

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 entities.
(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>

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter-1.html)

VirtualException

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

(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)

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>

Structs

1.html)

VirtualItem<T>

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 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.VirtualStorageLibrary)

ActionNodeDelegate

IVirtualWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary.IVirtualWildcardMatcher.html)

Provides functionality for wildcard matching.

1.html

IVirtualWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html)

PatternMatch

Specifies the types of virtual nodes.

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html)

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

Specifies the filters for virtual node types.

(VirtualDirectoryAdapter<T>)

(AkiraNetwork.VirtualStorageLibrary)

html

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

ActionNodeDelegate<T, TKey>

(AkiraNetwork.VirtualStorageLibrary.ActionNodeDelegate.html)

2.html

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

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate<T>

(AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate.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)

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](#) ([VirtualStorageLibrary](#))

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)

Declaration

VirtualException

T (AkiraNetwork.VirtualStorageLibrary) false

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

Parameters ([2.html](#))

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

1.html)

Returns VirtualItem<T>

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

Extension Methods

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



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

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

Interface IVirtualWildcardMatcher

Provides functionality for wildcard matching.

Filter by title

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

Library

Assembly: VirtualStorageLibrary.dll

(AkiraNetwork.VirtualStorageLibrary)

Syntax

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

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

```
(AkiraNetwork.VirtualStorageLibrary  
NotifyNodeDelegate
```

```
(AkiraNetwork.VirtualStorageLibrary  
PatternMatch
```

```
(AkiraNetwork.VirtualStorageLibrary  
Count
```

```
(AkiraNetwork.VirtualStorageLibrary  
Gets VirtualNodeDelegates.
```

```
(AkiraNetwork.VirtualStorageLibrary  
Declaration
```

```
VirtualDirectory
```

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

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

Property Value

```
VirtualException
```

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

```
VirtualID
```

```
(AkiraNetwork.VirtualStorageLibrary  
Patterns
```

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

```
VirtualItemAdapter<T>
```

Declaration

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

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

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

Property Value

```
(AkiraNetwork.VirtualStorageLibrary  
VirtualNodeContext
```

```
(AkiraNetwork.VirtualStorageLibrary  
VirtualNodeExtensions
```

```
(AkiraNetwork.VirtualStorageLibrary  
VirtualNodeListConditions
```

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

Library

WildcardDictionaryVirtualStorage

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

Declaration

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

Property Value

NotifyNodeDelegate

| Type | Description |
|--|---|
| AkiraNetwork.VirtualStorageLibrary PatternMatch ReadOnlyDictionary (AkiraNetwork.VirtualStorageLibrary) (https://learn.microsoft.com/dotnet/api/system.collections.objectmodel.VirtualCycleDetector.readonlydictionary) <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. |

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

Wildcards

VirtualException

Gets (AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

Declaration

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

Type

VirtualItemAdapter<T>

AkiraNetwork.VirtualStorageLibrary

1.html

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

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

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

Methods

PatternMatcher(string, string)

Determines whether the specified node name matches the specified pattern.

Declaration

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



Parameters

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

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

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

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

Extension Methods

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

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

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

[VirtualException](#)

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

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

[VirtualID](#)

([AkiraNetwork.VirtualStorageLibrary](#)
VirtualItem

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

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

[VirtualItem](#)<T>

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

[VirtualNode](#)

([AkiraNetwork.VirtualStorageLibrary](#)
VirtualNodeContext

([AkiraNetwork.VirtualStorageLibrary](#)
VirtualNodeExtensions

([AkiraNetwork.VirtualStorageLibrary](#)
VirtualNodeListConditions

Delegate NotifyNodeDelegate

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

Filter by title

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

Assembly: VirtualStorageLibrary.dll

ActionNodeDelegate

Syntax

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

IVirtualWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary)

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

(AkiraNetwork.VirtualStorageLibrary.PatternMatcher)

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

(AkiraNetwork.VirtualStorageLibrary.VirtualNode)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

Extension Methods

(AkiraNetwork.VirtualStorageLibrary)

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)

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

(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)

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

s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

VirtualItem (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)))
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)))
PatternMatch
object.GetHashCode() (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)
object.GetType() (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)
VirtualCycleDetector
object.MemberwiseClone() (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)
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)

VirtualExtensions
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary)

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

Declaration

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

Property Value

| Type | Description |
|---|--------------------------------|
| int (https://learn.microsoft.com/dotnet/api/system.int32) IVirtualDeepCloneable<T> (AkiraNetwork.VirtualStorageLibrary. 1.html) VirtualWildcardMatcher (AkiraNetwork.VirtualStorageLibrary. The dictionary of detected cycles: NotifyNodeDelegate Declaration (AkiraNetwork.VirtualStorageLibrary. PatternMatch p(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink) CycleDictionary { get; } VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary. VirtualDirectory Property Value VirtualDirectory | The number of detected cycles. |

CycleDictionary

Gets the dictionary of detected cycles.

NotifyNodeDelegate

Declaration

(AkiraNetwork.VirtualStorageLibrary.
PatternMatch
p(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink) CycleDictionary { get; }
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary.
VirtualDirectory
Property Value
VirtualDirectory

| Type | Description |
|--|---|
| AkiraNetwork.VirtualStorageLibrary. VirtualDirectoryAdapter<T> Dictionary (AkiraNetwork.VirtualStorageLibrary. (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2) VirtualException < (AkiraNetwork.VirtualStorageLibrary. VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualID.html) (AkiraNetwork.VirtualStorageLibrary. , VirtualSymbolicLink VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink. (AkiraNetwork.VirtualStorageLibrary. nk.html) VirtualItem > (AkiraNetwork.VirtualStorageLibrary. VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary. 1.html) | A dictionary where the keys are VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualID.html) s and the values are the corresponding VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink. html) s. |

Methods

| Method | Description |
|---------------------------|---|
| <code>void Clear()</code> | (AkiraNetwork.VirtualStorageLibrary. 1.html) |

Clear()

(AkiraNetwork.VirtualStorageLibrary.
Clears the dictionary of detected cycles:
VirtualNodeContext

Declaration

(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeExtensions

```
public void Clear()
```

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.
VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.
VirtualNodeName

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. |

Virtu
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)

Virtual

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualN

(AkiraNetwork.Virtual)

VirtualNodeExtensions

(AkiraNetwork.VirtualSt

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

Class VirtualDirectory

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

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

NotifyNodeDelegate

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

PatternMatch

Inheritance

VirtualCycleDetector

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

↳ VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

↳ VirtualDirectory

↳ VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)

Implementation

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

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

IVirtualException

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

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

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

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

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)

Inherited Members

2.html)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Name)

VirtualNodeCreatedDate

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_CreatedDate)

VirtualItemAdapter<T>

VirtualNodeUpdatedDate

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_UpdatedDate)

VirtualItem<T>

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_VID)

VirtualNodeIsReferencedInStorage

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_IsReferencedInStorage)

VirtualNodeContext

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

VirtualNodeExtensions

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

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)

VirtualNodeGetType

(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.html)
1.html)
```

Constructors

VirtualNodeMatcher

(AkiraNetwork.VirtualStorageLibrary.html)

NotifyNodeDelegate

VirtualDirectory()

(AkiraNetwork.VirtualStorageLibrary.html)

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

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

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

VirtualDirectory

Declaration

(AkiraNetwork.VirtualStorageLibrary.html)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.html)

1.html)

VirtualException

(AkiraNetwork.VirtualStorageLibrary.html)

VirtualDirectory(VirtualNodeName)

(AkiraNetwork.VirtualStorageLibrary.html)

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

VirtualName

Declaration

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.html)

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

(AkiraNetwork.VirtualStorageLibrary.html)

Parameters

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

VirtualDirectory(VirtualNodeName, DateTime, DateTime)

VirtualNodeExtensions

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

(AkiraNetwork.VirtualStorageLibrary.html)

Declaration

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.html)

VirtualNodeNotFoundException

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

Parameters

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

Properties

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

VirtualCycleDetector

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

VirtualDirectory

Gets the total number of nodes within the directory.

VirtualException

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

public int Count { get; }

VirtualException

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

Type
(VirtualItem.html)

Description

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

(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)

DirectoryCount
(VirtualItemAdapter<T>.DirectoryCount.html)

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

Gets the number of directories within the directory.

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)

Declaration
(VirtualItemAdapter<T>.DirectoryCount.html)

public int DirectoryCount { get; }

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

Property Value
(VirtualNodeContext.html)

Type
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html)

Description

VirtualNodeExtensions

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

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html)

VirtualNodeName
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html)

DirectoryViewCount
(VirtualNodeViewCount.html)

VirtualNodeViewCount
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeViewCount.html)

VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html)

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

ItemCount

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

IVirtualDeepCloneable<T>

Declaration

[AkiraNetwork.VirtualStorageLibrary](#)

1.html)

public int ItemCount { get; }

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

Property Value

(AkiraNetwork.VirtualStorageLibrary)

Type PatternMatch

(AkiraNetwork.VirtualStorageLibrary)

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

Description

VirtualWildcardMatcher

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

NodeNames

VirtualDirectoryAdapter<T>

Enum ([AkiraNetwork.VirtualStorageLibrary](#))

Gets the list of nodes in the directory.

1.html)

Declaration

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

public IEnumerable<VirtualNodeName> NodeNames { get; }

(AkiraNetwork.VirtualStorageLibrary)

2.html)

Property Value

VirtualID

Type (AkiraNetwork.VirtualStorageLibrary)

Description

VirtualItem

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualnodeName (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

NodeType

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

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

1.html)

Declaration

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

public override VirtualNodeType NodeType { get; }

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

Property Value

(AkiraNetwork.VirtualStorageLibrary)

Type VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html)

(AkiraNetwork.VirtualStorageLibrary)

Overrides

Description

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; }
VirtualWildcardMatcher
(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

VirtualException
(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) | |

SymbolicLinkCount

Gets the number of symbolic links within the directory.

Declaration

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

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

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

SymbolicLinkViewCount

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

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

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

Methods

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

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

| Adds a node to the directory. |
|--|
| (AkiraNetwork.VirtualStorageLibrary.Add(VirtualNode, bool).html) |

| Declaration |
|---|
| AkiraNetwork.VirtualStorageLibrary.VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) |
| 1.html |

| public VirtualNode Add(VirtualNode node, bool allowOverwrite = false) |
|---|
| (AkiraNetwork.VirtualStorageLibrary.Add(VirtualNode, bool).html) |

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

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

| 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

| Type | Condition |
|---|---|
| <code>IVirtualWildcardMatcher</code> (AkiraNetwork.VirtualStorageLibrary.Arguments.Matcher.html) | Thrown if an invalid node name is specified. |
| <code>InvalidOperationException</code> (https://learn.microsoft.com/dotnet/api/system.invalidoperationexception) | Thrown if a node with the same name already exists. |

AddDirectory(VirtualDirectoryName, bool)

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

Declaration

```
public VirtualDirectory AddDirectory(VirtualDirectoryName name, bool allowOverwrite = false)
{
    VirtualID
}
```

Parameters

| Type | Name | Description |
|--|-----------------------|--|
| <code>AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T></code> <code>VirtualDirectoryName</code> (AkiraNetwork.VirtualStorageLibrary.Arguments.Name.html) | <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

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

AddItem(T)(VirtualDirectoryName, 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 | AkiraNetwork.VirtualStorageLibrary |
| name | VirtualNodeName IVirtualWildcardMatcher (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) NotifyNodeDelegate |
| itemData | T (AkiraNetwork.VirtualStorageLibrary) PatternMatch bool (AkiraNetwork.VirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.boolean)) VirtualCycleDetector |
| allowOverwrite | VirtualDirectory VirtualException |

Returns

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

Type AkiraNetwork.VirtualStorageLibrary

| VirtualGroupCondition<T, TKey> | |
|--------------------------------------|----------------------------|
| (AkiraNetwork.VirtualStorageLibrary) | |
| T 2.html | The data type of the item. |

Type VirtualID

Type VirtualItem

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)
```

Parameters

| VirtualNodeContext | |
|--------------------|--|
| Type | AkiraNetwork.VirtualStorageLibrary |
| name | VirtualNodeExtensions VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) VirtualNodeListConditions |
| targetPath | (AkiraNetwork.VirtualStorageLibrary) |
| allowOverwrite | VirtualNodeName (AkiraNetwork.VirtualStorageLibrary) |
| | VirtualNodeNotFoundException |


```
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

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryName.html)

VirtualNodeNotFoundException

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

Parameters

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

Returns

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

Exceptions

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

GetEnumerator()

Returns

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

VirtualID
Declaration
(AkiraNetwork.VirtualStorageLibrary.VirtualID.html)

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

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

Returns
1.html)

| Type | Description |
|---|-----------------------------------|
| VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) | An enumerator for filtered nodes. |

GetItemByName(VirtualNodeName)

Gets the item with the specified name.

VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html)

Declaration
VirtualNodeName
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html)

```
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 VirtualNodeNameCloneable<T> (AkiraNetworkVirtualStorageLibraryVirtualNodeName.html) 1.html | | name | The name of the symbolic link. |

Return VirtualWildcardMatcher

| Type | (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate) | Description |
|------|---|--|
| Virt | (AkiraNetwork.VirtualStorageLibrary.(AkiPatternVirtualStorageLibrary.VirtualSymbolicLink.html)) | The symbolic link with the specified name. |

~~VirtualCycleDetector~~

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

ItemExists(VirtualNodeName)

(AkiraNetwork.VirtualStorageLibrary)

Check **Virtually there** if an item with the specified name exists.

(AkiraN)

```
Declaration  
VirtualItemAdapter<T>  
(AkiraNetwork.VirtualStorageLibrary  
public bool ItemExists(VirtualNodeName name)  
.html)
```

Parameters (AkiraN)

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

VirtualNetwork.Virtual

Returns: `VirtualNodeExtensions`

| Type | Description |
|---|--|
| (AkiraNetwork.VirtualStorageLibrary.IVirtualNodeListConditions) | True if the item exists; otherwise, false. |

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.PatternMatch |

Remove Virtual Node

(AkiraNetwork.VirtualStorageLibrary)

RemoveNode(VirtualNetwork, VirtualStorage, NodeName)

Virtual E-Declarations

(AkiraNetwork VirtualStorage) library

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

2.html)

Parameters

| Type | (AkiraNetwork.VirtualStorageLibrary.VirtualItem) | Name | Description |
|------|---|-------------|---------------------|
| Virt | (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.VirtualNode.html) | <i>node</i> | The node to remove. |

Exce~~mber~~ (Akira)

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

SymbolicLinkExists(VirtualNodeName)

VirtualNodeExtensions

`virtualNodeExtensions`: Checks whether a symbolic link with the alias `VirtualNode` exists.

(AkiraNetwork.VirtualStorageLibrary)

Declaration NodeListCondition

(AkiraNetwork.VirtualStorageLibrary)

`VirtualNodeName` `symbolicLinkExists()`

```
public bool symbolicLinkExists()
```

(AkiraNetwork.VirtualStorageLib)

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

```
VirtualCycleDetector  
(AkiraNetwork.VirtualStorageLibrary.VirtualNode)  
public override string ToString()  
    VirtualDirectory  
(AkiraNetwork.VirtualStorageLibrary.VirtualNode)
```

Returns

| Type | Description |
|--|---|
| IVirtualDirectoryAdapter<T> (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

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

VirtualNodeListConditions

```
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Update_A  
AkiraNetwork_VirtualStorageLibrary_VirtualNode_)
```

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualNode)

VirtualNodeNotFoundException

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

1

Operators

IVirtualDeepCloneable<T>

(AkiraNetwork.VirtualStorageLibrary

1.html) + VirtualDir

operator + (virtualDirectory, virtualNode)
IVirtualWildcardMatcher
Overlaid a new directory structure on top of a node to the directory.

NotifyNodeDelegate

Declaration

(AKHANETWOR
Battaw Matla

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

VirtualCycleDetector

VirtualCycleDetector

Wittman, D. J.

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

VirtualGroupCondition<T, TKey>

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

implicit operator VirtualDirectory(VirtualNodeName)

Performs an implicit conversion from VirtualNodeName^{1.html})

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) to VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryName.html).
1.html)

Declaration

(AlioNet and Virtuoso) and libraries

VirtuaLINX de Carteret | 1000 N. Main Street, Suite 100 | 1-800-334-3333 | www.virtualinx.com

(AlienNet and VietriSegment library)

Virtual Network Virtualization

VirtualNodeExtensions Parameters

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

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_) | |

VirtualNodeListConditions

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

| | |
|--|--|
| VirtualStorageExtensions.VirtualStorageNotFoundException | |
|--|--|

```
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

(AkiraNetwork.VirtualStorageLibrary 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)))
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)))
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>)
VirtualDirectoryAdapter<T>
object.ReferenceEquals(object, object) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)
object.ToString() (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)
1.html)

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

Assembly

(AkiraNetwork.VirtualStorageLibrary
VirtualGroupCondition<T, TKey>
 (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

Constructors

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

(AkiraNetwork.VirtualStorageLibrary
Declaration
VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary
public VirtualDirectoryAdapter(VirtualStorage<T> storage)
VirtualNodeName
(AkiraNetwork.VirtualStorageLibrary

VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary

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

1

Properties

(AkiraNetwork.VirtualStorageLibrary
1.html)

this **VirtualPath**, **book**
VirtualWildcardMatcher
Akurnetwerk.VirtualSto

Gets or sets the virtual directory corresponding to the specified path.

(AkiraN)

PatternMatch

```
public virtual string VirtualPath { get; set; }
```

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory Parameters

(AkiraNetwork.VirtualStorageLibrary)

| | | |
|--|-------------------|----------------------------|
| <code>VirtualPath</code> (<code>AkiraNetwork.VirtualStorageLibrary.VirtualPath.html</code>) | <code>path</code> | The path of the directory. |
| <code>VirtualException</code> | | |

| | | |
|--|--------------------------|--|
| <code>bool</code> (<code>AkiraNetwork.VirtualStorageLibrary.VirtualFileConditionToken</code>) | <code>followLinks</code> | A flag indicating whether to follow links. |
|--|--------------------------|--|

PropertiesValue

EntityValue
VirtualID

Type (AkiraNetwork.VirtualStorageLibrary)

The `VirtualDirectory` (`AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html`) corresponds to the specified path.

Virtual

`VirtualItem<I>`
`options`
(AkiraNetwork.VirtualStorageLibrary)

| Type | Condition |
|--|---|
| <code>.html)</code> VirtualNode InvalidOperationException (AkiraNetwork.VirtualStorageLibrary. (https://learn.microsoft.com/dotnet/api/system.invalidoperationexception) VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary. | Thrown when the node is not a VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) |

Extension Method

VirtualNodeListConditions

Altiris Network Virtual Storm

VirtualTextFormatMapper GenerateSingleTab

(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GeneratesStringTableDebugText_1__0_)
(AkiraNetwork.VirtualStorageLibrar



(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, 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)

message

The error message.

1.html)

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualException(string, Exception)

VirtualGroupCondition<T, Key>

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

VirtualID

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

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

VirtualItemAdapter<T>

Parameters

1.html)

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

1.html)

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

VirtualNode

Exception

(AkiraNetwork.VirtualStorageLibrary)

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

ISerializable

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

Implements

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) (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)
1.html
object.ReferenceEquals(object, object) (https://learn.microsoft.com/dotnet/api/system.object.referenceequals)
VirtualException
object.ToString() (https://learn.microsoft.com/dotnet/api/system.object.tostring)
(AkiraNetwork.VirtualStorageLibrary)

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

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

Syntax

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

Type Parameters

| Name | Description |
|------|---|
| T | VirtualItem<T> The type of the entity to be grouped. |
| TKey | (AkiraNetwork.VirtualStorageLibrary) The type of the key used for grouping. 1.html |

Constructors

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

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

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

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

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

Parameters

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

Properties

VirtualGroupAdapter<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) |
| VirtualTextFormatter | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualTextMatcher | _GenerateSingleTableDebugText_1__0_(AkiraNetwork.VirtualStorageLibrary) |
| VirtualCycleDetector | |
| VirtualDirectory | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualDirectoryAdapter<T> | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualException | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualGroupCondition<T, TKey> | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualID | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualItem | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualItemAdapter<T> | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualItem<T> | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNode | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeContext | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeExtensions | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeListConditions | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualnodeName | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeNotFoundException | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeType | (AkiraNetwork.VirtualStorageLibrary) |

Struct VirtualID

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

(AkiraNetwork.VirtualStorageLibrary)

Implements

IEquatable<VirtualID>

(AkiraNetwork.VirtualStorageLibrary)

PatternMatch

(AkiraNetwork.VirtualStorageLibrary.VirtualID.html)

(AkiraNetwork.VirtualStorageLibrary)

Inherited Members

VirtualCycleDetector

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

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

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

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 a unique identifier.

VirtualNode

(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

NodeFilter

NodeTypeFilter

Declaration

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

Property Value

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

Methods

`ToString()`

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

1.html)

Declaration

VirtualException

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

(AkiraNetwork.VirtualStorageLibrary
2.html)

Returns

VirtualID

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

Description

A string that represents the unique identifier.

Overrides

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

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary
1.html)

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

(AkiraNetwork.VirtualStorageLibrary
1.html)

(AkiraNetwork.VirtualStorageLibrary
VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)
VirtualNodeExtensions

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

s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

(AkiraNetwork.VirtualStorageLibrary
VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary
VirtualTextFormatter

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary
VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary
VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary
VirtualNodeTypeFilter

Class VirtualItem

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

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

([AkiraNetwork.VirtualStorageLibrary](#))

Inheritance

([AkiraNetwork.VirtualStorageLibrary](#))

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

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

↳ `VirtualItem`

↳ `VirtualCycledDetector`

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

Implements

([AkiraNetwork.VirtualStorageLibrary](#))

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

↳ `VirtualDirectoryAdapter<T>`

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

Inherited Members

1.html)

`VirtualException`

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualGroupCondition<T, TKey>`

([AkiraNetwork.VirtualStorageLibrary](#))

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

`VirtualID`

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualUpdatedDate`

([AkiraNetwork.VirtualStorageLibrary](#))

`DateVirtualItem`

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualNode`

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualNodeAdapter<T>`

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualNode` & `VirtualNodeTypeFilter`

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualNode.VID`

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualUpdateVirtualStorageLibrary`

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualNode` & `VirtualNodeUpdate`

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualNodeUpdate`

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualNode` & `VirtualNodeUpdate`

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualNodeUpdateCondition`

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualNode` & `VirtualNodeUpdate`

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualNodeHashCode`

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualNode` & `VirtualNodeHashCode`

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualNodeMemberwiseClone`

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualNode` & `VirtualNodeMemberwiseClone`

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualNodeReferenceEquals`

([AkiraNetwork.VirtualStorageLibrary](#))

↳ `VirtualNode` & `VirtualNodeReferenceEquals`

([AkiraNetwork.VirtualStorageLibrary](#))

`VirtualNodeType`

([AkiraNetwork.VirtualStorageLibrary](#))

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

Syntax

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



Constructors

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

VirtualItem(VirtualNodeName)

PatternMatch

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html

Type

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

2.html

VirtualItem(VirtualNodeName, DateTime)

(AkiraNetwork.VirtualStorageLibrary)

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

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

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

1.html

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

Type

1.html

VirtualNode

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

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

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

VirtualItem(VirtualNodeName, DateTime, DateTime)

(AkiraNetwork.VirtualStorageLibrary)

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

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

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

| Type | Name | Description |
|--------------------|-------------|--------------------------------|
| VirtualException | name | The name of the item. |
| VirtualNodeContext | createdDate | The creation date of the item. |

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

```
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)

recursive

In the case of items, this parameter is ignored.

VirtualNode
(AkiraNetwork.VirtualStorageLibrary

Returns

VirtualNodeContext

| Type | Description |
|------------------------------------|-------------|
| AkiraNetwork.VirtualStorageLibrary | |

VirtualNodeExtensions

VirtualNode
(AkiraNetwork.VirtualStorageLibrary

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions

INode.html)

(AkiraNetwork.VirtualStorageLibrary

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary

VirtualNodeDeepClone(bool)

VirtualNodeDeepClone(bool, Exception)

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_DeepClone_1e_System_Boolean_)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary

VirtualNodeTypeFilter

Implements

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

Extension Methods

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

Class VirtualItemAdapter<T>

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)))
↳ 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

VirtualStorageLibrary

Assembly

VirtualStorageLibrary.dll

Syntax

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

public class VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

Type

(AkiraNetwork.VirtualStorageLibrary)

Table

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

T (AkiraNetwork.VirtualStorageLibrary)

1.html

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

Constructors

VirtualItemAdapter(VirtualStorage<T>)

(AkiraNetwork.VirtualStorageLibrary)

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

1.html)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

public VirtualItemAdapter(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 item corresponding to the specified path.

VirtualCycleDetector

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

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

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

Parameters

1.html)

Type VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualPath

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

(AkiraNetwork.VirtualStorageLibrary)

bool)

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

Type VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

1.html)

(AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>)

VirtualItem<T> (https://learn.microsoft.com/dotnet/api/system.virtualitem-1.html)

<T>(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualNode

Exceptions

(AkiraNetwork.VirtualStorageLibrary)

Type VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

InvalidOperationException

(VirtualNodeExtensions)

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)

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

s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

VirtualNodeTypeFilter

Extension Methods

(AkiraNetwork.VirtualStorageLibrary)

VirtualTextFormatter.GenerateSingleTableDebugText<T>(T)

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

s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

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)
↳ VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)
(AkiraNetwork.VirtualStorageLibrary)
↳ VirtualItem<T>
1.html)

Implements

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

Inherited Members

VirtualID
VirtualNode.Name
(AkiraNetwork.VirtualStorageLibrary)
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Name)
VirtualItem
VirtualNode.CreatedDate
(AkiraNetwork.VirtualStorageLibrary)
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_CreatedD
ate)
VirtualNode.UpdatedDate
VirtualItem<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Updated
Date)
VirtualNode.VID
VirtualNode
VirtualNode.IsReferencedInStorage
(AkiraNetwork.VirtualStorageLibrary)
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_IsReferen
cedInStorage)
VirtualNodeContext
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
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))
VirtualNodeListConditions
object.GetHashCode() (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)
VirtualNodeName
object.GetType() (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)
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)

Parameters

VirtualNodeTypeFilter

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

VirtualItem(VirtualNodeName, T?, DateTime)

PatternMatch

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

VirtualCycleDetector

Declaration

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

Parameters

1.html)

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

T VirtualID

(AkiraNetwork.VirtualStorageLibrary.VirtualID.html)

VirtualItem

DateTime
(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html)

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

(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter.html)

1.html)

VirtualItem<T>

VirtualItem(VirtualNodeName, T?, DateTime, DateTime)

1.html)

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

VirtualNode

Declaration

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

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html)

VirtualNodeListConditions

Parameters

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html)

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

VirtualNodeNameType

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNameType.html)

VirtualNodeTypeFilter

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html)

| Type | Name | Description |
|--|--------------------|---|
| T | <i>itemData</i> | The data associated with this item. It is user-defined and can hold any type of data. |
| DateTime (https://learn.microsoft.com/dotnet/api/system.datetime) | <i>createdDate</i> | The creation date of the item. |
| DateTime NotifyNodeDelegate (https://learn.microsoft.com/dotnet/api/system.datetime) PatternMatch (AkiraNetwork.VirtualStorageLibrary.VirtualNodeDelegate.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeDelegate_PatternMatch) VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html) | <i>updatedDate</i> | The last update date of the item. |

Properties

ItemData
(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html#AkiraNetwork_VirtualStorageLibrary_VirtualItem_ItemData)

Gets or sets the data associated with this item. This value can be null.
VirtualException

Declaration
(AkiraNetwork.VirtualStorageLibrary.VirtualItemCondition.html#AkiraNetwork_VirtualStorageLibrary_VirtualItemCondition_Declaration)
VirtualGroupCondition<T, TKey>
(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition.html#AkiraNetwork_VirtualStorageLibrary_VirtualGroupCondition_Declaration)

VirtualID

PropertyValue
(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html#AkiraNetwork_VirtualStorageLibrary_VirtualItem_PropertyValue)

| Type | Description |
|------|---|
| T | (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter.html#AkiraNetwork_VirtualStorageLibrary_VirtualItemAdapter_Declaration) VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter.html#AkiraNetwork_VirtualStorageLibrary_VirtualItemAdapter_Declaration) |

NodeType

Declaration
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Declaration)

Gets the type of the node.

VirtualNode
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeContext_Declaration)
public override VirtualNodeType NodeType { get; }
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeContext_Declaration)
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeExtensions_Declaration)

PropertyValue
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_PropertyValue)

| Type | Description |
|---|-------------|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeType_Declaration) | |

Overrides
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.NodeType.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_NodeType_Declaration)
VirtualNode.nodeType
(AkiraNetwork.VirtualStorageLibrary.VirtualNode.nodeType.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_nodeType_Declaration)
e) VirtualNodeTypeFilter
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeTypeFilter_Declaration)

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. |

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)

(AkiraNetwork.VirtualStorageLibrary.VirtualItem.html#AkiraNetwork_VirtualStorageLibrary_VirtualItem_DeepClone_System_Boolean)

(AkiraNetwork.VirtualStorageLibrary.

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.

Dispose()

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

(AkiraNetwork.VirtualStorageLibrary.

Declaration

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.

public void Dispose()

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.

Dispose(bool)

VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary)

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

(AkiraNetwork.VirtualStorageLibrary.

Declaration

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary.

protected virtual void Dispose(bool disposing)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.

Parameters

VirtualNodeTypeFilter

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

▼

~VirtualItem()

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

Declaration

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

Tostring()

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

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

Returns

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

Description

A string that represents the current object.

Overrides

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

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

Update(VirtualNode)

VirtualNode

Updates the specified node's data.

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

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

VirtualNodeListConditions
([AkiraNetwork.VirtualStorageLibrary](#))

Parameters
VirtualNodeName

Type
([AkiraNetwork.VirtualStorageLibrary](#))

Name

Description

VirtualNodeNotFoundException
([AkiraNetwork.VirtualStorageLibrary](#))

node

The node containing the data to update.

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

VirtualNodeType
([AkiraNetwork.VirtualStorageLibrary](#))

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.VirtualNodeName.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeName_Implicit_operator_VirtualItem_T_VirtualNodeName)

Implicitly converts the specified node name to a VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html)
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeName_Implicitly_converts_the_specified_node_name_to_a_VirtualItem_T)

VirtualException
 (AkiraNetwork.VirtualStorageLibrary.VirtualException.html#AkiraNetwork_VirtualStorageLibrary_VirtualException)

```
public static implicit operator VirtualItem<T>(VirtualNodeName name)
  VirtualGroupCondition<T, TKey>
  (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeName_Implicit_operator_VirtualItem_T_VirtualNodeName)
```

VirtualID
 (AkiraNetwork.VirtualStorageLibrary.VirtualID.html#AkiraNetwork_VirtualStorageLibrary_VirtualID)

VirtualNodeName
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeName)
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeName_Implicit_operator_VirtualItem_T_VirtualNodeName)

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

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

<T>VirtualNode
 (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_T)
 (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode_Implicit_Conversion_of_VirtualNode_to_VirtualItem_T)

VirtualNodeContext
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeContext)

VirtualNodeExtensions
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeExtensions)

VirtualNodeListConditions
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeListConditions)

VirtualNotFoundException
 (AkiraNetwork.VirtualStorageLibrary.VirtualNotFoundException.html#AkiraNetwork_VirtualStorageLibrary_VirtualNotFoundException)

VirtualNodeType
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeType)

VirtualNodeTypeFilter
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeTypeFilter)

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

Converts a tuple (nodeName, itemData) to a VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html)
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeName_Implicit_operator_VirtualItem_T_VirtualNodeName_nodeName_T_itemData)

VirtualItem<T>((VirtualNodeName nodeName, T? itemData) tuple)
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeName_Implicit_operator_VirtualItem_T_VirtualNodeName_nodeName_T_itemData)

VirtualNodeName
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeName)

VirtualNotFoundException
 (AkiraNetwork.VirtualStorageLibrary.VirtualNotFoundException.html#AkiraNetwork_VirtualStorageLibrary_VirtualNotFoundException)

VirtualNodeType
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeType)

VirtualNodeTypeFilter
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeTypeFilter)

VirtualNodeName
 (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeName)

| 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.VirtualItem
VirtualItemAdapter<T>

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

Parameters
item<T>

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

Returns
VirtualNodeContext

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

Implements
VirtualNodeMetadatable
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType
VirtualNodeNotNotFoundException
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType-1.html)

IVirtualDeepCloneable
(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\(system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object)))
VirtualException
object.Equals(object) ([https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals\(system-object\)](https://learn.microsoft.com/dotnet/api/system.object.equals#system-object-equals(system-object)))
VirtualGroupCondition<TKey>
object.GetHashCode () (<https://learn.microsoft.com/dotnet/api/system.object.gethashcode>)
object.GetType () (<https://learn.microsoft.com/dotnet/api/system.object.gettype>)
object.MemberwiseClone () (<https://learn.microsoft.com/dotnet/api/system.object.memberwiseclone>)
object.ReferenceEquals(object, object) (<https://learn.microsoft.com/dotnet/api/system.object.referenceequals>)
object.ToString () (<https://learn.microsoft.com/dotnet/api/system.object.tostring>)

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 VirtualNodeNameException VirtualNodeName name)

(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeTypeFilter

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

VirtualNode(VirtualNodeName, DateTime)

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

Declaration

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

Parameters

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

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

VirtualNode(VirtualNodeName, DateTime, DateTime)

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

Declaration

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

Parameters

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

Properties

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

VirtualID

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

Name

(VirtualItemAdapter<T>)

(AkiraNetwork.VirtualStorageLibrary)

Gets the name of node.

1.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

```
public VirtualNodeName Name { get; }
```

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

Property Value

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

VirtualNodeExtensions

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

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

Gets [VirtualNodeNotFoundException](#).

(AkiraNetwork.VirtualStorageLibrary)

Declaration

(VirtualNodeType)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

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

Property Value

| Type | Description |
|--|-------------|
| VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html) (AkiraNetwork.VirtualStorageLibrary NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary PatternMatch (AkiraNetwork.VirtualStorageLibrary 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 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 VirtualData (AkiraNetwork.VirtualStorageLibrary.VirtualID.html) (AkiraNetwork.VirtualStorageLibrary VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary VirtualNodeName (AkiraNetwork.VirtualStorageLibrary DeepClone(bool) 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. | |

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

[VirtualDirectoryAdapter<T>](#)

(AkiraNetwork.VirtualStorageLibrary)

1.html)

[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 |
|---|----------------------|-----------------|
| VirtualItem (AkiraNetwork.VirtualStorageLibrary) | node | Value to update |

1.html)

[VirtualNode](#)

(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.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode

Extensions.ResolveNodeType_AkiraNetwork_VirtualStorageLibrary_VirtualNode_)

[VirtualNodeType](#)

(AkiraNetwork.VirtualStorageLibrary)

[VirtualNodeTypeFilter](#)

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.html#AkiraNetwork_VirtualStorageLibrary_VirtualNode

Extensions.ResolveNodeType_AkiraNetwork_VirtualStorageLibrary_VirtualNode_)

[VirtualNodeType](#)

(AkiraNetwork.VirtualStorageLibrary)

[VirtualNodeTypeFilter](#)

▼

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

Class VirtualNodeContext

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

Inheritance (AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

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

↳ VirtualNodeContext

PatternMatch

Inherited Members (AkiraNetwork.VirtualStorageLibrary)

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>)

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

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

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

2.html)

Syntax VirtualID

(AkiraNetwork.VirtualStorageLibrary)

public class VirtualNodeContext

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualItem<T>

VirtualNodeContext(VirtualNode?, VirtualPath, VirtualDirectory?, int, int,

VirtualPath?, bool, VirtualSymbolicLink?)

VirtualNode

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

(AkiraNetwork.VirtualStorageLibrary)

class VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

public VirtualNodeContext(VirtualNode? node, VirtualPath traversalPath, VirtualDirectory? pa

rentNode = null, int depth = 0, int index = 0, VirtualPath? resolvedPath = null, bool resolv

(AkiraNetwork.VirtualStorageLibrary)

ed = false, VirtualSymbolicLink? resolvedLink = null)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

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

Properties

Depth

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

VirtualNodeExtensions

Declaration

```
public int Depth { get; set; }
```

VirtualNodeName

Property Value

VirtualNodeNotFoundException

Type

AkiraNetwork.VirtualStorageLibrary.VirtualNodeType
(https://learn.microsoft.com/dotnet/api/system.int32?view=dotnet-plat-ext-2.2)

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  
     (AkiraNetwork.VirtualStorageLibrary.  
      Type patternMatch  
      (AkiraNetwork.VirtualStorageLibrary.  
       nt (https://learn.microsoft.com/dotnet/  
        VirtualCycleDetector
```

| Type | Description |
|---|------------------------|
| <code>(AkiraNetwork.VirtualStorageLibrary.PatternMatch int (https://learn.microsoft.com/dotnet/api/system.int32) VirtualCycleDetector)</code> | The index of the node. |

Nodes

VirtualDirectoryAdapter<T>
Gets the node and can only be set within the assembly.
(AkiraNetwork.VirtualStorageLibrary)

Declaration¹)

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

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

ParentDirectory

VirtualItem [T](#)
Gets the parent directory of the node and can only be set within the assembly.
([Assembly](#), [Node](#), [VirtualStorage](#), [Library](#))

(AkiraN
Declaration)

```
VirtualNode n(MultiNodetVirtualStorageabitentDirectory { get: }
```

| Property | Value |
|---------------------------|--|
| Type | Description |
| VirtualNodeContext | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeExtensions | (AkiraNetwork.VirtualStorageLibrary) |
| VirtualNodeListConditions | (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) |
| VirtualDirectory | (AkiraNetwork.VirtualStorageLibrary) |

Perspektiv

VirtualNodeType (AkiraNetwork.VirtualStorageLibrary) Gets a value indicating whether the sym

Declaration VirtualNodeTypeFilter
(AkiraNetwork.VirtualStorageLibrary)

```
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) | The resolved symbolic link, or null. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) | |

ResolvedPath

VirtualItem

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

VirtualItemAdapter<T>

Declaration

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

1.html)

VirtualNode

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

Virt

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

VirtualnodeName

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

(AkiraNetwork.VirtualStorageLibrary.TraversalPath.html)

TraversalNotFoundException

(AkiraNetwork.VirtualStorageLibrary.TraversalPath)

VirtualNodeType

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

(AkiraNetwork.VirtualStorageLibrary.TraversalPath)

VirtualNodeTypeFilter

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.

Overrides

2.html

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

(AkiraNetwork.VirtualStorageLibrary)

Extension Methods

(AkiraNetwork.VirtualStorageLibrary)

VirtualTextFormatterAdapter.GenerateSingleTableDebugText<T>(T)

(AkiraNetwork.VirtualStorageLibrary) [VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilities_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_](#)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Class VirtualNodeExtensions

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

Filter by title

Inheritance

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

Inherited Members

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

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

public static VirtualNodeType ResolveNodeType(this VirtualNode node)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

Parameters

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

Type

VirtualNodeNotFoundException

Name

Description

Virt

node

The virtual node to resolve the type for.

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Returns

| Type | Description |
|--|---|
| VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html) ▼ (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary.PatternMatch (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualID VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualNode VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter | The type of the node or, if the node is a symbolic link, the type of the target node. |

Struct VirtualNodeListConditions

Represents the conditions for creating a list of virtual nodes.

Filter by title

Inherited Members

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

NameSpace AkiraNetwork.VirtualStorageLibrary (AkiraNetwork.VirtualStorageLibrary.html)

Assembly!VirtualStorageLibrary.dll

(AkiraNetwork.VirtualStorageLibrary)

Syntax

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

Constructors

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

VirtualNodeListConditions()

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

Declaration

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

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

VirtualNodeListConditions(VirtualNodeTypeFilter,

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<VirtualNode, object>?,

(VirtualNodeExtensions)

List<VirtualSortCondition<VirtualNode>>?)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions(VirtualNodeListConditions)

(AkiraNetwork.VirtualStorageLibrary.VIRTUALNODELISTCONDITIONS)

```
VirtualNodeName
```

Declaration

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

Parameters

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

Properties

VirtualStorageLibrary

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary
2.html)

VirtualID

Specifies the filter criteria, determining which types of virtual nodes to include.

(AkiraNetwork.VirtualStorageLibrary
2.html)

Declaration

VirtualItem

(AkiraNetwork.VirtualStorageLibrary
1.html)

VirtualItemAdapter<T> TypeFilter Filter { readonly get; set; }

(AkiraNetwork.VirtualStorageLibrary
1.html)

Property Value

VirtualItem<T>

| Type | Description |
|--|--|
| 1.html) | |
| VirtualNodeTypeFilter | The filter specifying the types of virtual nodes. The default value is |
| VirtualNode | All |
| (AkiraNetwork.VirtualStorageLibrary ary.VirtualNodeTypeFilter.html) | (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html#AkiraNetwor k.VirtualStorageLibrary_VirtualNodeTypeFilter_All) |
| VirtualNodeContext | |
| (AkiraNetwork.VirtualStorageLibrary VirtualNodeExtensions | . |
| VirtualNodeExtensions | |
| (AkiraNetwork.VirtualStorageLibrary VirtualNodeListConditions | |
| VirtualNodeListConditions | |
| (AkiraNetwork.VirtualStorageLibrary VirtualnodeName | |

GroupCondition

VirtualStorageLibrary

VirtualGroupCondition<VirtualNode, object>? GroupCondition { readonly get; set; }

(AkiraNetwork.VirtualStorageLibrary
VirtualNodeTypeFilter

VirtualNodeType

public VirtualGroupCondition<VirtualNode, object>? GroupCondition { readonly get; set; }

(AkiraNetwork.VirtualStorageLibrary
VirtualNodeTypeFilter

| Type | Description |
|--|---|
| VirtualGroupCondition (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition-2.html)  | 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. |
| VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) , object (https://learn.microsoft.com/dotnet/api/system.object) (AkiraNetwork.VirtualStorageLibrary) | |

SortConditions

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

Declaration (Akiran)

```
VirtualDirectoryAdapter<T>
(AkiraNetwork.VirtualStorageLibrary
public List<VirtualSortCondition<VirtualNode>>? SortConditions { readonly get; set; }
1.html)
```

VirtualException

Property Value

| Type | Description |
|--|---|
| VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.List-2.html) (https://docs.microsoft.com/dotnet/api/system.collections.generic.list-1) AkiraNetwork.VirtualStorageLibrary.VirtualItem < VirtualSortCondition (AkiraNetwork.VirtualStorageLibrary.VirtualSortCondition-1.html) < (AkiraNetwork.VirtualStorageLibrary.VirtualNode) VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) >> (AkiraNetwork.VirtualStorageLibrary. | 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. |

Virtual Storage (Akira Network Virtual Storage)

```
VirtualTextFormatter GenerateSingleTableDebugText<T>(T)
(AkiraNetwork_VirtualStorageLibrary)
(AkiraNetwork_VirtualStorageLibrary.Uses.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilitie
s_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)
(AkiraNetwork_VirtualStorageLibrary)
```

```
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)

Pattern Match Indicates whether the node is the root node

(AkiraNetwork VirtualStorage) library

Declaration cycleDetector

(AkiraNetwork VirtualStorage library)

```
nVirtualDirectoriesRoot { get: }
```

(AkiraNetwork VirtualStorage) library

Property Value (AkiraNetwork.VirtualStorageLibrary)

| Type.html | Description |
|---|---|
| VirtualException bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork VirtualStorage Library) | true if this is the root node; otherwise, false . |

Name _____

VirtualID

Gets the name of the node.

(AKTA NETWORK.VIRTUAL STORAGE LIBRARY)

Declaration

(AkiraNetwork.VirtualStorageLibrary

```
VirtualItemAdapter<T>
```

```
public string Name { get; }  
(AkiraNetwork.VirtualStorageLibrar
```

1.html)

Property<T>

| Type | (AkiraNetwork.VirtualStorageLibrary.html) | Description |
|--------|---|------------------------|
| string | https://learn.microsoft.com/dotnet/api/system.string | The name of this node. |

Methods

(AkiraNetwork VirtualStorage Library)

VirtualNodeName

Com(AkiraNetwork)VirtualStorageLibrary VirtualNodeName

(AkiraVirtualNodeNotFoundException(VirtualNodeName.html)).

(AkiraNetwork.VirtualStorageLibrary

Declaration

Virtuaalide type

VirtualNetwork.virtual

```
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. |

Returns

| Type | Description |
|--|--|
| VirtualNodeContext int (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html) VirtualNodeListConditions (https://learn.microsoft.com/dotnet/api/system.int32) (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html) | A value indicating the relative order of the instances being compared. |

Exceptions

| Type | Condition |
|--|---|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) VirtualNodeNotFoundException (http://AkiraNetwork.VirtualStorageLibrary/api/system.argumentOutOfRangeException VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) 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

```
1.html)  
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) | obj The object to compare with this instance. |

Returns

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

| Type | Description |
|--|--|
| VirtualNodeName bool (AkiraNetwork.VirtualStorageLibrary. (https://learn.microsoft.com/dotnet/api/system.boolean) VirtualNodeNotFoundException VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | true if the specified object is equal to this instance; otherwise, false . |

Overrides

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

VirtualNodeTypeFilter
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

`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 (Akira)

VirtualCycleDetector

| Type | | Name | Description |
|--|--|--------|-------------------------------|
| VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary) | string (https://learn.microsoft.com/dotnet/api/system.string) (Akira Network Virtual Storage Library) | prefix | The prefix for the node name. |

Virt Returns

(AkiraNetwork VirtualStorage Library)

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

14

VirtualGroupCondition<T, TKey>

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.

Exceptionalism

VirtualItem

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

GetHashCode()

`VirtualNode`
Returns a hash code for this instance.
(AkiraNetwork VirtualStorage library)

Declaration

`VirtualNodeContext`
(Ali-Nat-SubVirtualContext.h)

(AkiraNetwork.VirtualStorageLibrary)

~~VirtualNodeExtensions~~.GetHashCode

(AkiraNetwork.VirtualStorageLibrary
VirtualNodeListConditions
Returns: NetworkVirtualStorageLibrary

| Type | Description |
|--|--------------------------------|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName int (https://learn.microsoft.com/dotnet/api/system.int32) VirtualNodeNotFoundException | A hash code for this instance. |

(Akira)

www.aktanetwork.org

`list(G, IN, LG, th, f, l, m, i, s, st, d, t, t2, t3, t4, t5, t6, t7, t8, t9, t10, t11)`

©AkiraNetwork. All rights reserved.

IsValidNodeName(VirtualNodeName)

Checks if a node name is valid.

Declaration

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

Parameters

NotifyNodeDelegate

Type AkiraNetwork.VirtualStorageLibrary

VirtualNodeName

(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

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeCondition<T> ToString()

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

Returns (AkiraNetwork.VirtualStorageLibrary)

Type VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

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

VirtualnodeName

Description

The name of the node as a string.

Overrides

VirtualNodeNotFoundException

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

```
public static bool operator ==(VirtualnodeName? left, VirtualnodeName? right)
(AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html)
    NotifyNodeDelegate
```

Parameters

| Type | Name | Description |
|--|-------|---|
| atternMatch (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) VirtualDirectoryName.html VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryName.html) VirtualDirectoryAdapter<T> VirtualnodeName (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) VirtualnodeNameException (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey>) | left | The left VirtualnodeName (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) to compare. |
| VirtualnodeName (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>) VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>) | right | The right VirtualnodeName (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) to compare. |

Returns

| Type | Description |
|--|---|
| VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualID.html) | true if the specified instances are equal; otherwise, false . |

implicit operator string(VirtualnodeName)

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

Declaration

```
public static implicit operator string(VirtualnodeName nodeName)
(AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html)
    VirtualNodeExtensions
```

Parameters

| Type | Name | Description |
|--|----------|---|
| VirtualnodeName (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) VirtualnodeNameException (AkiraNetwork.VirtualStorageLibrary.VirtualnodeNameException.html) VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html) VirtualNodeTypeFilter (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | nodeName | The VirtualnodeName (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.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)
VirtualItem

operator !=(VirtualNodeName?, VirtualNodeName?)

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

(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

(AkiraNetwork.VirtualStorageLibrary)

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

↳ PatternMatch

↳ VirtualNodeNotFoundException

↳ VirtualStorageLibrary

Implements

(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>)

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

ExceptionGroupCondition (<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>)

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

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

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

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

VirtualItemAdapter ([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)))

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

VirtualNodeContext

Assembly: VirtualStorageLibrary.dll

(AkiraNetwork.VirtualStorageLibrary)

Syntax

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

↳ **VirtualNodeListConditionsNotFoundException** : 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

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

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

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

VirtualDirectoryAdapter<T>

Parameters

1.html)

Type VirtualException

Name

Description

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

message

The error message.

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

2.html)

VirtualNodeNotFoundException(string, Exception)

(AkiraNetwork.VirtualStorageLibrary)

Initializes a new instance of the VirtualNodeNotFoundException

VirtualItem

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

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

VirtualItemAdapter<T>

Declaration

1.html)

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

1.html)

Parameters

VirtualNode

Type (AkiraNetwork.VirtualStorageLibrary)

Name

Description

VirtualNodeContext

message

The error message.

string (AkiraNetwork.VirtualStorageLibrary)

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

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

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

(AkiraNetwork.VirtualStorageLibrary)

Exception

innerException

The exception that is the cause of this exception.

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Implements

Extension Methods

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

▼

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

Enum VirtualNodeType

Specifies the types of virtual nodes.

Filter by title

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

Assembly: VirtualStorageLibrary.dll
NotifyNodeDelegate

Syntax
(AkiraNetwork.VirtualStorageLibrary)

PatternMatch

public enum VirtualNodeType

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

Fields

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

VirtualTextFormatter.GenerateSingleTableDebugText<VirtualNodeType>(VirtualNodeType)

(AkiraNetwork.VirtualStorageLibrary)
(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utilities_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_)

VirtualTextItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html

VirtualNode

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

Extension Methods

Enum VirtualNodeTypeFilter

Specifies the filters for virtual node types.

Filter by title

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

(AkiraNetwork.VirtualStorageLibrary)

Assembly: VirtualStorageLibrary.dll

NotifyNodeDelegate

Syntax

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

Fields

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

Extension Methods

1.html
VirtualTextFormatter.GenerateSingleTableDebugText<VirtualNodeTypeFilter>(VirtualNodeTypeFilter)

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

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| 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>

Declaration

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

PropertyValue

| Type | Description |
|--------------------|--|
| VirtualNodeContext | An integer representing the depth of the path. |

DirectoryPath

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

VirtualNodeName

Gets the directory part of the path.

(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

Declaration

```
VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.VirtualPath)
VirtualNodeType { 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

Property Value

VirtualNodeType

(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

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

```
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)

IsEmpty
VirtualItemAdapter<T>
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
VirtualNodeNotFoundException
Gets a value indicating whether the instance's path ends with a slash.
(AkiraNetwork.VirtualStorageLibrary)

Declaration
NodeType
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter
(AkiraNetwork.VirtualStorageLibrary)

```
public bool IsEndsWithSlash { get; }
```

Property Value

| Type | Description |
|--|---|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibraryExtensions.NotifyNodeDelegate<T>) | true if the instance's path ends with a slash; otherwise, false . |

IsRoot

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

Declaration
(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibraryExtensions.IsRoot<T>)

VirtualDirectory
VirtualCycleDetector
(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibraryExtensions.VirtualCycleDetector<T>)

VirtualException
(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibraryExtensions.VirtualException<T>)

Property Value
(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibraryExtensions.IsRoot<T>)

| Type | Description |
|---|--|
| virtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibraryExtensions.VirtualGroupCondition<T, TKey>) | true if the instance's path is the root path; otherwise, false . |

NodeName

Gets the node name of the path.

Declaration
(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibraryExtensions.NodeName<T>)

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

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

VirtualNode
(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibraryExtensions.VirtualNode<T>)

VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibraryExtensions.VirtualNodeContext<T>)

| Type | Description |
|---|--------------------|
| VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibraryExtensions.VirtualNodeExtensions) | A VirtualNodeName. |

VirtualNodeName
(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibraryExtensions.VirtualNodeName)

VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibraryExtensions.VirtualNodeNotFoundException)

PartsList
(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibraryExtensions.PartsList<T>)

Gets the list of parts that make up the path.

VirtualNodeTypeFilter

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)`

VirtualNodeListConditions
Type: `(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName)`

| Type | Description |
|---|--------------------------------------|
| <code>string(https://learn.microsoft.com/dotnet/api/system.string)</code> | A string representing the root path. |

Separator

VirtualNodeType
Type: `(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 |
| bool | (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary.VirtualItem) | 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) (AkiraNetwork.VirtualStorageLibrary.VirtualItem) | 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)
```

Parameters

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

| 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. | | |
| string (https://learn.microsoft.com/dotnet/api/system.string)[] | paths | An array of path strings to combine. |

Returns

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.

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

(AkiraNetwork.VirtualStorageLibrary.

1.html)

VirtualNode

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

Parameters

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeNotFoundException

(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
(AkiraNetwork.VirtualStorageLibrary)

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)

Parameters

(AkiraNetwork.VirtualStorageLibrary)

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

CompareTo(VirtualNode?)

(AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|--|--|
| VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary) | An integer indicating the relative order of the paths. |

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

VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeName
(AkiraNetwork.VirtualStorageLibrary)

CompareTo(VirtualStorageLibrary) other object.

VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary)

public int CompareTo(object? obj)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter
(AkiraNetwork.VirtualStorageLibrary)

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 (https://learn.microsoft.com/dotnet/api/system.int32) | An integer indicating the relative order of the paths. |

Exceptions

[PatternMatch](#)

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

VirtualEvent

[VirtualPath](#)

Indicates whether this path is equal to the specified VirtualPath object ([AkiraNetwork.VirtualStorageLibrary.VirtualPath.html](#)) object.

Declaration

[VirtualID](#)

([AkiraNetwork.VirtualStorageLibrary.VirtualItem](#))
public bool Equals(VirtualPath? other)

VirtualItem

[VirtualItemAdapter<T>](#)

| Type | Name | Description |
|-----------------------------|--------------|---|
| VirtualPath | <i>other</i> | The VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) object to compare with the instance's path. |

Returns

[VirtualNodeContext](#)

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

VirtualNodeName

[VirtualStorageLibrary](#)

Indicates whether this path is equal to the specified object.

Declaration

[VirtualNodeTypeFilter](#)

```
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)

Overrides

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

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

Returns

the hash code for this path.

VirtualID

Declaration
(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

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

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

Returns

1.html)

Type
VirtualItem<T>

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

1.html)

Description

The hash code for this path.

Overrides

(AkiraNetwork.VirtualStorageLibrary)

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

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

Returns

VirtualNodeList<Condition> from the instance's path.

(AkiraNetwork.VirtualStorageLibrary)

Declaration
VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

```
public VirtualNodeName GetNodeName()
```

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

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

GetRelativePath(VirtualPath)

VirtualGroupCondition<T, TKey>

Gets ([AleinNetPath](#)) from the specified base path.

2.html)
Declaration
Virtual

```
(AkiraNetwork.VirtualStorageLibrary  
public VirtualPath GetRelativePath(VirtualPath basePath)
```

(AkiraNetwork.VirtualStorageLibrary
Virtual Library Adapter Test)

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

[\(AkıraN
1.html\)](#)

Returns
VirtualNode

| Type | Description |
|--|--|
| <code>AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext</code> <code>(AkiraNetwork.VirtualStorageLibrary.VirtualPath)</code> <code>(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions)</code> <code>(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)</code> | A VirtualPath (<code>AkiraNetwork.VirtualStorageLibrary.VirtualPath.html</code>) representing the relative path. |

VirtualM
Exceptions

(AkiraNetwork.VirtualStorageLibrary
VirtualNodeName)

virtualNodeName

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

| 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, T)

```
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

```
public VirtualPath NormalizePath()
```

Returns

| Type | Description |
|---|--|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) VirtualNodeDelegate | A new VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) instance with the normalized path. |

NormalizePath(string)

Normalizes the specified path string.
VirtualDirectory

Declaration
(AkiraNetwork.VirtualStorageLibrary)

```
VirtualDirectoryAdapter<T>  
p(AkiraNetwork.VirtualStorageLibrary.Path(string path)  
1.html)
```

Parameters
(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|---|------|-------------------------------|
| VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary string (https://learn.microsoft.com/dotnet/api/system.string) 2.html) | path | The path string to normalize. |

VirtualID

Returns
(AkiraNetwork.VirtualStorageLibrary)

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

Exceptions
(VirtualItem)

| Type | Condition |
|---|--|
| VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary InvalidOperationException (https://learn.microsoft.com/dotnet/api/system.invalidoperationexception) VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary VirtualNodeListCondition Determines whether the instance's path starts with the specified path. (AkiraNetwork.VirtualStorageLibrary VirtualNodeName (AkiraNetwork.VirtualStorageLibrary VirtualNodeSortIndex<VirtualPath path> (AkiraNetwork.VirtualStorageLibrary VirtualNodeType (AkiraNetwork.VirtualStorageLibrary VirtualNodeTypeFilter | Thrown when attempting to navigate above the root directory. |

StartsWith(VirtualPath)

(AkiraNetwork.VirtualStorageLibrary)

Declaration
(VirtualNodeSortIndex<VirtualPath path>

```
VirtualNodeSortIndex<VirtualPath path>
```

(AkiraNetwork.VirtualStorageLibrary
VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary
VirtualNodeTypeFilter

Parameters
(AkiraNetwork.VirtualStorageLibrary
VirtualNodeTypeFilter

VirtualNodeTypeFilter

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

Returns

(AkiraNetwork.VirtualStorageLibrary.

Type

NotifyNodeDelegate

Description

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

true if the instance's path starts with the specified path;
otherwise, false.

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary.

VirtualDirectory

Tostring()

(AkiraNetwork.VirtualStorageLibrary.

VirtualDirectoryAdapter<T>

Returns a string that represents the instance's path.

(AkiraNetwork.VirtualStorageLibrary.

VirtualException

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

(AkiraNetwork.VirtualStorageLibrary.

Returns

VirtualID

Type

(AkiraNetwork.VirtualStorageLibrary.

Description

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

A string that represents the instance's path.

(AkiraNetwork.VirtualStorageLibrary.

Overrides

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.

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.

(AkiraNetwork.VirtualStorageLibrary.

Declaration

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary.
public VirtualPath TrimEndSlash()
VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.

Returns

VirtualNodeListConditions

Type

(AkiraNetwork.VirtualStorageLibrary.

Description

VirtualNodeName

Type

(AkiraNetwork.VirtualStorageLibrary.

Description

VirtualPath

Type

(AkiraNetwork.VirtualStorageLibrary.

Description

(AkiraNetwork.VirtualStorageLibrary.

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

VirtualNodeType

Type

(AkiraNetwork.VirtualStorageLibrary.

Description

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. (AkiraNetwork.VirtualStorageLibrary.VirtualPath VirtualDirectoryAdapter<T> h.html) (AkiraNetwork.VirtualStorageLibrary. 1.html) VirtualException | path | The VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.htm l) . . . |
| AkiraNetwork.VirtualStorageLibrary. VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNo deName.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. 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  
    public static VirtualPath operator +(VirtualPath path1, VirtualPath path2)  
    (AkiraNetwork.VirtualStorageLibrary.  
        VirtualNodeListConditions
```

Parameters

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

| Type | Name | Description |
|---|--------------|---|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>path1</i> | The first VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) . |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary.PatternMatch) | <i>path2</i> | The second VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) . |

Returns (AkiraNetwork.VirtualStorageLibrary.VirtualPath)

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

VirtualPathDirectory
(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

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

VirtualException
operator + (VirtualPath, char)
(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

VirtualGroupCondition<T, TKey>
Combines a VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) with a character.
(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

Declaration

```
VirtualID
p(AkiraNetwork.VirtualStorageLibrary.VirtualPath) operator +(VirtualPath path, char chr)
```

VirtualItem

VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualItem)

| Type | Name | Description |
|---|-------------|--|
| VirtualPath VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) | <i>path</i> | The VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) . |
| VirtualNode char (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContent.com.dotnet/api/system.char) | <i>chr</i> | The character to combine with the path. |

VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions)

Returns
VirtualNodeListConditions
VirtualNodeName
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName)

VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException)

VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType)

VirtualNodePermission
(AkiraNetwork.VirtualStorageLibrary.VirtualNodePermission)

operator + (VirtualPath, string)
(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

VirtualPath
(AkiraNetwork.VirtualStorageLibrary.VirtualPath)

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

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) (AkiraNetwork.VirtualStorageLibrary) VirtualCycleDetector | <i>path</i> The VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) . . |
| string VirtualDirectory (https://learn.microsoft.com/dotnet/api/system). string (AkiraNetwork.VirtualStorageLibrary) VirtualIDirectoryAdapter.T | <i>str</i> The string to combine with the path. |

Returns VirtualDirectoryAdapter<|>
(AkiraNetwork.VirtualStorageLibrary)

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

```
2.html)  
VirtualID  
(AkiraNetwork.VirtualStorageLibrary  
operator<(char, VirtualPath)
```

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

Declaration (AkiraNetwork.VirtualStorageLibrary-1.html)

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

Parameter Node

| Type | Description |
|--|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType) | |
| VirtualNodeContext | |
| char | The character to combine with the path. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPathExtension) | |
| (http://schemas.microsoft.com/dotnet/api/system.char) | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath) | |
| VirtualNodeListConditions | |
| VirtualPath | The |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath) | VirtualPath |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath.VirtualnodeName) | (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath) | . |
| VirtualNodeNotFoundException | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath) | |
| Returns | |
| VirtualNodeType | |
| (AkiraNetwork.VirtualStorageLibrary.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) | str | The string to combine with the path. |
| string (html) (https://learn.microsoft.com/dotnet/api/system.net.http/htmlexception) string (AkiraNetwork.VirtualStorageLibrary) | path | The VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) |

VirtualItem

Returns
(AkiraNetwork.VirtualStorageLibrary)

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

VirtualNodeName
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName)

VirtualNodeNotFoundException
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException)

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

(AkiraNetwork.VirtualStorageLibrary.VirtualPath).

VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType)

VirtualNodeTypeFilter
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter)

implicit operator VirtualPath(string)

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

(AkiraNetwork.VirtualStorageLibrary.VirtualPath).

VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType)

VirtualNodeTypeFilter
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter)

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

Parameters

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

Return: NotifyNodeDelegate

| Type | Description |
|---|--|
| (AkiraNetwork.VirtualStorageLibrary.PatternMatch) | A VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) object representing the given string path. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath) | A VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) object representing the given string path. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath) | A VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) object representing the given string path. |

operator !=(VirtualPath?, VirtualPath?)

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

Determines whether two VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) instances are not equal.

Declaración

VirtualGroupCondition<T, TKey>

(AkiraNetwork VirtualStorage library virtualPath) left VirtualPath) right)

pubfile
2.html)

2.11(III)

| Type | Name | Description |
|---|-------|---|
| VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualPath) VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) VirtualItem<T> | left | The left VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | right | The right VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) |

VirtualNodeContext

Returns (AkiraNetwork.VirtualSt

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

(AkiraNetwork.V)

VirtualNodeNotFoundException

|Equitable<T>(https://learn.microsoft.com)

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

IComparable<IClusteredVirtualStorageLibrary> (dotnet/api/system/icomparable)

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

VirtualNodeFilter

Extension Methods

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

Inheritance

NotifyNodeDelegate

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

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

Assembly

[VirtualStorageLibrary.dll](#)

Syntax

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

VirtualItem

public class VirtualSortCondition<T>
↳ [\(AkiraNetwork.VirtualStorageLibrary\)](#)

VirtualItemAdapter<T>

Type

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

| Name | Description |
|------|---|
| T | (AkiraNetwork.VirtualStorageLibrary) The type of the entity to be sorted. |

1.html

VirtualNode

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

VirtualNodeContext

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

VirtualSortCondition(Expression<Func<T, object>>, bool)

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

Initializes a new instance of the VirtualSortCondition<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualSortCondition-1.html) class with the specified sorting property and order

VirtualNodeName

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

VirtualNodeNotFoundException

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

public VirtualSortCondition(Expression<Func<T, object>> sortBy, bool ascending = true)
↳ [VirtualNodeType](#)

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

VirtualNodeTypeFilter

Parameters

| 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) bool VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary) (https://learn.microsoft.com/dotnet/api/system.boolean) VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary) 1.html) | A boolean value indicating whether the sorting order is ascending. |

SortBy

Gets or sets the property used for sorting.

[VirtualNodeContext](#)

Declaration

```
VirtualNodeExtensions
(AkiraNetwork.VirtualStorageLibrary)
public <Func<T>> SortBy { get; set; }
```

[VirtualNodeListConditions](#)

PropertyValue

[VirtualNodeName](#)

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

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)

Methods

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

ApplySortConditions<T>(IEnumerable<T>, List<VirtualSortCondition<T>>?)

VirtualNode

Applies the specified sort conditions to the elements of the source sequence.

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

(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. |

Return(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T>

| 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. |

VirtualID

Type Parameters

| Name | Description |
|-------------|---|
| VirtualItem | (AkiraNetwork.VirtualStorageLibrary.VirtualItem VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T> 1.html)) |

GroupAndSort<T>(IEnumerable<T>, VirtualGroupCondition<T, object>?, List<VirtualSortCondition<T>>?)

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtension

1.html)

Groups and sorts the elements of the source sequence based on the specified conditions.

VirtualNode

Declaration(AkiraNetwork.VirtualStorageLibrary)

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

VirtualNodeListConditions

Parameters(AkiraNetwork.VirtualStorageLibrary)

| 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 |

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>)

VirtualCycleDetector

[] (AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

NodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Gets or sets the conditions for listing nodes.

VirtualException

Declaration (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)

virtual NodeGetDot { 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

(AkiraNetwork.VirtualStorageLibrary)

PathDotDot

VirtualNodeTypeFilter

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

(AkiraNetwork.VirtualStorageLibrary)

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

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](#) [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)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Properties

(AkiraNetwork.VirtualStorageLibrary)

1.html)

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

[] VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

An array of characters that are invalid 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

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

2.html

VirtualNode

public VirtualNode GetRoot();

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

Property Value
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

Type AkiraNetwork.VirtualStorageLibrary

Description

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

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)

VirtualNodeName

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.VirtualWildcardMatcher.html) | The wildcard matcher. |

Methods

SetVirtualNodeListConditions(VirtualNodeListConditions)

Sets the conditions for listing nodes.

Declaration

```
public static void SetVirtualNodeListConditions(VirtualNodeListConditions conditions)
```

Parameters

| Type | Name | Description |
|---------------------------|------------|-----------------------------------|
| VirtualNodeListConditions | conditions | The conditions for listing nodes. |

SetVirtualNodeListConditions(VirtualNodeTypeFilter, VirtualGroupCondition<VirtualNode, object>?, List<VirtualSortCondition<VirtualNode>>?)

Sets the conditions for listing nodes.

Declaration

```
public static void SetVirtualNodeListConditions(VirtualNodeTypeFilter filter, VirtualGroupCondition<VirtualNode, object>? groupCondition = null, List<VirtualSortCondition<VirtualNode>>? sortConditions = null)
```

Parameters

| Type | Name | Description |
|-----------------------|--------|-------------|
| VirtualNodeTypeFilter | filter | |

| 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 VirtualCycleDetector (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

VirtualNode
(AkiraNetwork.VirtualStorageLibrary)
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
(AkiraNetwork.VirtualStorageLibrary)

Properties

CurrentPath

Gets or sets the current virtual path.

Declaration

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

Property Value
`(AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate)`

Type
`(AkiraNetwork.VirtualStorageLibrary.PatternMatch)`

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

Description

The current virtual path

CycleDetectorForTarget

`(AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T>)`
Gets the cycle detector class used for the WalkPathToTarget method.

Declaration

`1.html`

`VirtualException`
`public VirtualCycleDetector CycleDetectorForTarget { get; }`
`(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey>)`

Property Value
`(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector)`

Type
`VirtualID`

`(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector)`

`(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html)`

Description

An instance of the cycle detector class

CycleDetectorForTree

`VirtualItem<T>`
Gets the cycle detector class used for the WalkPathTree method.

Declaration

`1.html`

`VirtualNode`
`public VirtualCycleDetector CycleDetectorForTree { get; }`
`(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext)`

Property Value
`(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions)`

Type
`(AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions)`

`VirtualCycleDetector`

`(AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html)`

`VirtualnodeName`

An instance of the cycle detector class

Dir

`(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotedException)`
Gets the adapter providing operations for virtual directories.

`(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)
Gets the adapter providing operations for virtual symbolic links.
PatternMatch

Declaration (Akira) Network.VirtualStorageLibrary

VirtualCycleDetector

```
(AkiraNetworkVirtualStorageLibrary)
public VirtualSymbolicLinkAdapter<T> Link { get; }
    VirtualDirectory
```

VirtualDirectory (AlvinNetwork)

Property Value `VirtualStorageLibrary`

Properties of individual

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

LinkDictionary
(AkiraNetwork.)

VirtualItem

DeclarationNetwork.VirtualStorageLibrary

VirtualItemAdapter<T>

(Akira Network Virtual Storage Library) HashSet<VirtualPath>> LinkDictionary { get; }
1.html)

VirtualItem<T>

Property Value

(AkranNetwork.VirtualStorageLibrary)

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

Root^(A)

VirtualNodeName

Gets (Allel range) direct virt of the right lab storage.

VirtualNodeNotFoundException

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

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

VirtualNodeTypeFilter

Property Value

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

▼

Methods

| | |
|---|-------------------|
| NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary.NotifyNodeDelegate.html) | |
| PatternMatch (AkiraNetwork.VirtualStorageLibrary.PatternMatch.html) | |
| AddDirectory(VirtualPath, VirtualDirectory, bool) | |
| VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html) | Adds a directory. |
| Declaration VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) | |
| VirtualException Parameters (AkiraNetwork.VirtualStorageLibrary.VirtualException.html) | |

| Type | Name | Description |
|--|----------------------|---|
| virtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey>.html) | directoryPath | The path of the directory to add |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | directory | The VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) to add |
| VirtualItem<T> bool (AkiraNetwork.VirtualStorageLibrary.VirtualItem<T>.html) | createSubdirectories | Indicates whether to create intermediate directories |

AddDirectory(VirtualPath, bool)

| | |
|---|--|
| VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html) | |
| Adds a directory. (AkiraNetwork.VirtualStorageLibrary.AddDirectory.html) | |
| VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html) | |
| Declaration VirtualnodeName (AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html) | |
| public void AddDirectory(VirtualPath path, bool createSubdirectories = false) (AkiraNetwork.VirtualStorageLibrary.AddDirectory.html) | |

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

| 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) (AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions) | <i>createSubdirectories</i> | Indicates whether to create intermediate directories |

AddItem(VirtualPath, VirtualItem<T>, bool)

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

Adds a virtual item.

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

Declaration

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

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

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

1.html)

VirtualException

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

Type VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

2.html)

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualID

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualPath

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualItemAdapter<T>

<T>(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

1.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

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

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualNodeExtensions

Adds an item.

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualNodeName

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

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualNodeTypeFilter

| Type | Name | Description |
|--------------------------------|--------------------------|---|
| VirtualGroupCondition<T, TKey> | <i>itemDirectoryPath</i> | The path of the directory to add the item to |
| VirtualItem | <i>item</i> | The VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) to add |
| bool | <i>overwrite</i> | Indicates whether to overwrite an existing node |

AddItem(VirtualPath, T?, bool)

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

Adds an item.

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualNodeName

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

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.VirtualPathExtensions)

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
Declaration
VirtualCycleDetector

VirtuaCycleDetector
(AkiraNetwork Virtua)

(AkiraNetwork.virtualStorageLibrary
public void AddLinkToDictionary

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary

Parameters

Parameters (AkiraNetwork VirtualStorage library)

Type I_{atmos}D

| Parameters | (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | Name | Description |
|----------------------|---|-------------------|-------------|
| VirtualException | (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>targetPath</i> | |
| VirtualPathCondition | (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>linkPath</i> | |

AddNode(VirtualPath, VirtualNode, bool)

AkiraNetwork.VirtualStorageLibrary
(AkiraNetwork.VirtualStorageLibrary)

Adds VirtualAlloc to the specified directory

Adds or removes the specified

(AkiraNetwork.virtualStorageLibrary
declaration)

“VirtualItemAdapter<T>

(AkiraNetwork.)

```
public void AddNode(VirtualPath
```

Virtuall

| Parameters | | |
|---|--------------------------|--|
| Type | Name | Description |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath)(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext)(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)VirtualNodeExtensions | <i>nodeDirectoryPath</i> | The path of the directory to add the node to |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNode)(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)VirtualNodeName | <i>node</i> | The VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) to add |
| bool (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType)(https://learn.microsoft.com/dotnet/api/system.boolean)VirtualNodeTypeFilter | <i>overwrite</i> | Indicates whether to overwrite an existing node |

AddSymbolicLink(VirtualPath, VirtualPath?, bool)

Adds a symbolic link.

Declaration

```
public void AddSymbolicLink(VirtualPath linkPath, VirtualPath? targetPath = null, bool overw  
rite = false)
```

(AkiraNetwork.VirtualStorageLibrary)

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|--|-------------------|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary.VirtualPath VirtualPath VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>linkPath</i> | The path of the symbolic link |
| | <i>targetPath</i> | The path of the target |
| bool VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary (https://learn.microsoft.com/dotnet/api/system.boolean 1.html)) VirtualException | <i>overwrite</i> | Indicates whether to overwrite an existing node |

AddSymbolicLink(VirtualPath, VirtualSymbolicLink, bool)

2.html)

Adds a symbolic link.

VirtualID (Ali-Ni)

Declaration

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

```
public void AddSymbolicLink(VirtualPath linkDirectoryPath, VirtualSymbolicLink link, bool ov  
erwrite = false)  
(AkiraNetwork.VirtualStorageLibrary  
1.html)
```

metastem<T>

| Type | Name | Description |
|---|--------------------------|--|
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | | |
| VirtualPathNode | <i>linkDirectoryPath</i> | The path of the directory to add the symbolic link to |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | | |
| VirtualNodeContext | | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) | <i>link</i> | The VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) to add |
| VirtualNodeListConditions | | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) | | |
| VirtualnodeName | | |
| bool(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException) | <i>overwrite</i> | Indicates whether to overwrite an existing node |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | | |
| VirtualNodeType | | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | | |

ChangeDirectory(VirtualPath)

Changes the current working directory.

Declaration

```
public void ChangeDirectory(VirtualPath path)
```

Parameters

(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|---|------|---------------------------------------|
| NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary) | | |
| VirtualPath PatternMatch (AkiraNetwork.VirtualStorageLibrary) VirtualCycleDetector | path | The path of the new working directory |

ConvertToAbsolutePath(VirtualPath?, VirtualPath?)

VirtualDirectoryAdapter<T>

Converts a relative path to an absolute path.
(AkiraNetwork.VirtualStorageLibrary)

Declaration¹)

VirtualException

```
(AkiraNetwork.VirtualStorage.Ilibrar
public VirtualPath ConvertToAbsolutePath(VirtualPath? relativePath, VirtualPath? basePath =
    VirtualGroupCondition<T, TKey>
null)
(AkiraNetwork.VirtualStorage.Ilibrar
```

(AKTA NETWORK.VIRTUAL STORAGE LIBRARY 2.11.10)

2.html)

Parameters

| Type | Description |
|--|---|
| (AkiraNetwork.VirtualStorageLibrary.VirtualItem) | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>relativePath</i> The relative path. |
| VirtualPath 1.html (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) VirtualItem<T> | <i>basePath</i> The base path. If not specified, CurrentPath is used. |

(AkiraNetwork.VirtualStorageLibrary.Returns...)

| Type | Description |
|---|---|
| <code>VirtualNode</code> (<code>AkiraNetwork.VirtualStorageLibrary.VirtualPath</code>) | The absolute path represented by <code>VirtualPath</code> (<code>AkiraNetwork.VirtualStorageLibrary.VirtualPath.html</code>). |
| <code>VirtualNodeContext</code> (<code>AkiraNetwork.VirtualStorageLibrary.VirtualPath.html</code>) | |
| <code>VirtualNodeExtensions</code> | |

(AkiraNetwork.VirtualStorageLibrary
Exceptions) Node listConditions

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

CopyNode(VirtualPath, VirtualPath, bool, bool, bool, List<VirtualNodeContext>?)
VirtualNodeTypeFilter

Declaration

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



Parameters

| Type | Name | Description |
|---|------------------------|-------------|
| NotifyNodeDelegate | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | sourcePath | |
| PatternMatch | destinationPath | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory) | overwrite | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | recursive | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | followLinks | |
| VirtualDirectoryAdapter<T> List (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1) (AkiraNetwork.VirtualStorageLibrary.VirtualException) < VirtualException VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html) (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey>) | destinationContextList | |
| > (AkiraNetwork.VirtualStorageLibrary.VirtualID) 2.html) VirtualID | | |

DirectoryExists(VirtualPath, bool)

Checks if a directory exists at the specified path.

Declaration
VirtualItem
VirtualItemAdapter<T>
(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>)
1.html)
public bool DirectoryExists(VirtualPath path, bool followLinks = false)
(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T>)

Parameters

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

Returns `VirtualNodeListConditions`

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

VirtualNotFoundException
(AkiraNetwork.VirtualStorageLibrary.VirtualNotFoundException)

VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType)

VirtualNodeTypeFilter

ExpandPath(VirtualPath, VirtualNodeTypeFilter, bool, bool)

Expands a path that contains wildcards.

Declaration

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



Parameters

| | | | |
|------|--|--------------|------------------------------------|
| Type | (AkiraNetwork.VirtualStorageLibrary.IVirtualNodeDelegate) | Name | Description |
| | (AkiraNetwork.VirtualStorageLibrary.IVirtualPath) | path | The path. |
| | (AkiraNetwork.VirtualStorageLibrary.IVirtualNodeTypeFilter) | filter | The node type filter. |
| | (AkiraNetwork.VirtualStorageLibrary.IVirtualDirectoryAdapter<T>) | followLinks | Whether to follow symbolic links. |
| | (AkiraNetwork.VirtualStorageLibrary.IVirtualDirectoryAdapter<T>) | resolveLinks | Whether to resolve symbolic links. |

Returns

| | | |
|------|---|--------------------|
| Type | virtualGroupCondition<T, TKey> | Description |
| | (AkiraNetwork.VirtualStorageLibrary.IEnumerable<VirtualPath>) | The expanded paths |

Remarks

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

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

(AkiraNetwork.VirtualStorageLibrary.IVirtualItemAdapter<T>)

VirtualNodeContext

ExpandPathTree(VirtualPath, VirtualNodeTypeFilter, bool, bool)

VirtualNodeExtensions

Expands the virtual path containing wildcards.

(AkiraNetwork.VirtualStorageLibrary.IVirtualPath)

Declaration

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.IVirtualPath)

VirtualNodeName

```
public IEnumerable<VirtualNodeContext> ExpandPathTree(VirtualPath path, VirtualNodeTypeFilter filter = VirtualNodeTypeFilter.All, bool followLinks = true, bool resolveLinks = true)
```

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary.IVirtualPath)

Parameters

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.IVirtualPath)

VirtualNodeTypeFilter

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

Returns AkiraNetwork.VirtualStorageLibrary

| Type | Description |
|--|---|
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary) | An enumerable of VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html) containing information about the expanded nodes |

Remarks

VirtualItem
followLinks indicates whether to follow links during the recursive traversal process when the terminal node in the virtual path is a symbolic link.

VirtualItemAdapter<T>

resolveLinks indicates whether to resolve links when nodes other than the terminal node in the virtual path are symbolic links.

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary
.html)

GetDirectory(VirtualPath, bool)

VirtualNode

Gets the directory at the specified path.
(AkiraNetwork.VirtualStorageLibrary)

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtension GetDirectory(VirtualPath path, bool followLinks = false)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

Parameters
(AkiraNetwork.VirtualStorageLibrary)

| Type | Name | Description |
|--|-------------|-----------------------------------|
| VirtualNodeName (AkiraNetwork.VirtualStorageLibrary) | path | The path of the directory. |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | followLinks | Whether to follow symbolic links. |

VirtualNodeType
(AkiraNetwork.VirtualStorageLibrary)

Returns VirtualNodeTypeFilter
(AkiraNetwork.VirtualStorageLibrary)

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

Exceptions

(AkiraNetwork.VirtualStorageLibrary

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

VirtualCycleDetector

(AkiraNetwork.VirtualStorageLibrary)

GetItemVirtualPath, bool)

(AkiraNetwork.VirtualStorageLibrary)
Gets the item at the specified path.

Declaration (AkiraNetwork.VirtualStorageLibrary 1.html)

VirtualException

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

VirtualGroupCondition<T, TKey>

Parameters: Network.VirtualStorageLibrary

| Type | Name | Description |
|--|-------------|-----------------------------------|
| VirtualItem VirtualID | | |
| Virt (API Path) VirtualPath VirtualStorage | path | The path of the item. |
| VirtualItem bool (Akira Network Virtual Storage Library API) | followLinks | Whether to follow symbolic links. |

VirtualItemAdapter<T>

| Type | Description |
|--|--|
| VirtualItem<T> | The VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) |
| (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) | VirtualNode |
| <T>(AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) | at the specified path. |

VirtualNodeContext

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

GetLinksFromDictionary(VirtualPath)
(A link to the VirtualPath object)

BackingVirtualNodeType

(AkiraNetwork VirtualSt

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

Parameters

| Type | Name | Description |
|---|------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary) | targetPath | |

Returns

| Type | Description |
|--|-------------|
| HashSet<VirtualPath> (AkiraNetwork.VirtualStorageLibrary.dotnet/api/system.collections.generic.hashset-1).<VirtualPath> (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)> | |

GetNode(VirtualPath, bool)

Gets the node at the specified path.

1.html)

Declaration

AkiraNetwork.VirtualStorageLibrary

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

(AkiraNetwork.VirtualStorageLibrary)

2.html)

Parameters

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

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

(AkiraNetwork.VirtualStorageLibrary)

1.html)

Returns

VirtualItem<T>

| Type | Description |
|------------------------------------|-------------|
| AkiraNetwork.VirtualStorageLibrary | |

1.html)

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.VirtualN

ode.html)

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException.html)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

GetNodeType(VirtualPath, bool)

(AkiraNetwork.VirtualStorageLibrary)

Gets the node type at the specified path.

VIRTUALNODETYPE

| Type | Condition |
|-----------------|-------------------------------------|
| VIRTUALNODETYPE | Thrown if the node cannot be found. |

Thrown if the node cannot be found.

Declaration

```
public VirtualNodeType GetNodeType(VirtualPath path, bool followLinks = false)
```

Parameters

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

PatternMatch

Returns: (AkiraNetwork.VirtualStorageLibrary.VirtualNode)

| Type | Description |
|---|---|
| VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualNode) | |
| VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNode) | The VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html) of the node. |
| VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualNode) | |
| Type.html VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualNode) | |
| 1.html | |
| VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualNode) | |
| VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualNode) | |

GetNodes(VirtualNodeTypeFilter, bool, bool)

VirtualGroupCondition<T, TKey>

Gets (AkiraNetwork.VirtualStorageLibrary.VirtualNode) from the current path.

2.html)

Declaration:

VirtualID

(AkiraNetwork.VirtualStorageLibrary.VirtualNode)

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

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualNode)

Parameters:

1.html)

Type:

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualNode)

1.html)

VirtualNode

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

VirtualNodeContext

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

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.VirtualNode)

VirtualNodeListConditions

Returns:

(AkiraNetwork.VirtualStorageLibrary.VirtualNode)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualNode)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary.VirtualNode)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.VirtualNode)

VirtualNodeTypeFilter

| Type | Description |
|--|---|
| IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) < VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)) (AkiraNetwork.VirtualStorageLibrary. > NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary. PatternMatch Remarks (AkiraNetwork.VirtualStorageLibrary. This method may be integrated, reorganized, or deprecated in the near future. VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary. VirtualDirectory Gets VirtualDirectoryAdapter starting from the specified path. (AkiraNetwork.VirtualStorageLibrary. Declaration 1.html VirtualException public IEnumerable<VirtualNode> GetNodes(VirtualPath basePath , VirtualNodeTypeFilter nodeType , VirtualNodeCondition<T, TKey> , bool recursive = false , bool followLinks = false) (AkiraNetwork.VirtualStorageLibrary. ? html) Parameters VirtualID Type (AkiraNetwork.VirtualStorageLibrary. VirtualItem VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary. VirtualNodeTypeFilter VirtualNodeAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) 1.html) bool (https://learn.microsoft.com/dotnet/api/system.boolean) VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary. bool (https://learn.microsoft.com/dotnet/api/system.boolean) VirtualNode (AkiraNetwork.VirtualStorageLibrary. VirtualNodeContext Returns VirtualNodeContext Type (AkiraNetwork.VirtualStorageLibrary. VirtualNodeExtensions IEnumerable (AkiraNetwork.VirtualStorageLibrary. (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) < VirtualNodeName VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)) (AkiraNetwork.VirtualStorageLibrary. > VirtualNodeType (AkiraNetwork.VirtualStorageLibrary. VirtualNodeTypeFilter | The nodes in the path tree starting from the current path IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) . . |

GetNodes(VirtualPath, VirtualNodeTypeFilter, bool, bool)

(AkiraNetwork.VirtualStorageLibrary.

Gets VirtualDirectoryAdapter starting from the specified path.

(AkiraNetwork.VirtualStorageLibrary.

Declaration

```
1.html
VirtualException
public IEnumerable<VirtualNode> GetNodes(VirtualPath basePath, VirtualNodeTypeFilter nodeType,
VirtualNodeCondition<T, TKey>, bool recursive = false, bool followLinks = false)
(AkiraNetwork.VirtualStorageLibrary.  
? html)
```

Parameters

| Type | Name | Description |
|---|-------------|--|
| VirtualID Type (AkiraNetwork.VirtualStorageLibrary. VirtualItem VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary. VirtualNodeTypeFilter VirtualNodeAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) 1.html) bool (https://learn.microsoft.com/dotnet/api/system.boolean) VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary. bool (https://learn.microsoft.com/dotnet/api/system.boolean) VirtualNode (AkiraNetwork.VirtualStorageLibrary. VirtualNodeContext Returns VirtualNodeContext Type (AkiraNetwork.VirtualStorageLibrary. VirtualNodeExtensions IEnumerable (AkiraNetwork.VirtualStorageLibrary. (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) < VirtualNodeName VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)) (AkiraNetwork.VirtualStorageLibrary. > VirtualNodeType (AkiraNetwork.VirtualStorageLibrary. VirtualNodeTypeFilter | basePath | The base path. |
| nodeType VirtualNodeAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) 1.html) bool (https://learn.microsoft.com/dotnet/api/system.boolean) VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary. bool (https://learn.microsoft.com/dotnet/api/system.boolean) VirtualNode (AkiraNetwork.VirtualStorageLibrary. VirtualNodeContext Returns VirtualNodeContext Type (AkiraNetwork.VirtualStorageLibrary. VirtualNodeExtensions IEnumerable (AkiraNetwork.VirtualStorageLibrary. (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) < VirtualNodeName VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)) (AkiraNetwork.VirtualStorageLibrary. > VirtualNodeType (AkiraNetwork.VirtualStorageLibrary. VirtualNodeTypeFilter | nodeType | The node type filter. |
| recursive VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary. bool (https://learn.microsoft.com/dotnet/api/system.boolean) VirtualNode (AkiraNetwork.VirtualStorageLibrary. VirtualNodeContext Returns VirtualNodeContext Type (AkiraNetwork.VirtualStorageLibrary. VirtualNodeExtensions IEnumerable (AkiraNetwork.VirtualStorageLibrary. (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) < VirtualNodeName VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)) (AkiraNetwork.VirtualStorageLibrary. > VirtualNodeType (AkiraNetwork.VirtualStorageLibrary. VirtualNodeTypeFilter | recursive | Whether to retrieve nodes recursively. |
| followLinks VirtualNode (AkiraNetwork.VirtualStorageLibrary. VirtualNodeContext Returns VirtualNodeContext Type (AkiraNetwork.VirtualStorageLibrary. VirtualNodeExtensions IEnumerable (AkiraNetwork.VirtualStorageLibrary. (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) < VirtualNodeName VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)) (AkiraNetwork.VirtualStorageLibrary. > VirtualNodeType (AkiraNetwork.VirtualStorageLibrary. VirtualNodeTypeFilter | followLinks | Whether to follow symbolic links. |

VirtualNodeContext

| Type | Description |
|---|---|
| VirtualNodeContext Type (AkiraNetwork.VirtualStorageLibrary. VirtualNodeExtensions IEnumerable (AkiraNetwork.VirtualStorageLibrary. (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) < VirtualNodeName VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)) (AkiraNetwork.VirtualStorageLibrary. > VirtualNodeType (AkiraNetwork.VirtualStorageLibrary. VirtualNodeTypeFilter | 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

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

GetNodesWithPaths(VirtualNodeTypeFilter, bool, bool)

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

Declaration

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

Parameters

| Type | Name | Description |
|---|-------------|--|
| VirtualDirectory | | |
| VirtualPath | nodeType | The node type filter. |
| (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html) | | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | recursive | Whether to retrieve paths recursively. |
| VirtualException | | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | followLinks | Whether to follow symbolic links. |
| VirtualGroupCondition<T, TKey> | | |
| (AkiraNetwork.VirtualStorageLibrary | | |
| 2.html) | | |

Returns

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

Remarks

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

GetNodesWithPaths(VirtualPath, VirtualNodeTypeFilter, bool, bool)

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

(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
VirtualNodeListConditions
VirtualNodeExtensions
VirtualNodeException
VirtualNodeType
VirtualNodeName
VirtualNodeTypeFilter
VirtualNodeNotFoundException
GetNodesWithPaths(VirtualPath basePath, VirtualNodeTypeFilte
r(VirtualNodeTypeFilter.All, bool recursive = false, bool followLinks = false)
VirtualNodeType
VirtualNodeTypeFilter
VirtualNodeTypeFilter
```

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

Befu(AkiraNetwork.VirtualStorageLibrary)

| Type | Description |
|--|---|
| <pre>IEnumerable<VirtualDirectory> (AkiraNetwork.VirtualStorageLibrary/api/system.collection.generic.ienumerable-1) < (AkiraNetwork.VirtualStorageLibrary.VirtualPath) VirtualPath) (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) > (AkiraNetwork.VirtualStorageLibrary)</pre> | The paths in the path tree starting from the specified path IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collection.generic.ienumerable-1) |

Remarks (AkiraNetwork.VirtualStorageLibrary)

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

GetVirtualLink(VirtualPath)

(AkiraNetwork.VirtualStorageLibrary)

Gets the symbolic link at the specified path.

(AkiraNetwork.VirtualStorageLibrary
Declaration

```
1.html)
VirtualItem<T>
public VirtualSymbolicLink GetSymbolicLink(VirtualPath path)
(AkiraNetwork.VirtualStorageLibrary
1.html)
```

VirtualNode Parameters

| Type | Name | Description |
|---|-------------|--------------------------------|
| AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext | <i>path</i> | The path of the symbolic link. |

VirtualNodeExtensions

Returns `VirtualStorageLibrary`

| Type | Description |
|---|--|
| VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary) | The VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) |

VirtualNodeType

Remarks (AkiraNetwork.VirtualS
VirtualNodeTypeFilter

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

Exceptions

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

| | |
|--|---|
| ItemExists(VirtualPath, bool) (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | CheckIfItemExists at the specified path. (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) |
| Declaration VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector.html) | VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter.html) |
| Parameters VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path VirtualException (AkiraNetwork.VirtualStorageLibrary.VirtualException.html) |
| | followLinks bool (https://learn.microsoft.com/dotnet/api/system.boolean) (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) |
| Returns VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualID.html) | Description True if the item exists; otherwise, false. VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter.html) |

MoveNode(VirtualPath, VirtualPath, bool, bool)

| | |
|--|---|
| VirtualItem<T> Moves the node at the specified path to a new location. (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) | sourcePath, destinationPath, overwrite = false VirtualNodeContextLinks = true (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContextLinks.html) |
| Declaration VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html) |

| Type | Name | Description |
|--|-----------------|--------------------------------------|
| VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions.html) | | |
| VirtualPath VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | sourcePath | The source path of the node. |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | destinationPath | The destination path for the node. |
| VirtualNodeType bool (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html) | overwrite | Whether to overwrite existing nodes. |

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

Remarks

`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

| Type | Condition |
|---|---|
| (AkiraNetwork.VirtualStorageLibrary) PatternMatch InvalidOperationException (https://learn.microsoft.com/dotnet/api/system.invalidoperationexception) (AkiraNetwork.VirtualStorageLibrary) | 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. |
| VirtualDirectory VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary) (AkiraNetwork.VirtualStorageLibrary) VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary) odeNotFoundException.html (AkiraNetwork.VirtualStorageLibrary) 1.html VirtualException | Thrown if the source node is not found. |

NodeExists(VirtualPath, bool)

[VirtualGroupCondition<T, TKey>](#)
Checks if a node exists at the specified path.
[\(AkiraNetwork.VirtualStorageLibrary\)](#)

Declaration

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

Parameters

| Type | Name | Description |
|---|-------------|------------------|
| (AkiraNetwork.VirtualStorageLibrary) 1.html | | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary) VirtualPath.html | <i>path</i> | The node's path. |

Parameters

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

Return Value

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

RemoveLinkByLinkPath(VirtualPath)

[VirtualnodeName](#)

Declaration

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

Parameters

[VirtualNodeTypeFilter](#)

| Type | Name | Description |
|---|----------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | linkPath | |

RemoveLinkFromDictionary(VirtualPath, VirtualPath)

Declaration

```
(AkiraNetwork.VirtualStorageLibrary)
    NotifyNodeDelegate
public void RemoveLinkFromDictionary(VirtualPath targetPath, VirtualPath linkPath)
(AkiraNetwork.VirtualStorageLibrary)
    PatternMatch
```

Parameters

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

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

targetPath

(AkiraNetwork.VirtualStorageLibrary)

linkPath

(AkiraNetwork.VirtualStorageLibrary)

1.html)

VirtualException

RemoveNode(VirtualPath, bool, bool, bool)

(AkiraNetwork.VirtualStorageLibrary)

Remove the node and its children at the specified path.

(AkiraNetwork.VirtualStorageLibrary)

Declaration

2.html)

VirtualID

```
public void RemoveNode(VirtualPath nodePath, bool recursive = false, bool followLinks = false,
    bool resolveLinks = true)
VirtualItem
```

(AkiraNetwork.VirtualStorageLibrary)

VirtualItemAdapter<T>

| Type | Name | Description |
|--|--------------|-----------------------------------|
| (AkiraNetwork.VirtualStorageLibrary) | | |
| T.html) | | |
| VirtualItem<T> | nodePath | The path of the node to remove. |
| (AkiraNetwork.VirtualStorageLibrary) | | |
| 1.html) | | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | recursive | Whether to remove recursively. |
| VirtualNode | | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | followLinks | Whether to follow symbolic links. |
| VirtualNodeContext | | |
| (AkiraNetwork.VirtualStorageLibrary) | | |
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | resolveLinks | Whether to resolve link targets. |
| VirtualNodeExtensions | | |
| (AkiraNetwork.VirtualStorageLibrary) | | |

Remarks

VirtualNodeListConditions

If followLinks is true, and the node in the path is a symbolic link, the path tree after resolving the path will be removed. If true, 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)

VirtualNodeTypeFilter

| Type | Condition |
|--|--|
| InvalidOperationException (https://learn.microsoft.com/dotnet/api/system.invalidoperationexception) | Thrown when trying to remove the root node or a non-empty directory. |

▼

ResolveLinkTarget(VirtualPath)

NotifyNodeDelegate
Resolves the target of the symbolic link at the specified path.
(AkiraNetwork.VirtualStorageLibrary)

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualCycleDetector
public VirtualPath ResolveLinkTarget(VirtualPath path)
(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectory

Parameters
(AkiraNetwork.VirtualStorageLibrary)

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

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

path

The path of the symbolic link.

VirtualException

Return (AkiraNetwork.VirtualStorageLibrary)

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

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

The target path.

VirtualID

(AkiraNetwork.VirtualStorageLibrary)

| Type | Condition |
|---|-----------|
| VirtualItem (AkiraNetwork.VirtualStorageLibrary) | |

VirtualItemAdapter<T>

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

1.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary)

SetLinkTargetNodeType(HashSet<VirtualPath>, VirtualNodeType)

Declaration

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeContext

public void SetLinkTargetNodeType(HashSet<VirtualPath> linkPathSet, VirtualNodeType nodeTyp
e)
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

Parameters

NodeListConditions

| Type | Name | Description |
|---|-------------|-------------|
| VirtualNodeName | | |
| HashSet<VirtualPath> (https://learn.microsoft.com/dotnet/api/system.collections.generic.hashset-1) | linkPathSet | |
| 1) VirtualNodeNotFoundException <VirtualPath>(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)> | | |
| VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html) | nodeType | |
| VirtualNodeTypeFilter | | |

SetNode(VirtualPath, VirtualNode)

Sets a node at the specified path.

Declaration

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

Parameters

NotifyNodeDelegate

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

VirtualException

(AkiraNetwork.VirtualStorageLibrary.VirtualException.html)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition.html)

SetNodeName(VirtualPath, VirtualNodeName, bool)

Changes the name of a node.

VirtualID

Declaration

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html)

VirtualItem

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

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter.html)

1.html)

Parameters

VirtualItem<T>

| Type | Name | Description |
|---|--------------|--|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) | nodePath | The path of the node to rename |
| VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext.html) | newName | The new name |
| VirtualNodeExtensions bool (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html) VirtualNodeListConditions (https://learn.microsoft.com/dotnet/api/system.boolean) | resolveLinks | Indicates whether to resolve symbolic links |

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html)

VirtualNodeName

SymbolicLinkExists(VirtualPath)

VirtualNodeNotFoundException

Checks if a symbolic link exists at the specified path.

(AkiraNetwork.VirtualStorageLibrary.VirtualStorageLibrary.html)

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html)

VirtualNodeTypeFilter

```
public bool SymbolicLinkExists(VirtualPath path)
```

Parameters

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

Returns

| Type | Description |
|---|---|
| bool (AkiraNetwork.VirtualStorageLibrary.html/api/system.boolean) VirtualCycleDetector (AkiraNetwork.VirtualStorageLibrary) | True if the symbolic link exists; otherwise, false. |

TryGetDirectory(VirtualPath, bool)

Tries to get the directory at the specified path.
(AkiraNetwork.VirtualStorageLibrary)

Declaration

```
VirtualException  
(AkiraNetwork.VirtualStorageLibrary)  
public VirtualDirectory TryGetDirectory(VirtualPath path, bool followLinks = false)  
VirtualGroupCondition<T, TKey>  
(AkiraNetwork.VirtualStorageLibrary)
```

Parameters

| Type | Name | Description |
|--|-------------|-----------------------------------|
| VirtualID (AkiraNetwork.VirtualStorageLibrary) | path | The path of the directory. |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) VirtualItem (AkiraNetwork.VirtualStorageLibrary) | followLinks | Whether to follow symbolic links. |

Returns

VirtualDirectory
(AkiraNetwork.VirtualStorageLibrary)

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

TryGetItem(VirtualPath, bool)

VirtualNodeListConditions
Tries to get the item at the specified path.
(AkiraNetwork.VirtualStorageLibrary)

Declaration

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

Parameters

VirtualNodeFilter
(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

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

Returns

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

TryGetNode(VirtualPath, bool)

Tries to get the node at the specified path.
(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

Declaration

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

Parameters

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

Returns
VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

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

TryGetSymbolicLink(VirtualPath)

Tries to get the symbolic link at the specified path.
(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

Declaration

```
public VirtualSymbolicLink? TryGetSymbolicLink(VirtualPath path)
```

Parameters

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

VirtualNodeNotFoundException

VirtualNodeType

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

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

Returns

| Type | Description |
|--|---|
| VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) NotifyNodeDelegate (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) PatternMatch (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.
(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)

VirtualDirectory

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

TryResolveLinkTarget(VirtualPath)

Tries to resolve the target of the symbolic link at the specified path.
(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

Declaration

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)
pVirtualSymbolicLink TryResolveLinkTarget(VirtualPath path)

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

Parameters

VirtualID

| Type | Name | Description |
|---|------|--------------------------------|
| AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink VirtualItem VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | path | The path of the symbolic link. |

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter.html)

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

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

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.VirtualNode.html)

VirtualNodeContent

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContent.html)

UpdateAllTargetNodeTypesInDictionary()

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeContent.html)

VirtualNodeExtensions

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html)

pVirtualNodeList GetTargetNodeTypesInDictionary()

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeList.html)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualnodeName.html)

VirtualNodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)

UpdateDirectory(VirtualPath, VirtualDirectory)

(AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html)

Updates a directory.

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeType.html)

VirtualNodeTypeFilter

(AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter.html)

Declaration

VirtualNodeTypeFilter

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

Parameters

| Type | Name | Description |
|--|---------------|-------------------------------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | directoryPath | The path of the directory to update |
| NotifyNodeDelegate VirtualDirectory (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) PatternMatch (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory.html) | newDirectory | The new data for the directory |

UpdateItem(VirtualPath, VirtualItem<T>)

VirtualDirectory

Updates an item

(AkiraNetwork.VirtualStorageLibrary)

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary)

1.html)

```
public void UpdateItem(VirtualPath itemPath, VirtualItem<T> newItem)
```

VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary)

2.html)

| Type | Name | Description |
|--|----------|--------------------------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | itemPath | The path of the item to update |
| VirtualItem (AkiraNetwork.VirtualStorageLibrary.VirtualItem-1.html) | newItem | The new data for the item |

UpdateLinkInDictionary(VirtualPath, VirtualPath)

(AkiraNetwork.VirtualStorageLibrary)

Declaration

VirtualNode

```
public void UpdateLinkInDictionary(VirtualPath linkPath, VirtualPath newTargetPath)
```

VirtualNodeContext

(AkiraNetwork.VirtualStorageLibrary)

Parameters

VirtualNodeExtensions

| Type | Name | Description |
|---|---------------|-------------|
| VirtualNodeListConditions VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | linkPath | |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | newTargetPath | |

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary)

UpdateLinksToTarget(VirtualPath, VirtualPath)

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

```
public void UpdateLinksToTarget(VirtualPath oldTargetPath, VirtualPath newTargetPath)
```

Parameters

| Type | Name | Description |
|---|---------------|-------------|
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | oldTargetPath | |
| (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | newTargetPath | |

(AkiraNetwork.VirtualStorageLibrary.

PatternMatch

UpdateSymbolicLink(VirtualPath, VirtualSymbolicLink)

VirtualCycleDetector

Updates a symbolic link

(AkiraNetwork.VirtualStorageLibrary.

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary.

VirtualDirectoryAdapter<T>

```
public void UpdateSymbolicLink(VirtualPath linkPath, VirtualSymbolicLink newLink)
```

(AkiraNetwork.VirtualStorageLibrary.

1.html)

VirtualException

| Type | Name | Description |
|--|----------|---|
| VirtualGroupCondition<T, TKey> | | |
| Virt (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | linkPath | The path of the symbolic link to update |

2.html)

VirtualID

VirtualSymbolicLink

(AkiraNetwork.VirtualStorageLibrary.

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.

1.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary.

```
public void UpdateTargetNodeTypesInDictionary(VirtualPath targetPath)
```

VirtualNode

(AkiraNetwork.VirtualStorageLibrary.

Parameters

VirtualNodeContext

| Type | Name | Description |
|-------------------------------------|------------|-------------|
| AkiraNetwork.VirtualStorageLibrary. | | |
| VirtualNodeExtensions | targetPath | |

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeNotFoundException

Traverses the path to the target node specified by the virtual path.

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeTypeFilter

| Type | Name | Description |
|-------------------------------------|------------|-------------|
| AkiraNetwork.VirtualStorageLibrary. | | |
| VirtualNodeExtensions | targetPath | |

VirtualNodeNotFoundException

Traverses the path to the target node specified by the virtual path.

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeType

(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

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)
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 links.

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualnodeName

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualStorage<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

DebugText<T>(VirtualStorage<T>)

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

(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils_VirtualStorageDebugText_1_AkiraNetwork_VirtualStorageLibrary_VirtualStorage__0_)

VirtualStorage<T>

(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils_VirtualStorageDebugText_2_AkiraNetwork_VirtualStorageLibrary_VirtualStorage__1_)

GenerateSingleTableDebugText<T>(T)

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

VirtualNodeTypeFilter

Extension Methods

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

VirtualStorage<T>

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

DebugText<T>(VirtualStorage<T>)

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

(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils_VirtualStorageDebugText_1_AkiraNetwork_VirtualStorageLibrary_VirtualStorage__0_)

VirtualStorage<T>

(AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils_VirtualStorageDebugText_2_AkiraNetwork_VirtualStorageLibrary_VirtualStorage__1_)

GenerateSingleTableDebugText<T>(T)

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

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

Declaration

```
VirtualCycleDetector  
(AkiraNetwork.VirtualStorageLibrary  
public VirtualSymbolicLink()  
VirtualDirectory  
(AkiraNetwork.VirtualStorageLibrary  
VirtualDirectoryAdapter<T>  
(AkiraNetwork.VirtualStorageLibrary
```

VirtualSymbolicLink(VirtualNodeName)

1.html)

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
```

Parameters

(AkiraNetwork.VirtualStorageLibrary

| Type | Name | Description |
|--|------|--------------------------------|
| VirtualItemAdapter<T> (AkiraNetwork.VirtualStorageLibrary 1.html) (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary 1.html) | name | The name of the symbolic link. |

VirtualSymbolicLink(VirtualNodeName, VirtualPath?)

(AkiraNetwork.VirtualStorageLibrary
Initializes a new instance of the VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) class with the specified name and target path.
VirtualNodeContext
(AkiraNetwork.VirtualStorageLibrary

VirtualNodeExtensions

```
(AkiraNetwork.VirtualStorageLibrary  
VirtualNodeListConditions  
public VirtualSymbolicLink(VirtualNodeName name, VirtualPath? targetPath)  
(AkiraNetwork.VirtualStorageLibrary  
VirtualNodeName  
VirtualNodeName  
(AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html)  
VirtualNodeTypeFilter
```

Parameters

(AkiraNetwork.VirtualStorageLibrary

| Type | Name | Description |
|--|------|--------------------------------|
| VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName.html) VirtualNodeType | 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) | <i>name</i> | The name of the symbolic link. |
| VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html) | <i>targetPath</i> | The target path of the symbolic link. |
| DateTime (https://learn.microsoft.com/dotnet/api/system.datetime) | <i>createdDate</i> | The creation date. |
| DateTime (https://learn.microsoft.com/dotnet/api/system.datetime) | <i>updatedDate</i> | The update date. |

Properties

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

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<VirtualNodeType> NodeType { get; }
  : base(NodeType)
  {
    VirtualNodeName
    (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)
  }
```

Property Value

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

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

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink)

Overrides

VirtualNodeTypeFilter

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

TargetNodeType

Gets or sets the type of the target node of the symbolic link.

(AkiraNetwork.VirtualStorageLibrary.

Declaration

NotifyNodeDelegate

(AkiraNetwork.VirtualStorageLibrary.

public VirtualNodeType TargetNodeType { get; set; }

(AkiraNetwork.VirtualStorageLibrary.

Property Value

(AkiraNetwork.VirtualStorageLibrary.

Type

VirtualDirectory

(AkiraNetwork.VirtualStorageLibrary.

VirtualDirectoryAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.

1.html)

TargetException

(AkiraNetwork.VirtualStorageLibrary.

Gets or sets the target path of the symbolic link.

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary.

2.html)

VirtualPath

public VirtualPath? TargetPath { get; set; }

(AkiraNetwork.VirtualStorageLibrary.

VirtualItem

(AkiraNetwork.VirtualStorageLibrary.

Type

VirtualItemAdapter<T>

(AkiraNetwork.VirtualStorageLibrary.

VirtualPath (AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

1.html)

VirtualItem<T>

(AkiraNetwork.VirtualStorageLibrary.

1.html)

Methods

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeContext

DeepClone(bool)

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeExtensions

Creates a deep clone of the symbolic link.

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeListConditions

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeName

public override VirtualNode DeepClone(bool recursive = false)

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeNotFoundException

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeType

(AkiraNetwork.VirtualStorageLibrary.

VirtualNodeTypeFilter

| Type | Name | Description |
|--|-----------|--|
| bool (https://learn.microsoft.com/dotnet/api/system.boolean) | recursive | If true, all links are recursively cloned. |

Returns

| Type | Description |
|---|------------------------------------|
| VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNode.html) (AkiraNetwork.VirtualStorageLibrary.VirtualNode) | A deep clone of the symbolic link. |

Overrides

[NotifyNodeDelegate](#)
[VirtualNode.DeepClone\(bool\)](#)
[PatternMatch](#)
[VirtualNode.DeepClone\(bool\)](#)
[VirtualCycleDetector](#)

[VirtualException](#)
[VirtualDirectory](#)

[VirtualDirectoryAdapter<T>](#)
[VirtualDirectoryAdapter<T>](#)

Returns a string representation of the symbolic link.
[\(AkiraNetwork.VirtualStorageLibrary.VirtualItem\)](#)

Declaration

[VirtualException](#)
[VirtualGroupCondition<T, TKey>](#)
[VirtualItem](#)

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

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>)
[1.html](#)

Update(VirtualNode)

[VirtualNode](#)
[VirtualNode](#)

Updates the current symbolic link with the data from the specified node.

[VirtualNode](#)

[VirtualNodeContext](#)
[VirtualStorageLibrary](#)

[VirtualNode](#) ([VirtualNode](#))
[VirtualNodeExtensions](#)

[VirtualStorageLibrary](#)
[VirtualNodeListConditions](#)

| Type | Name | Description |
|--|------|---------------------------------|
| VirtualNode (VirtualNode) | node | The node to use for the update. |

Overrides

[VirtualStorageLibrary](#)
[VirtualNode.Update\(VirtualNode\)](#)

[VirtualStorageLibrary](#)
[VirtualStorageLibrary](#)
[VirtualStorageLibrary](#)
[VirtualStorageLibrary](#)

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](#))

2.html)

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](#))

VirtualPath

([AkiraNetwork.VirtualStorageLibrary](#))

VirtualNode

([AkiraNetwork.VirtualStorageLibrary](#))

VirtualNodeCondition

([AkiraNetwork.VirtualStorageLibrary](#))

VirtualNodeExtensions

([AkiraNetwork.VirtualStorageLibrary](#))

```
public static implicit operator VirtualSymbolicLink((VirtualNodeName nodeName, VirtualPath?
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.VirtualNodeDelegate,akiranetwork.virtualstoragelibrary.virtualpath).targetpath)) (AkiraNetwork.VirtualStorageLibrary.PatternMatch (AkiraNetwork.VirtualStorageLibrary.VirtualCycleDetector Returns VirtualCycleDetector Type (AkiraNetwork.VirtualStorageLibrary.VirtualDirectory VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualException IVirtualDeepCloneable<T> (AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html) (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualItem VirtualNodeExtensions.ResolveNodeType(VirtualNode) VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeExtensions_ResolveNodeType_AkiraNetwork_VirtualStorageLibrary_VirtualNode_) VirtualItem VirtualTextFormatter.GenerateSingleTableDebugText<T>(T) (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils_S_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_) (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T> 1.html) VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter | tuple | The tuple to convert. |

| Type | Description |
|---|-------------|
| (VirtualDirectory VirtualSymbolicLink (AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html) (AkiraNetwork.VirtualStorageLibrary.VirtualDirectoryAdapter<T> (AkiraNetwork.VirtualStorageLibrary.VirtualException IVirtualDeepCloneable<T> (AkiraNetwork.VirtualStorageLibrary.IVirtualDeepCloneable-1.html) (AkiraNetwork.VirtualStorageLibrary.VirtualGroupCondition<T, TKey> (AkiraNetwork.VirtualStorageLibrary.VirtualItem VirtualNodeExtensions.ResolveNodeType(VirtualNode) VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeExtensions_ResolveNodeType_AkiraNetwork_VirtualStorageLibrary_VirtualNode_) VirtualItem VirtualTextFormatter.GenerateSingleTableDebugText<T>(T) (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils_S_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_) (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T> 1.html) VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter | |

Implements

Extension Methods

| | |
|---|--|
| VirtualNodeExtensions.ResolveNodeType(VirtualNode) | |
| VirtualID (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions.html#AkiraNetwork_VirtualStorageLibrary_VirtualNodeExtensions_ResolveNodeType_AkiraNetwork_VirtualStorageLibrary_VirtualNode_) | |
| VirtualItem VirtualTextFormatter.GenerateSingleTableDebugText<T>(T) (AkiraNetwork.VirtualStorageLibrary.Utilities.VirtualTextFormatter.html#AkiraNetwork_VirtualStorageLibrary_Utils_S_VirtualTextFormatter_GenerateSingleTableDebugText_1__0_) (AkiraNetwork.VirtualStorageLibrary.VirtualItemAdapter<T> 1.html) | |
| VirtualItem<T> (AkiraNetwork.VirtualStorageLibrary.VirtualItem.html) | |
| VirtualNode (AkiraNetwork.VirtualStorageLibrary.VirtualNodeContext VirtualNodeContext (AkiraNetwork.VirtualStorageLibrary.VirtualNodeExtensions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeListConditions (AkiraNetwork.VirtualStorageLibrary.VirtualNodeName VirtualNodeName (AkiraNetwork.VirtualStorageLibrary.VirtualNodeNotFoundException (AkiraNetwork.VirtualStorageLibrary.VirtualNodeType (AkiraNetwork.VirtualStorageLibrary.VirtualNodeTypeFilter | |

| 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

1.html)

Type VirtualException

(AkiraNetwork.VirtualStorageLibrary)

VirtualPath

VirtualGroupCondition<T, TKey>

(AkiraNetwork.VirtualStorageLibrary.VirtualPath.html)

(AkiraNetwork.VirtualStorageLibrary)

bool

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

(AkiraNetwork.VirtualStorageLibrary)

VirtualItem

(AkiraNetwork.VirtualStorageLibrary)

Type

VirtualItemAdapter<T>

Name

Description

path

The path of the symbolic link.

followLinks

A flag indicating whether to follow links.

Property Value

(AkiraNetwork.VirtualStorageLibrary)

Description

VirtualSymbolicLink

The

1.html)

VirtualSymbolicLink

(AkiraNetwork.VirtualStorageLibrary.Virtua

lSymbolicLink.html)

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)

corresponding to the specified path.

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)

Exceptions

NodeExtensions

(AkiraNetwork.VirtualStorageLibrary)

Condition

VirtualNodeListConditions

Thrown when the node is not a

Inval

VirtualSymbolicLink

(AkiraNetwork.VirtualStorageLibrary.VirtualSymbolicLink.html)

.

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeNotFoundException

.

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeType

.

(AkiraNetwork.VirtualStorageLibrary)

VirtualNodeTypeFilter

.

Extension Methods

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

(AkiraNetwork.VirtualStorageLibrary)

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

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

object.MemberwiseClone() (<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)
VirtualTextFormatter
<T>
(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

 (AkiraNetwork.VirtualStorageLibrary)

- **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)

Class DefaultWildcardMatcher

Implements default wildcard matching using regular expression patterns. Provides a straightforward mapping of regular expression symbols.

Inherited from

AkiraNetwork.VirtualStorage

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

(AkiraNetwork.VirtualStorage)

Implements

AkiraNetwork.VirtualStorage

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

Library: Utilities

Inherited from

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)))

(AkiraNetwork.VirtualStorage)

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

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

(AkiraNetwork.VirtualStorageLibrary)

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

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

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

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

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.

Inherited from [AkiraNetwork.VirtualStorage](#)

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

(AkiraNetwork.VirtualStorage)

Implements [AkiraNetwork.VirtualStorage](#)

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

Library.Utilities

Inherited from [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

object.GetHashCode() (<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)