

Namespace AkiraNetwork.VirtualStorage Library

▼ Filter by title

Classes

- **AkiraNetwork.VirtualStorage**

Library

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html)

ActionNodeDelegate in virtual symbolic links. Determines if a given link is part of a cycle.

(AkiraNetwork.VirtualStorageLibrary)

IVirtualDeepCloneable<T>

VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html)

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

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

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

NotifyNodeDelegate

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

(AkiraNetwork.VirtualStorageLibrary)

PatternMatch

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

Additionally, nodes can be enumerated according to the specified display conditions.

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

VirtualDirectoryAdapter<T>

1.html) (AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter-1.html)

VirtualException

Adapter class to simplify operations on virtual directories. Reduces the need for casting and simplifies code,

(AkiraNetwork.VirtualStorageLibrary)

allowing users to easily manage specific node types.

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualException.html)

VirtualID

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

VirtualGroupCondition<T, TKey>

1.html) (AkiraNetwork.VirtualStorageLibrary)

(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).

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)

VirtualNode

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

(AkiraNetwork.VirtualStorageLibrary)

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

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html).

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

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

Adapter class to simplify operations on virtual items. Reduces the need for casting and simplifies code, allowing users to easily manage specific node types.

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

- **AkiraNetwork.VirtualStorage**
- Manages item data based on the user-defined type T, facilitating the management and manipulation of various library content.

(AkiraNetwork.VirtualStorage)

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

(AkiraNetwork.VirtualStorageLibrary
1.html)

VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

Represents a virtual storage node.

NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrary
1.html)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html)

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.

VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary
1.html)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html)

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

VirtualException
(AkiraNetwork.VirtualStorageLibrary
1.html)

VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

Represents the name of a virtual node. This class handles the generation, validation, and comparison of node names.

VirtualID
(AkiraNetwork.VirtualStorageLibrary
1.html)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html)

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

(AkiraNetwork.VirtualStorageLibrary
1.html)

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

Represents a path within the virtual storage. Provides functionality for manipulating, comparing, normalizing, and splitting paths.

VirtualNode
(AkiraNetwork.VirtualStorageLibrary
1.html)

VirtualSortCondition<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualSortCondition-1.html)

VirtualNodeExtensions

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

VirtualStorageExtensions

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageExtensions.html)

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

- **AkiraNetwork.VirtualStorageVirtualStorageSettings**

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageSettings.html)

(AkiraNetwork.VirtualStorage)

Manages settings for the virtual storage. This class implements the singleton pattern and holds default settings and various parameters.

(AkiraNetwork.VirtualStorageLibrary)

IVirtualDeepCloneable<T>

VirtualStorageState **(AkiraNetwork.VirtualStorageLibrary.VirtualStorageState.html)**

1.html)

Manages the state of the virtual storage. This class implements the singleton pattern and maintains the current settings and state.

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

VirtualStorage<T> **(AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html)**

PatternMatch

Manages virtual storage containing user-defined type T item data.

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

VirtualSymbolicLink

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)

(AkiraNetwork.VirtualStorageLibrary)

Represents a virtual symbolic link. 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.

1.html)

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

GroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

2.html)

VirtualSymbolicLinkAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLinkAdapter-1.html)

(Virtual)

(AkiraNetwork.VirtualStorageLibrary)

Adapteekers to simplify operations on virtual symbolic links. Reduces the need for casting and simplifies code, allowing Network Virtual Storage specific node types.

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

Structs

1.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

VirtualID **(AkiraNetwork.VirtualStorageLibrary.VirtualID.html)**

1.html)

A structure representing a unique identifier within the virtual storage library. This identifier is based on a

VirtualNode

randomly generated UUID version 4 (UUIDv4). UUIDv4 is a 128-bit value that ensures high uniqueness and has

(AkiraNetwork.VirtualStorageLibrary)

strong cryptographic properties.

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html)

Represents the conditions for creating a list of virtual nodes.

Interfaces



IVirtualDeepCloneable<T>

(AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html)

Library

Provides functionality to create deep clones of type T.

(AkiraNetwork.VirtualStorage)

ActionNodeDelegate

IVirtualWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher.html)

Provides functionality for wildcard matching.

1.html

IVirtualWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher.html)

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary.PatternMatch)

Specifies the types of virtual nodes.

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory)

VirtualNodeTypeFilter

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html)

Specifies the filters for virtual node types.

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter)

VirtualFile

(AkiraNetwork.VirtualStorageLibrary)

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeCondition.html)

2.html

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

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeDelegate

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeDelegate.html)

VirtualItemAdapter<T>

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

(AkiraNetwork.VirtualStorageLibrary)

1.html

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary.PatternMatch.html)

(AkiraNetwork.VirtualStorageLibrary)

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

1.html

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions



- **AkiraNetwork.VirtualStorageLibrary**
(AkiraNetwork.VirtualStorageL

ActionNodeDelegate
(AkiraNetwork.VirtualStorageLibrary)
IVirtualDeepCloneable<T>
(AkiraNetwork.VirtualStorageLibrary)
1.html)
IVirtualWildcardMatcher
(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

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)
(AkiraNetwork.VirtualStorageLibrary)
```

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) | |
| 1.html | |
| VirtualItem<T> | |
| (AkiraNetwork.VirtualStorageLibrary) | |
| 1.html | |
| VirtualNode | |
| (AkiraNetwork.VirtualStorageLibrary) | |
| VirtualNodeContext | |
| (AkiraNetwork.VirtualStorageLibrary) | |
| 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

(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.VirtualStorage)

Syntax

```
  ActionNodeDelegate  
  (AkiraNetwork.VirtualStorageLibrary  
  public interface IVirtualWildcardMatcher  
  IVirtualDeepCloneable</>
```

(AkiraNetwork.VirtualStorageLibrary 1.html)

|VirtualWildcardMatcher

Properties

(AkiraNetwork.VirtualStorageLibrary.PatternMatch

unt
(AkiraNetwork.VirtualStorageLibrary)

Gets ~~Virus~~

Gets VirtualDiskDefactor cards.
(AkiraNetwork VirtualStorage library)

Declaration

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary
in VirtualDiskFactoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary

(AKIaNETWORK.VirtualStorageLibrary
1.html)
Property Value

Type (AkiraNetwork.VirtualStorageLibrary)

| 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. |

2.HTML

VirtualDisk
(AkiraNetwork.VirtualStorageLibrary)

Patterns

Item

Gets a collection of available patterns.
VirtualItemAdapter<T>
(AlikeNetwork.VirtualStorage library)

Declaration 1.html)

```
VirtualItem<T>
IEnumerable<string> Patterns {
```

1.html)
Property Values -

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

VirtualDiskListConditions

| 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

DeepCloneable<T>
(AkiraNetwork.VirtualStorageLibrary)
1.html
ReadOnlyDictionary<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) <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

(AkiraNetwork.VirtualStorageLibrary)
2.html
IEnumerable<string> Wildcards { get; }

VirtualItem

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

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

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

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

1.html

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

| Type | Description |
|-----------------------|---|
| VirtualItemAdapter<T> | 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 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)
1.html)
```

IVirtualWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary)

Parameters

NotifyNodeDelegate

Type (AkiraNetwork.VirtualStorageLibrary)

Name

Description

PatternMatch

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

nodeName

The name of the node to be matched.

VirtualCycleDetector

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

pattern

The pattern to match against.

VirtualDirectory

Return VirtualDirectoryAdapter<T>

Type (AkiraNetwork.VirtualStorageLibrary)
1.html)

Description

bool VirtualException

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

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).Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils

s_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

Class for detecting cycles in virtual symbolic links. Determines if a given link is part of a cycle.

Filter by title

Inheritance

(AkiraNetwork.VirtualStorageLibrary)
↳ object (<https://learn.microsoft.com/dotnet/api/system.object>)
 ↳ VirtualDeepCloneable<T>
 ↳ VirtualCycleDetector
 (AkiraNetwork.VirtualStorageLibrary)

Inherited Members

IVirtualWildcardMatcher
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)
 NotifyNodeDelegate
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)
 PatternMatch
object.GetHashCode() (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)
 (AkiraNetwork.VirtualStorageLibrary)
object.GetType() (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)
 VirtualCycleDetector
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>)
 VirtualDirectory
object.ToString() (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)
 (AkiraNetwork.VirtualStorageLibrary)
Namespace: AkiraNetwork (AkiraNetwork.html).VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)

Assembly: AkiraNetwork.dll (AkiraNetwork.dll)

Syntax

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

Constructors

(AkiraNetwork.VirtualStorageLibrary)
VirtualItem
(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector()
(AkiraNetwork.VirtualStorageLibrary)
VirtualItemAdapter<T>
Initializes a new instance of the VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary)
(AkiraNetwork.VirtualStorageLibrary.VIRTUALCYCLEDETECTOR.html) class.
1.html)

VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary)
1.html)
public VirtualCycleDetector()
VirtualNode
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary)

Count
(AkiraNetwork.VirtualStorageLibrary)
VirtualnodeName
Gets the number of detected cycles
(AkiraNetwork.VirtualStorageLibrary)

IsNodeInCycle(VirtualSymbolicLink)

Determines whether the specified VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) is part of a cycle.

Declaration

```
    public bool IsNodeInCycle(VirtualSymbolicLink link)
```

Virtual Parameters

| Type | Name | Description |
|---|------|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualWildcardMatcher.html) | | |
| VirtualSymbolicLink | link | The VirtualSymbolicLink |
| (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLinkDelegate.html) | | (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) |
| (AkiraNetwork.VirtualStorageLibrary.PatternMatch.html) | | to check. |

Virt Returns

| Type | Description |
|--|--|
| <code>AkiraNetwork.VirtualStorageLibrary.VirtualDirectory</code> | True if the link is part of a cycle; otherwise, false. |

Extension Methods

(AkiraNetwork_VirtualStorageLibrary_VirtualTextFormatter_GenerateSingleTableDebugText<T>(T)
(AkiraNetwork_VirtualStorageLibrary_Utility_VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1_0_)
2.html)

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualIt

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

Virtua|N

(AkiraNetwork.Virtual)

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)

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

↳ VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.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.VirtualNode.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>

VirtualNodeVID

(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))

VirtualNodeHashCode

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

VirtualNodeType

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

VirtualNodeMemberwiseClone

(https://learn.microsoft.com/dotnet/api/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>>
    (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html)
    1.html)
```

Constructors

VirtualNodeMatcher

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeMatcher.html)

NotifyNodeDelegate

VirtualDirectory()

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.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.VirtualDirectory.html)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter.html)

1.html)

VirtualException

(AkiraNetwork.VirtualStorageLibrary.VirtualException.html)

VirtualDirectory(VirtualDirectoryName)

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryName.html)

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

VirtualName

Declaration

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)

```
public VirtualDirectory(VirtualDirectoryName name)
    VirtualItemAdapter<T>
```

(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter.html)

Parameters

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

VirtualDirectory(VirtualDirectoryName, DateTime, DateTime)

VirtualNodeExtensions

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

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html)

Declaration

VirtualDirectoryName

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryName.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.

VirtualDirectoryAdapter<T>

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

public int Count { get; }

VirtualException

(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey>.Count.html)

Property Value

(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey>.Count.html)

Type

Description

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

(AkiraNetwork.VirtualStorageLibrary.VirtualItem.Adapter<T>.Count.html)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.VirtualItem.Adapter<T>.Count.html)

DirectoryCount

VirtualItemAdapter<T>

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

Gets the number of directories within the directory.

VirtualItem<T>

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

public int DirectoryCount { get; }

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.Adapter<T>.Count.html)

Property Value

VirtualNodeContext

Description

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.Context.html)

VirtualNodeExtensions

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

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.Extensions.html)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.ListConditions.html)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.Name.html)

VirtualNodeNotFoundException

DirectoryViewCount

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.ViewCount.html)

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

Declaration

```
public int DirectoryViewCount { get; }
```



Property Value

| Type | Description |
|---|-------------|
| IVirtualDeepCloneable<T> (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | |

IVirtualWildcardMatcher
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

this[VirtualNodeName]

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

Gets or sets the node with the specified name using the indexer.

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

Declaration
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

public VirtualNode this[VirtualNodeName name] { get; set; }
VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

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

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

VirtualException
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

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

VirtualItem
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

Exceptions
VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html)

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

VirtualNode
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

ItemCount
Gets the number of items within the directory.

VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html)

Declaration
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html)

VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html)

VirtualStorageLibrary
(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html)

Property Value
VirtualNodeName
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html)

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

Nodes

Enumerates all nodes within the directory.

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

Property Value

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

NodesView

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

Declaration

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

Property Value

| Type | Description |
|---|-------------|
| VirtualItem IEnumerable<VirtualItem> (AkiraNetwork.VirtualStorageLibrary VirtualNode.html) | |

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

NodesViewCount

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

Declaration

VirtualNode
(AkiraNetwork.VirtualStorageLibrary
VirtualNodeContext
public int NodesViewCount { get; }
(AkiraNetwork.VirtualStorageLibrary
VirtualNodeExtensions

Property Value

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

| 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)
```

Parameters
`VirtualItem`

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

Returns
`VirtualNodeContext`
([AkiraNetwork.VirtualStorageLibrary](#))

| 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>VirtualStorageLibrary</code> | |
| <code>T</code> <code>2.html</code> | The data type of the item. |

VirtualID

`AkiraNetwork.VirtualStorageLibrary`

VirtualItem

`AkiraNetwork.VirtualStorageLibrary`

AddSymbolicLink(VirtualNodeName, VirtualPath, bool)

Adds a symbolic link with the specified name.

`AkiraNetwork.VirtualStorageLibrary`

Declaration

```
public void AddSymbolicLink(VirtualNodeName name, VirtualPath targetPath, bool allowOverwrite = false)
```

VirtualNode

`AkiraNetwork.VirtualStorageLibrary`

Parameters

| VirtualNodeContext | |
|--|-------------------|
| Type <code>AkiraNetwork.VirtualStorageLibrary</code> | Name |
| <code>VirtualNodeExtensions</code> <code>VirtualNodeName</code> (<code>AkiraNetwork.VirtualStorageLibrary</code> . <code>VirtualNodeName</code>) <code>VirtualNodeListConditions</code> <code>VirtualStorageLibrary</code> | <code>name</code> |
| <code>VirtualNodeName</code> (<code>AkiraNetwork.VirtualStorageLibrary</code>) | |
| <code>VirtualNodeNotFoundException</code> | |


```
public bool DirectoryExists(VirtualNodeName name)
```

Parameters

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

Returns

| Type | Description |
|---|---|
| IVirtualWildcardMatcher (AkiraNetwork.VirtualStorageLibrary.VirtualWildcardMatcher.html) | True if the directory exists; otherwise, false. |

Get(VirtualNodeName, bool)

Gets the node with the specified name.

VirtualDirectory

Declaration

```
public VirtualItem Get(VirtualNodeName name, bool exceptionEnabled = true)
```

Parameters

| Type | Name | Description |
|--|------|-----------------------|
| VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition.html) | | |
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualID.html) | name | The name of the node. |

Returns

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

GetDirectory(VirtualNodeName)

Gets the directory with the specified name.

VirtualNodeListConditions

Declaration

VirtualDirectoryName

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()

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

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>.GetEnumerator())
```

VirtualID

VirtualItem

```
public System.Collections.IEnumerable GetEnumerator()
```

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter.html)

Returns

1.html

Type

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)

IEnumerable

1.html

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

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

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html)

GetItemByName(VirtualNodeName)

(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)

Gets the item with the specified name.

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html)

Declaration

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

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 | Name | 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 | |
| InvalidOperationException 1.html (https://learn.microsoft.com/dotnet/api/system.invalidoperationexception) | Thrown if the specified node is not an item. (AkiraNetwork.VirtualStorageLibrary) |
| 2.html | |

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) |

INullable

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. |
| VirtualOperationAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html) 1.html) | Thrown if the specified node is not a symbolic link. |

ItemExists(VirtualNodeName)

(AkiraNetwork.VirtualStorageLibrary.VirtualID)

Checks whether an item with the specified name exists.

(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>)

(AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>)

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

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions)

Parameters

| Type | Name | Description |
|---|------|--------------------------------|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) 1.html) | name | The name of the item to check. |

Returns

| Type | Description |
|---|--|
| VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.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 | | Name | Description |
|---|-------------|------|--------------------------------|
| <code>IVirtualDeepCloneable<T></code> (AkiraNetwork.VirtualStorageLibrary.html) | | | |
| <code>IVirtualWildcardMatcher</code> <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.VirtualCycleDetector |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary) | True if the node exists; otherwise, false. |

Remove Virtual Node

(AkiraNetwork.VirtualStorageLibrary

RemoveNode(*node*)

VirtualException Declaration

(AkiraNetwork VirtualStorage library)

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

2.html)

| Parameter ID | | Name | Description |
|--------------|---|------|---------------------|
| Type | (AkiraNetwork.VirtualStorageLibrary.VirtualItem) | | |
| Value | (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | 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.cs

Checks whether a symbolic link with the
(AkiraNetwork)VirtualStorageLibrary

AkiraNetwork.Virtual

Declaration NodeListConditions

(AkiraNetwork.VirtualStorageLibrary

VirtualNodeName [same as linkExists]

Subfile B881. SYMBOLIC LINKS (AkiraNetwork VirtualStorage library)

(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
```

Return Value

IVirtualDirectoryAdapter<T>

| Type | Description |
|---|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | A string representation of the directory. |

Overrides

```
VirtualGroupCondition<T, TKey>
object.ToString() (https://learn.microsoft.com/dotnet/api/system.object.tostring)
    2.html
```

VirtualID

Update(VirtualNode)

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.VirtualNode)
```

Parameters

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

Overrides

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions)

```
VirtualNode.UpdateVirtualNode
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Update_A
    AkiraNetwork_VirtualStorageLibrary_VirtualNode_)
    VirtualNodeName
```

Exceptions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException)

VirtualNodeNotFoundException

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

1

Operators

VirtuaIDeep

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

(AKHANETWOR
Battaw Matla

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

VirtualCycleDetector

VirtualCycleDetector

parameters

| Type | VirtualDirectory (AkiraNetwork.VirtualStorageLibrary) | Name | Description |
|--|--|-----------|---|
| VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary) | VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) 1.html | directory | The directory to which the node is added. |
| VirtualNode (AkiraNetwork.VirtualStorageLibrary) | VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | node | The node to add. |

VirtualGroupCondition<T, TKey>

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

implicit operator VirtualDirectory(VirtualNodeName)

1.html) Performs an implicit conversion from VirtualNodeName

(AkiryaItemToVirtualStorageLibraryVirtualNodeName.html) to VirtualDirectory.

(AkiraNetwork VirtualStorageLibrary.vnl)

raNetwork.virtualStorage
1.html)

Declaration Node

(AlciraNetwork VirtualStorage) library

Vista de la red de almacenamiento virtual (VirtualStorageLibrary) en el Administrador de dispositivos.

Implicit operator

(AkiraNetwork.virtualSt

VirtualNodeExtensions Parameters

| Parameters | (AkiraNetwork.VirtualStorageLibrary) | | |
|------------------------------|---|-----------------|---------------------------|
| Type | <code>virtualNodeListConditions</code> | Name | Description |
| | (AkiraNetwork.VirtualStorageLibrary) | | |
| <code>VirtualNodeName</code> | <code>VirtualNodeName</code> | <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> | |

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

1.html

| | |
|--|--|
| VirtualTextFormatter.GenerateSingleTableDebugText<T>(T) | |
| (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils s_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>) | |
| (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie s_VirtualTextFormatter_GenerateTableDebugText_1_System_Collections_Generic_IEnumerable__0_) | |

VirtualNodeName

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

| | |
|------------------------------|--|
| 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>

Adapter class to simplify operations on virtual directories. Reduces the need for casting and simplifies code, allowing users to easily manage specific node types.

Inheritance

1.html

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

Inherited Members

(AkiraNetwork.VirtualStorageLibrary)
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)
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)
object.GetHashCode() (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)
object.GetType() (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)
(AkiraNetwork.VirtualStorageLibrary)
object.MemberwiseClone() (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)
VirtualDirectoryAdapter<T>
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>)
1.html

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

Assembly

AkiraNetwork.VirtualStorageLibrary

Syntax

```
(AkiraNetwork.VirtualStorageLibrary)  
2.html  
public class VirtualDirectoryAdapter<T>  
    VirtualID  
(AkiraNetwork.VirtualStorageLibrary)
```

Type Parameters

| Name | Description |
|------|---|
| T | The type of item stored in the virtual storage. |

1.html

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html

VirtualNode

VirtualDirectoryAdapter(VirtualStorage<T>)

VirtualNodeContext

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

```
public VirtualDirectoryAdapter(VirtualStorage<T> storage)  
    VirtualNodeName
```

(AkiraNetwork.VirtualStorageLibrary)

Parameters

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|--|---------|-------------------------------|
| VirtualStorage (AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html) <T> | storage | The virtual storage instance. |

▼

Properties

(AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html)

this[VirtualPath, bool]

(AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html)

Gets the VirtualDirectory corresponding to the specified path.

(AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html)

Declaration

PatternMatch

(AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html)

virtual VirtualDirectory this[VirtualPath path, bool followLinks = true] { get; set; }

(AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html)

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html)

Parameters

(AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html)

Type virtualDirectoryAdapter<T>

| Type | Name | Description |
|-------------|-------------|--|
| VirtualPath | path | The path of the directory. |
| bool | followLinks | A flag indicating whether to follow links. |

Property Value

| Type | Description |
|----------------|---|
| VirtualItem<T> | The VirtualDirectory corresponding to the specified path. |

Exceptions

| Type | Condition |
|-------------|---|
| VirtualNode | Thrown when the node is not a VirtualDirectory. |

Extension Methods

VirtualNodeName.GenerateSingleTableDebugText<T>(T)

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

s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

(AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html)



(AkiraNetwork.VirtualStorageLibrary
1.html)
IVirtualWildcardMatcher
(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

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

Implements

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

Inherited Members

VirtualDirectory

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

Namespace

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

.VirtualStorageLibrary ([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

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

Parameters

1.html)

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

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

1.html)

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

VirtualNode

| Type | Name | Description |
|---|---------|--------------------|
| (AkiraNetwork.VirtualStorageLibrary) VirtualNode | message | The error message. |

Exception (AkiraNetwork.VirtualStorageLibrary)

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

Implements

ISerializable (https://learn.microsoft.com/dotnet/api/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.TKey) The type of the key used for grouping. |

Constructors

(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition)

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

VirtualNodeListConditions

Initializes a new instance of the VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition-2.html) 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) |
| VirtualNodeDelegate | GenerateSingleTableDebugText<T>(T) |
| (AkiraNetwork.VirtualStorageLibrary) | (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilities_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_) |
| (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) | |

Struct VirtualID

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

(AkiraNetwork.VirtualStorageLibrary)

Implements

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

(AkiraNetwork.VirtualStorageLibrary.VirtualID.html)

(AkiraNetwork.VirtualStorageLibrary)

Inherited Members

VirtualCycleDetector

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

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

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

1.html).

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)

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

Property Value

| Type | Description |
|---|--|
| Guid (https://learn.microsoft.com/dotnet/api/system.guid) AkiraNetworkNodeDelegate | 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

(<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition.ToString>)

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

2.html)

Returns

VirtualID

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

Description

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

A string that represents the unique identifier.

Overrides

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

VirtualItem<T>

(<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualItem>)

1.html)

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

(<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualNode>)

Extension Methods
(<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualTextFormatter.GenerateSingleTableDebugText>)
VirtualTextFormatterExtensions

(<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter>)#AkiraNetwork_VirtualStorageLibrary_Utilities_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_

(<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions>)

(<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualTextFormatter>)

VirtualNodeName

(<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualNodeName>)

VirtualNodeNotFoundException

(<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException>)

VirtualNodeType

(<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualNodeType>)

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_AkiraNetwork_VirtualStorageLibrary_VirtualNode_](#))

`VirtualNodeVirtualStorage`

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualNodeIsReferencedInStorage`

([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)))

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualNodeObjectEquals<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](#))

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

([AkiraNetwork.VirtualStorageLibrary](#))

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

([AkiraNetwork.VirtualStorageLibrary](#))

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

([AkiraNetwork.VirtualStorageLibrary](#))

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

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualNodeToString` (<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)

DateOnly (<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

Declaration

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

| Type | Name | Description |
|--------------------------------|-------------|--------------------------------|
| VirtualException | name | The name of the item. |
| VirtualGroupCondition<T, TKey> | createdDate | The creation date of the item. |
| VirtualItem<T> | | |
| VirtualNode | | |
| VirtualNodeName | | |
| VirtualNodeContext | | |
| VirtualNodeExtensions | | |
| VirtualNodeListConditions | | |
| VirtualNodeType | | |
| VirtualNodeTypeFilter | | |

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

Parameters

| Type | | Name | Description |
|---|--|--------------------|-----------------------------------|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | | <i>name</i> | The name of the item. |
| NotifyNodeDelegate DateTime (AkiraNetwork.VirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.datetime) PatternMatch | | <i>createdDate</i> | The creation date of the item. |
| DateTime (AkiraNetwork.VirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.datetime) | | <i>updatedDate</i> | The last update date of the item. |

Methods

DeepClone(bool)
(AkiraNetwork.VirtualStorageLibrary

Creates a deep copy of the item. If the data of type T held by the item implements the IVirtualDeepCloneable<T> (AkiraNetwork.VirtualStorageLibrary.VirtualDeepCloneable-1.html) interface, the DeepClone method of the data type is called.

VirtualID
Declaration:

(AkiraNetwork.VirtualStorageLibrary

VirtualItem

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

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary

Parameters

1.html)

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

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

VirtualNodeExtensions

VirtualNode
(AkiraNetwork.VirtualStorageLibrary

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions

INode.html)

(AkiraNetwork.VirtualStorageLibrary

VirtualNodeName

Overrides
(AkiraNetwork.VirtualStorageLibrary

VirtualNode.DeepClone(bool)

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_DeepClone_System_Boolean)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary

VirtualNodeTypeFilter

Description

A deep clone of the current item as a VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html).

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
PatternMatch
ExtensionsResolveNodeType_AkiraNetwork_VirtualStorageLibrary_VirtualNode_)
(AkiraNetwork.VirtualStorageLibrar
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 VirtualItemAdapter<T>

Adapter class to simplify operations on virtual items. Reduces the need for casting and simplifies code, allowing users to easily manage specific node types.

Inheritance

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

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>)
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>)
VirtualGroupCondition<T, TKey>

Namespace

AkiraNetwork.AkiraNetwork.html.VIRTUALSTORAGELIBRARY (AkiraNetwork.VirtualStorageLibrary.html)

Assembly

VIRTUALSTORAGELIBRARY.dll

Syntax

VirtualID
(AkiraNetwork.VirtualStorageLibrary)
VirtualItem
public class VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary)
VirtualItemAdapter<T>

Type

(AkiraNetwork.VirtualStorageLibrary)

| Name | Description |
|------|---|
| T | The type of item stored in the virtual storage. 1.html |

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

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

VirtualNodeListConditions
[1.html](#) class
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName
[1.html](#)

Declaration
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException
[1.html](#)

VirtualNodeAdapter(VirtualStorage<T> storage)
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType
[1.html](#)

Parameters
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter
[1.html](#)

| Type | Name | Description |
|--|----------------|-------------------------------|
| VirtualStorage (AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html) <T> | <i>storage</i> | The virtual storage instance. |

二

Properties

VirtualNetwork.VirtualStorageLibrary

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary
VirtualPath hash)

VirtualPad PatternMatch

(AkiraNetworkVirtualStorageLibrary) or sets the VirtualItem corresponding to the specified path.

VirtualCycleDetector

Declaration

Vista IP

VirtualDirectory

```
public virtual NetworkVirtualStorage[VirtualPath path, bool followLinks = true] { get; set; }
```

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

Parameters

| Type | Name | Description |
|---|--------------------------|--|
| <code>VirtualException</code> (AkiraNetwork.VirtualStorageLibrary.VirtualPath) <code>VirtualPath</code> <code>VirtualGroupCondition<T, TKey></code> (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary.VirtualPath) | <code>path</code> | The path of the item. |
| <code>bool</code> (.html) https://learn.microsoft.com/dotnet/api/system.boolean (AkiraNetwork.VirtualStorageLibrary.VirtualPath) | <code>followLinks</code> | A flag indicating whether to follow links. |

VirtualItem Property Value

(AkiraNetwork VirtualStorage Library)

| Type | Description |
|---|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>) | |
| VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>.1.html) | The VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) corresponding to the specified path. |
| <T>(AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>.1.html) | |

~~VirtualNode Exceptions~~

(AkiraNetwork VirtualStorage Library)

| Type | Condition |
|---|---|
| (AkiraNetwork.VirtualStorageLibrary.InvalidOperationException VirtualNodeExtensions (https://learn.microsoft.com/dotnet/api/system.invalidoperationexception) VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.IVirtualNodeList<T>)) | Thrown when the node is not a VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) |
| (AkiraNetwork.VirtualStorageLibrary.IVirtualNodeList<T>)) | . |

Extension Methods

(AkiraNetwork.VirtualStorage
VirtualNetworkProtocolExtension)

(Ali, Naseem & Majeed, 1999)

(AkiraNetwork Virtual Storage Library) **qualTextFormatter.GenerateSingleTab**

VirtualNodeType
raNetwork.VirtualStorageLibrary.Util

(AkiraNetwork VirtualStorage Library)
VirtualVolumeFormatter generated by Sinfonia

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 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_CreatedDate)
VirtualItemAdapter<T>
VirtualNode.UpdatedDate
VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_UpdatedDate)
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.dll

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)

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
(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

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

VirtualDirectoryAdapter<T>

Parameters
1.html)

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

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

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
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

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

VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

Parameters
1.html)

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

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>

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 | Name | Description |
|--|-------------|---|
| (AkiraNetwork.VirtualStorageLibrary) | <i>node</i> | The node containing the data to update. |

VirtualNodeType
Overrides
VirtualNodeTypeFilter

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 (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_ArgumentException) | Thrown if the specified node is not of the same type as the current item. |

Operators

VirtualCycleDetector
 (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#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)
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

Declaration

```
VirtualException  

(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey>  

public static implicit operator VirtualItem<T>(VirtualNodeName name)
```

Parameters

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

Returns

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

| Type | Description |
|---|--|
| VirtVirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) | 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.html) Converts a tuple (nodeName, itemData) to a VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html)
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

Declaration

```
VirtualNotFoundException  

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

VirtualNodeType
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html)

Parameters

| 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
IVirtualNodeMetadatable (AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-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

Implements

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

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

Constructors

VirtualNode(VirtualNodeName)

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> | 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 |
| VirtualNodeName | VirtualNodeName (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 | VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary) | updatedDate | The updated date of node. |

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

| | |
|-----------------|---|
| VirtualNodeName | VirtualNodeName (AkiraNetwork.VirtualStorageLibrary) |
|-----------------|---|

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

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

CreatedDate

| | |
|-----------------------|---|
| 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 |
|---------|-------------|
| 2.html) | |

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

Name

(AkiraNetwork.VirtualStorageLibrary)

Gets the name of node.

1.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

VirtualnodeName (<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html>)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNameNotfoundException

(AkiraNetwork.VirtualStorageLibrary)

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 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 Declaration | Directory |

UpdatedDate

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
1.html)

| Property Value |
|---|
| VirtualException VirtualNetwork.VirtualStorageLibrary VirtualGroupCondition<T, TKey> DateTime (https://learn.microsoft.com/dotnet/api/system.datetime) (AkiraNetwork.VirtualStorageLibrary 2.html) VirtualID |

VID

(AkiraNetwork.VirtualStorageLibrary
VirtualItem
Gets the VID of node
(AkiraNetwork.VirtualStorageLibrary
VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary
1.html)
public VirtualID VID { get; }
VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary
1.html)

Property Value

| Type | Description |
|---|-------------|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary VirtualDataNodeContext.VirtualStorageLibrary.VirtualID.html) (AkiraNetwork.VirtualStorageLibrary VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary VirtualnodeName (AkiraNetwork.VirtualStorageLibrary VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary Creates a deep clone of the storage library. However, the CreatedDate and UpdatedDate should not be cloned as they are set to VirtualNode date and time at the time of cloning. | VirtualID |

Methods

DeepClone(bool)

Creates a deep clone of the storage library. 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
VirtualNodeFilter

```
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) |

.htm

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

Update(VirtualNode)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

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

2.html

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

public abstract void Update(VirtualNode node)

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

Parameters

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|--|------|-----------------|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | node | Value to update |

1.html

VirtualNode

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_Utilitie

s_VirtualTextFormatter GenerateSingleTableDebugText_1__0_

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions ResolveNodeType(VirtualNode)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions VirtualNodeExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_

ExtensionsResolveNodeType_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>

Constructors

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

Declaration

(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

(AkiraNetwork.VirtualStorageLibrary)
ed = false, VirtualSymbolicLink? resolvedLink = null)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Parameters

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.VirtualDirectoryAdapter<T> VirtualPathException (AkiraNetwork.VirtualStorageLibrary.VirtualPathVirtualGroupCondition<T, TKey> VirtualPath bool VirtualID VirtualItem VirtualItemLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLinkItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem<T> VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext VirtualNodeContext 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 VirtualStorageLibrary VirtualnodeName (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName Property Value VirtualNodeNotFoundException Type (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType int (https://learn.microsoft.com/dotnet/api/system.int32) (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter | <i>traversalPath</i> | The path used for node traversal. |
| <i>parentNode</i> | | The parent directory of the node. |
| <i>depth</i> | | The depth of the node. The root is considered 0. |
| <i>index</i> | | The index of the node. Corresponds to the enumeration order within the directory. |
| <i>resolvedPath</i> | | The result of resolving a symbolic link. |
| <i>resolved</i> | | Indicates whether the symbolic link has been resolved. |
| <i>resolvedLink</i> | | The resolved symbolic link. |

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

VirtualStorageLibrary

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualnodeName

VirtualNodeNotFoundException

Type
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType

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

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter

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)
```

```
public bool Resolved { get; }
```

Property Value

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

ResolvedLink

Gets the resolved symbolic link and can only be set within the assembly.

VirtualCycleDetector

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.ResolvedLink { get; })
```

VirtualDirectoryAdapter<T>

Property Value

1.html)

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

| | |
|---|--------------------------------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink VirtualGroupCondition<T, TKey>) | The resolved symbolic link, or null. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) | |

2.html)

VirtualID

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.ResolvedPath { get; })
```

VirtualItemAdapter<T>

Property Value

1.html)

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

Gets the resolved path and can only be set within the assembly. This represents the result of resolving a symbolic link.

VirtualItem

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.ResolvedPath { get; })
```

1.html)

VirtualNode

Property Value

VirtualPath

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

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

The resolved path, or null.

VirtualNodeExtensions

VirtualNodeListConditions

TraversalPath

VirtualNodeName

Gets the traversal path and can only be set within the assembly.

VirtualNodeType

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.TraversalPath { get; })
```

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter)

VirtualNodeType

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)

VirtualTextFormatterAdapter.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

(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 Type Equals(object) (https://learn.microsoft.com/dotnet/api/system.valuetype.equals)
Value Type GetHashCode() (https://learn.microsoft.com/dotnet/api/system.valuetype.gethashcode)
Value Type 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))
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)
Namespaces (AkiraNetwork.VirtualStorageLibrary.html).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)
VirtualNodeListConditions()
(AkiraNetwork.VirtualStorageLibrary)
Initializes a new instance of the VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary).VirtualNodeListConditions.html class with default conditions.
1.html)

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

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

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

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

Represents the name of a virtual node. This class handles the generation, validation, and comparison of node names.
Filter by title

Inheritance
(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

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

↳ VirtualNodeName

PatternMatch

Implementation
(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

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

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) >

VirtualDirectory

IComparable (<https://learn.microsoft.com/dotnet/api/system.icomparable-1>)<VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) >

VirtualDirectoryAdapter<T>

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

(AkiraNetwork.VirtualStorageLibrary)

Inherited Members
[1.html](#)

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

(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>)

VirtualID

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

(AkiraNetwork.VirtualStorageLibrary)

Assembly: VirtualStorageLibrary.dll

VirtualItem

Syntax

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

public class VirtualNodeName : IComparable<VirtualNodeName>, IComparable<VirtualNodeName>, ICmpable

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

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

class ([AkiraNetwork.VirtualStorageLibrary](#))

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

public class VirtualNodeName (string name)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

Parameters

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

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

Properties

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary)

PatternMatch

Indicates whether the node is the root node.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

[VirtualCycleDetector](#)

(AkiraNetwork.VirtualStorageLibrary)

`public bool IsRoot { get; }`

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

Property Value

(AkiraNetwork.VirtualStorageLibrary)

Type: [html](#)

Description

VirtualException

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

Name: [html](#)

VirtualID

Gets the name of the node.

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

`public string Name { get; }`

(AkiraNetwork.VirtualStorageLibrary)

[1.html](#)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

Type: [T.html](#)

Description

VirtualNode

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

CompareTo(VirtualnodeName?)

(AkiraNetwork.VirtualStorageLibrary)

Compares this node to the specified VirtualnodeName.

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

(AkiraNetwork.VirtualStorageLibrary)

Declaration

[VirtualNodeType](#)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

```
public int CompareTo(VirtualNodeName? other)
```

Parameters

| Type | Name | Description |
|---|-------|--|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) Delegate (AkiraNetwork.VirtualStorageLibrary.PatternMatch PatternMatch (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | other | The VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) to compare with. |

Returns

| Type | Description |
|--|--|
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) | A value indicating the relative order of the instances being compared. |

CompareTo(object?)

Compares this instance with a specified object.
(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey>
2.html)

Declaration

```
VirtualItem CompareTo(object? obj)
    (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>
VirtualItemAdapter<T>
```

Parameters

| Type | Name | Description |
|--|------|-----------------------------|
| VirtualItem<T> object (https://learn.microsoft.com/dotnet/api/system.object) (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) | obj | The object to compare with. |

Return Value

| Type | Description |
|---|--|
| VirtualNodeContext int (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html) VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html) | A value indicating the relative order of the instances being compared. |

Exceptions

| Type | Condition |
|--|---|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) VirtualNodeNotFoundException (https://learn.microsoft.com/dotnet/api/system.argumentexception) argVirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNameFilter) VirtualNodeTypeFilter | Thrown when the object is not a VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html). |

Equals(VirtualNodeName?)

Determines whether the specified VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) is equal to this instance.

Declaration

```
public bool Equals(VirtualNodeName? other)  
(AkiraNetwork.VirtualStorageLibrary.  
NotifyNodeDelegate
```

Parameters

| Type | Name | Description |
|---|-------|--|
| atternMatch (AkiraNetwork.VirtualStorageLibrary. VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) VirtualDirectoryName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | other | The VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) to compare with this instance. |

Returns

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

| Type | Description |
|--|---|
| VirtualException bool (AkiraNetwork.VirtualStorageLibrary. (https://learn.microsoft.com/dotnet/api/system.boolean) VirtualGroupCondition<T, TKey> net/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) 2.html) | true if the specified VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) is equal to this instance; otherwise, false . |

VirtualID

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

Equals(object?)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

Determines whether the specified object is equal to this instance.

Declaration

```
VirtualItem<T>  
public override bool Equals(object? obj)  
(AkiraNetwork.VirtualStorageLibrary.  
1.html)
```

Parameters

| Type | Name | Description |
|---|---|---|
| VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | obj (https://learn.microsoft.com/dotnet/api/system.object) | The object to compare with this instance. |

Returns

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

| Type | Description |
|---|--|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | true if the specified object is equal to this instance; otherwise, false . |

Overrides

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

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)

Generates a new node name using the specified prefix.

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    NotifyNodeDelegate
public static VirtualNodeName GenerateNodeName(string prefix)
(AkiraNetwork.VirtualStorageLibrary)
    PatternMatch
```

Parameters

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

| | | |
|--|--------|-------------------------------|
| string (https://learn.microsoft.com/dotnet/api/system.string) | prefix | The prefix for the node name. |
|--|--------|-------------------------------|

Returns

VirtualDirectoryName
(AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|--|--------------------------|
| VirtualException VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | The generated node name. |

Remarks

(AkiraNetwork.VirtualStorageLibrary)
The current implementation may be improved for better usability, including a possible re-implementation of the node name generation. Users are advised to check for updates in the documentation regularly.

Exceptions

| Type | Condition |
|---|--|
| ArgumentException (AkiraNetwork.VirtualStorageLibrary.Arguments.ArgumentException) | Thrown when the prefix is an empty string. |

1.html

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)
1.html

GetHashCode()

VirtualNode
Returns a hash code for this instance.
(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    VirtualNodeExtension
public int GetHashCode()
(AkiraNetwork.VirtualStorageLibrary)
    VirtualNodeListConditions
```

Returns

(AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|---|--------------------------------|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary) | A hash code for this instance. |

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

VirtualNodeNotFoundException

Overrides

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

VirtualNodeTypeFilter

IsValidNodeName(VirtualNodeName)

Checks if a node name is valid.

Declaration

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

Parameters

NotifyNodeDelegate

Type (AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

VirtualCycleDetector

Returns

VirtualDirectory

Type (AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

bool (AkiraNetwork.VirtualStorageLibrary)

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

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

ResetCounter()

(AkiraNetwork.VirtualStorageLibrary)

Resets the counter used for node name generation.

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

public static void ResetCounter()

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

ToString()

VirtualItem<T>

Returns the name of the node as a string.

1.html)

Declaration

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeConditions ToString()

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

Returns (AkiraNetwork.VirtualStorageLibrary)

Type VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

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

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Operators

operator ==(VirtualNodeName?, VirtualNodeName?)

Determines whether two VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) instances are equal.

Declaration

1

```
public static bool operator ==(VirtualNodeName? left, VirtualNodeName? right)
(AkiraNetwork.VirtualStorageLibrar
```

NotifyN Parameters

| Type | Name | Description |
|--|-------|--|
| (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryName.html) | left | The left VirtualDirectoryName (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryName.html) VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryName.html) |
| VirtualDirectoryNameAdapter<T> | | to compare. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryName.html) | right | The right VirtualDirectoryName (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryName.html) VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryName.html) |
| VirtualGroupCondition<T, TKey> | | to compare. |

(Aktiviteler)

| Type | Description |
|--|---|
| VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualItem) | |
| bool VirtualItem (http://AkiraNetwork.VirtualStorageLibrary/api/system.boolean) VirtualItemAdapter<T> | true if the specified instances are equal; otherwise, false . |

implicit operator string(VirtualNodeName)

(AkiraNetwork.virtualStorage.getLibrary)

Implicitly converts a VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) to a string.

Declaration

| Parameters | <p>VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary)</p> <p>public string(string VirtualnodeName nodeName)</p> <p>VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary)</p> <p>VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary)</p> | |
|---|--|--|
| Type VirtualnodeName (AkiraNetwork.VirtualStorageLibrary) (AkiraNetwork.VirtualStorageLibrary.VIRTUALNODENOTFOUNDException) (AkiraNetwork.VirtualStorageLibrary) VirtualNodeType (AkiraNetwork.VirtualStorageLibrary) VirtualNodeTypeFilter | Name <i>nodeName</i> | Description The VirtualnodeName (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) .html) to convert. |

Returns

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



implicit operator VirtualNodeName(string)

(AkiraNetwork.VirtualStorageLibrary)
Implicitly converts a string to a VirtualNodeName ([AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html](#)).
NotifyNodeDelegate

Declaration

```
PatternMatch  
(AkiraNetwork.VirtualStorageLibrary) public static implicit operator VirtualNodeName(string name)  
VirtualCycleDetector  
(AkiraNetwork.VirtualStorageLibrary)
```

Parameters

| Type | Name | Description |
|--|------|------------------------|
| VirtualDirectoryName (https://learn.microsoft.com/dotnet/api/system.string) | name | The string to convert. |

Returns

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

(AkiraNetwork.VirtualStorageLibrary)
VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)
VirtualItemAdapter<T>

2.html)

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

operator !=(VirtualNodeName?, VirtualNodeName?)

(AkiraNetwork.VirtualStorageLibrary)

Determines whether two VirtualNodeName ([AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html](#)) instances are not equal.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
VirtualItem<T>  
(AkiraNetwork.VirtualStorageLibrary) public static bool operator !=(VirtualNodeName? left, VirtualNodeName? right)  
1.html)
```

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|---|-------|--|
| VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary) VirtualNodeName VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary) Virtual (AkiraNetwork.VirtualStorageLibrary) NodeName.html) VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary) VirtualNodeName | left | The left VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) to compare. |
| VirtualNode (AkiraNetwork.VirtualStorageLibrary) VirtualNodeException (AkiraNetwork.VirtualStorageLibrary) Virtual (AkiraNetwork.VirtualStorageLibrary) VirtualNodeType (AkiraNetwork.VirtualStorageLibrary) VirtualNodeTypeFilter | right | The right VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) to compare. |

Returns

| Type | Description |
|---|--|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | true if the specified instances are not equal; otherwise, false. |

Implements

NotifyNodeDelegate

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>)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)

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

s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

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

Extension Methods

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)

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

s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

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 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)

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 ([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

VirtualNodeNotFoundException()

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

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  
PatternMatch  
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html) class with a specified error message.  
    (AkiraNetwork.VirtualStorageLibrary.  
VirtualNodeNotFoundException(string, Exception)  
    (AkiraNetwork.VirtualStorageLibrary.  
VirtualCycleDetector  
    (AkiraNetwork.VirtualStorageLibrary.  
VirtualDirectory  
public VirtualNodeNotFoundException(string message)  
    (AkiraNetwork.VirtualStorageLibrary.  
VirtualDirectoryAdapter<T>  
    (AkiraNetwork.VirtualStorageLibrary.  
Parameters  
    (AkiraNetwork.VirtualStorageLibrary.  
1.html)  
Type VirtualException  
    (AkiraNetwork.VirtualStorageLibrary.  
string (AkiraNet.WeakVirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.string))  
    (AkiraNetwork.VirtualStorageLibrary.  
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.  
    (AkiraNetwork.VirtualStorageLibrary.  
VirtualItemAdapter<T>  
    (AkiraNetwork.VirtualStorageLibrary.  
1.html)  
VirtualItem<T>VirtualNodeNotFoundException(string message, Exception innerException)  
    (AkiraNetwork.VirtualStorageLibrary.  
1.html)  
Parameters  
    VirtualNode  
Type (AkiraNetwork.VirtualStorageLibrary.  
VirtualNodeContext  
string (AkiraNetwork.VirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.string))  
VirtualNodeExtensions  
Exception (AkiraNetwork.VirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.exception))  
VirtualNodeCondition  
VirtualNodeName  
VirtualNodeNotFoundException  
VirtualNodeType  
ISerializable (AkiraNetwork.VirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.runtime.serialization.iserializable))  
VirtualNodeTypeFilter
```

| Type | Name | Description |
|--|---------|--------------------|
| string (AkiraNet.WeakVirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.string)) | message | The error message. |

VirtualNodeNotFoundException(string, Exception)

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

```
VirtualItem<T>VirtualNodeNotFoundException(string message, Exception innerException)  
    (AkiraNetwork.VirtualStorageLibrary.  
1.html)
```

| Type | Name | Description |
|--|----------------|--|
| string (AkiraNet.WeakVirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.string)) | message | The error message. |
| Exception (AkiraNetwork.VirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.exception)) | innerException | The exception that is the cause of this exception. |

Implements

```
VirtualNode (AkiraNetwork.VirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.runtime.serialization.iserializable))  
VirtualNodeType (AkiraNetwork.VirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.runtime.serialization.iserializable))  
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

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 |
|------------------|---|
| Directory | Represents a directory node. |
| VirtualException | Represents a user-defined exception type. |
| Item | Represents an item node with a user-defined type T. |
| None | Represents an unspecified or unknown node type. |
| SymbolicLink | Represents a symbolic link node. |

(AkiraNetwork.VirtualStorageLibrary)
 VirtualItem

(AkiraNetwork.VirtualStorageLibrary)
 VirtualItemAdapter<T>
VirtualTextFormatter.GenerateSingleTableDebugText<VirtualNodeType>(VirtualNodeType)
(AkiraNetwork.VirtualStorageLibrary)
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
1.html)
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

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) Filter for 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

Represents a path within the virtual storage. Provides functionality for manipulating, comparing, normalizing, and splitting paths.

Inheritance

NotifyNodeDelegate

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

↳ VirtualPath

PatternMatch

Implementation

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

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

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

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

Inherited Members

1.html

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

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

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

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

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

2.html

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

VirtualID

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

(AkiraNetwork.VirtualStorageLibrary)

Assembly: VirtualStorageLibrary.dll

VirtualItem

Syntax

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

public class VirtualPath : IEquatable<VirtualPath>, IComparable<VirtualPath>, IComparable

1.html

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

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

specified connection parts.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

public VirtualPath(IEnumerable<VirtualNodeName> parts)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Parameters

| Type | Name | Description |
|---|-------|-------------------------------|
| IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) <VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)> | parts | The collection of path parts. |

▼

VirtualPath(string)

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

NotifyNodeDelegate
Initializes a new instance of the VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) class with the specified path string.
PatternMatch

Declaration

```
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary.VirtualPath(string path))
VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary.VirtualPath)
VirtualDirectoryAdapter<T>
```

Parameters

| Type | Name | Description |
|---|------|--------------------------|
| AkiraNetwork.VirtualStorageLibrary.VirtualPath(string path) | path | The initial path string. |

Properties

VirtualID

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

Depth

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

Gets the depth of the path.

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

Declaration

```
1.html)
VirtualItem<T>
public int Depth { get; }
```

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

1.html)

VirtualItem<T>

public int Depth { get; }

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

1.html)

PropertyValue

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

VirtualNodeContext

| Type | Description |
|---|--|
| int (https://learn.microsoft.com/dotnet/api/system.int32) | An integer representing the depth of the path. |

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

VirtualNodeName

Gets the directory part of the path.

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

Declaration

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

VirtualNodeType

VirtualNodeType DirectoryPath { get; }

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

VirtualNodeTypeFilter

Property Value

| Type | Description |
|--|--|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | A VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) representing the directory part of the path. |

(AkiraNetwork.VirtualStorageLibrary)

Dot

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary)
Gets the dot (".") symbol.

PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)
public static string Dot { get; }

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

Property Value

VirtualDirectoryAdapter<T>

| Type | Description |
|---------------------------------------|---------------------------------------|
| VirtualDirectoryAdapter<T> 1.html) | A string representing the dot symbol. |

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

DotDot

2.html)

Gets the double dot ("..") symbol.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

public static string>DotDot { get; }

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Property Value

VirtualItem<T>

| Type | Description |
|---------------------------|--|
| VirtualItem<T> 1.html) | A string representing the double dot symbol. |

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

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

FixedDepth

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

Gets the depth of the fixed path.

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

public int FixedDepth { get; }

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Description |
|---|--|
| int (https://learn.microsoft.com/dotnet/api/system.int32) | An integer representing the depth of the fixed path. |

Remarks

The fixed depth indicates the number of parts in the path, excluding those with wildcards.

FixedPath

NotifyNodeDelegate

Gets the [AkiraNetwork.VirtualStorageLibrary.VirtualPath](#) matching.

PatternMatch

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

```
public VirtualPath FixedPath { get; }
```

VirtualDirectory

Property Value

VirtualDirectoryAdapter<T>

Type

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualPath

VirtualException

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualGroupCondition<T, TKey>

Remarks

2.html)

The fixed path excludes parts containing wildcards.

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

IsAbsolute

(AkiraNetwork.VirtualStorageLibrary)

Declaration

1.html)

VirtualItem<T>

Type

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualNode

Property Value

(AkiraNetwork.VirtualStorageLibrary)

Type

virtualNodeContext

Type

(AkiraNetwork.VirtualStorageLibrary)

bool

(VirtualNodeExtensions)

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

IsDot

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

Gets a value indicating whether the instance's path is a dot.

VirtualNodeNotFoundException

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

```
public bool IsDot { get; }
```

Property Value

| Type | Description |
|---|--|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | true if the instance's path is a dot; otherwise, false . |

NotifyNodeDelegate
(AkiraNetwork.VirtualStorageLibrary)

PatternMatch
(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector
Gets a value indicating whether the instance's path is a double dot.
(AkiraNetwork.VirtualStorageLibrary)

Declaration
Directory
(AkiraNetwork.VirtualStorageLibrary)

```
public VirtualDirectoryAdapter<T> Get();
```

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Property Value

| Type | Description |
|--|---|
| AkiraNetwork.VirtualStorageLibrary VirtualGroupCondition<T, TKey> bool (https://learn.microsoft.com/dotnet/api/system.boolean) n) VirtualID (AkiraNetwork.VirtualStorageLibrary) VirtualItem (AkiraNetwork.VirtualStorageLibrary) | true if the instance's path is a double dot; otherwise, false . |

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

IsEmpty
Gets a value indicating whether the instance's path is empty.
(AkiraNetwork.VirtualStorageLibrary)

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

```
public bool IsEmpty { get; }
```

1.html)

VirtualNode
(AkiraNetwork.VirtualStorageLibrary)

Property Value
VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|--|--|
| AkiraNetwork.VirtualStorageLibrary VirtualNodeExtensions (https://learn.microsoft.com/dotnet/api/system.boolean) | true if the instance's path is empty; otherwise, false . |

VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName
(AkiraNetwork.VirtualStorageLibrary)

IsEndsWithSlash
Gets a value indicating whether the instance's path ends with a slash.
(AkiraNetwork.VirtualStorageLibrary)

Declaration
NodeType
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter
(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
public List<VirtualNodeName> PartsList { get; }
```

Property Value

| Type | Description |
|---|---|
| List<(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName)(https://learn.microsoft.com/dotnet/api/system.collection.s.generic.list-1)> | A list of VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) objects representing the parts of the path. |

Path

Gets the path string.
Declaration: `public string Path { get; }`

Property Value
Type: `(AkiraNetwork.VirtualStorageLibrary.VirtualItem)(1.html)`

| Type | Description |
|---|--|
| <code>(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>)(string)(https://learn.microsoft.com/dotnet/api/system.string)</code> | The string representation of the path. |

Root

Gets the root path.
Declaration: `public static string Root { get; }`

Property Value
Type: `(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext)(1.html)`

| Type | Description |
|--|--------------------------------------|
| <code>(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions)(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext)(VirtualNodeExtensions.Root)</code> | A string representing the root path. |

| Type | Description |
|---|--------------------------------------|
| <code>(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter)(VirtualNodeTypeFilter.Root)</code> | A string representing the root path. |

Gets the path separator character.

Declaration

```
public static char Separator { get; }
```



Property Value

| Type | Description |
|---|--|
| char (https://learn.microsoft.com/dotnet/api/system.char) | A character representing the path separator. |

PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

AddEndSlash()

(AkiraNetwork.VirtualStorageLibrary)

Adds a trailing slash to the path if it doesn't already have one.

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

2.html)

Returns

VirtualID

| Type | Description |
|---|--|
| VirtualItem (AkiraNetwork.VirtualStorageLibrary) | A new VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem) |

1.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

AddStartSlash()

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeCreation

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

Returns

(AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|---|---|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary) | A new VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) with a leading slash. |

VirtualPath

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualPath

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

ArePathsSubdirectories(VirtualPath, VirtualPath)

Determines whether one path is a subdirectory of another path.

Declaration

```
public static bool ArePathsSubdirectories(VirtualPath path1, VirtualPath path2)
```

Parameters

| | | | |
|-------------|---|-------|---|
| Type | (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate) | Name | Description |
| VirtualPath | (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path1 | The first VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) |
| VirtualPath | (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path2 | The second VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) |

Exceptions

| | | |
|-------------|---|--|
| Type | (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey>) | Description |
| VirtualItem | (https://learn.microsoft.com/dotnet/api/system.boolean) | true if one path is a subdirectory of the other; otherwise, false. |

Exceptions

| | | |
|-------------------|---|---------------------------------|
| Type | (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>) | Condition |
| ArgumentException | (https://learn.microsoft.com/dotnet/api/system.argumentexception) | Thrown if either path is empty. |

Combine(params VirtualPath[])

VirtualNodeContext

Combines the instance's path with the specified paths.
(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
public VirtualPath Combine(params VirtualPath[] paths)
```

VirtualNodeName

Parameters

| | |
|-----------------|---|
| Type | (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException) |
| VirtualNodeType | (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter) |
| VirtualNodeType | (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter) |

| Type | Name | Description |
|--|-------|---|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) [] | paths | An array of VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) objects to combine. |

Returns

(AkiraNetwork.VirtualStorageLibrary.

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

| | |
|--|---|
| VirtualPath PatternMatch (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | A new VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) representing the combined path. |
| VirtualCycleDetector | |

(AkiraNetwork.VirtualStorageLibrary.

VirtualDirectory

Combine(params string[])

VirtualDirectoryAdapter<T>

Combines the specified paths into a single path.
(AkiraNetwork.VirtualStorageLibrary.

Declaration

VirtualException

```
(AkiraNetwork.VirtualStorageLibrary)
public static string Combine(params string[] paths)
    VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary.
```

Parameters

| Type | Name | Description |
|---|-------|--------------------------------------|
| VirtualID (AkiraNetwork.VirtualStorageLibrary. | paths | An array of path strings to combine. |

Returns

VirtualItemAdapter<T>

| Type | Description |
|--|-------------------------|
| VirtualItem<T> (https://learn.microsoft.com/dotnet/api/system.string)[] | A combined path string. |

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

| Type | Description |
|--|-------------------------|
| VirtualItem<T> (https://learn.microsoft.com/dotnet/api/system.string) | A combined path string. |

CombineFromIndex(VirtualPath, int)

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeContext

Combines the instance's path with the parts of another path starting from a specified index.
(AkiraNetwork.VirtualStorageLibrary.

Declaration

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeListConditions

```
public VirtualPath CombineFromIndex(VirtualPath path, int index)
(AkiraNetwork.VirtualStorageLibrary.
```

VirtualNodeName

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

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeTypeFilter

| Type | Name | Description |
|---|---------------|--|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) └ int (https://learn.microsoft.com/dotnet/api/system.int32) NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary) | path index | The VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) whose parts to combine. The starting index from which to combine the parts. |

Returns

PatternMatch

| Type | Description |
|--|---|
| VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary) | A new VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) representing the combined path. |

CompareTo(VirtualPath?)

VirtualException

Compares the instance's path with another VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html).

VirtualGroupCondition<T, TKey>

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
2.html)
```

```
public int CompareTo(VirtualPath? other)
```

```
(AkiraNetwork.VirtualStorageLibrary)
VirtualItem  
Parameters
```

```
(AkiraNetwork.VirtualStorageLibrary)
```

| Type | Name | Description |
|--|-------|--|
| VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary) VirtualPath 1.html) (AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>) VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary) 1.html) | other | The other VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) to compare with. |

VirtualNode

VirtualStorageLibrary

| Type | Description |
|---|--|
| VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary) int (https://learn.microsoft.com/dotnet/api/system.int32) VirtualNodeExtensions | An integer indicating the relative order of the paths. |

```
(AkiraNetwork.VirtualStorageLibrary)
```

```
VirtualNodeListConditions
```

```
(AkiraNetwork.VirtualStorageLibrary)
```

```
VirtualnodeName
```

Compares the instance's path with another object.

VirtualNodeNotFoundException

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
```

```
VirtualNodeType
```

```
public int CompareTo(object? obj)
```

```
(AkiraNetwork.VirtualStorageLibrary)
```

```
VirtualNodeTypeFilter
```

Parameters

| Type | Name | Description |
|--|------------|-----------------------------|
| object (https://learn.microsoft.com/dotnet/api/system.object) | <i>obj</i> | The object to compare with. |

Returns

| Type | Description |
|---|--|
| int (AkiraNetwork.VirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.int32) NotifyNodeDelegate) | An integer indicating the relative order of the paths. |

NOTES

(AkiraNetwork)

| Type | Condition |
|---|--|
| (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector AkiraNetwork.VirtualStorageLibrary.ArgumentException (AkiraNetwork.VirtualStorageLibrary. https://learn.microsoft.com/dotnet/api/system.argumentexception) VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualPath VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) VirtualDirectoryAdapter<T> | Thrown if the object is not a VirtualPath |

VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary 1.html)

Equals(VirtualPath?)
(AkiraNetwork.VirtualStorageLibrary)
Indicates whether this path is equal to the specified VirtualPath
(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) object.

2.html)

```
Declaration  
VirtualID  
(AkiraNetwork.VirtualStorageLibrary  
public bool Equals(VirtualPath? other)
```

virtual
(Alien)

(AkiraNetwork.VirtualStorage)

Virtual Parameters

(AkiraNetwork.VirtualStorageLibrar

Type
f.html)

Virtually all items <

Virtual
Clouds

(AkilaNetwork Virtual storage Library! at
http://1.htm)

Virtua

VirtualNode

Returns^(Aki)

Returns VirtualNodeContext

| Type | Description |
|---|--|
| bool (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary. VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary. | true if the specified VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) is equal to the instance's path; otherwise, false. |

VirtualN

VirtualNodeName
(Alio, Net, and Mist) [String] [Required]

(AkiraNetwork.VirtualStorageLib)

Equals(*object*): FoundException
(AkiraNetwork.VirtualStorageLibrary)
Indicates whether this path is equal to the specified object.

Indice Virtual

Declaration (AkiraNetwork.VirtualStorageLibrary)

```
public override bool Equals(object? obj)
```

Parameters

| Type | Name | Description |
|---|------|---|
| object (https://learn.microsoft.com/dotnet/api/system.object) | obj | The object to compare with the instance's path. |

NotifyNodeDelegate

Returns

AkiraNetwork.VirtualStorageLibrary

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

booVirtualCycleDetector

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

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

(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

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

Return the hash code for this path.

VirtualID

Declaration
(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

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

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Type
VirtualItem<T>

Description

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

VirtualNode

Overrides
(AkiraNetwork.VirtualStorageLibrary)

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

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

Retrieves the VirtualNodeCondition from the instance's path.

(AkiraNetwork.VirtualStorageLibrary)

Declaration
VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

public VirtualNodeName GetNodeName() (<https://learn.microsoft.com/dotnet/api/system.virtualnodeexception>)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Description |
|--|---|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.Virtual NodeName.html) | A VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) object representing the node name. |

(AkiraNetwork.VirtualStorageLibrary)

GetParentPath()

(AkiraNetwork.VirtualStorageLibrary)
Gets the parent path of the instance's path.
PatternMatch

DeclarationNetwork.VirtualStorageLibrary

VirtualCycleDetector

(AkiraNetwork-VirtualStorage) library

public virtual VirtualDirectory

(AkiraNetwork VirtualStorage) library

Returns: `void` `Director::Adapter::T`

| Type | Description |
|---|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | A VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) representing the parent path. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | A VirtualGroupCondition< T, TKey > |

(AkiraNetwork.VirtualStorageLibrary)

?html)

VirtualID: - - - -

GetPartitionsLinkedList()

www.akiar.com

GetsVirtualPath Gets the parts of the path as a linked list.

(AkiraNetwork.VirtualStation

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

public final) LinkedL

VirtualItem<T>

- (AkiraN

| Type | Description |
|--|--|
| <code>(AkiraNetwork.VirtualStorageLibrary.VirtualNode VirtualNodeName VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNo deName.html) > VirtualNodeName < VirtualNodeExtensions < VirtualNodeName < VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNo deName.html)</code> | A <code>LinkedList<T></code> (https://learn.microsoft.com/dotnet/api/system.collections.generic.linkedlist-1) of <code>VirtualNodeName</code> objects representing the path parts. |

VirtualDiskOnNetFx.cs - IFE section

VirtualNodeNotF

(AkiraNetwork.VirtualStorageLibrary

~~ParticipantsListType~~

Retrieves the node parts of the path as a list. This method converts the node parts stored in a linked list structure into a list structure, making it more convenient for sequential access and operations that require indexing.

Declaration

```
public List<VirtualNodeName> GetPartsList()
▼
```

Returns `(AkiraNetwork.VirtualStorageLibrary)`

| Type | Description |
|---|--|
| <code>NotifyNodeDelegate</code> <code>(AkiraNetwork.VirtualStorageLibrary)</code> <code>List</code> <code>PatternMatch</code> <code>(https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1)</code> <code>(AkiraNetwork.VirtualStorageLibrary)</code> <code>m.collections.generic.list-1)</code> <code><</code> <code>(AkiraNetwork.VirtualStorageLibrary)</code> <code>VirtualNodeName</code> <code>VirtualDirectory</code> <code>(AkiraNetwork.VirtualStorageLibrary)</code> <code>VirtualNodeName.html</code> <code>VirtualDirectoryAdapter<T></code> <code>></code> <code>(AkiraNetwork.VirtualStorageLibrary)</code> <code>1.html)</code> <code>VirtualException</code> | A <code>List<T></code> <code>(https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1)</code> of <code>VirtualNodeName</code> <code>(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)</code> objects representing the node parts of the path. |

GetRelativePath(VirtualPath)

`(AkiraNetwork.VirtualStorageLibrary)`

`VirtualGroupCondition<T, TKey>`

Gets `(AkiraNetwork.VirtualStorageLibrary)` the relative path from the specified base path.

`2.html)`

Declaration

`VirtualID`

`(AkiraNetwork.VirtualStorageLibrary)`

```
public VirtualPath GetRelativePath(VirtualPath basePath)
```

`VirtualItem`

`(AkiraNetwork.VirtualStorageLibrary)`

`VirtualItemAdapter<T>`

`(AkiraNetwork.VirtualStorageLibrary)`

`Type`
`f.html)`

`VirtualPath`
`(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)`

`(AkiraNetwork.VirtualStorageLibrary)`

`1.html)`

Returns `(VirtualNode)`

`Type`
`(AkiraNetwork.VirtualStorageLibrary)`

`VirtualNodeContext`

`VirtualPath`
`(AkiraNetwork.VirtualStorageLibrary)`

`(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)`

`(AkiraNetwork.VirtualStorageLibrary)`

`VirtualNodeListConditions`

Exceptions `(AkiraNetwork.VirtualStorageLibrary)`

`VirtualNodeName`

`(AkiraNetwork.VirtualStorageLibrary)`

`VirtualNodeNotFoundException`

`(AkiraNetwork.VirtualStorageLibrary)`

`VirtualNodeType`

`(AkiraNetwork.VirtualStorageLibrary)`

`VirtualNodeTypeFilter`

Type

| Type | Name | Description |
|---|-----------------------|----------------|
| <code>AkiraNetwork.VirtualStorageLibrary</code> <code>VirtualPath</code> <code>(AkiraNetwork.VirtualStorageLibrary)</code> <code>(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)</code> <code>(AkiraNetwork.VirtualStorageLibrary)</code> | <code>basePath</code> | The base path. |

`VirtualNodeContext`
`VirtualPath`
`(AkiraNetwork.VirtualStorageLibrary)`
`(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)`
representing the relative path.

| Type | Condition |
|---|--|
| InvalidOperationException (https://learn.microsoft.com/dotnet/api/system.invalidoperationexception) (AkiraNetwork.VirtualStorageLibrary.NotifNodeDataException) | Thrown if the instance's path is not absolute: This exception occurs when the method is called on a VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) instance that does not start with the root path separator, indicating that it is not an absolute path. Thrown if the base path is not absolute: This exception occurs when the basePath argument does not start with the root path separator, indicating that it is not an absolute path. Both the instance's path and the base path must be absolute paths to compute a relative path. |

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary)

PatternMatch

IsSubdirectory(VirtualPath)

VirtualCycleDetector.

etermines whether this instance's path
[AkiraNetwork Virtual Storage] library

Declaring Virtual Directory

(AkiraNetwork VirtualStorage) library

VirtualDirector (Adapter)

```
public bool IsSubdirectory(VirtualDirectory virtualDir)
```

(Akira N)

l.html)

MetersException

| Type | Name | Description |
|---|------------|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey>) | | |
| VirtualPath | parentPath | The parent VirtualPath |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | | (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) |
| VirtualID | | . |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath) | | |

VirtualItem

Return AkiraNetwork.VirtualStorageLibrary

| Type | Description |
|---|---|
| VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary) | |
| bool.html) (http://urlitemid@soft.com/dotnet/api/system. | true if this instance's path is a subdirectory of the parent path; otherwise, false. |
| bool(AkiraNetwork.VirtualStorageLibrary) | |

1.1.1.11)

VirtualNode Options

| Type | Condition |
|---|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext) | |
| (AkiraNetwork.VirtualStorageLibrary.ArgumentException) (VirtualNodeExtensions)(https://learn.microsoft.com/dotnet/api/system.argumentexception) on) (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext) | Thrown if this instance's path or the parent path is empty. |

VirtualNodeListCo

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

NormalizePath() (Akka.NETCore.Visitors)

VirtualNodeNotFoundException

(AkiraNetwork VirtualStorage library)

(AkiraNetwork.virtualStorageLibrary)

Declaration Node type

(AkiraNetwork.VirtualStorageLibrary

VirtualNodeTypeFilter

| Type | Name | Description |
|--|------|--|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | The VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) to compare with. |

Returns

(AkiraNetwork.VirtualStorageLibrary.

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

Tostring()

(AkiraNetwork.VirtualStorageLibrary.

VirtualDirectoryAdapter<T>

Returns a string that represents the instance's path.
(AkiraNetwork.VirtualStorageLibrary.

Declaration

VirtualException

(AkiraNetwork.VirtualStorageLibrary.

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

VirtualPath.html)

Returns

VirtualID

Type
(AkiraNetwork.VirtualStorageLibrary.

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

Description

A string that represents the instance's path.

Overrides

VirtualItemAdapter<T>

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

1.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary.

TrimEndSlash()

1.html)

Removes the trailing slash from the path.

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.

Declaration

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary.

```
public VirtualPath TrimEndSlash()
    VirtualNodeExtensions
```

VirtualPath.html)

Returns

VirtualNodeListConditions

Type
(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeName

VirtualPath
(AkiraNetwork.VirtualStorageLibrary.

(AkiraNetwork.NodeStorageException.

VirtualPath
(AkiraNetwork.VirtualStorageLibrary.

Description

A new

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) without the trailing slash.

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeTypeFilter

Operators

operator +(VirtualPath, VirtualNodeName)

Combines a VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) with a VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html).

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.  
    NotifyNodeDelegate  
    p(AkiraNetwork.VirtualStorageLibrary.  
        operator +(VirtualPath path, VirtualNodeName nodeName)  
        PatternMatch  
        (AkiraNetwork.VirtualStorageLibrary.  
            VirtualCycleDetector
```

Parameters

| Type | Name | Description |
|---|----------|--|
| AkiraNetwork.VirtualStorageLibrary. VirtualDirectory VirtualPath (AkiraNetwork.VirtualStorageLibrary. VirtualPathAdapter<T> h.html) (AkiraNetwork.VirtualStorageLibrary. 1.html) VirtualException | path | The VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.htm l) . . . |
| AkiraNetwork.VirtualStorageLibrary. VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNo deAdapter<T> de.html) VirtualID (AkiraNetwork.VirtualStorageLibrary. VirtualItem | nodeName | The VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNa me.html) . . . |

Returns

| Type | Description |
|--|---|
| AkiraNetwork.VirtualStorageLibrary. VirtualItemAdapter<T> VirtualPath (AkiraNetwork.VirtualStorageLibrary.Virt ualPath.html) VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary. VirtualItem 1.html) | A new VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) representing the combined path. |

operator +(VirtualPath, VirtualPath)

Combines two VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) instances.

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.  
    VirtualNodeContext  
    VirtualPath  
    public static VirtualPath operator +(VirtualPath path1, VirtualPath path2)  
    (AkiraNetwork.VirtualStorageLibrary.  
        VirtualNodeName
```

Parameters

| |
|---|
| AkiraNetwork.VirtualStorageLibrary. VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary. VirtualNodeType (AkiraNetwork.VirtualStorageLibrary. VirtualNodeTypeFilter |
|---|

Combines a VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) with a string.

Declaration

```
public static VirtualPath operator +(VirtualPath path, string str)
```

Parameters

| Type | Description |
|--|---|
| NotifyNodeDelegate | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.AkiraNetwork_VirtualStorageLibrary_VirtualPath.PatternMatch.html) | <p><i>path</i></p> <p>The VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)</p> |
| VirtualCycleDetector | . |
| string (https://learn.microsoft.com/dotnet/api/system.string) | <p><i>str</i></p> <p>The string to combine with the path.</p> |

Returns VirtualDirectoryAdapter<|>
(AkiraNetwork.VirtualStorageLibrary)

| Returns 1.html) | |
|---|---|
| Type virtualException | Description |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath VirtualPath VirtualGroupCondition< T, TKey > (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html alPath.html) 2.html) | A new VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) representing the combined path. |

VirtualID
(AkiraNetwork.VirtualStorageLibrary
operator<(char, VirtualPath)

(AkiraNetwork.VirtualStorageLibrary VirtualPath.html)

(AkiraNetwork.VirtualStorageLibrary
Declaration 1.html)

```
VirtualItem<T>
public static VirtualPath operator +(char chr, VirtualPath path)
(AkiraNetwork.VirtualStorageLibrary
1.html)
```

Parameters Node

| Type | Description |
|--|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType) VirtualNodeType | |
| VirtualNodeContext | |
| char | The character to combine with the path. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions) System.String | |
| char | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPathConditions) VirtualPathConditions | |
| VirtualPath | The |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath) VirtualPath | VirtualPath |
| VirtualnodeName | (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath) VirtualPath | |
| VirtualNodeNotFoundException | . |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter) VirtualNodeTypeFilter | |

| Type | Description |
|--|---|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | A new VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) representing the combined path. |

▼

operator +(string, VirtualPath)

NotifyNodeDelegate
Combines a string with a VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html).
(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
VirtualCycleDetector
public static VirtualPath operator +(string str, VirtualPath path)
(AkiraNetwork.VirtualStorageLibrary)
```

VirtualDirectory

Parameters
(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|---|------|---|
| VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary) string (html) (https://learn.microsoft.com/dotnet/api/system. StringException) string (AkiraNetwork.VirtualStorageLibrary) | str | The string to combine with the path. |
| VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary) (AkiraNetwork.VirtualStorageLibrary.VirtualPath. html) VirtualID (AkiraNetwork.VirtualStorageLibrary) | path | The VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) |

VirtualItem

Returns
(AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|--|---|
| VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary) VirtualPath (1.html) (AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>. VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary) | A new VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) representing the combined path. |

1.html)

VirtualNode

operator ==(VirtualPath?, VirtualPath?)

VirtualNodeContext
Determines whether two VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) instances are equal.
(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeListConditions
public static bool operator ==(VirtualPath? left, VirtualPath? right)
(AkiraNetwork.VirtualStorageLibrary)
```

VirtualNodeName

Parameters
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Name | Description |
|---|--------------|--|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>left</i> | The left VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) . |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary.PatternMatch) | <i>right</i> | The right VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) . |

Return Value

| Type | Description |
|---|--|
| VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector) | true if the paths are equal; otherwise, false. |

implicit operator string(VirtualPath?)

Implicitly converts a VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) to a string (https://learn.microsoft.com/dotnet/api/system.string).

VirtualGroupCondition<T, TKey>

Declaration

```
VirtualGroupCondition<T, TKey>
public static implicit operator string(VirtualPath? virtualPath)
{
    return virtualPath?.ToString();
}
```

Parameters

virtualPath

| Type | Name | Description |
|--|--------------------|--|
| VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter) | <i>virtualPath</i> | The VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) object. |

VirtualNode

Return Value

| Type | Description |
|---|---|
| VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext) | The string representation of the VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html), or an empty string if the path is null. |

VirtualNodeListConditions

VirtualNodeName

VirtualNodeType

VirtualNodeTypeFilter

implicit operator VirtualPath(string)

Implicitly converts a string (https://learn.microsoft.com/dotnet/api/system.string) to a VirtualPath

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html).

VirtualNodeType

VirtualNodeTypeFilter

```
public static implicit operator VirtualPath(string path)
```

Parameters

| Type | | Name | Description |
|--|--|------|--|
| string (https://learn.microsoft.com/dotnet/api/system.string) | | path | The string representation of the path. |

Returns

| Type | Description |
|---|--|
| VirtualPath (https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualPath) | A VirtualPath (https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) object representing the given string path. |

operator !=(VirtualPath?, VirtualPath?)

Declared in [AkiraNetwork.VirtualStorageLibrary.VirtualPath](https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualPath)

VirtualGroupCondition<T, TKey>

```
public static bool operator !=(VirtualPath? left, VirtualPath? right)
```

2.html

VirtualID

Parameters

left ([AkiraNetwork.VirtualStorageLibrary.VirtualPath](https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualPath))

right ([AkiraNetwork.VirtualStorageLibrary.VirtualPath](https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualPath))

| Type | Name | Description |
|--|-------|--|
| VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem) | left | The left VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath) |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath) | right | The right VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath) |

Returns [VirtualNodeExtensions](https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions)

| Type | Description |
|--|--|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | true if the paths are not equal; otherwise, false. |

VirtualNodeName

VirtualNodeType

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>)

VirtualNodeTypeFilter

Implements

IMissingNodeFoundException

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>)

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 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)))
↳ VirtualDirectoryAdapter<T>

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

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

AkiraNetwork.html

AkiraNetwork.VirtualStorageLibrary

Assembly

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)

Constructors

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

```
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeExtensions
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. |

| Type Parameters | |
|--|--|
| Name | Description |
| T (AkiraNetwork.VirtualStorageLibrary.VirtualItem) | The type of elements in the source sequence. |

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 VirtualNodeGroup GroupAndSort<T>(this IEnumerable<T> source, VirtualGroupCondition<T, object>? groupCondition = null, List<VirtualSortCondition<T>>? sortConditions = null)
            (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions)
```

Parameters

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName)

| Type | Name | Description |
|--|--------|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName IEnumerable VirtualNodeNotFoundException (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) <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) (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. |

Return: [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 |

Class VirtualStorageSettings

Manages settings for the virtual storage. This class implements the singleton pattern and holds default settings and values by type.

Inheritance

NotifyNodeDelegate

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

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

Assembly

VirtualStorageLibrary.dll

Syntax

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

public class VirtualStorageSettings
(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)
1.html

VirtualList<T>

(AkiraNetwork.VirtualStorageLibrary)
1.html

Properties

InvalidNodeNameCharacters

(AkiraNetwork.VirtualStorageLibrary)

Gets or sets an array of characters that are invalid in node names.

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

public char[] InvalidNodeNameCharacters { get; set; }
VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

Property Value

(AkiraNetwork.VirtualStorageLibrary)

Type VirtualNodeNotFoundException

char (AkiraNetwork.VirtualStorageLibrary)

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

[] (AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Description |
|------------------------------|---|
| VirtualNodeNotFoundException | An array of characters that cannot be used in node names. |

InvalidNodeNames

Gets or sets an array of invalid node names.

Declaration

```
public string[] InvalidNodeNames { get; set; }
```

Property Value
(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

Type AkiraNetwork.VirtualStorageLibrary

string PatternMatch

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

g) VirtualCycleDetector

[] (AkiraNetwork.VirtualStorageLibrary

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary

VirtualDirectoryAdapter<T>

NodeListConditions

(AkiraNetwork.VirtualStorageLibrary

1.html)

Gets or sets the conditions for listing nodes.

VirtualException

(AkiraNetwork.VirtualStorageLibrary

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary

public VirtualNodeListConditions NodeListConditions { get; set; }

2.html)

VirtualID

Property Value

(AkiraNetwork.VirtualStorageLibrary

VirtualItem

(AkiraNetwork.VirtualStorageLibrary

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary

[VirtualNodeListConditions.html](#)

1.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary

PathDot

1.html)

Gets or sets the path representing the current directory. The default value is ".".

(AkiraNetwork.VirtualStorageLibrary

Declaration

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary

virtualNodeContext { get; set; }

(AkiraNetwork.VirtualStorageLibrary

VirtualNodeListConditions

Property Value

(AkiraNetwork.VirtualStorageLibrary

virtualNodeName

(AkiraNetwork.VirtualStorageLibrary

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

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary

VirtualNodeType

PathDotDot

(AkiraNetwork.VirtualStorageLibrary

VirtualNodeTypeFilter

Gets or sets the path representing the parent directory. The default value is "..".

Declaration

```
public string PathDotDot { get; set; }
```

Property Value

| Type | Description |
|---|---|
| (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate<T>)(https://learn.microsoft.com/dotnet/api/system.string) | The representation of the parent directory. |

PathRoot

Gets [VirtualDirectory](#) path. The default value is "/".

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

```
public string PathRoot { get; set; }
```

1.html

VirtualException

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

2.html

VirtualID

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

VirtualItem

PathSeparator

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

Gets or sets the path separator character. The default value is '/'.
(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualItem<T>

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

1.html

VirtualNode

Property Value

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

VirtualNodeContext

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

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

VirtualNodeExtensions

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

VirtualNodeListConditions

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

VirtualNodeName

Gets [VirtualNode](#) names. The default value is "dir". Used for auto-generating node names.

VirtualNodeNotFoundException

Declaration

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

VirtualNodeType

```
public string PrefixDirectory { get; set; }
```

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

Property Value

| Type | Description |
|---|---|
| string (https://learn.microsoft.com/dotnet/api/system.string) | The prefix added to auto-generated directory names. |

▼

PrefixItem

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

Gets or sets the prefix used for item names. The default value is "item". Used for auto-generating node names.
(AkiraNetwork.VirtualStorageLibrary)

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

public string PrefixItem { get; set; }

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

Property Value

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

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

Description

The prefix added to auto-generated item names.

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

Gets or sets the prefix used for symbolic link names. The default value is "link". Used for auto-generating node names.
VirtualID

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

public string PrefixSymbolicLink { get; set; }

(AkiraNetwork.VirtualStorageLibrary)

1.html

Property Value

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

Description

1.html

string

VirtualNode

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

The prefix added to auto-generated symbolic link names.

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

Settings

(AkiraNetwork.VirtualStorageLibrary)

Gets the current instance of the settings.

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

public static VirtualStorageSettings Settings { get; }

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

Property Value

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Description |
|--|---|
| VirtualStorageSettings (AkiraNetwork.VirtualStorageLibrary.VirtualStorageSettings.html) | The current instance of VirtualStorageSettings (AkiraNetwork.VirtualStorageLibrary.VirtualStorageSettings.html) . |

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

WildcardMatcher

Gets or sets the wildcard matcher.
(AkiraNetwork.VirtualStorageLibrary)

Declaration

(AkiraNetwork.VirtualStorageLibrary)

```
public IVirtualWildcardMatcher? WildcardMatcher { get; set; }
```

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

Property Value

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualException

IVirtualWildcardMatcher
(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<Key>

ml) (AkiraNetwork.VirtualStorageLibrary)

2.html)

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

Methods

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

Declaration

1.html)

VirtualItem

Initializes the settings. Existing settings are reset, and default settings are applied.

VirtualNode

Declaration

1.html)

VirtualNode

```
public static void Initialize()
```

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeException

(AkiraNetwork.VirtualStorageLibrary)

VirtualTextFormatter

Declaration

1.html)

VirtualTextFormatter

GenerateSingleTableDebugText<T>(T)

(AkiraNetwork.VirtualStorageLibrary)

Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilsili

s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Class VirtualStorageState

Manages the state of the virtual storage. This class implements the singleton pattern and maintains the current setting of [FilterAndState](#).

Inheritance

NotifyNodeDelegate

↳ object (<https://learn.microsoft.com/dotnet/api/system.object>)
↳ [\(AkiraNetwork.VirtualStorageLibrary\)](#)
↳ [VirtualStorageState](#)
↳ [PatternMatch](#)

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)))
↳ [VirtualDirectoryAdapter<T>](#)

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

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

Assembly

[VirtualStorageLibrary.dll](#)

Syntax

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

VirtualItem

public class VirtualStorageState
↳ [\(AkiraNetwork.VirtualStorageLibrary\)](#)

VirtualItemAdapter<T>

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

1.html)

VirtualList<T>

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

1.html)

Properties

[InvalidNodeNameCharacters](#)

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

Gets or sets an array of characters that are invalid in node names.

[VirtualNodeContext](#)

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

Declaration

VirtualNodeExtensions

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

public char[] InvalidNodeNameCharacters { get; set; }

[VirtualNodeListConditions](#)

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

VirtualNodeName

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

Type

VirtualNodeNotFoundException

Description

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

An array of characters that are invalid in node names.

VirtualNodeType

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

VirtualNodeTypeFilter

InvalidNodeNames

Gets or sets an array of invalid node names.

Declaration

```
public string[] InvalidNodeNames { get; set; }
```

Property Value
(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

Type AkiraNetwork.VirtualStorageLibrary

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

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

NodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

Gets or sets the conditions for listing nodes.

VirtualDirectoryAdapter<T>

Declaration

1.html

VirtualException

public VirtualNodeListConditions NodeListConditions { get; set; }

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

Property Value
(AkiraNetwork.VirtualStorageLibrary)

2.html

VirtualID

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html)

Description

The conditions for listing nodes.

PathDot

1.html

Gets or sets the path representing the current directory. The default value is ":".

(AkiraNetwork.VirtualStorageLibrary)

Declaration

1.html

VirtualNode

public VirtualNode PathDot { get; set; }

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

Property Value
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

Description

VirtualNodeListConditions

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

(AkiraNetwork.VirtualStorageLibrary)

The path representing the current directory.

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

PathDotDot

NotFoundException

(AkiraNetwork.VirtualStorageLibrary)

Gets or sets the path representing the parent directory. The default value is "..".

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualNodeTypeFilter

```
public string PathDotDot { get; set; }
```

Property Value

| Type | Description |
|--|---|
| string (https://learn.microsoft.com/dotnet/api/system.string) | The path representing the parent directory. |

PathRoot

Gets or sets the root path. The default value is "/".

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
```

VirtualDirectory

```
public string PathRoot { get; set; }
```

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

Property Value

1.html

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

(AkiraNetwork.VirtualStorageLibrary)

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

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

2.html

PathSeparator

(AkiraNetwork.VirtualStorageLibrary)

Gets or sets the path separator character. The default value is '/'.

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

```
public char PathSeparator { get; set; }
```

1.html

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

Property Value

1.html

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

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

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

Gets or sets the prefix used for directory names. The default value is "dir". Used for auto-generating node names.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

NodeName

(AkiraNetwork.VirtualStorageLibrary)

```
public string PrefixDirectory { get; set; }
```

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Description |
|--|--------------------------------------|
| string (https://learn.microsoft.com/dotnet/api/system.string) | The prefix used for directory names. |

PrefixItem

Gets or sets the prefix used for item names. The default value is "item". Used for auto-generating node names.
 (AkiraNetwork.VirtualStorageLibrary)

Declaration

(AkiraNetwork.VirtualStorageLibrary)

```
public string PrefixItem { get; set; }
```

(AkiraNetwork.VirtualStorageLibrary)

Property Value

(AkiraNetwork.VirtualStorageLibrary)

Type

Description

VirtualDirectory
 (AkiraNetwork.VirtualStorageLibrary)

The prefix used for item names.

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

(AkiraNetwork.VirtualStorageLibrary)

1.html)

PrefixSymbolicLink

(AkiraNetwork.VirtualStorageLibrary)

Gets or sets the prefix used for symbolic link names. The default value is "link". Used for auto-generating node names.
 (AkiraNetwork.VirtualStorageLibrary)

2.html)

Declaration

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

```
public string PrefixSymbolicLink { get; set; }
```

(AkiraNetwork.VirtualStorageLibrary)

Property Value

(AkiraNetwork.VirtualStorageLibrary)

Type

Description

.html)

The prefix used for symbolic link names.

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

(AkiraNetwork.VirtualStorageLibrary)

1.html)

State

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

Gets the current instance of the state.
 (AkiraNetwork.VirtualStorageLibrary)

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

```
public static VirtualStorageState State { get; }
```

(AkiraNetwork.VirtualStorageLibrary)

Property Value

(AkiraNetwork.VirtualStorageLibrary)

Type

Description

VirtualNodeNotFoundException
 (AkiraNetwork.VirtualStorageLibrary)

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

WildcardMatcher

Gets or sets the wildcard matcher.

Declaration

```
public IVirtualWildcardMatcher? WildcardMatcher { get; set; }
```

| Property | Value | Description |
|-------------------------|--|-----------------------|
| Type | (AkiraNetwork.VirtualStorageLibrary) NotifyNodeDelegate | |
| IVirtualWildcardMatcher | (AkiraNetwork.VirtualStorageLibrary) (AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher.html) | The wildcard matcher. |

Methods

(AkiraNetwork.VirtualStorageLibrary) VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary) VirtualDirectoryAdapter<T>(1.html)

Set NodeListConditions(Virtual NodeListConditions)

Sets the conditions for listing nodes.

(AkiraNetwork.VirtualStorageLibrary) VirtualGroupCondition<T, TKey>

Declaration

(AkiraNetwork.VirtualStorageLibrary) VirtualException(2.html)

```
public static void Set NodeListConditions(Virtual NodeListConditions conditions)
```

(AkiraNetwork.VirtualStorageLibrary) VirtualItem

Parameters

(AkiraNetwork.VirtualStorageLibrary) VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary) Virtual NodeListConditions(1.html)

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

(AkiraNetwork.VirtualStorageLibrary) Virtual NodeListConditions(1.html)

(AkiraNetwork.VirtualStorageLibrary) Virtual NodeListConditions(1.html)

Set NodeListConditions(Virtual NodeTypeFilter,

(AkiraNetwork.VirtualStorageLibrary) Virtual GroupCondition<VirtualNode, object>?,

(AkiraNetwork.VirtualStorageLibrary) List<VirtualSortCondition<VirtualNode>>?)

(AkiraNetwork.VirtualStorageLibrary) Virtual NodeListConditions

Sets the conditions for listing nodes.

(AkiraNetwork.VirtualStorageLibrary) Virtual NodeListConditions

(AkiraNetwork.VirtualStorageLibrary) public static void Set NodeListConditions(Virtual NodeTypeFilter filter, Virtual GroupCondition

```
<VirtualNode, object>? groupCondition = null, List<VirtualSortCondition<VirtualNode>>? sortC
```

(AkiraNetwork.VirtualStorageLibrary) onditions = null)

(AkiraNetwork.VirtualStorageLibrary) VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary) Virtual NodeListConditions

Parameters

(AkiraNetwork.VirtualStorageLibrary) Virtual NodeType

(AkiraNetwork.VirtualStorageLibrary) Virtual NodeTypeFilter

| Type | Name | Description |
|---|-----------------------|-----------------------------------|
| VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | <i>filter</i> | The filter for node types. |
| VirtualGroupCondition (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition-2.html) <VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html), object (https://learn.microsoft.com/dotnet/api/system.object)> (AkiraNetwork.VirtualStorageLibrary) | <i>groupCondition</i> | The condition for grouping nodes. |
| NotifyNodeDelegate List (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1) (AkiraNetwork.VirtualStorageLibrary) < PatternMatch (AkiraNetwork.VirtualStorageLibrary) VirtualSortCondition (AkiraNetwork.VirtualStorageLibrary) <VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)>> VirtualDirectory (AkiraNetwork.VirtualStorageLibrary) VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary) | <i>sortConditions</i> | The conditions for sorting nodes. |

Extension Methods

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

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

Class VirtualStorage<T>

Manages virtual storage containing user-defined type T item data.

Filter by title

Inheritance

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

Syntax

VirtualID
(AkiraNetwork.VirtualStorageLibrary)
public class VirtualStorage<T>
 VirtualItem
(AkiraNetwork.VirtualStorageLibrary)

Type Parameters Adapter<T>

| Name | Description |
|------|--|
| T | The user-defined type within the virtual storage |

(AkiraNetwork.VirtualStorageLibrary)

1.html

Constructors

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

VirtualStorage()

VirtualNodeExtensions
Initializes a new instance of the VirtualStorage class.
(AkiraNetwork.VirtualStorageLibrary)

Declaration
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName
public VirtualStorage()
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Properties

CurrentPath

Gets or sets the current virtual path.

Declaration

```
public VirtualPath CurrentPath { get; }
```

Property Value
(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

Type (AkiraNetwork.VirtualStorageLibrary)

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)
(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

CycleDetectorForTarget

(AkiraNetwork.VirtualStorageLibrary)

Gets the cycle detector class used for the WalkPathToTarget method.

VirtualDirectoryAdapter<T>

Declaration

1.html)

VirtualException

```
public VirtualCycleDetector CycleDetectorForTarget { get; }
```

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

Property Value
(AkiraNetwork.VirtualStorageLibrary)

2.html)

VirtualID

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html)

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

CycleDetectorForTree

1.html)

VirtualItemAdapter<T>

Gets the cycle detector class used for the WalkPathTree method.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualNode

public VirtualCycleDetector CycleDetectorForTree { get; }

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

Property Value
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

Type (AkiraNetwork.VirtualStorageLibrary)

Description

VirtualNodeListConditions

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

Dir (AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

Gets the adapter providing operations for virtual directories.

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Description

An instance of the cycle detector class

Declaration

```
public VirtualDirectoryAdapter<T> Dir { get; }
```

Property Value

| Type | Description |
|---|--|
| VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T>).PatternMatch <T>PatternMatch | An instance of VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter -1.html) (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory this[VirtualPath, bool] (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory Gets or sets the Virtual Node corresponding to the specified virtual path. VirtualDirectoryAdapter<T> |

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T>).  
1.html)  
VirtualExceptionNode this[VirtualPath path, bool followLinks = true] { get; set; }
```

Parameters

(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>)

| Type | Name | Description |
|--|-------------|--|
| VirtualID VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) VirtualItem | path | The virtual path |
| bool VirtualItemAdapter<T>(dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) 1.html) | followLinks | Flag indicating whether to follow symbolic links |

Property value

(AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>)

| Type | Description |
|--|--|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) VirtualNodeContext | The virtual node corresponding to the specified virtual path |

Item

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName)

VirtualNodeListConditions

Gets the adapter providing operations for virtual items.

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName).  
public VirtualItemAdapter<T> Item { get; }
```

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName)

VirtualNodeType

Property Value

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter)

| Type | Description |
|--|---|
| VirtualItemAdapter (AkiraNetwork.VirtualStorageLibrary.VirtualItemA dapter-1.html) <code><T></code> | An instance of VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter- 1.html) |

(AkiraNetwork.VirtualStorageLibrary)

LinkNotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary) is the adapter providing operations for PatternMatch

Declaring Akira Network Virtual Storage Library

VirtualCycleDetector

```
public VirtualStorageLibrary<T> Link { get; }  
VirtualDirector;
```

VirtualDirectory (Ali: N : L)

Property Value : [Add](#) [Edit](#)

Property Value

| Type | Description |
|--|--|
| (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary-1.html) | |
| VirtualSymbolicLinkAdapter | An instance of |
| (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary-1.html) | VirtualSymbolicLinkAdapter<T> |
| VirtualGroupCondition<T, TKey> | (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLinkAda |
| <T>(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary-1.html) | pter-1.html) |

2.html)

VirtualID

LinkDictionary

VirtualItem

Gets (Akira Network Virtual Storage Library)

VirtualItemAdapter

Declaration

1.html

```
public IDictionary<VirtualPath, HashSet<VirtualPath>> LinkDictionary { get; }
```

(AkiraNetwork VirtualStorage) library

(AkiraN)

| Property | Type | Description |
|----------------------|---|--|
| Value VirtualNode | (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext Dictionary (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2) <VirtualNodeExtensions <VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html), (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)> HashSet (https://learn.microsoft.com/dotnet/api/system.collections.generic.hashset-1)< VirtualNodeListConditions <VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)>> (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)) | A dictionary representing the link mappings. |

VirtualNodeName

(AkiraNetwork VirtualStorage) library

(A)
Book

(AkiraNetworks VirtualStorageLibrary)

(AkiraNetwork.virtualStorageLibrary)
Gets the root directory of the virtual storage.

VirtualNodeTyp

Declaration (Akira N)

VirtualNodeTypeFilter

```
public VirtualDirectory Root { get; }
```

Property Value

| Type | Description |
|--|---|
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) | The root directory of the virtual storage |

Methods

| |
|---|
| NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| PatternMatch (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |

AddDirectory(VirtualPath, VirtualDirectory, bool)

| |
|---|
| Adds a directory (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| Declaration 1.html |

```
public void AddDirectory(VirtualPath directoryPath, VirtualDirectory directory, bool createsSubdirectories = false)
    VirtualGroupCondition<T, TKey>
        (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html)
```

Parameters

| Type | Name | Description |
|--|----------------------|---|
| VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) | directoryPath | The path of the directory to add |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) | directory | The VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.VirtualDirectory.html) to add |
| bool (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) | createSubdirectories | Indicates whether to create intermediate directories |

AddDirectory(VirtualPath, bool)

| |
|---|
| Adds a directory (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html) |
| Declaration 1.html |

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

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

```
public void AddDirectory(VirtualPath path, bool createSubdirectories = false)
    (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html)
```

VirtualNodeTypeFilter

Parameters

| Type | Name | Description |
|---|-----------------------------|--|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>path</i> | The path of the directory to add |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | <i>createSubdirectories</i> | Indicates whether to create intermediate directories |

AddItem(VirtualPath, VirtualItem<T>, bool)

Declaration
(AkiraNetwork.VirtualStorageLibrary.AddItem)

```
public void AddItem(VirtualPath itemDirectoryPath, VirtualItem<T> item, bool overwrite = false)
```

Parameters

| Type | Name | Description |
|---|--------------------------|---|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>itemDirectoryPath</i> | The path of the directory to add the item to |
| VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>.html) | <i>item</i> | The VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>.html) to add |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | <i>overwrite</i> | Indicates whether to overwrite an existing node |

AddItem(VirtualPath, T?, bool)

Declaration
(AkiraNetwork.VirtualStorageLibrary.AddItem)

```
public void AddItem(VirtualPath itemPath, T? data = default, bool overwrite = false)
```

Parameters

| Type | Name | Description |
|---|------|-------------|
| VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter) | | |

| Type | Name | Description |
|---|------------------|---|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>itemPath</i> | The path of the item to add |
| T | <i>data</i> | The data for the item |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary) | <i>overwrite</i> | Indicates whether to overwrite an existing node |

AddLinkToDictionary(VirtualPath, VirtualPath)

(AkiraNetwork.VirtualStorageLibrary)
Adds a new link to the link dictionary.
VirtualCycleDetector

VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

```
public void AddLinkToDictionary(VirtualPath targetPath, VirtualPath linkPath)  
    VirtualDirectoryAdapter<T>
```

(AkiraNetwork.VirtualStorageLibrary)

Parameters

| Type | Description | Name |
|--|------------------------------|------------|
| VirtualException (AkiraNetwork.VirtualStorageLibrary) | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | The target path of the link. | targetPath |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) ? html | The path of the link. | linkPath |

Virtual
Exceptions

| Type | Condition |
|---|--|
| VirtualItem (AkiraNetwork.VirtualStorageLibrary) | |
| ArgumentException VirtualItemAdapter<T> (https://learn.microsoft.com/dotnet/api/system.argumentexception) (AkiraNetwork.VirtualStorageLibrary) | Thrown if the target path is not an absolute path. |

AddNode(VirtualPath, VirtualNode, bool)

1.html)

Adds a virtual node to the specified directory.
D:\> (AkiraNetwork.VirtualStorageLibrary)

Declaration VirtualNodeContext

```
(AkiraNetwork.VirtualStorageLibrary  
public void AddNode(VirtualPath  
    public void AddNode(VirtualPath
```

VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary)

Parameters

Type virtualnodeName

--VirtualNodeName
VirtualNetwork.VirtualStorageLibrary

(AkilonNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException)

VirtualPathName
(AkiraNetwork.VirtualStorageLibrary)

VirtualNode type

VirtualNodeTypeFilter

| Type | Name | Description |
|---|--------------------------|--|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName) | | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath) | <i>nodeDirectoryPath</i> | The path of the directory to add the node to |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNotFoundException) | | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPathItem) | | |
| VirtualNodeType | | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter) | | |

| Type | Name | Description |
|---|------------------|--|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | <i>node</i> | The VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) to add |
| bool (AkiraNetwork.VirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/ NotifyNodeDelegate system.boolean) (AKIRANetwork.VirtualStorageLibrary | <i>overwrite</i> | Indicates whether to overwrite an existing node |

AddSymbolicLink(VirtualPath, VirtualPath?, bool)
(AkiraNetwork.VirtualStorageLibrary)

Adds a virtual directory.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
public void AddSymbolicLink(VirtualPath linkPath, VirtualPath? targetPath = null, bool overwrite = false)
    throws VirtualException
```

Virtual Parameters

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|--|-------------------|---|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>linkPath</i> | The path of the symbolic link |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>targetPath</i> | The path of the target |
| bool (AkiraNetwork.VirtualStorageLibrary.VirtualItem<T> (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>) | <i>overwrite</i> | Indicates whether to overwrite an existing node |

AddSymbolicLink(VirtualPath, VirtualSymbolicLink, bool)

VirtualNodeContext

Adds a symbolic link to the AkraNetwork virtual storage library.

- VirtualNodeExtensions

```
claration NOGCEXTENSIONS  
(AkiraNetwork.VirtualStorageLibrary  
VirtualNodeListConditions  
public void AddSymbolicLink(Virt  
(AkiraNetwork.VirtualStorageLibrary  
erwrite = false)  
VirtualNodeName
```

VirtualNodeName

(AkilaNetwork.VirtualStorageLibrary)

```
VirtualNodeNotFoundException  
(AkiraNetwork.VirtualStorageLibrary)  
VirtualNodeType  
(AkiraNetwork.VirtualStorageLibrary)  
VirtualNodeTypeFilter
```

| Type | Name | Description |
|--|--------------------------|--|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>linkDirectoryPath</i> | The path of the directory to add the symbolic link to |
| VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) | <i>link</i> | The VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) to add |
| bool (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector System.Boolean) | <i>overwrite</i> | Indicates whether to overwrite an existing node |

ChangeDirectory(VirtualPath)

(AkiraNetwork.VirtualStorageLibrary.ChangeDirectory.html)
Changes the current working directory.

Declaration

```
VirtualException
(AkiraNetwork.VirtualStorageLibrary.VirtualException.html)
VirtualGroupCondition<T, TKey>
public void ChangeDirectory(VirtualPath path)
(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)
2.html)
```

Parameters

| Type | Name | Description |
|--|-------------|---------------------------------------|
| VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>path</i> | The path of the new working directory |

ConvertToAbsolutePath(VirtualPath?, VirtualPath?)

(AkiraNetwork.VirtualStorageLibrary.ConvertToAbsolutePath.html)

Converts a relative path to an absolute path.

Declaration

```
VirtualNode
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)
VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html)
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html)
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html)
```

Parameters

| Type | Name | Description |
|--|---------------------|--------------------|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>relativePath</i> | The relative path. |

VirtualPath
(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html)

VirtualNodeTypeFilter
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html)

Returns

| Type | Description |
|--|---|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | The absolute path represented by VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html). |

Exceptions

(AkiraNetwork.VirtualStorageLibrary

| Type | Condition |
|--|--|
| NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary.ArgumentException PatternMatch (https://learn.microsoft.com/dotnet/api/system.argumentexception) (AkiraNetwork.VirtualStorageLibrary) | Thrown if relativePath is null or an empty string. |

CopyNodes(VirtualPath, VirtualPath, bool, bool, bool, List<VirtualNodeContext>?)

(AkiraNetwork VirtualStorage) library

(AkiraNetwork.VirtualStorageLIB) Copies the node at the specified path.

Copies the file at the specified path.

© AkiraNetwork Virtual Storage Library

Declaration

1.11(11)

```
VirtualException
public void CopyNode(VirtualPath sourcePath, VirtualPath destinationPath, bool overwrite = false,
(AkiraNetwork.VirtualStorageLibrary)
false, bool recursive = false, bool followLinks = false, List<VirtualNodeContext>? destinationContextList = null)
(AkiraNetwork.VirtualStorageLibrary)
```

(AkiraNetwork.VirtualStorageLibrary)

2.111111111111111

| Type | Name | Description |
|--|------------------------|--------------------------------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualItem) | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | sourcePath | The source path of the copy. |
| VirtualItemAdapter<T> | | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | destinationPath | The destination path of the copy. |
| VirtualItem<T> | | |
| bool (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | overwrite | Whether to overwrite existing nodes. |
| VirtualNode | | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | recursive | Whether to copy recursively. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext) | | |
| VirtualNodeExtensions | | |
| bool (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html) | followLinks | Whether to follow symbolic links. |
| List<VirtualNodeListConditions> | | |
| (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1) | destinationContextList | List of destination contexts. |
| VirtualNodeName | | |
| < VirtualNodeContext | | |
| VirtualNodeNotFoundException | | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html) | | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType) | | |
| > VirtualNodeType | | |
| (AkiraNetwork.VirtualStorageLibrary.Exceptions) | | |
| VirtualNodeTypeFilter | | |

| Type | Condition |
|--|--|
| VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html) | Thrown if the source node is not found. |
| InvalidOperationException (https://learn.microsoft.com/dotnet/api/system.invalidoperationexception) AKIRANETWORK.VirtualStorageLibrary | Thrown if the source and destination paths are the same or if a recursive copy would result in a circular reference. |

DirectoryExists(VirtualPath, bool)

(AkiraNetwork.VirtualStorageLibrary)

Check if a directory exists at the specified path.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

virtual bool DirectoryExists(VirtualPath path, bool followLinks = false)

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Parameters

VirtualException

| Type | Name | Description |
|---|-------------|-----------------------------------|
| AkiraNetwork.VirtualStorageLibrary | path | The directory's path. |
| VirtualGroupCondition<T, TKey> VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary) | followLinks | Whether to follow symbolic links. |

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

| Type | Description |
|--|---|
| VirtualItem<T> bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary) | True if the directory exists; otherwise, false. |

1.html)

VirtualItem<T>

ExpandPath(VirtualPath, VirtualNodeTypeFilter, bool, bool)

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Expands a path that contains wildcards.

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

public IEnumerable<VirtualPath> ExpandPath(VirtualPath path, VirtualNodeTypeFilter filter = VirtualNodeExtensions.All, bool followLinks = true, bool resolveLinks = true)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

Parameters

VirtualNodeName

| Type | Name | Description |
|--|--------|-----------------------|
| AkiraNetwork.VirtualStorageLibrary | path | The path. |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html) | filter | The node type filter. |

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Name | Description |
|---|---------------------|------------------------------------|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | <i>followLinks</i> | Whether to follow symbolic links. |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) ▼ | <i>resolveLinks</i> | Whether to resolve symbolic links. |

Retu(AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|---|---|
| NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary) | |
| IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) | The expanded paths IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) |
| < (AkiraNetwork.VirtualStorageLibrary) | . |
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | . |
| > VirtualDirectoryAdapter<T> | . |

Report.html

VirtualException indicates whether to follow links during the recursive traversal of nodes when the terminal node of the path is a symbolic link.

VirtualGroupCondition<T, TKey>

`resolver.links` indicates whether to resolve links when non-terminal nodes of the path are symbolic links.

2.htm|)

VirtualID

ExpandPathTree(VirtualPath, VirtualNodeTypeFilter, bool, bool)
(AkiraNetwork.VirtualStorageLibrary)

`VirtualItem` expands the virtual path context.

Expands the virtual path containing wildcards.
(AkiraNetwork VirtualStorageLibrary)

Declaration: ItemAdapter<T>

DeclarationItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

public **virtual** **IEnumerable**<**VirtualNodeContext**> ExpandPathTree(**VirtualPath** path, **VirtualNodeTypeFilter** filter)

```
rVirtualItem.VirtualNodeTypeFilter.All, bool followLinks = true, bool resolveLinks = true)
```

(AkiraNetwork.VirtualStorageLibrary)

1.html)
parameters

Parameters

| Type | Name | Description |
|---|--------------|---|
| VirtualNodeContext VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) VirtualNodeExtensions | path | The virtual path containing wildcards |
| VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | filter | The filter to apply for node types |
| VirtualNodeName bool (AkiraNetwork.VirtualStorageLibrary. https://learn.microsoft.com/dotnet/api/system.boolean) VirtualNotFoundException | followLinks | A flag indicating whether to follow symbolic links |
| bool (AkiraNetwork.VirtualStorageLibrary. https://learn.microsoft.com/dotnet/api/system.boolean) VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter) | resolveLinks | A flag indicating whether to resolve symbolic links |

Returns

| Type | Description |
|--|--|
| IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) < VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html) > (AkiraNetwork.VirtualStorageLibrary.PatternMatch (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector followLinks indicates whether to follow links during the recursive traversal process when the terminal node in the virtual path is a symbolic link. VirtualDirectory resolves links when nodes other than the terminal node in the virtual path are symbolic links. VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T> 1.html) VirtualPath Gets the directory at the specified path. VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey> 2.html) VirtualID public VirtualDirectory GetDirectory(VirtualPath path, bool followLinks = false) (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) VirtualItem VirtualItem<T> bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary.VirtualItem<T> 1.html) | An enumerable of VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html) containing information about the expanded nodes |

Remarks

VirtualCycleDetector
followLinks indicates whether to follow links during the recursive traversal process when the terminal node in the virtual path is a symbolic link.
VirtualDirectory
resolves links when nodes other than the terminal node in the virtual path are symbolic links.
VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>
1.html)

GetDirectory(VirtualPath, bool)

(AkiraNetwork.VirtualStorageLibrary.VirtualPath
Gets the directory at the specified path.
VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey>

Declaration

2.html)

VirtualID
public VirtualDirectory GetDirectory(VirtualPath path, bool followLinks = false)
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html)

VirtualItem

Parameters

(AkiraNetwork.VirtualStorageLibrary.VirtualPath
VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>

VirtualPath (<https://learn.microsoft.com/dotnet/api/system.uri>)
path The path of the directory.

VirtualItem<T>
bool (<https://learn.microsoft.com/dotnet/api/system.boolean>)
(AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>
1.html)

Returns

VirtualNode

| Type | Description |
|---|---|
| AkiraNetwork.VirtualStorageLibrary.VirtualPath VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html) VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html) | The VirtualDirectory at the specified path. |

Exceptions

VirtualNodeName

| Type | Condition |
|---|-------------------------------------|
| AkiraNetwork.VirtualStorageLibrary.VirtualNodeName VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html) VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html) VirtualNodeTypeError (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | Thrown if the node cannot be found. |

GetItem(VirtualPath, bool)

Gets the item at the specified path.

Declaration

```
public VirtualItem<T> GetItem(VirtualPath path, bool followLinks = false)
```

Parameters

| | | | |
|-------------|---|-------------|-----------------------------------|
| Type | (AkiraNetwork.VirtualStorageLibrary) | Name | Description |
| VirtualPath | (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | The path of the item. |
| bool | (https://learn.microsoft.com/dotnet/api/system.boolean) | followLinks | Whether to follow symbolic links. |
| | (AkiraNetwork.VirtualStorageLibrary) | | |

Returns

| | | |
|---|---|--|
| Type | (AkiraNetwork.VirtualStorageLibrary) | Description |
| VirtualDirectory | (VirtualDirectoryAdapter<T>) | |
| VirtualItem<T> | (AkiraNetwork.VirtualStorageLibrary) | The |
| VirtualException | (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) | VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) |
| <T>(AkiraNetwork.VirtualStorageLibrary) | | at the specified path. |
| | (VirtualGroupCondition<T, TKey>) | |

Exceptions

| | | |
|--|---------------|-------------------------------------|
| Type | VirtualID | Condition |
| VirtualNotFoundException | (VirtualItem) | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html) | | Thrown if the node cannot be found. |
| (AkiraNetwork.VirtualStorageLibrary) | | |

GetLinksFromDictionary(VirtualPath)

Declaration

```
public HashSet<VirtualPath> GetLinksFromDictionary(VirtualPath targetPath)
```

Retrieves all link paths associated with the specified target path from the link dictionary.

(AkiraNetwork.VirtualStorageLibrary)

| | |
|--------------------------------------|--------------------------------------|
| Type | VirtualNode |
| VirtualNodeContext | (AkiraNetwork.VirtualStorageLibrary) |
| (AkiraNetwork.VirtualStorageLibrary) | |
| | (VirtualNodeAdapter<T>) |

Parameters

| | | | |
|--------------------------------------|--------------------------------------|------|------------------|
| Type | (AkiraNetwork.VirtualStorageLibrary) | Name | Description |
| VirtualNodeListConditions | (VirtualPath) | | |
| (AkiraNetwork.VirtualStorageLibrary) | targetPath | | The target path. |

Parameters

| | | |
|--------------------------------------|--|----------------------|
| Type | VirtualNodeName | Description |
| VirtualNodeNotFoundException | (VirtualPath) | |
| (AkiraNetwork.VirtualStorageLibrary) | | A set of link paths. |
| HashSet<VirtualPath> | (https://learn.microsoft.com/dotnet/api/system.collections.generic.hashset-1)<VirtualPath>(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)> | |
| VirtualNodeTypeFilter | (VirtualPath) | |

GetNode(VirtualPath, bool)

Gets the node at the specified path.

Declaration

```
public VirtualNode GetNode(VirtualPath path, bool followLinks = false)
```

Parameters

NotifyNodeDelegate

| Type | Name | Description |
|---|------|------------------|
| (AkiraNetwork.VirtualStorageLibrary) | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary) | path | The node's path. |

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

(AkiraNetwork.VirtualStorageLibrary)

Returns

VirtualDirectory

| Type | Description |
|---|--|
| (AkiraNetwork.VirtualStorageLibrary) | |
| VirtualDirectoryAdapter<T> | |
| VirtualNode (AkiraNetwork.VirtualStorageLibrary) | The |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNode.VIRTUALException) | VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) |
| (AkiraNetwork.VirtualStorageLibrary) | at the specified path. |
| VirtualGroupCondition<T, TKey> | |

Exceptions

2.html)

| Type | Condition |
|---|-------------------------------------|
| VirtualID (AkiraNetwork.VirtualStorageLibrary) | |
| VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary) | Thrown if the node cannot be found. |

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem (AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary)

GetNodeType(VirtualPath, bool)

VirtualItem<T>

Gets the node type at the specified path.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualNode

| Type | Condition |
|---|-----------|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary) | |
| VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary) | |

(AkiraNetwork.VirtualStorageLibrary)

Parameters

VirtualNodeExtensions

| Type | Name | Description |
|---|------|------------------|
| (AkiraNetwork.VirtualStorageLibrary) | | |
| VirtualNodeListConditions | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | The node's path. |

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName (AkiraNetwork.VirtualStorageLibrary)

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

(AkiraNetwork.VirtualStorageLibrary)

Returns

VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType (AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|--|---|
| VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNode Type.html) | The VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html) of the node. |

(AkiraNetwork.VirtualStorageLibrary)

GetNodes(VirtualNodeTypeFilter, bool, bool)

NodeNodeDelegate

(AkiraNetwork.VirtualStorageLibrary)
Gets the nodes in the path tree starting from the current path.
PatternMatch

Declaration

VirtualCycleDetector

```
public I Enumerable<VirtualNode> GetNodes(VirtualNodeTypeFilter nodeType = VirtualNodeTypeFil  
ter.All, bool recursive = false, bool followLinks = false)
```

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

Parameters

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|--|-------------|--|
| VirtualException | | |
| VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary) | nodeType | The node type filter. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) 2.html) | recursive | Whether to retrieve nodes recursively. |
| VirtualID | | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | followLinks | Whether to follow symbolic links. |
| VirtualItem (AkiraNetwork.VirtualStorageLibrary) | | |

Return Value

VirtualItemAdapter<T>

| Type | Description |
|--|--|
| IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collection.generic.ienumerable-1) < VirtualNode VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) > VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary) | The nodes in the path tree starting from the current path IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collection.generic.ienumerable-1) . . |

Remarks

This method may be integrated, reorganized, or deprecated in the near future.

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

GetNodes(VirtualPath, VirtualNodeTypeFilter, bool, bool)

(AkiraNetwork.VirtualStorageLibrary)

Gets the nodes in the path tree starting from the specified path.

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

```
public IEnumerable<VirtualNode> GetNodes(VirtualPath basePath, VirtualNodeTypeFilter nodeType)
e = VirtualNodeTypeFilter.All, bool recursive = false, bool followLinks = false)
```

Parameters

| Type | Name | Description |
|---|-------------|--|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | basePath | The base path. |
| NotifyNodeDelegate VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) PatternMatch | nodeType | The node type filter. |
| bool (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector VirtualCycleDetector) | recursive | Whether to retrieve nodes recursively. |
| bool (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory VirtualDirectory) (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T>) | followLinks | Whether to follow symbolic links. |

Returns

| Type | Description |
|---|--|
| VirtualException IEnumerable<VirtualNode> (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) < 2.html> VirtualNode VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)> VirtualItem > (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T> VirtualItemAdapter<T>) | The nodes in the path tree starting from the specified path IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) |

Remarks

(AkiraNetwork.VirtualStorageLibrary)

This method may be integrated, reorganized, or deprecated in the near future.

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

GetNodesWithPaths(VirtualNodeTypeFilter, bool, bool)

VirtualNode

Gets the paths in the path tree starting from the current path.
(AkiraNetwork.VirtualStorageLibrary)

Declaration

(AkiraNetwork.VirtualStorageLibrary)

```
public IEnumerable<VirtualPath> GetNodesWithPaths(VirtualNodeTypeFilter nodeType = VirtualNo
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.All, bool recursive = false, bool followLinks = false)
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary)
```

Parameters

| Type | Name | Description |
|---|----------|-----------------------|
| VirtualNodeNotFoundException VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) VirtualNodeType | nodeType | The node type filter. |

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Name | Description |
|---|--------------------|--|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | <i>recursive</i> | Whether to retrieve paths recursively. |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) ▼ | <i>followLinks</i> | Whether to follow symbolic links. |

Returns [AkiraNetwork.VirtualStorageLibrary](#)

| Type | Description |
|--|---|
| NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary) | |
| IEnumerable (AkiraNetwork.VirtualStorageLibrary) | The paths in the path tree starting from the current path |
| VirtualPathMatch (AkiraNetwork.VirtualStorageLibrary) | IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collection.generic.ienumerable-1) |
| < (AkiraNetwork.VirtualStorageLibrary) | |
| VirtualPathDirectory (AkiraNetwork.VirtualStorageLibrary) | |
| > VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary) | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary) | . |

Remarks

This method may be integrated, reorganized, or deprecated in the near future.
([AkiraNetwork.VirtualStorageLibrary](#))

VirtualException
([AkiraNetwork.VirtualStorageLibrary](#))

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

GetNodesWithPaths(VirtualPath, VirtualNodeTypeFilter, bool, bool)

2.html

Gets the paths in the path tree starting from the specified path.

VirtualPath
([AkiraNetwork.VirtualStorageLibrary](#))

VirtualItem
([AkiraNetwork.VirtualStorageLibrary](#))

```
public IEnumerable<VirtualPath> GetNodesWithPaths(VirtualPath basePath, VirtualNodeTypeFilter nodeType = VirtualNodeTypeFilter.All, bool recursive = false, bool followLinks = false)
(AkiraNetwork.VirtualStorageLibrary)
1.html
```

Parameters

| Type | Name | Description |
|--|--------------------|--|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary) | <i>basePath</i> | The base path. |
| VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary) | <i>nodeType</i> | The node type filter. |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | <i>recursive</i> | Whether to retrieve paths recursively. |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | <i>followLinks</i> | Whether to follow symbolic links. |

Returns [VirtualNodeNotFoundException](#)

([AkiraNetwork.VirtualStorageLibrary](#))

VirtualNodeType
([AkiraNetwork.VirtualStorageLibrary](#))

VirtualNodeTypeFilter
([AkiraNetwork.VirtualStorageLibrary](#))

VirtualNodeName
([AkiraNetwork.VirtualStorageLibrary](#))

| Type | Description |
|---|-------------|
| IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) < VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) > (AkiraNetwork.VirtualStorageLibrary. NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary. Remarks PatternMatch This method may be integrated, reorganized, or deprecated in the near future. (AkiraNetwork.VirtualStorageLibrary. VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary. VirtualDirectory (AkiraNetwork.VirtualStorageLibrary. Gets (https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualPath). VirtualDirectoryAdapter<T> Declaration (AkiraNetwork.VirtualStorageLibrary. 1.html) public VirtualSymbolicLink GetSymbolicLink(VirtualPath path) VirtualException (AkiraNetwork.VirtualStorageLibrary. VirtualGroupCondition<T, TKey> Parameters (AkiraNetwork.VirtualStorageLibrary. Type (https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualPath) VirtualID VirtualPath (https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary. VirtualItem Returns (AkiraNetwork.VirtualStorageLibrary. Type VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary. VirtualSymbolicLink 1.html) (AkiraNetwork.VirtualStorageLibrary.VirtualSy mbolicLink.html) (AkiraNetwork.VirtualStorageLibrary. 1.html) VirtualNode Remarks (AkiraNetwork.VirtualStorageLibrary. This method does not have a followLinks parameter because it retrieves the symbolic link itself. VirtualNodeContext Exceptions (AkiraNetwork.VirtualStorageLibrary. VirtualNodeExtensions Type (AkiraNetwork.VirtualStorageLibrary. VirtualNodeListConditions VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html) VirtualnodeName (AkiraNetwork.VirtualStorageLibrary. VirtualNodeNotFoundException VirtualNodeType ItemExists(VirtualPath, bool) Checks if an item exists at the specified path. (AkiraNetwork.VirtualStorageLibrary. VirtualNodeTypeFilter Declaration (AkiraNetwork.VirtualStorageLibrary. Description The paths in the path tree starting from the specified path IEnumeration<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) . . | |

GetSymbolicLink(VirtualPath)

Gets (<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualPath>).

```
VirtualDirectoryAdapter<T>
public VirtualSymbolicLink GetSymbolicLink(VirtualPath path)
{
    throw new VirtualException();
}
```

Parameters

(AkiraNetwork.VirtualStorageLibrary.
Type
(<https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualPath>)

| Type | Name | Description |
|---|------|--------------------------------|
| VirtualPath (https://learn.microsoft.com/dotnet/api/AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | The path of the symbolic link. |

Returns

(AkiraNetwork.VirtualStorageLibrary.
Type
VirtualItemAdapter<T>

| Type | Description |
|---|---|
| VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary. VirtualSymbolicLink 1.html) (AkiraNetwork.VirtualStorageLibrary.VirtualSy mbolicLink.html) (AkiraNetwork.VirtualStorageLibrary. 1.html) | The VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) at the specified path. |

Exceptions

(AkiraNetwork.VirtualStorageLibrary.
Type
VirtualNodeExtensions
VirtualNodeListConditions
VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html)
VirtualnodeName
(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeNotFoundException
VirtualNodeType
ItemExists(VirtualPath, bool)
Checks if an item exists at the specified path.
(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeTypeFilter
Declaration
(AkiraNetwork.VirtualStorageLibrary.
Description
The

| Type | Condition |
|--|-------------------------------------|
| VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html) | Thrown if the node cannot be found. |

ItemExists(VirtualPath, bool)

Checks if an item exists at the specified path.
(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeTypeFilter
Declaration
(AkiraNetwork.VirtualStorageLibrary.
Description
The

```
public bool ItemExists(VirtualPath path, bool followLinks = false)
```

Parameters

| Type | Name | Description |
|--|-------------|-----------------------------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | The item's path. |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | followLinks | Whether to follow symbolic links. |

Returns

| Type | Description |
|----------------------|--|
| VirtualCycleDetector | True if the item exists; otherwise, false. |

MoveNode(VirtualPath, VirtualPath, bool, bool)

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

Moves the node at the specified path to a new location.

VirtualException

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    public void MoveNode(VirtualPath sourcePath, VirtualPath destinationPath, bool overwrite = false, bool resolveLinks = true)
```

VirtualID

Parameters

| Type | Name | Description |
|-----------------------|-----------------|--------------------------------------|
| VirtualItem | | |
| VirtualPath | sourcePath | The source path of the node. |
| VirtualItemAdapter<T> | | |
| VirtualPath | destinationPath | The destination path for the node. |
| VirtualItem<T> | | |
| bool | overwrite | Whether to overwrite existing nodes. |
| VirtualNode | | |
| bool | resolveLinks | Whether to resolve symbolic links. |

VirtualNodeExtensions

Remarks

resolves symbolic links to resolve links when non-terminal nodes (from the top node to the parent of the terminal node) have symbolic links.

VirtualNodeName

Exceptions

| Type | Condition |
|------------------------------|---|
| VirtualNodeNotFoundException | Thrown if the source and destination paths are the same, if attempting to move the root directory, or if the destination is a subdirectory of the source. |

| Type | Condition |
|--|---|
| VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html) | Thrown if the source node is not found. |

二

NodeExists(VirtualPath, bool)

NotifyNodeDelegate
Checks if a node exists at the specified path.
(AkiraNetwork.VirtualStorageLibrary)

Declaration Pattern Match

(AkiraNetwork.VirtualStorageLibrary)

```
public bool NodeExists(VirtualPath path, bool followLinks = false)  
(AkiraNetwork.VirtualStorageLibrar
```

VirtualDirectory

Parameters

- VirtualDirectoryAdapter<T>

| Type | Description | Name | |
|--|-----------------------------------|-------------|--|
| VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary) | | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) .html | The node's path. | path | |
| VirtualException bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary) | Whether to follow symbolic links. | followLinks | |

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

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

VirtualItem

(AkiraNetwork.VirtualStorageLibrary

RemoveLinkByLinkPath(VirtualPath)

(AkiraNetwork.VirtualStorageLibrary

RemoveLink(*linkPath*)
Removes the link with the specified link path from the link dictionary.

B. Virtual

Declaration

1.html

```
public void RemoveLinkByLinkPath(VirtualPath linkPath)
```

(AkiraNetwork VirtualStorage) library

Basic VirtualN

Parameters

| Type | (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions) | Name | Description |
|------|--|-----------------|----------------|
| Virt | (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>linkPath</i> | The link path. |

virtual

(AkiraNetwork.Virt eptions

| Type | Condition |
|--|--|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary) | |
| VirtualNodeNotFoundException ArgumentException (AkiraNetwork.VirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.argumentexception)) | Thrown if the link path is not an absolute path. |
| VirtualNodeType (AkiraNetwork.VirtualStorageLibrary) | |
| VirtualNodeTypeFilter | |

RemoveLinkFromDictionary(VirtualPath, VirtualPath)

Removes a link from the link dictionary.

Declaration

```
public void RemoveLinkFromDictionary(VirtualPath targetPath, VirtualPath linkPath)
```

Parameters

NotifyNodeDelegate

Type (AkiraNetwork.VirtualStorageLibrary)

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

(AkiraNetwork.VirtualStorageLibrary)

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

RemoveNode(VirtualPath, bool, bool, bool)

Removes the node at the specified path.

1.html)

Declaration

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T> VirtualPath nodePath, bool recursive = false, bool followLinks = false

(AkiraNetwork.VirtualStorageLibrary)

2.html)

VirtualID

Parameters

(AkiraNetwork.VirtualStorageLibrary)

Type virtualItem

Name

Description

(AkiraNetwork.VirtualStorageLibrary)

nodePath

The path of the node to remove.

VirtualPath

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

(AkiraNetwork.VirtualStorageLibrary)

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

recursive

Whether to remove recursively.

VirtualItem<T>

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

(AkiraNetwork.VirtualStorageLibrary)

1.html)

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

followLinks

Whether to follow symbolic links.

(AkiraNetwork.VirtualStorageLibrary)

resolveLinks

Whether to resolve link targets.

(AkiraNetwork.VirtualStorageLibrary)

Remarks

If followLinks is true and the terminal node in the path is a symbolic link, the path tree after resolving the path will be removed, and the symbolic link will also be removed. If false, only the symbolic link itself is removed.

(AkiraNetwork.VirtualStorageLibrary)

resolveLinks indicates whether to resolve links when non-terminal nodes (from the top node to the parent of the terminal node) in the path are symbolic links.

(AkiraNetwork.VirtualStorageLibrary)

Exceptions

(AkiraNetwork.VirtualStorageLibrary)

Type VirtualNodeNotFoundException

Condition

(AkiraNetwork.VirtualStorageLibrary)

Thrown when trying to remove the root node

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

or a non-empty directory.

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

ResolveLinkTarget(VirtualPath)

Resolves the target of the symbolic link at the specified path.

Declaration

```
public VirtualPath ResolveLinkTarget(VirtualPath path)
```

Parameters

NotifyNodeDelegate

Type (AkiraNetwork.VirtualStorageLibrary)

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

Returns (AkiraNetwork.VirtualStorageLibrary)

Type VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

Exceptions

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNotFoundException

VirtualGroupCondition<T, Key>

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNotFoundException.html

2.html)

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

SetLinkTargetNodeType(HashSet<VirtualPath>, VirtualNodeType)

Sets (AkiraNetwork.VirtualStorageLibrary) symbolic links in the specified list of link paths.

VirtualItemAdapter<T>

Declaration (AkiraNetwork.VirtualStorageLibrary)

1.html)

```
public void SetLinkTargetNodeType(HashSet<VirtualPath> linkPathSet, VirtualNodeType nodeTyp
```

```
e)
```

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Parameters

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

Type VirtualNodeContext

HashSet<VirtualPath>

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

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

VirtualNodeListConditions

VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html)

(AkiraNetwork.VirtualStorageLibrary)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

SetNode(VirtualPath, VirtualNode)

VirtualNodeType

Sets (AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Declaration

```
public void SetNode(VirtualPath destinationPath, VirtualNode node)
```

Parameters

| Type | Name | Description |
|---|-----------------|---|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | destinationPath | The path where the node is set |
| PatternMatch VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html) VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter.html) | node | The VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) to set |

SetnodeName(VirtualPath, VirtualNodeName, bool)

VirtualException

Changes the name of a node.

(AkiraNetwork.VirtualStorageLibrary.VirtualException.html)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition.html)

2.html)

```
public void SetnodeName(VirtualPath nodePath, VirtualNodeName newName, bool resolveLinks = true)
```

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter.html)

VirtualPath

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

bool

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

(http://schemas.microsoft.com/dotnet/api/system.boolean)

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html)

Checks if a symbolic link exists at the specified path.

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

Declaration

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html)

public bool SymbolicLinkExists(VirtualPath path)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html)

Parameters

VirtualNodeTypeFilter

| Type | Name | Description |
|---|------|---------------------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | The symbolic link's path. |

Returns

| Type | Description |
|--|---|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions.TryGetDirectory) | True if the symbolic link exists; otherwise, false. |

TryGetDirectory(VirtualPath, bool)

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

Tries to get the directory at the specified path.

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

```
public VirtualDirectory? TryGetDirectory(VirtualPath path, bool followLinks = false)
```

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

Parameters

| Type | Name | Description |
|--|------|----------------------------|
| VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions) | | |
| Virtu VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | The path of the directory. |

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

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

| Type | Description |
|---|---|
| VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions) | The found VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) , or null if not found. |

TryGetItem(VirtualPath, bool)

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

Tries to get the item at the specified path.

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

```
public VirtualItem? TryGetItem(VirtualPath path, bool followLinks = false)
```

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

| Type | Name | Description |
|--|------|-----------------------|
| VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions) | | |
| Virtu VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | The path of the item. |

| Type | Name | Description |
|--|------|-------------|
| VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions) | | |

| Type | Name | Description |
|--|------|-------------|
| VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions) | | |

VirtualNodeTypeFilter

Returns

| Type | Description |
|---|---|
| VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) <T> | The found VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html), or null if not found. |

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

TryGetNode(VirtualPath, bool)

PatternMatch

Tries to get the node at the specified path.
(AkiraNetwork.VirtualStorageLibrary)**Declaration**

(AkiraNetwork.VirtualStorageLibrary)

```
public VirtualNode? TryGetNode(VirtualPath path, bool followLinks = false)
(AkiraNetwork.VirtualStorageLibrary)
VirtualDirectoryAdapter<T>
```

Parameters

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|--|------|------------------|
| VirtualException | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | The node's path. |

(AkiraNetwork.VirtualStorageLibrary)

| | | |
|---|-------------|-----------------------------------|
| VirtualGroupCondition<T, TKey> bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary) | followLinks | Whether to follow symbolic links. |
|---|-------------|-----------------------------------|

2.html)

Returns

VirtualID

| Type | Description |
|---|---|
| AkiraNetwork.VirtualStorageLibrary.VirtualItem VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) VirtualItemAdapter<T> VirtualNode.html (AkiraNetwork.VirtualStorageLibrary) | The found VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html), or null if not found. |

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

TryGetSymbolicLink(VirtualPath)

VirtualNode

Tries to get the symbolic link at the specified path.
(AkiraNetwork.VirtualStorageLibrary)**Declaration**

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

```
public VirtualSymbolicLink? TryGetSymbolicLink(VirtualPath path)
(AkiraNetwork.VirtualStorageLibrary)
```

VirtualNodeListConditions

Parameters

(AkiraNetwork.VirtualStorageLibrary)

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

VirtualNodeNotFoundException

| | | |
|--|------|--------------------------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | The path of the symbolic link. |
|--|------|--------------------------------|

(AkiraNetwork.VirtualStorageLibrary)

Returns

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| Type | Description |
|--|---|
| VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) | The found VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html), or null if not found. |

Remarks

This method does not have a followLinks parameter because it retrieves the symbolic link itself.

TryResolveLinkTarget(VirtualPath)

(AkiraNetwork.VirtualStorageLibrary)

Tries VirtualSymbolicLinkTarget of the symbolic link at the specified path.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
public VirtualDirectory TryResolveLinkTarget(VirtualPath path)
    (AkiraNetwork.VirtualStorageLibrary)
    1.html)
```

Parameters

VirtualException

| Type | Name | Description |
|------------------------------------|------|--------------------------------|
| AkiraNetwork.VirtualStorageLibrary | path | The path of the symbolic link. |

Returns

VirtualID

| Type | Description |
|------------------------------------|---|
| AkiraNetwork.VirtualStorageLibrary | The target path, or null if the node cannot be found. |

UpdateAllTargetNodeTypesInDictionary()

(AkiraNetwork.VirtualStorageLibrary)

Updates all target node types in the link dictionary.

Declaration

```
public void UpdateAllTargetNodeTypesInDictionary()
    (AkiraNetwork.VirtualStorageLibrary)
    1.html)
```

UpdateDirectory(VirtualPath, VirtualDirectory)

(AkiraNetwork.VirtualStorageLibrary)

Updates a directory.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
public void UpdateDirectory(VirtualPath directoryPath, VirtualDirectory newDirectory)
    (AkiraNetwork.VirtualStorageLibrary)
    1.html)
```

Parameters

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

```
public void UpdateDirectory(VirtualPath directoryPath, VirtualDirectory newDirectory)
```

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter


```
public VirtualNodeContext WalkPathToTarget(VirtualPath targetPath, NotifyNodeDelegate? notifyNode = null, ActionNodeDelegate? actionNode = null, bool followLinks = true, bool exceptionEnabled = true)
```

Parameters

| Type | Name | Description |
|---|------------------|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath) | targetPath | The target virtual path |
| (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate) | notifyNode | A delegate to notify information about the node when each node is reached |
| (AkiraNetwork.VirtualStorageLibrary.ActionNodeDelegate) | actionNode | A delegate to perform actions when each node is reached |
| bool | followLinks | A flag indicating whether to follow symbolic links |
| boo | exceptionEnabled | A flag indicating whether to enable exceptions. If true, exceptions are thrown; if false, the Node in VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html) will be returned as null. |

Returns

| Type | Description |
|---|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext) | An instance of VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html) containing information about the target node |

WalkPathTree(VirtualPath, VirtualNodeTypeFilter, bool, bool)

Traverses the path tree starting from the specified path.

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

```
public IEnumerable<VirtualNodeContext> WalkPathTree(VirtualPath basePath, VirtualNodeTypeFilter filter = VirtualNodeTypeFilter.All, bool recursive = true, bool followLinks = true, bool resolveLinks = true)
```

Parameters

| Type | Name | Description |
|---|--------------|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualPathNotifyNodeDelegate)(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | basePath | The base virtual path |
| PatternMatchVirtualNodeTypeFilter(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | filter | The filter to apply for node types |
| VirtualCycleDetector(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | | |
| boolVirtualDirectory(https://learn.microsoft.com/dotnet/api/system.boolean) | recursive | A flag indicating whether to traverse recursively |
| boolVirtualDirectoryAdapter<T>(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | followLinks | A flag indicating whether to follow symbolic links |
| boolVirtualException(https://learn.microsoft.com/dotnet/api/system.boolean) | resolveLinks | A flag indicating whether to resolve symbolic links |

VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

Returns

| Type | Description |
|-----------|--|
| VirtualID | An enumerable of VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) containing information about the traversed nodes |

Remarks

followLinks indicates whether to follow links during the recursive traversal process when the terminal node in the virtual path is a symbolic link.
(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

resolveLinks indicates whether to resolve links when nodes other than the terminal node in the virtual path are symbolic.
(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualNodeListConditions

Extension Methods

| | |
|--|--|
| VirtualTextFormatter | DebugText<T>(VirtualStorage<T>) |
| (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils_VirtualTextFormatter_VirtualStorageTable) | Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils_VirtualTextFormatter_VirtualStorageTableDebugText_1_AkiraNetwork_VirtualStorageLibrary_VirtualStorage__0_ |
| VirtualTextFormatter | GenerateSingleTableDebugText<T>(T) |
| (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils_VirtualTextFormatter_VirtualStorageTable) | Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils_VirtualTextFormatter_VirtualStorageTableDebugText_1_AkiraNetwork_VirtualStorageLibrary_VirtualStorage__0_ |
| VirtualNodeTypeFilter | |

```
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
VirtualTextFormatter.GenerateTreeDebugText<T>(VirtualStorage<T>, VirtualPath, bool, bool)
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateTreeDebugText_1_AkiraNetwork_VirtualStorageLibrary_VirtualStorage__0_Akira
Network_VirtualStorageLibrary_VirtualPath_System_Boolean_System_Boolean_)
```

▼

```
(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 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)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

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)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNameType

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNameType)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

VirtualNodeTypeFilter

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

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

| 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) | | |
| VirtualNodeName | <i>name</i> | The name of the symbolic link. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>targetPath</i> | The target path of the symbolic link. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | | |
| DateTime | <i>createdDate</i> | The creation date. |
| VIRTUALID (https://learn.microsoft.com/dotnet/api/system.datetime) | | |
| DateTime | <i>updatedDate</i> | The update date. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) | | |
| VirtualItem | | |
| Date (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) | | |
| VirtualItemAdapter<T> | | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) | | |
| VirtualItem<T> | | |

Properties

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

Type

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

NodeType

(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; }
```

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

Property Value

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

Type

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

VirtualNodeFilter

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

Overrides

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

| Type | Description |
|---|-------------|
| VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink) | |

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

ToString()

[VirtualDirectory](#)
[\(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory\)](#)
[VirtualDirectoryAdapter<T>](#)
[Returns a string representation of the symbolic link.](#)

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

Declaration

[VirtualException](#)
[p\(AkiraNetwork.VirtualStorageLibrary.ToString\(\)\)](#)

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

Returns

| Type | Description |
|---|---|
| virtualID (AkiraNetwork.VirtualStorageLibrary.VirtualItem) | A string representation of the symbolic link. |

Overrides

[VirtualItemAdapter<T>](#)
[object.ToString\(\) \(\[https://learn.microsoft.com/dotnet/api/system.object.tostring\]\(#\)\)](#)
[\(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter\)](#)
[1.html](#)

VirtualItem<T>

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

[Updates the current symbolic link with the data from the specified node.](#)

[VirtualNode](#)

Declaration

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

[VirtualNodeContext](#)

[p\(AkiraNetwork.VirtualStorageLibrary.VirtualNode virtualNode node\)](#)

[VirtualNodeExtensions](#)

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

Parameters

[VirtualNodeListConditions](#)

| Type | Name | Description |
|--|----------------------|---------------------------------|
| AkiraNetwork.VirtualStorageLibrary.VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNode) | node | The node to use for the update. |

Overrides

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

[VirtualNodeUpdate\(VirtualNode\)](#)

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

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

Exceptions

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

Operators

implicit operator VirtualSymbolicLink(VirtualPath?)

Performs an implicit conversion from VirtualPath ([AkiraNetwork.VirtualStorageLibrary.VirtualPath.html](#)) to VirtualSymbolicLink ([AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html](#)). The node name is automatically generated using the `VirtualNodeName.GenerateNodeName` method.

([AkiraNetwork.VirtualStorageLibrary](#))

Declaration

```
VirtualDirectoryAdapter<T>
{
    public static implicit operator VirtualSymbolicLink(VirtualPath? targetPath)
```

VirtualException

([AkiraNetwork.VirtualStorageLibrary](#))

Parameters

VirtualGroupCondition<T, TKey>

([AkiraNetwork.VirtualStorageLibrary](#))

Type

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

VirtualID

([AkiraNetwork.VirtualStorageLibrary](#))

Returns

VirtualItem

([AkiraNetwork.VirtualStorageLibrary](#))

VirtualItemAdapter<T>

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

([AkiraNetwork.VirtualStorageLibrary](#))

1.html)

VirtualItem<T>

([AkiraNetwork.VirtualStorageLibrary](#))

VirtualNode

([AkiraNetwork.VirtualStorageLibrary](#))

Perform an implicit conversion from a tuple to VirtualSymbolicLink

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

([AkiraNetwork.VirtualStorageLibrary](#))

Declaration

VirtualNodeExtensions

([AkiraNetwork.VirtualStorageLibrary](#))

```
public static implicit operator VirtualSymbolicLink((VirtualNodeName nodeName, VirtualPath?
VirtualNodeListConditions targetPath) tuple)
```

([AkiraNetwork.VirtualStorageLibrary](#))

VirtualNodeName

([AkiraNetwork.VirtualStorageLibrary](#))

VirtualNotFoundException

([AkiraNetwork.VirtualStorageLibrary](#))

VirtualNodeType

([AkiraNetwork.VirtualStorageLibrary](#))

VirtualNodeTypeFilter

| 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) NotifyNodeDelegate) (AkiraNetwork.VirtualStorageLibrary.PatternMatch | tuple | The tuple to convert. |

Returns VirtualCycleDetector

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

Implements 1.html

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

VirtualGroupCondition<T, TKey>

Extension Methods

VirtualNodeExtensions.ResolveNodeType(VirtualNode)
VirtualID

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode
(AkiraNetwork.VirtualStorageLibrary.Extensions.ResolveNodeType_AkiraNetwork_VirtualStorageLibrary_VirtualNode_)

```
VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)
(AkiraNetwork.VirtualStorage library)
```

(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils
ties_VirtualTextFormatter_Adapter_<T>_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualItem<T>
(AkiraNetwork VirtualStorage library)

1.html)

VirtualNode
(AkiraNetwork VirtualStorageLibrary)

VirtualNode

VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary)

VirtualM

(AkiraNetwork.VirtualStorageLibrary)

VirtualNode1 istCondition

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork VirtualStorage) library

VirtualNodeNotEc

Virtueller Speicher: CacheException

Virtual INI file

VirtualNode type

(AkiraNetwork.VirtualStorageLibrary

VirtualNodeTypeFilter

Class VirtualSymbolicLinkAdapter<T>

Adapter class to simplify operations on virtual symbolic links. Reduces the need for casting and simplifies code, allowing users to easily manage specific node types.

Inheritance

NotifyNodeDelegate

↳ object (<https://learn.microsoft.com/dotnet/api/system.object>)
↳ AkiraNetwork.VirtualStorageLibrary
↳ VirtualSymbolicLinkAdapter<T>
↳ PatternMatch

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>)

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.AkiraNetwork.html.VirtualStorageLibrary

Assembly

VirtualStorageLibrary.dll

Syntax

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

public class VirtualSymbolicLinkAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

Type

(AkiraNetwork.VirtualStorageLibrary)

| Name | Description |
|------|---|
| T | The type of storage item stored in the virtual storage. 1.html |

T (AkiraNetwork.VirtualStorageLibrary) The type of storage item stored in the virtual storage.

1.html

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualSymbolicLinkAdapter(VirtualStorage<T>)

(AkiraNetwork.VirtualStorageLibrary)

Initializes a new instance of the VirtualSymbolicLinkAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLinkAdapter-1.html) class.

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

public VirtualSymbolicLinkAdapter(VirtualStorage<T> storage)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Parameters

| Type | Name | Description |
|--|---------|-------------------------------|
| VirtualStorage (AkiraNetwork.VirtualStorageLibrary.VirtualStorage-1.html) <T> | storage | The virtual storage instance. |

▼

Properties

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary)

this[VirtualPath, bool]

PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

Gets or sets the virtual symbolic link corresponding to the specified path.

VirtualCycleDetector

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

public VirtualSymbolicLink this[VirtualPath path, bool followLinks = true] { get; set; }

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

Parameters

VirtualPath

(AkiraNetwork.VirtualStorageLibrary)

bool

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

Property Value

(AkiraNetwork.VirtualStorageLibrary)

Type

VirtualItemAdapter<T>

Description

(AkiraNetwork.VirtualStorageLibrary)

VirtualSymbolicLink

1.html

(AkiraNetwork.VirtualStorageLibrary.Virtua

lSymbolicLink.html)

(AkiraNetwork.VirtualStorageLibrary)

1.html)

| Type | Description |
|-----------------------|--|
| VirtualItemAdapter<T> | The VirtualSymbolicLink corresponding to the specified path. |

VirtualNode

Remarks

(AkiraNetwork.VirtualStorageLibrary)

To operate on the symbolic link itself, specify `followLinks` as false. Note that the default value of this parameter may change in the future.

(AkiraNetwork.VirtualStorageLibrary)

Exception

NodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

Type

VirtualNodeListConditions

Condition

(AkiraNetwork.VirtualStorageLibrary)

Inval

(AkiraNetwork.VirtualStorageLibrary)

(https://docs.microsoft.com/dotnet/api/system.inv

alidOperationException)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Thrown when the node is not a

VirtualSymbolicLink

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.ht

ml)

.

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

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

Utility class for generating text-based representations of the virtual storage's tree structure. Provides static methods for generating various debug texts.

Inheritance

AkiraNetwork.VirtualStorage

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

(AkiraNetwork.VirtualStorage)

Inherited Members

AkiraNetwork.VirtualStorage

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.VirtualStorage)

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

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

(AkiraNetwork.VirtualStorage)

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

Assembly: VirtualStorageLibrary.dll

Syntax

```
public static class VirtualTextFormatter
```

Remarks

This class distinguishes between full-width and half-width characters based on specific character code ranges. The following ranges are considered full-width characters:

- 0x1100 to 0x115F: Korean Hangul Jamo characters
- 0xE80 to 0xA4CF (excluding 0x303F): CJK Unified Ideographs and compatibility characters
- 0xAC00 to 0xD7A3: Korean Hangul syllables
- 0xF900 to 0xFAFF: CJK compatibility ideographs
- 0xFE10 to 0xFE19: Vertical forms punctuation
- 0xFE30 to 0xFE6F: CJK compatibility forms
- 0xFF00 to 0xFF60: Full-width ASCII and symbols
- 0FFE0 to 0FFE6: Full-width special symbols

The distinction between full-width and half-width characters is important for ensuring proper alignment and formatting in text-based representations, especially when working with mixed-language data or presenting data in a tabular format. This classification helps in correctly calculating string widths and aligning elements in debug text outputs, such as file paths or tabular data, where accurate spacing is crucial.

Methods

GenerateLinkTableDebugText<T>(VirtualStorage<T>)

Generates and returns a table representation of symbolic link information.



Declaration

+ AkiraNetwork.VirtualStorage

```
public static string GenerateLinkTableDebugText<T>(this VirtualStorage<T> vs)
```

Library

(AkiraNetwork.VirtualStorage)

Parameters

- AkiraNetwork.VirtualStorage

Type

Library.Utilities

VirtualStorage (AkiraNetwork.VirtualStorageLibrary.VirtualStorage-

1.html)

<T> VirtualTextFormatter

(AkiraNetwork.VirtualStorageLibrary.

Name

Description

vs

The instance of the virtual storage.

Returns

AkiraNetwork.VirtualStorage

Library.WildcardMatchers

Type

(AkiraNetwork.VirtualStorage)

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

Description

The generated debug text of the link table.

Type Parameters

| Name | Description |
|------|----------------------------------|
| T | The type of data in the storage. |

GenerateSingleTableDebugText<T>(T)

Generates and returns a table representation of a single object's properties.

Declaration

```
public static string GenerateSingleTableDebugText<T>(this T singleObject)
```

Parameters

| Type | Name | Description |
|------|--------------|-------------------------------|
| T | singleObject | The object to be represented. |

Returns

| Type | Description |
|--|--|
| string (https://learn.microsoft.com/dotnet/api/system.string) | The generated debug text of the table. |

Type Parameters

| Name | Description |
|----------|-------------------------|
| <i>T</i> | The type of the object. |

GenerateTableDebugText<T>(IEnumerable<T>)

Generates and returns a table representation of the collection's contents.

+ AkiraNetwork.VirtualStorage

Declaration
Library

(AkiraNetwork.VirtualStorage)

```
public static string GenerateTableDebugText<T>(this IEnumerable<T> enumerableObject)
```

- AkiraNetwork.VirtualStorage

Library Utilities

Parameters

(AkiraNetwork.VirtualStorage)

Type

VirtualTextFormatter

IEnumerable

(AkiraNetwork.VirtualStorageLibrary

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

+ AkiraNetwork.VirtualStorage

<Library.WildcardMatchers

(AkiraNetwork.VirtualStorage)

Returns

| Type | Description |
|--|--|
| string (https://learn.microsoft.com/dotnet/api/system.string) | The generated debug text of the table. |

Type Parameters

| Name | Description |
|----------|--|
| <i>T</i> | The type of objects in the collection. |

GenerateTreeDebugText<T>(VirtualStorage<T>, VirtualPath, bool, bool)

Generates and returns a text-based representation of the virtual storage tree structure.

Declaration

```
public static string GenerateTreeDebugText<T>(this VirtualStorage<T> vs, VirtualPath basePath, bool recursive = true, bool followLinks = false)
```

Parameters

| Type | Name | Description |
|-------------------|------|--------------------------------------|
| VirtualStorage<T> | vs | The instance of the virtual storage. |

| Type | | Name | Description |
|--|--|--------------------|---|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | | <i>basePath</i> | The base path from which to start. |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) ▼ | | <i>recursive</i> | Whether to list subdirectories recursively. |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | | <i>followLinks</i> | Whether to follow symbolic links. |

+ AkiraNetwork.VirtualStorage

Library

| Type | Description |
|--|---|
| -string (https://learn.microsoft.com/dotnet/api/system.string) | The generated debug text of the tree structure. |

Library.Utilities

Type AkiraNetwork.VirtualStorage

| Name | Description |
|--|----------------------------------|
| <i>T</i> (AkiraNetwork.VirtualStorageLibrary) | The type of data in the storage. |

+ AkiraNetwork.VirtualStorage

Library.WildcardMatchers

The generated text represents the tree structure starting from the specified base path. Each node is represented by the result of the `ToString` method overridden in the specific derived class of the node.

When `recursive` is true, subdirectories are listed recursively. If false, subdirectories are not included in the output.

When `followLinks` is true, symbolic links are replaced with their target paths in the output. If false, the symbolic links themselves are displayed.

The output format includes indents and special characters to denote directory hierarchies and symbolic links.
Example:

```

/
├dir1/
│├subdir1/
││└item3
│└item2
└link-to-dir -> /dir1
└item1
└link-to-item -> /item1

```

Namespace AkiraNetwork.VirtualStorage Library.WildcardMatchers

▼ Filter by title

Classes

+ **AkiraNetwork.VirtualStorage**

DefaultWildcardMatcher

 (AkiraNetwork.VirtualStorageLibrary.WildcardMatchers.DefaultWildcardMatcher.html)

- **AkiraNetwork.VirtualStorage**

LibraryUtilities

 Implements default wildcard matching using regular expression patterns. Provides a straightforward mapping of

 regular expression symbols.

 (AkiraNetwork.VirtualStorageLibrary.WildcardMatchers.DefaultWildcardMatcher.html)

- **AkiraNetwork.VirtualStorage**

PowerShellWildcardMatcher

 (AkiraNetwork.VirtualStorageLibrary.WildcardMatchers.PowerShellWildcardMatcher.html)

 DefaultWildcardMatcher

 Implements wildcard matching based on PowerShell's wildcard patterns. Provides pattern matching with

 support for specific escape characters.

 PowerShellWildcardMatcher

 (AkiraNetwork.VirtualStorageLibrary.WildcardMatchers.PowerShellWildcardMatcher.html)

Class DefaultWildcardMatcher

Implements default wildcard matching using regular expression patterns. Provides a straightforward mapping of regular expression symbols.

Implementation

AkiraNetwork.VirtualStorage

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

(AkiraNetwork.VirtualStorage)

Implementation

AkiraNetwork.VirtualStorage

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

Library.Utilities

(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))

Library.WildcardMatchers

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)))

DefaultWildcardMatcher ([https://learn.microsoft.com/dotnet/api/system.object.gethashcode](#))

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

PowerShellWildcardMatcher (<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 ([AkiraNetwork.html](#)).VirtualStorageLibrary ([AkiraNetwork.VirtualStorageLibrary.html](#)).Wildcard Matchers ([AkiraNetwork.VirtualStorageLibrary.WildcardMatchers.html](#))

Assembly: VirtualStorageLibrary.dll

Syntax

```
public class DefaultWildcardMatcher : IVirtualWildcardMatcher
```

Properties

Count

Gets the number of supported wildcard characters.

Declaration

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

Property Value

| Type | Description |
|---|--|
| int (https://learn.microsoft.com/dotnet/api/system.int32) | The number of supported wildcard characters. |

Patterns

Gets the list of regular expression patterns corresponding to the wildcard characters.

+ AkiraNetwork.VirtualStorage

Declaration
Library

(AkiraNetwork.VirtualStorage)

```
public IEnumerable<string> Patterns { get; }
```

+ AkiraNetwork.VirtualStorage

Library Utilities

Property Value

(AkiraNetwork.VirtualStorage)

Type

- AkiraNetwork.VirtualStorage

IEnumerable

Library.WildcardMatchers

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

(AkiraNetwork.VirtualStorage)

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

(AkiraNetwork.VirtualStorageLibrary)

PowerShellWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary)

WildcardDictionary

Gets the dictionary of regular expression symbols and their corresponding literals.

Declaration

```
public ReadOnlyDictionary<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 where the keys are regular expression symbols and the values are their literal equivalents. |

Wildcards

Gets the list of supported wildcard characters.

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)> | A collection of supported wildcard characters. |

+ AkiraNetwork.VirtualStorage

Library

Methods

+ AkiraNetwork.VirtualStorage

PatternMatcher(string, string)

(AkiraNetwork.VirtualStorage)

Performs simple regular expression-based wildcard matching.

- AkiraNetwork.VirtualStorage

Declaration

Library.WildcardMatchers

(AkiraNetwork.VirtualStorage)

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

DefaultWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary)

Parameters

PowerShellWildcardMatcher

| Type | Name | Description |
|---|----------|---|
| string (https://learn.microsoft.com/dotnet/api/system.string) | nodeName | The name of the node to match against the pattern. |
| string (https://learn.microsoft.com/dotnet/api/system.string) | pattern | The regular expression pattern to use for matching. |

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_)

▼

+ **AkiraNetwork.VirtualStorage
Library**
[\(AkiraNetwork.VirtualStorageLibrary\)](#)

+ **AkiraNetwork.VirtualStorage
Library.Utilities**
[\(AkiraNetwork.VirtualStorageLibraryUtilities\)](#)

- **AkiraNetwork.VirtualStorage
Library.WildcardMatchers**
[\(AkiraNetwork.VirtualStorageLibraryWildcardMatchers\)](#)

 DefaultWildcardMatcher
[\(AkiraNetwork.VirtualStorageLibraryDefaultWildcardMatcher\)](#)
 PowerShellWildcardMatcher
[\(AkiraNetwork.VirtualStorageLibraryPowerShellWildcardMatcher\)](#)

Class PowerShellWildcardMatcher

Implements wildcard matching based on PowerShell's wildcard patterns. Provides pattern matching with support for ~~Filter by title~~ specific escape characters.

Implementation

AkiraNetwork.VirtualStorage

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

(AkiraNetwork.VirtualStorage)

Implementation

AkiraNetwork.VirtualStorage

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

Library.Utilities

(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))

Library.WildcardMatchers

object.Equals(object, object) (

(AkiraNetwork.VirtualStorage)

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

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

PowerShellWildcardMatcher

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 ([AkiraNetwork.html](#)).VirtualStorageLibrary ([AkiraNetwork.VirtualStorageLibrary.html](#)).Wildcard Matchers ([AkiraNetwork.VirtualStorageLibrary.WildcardMatchers.html](#))

Assembly: VirtualStorageLibrary.dll

Syntax

```
public class PowerShellWildcardMatcher : IVirtualWildcardMatcher
```

Properties

Count

Gets the number of supported wildcard characters.

Declaration

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

Property Value

| Type | Description |
|---|--|
| int (https://learn.microsoft.com/dotnet/api/system.int32) | The number of supported wildcard characters. |

Patterns

Gets the list of regular expression patterns corresponding to the wildcard characters.

+ AkiraNetwork.VirtualStorage

Declaration
Library

(AkiraNetwork.VirtualStorage)

```
public IEnumerable<string> Patterns { get; }
```

+ AkiraNetwork.VirtualStorage

Library Utilities

Property Value

(AkiraNetwork.VirtualStorage)

Type

- AkiraNetwork.VirtualStorage

IEnumerable

Library.WildcardMatchers

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

(AkiraNetwork.VirtualStorage)

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

(AkiraNetwork.VirtualStorageLibrary)

PowerShellWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary)

WildcardDictionary

Gets the dictionary of wildcard characters and their corresponding regular expression patterns.

Declaration

```
public ReadOnlyDictionary<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 where the keys are wildcard characters and the values are their corresponding regular expression patterns. |

Wildcards

Gets the list of supported wildcard characters.

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)> | A collection of supported wildcard characters. |

+ AkiraNetwork.VirtualStorage

Library

Methods

+ AkiraNetwork.VirtualStorage

PatternMatcher(string, string)

(AkiraNetwork.VirtualStorage)

Performs pattern matching using the specified PowerShell wildcard pattern.

- AkiraNetwork.VirtualStorage

Declaration

Library.WildcardMatchers

(AkiraNetwork.VirtualStorage)

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

DefaultWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary)

Parameters

PowerShellWildcardMatcher

| Type | Name | Description |
|---|----------|--|
| string (https://learn.microsoft.com/dotnet/api/system.string) | nodeName | The name of the node to match against the pattern. |
| string (https://learn.microsoft.com/dotnet/api/system.string) | pattern | The PowerShell wildcard pattern to use for matching. |

Returns

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

Remarks

If a wildcard character is escaped using a backtick (`), the character is treated as a literal. The escape character itself is represented by the backtick (`), and the character following it is the one being escaped.

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_)

▼

- + **AkiraNetwork.VirtualStorage**
Library
(AkiraNetwork.VirtualStorage)
- + **AkiraNetwork.VirtualStorage**
Library.Utilities
(AkiraNetwork.VirtualStorage)
- **AkiraNetwork.VirtualStorage**
Library.WildcardMatchers
(AkiraNetwork.VirtualStorage)

DefaultWildcardMatcher
(AkiraNetwork.VirtualStorageLibrary.WildcardMatchers.DefaultWildcardMatcher)

PowerShellWildcardMatcher
(AkiraNetwork.VirtualStorageLibrary.WildcardMatchers.PowerShellWildcardMatcher)