageLibrary.

 VirtualDirectory
Returns
 (AkiraNetwork.VirtualStorageLibrary.

Type VirtualDirectoryAdapter<T>

 (AkiraNetwork.VirtualStorageLibrary.
 VirtualNodeName ([AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html](https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html))
 1.html)

 VirtualException

 (AkiraNetwork.VirtualStorageLibrary.

GetHashCode()
 VirtualGroupCondition<T, TKey>
 (AkiraNetwork.VirtualStorageLibrary.

Serves as the default hash function.

 VirtualID

Declaration
 (AkiraNetwork.VirtualStorageLibrary.

 VirtualItem

```
public override int GetHashCode()
```

 VirtualItemAdapter<T>

Returns
 (AkiraNetwork.VirtualStorageLibrary.

 1.html)

Type VirtualItem<T>

 int (<https://learn.microsoft.com/dotnet/api/system.int32>)
 1.html)

Overrides
 (AkiraNetwork.VirtualStorageLibrary.

object.GetHashCode() (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)
 VirtualNodeContext

 (AkiraNetwork.VirtualStorageLibrary.

IsValidNodeName(VirtualNodeName)
 (AkiraNetwork.VirtualStorageLibrary.

Declaration
 VirtualNodeListConditions

 (AkiraNetwork.VirtualStorageLibrary.

```
public bool IsValidNodeName(VirtualNodeName nodeName)
```

 (AkiraNetwork.VirtualStorageLibrary.

 VirtualNodeNotFoundException

Parameters
 (AkiraNetwork.VirtualStorageLibrary.

Type VirtualNodeType

 (AkiraNetwork.VirtualStorageLibrary.
 VirtualNodeName ([AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html](https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html))
 VirtualNodeTypeFilter

| Type | Name | Description |
|-------------------------------------|--------|-------------|
| AkiraNetwork.VirtualStorageLibrary. | prefix | |

| Type | Description |
|----------------------------|---|
| VirtualDirectoryAdapter<T> | (AkiraNetwork.VirtualStorageLibrary. VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) 1.html) |

| Type | Description |
|----------------|-------------------------------------|
| VirtualItem<T> | A hash code for the current object. |

| Type | Name | Description |
|-------------------------------------|----------|-------------|
| AkiraNetwork.VirtualStorageLibrary. | nodeName | |

Returns

| Type | Description |
|--|-------------|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | |



ResetCounter()

(AkiraNetwork.VirtualStorageLibrary)
Declaration
NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrary)
public static void ResetCounter()
PatternMatch

(AkiraNetwork.VirtualStorageLibrary)
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary)
VirtualDirectory

ToString()
(AkiraNetwork.VirtualStorageLibrary)
Returns a string that represents the current object.
VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)
1.html)
pVirtualException string ToString()
(AkiraNetwork.VirtualStorageLibrary)

Returns
(AkiraNetwork.VirtualStorageLibrary)
Type.html)

| Type | Description |
|---|--|
| VirtualID string (https://learn.microsoft.com/dotnet/api/system.string) (AkiraNetwork.VirtualStorageLibrary) | A string that represents the current object. |

VirtualItem

Overrides
(AkiraNetwork.VirtualStorageLibrary)
object.ToString() (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)
(AkiraNetwork.VirtualStorageLibrary)

Operators

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

operator ==(VirtualNodeName?, VirtualNodeName?)

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

p(AkiraNetwork.VirtualStorageLibrary) VirtualNodeName? left, VirtualNodeName? right)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

Parameters

VirtualNodeListConditions

| Type | Name | Description |
|--|-------|-------------|
| VirtualNodeName VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) (AkiraNetwork.VirtualStorageLibrary) | left | |
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) (AkiraNetwork.VirtualStorageLibrary) | right | |

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Description |
|--|-------------|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | |

implicit operator string(VirtualnodeName)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    NotifyNodeDelegate
public static implicit operator string(VirtualnodeName nodeName)
(AkiraNetwork.VirtualStorageLibrary)
    PatternMatch
```

Parameters

| Type | Name | Description |
|--|------|-------------|
| VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary) | | |

Returns

```
VirtualDirectoryName  
(AkiraNetwork.VirtualStorageLibrary)
VirtualDirectoryName.html
```

| Type | Description |
|--|-------------|
| string (https://learn.microsoft.com/dotnet/api/system.string) (AkiraNetwork.VirtualStorageLibrary) | |

```
VirtualGroupCondition<T, TKey>  
(AkiraNetwork.VirtualStorageLibrary)
```

implicit operator VirtualnodeName(string)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    VirtualItem
public static implicit operator VirtualnodeName(string name)
(AkiraNetwork.VirtualStorageLibrary)
    VirtualItemAdapter<T>
```

Parameters

| Type | Name | Description |
|--|------|-------------|
| VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary) | | |

Returns

```
VirtualNode  
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeContext
```

| Type | Description |
|--|-------------|
| VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary) | |

```
VirtualnodeName  
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeExtensions
```

```
(AkiraNetwork.VirtualStorageLibrary)
```

```
VirtualNodeListConditions
```

operator !=(VirtualnodeName?, VirtualnodeName?)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    VirtualNodeNotFoundException
public static bool operator !=(VirtualnodeName? left, VirtualnodeName? right)
(AkiraNetwork.VirtualStorageLibrary)
    VirtualNodeType
```

Parameters

```
VirtualNodeTypeFilter  
(AkiraNetwork.VirtualStorageLibrary)
```

| Type | Name | Description |
|---|--------------|-------------|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | <i>left</i> | |
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | <i>right</i> | |

Returns

| Type | Description |
|---|-------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | |

Implements

| |
|--|
| VirtualCycleDetector |
| IEquatable<T> (https://learn.microsoft.com/dotnet/api/system.iequatable-1) |
| IComparable<T> (https://learn.microsoft.com/dotnet/api/system.icomparable-1) |
| IComparable (https://learn.microsoft.com/dotnet/api/system.icomparable) |
| VirtualDirectoryAdapter<T> |

Extension Methods

| |
|---|
| GenerateSingleTableDebugText<T>(T) |
| (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilsilites_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_) |

| |
|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| 2.html |

VirtualID

| |
|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| VirtualItem |

| |
|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| VirtualItemAdapter<T> |

| |
|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| VirtualItem<T> |

| |
|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| VirtualNode |

| |
|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| VirtualNodeContext |

| |
|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| VirtualNodeExtensions |

| |
|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| VirtualNodeListConditions |

| |
|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| VirtualnodeName |

| |
|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| VirtualNodeNotFoundException |

| |
|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| VirtualNodeType |

| |
|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| VirtualNodeTypeFilter |

Class VirtualNodeNotFoundException

The exception that is thrown when a node is not found in the virtual storage.

Filter by title

Inheritance

(AkiraNetwork.VirtualStorageLibrary)

↳ object (<https://learn.microsoft.com/dotnet/api/system.object>)

↳ NotifyNodeDelegate

↳ Exception (<https://learn.microsoft.com/dotnet/api/system.exception>)

↳ VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualException.html)

↳ PatternMatch

↳ VirtualNodeNotFoundException

↳ (AkiraNetwork.VirtualStorageLibrary)

Implements

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

ISerializable (<https://learn.microsoft.com/dotnet/api/system.runtime.serialization.iserializable>)

VirtualDirectory

Inherited Members

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter (<https://learn.microsoft.com/dotnet/api/system.exception.getbaseexception>)

ExceptionGettype (<https://learn.microsoft.com/dotnet/api/system.exception.gettype>)

ExceptionToString () (<https://learn.microsoft.com/dotnet/api/system.exception.tostring>)

ExceptionData (<https://learn.microsoft.com/dotnet/api/system.exception.data>)

(AkiraNetwork.VirtualStorageLibrary)

ExceptionHelpLink (<https://learn.microsoft.com/dotnet/api/system.exception.helplink>)

ExceptionResultCondition (https://learn.microsoft.com/dotnet/api/system.exception_hresult)

(AkiraNetwork.VirtualStorageLibrary)

ExceptionInnerException (<https://learn.microsoft.com/dotnet/api/system.exception.innerexception>)

ExceptionMessage (<https://learn.microsoft.com/dotnet/api/system.exception.message>)

ExceptionSource (<https://learn.microsoft.com/dotnet/api/system.exception.source>)

(AkiraNetwork.VirtualStorageLibrary)

ExceptionStackTrace (<https://learn.microsoft.com/dotnet/api/system.exception.stacktrace>)

ExceptionTargetSite (<https://learn.microsoft.com/dotnet/api/system.exception.targetsite>)

ExceptionObjectState (<https://learn.microsoft.com/dotnet/api/system.exception.serializeobjectstate>)

object.Equals (object, object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-object)))

(AkiraNetwork.VirtualStorageLibrary)

object.GetHashCode () (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)

object.MemberwiseClone () (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)

object.ReferenceEquals (object, object) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)

(AkiraNetwork.VirtualStorageLibrary)

Namespace: AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)

VirtualNodeContext

Assembly: VirtualStorageLibrary.dll

(AkiraNetwork.VirtualStorageLibrary)

Syntax

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

↳ **VirtualNodeNotFoundException** : VirtualException, ISerializable

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

VirtualNodeNotFoundException()

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Constructors

Initializes a new instance of the VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html) class.

Declaration

```
public VirtualNodeNotFoundException()  
    ▾
```

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException(string)

(AkiraNetwork.VirtualStorageLibrary)

Initializes a new instance of the VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html) class with a specified error message.

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

```
public VirtualNodeNotFoundException(string message)  
(AkiraNetwork.VirtualStorageLibrary)
```

VirtualDirectoryAdapter<T>

Parameters

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Type VirtualException

Name

Description

string (AkiraNetwork.VirtualStorageLibrary) (https://learn.microsoft.com/dotnet/api/system.string)

message

The error message.

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

2.html)

VirtualNodeNotFoundException(string, Exception)

(AkiraNetwork.VirtualStorageLibrary)

Initializes a new instance of the VirtualNodeNotFoundException

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html) class with a specified error message

and a reference to the inner exception that is the cause of this exception.

VirtualItemAdapter<T>

Declaration

(AkiraNetwork.VirtualStorageLibrary)

1.html)

```
public VirtualNodeNotFoundException(string message, Exception innerException)  
(AkiraNetwork.VirtualStorageLibrary)
```

1.html)

Parameters

VirtualNode

Type (AkiraNetwork.VirtualStorageLibrary)

Name

Description

VirtualNodeContext

message

The error message.

string (AkiraNetwork.VirtualStorageLibrary)

(https://learn.microsoft.com/dotnet/api/system.string)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

(https://learn.microsoft.com/dotnet/api/system.exception)

(AkiraNetwork.VirtualStorageLibrary)

innerException

The exception that is the cause of this exception.

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

ISerializable (VirtualNodeType) (https://learn.microsoft.com/dotnet/api/system.runtime.serialization.iserializable)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Implements

Extension Methods

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

▼

(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
PatternMatch
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualException
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualID
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualItem
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualNode
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualnodeName
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualNodeTypeFilter

Enum VirtualNodeType

Specifies the types of virtual nodes.

Filter by title

Namespace: AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)
(AkiraNetwork.VirtualStorageLibrary)

Assembly: VirtualStorageLibrary.dll
NotifyNodeDelegate

Syntax
(AkiraNetwork.VirtualStorageLibrary)

PatternMatch

public enum VirtualNodeType

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

Fields

| Name | Description |
|--------------------------------|---|
| 1.html | |
| Directory | Represents a directory node. |
| VirtualException | Represents an item node with a user-defined type T. |
| Item | Represents an item node with a user-defined type T. |
| VirtualGroupCondition<T, TKey> | |
| None | Represents an unspecified or unknown node type. |
| 2.html | |
| SymbolicLink | Represents a symbolic link node. |
| VirtualID | |

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

VirtualTextFormatter.GenerateSingleTableDebugText<VirtualNodeType>(VirtualNodeType)

(AkiraNetwork.VirtualStorageLibrary)
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Extension Methods

Enum VirtualNodeTypeFilter

Specifies the filters for virtual node types.

Filter by title

Namespace: AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)

(AkiraNetwork.VirtualStorageLibrary)

Assembly: VirtualStorageLibrary.dll

NotifyNodeDelegate

Syntax

```
(AkiraNetwork.VirtualStorageLibrary)
  PatternMatch
public enum VirtualNodeTypeFilter
  VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary)
  VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary)
  VirtualDirectoryAdapter<T>
```

Fields

| Name | Description |
|--------------|---|
| 1.html | |
| All | VirtualException Filter for all node types. |
| Directory | (AkiraNetwork.VirtualStorageLibrary) Filter for directory nodes. |
| Item | (AkiraNetwork.VirtualStorageLibrary) Item nodes with a user-defined type T. |
| 2.html | |
| None | VirtualID No specific node type is specified. |
| SymbolicLink | (AkiraNetwork.VirtualStorageLibrary) Filter for symbolic link nodes. |
| VirtualItem | |

Extension Methods

1.html

VirtualTextFormatter.GenerateSingleTableDebugText<VirtualNodeTypeFilter>(VirtualNodeTypeFilter)

(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils

s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

1.html

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Class VirtualPath

Inheritance

↳ [Object](#) (<https://learn.microsoft.com/dotnet/api/system.object>)

↳ [VirtualPath](#)
([AkiraNetwork.VirtualStorageLibrary](#))

Implements

[IEquatable<VirtualPath>](#) (<https://learn.microsoft.com/dotnet/api/system.iequatable-1>) <[VirtualPath](#)

([AkiraNetwork.VirtualStorageLibrary.VirtualPath.html](#))>

[IComparable<VirtualPath>](#) (<https://learn.microsoft.com/dotnet/api/system.icomparable-1>) <[VirtualPath](#)

([AkiraNetwork.VirtualStorageLibrary.VirtualPath.html](#))>

[IComparable](#) (<https://learn.microsoft.com/dotnet/api/system.icomparable>)

Inherited Members

([AkiraNetwork.VirtualStorageLibrary](#))

[object.Equals\(object, object\)](#) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object)))

([AkiraNetwork.VirtualStorageLibrary](#))

[object.GetType\(\)](#) (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)

[object.MemberwiseClone\(\)](#) (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)

[object.ReferenceEquals\(object, object\)](#) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)

[Namespace](#) ([AkiraNetwork](#)) ([AkiraNetwork.Key.html](#)).VirtualStorageLibrary ([AkiraNetwork.VirtualStorageLibrary.html](#))

[Assembly](#) ([AkiraNetwork.VirtualStorageLibrary.dll](#))

Syntax

[VirtualID](#)

([AkiraNetwork.VirtualStorageLibrary](#))
public class VirtualPath : IEquatable<VirtualPath>, IComparable<VirtualPath>, IComparable<VirtualItem>

([AkiraNetwork.VirtualStorageLibrary](#))

[VirtualItemAdapter<T>](#)

([AkiraNetwork.VirtualStorageLibrary](#))

1.html)

[VirtualItem<T>](#)

Constructors

[VirtualPath\(IEnumerable<VirtualNodeName>\)](#)

1.html)

Declaration

[VirtualNode](#)

([AkiraNetwork.VirtualStorageLibrary](#))

public VirtualPath(IEnumerable<VirtualNodeName> parts)

([AkiraNetwork.VirtualStorageLibrary](#))

Parameters

([AkiraNetwork.VirtualStorageLibrary](#))

Type [VirtualNodeListConditions](#)

([AkiraNetwork.VirtualStorageLibrary](#))

IEnumerable<VirtualNodeName>

(<https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1>)

<[VirtualNodeName](#) ([AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html](#))>

[VirtualNodeNotFoundException](#)

([AkiraNetwork.VirtualStorageLibrary](#))

[VirtualNodeType](#)

[VirtualPath\(string\)](#)

([AkiraNetwork.VirtualStorageLibrary](#))

[VirtualNodeTypeFilter](#)

| Type | Name | Description |
|---|--------------|-------------|
| VirtualNodeListConditions | <i>parts</i> | |

Declaration

```
public VirtualPath(string path)
```

Parameters

| Type | Name | Description |
|--|------|-------------|
| string (https://learn.microsoft.com/dotnet/api/system.string) | path | |

Properties

BaseDepth

Declaration: `int BaseDepth { get; }`

(AkiraNetwork.VirtualStorageLibrary)

Property Value: `VirtualGroupCondition<T, TKey>`

| Type | Description |
|------------------------------------|-------------|
| AkiraNetwork.VirtualStorageLibrary | |

int (<https://learn.microsoft.com/dotnet/api/system.int32>)

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

Declaration: `VirtualItemAdapter<T>`

1.html

Property Value: `VirtualItem`

| Type | Description |
|------------------------------------|-------------|
| AkiraNetwork.VirtualStorageLibrary | |

int (<https://learn.microsoft.com/dotnet/api/system.int32>)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

int (<https://learn.microsoft.com/dotnet/api/system.int32>)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

DirectoryPath

Declaration: `VirtualPath DirectoryPath { get; }`

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Description |
|---|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |

Dot

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    NotifyNodeDelegate
public static string Dot { get; }
```

PatternMatch

Property Value

| Type | Description |
|--|-------------|
| string (https://docs.microsoft.com/en-us/dotnet/api/system.string) | |

DotDot

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    VirtualGroupCondition<T, TKey>
public static string DotDot { get; }
```

2.html

Property Value

| Type | Description |
|--|-------------|
| string (https://docs.microsoft.com/en-us/dotnet/api/system.string) | |

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)
 1.html

FixedPath

VirtualItem<T>

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    1.html
    VirtualPath
public VirtualPath FixedPath { get; }
```

VirtualNodeContext

Property Value

| Type | Description |
|---|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    VirtualNodeType
    VirtualNodeTypeFilter
```

```
public bool IsAbsolute { get; }
```

Property Value

| Type | Description |
|--|-------------|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | |

(AkiraNetwork.VirtualStorageLibrary)
NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrary)
IsDot
PatternMatch

(AkiraNetwork.VirtualStorageLibrary)
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary)
public bool IsDot { get; }
VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary)

Property Value

| Type | Description |
|--|-------------|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | |

(AkiraNetwork.VirtualStorageLibrary)
VirtualException
(AkiraNetwork.VirtualStorageLibrary)
VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary)
IsDotDot
2.html)

(AkiraNetwork.VirtualStorageLibrary)
VirtualItem
public bool IsDotDot { get; }
(AkiraNetwork.VirtualStorageLibrary)
VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)
Property Value
1.html)

| Type | Description |
|----------------|-------------|
| VirtualItem<T> | |

bool (<https://learn.microsoft.com/dotnet/api/system.boolean>)

(AkiraNetwork.VirtualStorageLibrary)
VirtualNode
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary)

Property Value
VirtualNodeName

| Type | Description |
|------------------------------|-------------|
| VirtualNodeNotFoundException | |

bool (<https://learn.microsoft.com/dotnet/api/system.boolean>)

(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeTypeFilter

IsEndsWithSlash

Declaration

```
public bool IsEndsWithSlash { get; }
```



Property Value

| Type | Description |
|---|-------------|
| NotifyNodeDelegate<VirtualStorageLibrary> bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary) | |

PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

IsRoot

(AkiraNetwork.VirtualStorageLibrary)

Declaration

Directory
(AkiraNetwork.VirtualStorageLibrary)

~~VirtualDirectoryNameAdapter~~ { get; }

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Property Value

VirtualException

| Type | Description |
|--|-------------|
| VirtualGroupCondition<T, TKey> bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary) | |

2.html)

VirtualID

NodeName

VirtualItem

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

~~VirtualDirectoryNameAdapter~~ { get; }

1.html)

Property Value

(AkiraNetwork.VirtualStorageLibrary)

Type

.html)

| | |
|--------------------------------------|---|
| VirtualNode | (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) |
| (AkiraNetwork.VirtualStorageLibrary) | |

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

PartsList

NodeExtensions

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

public List<VirtualNodeName> PartsList { get; }

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Description |
|---|-------------|
| List (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1)<VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)> | |

Path

| Declaration |
|---|
| NotifyNodeDelegate |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) |
| PatternMatch |

| Property Value |
|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) |

| Type | Description |
|------------------|-------------|
| VirtualDirectory | |

Root

| Declaration |
|--|
| VirtualGroupCondition<T, TKey> |
| (AkiraNetwork.VirtualStorageLibrary.Root.html) |
| VirtualID |

| Property Value |
|--|
| (AkiraNetwork.VirtualStorageLibrary.Root.html) |

| Type | Description |
|-------------|-------------|
| VirtualItem | |

| Declaration |
|---|
| VirtualNode |
| (AkiraNetwork.VirtualStorageLibrary.Separator.html) |

| Property Value |
|---|
| (AkiraNetwork.VirtualStorageLibrary.Separator.html) |

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

| Declaration |
|---|
| VirtualNodeName |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html) |
| VirtualNodeNotFoundException |

| Method |
|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) |

AddEndSlash()

Declaration

```
public VirtualPath AddEndSlash()
```



Returns

| Type | Description |
|---|-------------|
| NotifyNodeDelegate<VirtualPath> (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |

PatternMatch

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualCodeDetector

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

AddStartSlash()

Declaration

```
Directory (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)
VirtualDirectoryAdapter<VirtualPath> AddStartSlash()
(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)
1.html)
```

Returns

| Type | Description |
|--|-------------|
| AkiraNetwork.VirtualStorageLibrary.VirtualPath.html | |
| VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualID

ArePathsSubdirectories(VirtualPath, VirtualPath)

Declaration

```
VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)
VirtualItemAdapter<T>
public static bool ArePathsSubdirectories(VirtualPath path1, VirtualPath path2)
1.html)
```

Parameters

| Type | Name | Description |
|---|-------|-------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path1 | |

| | | |
|---|-------|--|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path2 | |
|---|-------|--|

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

Returns

| Type | Description |
|---|-------------|
| VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |

| | |
|--|--|
| bool (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |
|--|--|

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualNodeNotFoundException

Combine(params VirtualPath[])

Declaration

```
VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)
VirtualNodeTypeFilter
```

```
public VirtualPath Combine(params VirtualPath[] paths)
```

Parameters

| Type | | Name | Description |
|---|--|--------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)[] | | <i>paths</i> | |

Put-NotifyNodeDelegate

| Returns | VirtualPath | Description |
|--------------|---|-------------|
| PatternMatch | (AkiraNetwork.VirtualStorageLibrary) | |
| VirtualPath | (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |

VirtualCycleDetector

Combine(params string[])
AKIRANETWORK.VIRTUALSTORAGELIBRARY

Combine(params string[1])
TAKANETWORK VirtualString

VirtualDirectoryAdapter<T>

Declaration

(AkiraNetwork.VirtualStorageLibrary)

```
1.html)  
public static string Combine(params string[] paths)  
VirtualException
```

Parametric

| Type | Name | Description |
|--|-------|-------------|
| string[] (https://learn.microsoft.com/dotnet/api/system.string)[] | paths | |

(AkiraNetw
Visited on

| Returns | (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>) string | (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>) |
|---------|---|--|
| Type | Description | Description |
| | | |

string (<https://learn.microsoft.com/dotnet/html>)

CombineFromIndex(VirtualPath, int)

Microbiology

Declaration (AkiraNetwork)

VirtualNodeContext

```
public VirtualPath CombineFromIndex(VirtualPath path, int index)  
(AkiraNetwork.VirtualStorageLibrar
```

VirtualNodeExtensions

| Parameter | Type | Name | Description |
|--|--|-------|-------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions) | (AkiraNetwork.VirtualStorageLibrary.VirtualPath) | path | |
| (AkiraNetwork.VirtualStorageLibrary.int) | (AkiraNetwork.VirtualStorageLibrary.int32) | index | |
| | | | |

VirtualNodeUnit

Returns VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeTypeFilter

| Type | Description |
|---|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |

CompareTo(VirtualPath?)

Compares the current instance with another object of the same type and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object.

(AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate)

Declaration:

```
PatternMatch
public int CompareTo(VirtualPath? other)
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary)
```

Parameters:

| Type | Name | Description |
|--|-------|--|
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T>) | other | An object to compare with this instance. |

VirtualException

Return Value:

| Type | Description | | | | | | | | |
|---|---|-------|---------|----------------|--|------|---|-------------------|---|
| VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary) | A value that indicates the relative order of the objects being compared. | | | | | | | | |
| int (2.html) (http://srftable.microsoft.com/dotnet/) | The return value has these meanings: | | | | | | | | |
| api/AkiraNetwork/VirtualStorageLibrary/VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>) | <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>Less than zero</td> <td>This instance precedes <code>other</code> in the sort order.</td> </tr> <tr> <td>Zero</td> <td>This instance occurs in the same position in the sort order as <code>other</code>.</td> </tr> <tr> <td>Greater than zero</td> <td>This instance follows <code>other</code> in the sort order.</td> </tr> </tbody> </table> | Value | Meaning | Less than zero | This instance precedes <code>other</code> in the sort order. | Zero | This instance occurs in the same position in the sort order as <code>other</code> . | Greater than zero | This instance follows <code>other</code> in the sort order. |
| Value | Meaning | | | | | | | | |
| Less than zero | This instance precedes <code>other</code> in the sort order. | | | | | | | | |
| Zero | This instance occurs in the same position in the sort order as <code>other</code> . | | | | | | | | |
| Greater than zero | This instance follows <code>other</code> in the sort order. | | | | | | | | |
| VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext) | | | | | | | | | |

CompareTo(object?)

(AkiraNetwork.VirtualStorageLibrary)

Compares the current instance with another object of the same type and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object.

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions)

Declaration:

```
VirtualNodeName
public int CompareTo(object? obj)
VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary)
```

Parameters:

```
VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeTypeFilter
```

| Type | Name | Description |
|--|------------|--|
| object (https://learn.microsoft.com/dotnet/api/system.object) | <i>obj</i> | An object to compare with this instance. |

Returns

| Type | Description |
|---|--|
| int (AkiraNetwork.VirtualStorageLibrary (https://learn.microsoft.com/dotnet/ NotifyNodeDelegate api/system.int32) (AkiraNetwork.VirtualStorageLibrary PatternMatch (AkiraNetwork.VirtualStorageLibrary VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary VirtualDirectory (AkiraNetwork.VirtualStorageLibrary VirtualDirectoryAdapter<T> | A value that indicates the relative order of the objects being compared. The return value has these meanings: |
| Value | Meaning |
| Less than zero | This instance precedes <code>obj</code> in the sort order. |
| Zero | This instance occurs in the same position in the sort order as <code>obj</code> . |
| Greater than zero | This instance follows <code>obj</code> in the sort order. |

Exceptions

VirtualException

| Type | Condition |
|--|---|
| <code>AkiraNetwork.VirtualStorageLibrary</code> <code>VirtualGroupCondition<T, TKey></code> <code>(AkiraNetwork.VirtualStorageLibrary)</code> (https://learn.microsoft.com/dotnet/api/system.argumentexception) 2.html | <code>obj</code> is not the same type as this instance. |

Equals(VirtualPath?)

(AkiraNetwork VirtualStorage) library

IsEqual (AKRA.NETWORK.VIRTUALStorageLibrary) Indicates whether the current object is equal to another object of the same type.

八九〇

declaration

`VirtualItem<T>`

```
public bool Equals(VirtualPath? other)  
(AkiraNetwork.VirtualStorage library)
```

1.html

Parameter Nodes

| Type | (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext) | Name | Description |
|-------------|---|-------|--|
| VirtualPath | (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | other | An object to compare with this object. |

Bsp: VirtualNode1.istConditions

| Type | Description |
|-----------------------|--|
| VirtualNodeName | (AkiraNetwork.VirtualStorageLibrary) |
| bool | (AkiraNetwork.VirtualStorageLibrary) microsoft.com/dotnet/csharp/language-reference/builtin-types/bool) if the VirtualNodeNotEqualException parameter is equal to the other parameter; otherwise, Microsoft.NETCore.UnauthorizedAccessException. |
| cross | (AkiraNetwork.VirtualStorageLibrary) microsoft.com/dotnet/csharp/language-reference/builtin-types/bool). |
| VirtualNodeType | (AkiraNetwork.VirtualStorageLibrary) |
| bool | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeTypeFilter | (AkiraNetwork.VirtualStorageLibrary) |

Equals(object?)

Determines whether the specified object is equal to the current object.

Declaration

```
public override bool Equals(object? obj)
```

Parameters

NotifyNodeDelegate

| Type | Description |
|--------|---|
| object | <code>obj</code> The object to compare with the current object. |

Returns

VirtualDirectory

| Type | Description |
|----------------------------|--|
| VirtualDirectoryAdapter<T> | <code>true</code> (https://learn.microsoft.com/dotnet/csharp/language-reference/builtin-types/bool) if the specified object is equal to the current object; otherwise, <code>false</code> (https://learn.microsoft.com/dotnet/csharp/language-reference/builtin-types/bool). |

Overrides

[Object.Equals](#)

| | |
|---|--------------------------------------|
| object.Equals(object) (https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object)) | (AkiraNetwork.VirtualStorageLibrary) |
|---|--------------------------------------|

VirtualItem

GetHashCode()

Serves as the default hash function.

[Object.GetHashCode](#)

Declaration

```
public override int GetHashCode()
```

Returns

VirtualNodeContext

| Type | Description |
|-----------------------|-------------------------------------|
| VirtualNodeExtensions | A hash code for the current object. |

VirtualNodeListConditions

Overrides

object.GetHashCode() (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

GetnodeName()

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

```
public VirtualNodeName GetNodeName()
```

Returns

| Type | Description |
|---|---|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary.PatternMatch)) |

GetParentPath()

Declaration

```
public VirtualPath GetParentPath()
{
    VirtualDirectory
}
```

Returns

```
VirtualDirectoryAdapter<T>
{
    VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)
    VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualException.html)
    VirtualGroupCondition<T, TKey>
}
```

GetPartsLinkedList()

Declaration

```
public VirtualItem<T> GetPartsLinkedList<VirtualNodeName>()
{
    VirtualItemAdapter<T>
}
```

Returns

```
VirtualItem<T>
{
    LinkedList<(https://learn.microsoft.com/dotnet/api/system.collections.generic.linkedlist-1)<VirtualNodeName> (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)
}
```

GetPartsList()

Declaration

```
public VirtualNodeList<VirtualNodeName> GetPartsList()
{
    VirtualNodeList<VirtualNodeName>
}
```

Returns

```
VirtualNodeList<VirtualNodeName>
{
    List<(https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1)<VirtualNodeType>> (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)
    VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)
}
```

VirtualNodeTypeFilter

GetRelativePath(VirtualPath)

Declaration

```
public VirtualPath GetRelativePath(VirtualPath basePath)
```



Parameters

| Type | Name | Description |
|---|----------|-------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath) NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary.PatternMatch) | basePath | |

Return (AkiraNetwork.VirtualStorageLibrary.VirtualPath)

| Type | Description |
|--|-------------|
| (VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualPath) VirtualPath(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)) | |

IsSubdirectory(VirtualPath)
1.html)

Declaration

```
virtual void IsSubdirectory(TKey virtualGroupCondition, TKey virtualPath parentPath)
    (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>  
2.html)
```

Parameters

| Type | Name | Description |
|--|------------|-------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualItem VirtualPath(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary.VirtualPath)) | parentPath | |

VirtualItemAdapter<T>

Return (AkiraNetwork.VirtualStorageLibrary.VirtualItem)

| Type | Description |
|---|-------------|
| VirtualItem<T> bool(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext 1.html) | |

VirtualNode
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext
VirtualNode)

Declaration

```
VirtualNodeExtensions
public VirtualPath NormalizePath()
    VirtualNodeListConditions
    (AkiraNetwork.VirtualStorageLibrary.VirtualPath)
```

Returns (VirtualnodeName)

| Type | Description |
|--|-------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName VirtualNodeNotFoundException VirtualPath(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary.VirtualPath)) | |

VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter
VirtualNodeType)

NormalizePath(string)

Declaration

```
public static string NormalizePath(string path)
```



Parameters

| Type | Name | Description |
|--|-------------|-------------|
| (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate<T> (https://learn.microsoft.com/dotnet/api/system.string)) | <i>path</i> | |

PatternMatch

Return (AkiraNetwork.VirtualStorageLibrary.

| Type | Description |
|---|-------------|
| (VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary. string (https://learn.microsoft.com/dotnet/api/system.string)) | |

(AkiraNetwork.VirtualStorageLibrary.

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.
1.html)

StartsWith(VirtualPath)

Declaration

(VirtualException
(AkiraNetwork.VirtualStorageLibrary.

VirtualGroupCondition<T>Key1Path path)

(AkiraNetwork.VirtualStorageLibrary.

2.html)

Parameters

| Type | Name | Description |
|--|-------------|-------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualItem VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary. | <i>path</i> | |

VirtualItemAdapter<T>

Return (AkiraNetwork.VirtualStorageLibrary.

| Type | Description |
|--|-------------|
| (VirtualItem<T> bool (https://learn.microsoft.com/dotnet/api/system.boolean)) | |

1.html)

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeContext

Returns a string that represents the current object.

VirtualNodeExtensions

Declaration (AkiraNetwork.VirtualStorageLibrary.

VirtualNodeListConditions

public NetworkVirtualStorageLibrary()

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.

Returns VirtualNodeNotFoundException

| Type | Description |
|---|--|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType string (https://learn.microsoft.com/dotnet/api/system.string)) | A string that represents the current object. |

VirtualNodeTypeFilter

Overrides

object.ToString() (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)

TrimEndSlash()

Declaration

```
p(AkiraNetwork.VirtualStorageLibrary)
```

 NotifyNodeDelegate

 (AkiraNetwork.VirtualStorageLibrary)

Returns

 PatternMatch

| Type | Description |
|------------------------------------|-------------|
| AkiraNetwork.VirtualStorageLibrary | |

 VirtualCycleDetector

 VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

 (AkiraNetwork.VirtualStorageLibrary)

 VirtualDirectory

 (AkiraNetwork.VirtualStorageLibrary)

 VirtualDirectoryAdapter<T>

 (AkiraNetwork.VirtualStorageLibrary)

 1.html)

operator +(VirtualPath, VirtualNodeName)

 VirtualException

 (AkiraNetwork.VirtualStorageLibrary)

 VirtualGroupCondition<T, TKey>

 (AkiraNetwork.VirtualStorageLibrary)

 public static VirtualPath operator +(VirtualPath path, VirtualNodeName nodeName)

 2.html)

 VirtualID

Parameters

 VirtualPath (AkiraNetwork.VirtualStorageLibrary)

 VirtualItem

 (AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|------------------------------------|-------------|
| AkiraNetwork.VirtualStorageLibrary | |

 Name

 Description

 VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

 path

 VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

 nodeName

 1.html)

Returns

 VirtualItem<T>

 (AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|----------------|-------------|
| VirtualItem<T> | |

 VirtualNode

 VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

 (AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|----------------|-------------|
| VirtualItem<T> | |

 VirtualNode

 VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

 (AkiraNetwork.VirtualStorageLibrary)

 VirtualNodeContext

 (AkiraNetwork.VirtualStorageLibrary)

 VirtualNodeFilter

 (AkiraNetwork.VirtualStorageLibrary)

operator +(VirtualPath, VirtualPath)

 (AkiraNetwork.VirtualStorageLibrary)

Declaration

 VirtualNodeListConditions

 (AkiraNetwork.VirtualStorageLibrary)

 VirtualNodeName VirtualPath operator +(VirtualPath path1, VirtualPath path2)

 (AkiraNetwork.VirtualStorageLibrary)

 VirtualNodeNotFoundException

 (AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|-----------------|-------------|
| VirtualNodeName | |

 VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

 VirtualNodeTypeFilter

| Name | Description |
|-----------------|-------------|
| VirtualNodeName | |

 path1

| Type | Name | Description |
|---|-------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path2 | |

Returns

| Type | Description |
|--|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

operator +(VirtualPath, char)

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

Declaration

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

```
public static VirtualPath operator +(VirtualPath path, char chr)
```

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

Parameters

Type

.html)

VirtualException

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

char (<https://learn.microsoft.com/dotnet/api/system.char>)

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

Returns

.html)

VirtualID

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

operator +(VirtualPath, string)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

.html)

```
public static VirtualPath operator +(VirtualPath path, string str)
```

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

Parameters

Type

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualNodeExtensions

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualNodeListConditions

string (<https://learn.microsoft.com/dotnet/api/system.string>)

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

Returns

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

Type

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualNodeTypeFilter

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

operator +(char, VirtualPath)

Declaration

```
public static VirtualPath operator +(char chr, VirtualPath path)
```



Parameters

| Type | Name | Description |
|--|-------------|-------------|
| (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate<char> char (https://learn.microsoft.com/dotnet/api/system.char) (AkiraNetwork.VirtualStorageLibrary) | chr | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary) | path | |
| VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary) | | |
| Type | Description | |
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary) | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary) | | |
| 1.html) | | |

operator +(string, VirtualPath)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary)
public static VirtualPath operator +(string str, VirtualPath path)
```

VirtualID

Parameters

| Type | Name | Description |
|---|------|-------------|
| (AkiraNetwork.VirtualStorageLibrary) | | |
| VirtualItemAdapter<T> string (https://learn.microsoft.com/dotnet/api/system.string) (AkiraNetwork.VirtualStorageLibrary) | str | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | |
| VirtualItem<T> | | |

Returns

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Type

| Type | Description |
|-------------|-------------|
| VirtualNode | |

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

operator ==(VirtualPath?, VirtualPath?)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeName
public static bool operator ==(VirtualPath? left, VirtualPath? right)
```

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

Parameters

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Name | Description |
|---|-------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | left | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | right | |

Returns

| Type | Description |
|---|-------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |

implicit operator string(VirtualPath?)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory)
public static implicit operator string(VirtualPath? virtualPath)
    VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory)
```

Parameters

| Type | Name | Description |
|---|------|-------------|
| VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualPathCondition, Key) | | |

Returns

| Type | Description |
|---|-------------|
| VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |

implicit operator VirtualPath(string)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.VirtualItem)
1.html)
public static implicit operator VirtualPath(string path)
    VirtualNode
(AkiraNetwork.VirtualStorageLibrary.VirtualItem)
```

Parameters

| Type | Name | Description |
|--|------|-------------|
| VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary.VirtualPath) | | |

Returns

| Type | Description |
|--|-------------|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualPath) | |

operator !=(VirtualPath?, VirtualPath?)

Class VirtualSortCondition<T>

Represents the conditions for sorting data, holding the property to sort by and the order (ascending or descending).

Inheritance

NotifyNodeDelegate

↳ object (<https://learn.microsoft.com/dotnet/api/system.object>)
↳ (AkiraNetwork.VirtualStorageLibrary)

↳ VirtualSortCondition<T>

Inherited Members

VirtualCycleDetector

object.Equals(object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object)))
↳ (AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

object.Equals(object, object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object)))
↳ (AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

object.GetHashCode() (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)
↳ (AkiraNetwork.VirtualStorageLibrary)

object.GetType() (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)
↳ 1.html

object.MemberwiseClone() (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)

object.ReferenceEquals(object, object) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)
↳ (AkiraNetwork.VirtualStorageLibrary)

object.ToString() (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)
↳ VirtualGroupCondition<T, TKey>

Namespace

(AkiraNetwork.VirtualStorageLibrary)

Assembly

(AkiraNetwork.VirtualStorageLibrary.dll)

Syntax

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

public class VirtualSortCondition<T>

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

Type

(AkiraNetwork.VirtualStorageLibrary)

Table

| Name | Description |
|------|---|
| T | (AkiraNetwork.VirtualStorageLibrary) The type of the entity to be sorted. |

1.html

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeFilter

(AkiraNetwork.VirtualStorageLibrary)

VirtualSortCondition(Expression<Func<T, object>>, bool)

(AkiraNetwork.VirtualStorageLibrary)

Initializes a new instance of the VirtualSortCondition<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualSortCondition-1.html) class with the specified sorting property and order.

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

public VirtualSortCondition(Expression<Func<T, object>> sortBy, bool ascending = true)
↳ VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

Parameters

VirtualNodeTypeFilter

| Type | Name | Description |
|--|-----------|--|
| Expression (https://learn.microsoft.com/dotnet/api/system.linq.expressions.expression-1) <Func (https://learn.microsoft.com/dotnet/api/system.func-2)<T, object (https://learn.microsoft.com/dotnet/api/system.object)>> | sortBy | The property to sort by. |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary) NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary) PatternMatch (AkiraNetwork.VirtualStorageLibrary) VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary) VirtualDirectory (AkiraNetwork.VirtualStorageLibrary) VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary) | ascending | A value indicating whether the sorting order is ascending. |

Properties

Ascending

Gets or sets a value indicating whether the sorting order is ascending. True if the order is ascending; otherwise, false.

[VirtualException](#)

Declaration

```
VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary)
public bool Ascending { get; set; }
```

[VirtualID](#)

PropertyValue

| Type | Description |
|---|--|
| VirtualItem (AkiraNetwork.VirtualStorageLibrary) | A boolean value indicating whether the sorting order is ascending. |

SortBy

Gets or sets the property used for sorting.

[VirtualNodeContext](#)

Declaration

```
VirtualNode
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary)
public VirtualNode<T>> SortBy { get; set; }
```

[VirtualNodeListConditions](#)

PropertyValue

[VirtualNodeName](#)

| Type | Description |
|--|---|
| VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary) <Func (https://learn.microsoft.com/dotnet/api/system.func-2)<T, object (https://learn.microsoft.com/dotnet/api/system.object)>> | An expression that specifies the property to use for sorting. |

Extension Methods

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

▼

(AkiraNetwork.VirtualStorageLibrary.
NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrary.
PatternMatch
(AkiraNetwork.VirtualStorageLibrary.
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary.
VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary.
VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.
1.html)
VirtualException
(AkiraNetwork.VirtualStorageLibrary.
VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary.
2.html)
VirtualID
(AkiraNetwork.VirtualStorageLibrary.
VirtualItem
(AkiraNetwork.VirtualStorageLibrary.
VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.
1.html)
VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary.
1.html)
VirtualNode
(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary.
VirtualnodeName
(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeTypeFilter

Class VirtualStorageExtensions

Provides extension methods for various operations on collections in the Virtual Storage Library.

Filter by title

Inheritance

(AkiraNetwork.VirtualStorageLibrary)
↳ object (https://learn.microsoft.com/dotnet/api/system.object)
 ↳ NotifyNodeDelegate
 ↳ VirtualStorageExtensions
 ↳ AkiraNetwork.VirtualStorageLibrary

Inherited Members

(AkiraNetwork.VirtualStorageLibrary)
object.Equals(object) (https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object))
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary)
object.Equals(object, object) (https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object))
VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary)
object.GetHashCode() (https://learn.microsoft.com/dotnet/api/system.object.gethashcode)
VirtualDirectoryAdapter<T>
object.GetType() (https://learn.microsoft.com/dotnet/api/system.object.gettype)
(AkiraNetwork.VirtualStorageLibrary)
object.MemberwiseClone() (https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone)
VirtualException
object.ReferenceEquals(object, object) (https://learn.microsoft.com/dotnet/api/system.object.referenceequals)
object.ToString() (https://learn.microsoft.com/dotnet/api/system.object.tostring)
(AkiraNetwork.VirtualStorageLibrary)

Name space AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)

Assembly AkiraNetwork.VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.dll)

Syntax

VirtualID
(AkiraNetwork.VirtualStorageLibrary)
public static class VirtualStorageExtensions
 VirtualItem

(AkiraNetwork.VirtualStorageLibrary)
VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary)
VirtualItem<T>

Methods

VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary)
VirtualNode

ApplySortConditions<T>(IEnumerable<T>, List<VirtualSortCondition<T>>?)

Applies the specified sort conditions to the elements of the source sequence.
(AkiraNetwork.VirtualStorageLibrary)

Declaration

(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeExtensions
public static IEnumerable<T> ApplySortConditions<T>(this IEnumerable<T> source, List<VirtualSortCondition<T>>? sortConditions = null)
 VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary)

Parameters

VirtualNodeName
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeTypeFilter

| Type | Name | Description |
|---|--------|-----------------------------------|
| IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) <T> List (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1) NotifyNodeDelegate < (AkiraNetwork.VirtualStorageLibrary.VirtualSortCondition VirtualSortCondition PatternMatch (AkiraNetwork.VirtualStorageLibrary.VirtualSortCondition- 1.html) VirtualCycleDetector <T>(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory VirtualDirectory Return(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T> IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) <T>VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey> 2.html) | source | The sequence of elements to sort. |

| Type | Description |
|--|--|
| (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T> IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) <T>VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey> 2.html) | An IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) containing the sorted elements. |

| Name | Description |
|--|--|
| T (AkiraNetwork.VirtualStorageLibrary.VirtualItem) | The type of elements in the source sequence. |

| Name | Description |
|-----------------------|--|
| VirtualItemAdapter<T> | (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T> 1.html) |

GroupAndSort<T>(IEnumerable<T>, VirtualGroupCondition<T, object>?, List<VirtualSortCondition<T>>?)

Groups and sorts the elements of the source sequence based on the specified conditions.
VirtualNode

Declaration
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext
public static VirtualNodeGroupAndSort<T>(this IEnumerable<T> source, VirtualGroupCondition<T, object>? groupCondition = null, List<VirtualSortCondition<T>>? sortConditions = null)
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName
VirtualNodeName
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName
IEnumerable
VirtualNodeNotFoundException
(<https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1>)
VirtualNodeType
<T>(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType
VirtualNodeTypeFilter

| Type | Name | Description |
|---|--------|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName IEnumerable VirtualNodeNotFoundException (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) VirtualNodeType <T>(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType VirtualNodeTypeFilter | source | The sequence of elements to group and sort. |

| Type | Name | Description |
|--|-----------------------|--|
| VirtualGroupCondition (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition-2.html) <T, object (https://learn.microsoft.com/dotnet/api/system.object) > (AkiraNetwork.VirtualStorageLibrary. NotifyNodeDelegate List (AkiraNetwork.VirtualStorageLibrary. (https://learn.microsoft.com/dotnet/api/system.collection.generic.list-1) AkiraNetwork.VirtualStorageLibrary. < VirtualCycleDetector VirtualSortCondition (AkiraNetwork.VirtualStorageLibrary. (AkiraNetwork.VirtualStorageLibrary.VirtualSortCondition-1.html) 1.html) (AkiraNetwork.VirtualStorageLibrary. <T>VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary | <i>groupCondition</i> | The condition that defines how to group the elements. If null, no grouping is applied. |

Returns [IList<T>](#)

| Type | Description |
|---|---|
| VirtualException (AkiraNetwork.VirtualStorageLibrary. IEnumerable<T>VirtualGroupCondition<T, TKey> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) <T>VirtualID (AkiraNetwork.VirtualStorageLibrary. VirtualItem (AkiraNetwork.VirtualStorageLibrary | An IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) containing the grouped and sorted elements. |

Type Parameters

| Name | Description |
|----------|--|
| <i>T</i> | The type of elements in the source sequence. VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary. 1.html) VirtualItem (AkiraNetwork.VirtualStorageLibrary. VirtualNode (AkiraNetwork.VirtualStorageLibrary. VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary. VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary. VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary. VirtualnodeName (AkiraNetwork.VirtualStorageLibrary. VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary. VirtualNodeType (AkiraNetwork.VirtualStorageLibrary. VirtualNodeTypeFilter |


```
public char[] InvalidNodeNameCharacters { get; set; }
```

Property Value

| Type | Description |
|---|--|
| char (https://learn.microsoft.com/dotnet/api/system.char)[] | (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate)(AkiraNetwork.VirtualStorageLibrary.PatternMatch) |

NodeListConditions

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector)(AkiraNetwork.VirtualStorageLibrary.VirtualException) public VirtualNodeListConditions NodeListConditions { get; set; } VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualException)
```

Property Value

| Type | Description |
|---|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualException) 1.html) | VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html) |

PathDot

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.VirtualItem) public string PathDot { get; set; } (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>) VirtualItemAdapter<T>
```

Property Value

| Type | Description |
|---------|----------------|
| 1.html) | VirtualItem<T> |

string (https://learn.microsoft.com/dotnet/api/system.string)
(AkiraNetwork.VirtualStorageLibrary.VirtualItem) 1.html)

VirtualNode

PathDotDot

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions) public VirtualNodeContext VirtualNodeContext { get; set; } (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions) VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName) VirtualNodeName
```

Property Value

| Type | Description |
|--|--|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName) VirtualNodeNotFoundException string (https://learn.microsoft.com/dotnet/api/system.string) (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType) VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter) | (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName) VirtualNodeNotFoundException string (https://learn.microsoft.com/dotnet/api/system.string) (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter) |

PathRoot

Declaration

```
public string PathRoot { get; set; }
```



Property Value

| Type | Description |
|--|-------------|
| NotifyNodeDelegate<string> (AkiraNetwork.VirtualStorageLibrary) | |

PathSeparator

Declaration
Directory

```
(AkiraNetwork.VirtualStorageLibrary)  
virtualDirectoryPathSeparator { get; set; }  
(AkiraNetwork.VirtualStorageLibrary)  
1.html)
```

| Type | Description |
|------------------|-------------|
| VirtualException | |

| Type | Description |
|--|-------------|
| VirtualGroupCondition<T, TKey> char (AkiraNetwork.VirtualStorageLibrary) | |

2.html)
VirtualID

PrefixDirectory

Declaration
VirtualItem

```
(AkiraNetwork.VirtualStorageLibrary)  
VirtualItemAdapter<T>  
prefixDirectoryName { get; set; }  
1.html)
```

| Type | Description |
|--|-------------|
| VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary) | |

| Type | Description |
|---|-------------|
| VirtualNode string (AkiraNetwork.VirtualStorageLibrary) | |

PrefixItem

Declaration
VirtualNodeListConditions

```
(AkiraNetwork.VirtualStorageLibrary)  
public string PrefixItem { get; set; }  
virtualNodeName  
(AkiraNetwork.VirtualStorageLibrary)
```

| Type | Description |
|--|-------------|
| VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary) | |

| Type | Description |
|---|-------------|
| VirtualNodeType string (AkiraNetwork.VirtualStorageLibrary) | |

VirtualNodeTypeFilter

PrefixSymbolicLink

Declaration

```
public string PrefixSymbolicLink { get; set; }
```



Property Value

| Type | Description |
|--|-------------|
| (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate<T> string (https://learn.microsoft.com/dotnet/api/system.string)) | |

PatternMatch

(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageSettings.Settings)

Settings

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageSettings.Settings)

Declaration

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageSettings.Settings)

~~public VirtualStorageAdapter<T> VirtualStorageSettings Settings { get; }~~

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageSettings.Settings)

1.html)

Property Value

VirtualException

| Type | Description |
|---|-------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey> VirtualStorageSettings (AkiraNetwork.VirtualStorageLibrary.VirtualStorageSettings.html)) | |

2.html)

VirtualID

WildcardMatcher

VirtualItem

Declaration

(AkiraNetwork.VirtualStorageLibrary.VirtualItem)

VirtualItemAdapter<T>

~~public VirtualWildcardMatcher WildcardMatcher { get; set; }~~

1.html)

Property Value

(AkiraNetwork.VirtualStorageLibrary.VirtualItem)

| Type | Description |
|---|-------------|
| (AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher.html) | |

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher (AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher.html))

(AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions)

Methods

Initialize()

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName)

VirtualNotFoundException

(AkiraNetwork.VirtualStorageLibrary.VirtualNotFoundException)

~~public static void Initialize();~~

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType)

VirtualNodeTypeFilter

Extension Methods

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

▼

(AkiraNetwork.VirtualStorageLibrary
NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrary
PatternMatch
(AkiraNetwork.VirtualStorageLibrary
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary
VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary
VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary
1.html)
VirtualException
(AkiraNetwork.VirtualStorageLibrary
VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary
2.html)
VirtualID
(AkiraNetwork.VirtualStorageLibrary
VirtualItem
(AkiraNetwork.VirtualStorageLibrary
VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary
1.html)
VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary
1.html)
VirtualNode
(AkiraNetwork.VirtualStorageLibrary
VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary
VirtualnodeName
(AkiraNetwork.VirtualStorageLibrary
VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary
VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary
VirtualNodeTypeFilter


```
public char[] InvalidNodeNameCharacters { get; set; }
```

Property Value

| Type | Description |
|---|-------------|
| char (https://learn.microsoft.com/dotnet/api/system.char)[] (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions) | |

NodeListConditions

Declaration
(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector)

```
public VirtualNodeListConditions NodeListConditions { get; set; }  
VirtualDirectory  
(AkiraNetwork.VirtualStorageLibrary.VirtualException)
```

Property Value

| Type | Description |
|--|-------------|
| VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html) (AkiraNetwork.VirtualStorageLibrary.VirtualException) | |

PathDot

Declaration
(AkiraNetwork.VirtualStorageLibrary.VirtualItem)

```
public string PathDot { get; set; }  
(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter)
```

Property Value

| Type | Description |
|---|-------------|
| VirtualItem<T> string (https://learn.microsoft.com/dotnet/api/system.string) | |

PathDotDot

Declaration
(AkiraNetwork.VirtualStorageLibrary.VirtualNode)

```
public VirtualNodeExtensions  
VirtualNodeListConditions  
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext)
```

Property Value

| Type | Description |
|---|-------------|
| VirtualNodeNotFoundException string (https://learn.microsoft.com/dotnet/api/system.string) | |

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter)

VirtualNodeTypeFilter

PathRoot

Declaration

```
public string PathRoot { get; set; }
```



Property Value

| Type | Description |
|--|-------------|
| NotifyNodeDelegate<string> (AkiraNetwork.VirtualStorageLibrary) | |

PathSeparator

Declaration
Directory

```
(AkiraNetwork.VirtualStorageLibrary)
```

VirtualDirectoryAdapter { get; set; }

```
(AkiraNetwork.VirtualStorageLibrary)
```

1.html)

Property Value

| Type | Description |
|------------------------------------|-------------|
| AkiraNetwork.VirtualStorageLibrary | |

PrefixDirectory

Declaration
VirtualItem

```
(AkiraNetwork.VirtualStorageLibrary)
```

VirtualItemAdapter<T>

```
(AkiraNetwork.VirtualStorageLibrary)
```

PrefixDirectory { get; set; }

```
1.html)
```

Property Value

| Type | Description |
|--------|-------------|
| .html) | |

PrefixItem

Declaration
VirtualItemExtensions

```
(AkiraNetwork.VirtualStorageLibrary)
```

VirtualNodeListConditions

```
(AkiraNetwork.VirtualStorageLibrary)
```

PrefixItem { get; set; }

```
VirtualNodeName
```

```
(AkiraNetwork.VirtualStorageLibrary)
```

Property Value

| Type | Description |
|-----------------|-------------|
| VirtualNodeType | |

VirtualStorageLibrary

```
string (https://learn.microsoft.com/dotnet/api/system.string)
```

VirtualNodeTypeFilter

PrefixSymbolicLink

Declaration

```
public string PrefixSymbolicLink { get; set; }
```



Property Value

| Type | Description |
|--|-------------|
| NotifyNodeDelegate<string> (AkiraNetwork.VirtualStorageLibrary) | |

PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

Declaration

Directory
(AkiraNetwork.VirtualStorageLibrary)

```
public VirtualDirectoryAdapter<VirtualStorageState> State { get; }
```

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Property Value

VirtualException

| Type | Description |
|---|---|
| VirtualGroupCondition<T, TKey> VirtualStorageState (AkiraNetwork.VirtualStorageLibrary) | (AkiraNetwork.VirtualStorageLibrary.VirtualStorageState.html) |

2.html)

VirtualID

WildcardMatcher

VirtualItem

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

```
public VirtualWildcardMatcher<T> WildcardMatcher { get; set; }
```

1.html)

Property Value

(AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|--|---|
| IVirtualWildcardMatcher<T> (AkiraNetwork.VirtualStorageLibrary) | (AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher.html) |

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

Methods

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

Set NodeList Conditions(VirtualNodeListConditions)

Declaration

VirtualNotFoundException

```
public static void Set NodeList Conditions(VirtualNodeListConditions conditions)
```

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

Parameters

VirtualNodeTypeFilter

| Type | Name | Description |
|--|------------|-------------|
| VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html) | conditions | |

▼

SetNodeListConditions(VirtualNodeTypeFilter, VirtualGroupCondition<VirtualNode, object>?, List<VirtualSortCondition<VirtualNode>>?)

Declaration
PatternMatch

```
(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector)
public static void SetNodeListConditions(VirtualNodeTypeFilter filter, VirtualGroupCondition<VirtualNode, object>? groupCondition = null, List<VirtualSortCondition<VirtualNode>>? sortConditions = null)
    VirtualDirectory
```

Parameters
VirtualNodeTypeAdapter<T>

| Type | Name | Description |
|--|----------------|-------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeAdapter.html) | filter | |
| VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition.html) | groupCondition | |
| List<VirtualSortCondition<VirtualNode>> (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1)<VirtualSortCondition<VirtualNode>> (AkiraNetwork.VirtualStorageLibrary.VirtualSortCondition-1.html) | sortConditions | |

Extension Methods

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilsilites_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

```
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions)
VirtualNodeListConditions
VirtualNodeName
VirtualNodeType
VirtualNodeTypeFilter
```

Class VirtualStorage<T>

Inheritance

↳ [Object](#) (<https://learn.microsoft.com/dotnet/api/system.object>)
↳ [VirtualStorage<T>](#)
(AkiraNetwork.VirtualStorageLibrary)

Inherited Members

NotifyNodeDelegate

object.Equals(object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object)))
PatternMatch
(AkiraNetwork.VirtualStorageLibrary)
object.Equals(object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object)))
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary)
object.GetHashCode() (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)
object.GetType() (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)
object.MemberwiseClone()
(AkiraNetwork.VirtualStorageLibrary)
object.ReferenceEquals(object1, object2) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)
object.ToString() (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)
1.html)

Namespace: AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)
VirtualException

Assembly: VirtualStorageLibrary.dll

Syntax: VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

public class VirtualStorage<T>

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

Type Parameters:

VirtualItem

| Name | Description |
|-----------------------|--------------------------------------|
| VirtualItemAdapter<T> | |
| T | (AkiraNetwork.VirtualStorageLibrary) |

1.html)

VirtualItem<T>

Constructors

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualNode

VirtualStorage()

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

public VirtualStorage() (AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

Properties

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualNodeTypeFilter

```
public VirtualPath CurrentPath { get; }
```

Property Value

| Type | Description |
|---|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |

CycleDetectorForTarget

Declaration
VirtualCycleDetectorForTarget (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetectorForTarget.html)

Property Value

| Type | Description |
|---|-------------|
| VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html) | |

CycleDetectorForTree

Declaration
VirtualCycleDetectorForTree (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetectorForTree.html)

Property Value
VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html)

| Type | Description |
|---|-------------|
| VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html) | |

Dir

Declaration
Dir (AkiraNetwork.VirtualStorageLibrary.Dir.html)

Property Value
Dir (AkiraNetwork.VirtualStorageLibrary.Dir.html)

| Type | Description |
|---|-------------|
| Dir (AkiraNetwork.VirtualStorageLibrary.Dir.html) <T> | |

this[VirtualPath, bool]

Declaration

```
public VirtualNode this[VirtualPath path, bool followLinks = true] { get; set; }
```



Parameters

| Type | Name | Description |
|--|-------------|-------------|
| NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector) | followLinks | |

Property Value

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory)

| Type | Description |
|--|-------------|
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualNode) VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) 1.html) | |

Item

(AkiraNetwork.VirtualStorageLibrary.VirtualException)

Declaration

```
VirtualGroupCondition<T, TKey>  
(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T> Item { get; })  
2.html)
```

VirtualID

(AkiraNetwork.VirtualStorageLibrary.VirtualItem)

| Type | Description |
|---|-------------|
| VirtualItemAdapter<T> VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter-1.html)<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) | |

Link

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

1.html)

Declaration

```
VirtualNode  
(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLinkAdapter<T> Link { get; })  
VirtualNodeContext
```

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

Property Value

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

| Type | Description |
|--|-------------|
| VirtualNodeListConditions VirtualNodeName <T>(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotNotFoundException) (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType) VirtualNodeTypeFilter | |

LinkDictionary

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

VirtualNodeTypeFilter

Declaration

```
public Dictionary<VirtualPath, HashSet<VirtualPath>> LinkDictionary { get; }
```

| Property | Value |
|--|--|
| Type | Description |
| Dictionary<VirtualPath, VirtualStorageLibrary> | (AkinaNetwork.VirtualStorageLibrary/dotnet/api/system.collections.generic.dictionary-2) < VirtualPath> (AkinaNetwork.VirtualStorageLibrary.VirtualPath.html), HashSet<VirtualPath> (AkinaNetwork.VirtualStorageLibrary/dotnet/api/system.collections.generic.hashset-1)< VirtualPath> (AkinaNetwork.VirtualStorageLibrary.VirtualPath.html)>> |

| Root | (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector) |
|--------------|--|
| Declarations | (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory) |
| Type | VirtualDirectoryAdapter<T> |
| Value | (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.Root { get; } 1.html) |
| Type | VirtualException |
| Type | VirtualGroupCondition<T, TKey> |
| Type | (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory 2.html) |
| Description | (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) |

Methods

AddDirectory(VirtualPath, VirtualDirectory, bool)

VirtualItem<T>

Declaration

```
    VirtualNode AddDirectory(VirtualPath directoryPath, VirtualDirectory directory, bool createsSubdirectories, virtualFileLibrary)
```

VirtualDiskContent

VirtualNodeContext

Parameters (AkiraNetwork.VirtualStorageLibrary)

| Parameters | | | |
|--|----------------------|-------------|--|
| Type | Name | Description | |
| VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary) | | | |
| VirtualNodeListConditions VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary) | directoryPath | | |
| VirtualDirectoryName (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) (AkiraNetwork.VirtualStorageLibrary) | directory | | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) VirtualNodeNotFoundException | createSubdirectories | | |

VirtualNodeNetIboundException (AkiraNetworks VirtualStorage library)

(AkiraNetwork.VI)

VirtualNodeType

Virtual Directories

Declaration

```
public void AddDirectory(VirtualPath path, bool createSubdirectories = false)
```

Parameters

| Type | Name | Description |
|--|----------------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary) | createSubdirectories | |

AddItem(VirtualPath, VirtualItem<T>, bool)

(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
VirtualDirectory  
    (AkiraNetwork.VirtualStorageLibrary)  
VirtualDirectoryAdapter  
    (AkiraNetwork.VirtualStorageLibrary)  
VirtualPath AddItem(VirtualPath itemDirectoryPath, VirtualItem<T> item, bool overwrite = false)  
    (AkiraNetwork.VirtualStorageLibrary)  
        1.html)
```

Parameters

| Type | Name | Description |
|--|-------------------|-------------|
| VirtualGroupCondition<T, TKey> | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) 2.html) | itemDirectoryPath | |
| VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html)<T> bool (https://learn.microsoft.com/dotnet/api/system.boolean) VirtualItem (AkiraNetwork.VirtualStorageLibrary) VirtualItemAdapter<T> | item | |
| | overwrite | |

AddItem(VirtualPath, T?, bool)

Declaration

```
VirtualItem<T>  
    (AkiraNetwork.VirtualStorageLibrary)
```

```
public void AddItem(VirtualPath itemPath, T? data = default, bool overwrite = false)  
    .html)
```

Parameters

| Type | Name | Description |
|---|-----------|-------------|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary) | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary) | itemPath | |
| T VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary) bool (https://learn.microsoft.com/dotnet/api/system.boolean) VirtualNodeName (AkiraNetwork.VirtualStorageLibrary) VirtualNodeNotFoundException | data | |
| | overwrite | |

AddLinkToDictionary(VirtualPath, VirtualPath)

Declaration

```
VirtualNodeType  
    (AkiraNetwork.VirtualStorageLibrary)  
VirtualNodeTypeFilter
```

```
public void AddLinkToDictionary(VirtualPath targetPath, VirtualPath linkPath)
```

Parameters

| Type | Name | Description |
|---|------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | targetPath | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | linkPath | |

(AkiraNetwork.VirtualStorageLibrary.

PatternMatch

AddNode(VirtualPath, VirtualNode, bool)

VirtualCycleDetector

Declaration (AkiraNetwork.VirtualStorageLibrary.

VirtualDirectory

```
public void AddNode(VirtualPath nodeDirectoryPath, VirtualNode node, bool overwrite = false)
```

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.

1.html)

Type VirtualException

Name

Description

(AkiraNetwork.VirtualStorageLibrary.

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

nodeDirectoryPath

VirtualGroupCondition<T, TKey>

VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

node

2.html)

bool (<https://learn.microsoft.com/dotnet/api/system.boolean>)

overwrite

VirtualID

AddSymbolicLink(VirtualPath, VirtualPath?, bool)

VirtualItemAdapter<T>

Declaration (AkiraNetwork.VirtualStorageLibrary.

1.html)

```
public void AddSymbolicLink(VirtualPath linkPath, VirtualPath? targetPath = null, bool overwrite = false)
```

(AkiraNetwork.VirtualStorageLibrary.

1.html)

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.

Type VirtualNodeContext

Name

Description

(AkiraNetwork.VirtualStorageLibrary.

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

linkPath

VirtualNodeExtensions

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

targetPath

VirtualNodeListConditions

bool (<https://learn.microsoft.com/dotnet/api/system.boolean>)

overwrite

(AkiraNetwork.VirtualStorageLibrary.

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary.

AddSymbolicLink(VirtualPath, VirtualSymbolicLink, bool)

VirtualNodeAdapter<T>

Declaration (AkiraNetwork.VirtualStorageLibrary.

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeTypeFilter

```
public void AddSymbolicLink(VirtualPath linkDirectoryPath, VirtualSymbolicLink link, bool overwrite = false)
```

Parameters

| Type | Name | Description |
|--|--------------------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>linkDirectoryPath</i> | |
| NotifyNodeDelegate VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) PatternMatch | <i>link</i> | |
| bool (AkiraNetwork.VirtualStorageLibrary.net/api/system.boolean) | <i>overwrite</i> | |

ChangeDirectory(VirtualPath)

Declaration

```
VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary
1.html)
public void ChangeDirectory(VirtualPath path)
VirtualException
(AkiraNetwork.VirtualStorageLibrary)
```

Parameters

VirtualGroupCondition<T, TKey>

| Type | Name | Description |
|---|-------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>path</i> | |

ConvertToAbsolutePath(VirtualPath?, VirtualPath?)

Declaration

```
AkiraNetwork.VirtualStorageLibrary
1.html)
VirtualItem<T> ConvertToAbsolutePath(VirtualPath? relativePath, VirtualPath? basePath =
(AkiraNetwork.VirtualStorageLibrary
1.html)
```

Parameters

VirtualNodeContext

| Type | Name | Description |
|--|---------------------|-------------|
| VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html) | <i>relativePath</i> | |

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

| Type | Name | Description |
|---|-----------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>basePath</i> | |

VirtualNodeListConditions

VirtualNodeName

| Type | Description |
|---|-------------|
| VirtualNodeNotFoundException VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |

VirtualNodeType

VirtualStorageLibrary

CopyNode(VirtualPath, VirtualPath, bool, bool, List<VirtualNodeContext>?)

Declaration

```
public void CopyNode(VirtualPath sourcePath, VirtualPath destinationPath, bool overwrite = false, bool recursive = false, bool followLinks = false, List<VirtualNodeContext>? destinationContextList = null)
```



Parameters

| Type | Name | Description |
|---|------------------------|-------------|
| NotifyNodeDelegate | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | sourcePath | |
| PatternMatch | destinationPath | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | overwrite | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | recursive | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | followLinks | |
| VirtualDirectoryAdapter<T> List (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1) (AkiraNetwork.VirtualStorageLibrary.VirtualException 1) 1.html < VirtualException VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html) (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey> > (AkiraNetwork.VirtualStorageLibrary.VirtualID 2.html) VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualItem VirtualItem VirtualItemAdapter<T> p(AkiraNetwork.VirtualStorageLibrary.VirtualPath path, bool followLinks = false) 1.html) | destinationContextList | |

DirectoryExists(VirtualPath, bool)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.VirtualPath  
VirtualItem  
VirtualItemAdapter<T>  
p(AkiraNetwork.VirtualStorageLibrary.VirtualPath path, bool followLinks = false)  
1.html)
```

Parameters

| Type | Name | Description |
|---|-------------|-------------|
| .html) | | |
| VirtualNode VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary.VirtualException VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) | path | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | followLinks | |

Returns

| Type | Description |
|---------------------------|-------------|
| VirtualNodeListConditions | |

ExpandPath(VirtualPath, VirtualNodeTypeFilter, bool, bool)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.VirtualPath  
VirtualNodeType  
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter
```

```
public IEnumerable<VirtualPath> ExpandPath(VirtualPath path, VirtualNodeTypeFilter filter =
VirtualNodeTypeFilter.All, bool followLinks = true, bool resolveLinks = true)
```

Parameters

| Type | Name | Description |
|--|--------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | |
| NotifyNodeDelegate VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) PatternMatch | filter | |
| bool (AkiraNetwork.VirtualStorageLibrary.html (https://learn.microsoft.com/dotnet/api/system.boolean)) | followLinks | |
| VirtualCycleDetector bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary) | resolveLinks | |

VirtualDirectory

Returns
(AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|--|-------------|
| VirtualDirectoryAdapter<T> | |
| (AkiraNetwork.VirtualStorageLibrary) | |
| IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1)< VirtualPath> (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |
| VirtualException | |

ExpandPathTree(VirtualPath, VirtualNodeTypeFilter, bool, bool)

Declaration

```
virtual IEnumerable<VirtualNodeContext> ExpandPathTree(VirtualPath path, VirtualNodeTypeFilte
r (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.All, bool followLinks = true, bool resolveLinks = true)
```

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

Parameters
1.html)

| Type | Name | Description |
|---|--------------|-------------|
| VirtualItem<T> | | |
| (AkiraNetwork.VirtualStorageLibrary) | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | |
| VirtualNodeTypeFilter | filter | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | | |
| VirtualNodeContext | | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | followLinks | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | resolveLinks | |

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

Type VirtualnodeName

| Type | Description |
|---|-------------|
| (AkiraNetwork.VirtualStorageLibrary) | |
| IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1)< VirtualNodeNotFoundException> | |
| VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html) | |
| (AkiraNetwork.VirtualStorageLibrary) | |

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

GetDirectory(VirtualPath, bool)

Returns

| Type | Description |
|--|-------------|
| HashSet (https://learn.microsoft.com/dotnet/api/system.collections.generic.hashset-1)<VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)> | |

▼

GetNode(VirtualPath, bool)

Declaration
(AkiraNetwork.VirtualStorageLibrary)

PatternMatch
public VirtualNode GetNode(VirtualPath path, bool followLinks = false)
(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary)

Parameters
Type
(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|---|------|-------------|
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary) | | |
| VirtualDirectoryAdapter (AkiraNetwork.VirtualStorageLibrary) | path | |

VirtualPath (<https://learn.microsoft.com/dotnet/api/system.string>)

bool (<https://learn.microsoft.com/dotnet/api/system.boolean>)

VirtualException

Returns
(AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|--|-------------|
| VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary) | |

VirtualNode (<https://learn.microsoft.com/dotnet/api/system.object>)

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

GetNodeType(VirtualPath, bool)

Declaration
VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary)

1.html
public VirtualNodeType GetNodeType(VirtualPath path, bool followLinks = false)
VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary)

Parameters
Type

| Type | Name | Description |
|---|------|-------------|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary) | | |
| VirtualPath (https://learn.microsoft.com/dotnet/api/system.string) | path | |

VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary)

bool (<https://learn.microsoft.com/dotnet/api/system.boolean>)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

Returns
VirtualNodeListConditions

| Type | Description |
|--|-------------|
| VirtualNodeName VirtualNodeType (https://learn.microsoft.com/dotnet/api/system.object) | |

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

GetNodes(VirtualNodeTypeFilter, bool, bool)

Declaration
VirtualNodeTypeFilter

```
public IEnumerable<VirtualNode> GetNodes(VirtualNodeTypeFilter nodeType = VirtualNodeTypeFilter.All, bool recursive = false, bool followLinks = false)
```

Parameters

| Type | Name | Description |
|--|-------------|-------------|
| VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | nodeType | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) PatternMatch | recursive | |
| bool (AkiraNetwork.VirtualStorageLibrary.dotnet/api/system.boolean) | followLinks | |

Returns

| Type | Description |
|--|-------------|
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary) | |

GetNodes(VirtualPath, VirtualNodeTypeFilter, bool, bool)

Declaration

```
public IEnumerable<VirtualNode> GetNodes(VirtualPath basePath, VirtualNodeTypeFilter nodeTyp  
(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)
```

Parameters

| Type | Name | Description |
|---|-------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | basePath | |
| VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | nodeType | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | recursive | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary) | followLinks | |

Returns

| Type | Description |
|---|-------------|
| VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary) | |

GetNodesWithPaths(VirtualNodeTypeFilter, bool, bool)

Declaration

```
public IEnumerable<VirtualNode> GetNodesWithPaths(VirtualNodeTypeFilter nodeType = VirtualNod  
(AkiraNetwork.VirtualStorageLibrary)
```

Parameters

| Type | Description |
|---|-------------|
| VirtualNodeNotNotFoundException (AkiraNetwork.VirtualStorageLibrary) | |

```
public IEnumerable<VirtualPath> GetNodesWithPaths(VirtualNodeTypeFilter nodeType = VirtualNodeTypeFilter.All, bool recursive = false, bool followLinks = false)
```

Parameters

| Type | Name | Description |
|--|-------------|-------------|
| VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | nodeType | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) PatternMatch | recursive | |
| bool (AkiraNetwork.VirtualStorageLibrary.dotnet/api/system.boolean) | followLinks | |

Returns

| Type | Description |
|--|-------------|
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary) | |

GetNodesWithPaths(VirtualPath, VirtualNodeTypeFilter, bool, bool)

Declaration

```
public IEnumerable<VirtualPath> GetNodesWithPaths(VirtualPath basePath, VirtualNodeTypeFilter nodeType = VirtualNodeTypeFilter.All, bool recursive = false, bool followLinks = false)
    VirtualItem
    (AkiraNetwork.VirtualStorageLibrary)
```

Parameters

| Type | Name | Description |
|--|-------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | basePath | |
| VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | nodeType | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | recursive | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | followLinks | |

Returns

| Type | Description |
|---|-------------|
| VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary) | |

GetSymbolicLink(VirtualPath)

Declaration

```
VirtualNodeFilter
```

```
public VirtualSymbolicLink GetSymbolicLink(VirtualPath path)
```

Parameters

| Type | Name | Description |
|---|------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary) | path | |

Returns

| Type | Description |
|---|-------------|
| VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.VirtualSymbolicLink.html) | |

ItemExists(VirtualPath, bool)

Declaration

```
public bool ItemExists(VirtualPath path, bool followLinks = false)  
    VirtualException
```

Parameters

| Type | Name | Description |
|---|-------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) VirtualItem | followLinks | |

Returns

| Type | Description |
|-----------------------|-------------|
| VirtualItemAdapter<T> | |

MoveNode(VirtualPath, VirtualPath, bool, bool)

Declaration

```
public void MoveNode(VirtualPath sourcePath, VirtualPath destinationPath, bool overwrite = false)  
    VirtualNodeExtensions (VirtualPath sourcePath, VirtualPath destinationPath, bool overwrite = false)  
    VirtualNodeContext
```

Parameters

| Type | Name | Description |
|---|-----------------|-------------|
| VirtualNodeNotNotFoundException | sourcePath | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary) | destinationPath | |
| VirtualNodeContext | overwrite | |

| Type | Name | Description |
|--|---------------------|-------------|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | <i>resolveLinks</i> | |

NodeExists(VirtualPath, bool)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    NotifyNodeDelegate
public bool NodeExists(VirtualPath path, bool followLinks = false)
(AkiraNetwork.VirtualStorageLibrary)
    PatternMatch
```

Parameters

| Type | Name | Description |
|--|-----------------------------------|-------------|
| VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary) | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) bool (https://learn.microsoft.com/dotnet/api/system.boolean) | <i>path</i> <i>followLinks</i> | |

Returns

| Type | Description |
|--|-------------|
| VirtualException (AkiraNetwork.VirtualStorageLibrary) | |

RemoveLinkByLinkPath(VirtualPath)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
VirtualItemAdapter<T>
public void RemoveLinkByLinkPath(VirtualPath linkPath)
(AkiraNetwork.VirtualStorageLibrary)
    1.html)
```

Parameters

| Type | Name | Description |
|--|-----------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>linkPath</i> | |

RemoveLinkFromDictionary(VirtualPath, VirtualPath)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeListConditions
public void RemoveLinkFromDictionary(VirtualPath targetPath, VirtualPath linkPath)
    VirtualnodeName
```

Parameters

| Type | Name | Description |
|---|-------------------|-------------|
| VirtualNodeType VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>targetPath</i> | |

VirtualNodeTypeFilter

| Type | Name | Description |
|---|----------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | linkPath | |

RemoveNode(VirtualPath, bool, bool, bool)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    NotifyNodeDelegate
public void RemoveNode(VirtualPath nodePath, bool recursive = false, bool followLinks = false,
(AkiraNetwork.VirtualStorageLibrary)
    bool resolveLinks = true)
PatternMatch
(AkiraNetwork.VirtualStorageLibrary)
```

Parameters

| Type | Name | Description |
|---|--------------|-------------|
| VirtualDirectory | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | nodePath | |
| VirtualDirectoryAdapter<T> | | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | recursive | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | followLinks | |
| VirtualException | | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | resolveLinks | |

ResolveLinkTarget(VirtualPath)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    VirtualItem
public VirtualItem Target(VirtualPath path)
    VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary)
```

Parameters

| Type | Name | Description |
|---|------|-------------|
| VirtualItem<T> | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | |

Returns

| Type | Description |
|---|-------------|
| VirtualNodeContext | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |

SetLinkTargetNodeType(HashSet<VirtualPath>, VirtualNodeType)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    VirtualNotFoundException
public void SetLinkTargetNodeType(HashSet<VirtualPath> linkPathSet, VirtualNodeType nodeType)
    VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary)
    VirtualNodeTypeFilter
```

Parameters

| Type | Name | Description |
|---|--------------------|-------------|
| HashSet (https://learn.microsoft.com/dotnet/api/system.collections.generic.hashset-1) <VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)> ▼ | <i>linkPathSet</i> | |
| VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html) | <i>nodeType</i> | |

SetNode(VirtualPath, VirtualNode)

PatternMatch

DeclarationNetwork.VirtualStorageLibrary

VirtualCycleDetector

(AkiraNetwork VirtualStorage library)

VirtualDirectory

(AkiraNetwork VirtualStorage) library

(AKRAN) Parameters

VirtualDirectoryAdapter< I

| Type | Description | Name | Description |
|--|-------------|-----------------|-------------|
| AkiraNetwork.VirtualStorageLibrary 1.html) | | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) VirtualException | | destinationPath | |
| (AkiraNetwork.VirtualStorageLibrary VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | | node | |

SetNodeName(VirtualPath, VirtualnodeName, bool)

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork VirtualStorage) library

```
public void SetNodeName(VirtualPath nodePath, VirtualNodeName newName, bool resolveLinks = true)
```

(AkiraNetwork.VirtualStorageLibrary)

[1.html](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1111111/)

1.html)

Parameters

(AkiraNetwork.VirtualStorageLibrary
1.html)

| Type | Description | Name |
|---|-------------|---------------------|
| (AkiraNetwork.VirtualStorageLibrary.html) | | |
| VirtualPathNode (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | | <i>nodePath</i> |
| (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) | | <i>newName</i> |
| VirtualNodeContext | | |
| bool (AkiraNetwork.VirtualStorageLibrary.net/api/system.boolean) | | <i>resolveLinks</i> |

SymbolicLinkExists(VirtualPath)
(MAXTRANSNODENAME) throws an exception

▶ VirtualNodeName

(AkiraNetwork VirtualStorage) library

VirtualNodeNetFoundException

```
public bool SymbolicLinkExists(  
    (AkiraNetwork)virtualStorageLibrary
```

VirtuaNandaTrunc

VirtualNode type

| Type | Name | Description |
|---|------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | |

Returns

| Type | Description |
|---|-------------|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate) | |

TryGetDirectory(VirtualPath, bool)

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory? TryGetDirectory(VirtualPath path, bool followLinks = false))

Parameters

| Type | Name | Description |
|---|-------------|-------------|
| VirtualPath (https://learn.microsoft.com/dotnet/api/system.boolean) | path | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | followLinks | |

Returns

| Type | Description |
|--|-------------|
| VirtualDirectory (https://learn.microsoft.com/dotnet/api/system.boolean) | |

TryGetItem(VirtualPath, bool)

(AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>? TryGetItem(VirtualPath path, bool followLinks = false))

Parameters

| Type | Name | Description |
|---|-------------|-------------|
| VirtualPath (https://learn.microsoft.com/dotnet/api/system.boolean) | path | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | followLinks | |

Returns

| Type | Description |
|--|-------------|
| VirtualItem<T> (https://learn.microsoft.com/dotnet/api/system.boolean) | |

TryGetNode(VirtualPath, bool)

Declaration

```
public VirtualNode? TryGetNode(VirtualPath path, bool followLinks = false)
```



Parameters

| Type | Name | Description |
|---|--------------------|-------------|
| (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate)(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>path</i> | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean)(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector) | <i>followLinks</i> | |

Returns

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory)

| Type | Description |
|---|-------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNode)(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | |

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T>)(AkiraNetwork.VirtualStorageLibrary.VirtualFile)(1.html)

TryGetSymbolicLink(VirtualPath)

Declaration

```
VirtualGroupCondition<T, TKey>
public VirtualSymbolicLink? TryGetSymbolicLink(VirtualPath path)
```

VirtualID

Parameters

| Type | Name | Description |
|---|-------------|-------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualItem)(VirtualItemAdapter<T>)(VirtualPath)(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>path</i> | |

1.html)

Returns

| Type | Description |
|--|-------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)(VirtualSymbolicLink.html) | |

VirtualNode

TryResolveLinkTarget(VirtualPath)

Declaration

```
VirtualNodeExtensions
public VirtualPath? TryResolveLinkTarget(VirtualPath path)
```

Parameters

| Type | Name | Description |
|--|-------------|-------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException)(VirtualPath)(VirtualNodeType)(VirtualNodeTypeFilter) | <i>path</i> | |

Returns

| Type | Description |
|---|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | |

UpdateAllTargetNodeTypesInDictionary()

(AkiraNetwork.VirtualStorageLibrary)
Declaration
NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrary)
public void UpdateAllTargetNodeTypesInDictionary()
PatternMatch

(AkiraNetwork.VirtualStorageLibrary)
VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)
VirtualDirectory

UpdateDirectory(VirtualPath, VirtualDirectory)

(AkiraNetwork.VirtualStorageLibrary)
Declaration
VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary)
public void UpdateDirectory(VirtualPath directoryPath, VirtualDirectory newDirectory)
1.html)

VirtualException

Parameters
(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|---|---------------|-------------|
| VirtualGroupCondition<T, TKey> | | |
| (AkiraNetwork.VirtualStorageLibrary) VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | directoryPath | |

VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html)

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|-------------|------|-------------|
| VirtualItem | | |

(AkiraNetwork.VirtualStorageLibrary)

UpdateItem(VirtualPath, VirtualItem<T>)

(AkiraNetwork.VirtualStorageLibrary)
Declaration
1.html)
VirtualItem<T>
public void UpdateItem(VirtualPath itemPath, VirtualItem<T> newItem)
(AkiraNetwork.VirtualStorageLibrary)
1.html)

Parameters

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|---|----------|-------------|
| VirtualNodeContext | | |
| (AkiraNetwork.VirtualStorageLibrary) VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | itemPath | |

VirtualNodeExtensions

VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html)<T>

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

UpdateLinkInDictionary(VirtualPath, VirtualPath)

(AkiraNetwork.VirtualStorageLibrary)
Declaration
VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeType
public void UpdateLinkInDictionary(VirtualPath linkPath, VirtualPath newTargetPath)
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeTypeFilter

Parameters

| Type | Name | Description |
|---|----------------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>linkPath</i> | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>newTargetPath</i> | |

▼

UpdateLinksToTarget(VirtualPath, VirtualPath)

(AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate)

Declaration: AkiraNetwork.VirtualStorageLibrary.

PatternMatch

```
public void UpdateLinksToTarget(VirtualPath oldTargetPath, VirtualPath newTargetPath)
    VirtualCycleDetector
```

(AkiraNetwork.VirtualStorageLibrary.

Parameters:

VirtualDirectory

| Type | Name | Description |
|---|----------------------|-------------|
| VirtualDirectoryAdapter<T> | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>oldTargetPath</i> | |

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualException

(AkiraNetwork.VirtualStorageLibrary.

VirtualGroupCondition<T, TKey>

UpdateSymbolicLink(VirtualPath, VirtualSymbolicLink)

(AkiraNetwork.VirtualStorageLibrary.

2.html)

Declaration:

VirtualID

(AkiraNetwork.VirtualStorageLibrary.

```
public void UpdateSymbolicLink(VirtualPath linkPath, VirtualSymbolicLink newLink)
```

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.

VirtualItemAdapter<T>

| Type | Name | Description |
|---|-----------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>linkPath</i> | |
| VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) | <i>newLink</i> | |

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.

```
public void UpdateTargetNodeTypesInDictionary(VirtualPath targetPath)
```

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.

Parameters:

VirtualNodeNotFoundException

| Type | Name | Description |
|---|-------------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>targetPath</i> | |

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeTypeFilter

WalkPathToTarget(VirtualPath, NotifyNodeDelegate?, ActionNodeDelegate?, bool, bool)

Declaration

```
public VirtualNodeContext WalkPathToTarget(VirtualPath targetPath, NotifyNodeDelegate? notifyNode = null, ActionNodeDelegate? actionNode = null, bool followLinks = true, bool exceptionEnabled = true)
```

Parameters
NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate.html)

Type
patternMatch

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)
VirtualCycleDetector

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate.html)

ActionNodeDelegate

VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.ActionNodeDelegate.html)
(AkiraNetwork.VirtualStorageLibrary)

bool (https://learn.microsoft.com/dotnet/api/system.boolean)

VirtualException

bool (https://learn.microsoft.com/dotnet/api/system.boolean)

VirtualGroupCondition<T, TKey>

Return
(AkiraNetwork.VirtualStorageLibrary)

Type
? html)

VirtualID

VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

WalkPathTree(VirtualPath, VirtualNodeTypeFilter, bool, bool, bool)

Declaration

VirtualItem<T>

```
public IEnumerable<VirtualNodeContext> WalkPathTree(VirtualPath basePath, VirtualNodeTypeFilter filter = VirtualNodeTypeFilter.All, bool recursive = true, bool followLinks = true, bool resolveLinks = true)
```

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

Type
VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)
VirtualNodeListConditions

VirtualNodeTypeFilter

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

VirtualNotFoundException

bool (https://learn.microsoft.com/dotnet/api/system.boolean)

VirtualNodeNotFoundException

bool (https://learn.microsoft.com/dotnet/api/system.boolean)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html)

VirtualNodeTypeFilter

| Type | Name | Description |
|---|--------------|-------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath)(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | basePath | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | filter | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | recursive | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | followLinks | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | resolveLinks | |

Returns

| Type | Description |
|--|-------------|
| IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1)<VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html)> | |

▼

Extension Methods

| | |
|---|--|
| (AkiraNetwork.VirtualStorageLibrary) | |
| VirtualTextFormatter.GenerateLinkTableDebugText<T>(VirtualStorage<T>) | (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilsilites_VirtualTextFormatter_GenerateLinkTableDebugText_1_AkiraNetwork_VirtualStorageLibrary_VirtualStorage_0_) |
| VirtualTextFormatter.GenerateLinkTableDebugText<T>(T) | (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilsilites_VirtualTextFormatter_GenerateLinkTableDebugText_1_0_) |
| VirtualTextFormatter.GenerateTreeDebugText<T>(VirtualStorage<T>, VirtualPath, bool, bool) | (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilsilites_VirtualTextFormatter_GenerateTreeDebugText_1_AkiraNetwork_VirtualStorageLibrary_VirtualStorage_0_AkiraNetwork_VirtualStorageLibrary_VirtualPath_System_Boolean_System_Boolean_1.html) |
| VirtualException | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualGroupCondition<T, TKey> | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualItemAdapter<T> | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualItem<T> | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualItem | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNode | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeContext | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeExtensions | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeListConditions | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualnodeName | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeNotFoundException | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeType | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeTypeFilter | |

Class VirtualSymbolicLink

Represents a virtual symbolic link. A virtual symbolic link provides a virtual reference to other nodes (directories, items, files, etc.) This class represents symbolic links within the virtual storage library and manages paths to other nodes.

(AkiraNetwork.VirtualStorageLibrary)

The virtual symbolic link supports the concept of NULL links. When the TargetPath is null, the link resolution is not performed during path traversal. This feature can be useful when the target is undefined or dynamically determined.

PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

Inheritance

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

↳ object (<https://learn.microsoft.com/dotnet/api/system.object>)

↳ VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

Implements

(AkiraNetwork.VirtualStorageLibrary)

IVirtualDeepCloneable (AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html) <VirtualNode

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)>

Inherited Members

VirtualGroupCondition<T, TKey>

VirtualNode.Name

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Name)

2.html)

VirtualNode.CreatedDate

VirtualID

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_CreatedD

ate)

VirtualItem

VirtualNode.UpdatedDate

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Updated

Date)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNode.VID

1.html)

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_VID)

VirtualNode.IsReferencedInStorage

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_IsReferen

cedinStorage)

VirtualNode

object.Equals(object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-object)))

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

object.Equals(object, object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-object-equals(system-object-system-object)))

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

object.GetHashCode() (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)

(AkiraNetwork.VirtualStorageLibrary)

object.GetType() (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)

(VirtualNodeListConditions)

object.MemberwiseClone() (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)

(AkiraNetwork.VirtualStorageLibrary)

object.ReferenceEquals(object, object) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)

VirtualNodeName

Name (AkiraNetwork.VirtualStorageLibrary.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

Syntax

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

```
public class VirtualSymbolicLink : VirtualNode, IVirtualDeepCloneable<VirtualNode>
```

Constructors

VirtualSymbolicLink()

NotifyNodeDelegate

Initializes a new instance of the VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) class. The node name is automatically generated using the VirtualNodeName.GenerateNodeName method.

PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

```
public VirtualSymbolicLink()
```

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

VirtualSymbolicLink(VirtualNodeName)

1.html)

VirtualException

Initializes a new instance of the VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)

class (AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

Declaration

(AkiraNetwork.VirtualStorageLibrary)

2.html)

```
public VirtualSymbolicLink(VirtualNodeName name)
```

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

Parameters

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|-----------------------|------|-------------|
| VirtualItemAdapter<T> | | |

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

name

The name of the symbolic link.

1.html)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualSymbolicLink(VirtualNodeName, VirtualPath?)

(AkiraNetwork.VirtualStorageLibrary)

Initializes a new instance of the VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)

VirtualNodeContext

class with the specified name and target path.

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

```
public VirtualSymbolicLink(VirtualNodeName name, VirtualPath? targetPath)
```

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

Parameters

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|------------------------------|------|-------------|
| VirtualNodeNotFoundException | | |

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

name

The name of the symbolic link.

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

VirtualNodeTypeFilter

| Type | Name | Description |
|---|-------------------|---------------------------------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>targetPath</i> | The target path of the symbolic link. |

VirtualSymbolicLink(VirtualNodeName, VirtualPath?, DateTime, DateTime)

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)
 Initializes a new instance of the VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) class with the specified name, target path, creation date, and update date.
 (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)
public VirtualSymbolicLink(VirtualNodeName name, VirtualPath? targetPath, DateTime createdDate,
                           DateTime updatedDate)
  : base(updatedDate)
  {
    VirtualDirectory
    (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)
  }
```

Parameters

VirtualDirectoryAdapter<T>

| Type | Name | Description |
|---|--------------------|---------------------------------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) | <i>name</i> | The name of the symbolic link. |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>targetPath</i> | The target path of the symbolic link. |
| DateTime (https://learn.microsoft.com/dotnet/api/system.datetime) | <i>createdDate</i> | The creation date. |
| DateTime (https://learn.microsoft.com/dotnet/api/system.datetime) | <i>updatedDate</i> | The update date. |

Properties

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

1.html

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

1.html

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

VirtualNodeContext

Gets the type of the node. This class always returns SymbolicLink.

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)
public VirtualNodeListCondition<T> NodeType { get; }
```

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

Overrides

VirtualNodeTypeFilter

Description

| Type | Description |
|-----------------|-------------|
| VirtualNodeType | |

VirtualNode.NodeType

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_NodeType)

TargetNodeType

Gets or sets the type of the target node of the symbolic link.

(AkiraNetwork.VirtualStorageLibrary)
Declaration
Akira Node Developers

```
    NotifyNodeDelegate  
    (AkiraNetwork.VirtualStorageLibrary  
public VirtualNodeType TargetNodeType { get; set; }  
    patternMatch
```

VirtualCycleDetector

Property Value
(AkiraNetwork.VirtualStorageLibrary)

| Type | (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory) | Description |
|-------|---|---|
| Virtu | (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.VirtualNodeType.html) VirtualDirectory | AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.VirtualNodeType.html VirtualDirectory |

TargetPath

- **VirtualException**
 (AkiraNetwork.VirtualStorageLibrary)
Gets or sets the target path of the symbolic link.

Declaration Network.VirtualStorageLibrary

<VirtualDir virtualPath>, TargetPath

```
public VirtualPath TargetPath { get; set; }  
(AkiraNetwork.VirtualStorageLibrary)
```

VirtualItem
Property Value `VirtualItem`

Type VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualPath) (AkiraNetwork.VirtualStorageLibrary.VirtualPath)

VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary
1.html)

| Type | VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)) | Description |
|------|---|-------------|
|------|---|-------------|

Methods

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

DeepClone(book)

VirtualNodeExtensions
Creates a deep clone of the symbolic link.
AkiraNetwork.VirtualStorageLibrary

VirtualNodeListConditions Declaration

(AkiraNetwork.VirtualStorageLibrary
VirtualNodeName

```
    public override VirtualNode DeepClone()
        (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName)
```

VirtualNodeNotFoundException

Param(~~After~~) **Network.VirtualStorageLibrary**
VirtualNodeType

VirtualNodeType
(AkiraNetwork.VirtualStorageLibrar

VirtualNodeTypeFilter

| Type | Name | Description |
|--|-----------|--|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | recursive | If true, all links are recursively cloned. |

Returns

| Type | Description |
|---|------------------------------------|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) (AkiraNetwork.VirtualStorageLibrary.VirtualNode) | A deep clone of the symbolic link. |

Overrides

[NotifyNodeDelegate](#)
[VirtualNode.DeepClone\(bool\)](#)
[PatternMatch](#)
[VirtualNode.DeepClone\(bool\)](#)
[\(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_DeepClone_System_Boolean\)](#)
[VirtualCycleDetector](#)

[VirtualException](#)
[VirtualDirectory](#)
[VirtualDirectoryAdapter<T>](#)

ToString()
 Returns a string representation of the symbolic link.
[\(AkiraNetwork.VirtualStorageLibrary.VirtualItem\)](#)

Declaration

[VirtualException](#)
[p\(AkiraNetwork.VirtualStorageLibrary\)](#)

[VirtualGroupCondition<T, TKey>](#)
[\(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey>\)](#)

Returns
[2.html](#)

| Type | Description |
|---|---|
| VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualID) | A string representation of the symbolic link. |

Overrides

[VirtualItemAdapter<T>](#)
[object.ToString\(\) \(\[https://learn.microsoft.com/dotnet/api/system.object.tostring\]\(#\)\)](#)

[1.html](#)
[VirtualItem<T>](#)

Update(VirtualNode)
[\(AkiraNetwork.VirtualStorageLibrary.VirtualNode\)](#)

Updates the current symbolic link with the data from the specified node.

[VirtualNode](#)

Declaration
[\(AkiraNetwork.VirtualStorageLibrary\)](#)

[VirtualNodeContext](#)

[p\(AkiraNetwork.VirtualStorageLibrary.VirtualNode node\)](#)

[VirtualNodeExtensions](#)

[\(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions\)](#)

Parameters
[VirtualNodeListConditions](#)

| Type | Name | Description |
|--|----------------------|---------------------------------|
| AkiraNetwork.VirtualStorageLibrary.VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName) | node | The node to use for the update. |

Overrides

[\(AkiraNetwork.VirtualStorageLibrary\)](#)

[VirtualNode.Update\(VirtualNode\)](#)

[\(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Update_AkiraNetwork_VirtualStorageLibrary_VirtualNode_\)](#)

| Type | Name | Description |
|--|-------|-----------------------|
| (VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) nodeName (https://learn.microsoft.com/dotnet/api/system.valuetuple-akiranetwork.virtualstoragelibrary.virtualnodename,akiranetwork.virtualstoragelibrary.virtualpath-.nodename) , VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) targetPath (https://learn.microsoft.com/dotnet/api/system.valuetuple-(AkiraNetwork.VirtualStorageLibrary.Virtualnodename,akiranetwork.virtualstoragelibrary.virtualpath!.targetpath))) (AkiraNetwork.VirtualStorageLibrary.PatternMatch (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector Returns VirtualCycleDetector Type (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualException IVirtualDeepCloneable<T> (AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html) (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualItem VirtualNodeExtensions.ResolveNodeType(VirtualNode) VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeExtensions_ResolveNodeType_AkiraNetwork_VirtualStorageLibrary_VirtualNode_) VirtualItem VirtualTextFormatter.GenerateSingleTableDebugText<T>(T) (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils_S_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_) (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T> 1.html) VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter | tuple | The tuple to convert. |

| Type | Description |
|---|-------------|
| (VirtualDirectory VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualException IVirtualDeepCloneable<T> (AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html) (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualItem VirtualNodeExtensions.ResolveNodeType(VirtualNode) VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeExtensions_ResolveNodeType_AkiraNetwork_VirtualStorageLibrary_VirtualNode_) VirtualItem VirtualTextFormatter.GenerateSingleTableDebugText<T>(T) (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils_S_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_) (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T> 1.html) VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter | |

Implements

| Extension Methods |
|-------------------|
|-------------------|

| | | |
|------------------|---|--|
| VirtualException | IVirtualDeepCloneable<T> (AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html) (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualItem VirtualNodeExtensions.ResolveNodeType(VirtualNode) VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeExtensions_ResolveNodeType_AkiraNetwork_VirtualStorageLibrary_VirtualNode_) VirtualItem VirtualTextFormatter.GenerateSingleTableDebugText<T>(T) (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils_S_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_) (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T> 1.html) VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter | |
|------------------|---|--|

Class VirtualSymbolicLinkAdapter<T>

Inheritance

↳ [Object](#) (<https://learn.microsoft.com/dotnet/api/system.object>)
↳ [VirtualSymbolicLinkAdapter<T>](#)
 (AkiraNetwork.VirtualStorageLibrary)

Inherited Members

NotifyNodeDelegate

object.Equals(object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object)))
PatternMatch
(AkiraNetwork.VirtualStorageLibrary)
object.Equals(object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object)))
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary)
object.GetHashCode() (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)
object.GetType() (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)
object.MemberwiseClone()
(AkiraNetwork.VirtualStorageLibrary)
VirtualDirectoryAdapter<T>
object.ReferenceEquals(object1, object2) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)
object.ToString() (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)
1.html)

Namespace: AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)
VirtualException

Assembly: VirtualStorageLibrary.dll

Syntax: VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)
public class VirtualSymbolicLinkAdapter<T>
 VirtualID
(AkiraNetwork.VirtualStorageLibrary)

Type Parameters

VirtualItem

| Name | Description |
|------|---|
| T | (AkiraNetwork.VirtualStorageLibrary) 1.html) |

VirtualItem<T>

Constructors

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualNode

VirtualSymbolicLinkAdapter(VirtualStorage<T>)

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

public VirtualSymbolicLinkAdapter(VirtualStorage<T> storage)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNameFilter (AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html)<T>

storage

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Name | Description |
|---|---------|-------------|
| VirtualNodeNameFilter (AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html)<T> | storage | |

Properties

this[VirtualPath, bool]

Declaration

```
public VirtualSymbolicLink this[VirtualPath path, bool followLinks = true] { get; set; }
```

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

Parameters

(AkiraNetwork.VirtualStorageLibrary)

Type

PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

VirtualPath

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

bool

(<https://learn.microsoft.com/dotnet/api/system.boolean>)

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

Property Value

VirtualDirectoryAdapter<T>

Type

AkiraNetwork.VirtualStorageLibrary

1.html)

VirtualSymbolicLink

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

VirtualText

([VirtualTextFormatter.GenerateSingleTableDebugText<T>\(T\)](#))

VirtualID

(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilsilie

s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Extension Methods

(AkiraNetwork.VirtualStorageLibrary)

2.html)

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)

VirtualID

(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilsilie

s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Namespace AkiraNetwork.VirtualStorage Library.Utilities

▼ Filter by title

Classes

+ **AkiraNetwork.VirtualStorage**

Library

VirtualTextFormatter

 (AkiraNetwork.VirtualStorage)

 (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html)

- **AkiraNetwork.VirtualStorage**

Library.

Utilities

 (AkiraNetwork.VirtualStorage)

 VirtualTextFormatter

 (AkiraNetwork.VirtualStorageLibrary)

+ **AkiraNetwork.VirtualStorage**

Library.

WildcardMatchers

 (AkiraNetwork.VirtualStorage)

Class VirtualTextFormatter

Inheritance

↳ [Object](#) (<https://learn.microsoft.com/dotnet/api/system.object>)
↳ [VirtualTextFormatter](#)

+ [AkiraNetwork.VirtualStorage](#)

Inherited Members

Library
[object.Equals\(object\)](#) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-object)))
(AkiraNetwork.VirtualStorage)

[object.Equals\(object, object\)](#) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object-system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object-system-object)))
AkiraNetwork.VirtualStorage

Library.Utilities
[object.GetHashCode\(\)](#) (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)
(AkiraNetwork.VirtualStorage)

[object.GetType\(\)](#) (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)

[object.MemberwiseClone\(\)](#) (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)

[object.ReferenceEquals\(object, object\)](#) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)

[object.ToString\(\)](#) (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)

+ [AkiraNetwork.VirtualStorage](#)

Namespace: [AkiraNetwork](#) ([AkiraNetwork.html](#)).VirtualStorageLibrary ([AkiraNetwork.VirtualStorageLibrary.html](#)).Utilities
([AkiraNetwork.VirtualStorageLibrary.Utilities.html](#))

(AkiraNetwork.VirtualStorage)

Assembly: [VirtualStorageLibrary.dll](#)

Syntax

```
public static class VirtualTextFormatter
```

Methods

GenerateLinkTableDebugText<T>(VirtualStorage<T>)

Declaration

```
public static string GenerateLinkTableDebugText<T>(this VirtualStorage<T> vs)
```

Parameters

| Type | Name | Description |
|--|------|-------------|
| VirtualStorage (AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html)<T> | vs | |

Returns

| Type | Description |
|--|-------------|
| string (https://learn.microsoft.com/dotnet/api/system.string) | |

Type Parameters

| Name | Description |
|------|-------------|
| T | |

GenerateSingleTableDebugText<T>(T)

Declaration

+ AkiraNetwork.VirtualStorage

Library static string GenerateSingleTableDebugText<T>(this T singleObject)

(AkiraNetwork.VirtualStorage)

Parameters

AkiraNetwork.VirtualStorage

| Type | Utilities | Name | Description |
|------|-----------|-------------------------------|-------------|
| T | | (AkiraNetwork.VirtualStorage) | |

VirtualTextFormatter

(AkiraNetwork.VirtualStorageLibrary)

Returns

+ AkiraNetwork.VirtualStorage

| Type | WildcardMatchers | Description |
|------|--|-------------|
| | string (https://learn.microsoft.com/dotnet/api/system.string) | |

(AkiraNetwork.VirtualStorage)

Type Parameters

| Name | Description |
|------|-------------|
| T | |

GenerateTableDebugText<T>(IEnumerable<T>)

Declaration

```
public static string GenerateTableDebugText<T>(this IEnumerable<T> enumerableObject)
```

Parameters

| Type | Name | Description |
|----------------|------------------|-------------|
| IEnumerable<T> | enumerableObject | |

Returns

| Type | Description |
|--|-------------|
| string (https://learn.microsoft.com/dotnet/api/system.string) | |

Type Parameters

| Name | Description |
|------|-------------|
| T | |

GenerateTreeDebugText<T>(VirtualStorage<T>, VirtualPath, bool, bool)

Declaration

+ AkiraNetwork.VirtualStorage

```
Library static string GenerateTreeDebugText<T>(this VirtualStorage<T> vs, VirtualPath basePath
    bool recursive = true, bool followLinks = false)
(AkiraNetwork.VirtualStorage)
```

- AkiraNetwork.VirtualStorage

Parameters

Library.Utilities

Type

AkiraNetwork.VirtualStorage

VirtualStorage (AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html) <T>

(AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html)

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

+ AkiraNetwork.VirtualStorage

bool (<https://learn.microsoft.com/dotnet/api/system.boolean>)

Library.WildcardMatchers

bool (<https://learn.microsoft.com/dotnet/api/system.boolean>)

Type

AkiraNetwork.VirtualStorage

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

Name

Description

Returns

| Type | Description |
|--|-------------|
| string (https://learn.microsoft.com/dotnet/api/system.string) | |

Type Parameters

| Name | Description |
|------|-------------|
| T | |

Namespace AkiraNetwork.VirtualStorage Library.WildcardMatchers

 Filter by title

Classes

+ AkiraNetwork.VirtualStorage

Library Default WildcardMatcher

DefaultWildcardMatcher (AkkaNetwork.VirtualStorage) AkkaNetwork.VirtualStorage tml)

tmp + AkiraNetwork.VirtualStorage

Library.Utilities

Powershell Network Virtual Storage Manager

AkiraNetwork-VirtualStorage

- AkiraNetwork.VirtualStorage.htm

Library.WildcardMatchers

(AkiraNetwork.VirtualStorage)

DefaultWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary)

PowerShellWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary)

Class DefaultWildcardMatcher

Inheritance

↳ [Object](#) (https://learn.microsoft.com/dotnet/api/system.object)
↳ [DefaultWildcardMatcher](#)

+ [AkiraNetwork.VirtualStorage](#)

Implements

IVirtualWildcardMatcher ([AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher.html](#))

- [\(AkiraNetwork.VirtualStorage\)](#)

Inherited Members

+ [AkiraNetwork.VirtualStorage](#)

object.Equals(object) (https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object))

- [\(AkiraNetwork.VirtualStorage\)](#)

object.GetHashCode() (https://learn.microsoft.com/dotnet/api/system.object.gethashcode)
object.GetType() (https://learn.microsoft.com/dotnet/api/system.object.gettype)

object.MemberwiseClone() (https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone)

object.ReferenceEquals(object, object) (https://learn.microsoft.com/dotnet/api/system.object.referenceequals)

object.ToString() (https://learn.microsoft.com/dotnet/api/system.object.tostring)

Namespace: [AkiraNetwork](#).[VirtualStorage](#).[Library](#).[WildcardMatchers](#)

Namespace: [AkiraNetwork](#).[VirtualStorage](#).[Library](#).[WildcardMatchers](#) ([AkiraNetwork.VirtualStorageLibrary.WildcardMatchers.html](#))

Assembly: [VirtualStorageLibrary.dll](#)

Syntax

```
public class DefaultWildcardMatcher : IVirtualWildcardMatcher
```

Properties

Count

Gets the number of wildcards.

Declaration

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

Property Value

| Type | Description |
|--|--|
| int (https://learn.microsoft.com/dotnet/api/system.int32) | An integer representing the number of wildcards available. |

Patterns

Gets a collection of available patterns.

Declaration

```
public IEnumerable<string> Patterns { get; }
```

Property Value

| Type | Description |
|--|--|
| (AkiraNetwork.VirtualStorage) IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) (AkiraNetwork.VirtualStorage) Library.Utilities <string (https://learn.microsoft.com/dotnet/api/system.string)> (AkiraNetwork.VirtualStorage) | An enumerable collection of strings representing the available patterns. |

- AkiraNetwork.VirtualStorage

Library.WildcardMatchers

WildcardDictionary

(AkiraNetwork.VirtualStorage)

Gets a dictionary of wildcards and their corresponding regex patterns.

DefaultWildcardMatcher

Declaration

PowerShellWildcardMatcher

(AkiraNetwork.VirtualStorage)

```
public IReadOnlyDictionary<string, string> WildcardDictionary { get; }
```

Property Value

| Type | Description |
|---|---|
| ReadOnlyDictionary (https://learn.microsoft.com/dotnet/api/system.collections.objectmodel,readonlydictionary-2) <string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)> | A read-only dictionary containing wildcards as keys and their corresponding regex patterns as values. |

Wildcards

Gets a collection of available wildcards.

Declaration

```
public IEnumerable<string> Wildcards { get; }
```

Property Value

| Type | Description |
|--|---|
| IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) <string (https://learn.microsoft.com/dotnet/api/system.string)> | An enumerable collection of strings representing the available wildcards. |

+ AkiraNetwork.VirtualStorage

Methods

(AkiraNetwork.VirtualStorage)

PatternMatcher(string, string)

Library.Utilities

Determines whether the specified node name matches the specified pattern.

(AkiraNetwork.VirtualStorage)

Declaration

- AkiraNetwork.VirtualStorage

Library.WildcardMatchers

```
public bool PatternMatcher(string nodeName, string pattern)
```

(AkiraNetwork.VirtualStorage)

DefaultWildcardMatcher

Parameters

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|--|----------|---------------------------------|
| PowerShellWildcardMatcher (AkiraNetwork.VirtualStorageLibrary) | nodeName | The node name to check. |
| string (https://learn.microsoft.com/dotnet/api/system.string) | pattern | The pattern to compare against. |

Returns

| Type | Description |
|---|--|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | True if the node name matches the pattern; otherwise, false. |

Implements

IVirtualWildcardMatcher (AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher.html)

Extension Methods

VirtualTextFormatter.GenerateSingleTableDebugText < T > (T)

(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

Class PowerShellWildcardMatcher

Inheritance

↳ [Object](#) (<https://learn.microsoft.com/dotnet/api/system.object>)
↳ [PowerShellWildcardMatcher](#)

+ [AkiraNetwork.VirtualStorage](#)

Implements

[IVirtualWildcardMatcher](#) ([AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher.html](https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher.html))

- [\(AkiraNetwork.VirtualStorage\)](#)

Inherited Members

+ [AkiraNetwork.VirtualStorage](#)

[object.Equals\(object\)](#) ([<https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object\)>](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object)))

- [\(AkiraNetwork.VirtualStorage\)](#)

[object.GetHashCode\(\)](#) ([<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>](https://learn.microsoft.com/dotnet/api/system.object.gethashcode))

[object.GetType\(\)](#) ([<https://learn.microsoft.com/dotnet/api/system.object.gettype>](https://learn.microsoft.com/dotnet/api/system.object.gettype))

[object.MemberwiseClone\(\)](#) ([<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>](https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone))

[object.ReferenceEquals\(object, object\)](#) ([<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>](https://learn.microsoft.com/dotnet/api/system.object.referenceequals))

[object.ToString\(\)](#) ([<https://learn.microsoft.com/dotnet/api/system.object.tostring>](https://learn.microsoft.com/dotnet/api/system.object.tostring))

[PowerShellWildcardMatcher](#)
Namespace: [AkiraNetwork](#) ([AkiraNetwork.WildcardMatchers.html](https://learn.microsoft.com/dotnet/api/AkiraNetwork.WildcardMatchers)).[VirtualStorageLibrary](#) ([AkiraNetwork.VirtualStorageLibrary.html](https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary)).[WildcardMatchers](#) ([AkiraNetwork.VirtualStorageLibrary.WildcardMatchers.html](https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.WildcardMatchers))

Assembly: [VirtualStorageLibrary.dll](#)

Syntax

```
public class PowerShellWildcardMatcher : IVirtualWildcardMatcher
```

Properties

Count

Gets the number of wildcards.

Declaration

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

Property Value

| Type | Description |
|--|--|
| int (https://learn.microsoft.com/dotnet/api/system.int32) | An integer representing the number of wildcards available. |

Patterns

Gets a collection of available patterns.

Declaration

```
public IEnumerable<string> Patterns { get; }
```

Property Value

| Type | Description |
|--|--|
| (AkiraNetwork.VirtualStorage) IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) (AkiraNetwork.VirtualStorage) Library.Utilities <string (https://learn.microsoft.com/dotnet/api/system.string)> (AkiraNetwork.VirtualStorage) | An enumerable collection of strings representing the available patterns. |

- AkiraNetwork.VirtualStorage

Library.WildcardMatchers

WildcardDictionary

(AkiraNetwork.VirtualStorage)

Gets a dictionary of wildcards and their corresponding regex patterns.

DefaultWildcardMatcher

Declaration

PowerShellWildcardMatcher

(AkiraNetwork.VirtualStorage)

```
public IReadOnlyDictionary<string, string> WildcardDictionary { get; }
```

Property Value

| Type | Description |
|---|---|
| ReadOnlyDictionary (https://learn.microsoft.com/dotnet/api/system.collections.objectmodel,readonlydictionary-2) <string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)> | A read-only dictionary containing wildcards as keys and their corresponding regex patterns as values. |

Wildcards

Gets a collection of available wildcards.

Declaration

```
public IEnumerable<string> Wildcards { get; }
```

Property Value

| Type | Description |
|--|---|
| IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) <string (https://learn.microsoft.com/dotnet/api/system.string)> | An enumerable collection of strings representing the available wildcards. |

+ AkiraNetwork.VirtualStorage

Methods

(AkiraNetwork.VirtualStorage)

PatternMatcher(string, string)

Library.Utilities

Determines whether the specified node name matches the specified pattern.

(AkiraNetwork.VirtualStorage)

Declaration

- AkiraNetwork.VirtualStorage

Library.WildcardMatchers

```
public bool PatternMatcher(string nodeName, string pattern)
```

(AkiraNetwork.VirtualStorage)

DefaultWildcardMatcher

Parameters

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|---|----------|---------------------------------|
| PowerShellWildcardMatcher (AkiraNetwork.VirtualStorageLibrary) string (https://learn.microsoft.com/dotnet/api/system.string) | nodeName | The node name to check. |
| string (https://learn.microsoft.com/dotnet/api/system.string) | pattern | The pattern to compare against. |

Returns

| Type | Description |
|---|--|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | True if the node name matches the pattern; otherwise, false. |

Implements

IVirtualWildcardMatcher (AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher.html)

Extension Methods

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)

(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)