

- Developer
- / Documentation
- / Unreal Engine ▾
- / Unreal Engine 5.4 Documentation
- / Programming and Scripting
- / Blueprints Visual Scripting
- / Anatomy of a Blueprint
- / Blueprint Search

Blueprint Search

Getting the most out of searching in Blueprints, from indexing all Blueprints to advanced search syntax



Searching Blueprints

When in the Blueprint Editor, you can click on **Search** in the **Toolbar** or use **Ctrl+F** to bring up the [Find Results](#) window. Here, you can search your Blueprints for nodes, pins, pin values, graphs, variables, and variable values that match your search query. You are also able to search animation graphs.

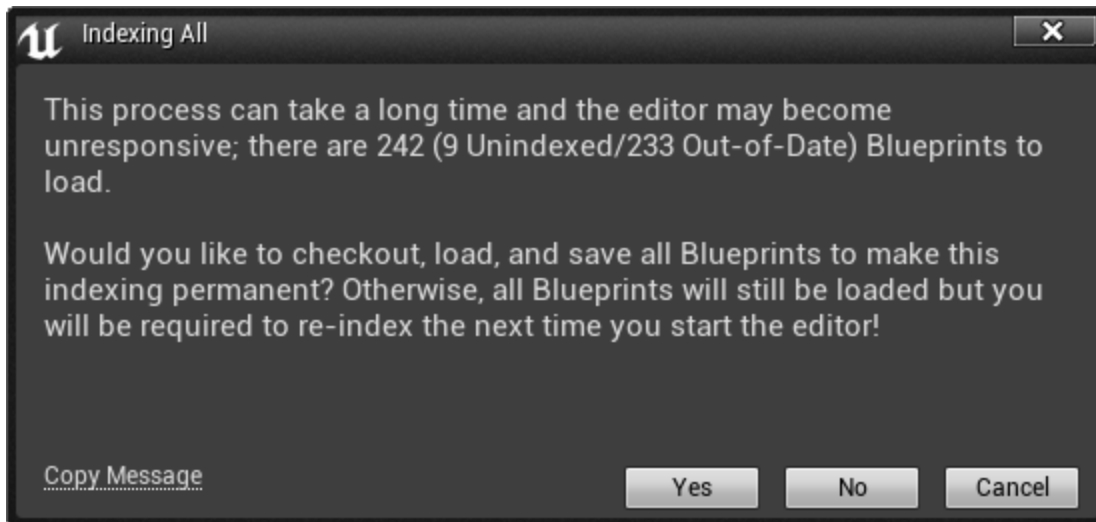
Right-clicking on a node in a Blueprint or an element in the **My Blueprint** window and clicking on **Find References** will also open the **Find Results** window, with the search field populated with the name of the element as well as the MemberGuid of the element. This ensures that you are searching for that specific variable or function, even if you have multiple instances of that name in other parts of your project.

Searching is an asynchronous process, meaning it will not prevent you from using the editor as the search completes. You can also make multiple searches in different Blueprints at the same time. It leverages the Asset Registry for the searchable data, so the most recently indexed data is always with the asset file.

By default, this tool searches your current Blueprint, but by unchecking the **Find in Current Blueprint Only** checkbox, you can search all Blueprints in the project. When Blueprints are not indexed in the system, it will inform you, as in the below image:



Clicking **Index All** results in an extremely slow process that will stall the editor, as it loads all un-indexed Blueprints and re-saves them to cache searchable data. If you do not wish to re-save the content, or the content is protected by source control and cannot be checked out, it will simply load all Blueprint assets to cache the latest searchable data in memory. However, if you do not re-save the content, **Index All** will need to be done each time you open the editor.



Advanced Search Syntax

Filters are an advanced search feature for targeting searches to specific subsets of data in Blueprints. For example, you can isolate only Blueprints that have nodes named something specific or properties with certain names and flags set. These can be nested and combined to form advanced queries for very specific needs. Certain filters can only be used within other filters, but all filters can be used as a starting point. Below is a non-exhaustive list of searchable tags and with what sort of data that are used for. All tags can be searched for without the use of filters.

Some items can only be searched using tags to help prevent clutter when doing searches without tags. These items are marked below using (Explicit). Currently, filtering is limited to user-added member variables.

An example search string is `Nodes(Name=Coin)`, which will find all nodes in the Blueprint that contain "coin" in the name.

Filter	Searchable Tags	Subfilters
Blueprint	<ul style="list-style-type: none">• Name• ParentClass (Explicit)• Path (Explicit)• Interfaces (Explicit)	<ul style="list-style-type: none">• Graphs• Functions• Macros• EventGraphs• Nodes• Pins• Variables/Properties• Components
<ul style="list-style-type: none">• Graphs• Functions• Macros• EventGraphs	<ul style="list-style-type: none">• Name• Description	<ul style="list-style-type: none">• Nodes• Pins• Variables/Properties (for local variables)
Nodes	<ul style="list-style-type: none">• Name• ClassName (Explicit)• Comment	Pins

Filter	Searchable Tags	Subfilters
<ul style="list-style-type: none"> • Pins • Variables/Properties • Components 	<ul style="list-style-type: none"> • Name • DefaultValue • IsArray (Explicit) • IsReference (Explicit) • IsSCSCComponent (Explicit) • PinCategory (Explicit) • ObjectClass (Explicit) 	



PinCategory is the type of pin, "bool", "string", "Actor", "MyBlueprint". ObjectClass is the struct or object represented by the pin/variable/component.

The All Subfilter

All is a special subfilter. When nested in another filter, it will allow any data contained within it to be tested against all children of any objects that pass the filter otherwise.

Example: `Graphs(Name=MyFunction && All(Return))`

Any graph that contains the name "MyFunction" will test all of its children (nodes, local variables, and pins) for the string "Return".