

Neato Tags “Quickstart” Guide

Unity 2022.2+

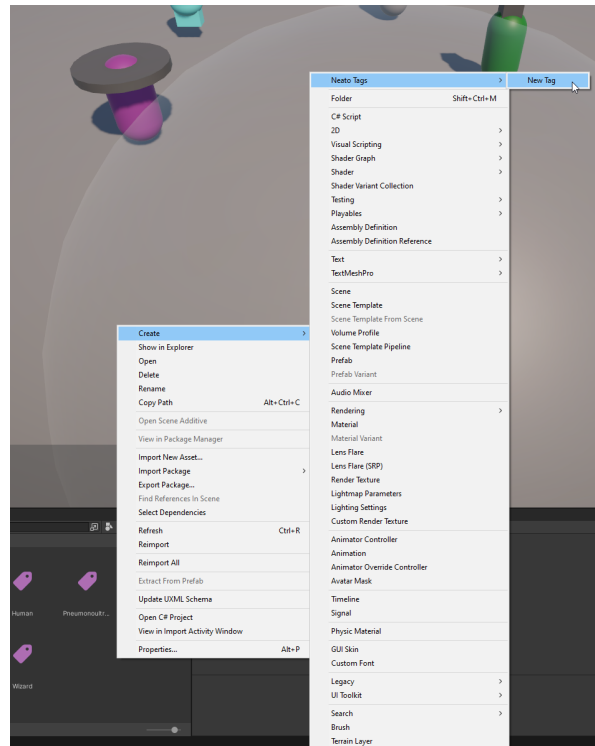
Charlie Lindler

<https://charliemadeathing.com/>

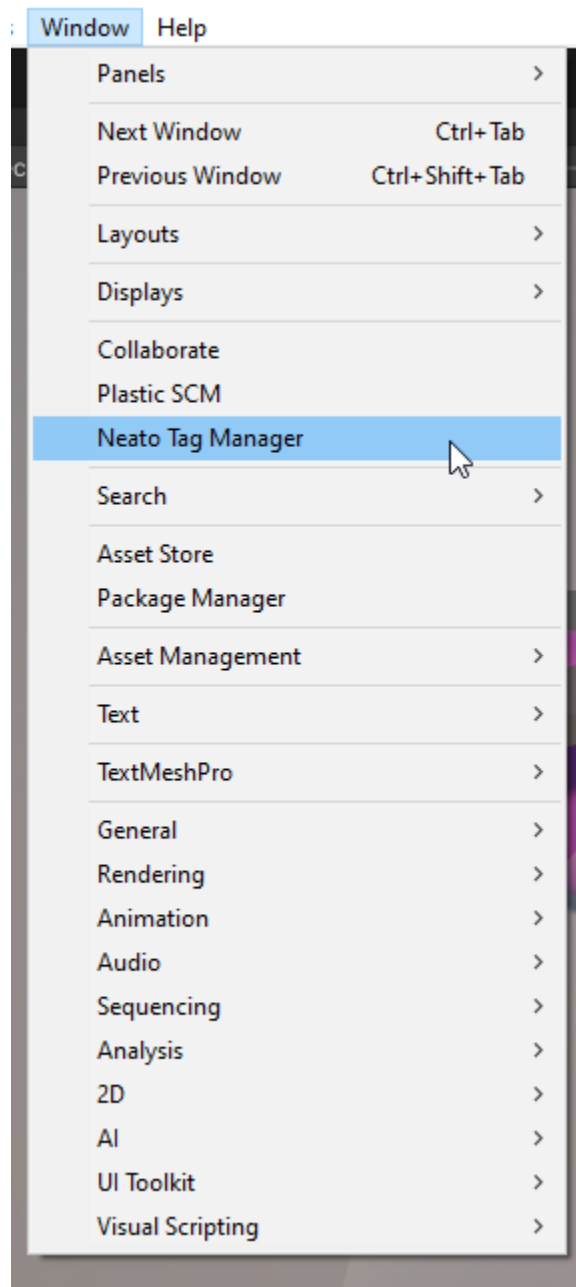
Creating Tags

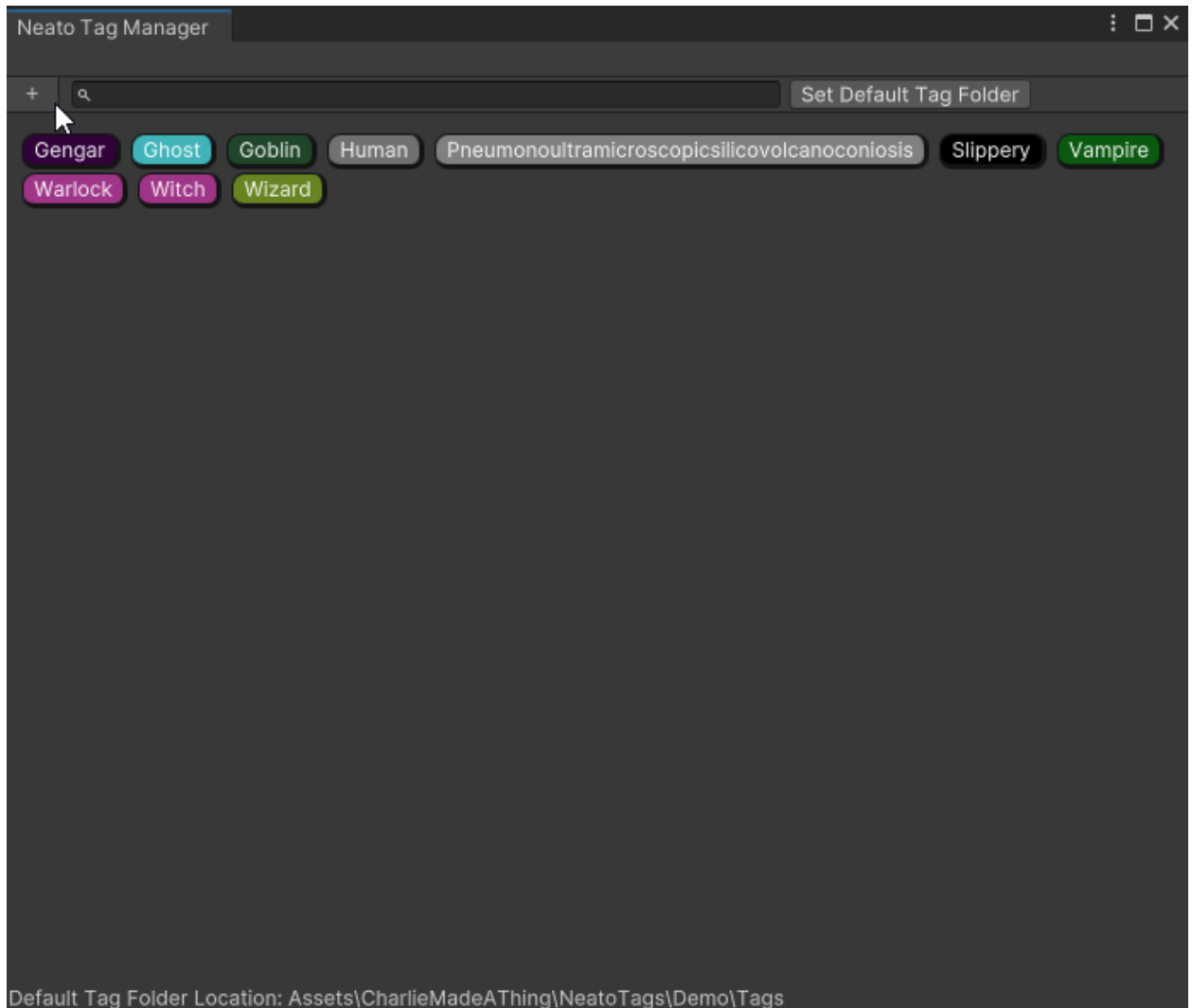
There are multiple ways to creating tags.

1. Right-Click in your Assets folder then click Neato Tags then New Tag.

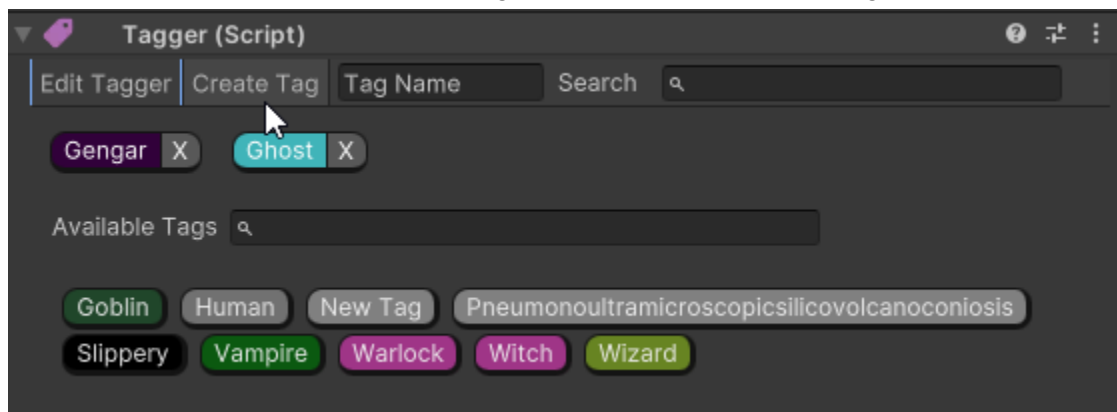


2. In the Tag Manager. Click Window then Neato Tag Manager. Then press the + button.





3. The Tagger Component. When attached to a gameobject press the Edit Tagger button and then Create Tag. *This will get added immediately to the GameObject's tags.* Use the TextField next to it to give it a name before creating it.



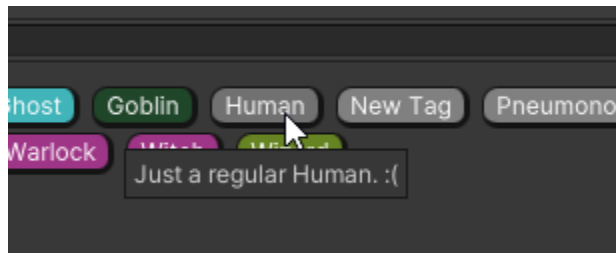
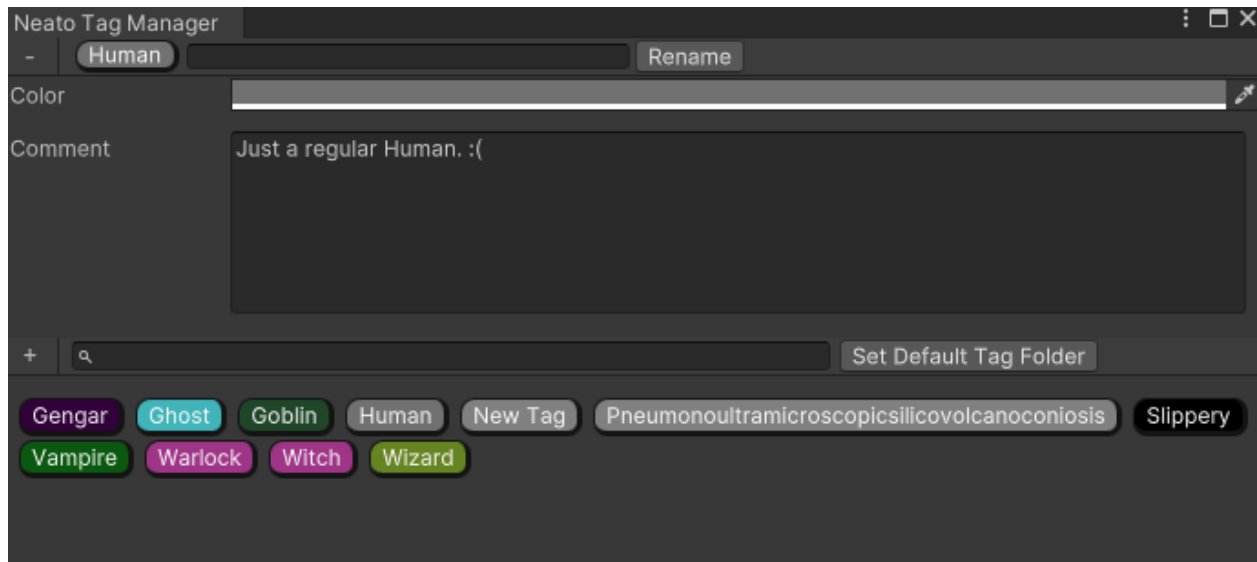
Editing Tags

Once your tag(s) are created. Now you want to edit them, right?

Neato Tags have 2 properties, Color and Comment.

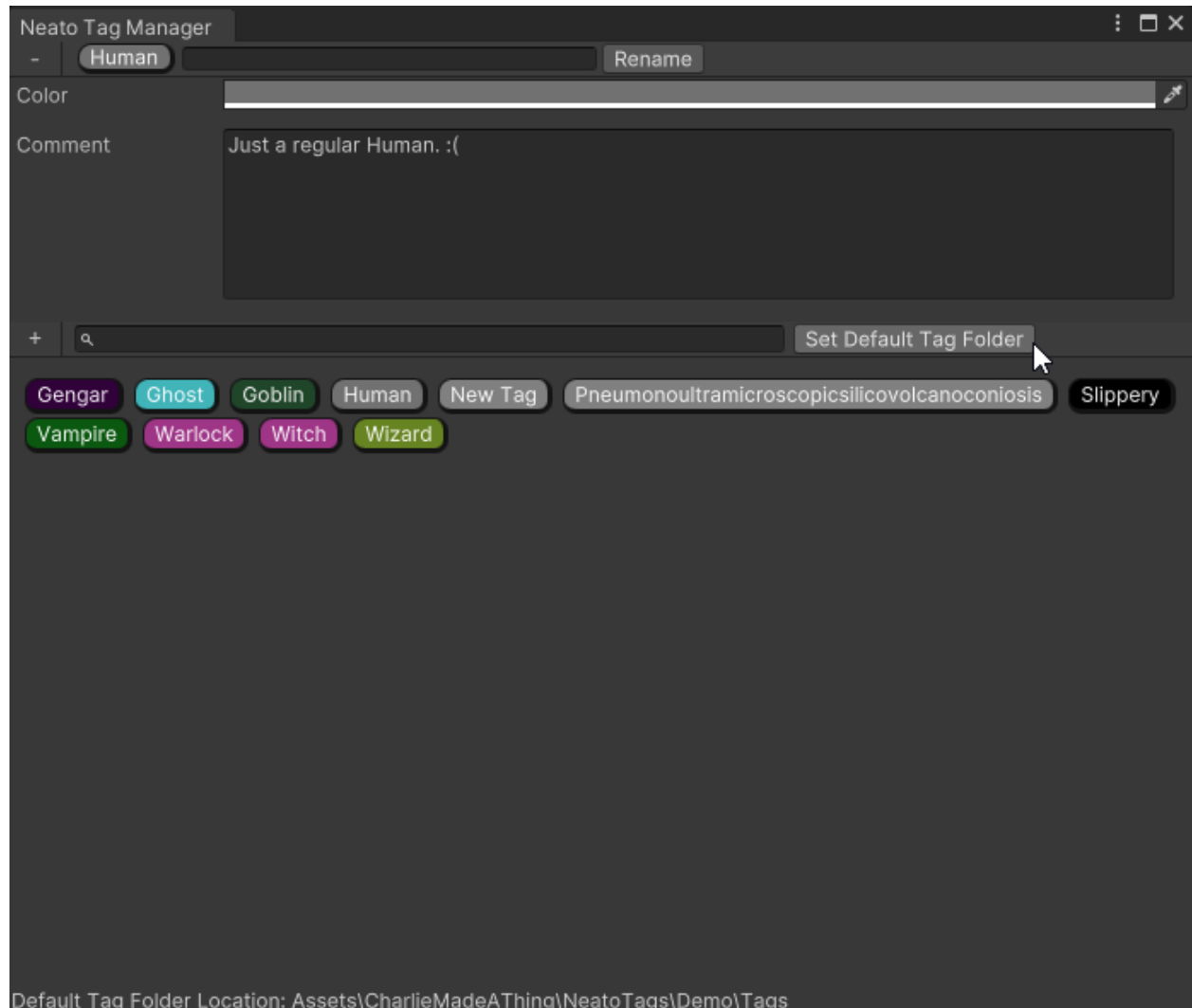
You may add more fields if needed but they might not be displayed properly or at all.

The Color field is used to give tags their color and the Comment field is to leave a note about the tag. When your mouse hovers over a tag the comment is displayed as a tooltip and when a tag is clicked in the Tagger Component it will open up the Neato Tag Manager with the tag selected.

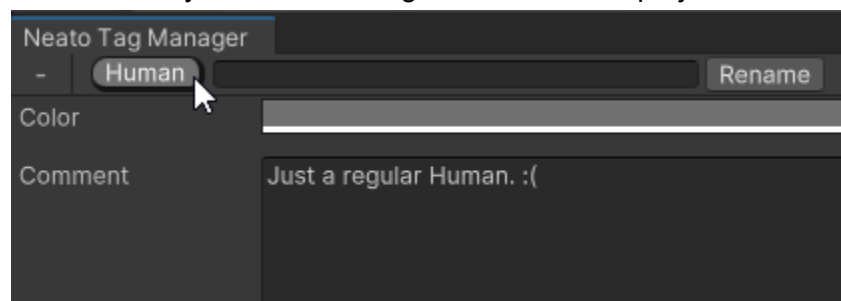


The recommended way to edit a tag is from the Neato Tag Manager but you may also click on the asset itself in the project folder and edit it from there in the inspector.

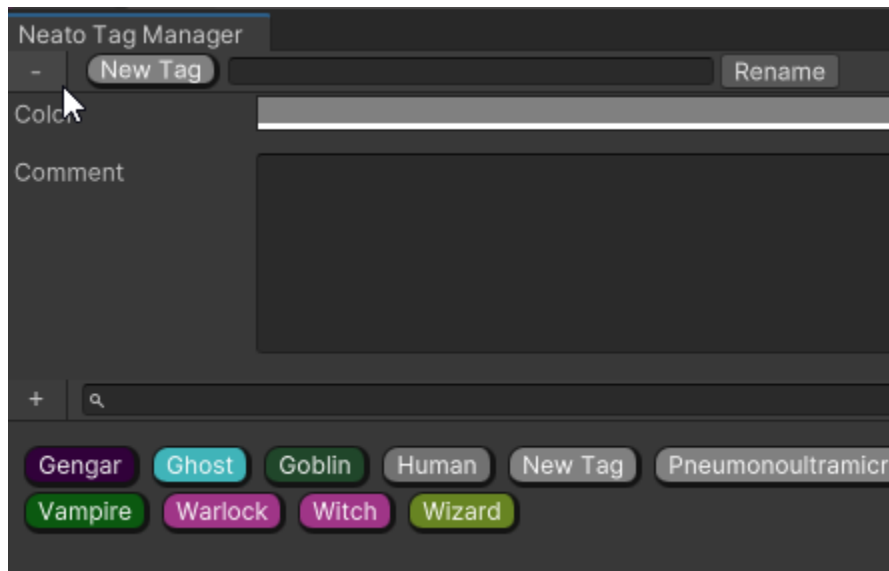
NeatoTags will find all tags in the project but it is recommended to set a default location for tags so when new ones are created you know where they are. You can do so by clicking the “Set Default Tag Folder” button in the Neato Tag Manager. The location will be displayed on the bottom of the window.



Hot Tip: Clicking the tag at the top of the Neato Tag Manager when a tag is selected will select and show you where the tag is located in the project folder.



To delete a tag either select a tag in the Neato Tag Manager and click the “-” button.



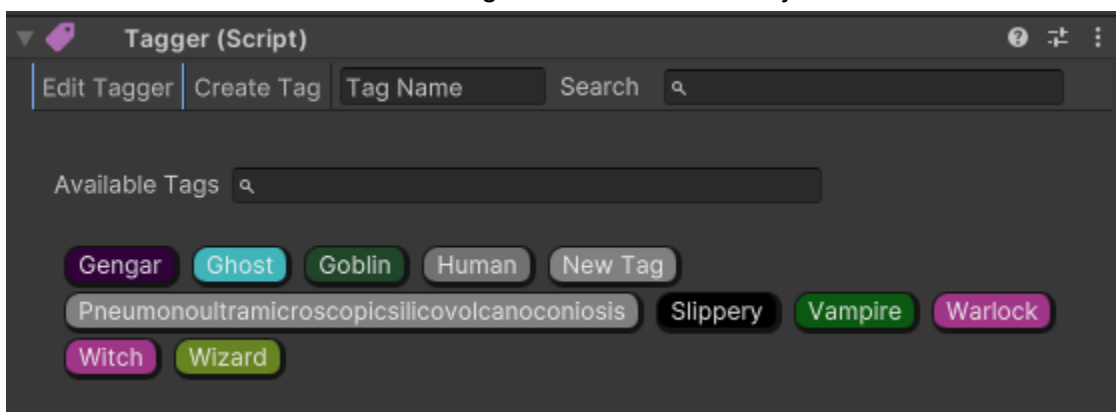
Or delete the asset directly in the project folder.

The Tagger and Tagging

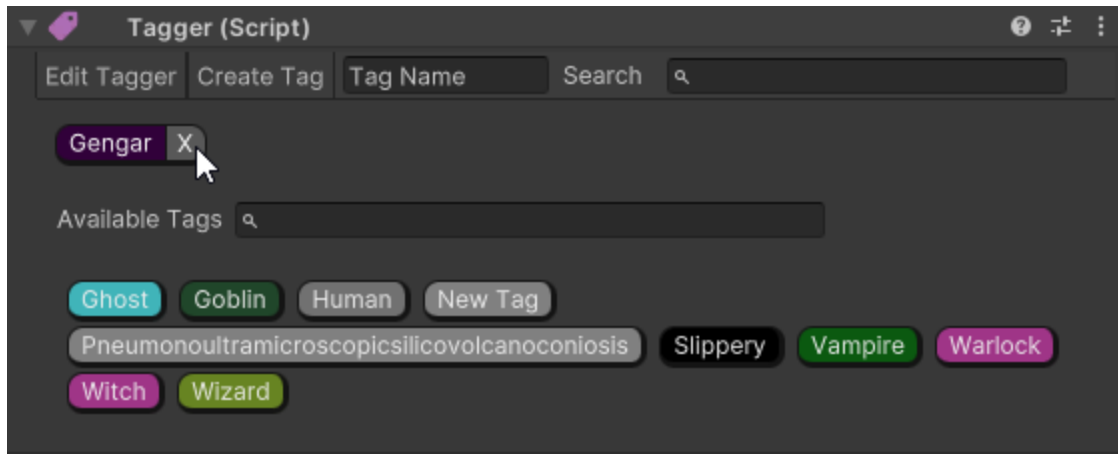
Now you have your tags and you are ready to use them!

Any GameObject that you want to contain tags will require the “**Tagger**” Component. To add a tag, click the Edit Tagger button and then click the tags you want. You can also search to quickly find what you want.

The top search bar searches the tags marked on this GameObject and the bottom search bar searches all tags not on this GameObject.



To remove a tag you can click the “X” next to the tag when Edit Tagger is clicked.



Using Tags

Now you are all set and want to find things using tags, I bet!

There are 2 ways of using functions to do things with tags.

1. gameObject

I have added extensions methods to the gameObject to access the tagger functions.

Examples:

*gameObject.HasTag()
gameObject.IsTagged()
Etc.*

2. Tagger Component

You can use it to also call functions.

Examples:

*tagger.HasTag()
tagger.AnyTagsMatch()
etc.*

Extension Methods

static bool	IsTagged (this GameObject gameObject)	Returns true if the gameObject has a Tagger component. More...
static void	AddTag (this GameObject gameObject, NeatoTagAsset tag)	Adds a tag to this gameObject. More...
static void	RemoveTag (this GameObject gameObject, NeatoTagAsset tag)	Removes a tag from this gameObject. More...
static bool	HasTag (this GameObject gameObject, NeatoTagAsset tagAsset)	Returns true if the gameObject is tagged with the given tag. More...
static bool	HasAnyTagsMatching (this GameObject gameObject, params NeatoTagAsset [] tagParams)	Returns true if the gameObject is tagged with any of the given tags. More...
static bool	HasAnyTagsMatching (this GameObject gameObject, IEnumerable< NeatoTagAsset > tagList)	Returns true if the gameObject is tagged with any of the given tags. More...
static bool	HasAllTagsMatching (this GameObject gameObject, params NeatoTagAsset [] tagParams)	Returns true if the gameObject is tagged with all of the given tags. More...
static bool	HasAllTagsMatching (this GameObject gameObject, IEnumerable< NeatoTagAsset > tagList)	Returns true if the gameObject is tagged with all of the given tags. More...
static bool	HasNoTagsMatching (this GameObject gameObject, params NeatoTagAsset [] tagList)	Return true if the gameObject is not tagged with any of the given tags. More...
static bool	HasNoTagsMatching (this GameObject gameObject, IEnumerable< NeatoTagAsset > tagList)	Return true if the gameObject is not tagged with any of the given tags. More...
static Tagger.TagFilter	TagFilter (this GameObject gameObject)	Starts a tag filter. Starts a filter for chaining filter functions. WithTag(), WithTags(), WithoutTag(), WithoutTags(), WithAnyTags() To get result call .IsMatch() or GetMatches() Can be null! More...

Tagger Methods

Public Member Functions

bool	HasTag (NeatoTagAsset tagAsset)	Checks if Tagger has a specific tag. More...
bool	AnyTagsMatch (IEnumerable< NeatoTagAsset > tagList)	Checks if Tagger has any of the tags in the list. More...
bool	AllTagsMatch (IEnumerable< NeatoTagAsset > tagList)	Checks if all of the tags in the list are in the Tagger . More...
bool	NoTagsMatch (IEnumerable< NeatoTagAsset > tagList)	Checks if Tagger doesn't have any of the tags in the list. More...
HashSet< GameObject >	GetTaggedGameObjectsWithTag (NeatoTagAsset tagAsset)	Returns a HashSet of GameObjects that have the tag. More...
HashSet< GameObject >	GetTaggedGameObjectsWithoutTag (NeatoTagAsset tagAsset)	Returns a HashSet of GameObjects that don't have the tag. More...
TagFilter	StartFilter ()	Starts a filter for chaining filter functions. WithTag(), WithTags(), WithoutTag(), WithoutTags(), WithAnyTags() To get result call .IsMatch() or .GetMatches() More...
void	AddTag (NeatoTagAsset neatoTagAsset)	Add a tag to the tagger. More...
void	RemoveTag (NeatoTagAsset neatoTagAsset)	Remove a tag from the tagger. More...

Static Public Member Functions

static HashSet< NeatoTagAsset >	GetAllTags ()	Gives back a HashSet of all tags in the project. More...
static bool	IsTagged (GameObject gameObject)	Checks if a gameObject has a Tagger component. More...
static bool	TryGetTagger (GameObject gameObject, out Tagger tagger)	Outs the Tagger component if it has one. More...
static Dictionary< GameObject, Tagger >	GetTagged ()	A Dictionary of all the gameobjects that have a Tagger component. More...

Properties

List< NeatoTagAsset >	GetTags [get]
------------------------------	----------------------