**Query an index, including syntax, sorting, filtering, and wildcards**

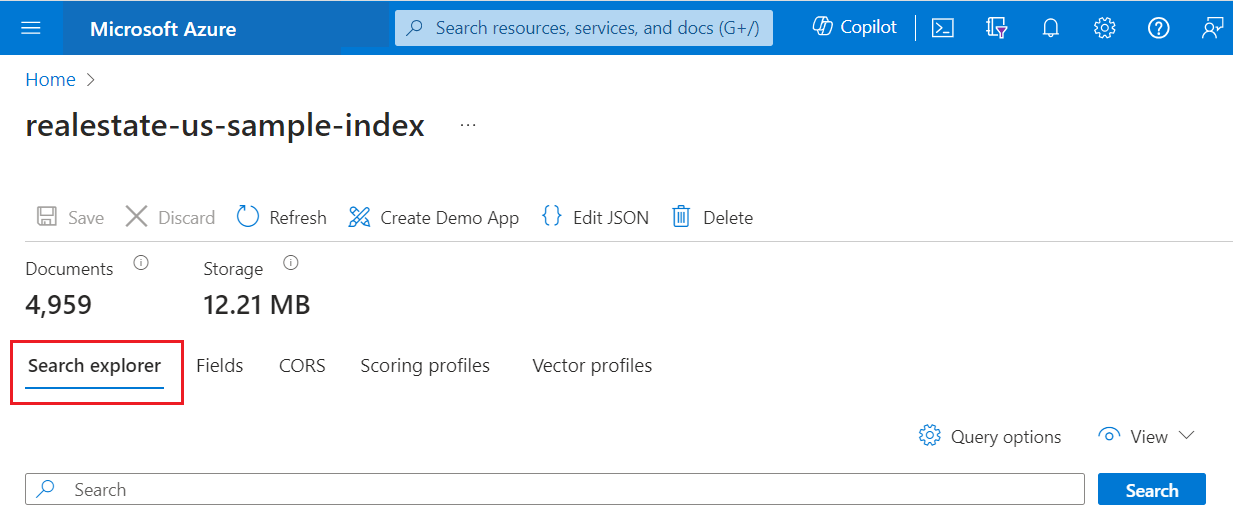
**Search explorer**, a built-in query tool in the Azure portal used for running queries against a search index in Azure AI Search. Use it to test a query or filter expression, or confirm whether content exists in the index.

**Start Search explorer**

1. In the [Azure portal](https://portal.azure.com/), open the search overview page from the dashboard or [find your service](https://portal.azure.com/#blade/HubsExtension/BrowseResourceBlade/resourceType/Microsoft.Search%2FsearchServices).
2. Open Search explorer from the command bar:

Screenshot of the Search explorer command in portal.

Or use the embedded **Search explorer** tab on an open index:



**Query two ways**

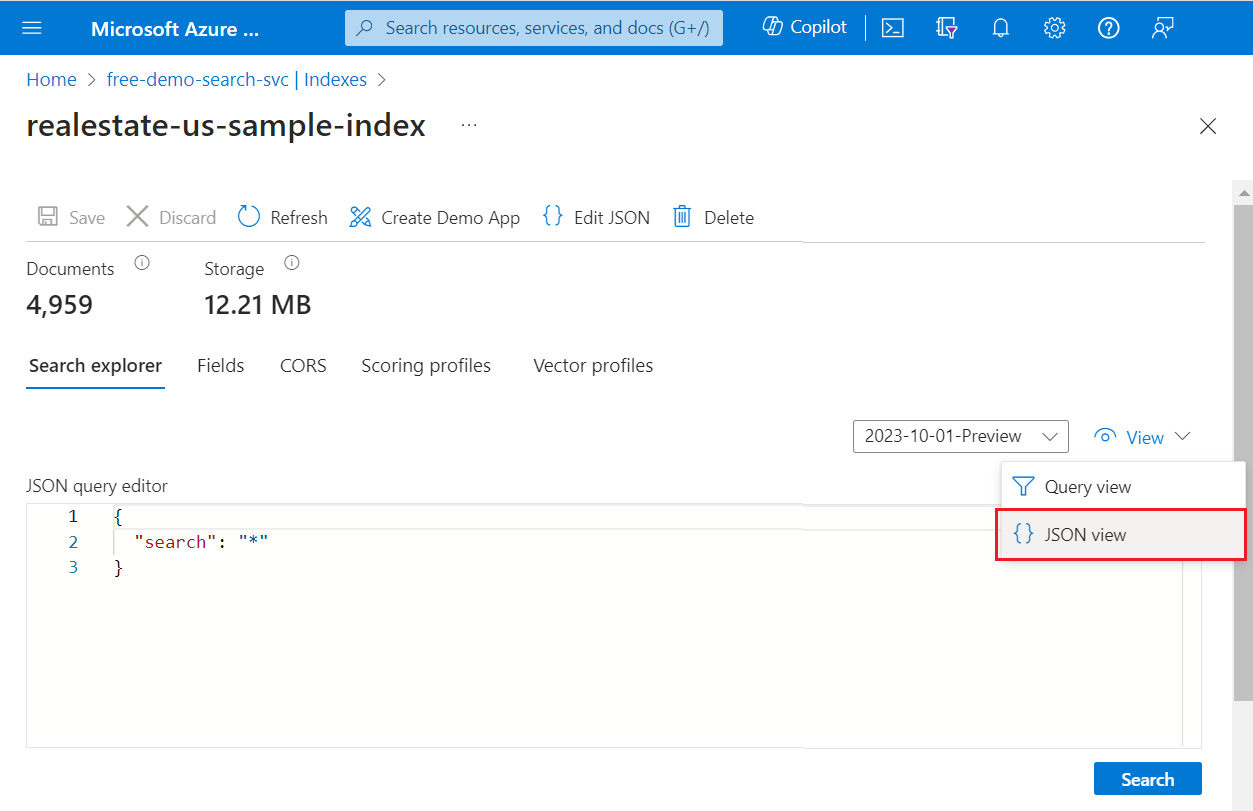
There are two approaches for querying in Search explorer.

* Query view provides a default search bar. It accepts an empty query or free text query with booleans. For example, seattle condo +parking.
* JSON view supports parameterized queries. Filters, orderby, select, count, searchFields, and all other parameters must be set in JSON view.

**Tip**

JSON view provides intellisense for parameter name completion. Place the cursor inside the JSON view and type a space character to show a list of all query parameters, or type a single letter like "s" to show just the query parameters starting with "s". Intellisense doesn't exclude invalid parameters so use your best judgement.

Switch to **JSON view** for parameterized queries. The examples in this article assume JSON view throughout. You can paste JSON examples from this article into the text area.



**Run an unspecified query**

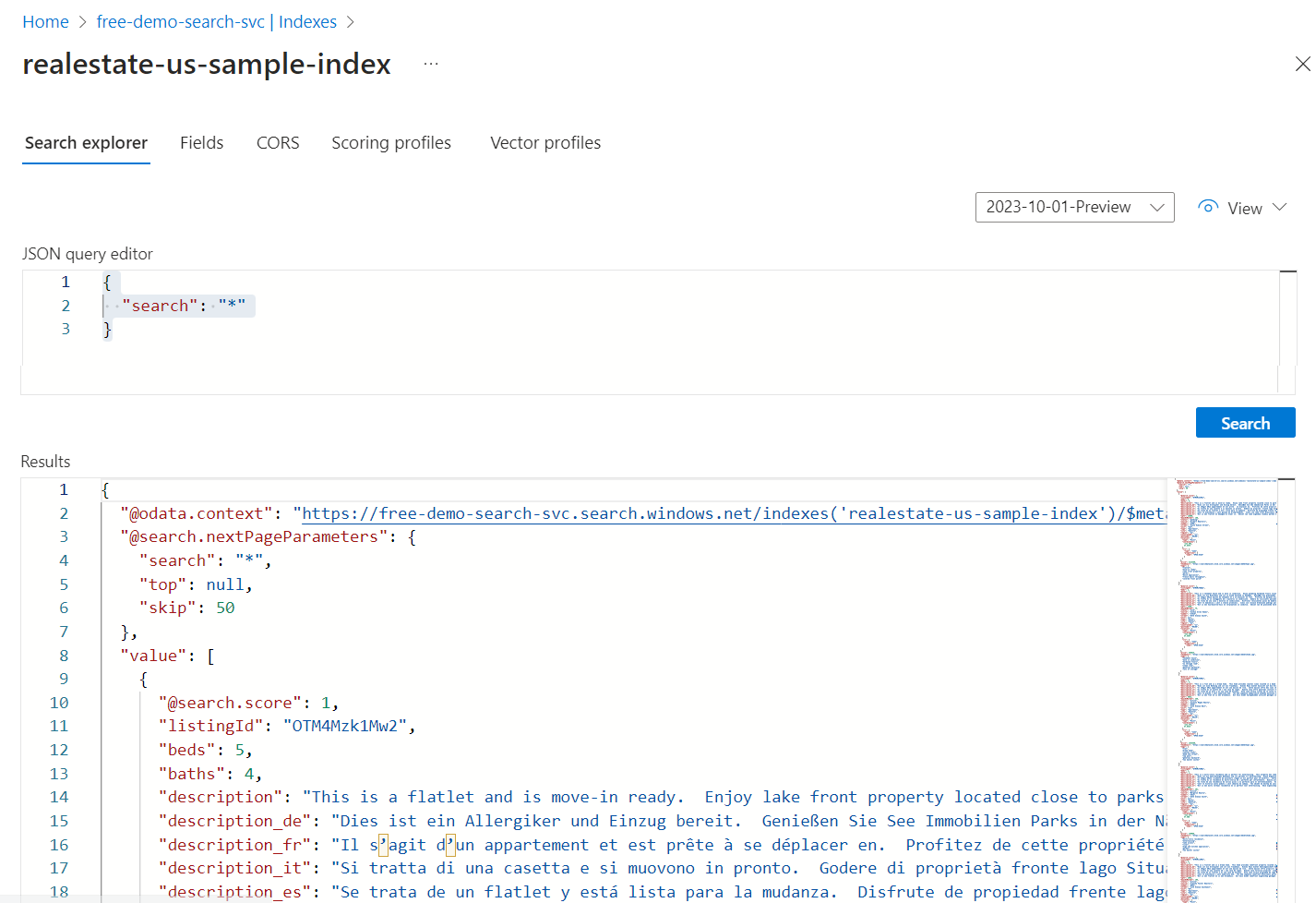
In Search explorer, POST requests are formulated internally using the [Search POST REST API](https://learn.microsoft.com/en-us/rest/api/searchservice/documents/search-post?view=rest-searchservice-2024-05-01-preview&preserve-view=true), with responses returned as verbose JSON documents.

For a first look at content, execute an empty search by clicking **Search** with no terms provided. An empty search is useful as a first query because it returns entire documents so that you can review document composition. On an empty search, there's no search score and documents are returned in arbitrary order ("@search.score": 1 for all documents). By default, 50 documents are returned in a search request.

Equivalent syntax for an empty search is \* or "search": "\*".



**Results**



**Free text search**

Free-form queries, with or without operators, are useful for simulating user-defined queries sent from a custom app to Azure AI Search. Only those fields attributed as "searchable" in the index definition are scanned for matches.

You don't need JSON view for a free text query, but we provide it in JSON for consistency with other examples in this article.

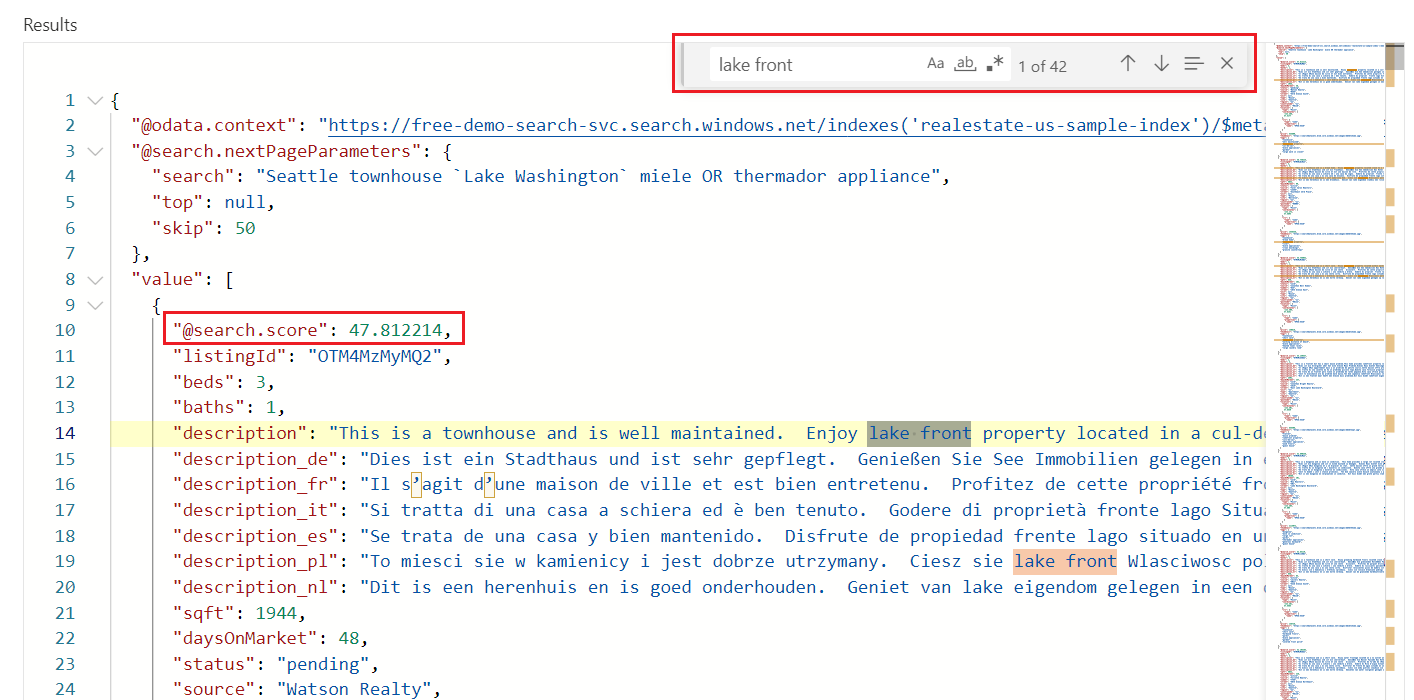
Notice that when you provide search criteria, such as query terms or expressions, search rank comes into play. The following example illustrates a free text search. The "@search.score" is a relevance score computed for the match using the [default scoring algorithm](https://learn.microsoft.com/en-us/azure/search/index-ranking-similarity#default-scoring-algorithm).

A close-up of a computer screen

Description automatically generated

**Results**

You can use Ctrl-F to search within results for specific terms of interest.



**Count of matching documents**

Add "count": true to get the number of matches found in an index. On an empty search, count is the total number of documents in the index. On a qualified search, it's the number of documents matching the query input. Recall that the service returns the top 50 matches by default, so the count might indicate more matches in the index than what's returned in the results.

A close-up of a computer screen

Description automatically generated

**Results**



**Limit fields in search results**

Add ["select"`](https://learn.microsoft.com/en-us/azure/search/search-query-odata-select) to limit results to the explicitly named fields for more readable output in **Search explorer**. Only fields marked as "retrievable" in the search index can show up in results.

A close-up of a white envelope

Description automatically generated

**Results**



**Return next batch of results**

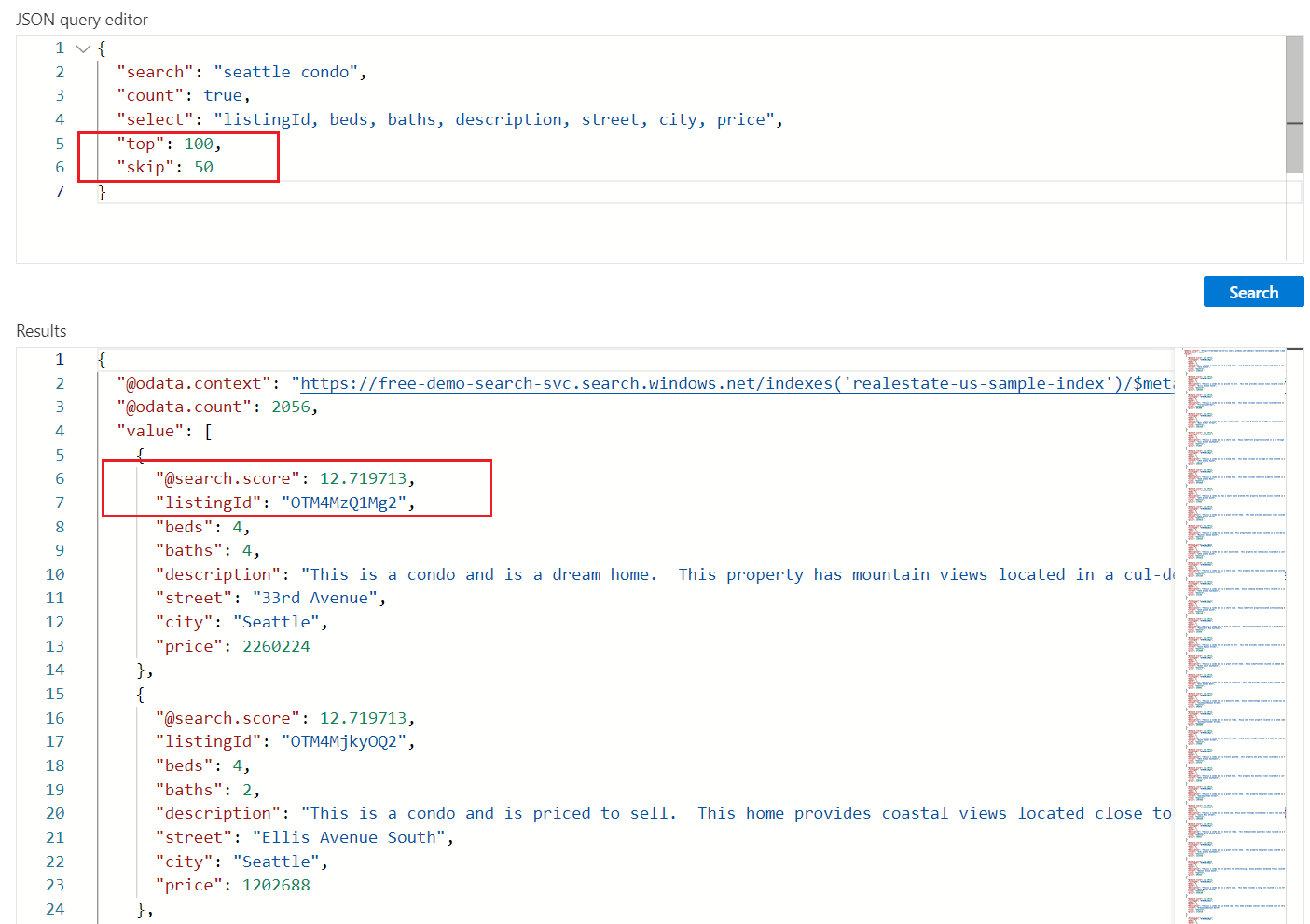
Azure AI Search returns the top 50 matches based on the search rank. To get the next set of matching documents, append "top": 100 and "skip": 50 to increase the result set to 100 documents (default is 50, maximum is 1000), skipping the first 50 documents. You can check the document key (listingID) to identify a document.

Recall that you need to provide search criteria, such as a query term or expression, to get ranked results. Notice that search scores decrease the deeper you reach into search results.

A close-up of a computer screen

Description automatically generated

**Results**



**Filter expressions (greater than, less than, equal to)**

Use the [filter](https://learn.microsoft.com/en-us/azure/search/search-query-odata-filter) parameter to specify inclusion or exclusion criteria. The field must be attributed as "filterable" in the index. This example searches for bedrooms greater than 3:

A screen shot of a computer

Description automatically generated

**Results**



**Sorting results**

Add [orderby](https://learn.microsoft.com/en-us/azure/search/search-query-odata-orderby) to sort results by another field besides search score. The field must be attributed as "sortable" in the index. In situations where the filtered value is identical (for example, same price), the order is arbitrary, but you can add more criteria for deeper sorting. An example expression you can use to test this out is:

A close-up of a computer screen

Description automatically generated

**Results**



**Takeaways**

In this quickstart, you used **Search explorer** to query an index using the REST API.

* Results are returned as verbose JSON documents so that you can view document construction and content, in entirety. The select parameter in a query expression can limit which fields are returned.
* Search results are composed of all fields marked as "retrievable" in the index. Select the adjacent **Fields** tab to review attributes.
* Keyword search, similar to what you might enter in a commercial web browser, are useful for testing an end-user experience. For example, assuming the built-in real estate sample index, you could enter "Seattle apartments lake washington", and then you can use Ctrl-F to find terms within the search results.
* Query and filter expressions are articulated in a syntax implemented by Azure AI Search. The default is a [simple syntax](https://learn.microsoft.com/en-us/rest/api/searchservice/simple-query-syntax-in-azure-search), but you can optionally use [full Lucene](https://learn.microsoft.com/en-us/rest/api/searchservice/lucene-query-syntax-in-azure-search) for more powerful queries. [Filter expressions](https://learn.microsoft.com/en-us/rest/api/searchservice/odata-expression-syntax-for-azure-search) are articulated in an OData syntax.

A screenshot of a computer

Description automatically generated

A person pointing at a screen

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Keyword search by using REST**

<https://learn.microsoft.com/en-us/azure/search/search-get-started-rest?tabs=azure-cli>

**Full text search using the Azure SDKs**

<https://learn.microsoft.com/en-us/azure/search/search-get-started-text?tabs=dotnet>