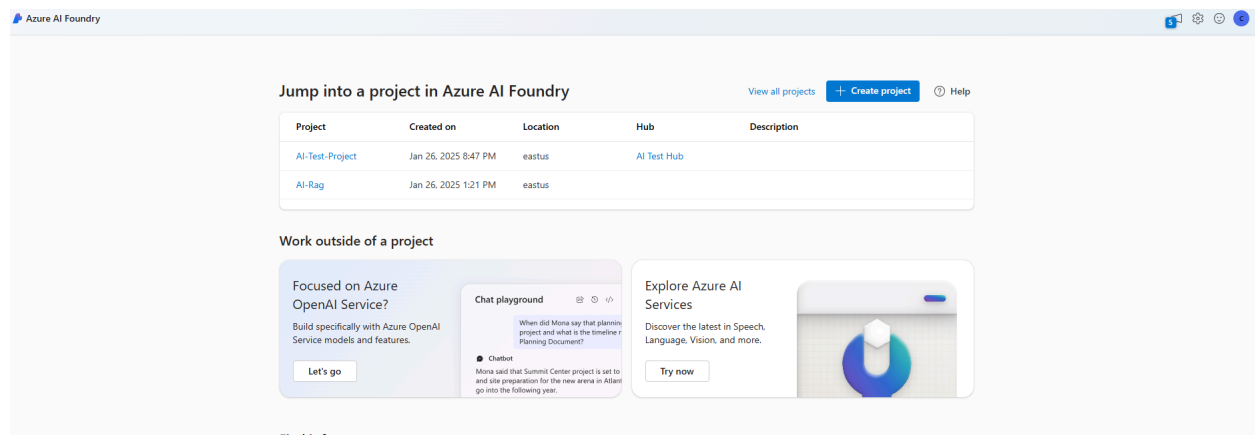
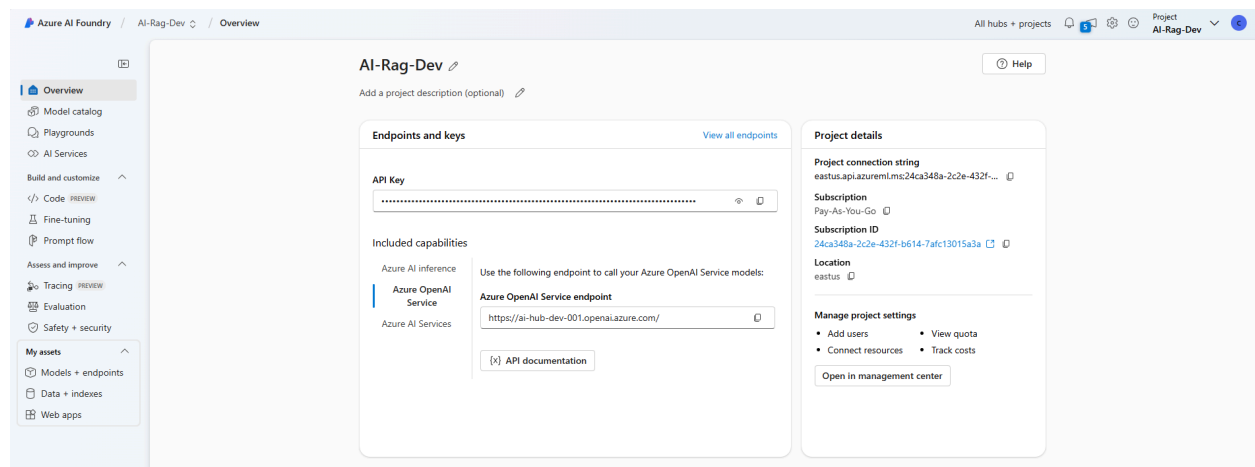


## Set up resource group for Hub

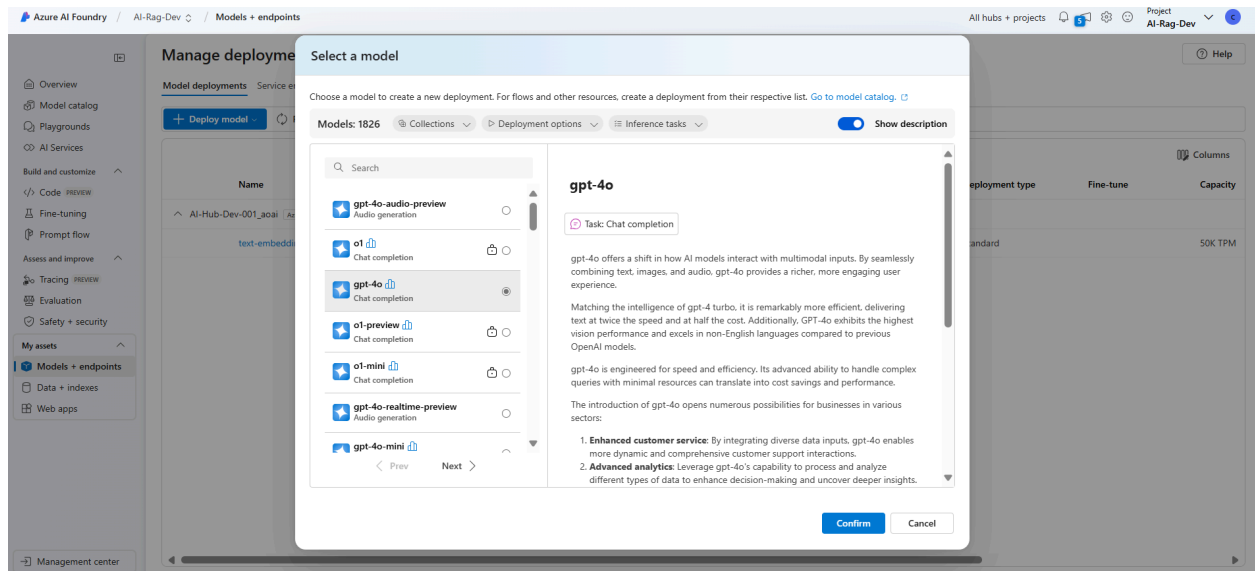
Create AI hub under Azure AI Foundry. This will include storage, networking, encryption, identity and tags. Note the networking for all projects under the hub will share the same networking. If you get a networking error during creation you will need to have a unique name for the “Connect AI Services incl. OpenAI” property. Once resource is create navigate to <https://ai.azure.com/> (hub dashboard)



Create a new project under the hub created. Note regardless a project needs to have a Hub as a parent if one is not specified that one by default is created. Once the project is created and you navigate to the dashboard for the project you will see something similar to this. Note the project name in the top left “AI-Rag-Dev”.



Deploy base model. Select Models + endpoints. Note you will need to deploy a model at allows function calling if you are using function calling in flow. I will deploy gpt-4o.



Select deployment type I am selecting Global Standard.

## Deploy model gpt-4o

Deployment name \*



gpt-4o

Deployment type

Global Standard



Global Standard: Pay per API call with the highest rate limits. Learn more about [Global deployment types](#).

Data might be processed globally, outside of the resource's Azure geography, but data storage remains in the AI resource's Azure geography. Learn more about [data residency](#).

### Deployment details



Customize

Model version  
2024-11-20

Connected AI resource  
AI-Hub-Dev-001\_aoui

Project  
AI-Rag-Dev

Authentication type  
Key

Capacity  
10K tokens per minute (TPM)

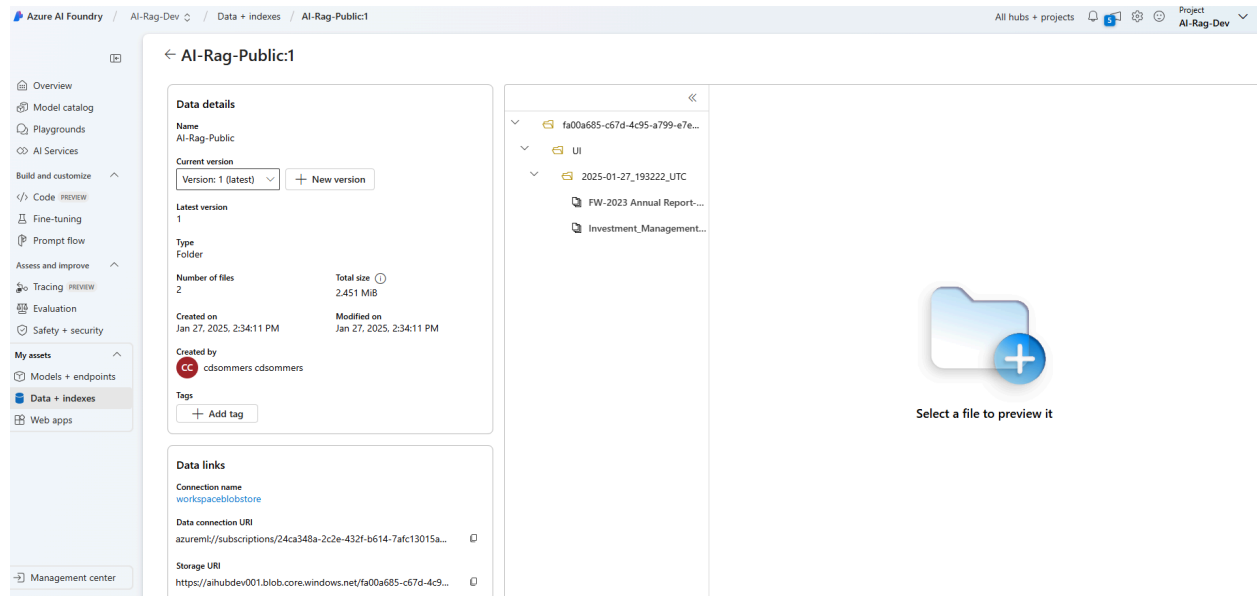
Resource location  
East US

Content safety  
DefaultV2

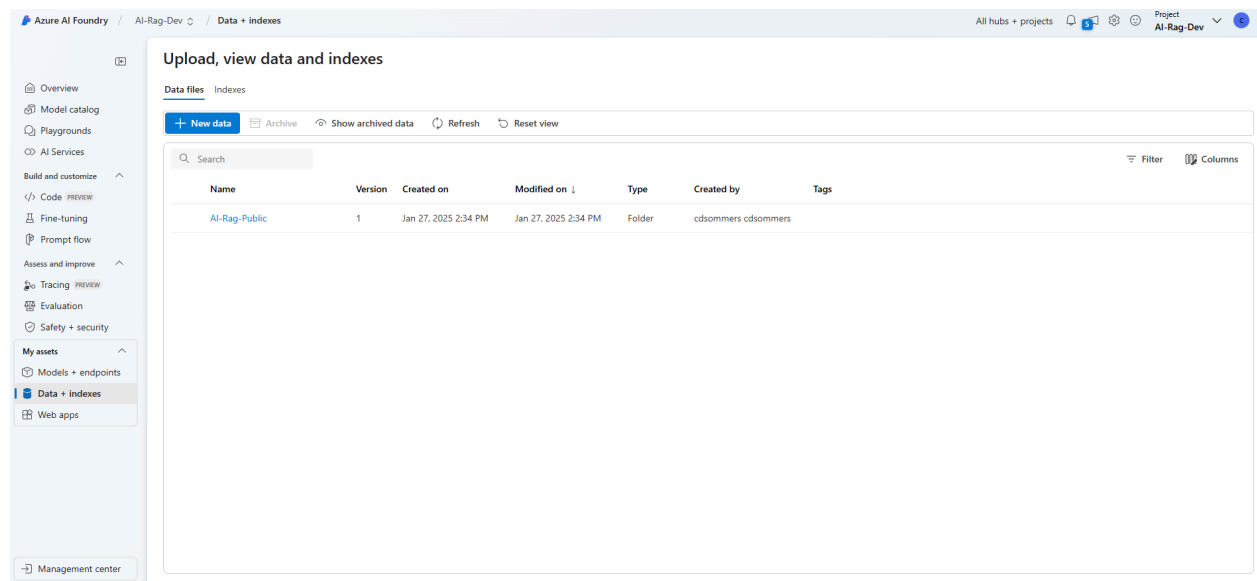
Deploy

Cancel

Create a data store. Click Date + indexes and walk through the wizard.



Note the connection name at the bottom left. The documents are stored within a blob storage.



The Data store should be visible on the Data + indexes pane.

Create an index on the data. Click the “Indexes” tab on the Data + indexes pane. When creating a vector index click “Data in Azure AI foundry” to access the data stored in the data store in the previous step. (see below)

Create a vector indexPREVIEW

1Source location

2Index configuration

3Search settings

4Review and finish

Choose to location of the input data source that you want to index.

Data source \* ⓘ

Select your data source

Connect to an existing index

An Azure AI Search resource ⓘ

Search resource

Azure AI Search

MongoDB Atlas

Create a new index

Data in Azure AI Foundry

Azure Blob Storage

Storage URL

Upload files

Next

Create vector indexCancel

For index configuration you will need to create a new AI search service. This should open a new tab to create a new vector search service.

Home >

Create a search service

BasicsScaleTagsReview + create

Project details

Subscription \*

Pay-As-You-Go

Resource Group \*

AI-Hub-Dev

Create new

Instance Details

Service name \* ⓘ

ai-rag-search-service-dev

Location \*

East US

Pricing tier \* ⓘ

Free

50 MB, max 1 replicas, max 1 partitions, max 1 search units

Change Pricing Tier

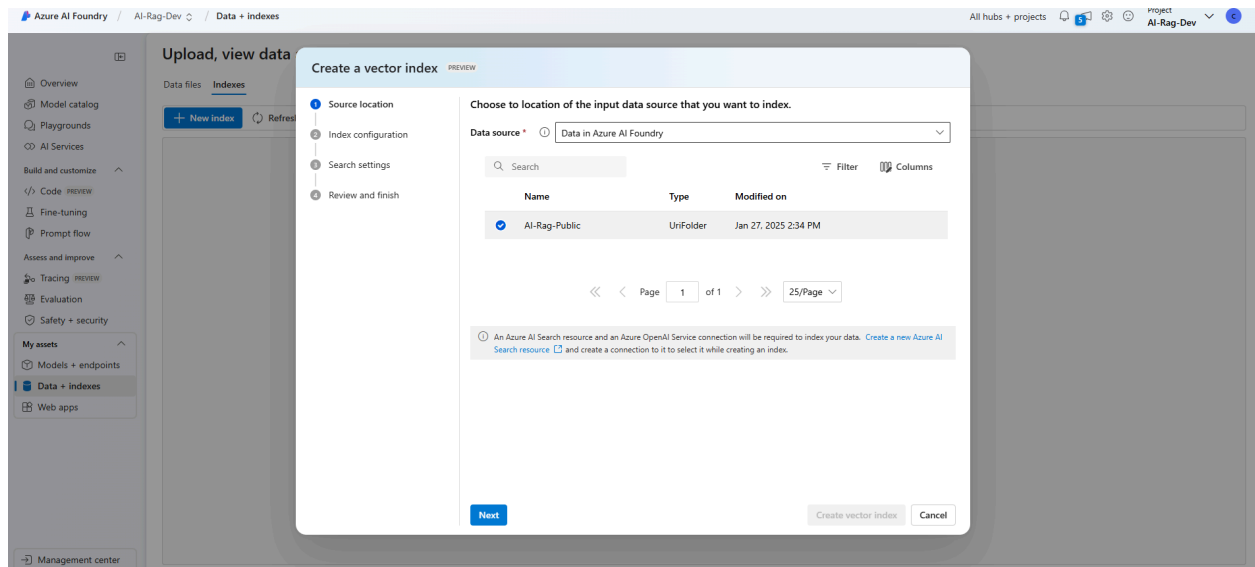
Review + create

Previous

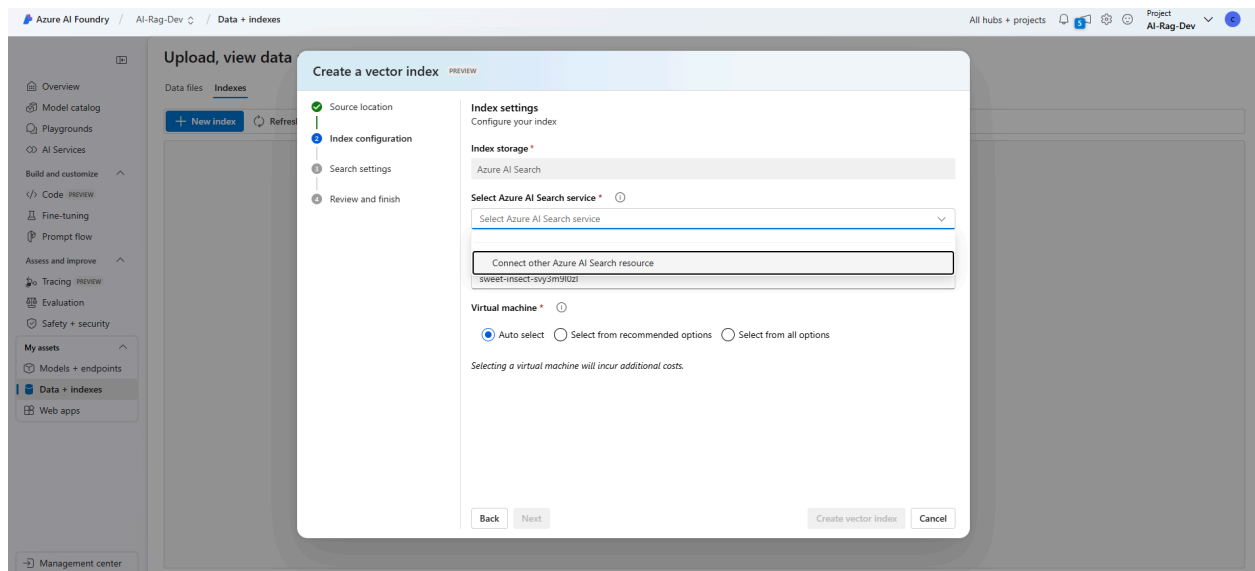
Next: Scale

Note the pricing tier in this case I leverage a free tier for dev. Please take note of the pricing for each tier.

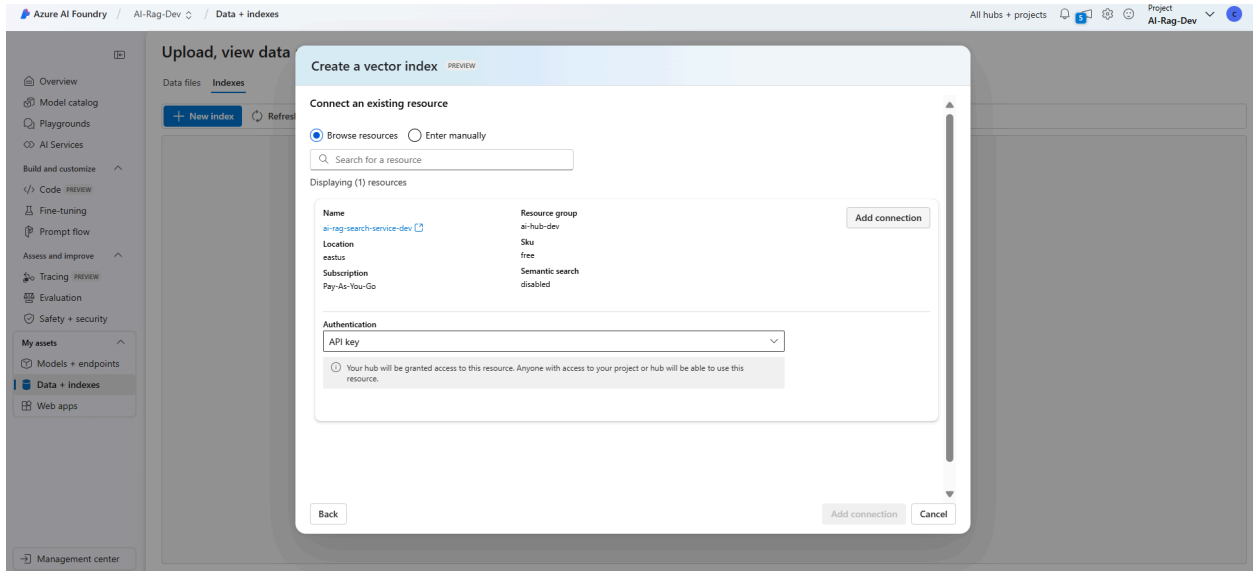
Once the search service is created create go back to the indexes tab and create index.  
Select the data store in the previous step



Select connect to azure search resource



Select add connection to search service.



Give the vector index a logical name

Create a vector indexPREVIEW

✓ Source location

2 Index configuration

3 Search settings

4 Review and finish

Index settings

Configure your index

Index storage \*

Azure AI Search

Select Azure AI Search service \* ⓘ

airagsearchservice-dev

Create a new Azure AI Search resource

Vector index \* ⓘ

ai-rag-index-public

Virtual machine \* ⓘ

☒ Auto select

☐ Select from recommended options

☐ Select from all options

Selecting a virtual machine will incur additional costs.

Back

Next

Create vector index

Cancel

Create a vector indexPREVIEW

✓ Source location

✓ Index configuration

3 Search settings

4 Review and finish

Configure search settings

Combining hybrid retrieval with semantic ranking (Hybrid + Semantic) gives most accurate search results for generative AI applications. To generate vector index, embedding model is required.

Vector settings

☒ Add vector search to this search resource

Azure OpenAI Service connection \* ⓘ

AI-Hub-Dev-001\_aoai

ⓘ This resource requires an embedding model. If you don't have one already, **text-embedding-ada-002 (Version 2)** will be deployed for you. Using vector embeddings will incur usage to your account. [View Azure OpenAI Service pricing](#)

BackNext

Create vector indexCancel

Click create vector index.

Azure AI Foundry

AI-Rag-Dev

Data + indexes

ai-rag-index-public

All hubs + projects

Project AI-Rag-Dev

Overview

Model catalog

Playgrounds

AI Services

Build and customize

Code

Fine-tuning

Prompt flow

Assess and improve

Tracing

Evaluation

Safety + security

My assets

Models + endpoints

Data + indexes

Web apps

Management center

ai-rag-index-public

Status

Running

Refresh

Version

-

Embed with model

No

Source type

Azure AI On Your Data

Vector store

-

Total indexing time

-

Compute

Serverless compute

Created on

Jan 27, 2025, 3:04:07 PM

Created by

cdsommers cdsommers

Job details

Test data

Source data

Name	Type	Size
FW-2023 Annual Report-ACC...	.pdf	1.58 MB
Investment_Management_Busi...	.pdf	891.47 KB

Status

Step 1 of 3

Allocating compute resource - In progress

Vector index job should start. This should take some time.



Azure AI Foundry / AI-Rag-Dev / Data + indexes / ai-rag-index-public

← ai-rag-index-public

**Status**

Completed

Version

Embed with model

No

Source type

Azure AI On Your Data

Vector store

-

Total indexing time

8m

Compute

Serverless compute

Created on

Jan 27, 2025, 3:04:07 PM

Created by

cdsommers cdsommers

Test data

**Source data**

Name	Type	Size
FW-2023 Annual Report-ACC...	.pdf	1.58 MB
Investment_Management_Busi...	.pdf	891.47 KB

**Status**

Step 1 of 3

Cracking and chunking - Completed

Step 2 of 3

Creating Azure AI Search Index - Completed

Step 3 of 3

Registering Index - Completed

Azure AI Foundry / AI-Rag-Dev / Data + Indexes

Upload, view data and indexes

Data files Indexes

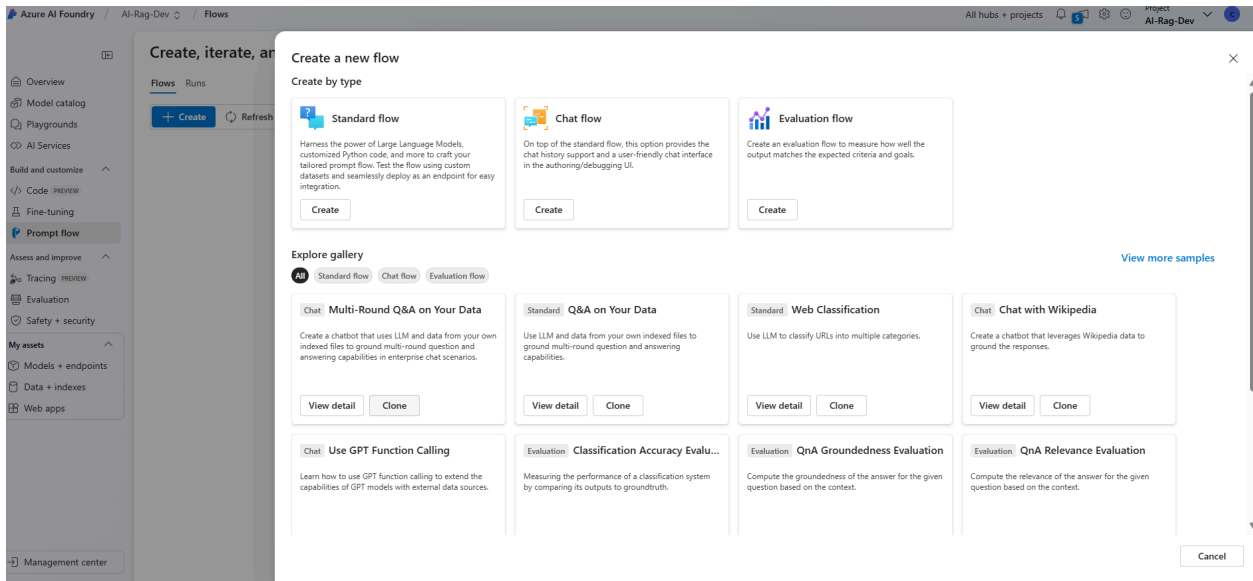
+ New index Refresh Reset view

Search Filter

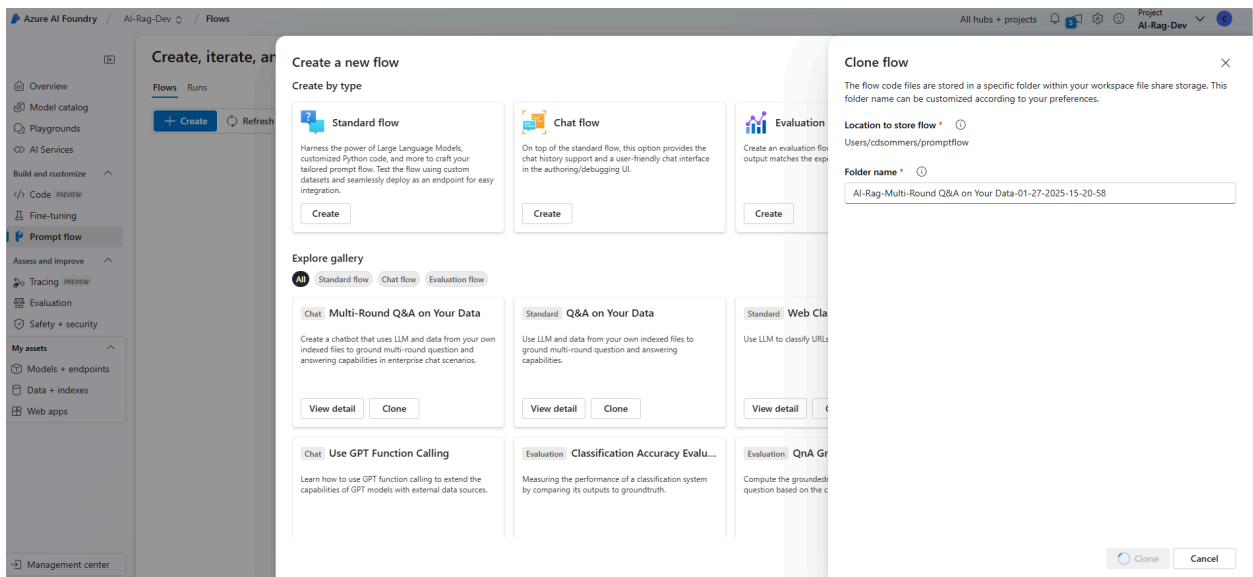
Vector index name	Version	Created by	Data source	Status	Updated on ↓
ai-rag-index-public	1	cdsommers cdsommers	Azure AI On Your Data	Ready	Jan 27, 2025, 3:12:35 PM

Once completed the index should have a status of ready.

Create prompt flow. Create Prompt flow on left pane.  
Clone “multi round Q&A on your data”



Create folder workspace



A new prompt flow should be generated

Azure AI Foundry / AI-Rag-Dev / Flows / AI-RAG

AI-RAG Chat View batch runs Clone Save Deploy Evaluate Start compute session Chat

Flow

Inputs

Name	Type	Value
mlindex_content	string	
queries	object	\$(modify_query_with_history.output)
query_type	string	
top_k	int	2

generate\_prompt\_context python

```
1 from typing import List
2 from promptflow import tool
3 from promptflow_vectordb.core.contracts import SearchResultEntity
4
5
6 @tool
7 def generate_prompt_context(search_result: List[dict]) -> str:
8     def format_doc(doc: dict):
9         return f"Content: {doc['Content']}\nSource: {doc['Source']}"
```

Graph

inputs

modify\_query\_with\_history

lookup

generate\_prompt\_context

Prompt\_variants

chat\_with\_context

outputs

Start compute session

Azure AI Foundry / AI-Rag-Dev / Flows / AI-RAG

AI-RAG Chat View batch runs Clone Save Deploy Evaluate Compute session running Chat

Flow

Inputs

Name	Type	Value
mlindex_content	string	
queries	object	\$(modify_query_with_history.output)
query_type	string	
top_k	int	2

generate\_prompt\_context python

```
1 from typing import List
2 from promptflow import tool
3 from promptflow_vectordb.core.contracts import SearchResultEntity
4
5
6 @tool
7 def generate_prompt_context(search_result: List[dict]) -> str:
8     def format_doc(doc: dict):
9         return f"Content: {doc['Content']}\nSource: {doc['Source']}"
```

Graph

inputs

modify\_query\_with\_history

lookup

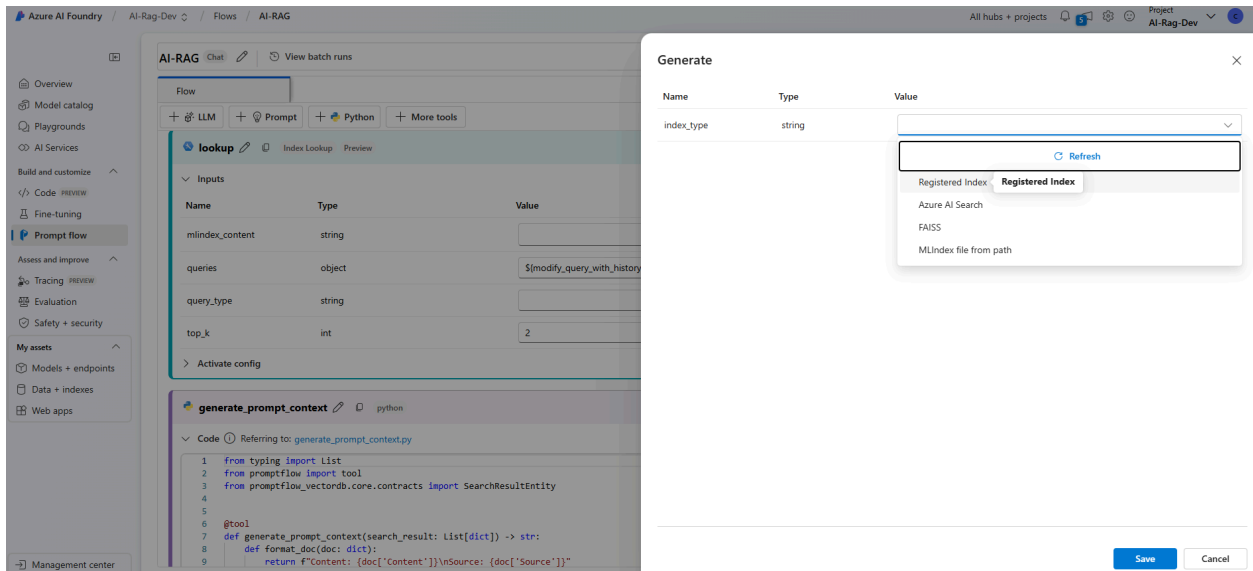
generate\_prompt\_context

Prompt\_variants

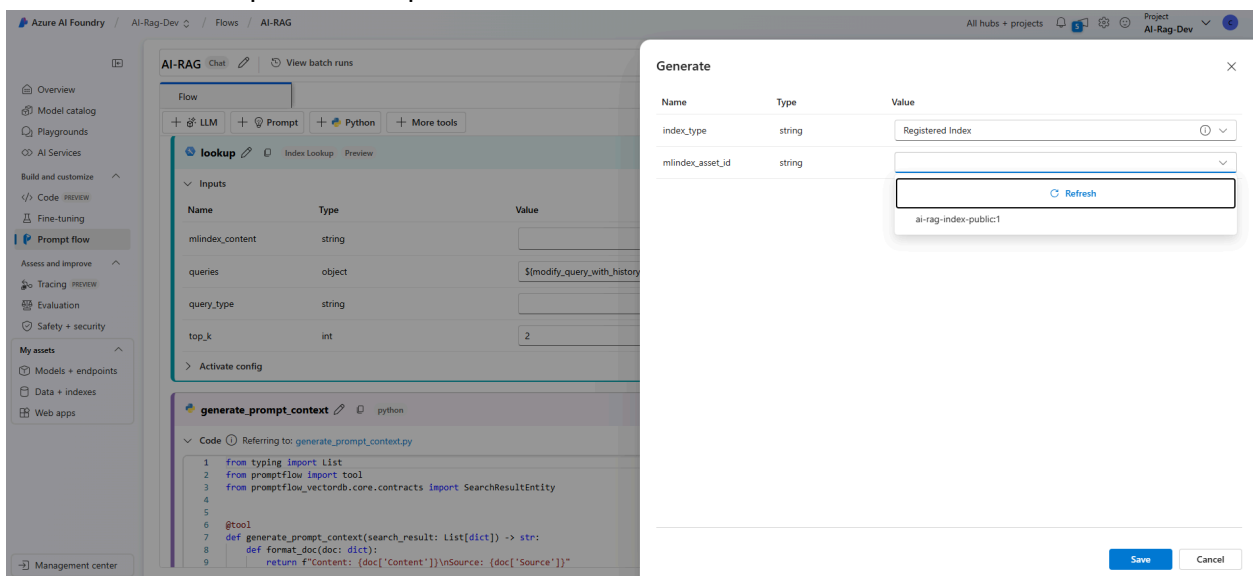
chat\_with\_context

outputs

Select lookup node once session is started. Select mlindex\_content and select registered index

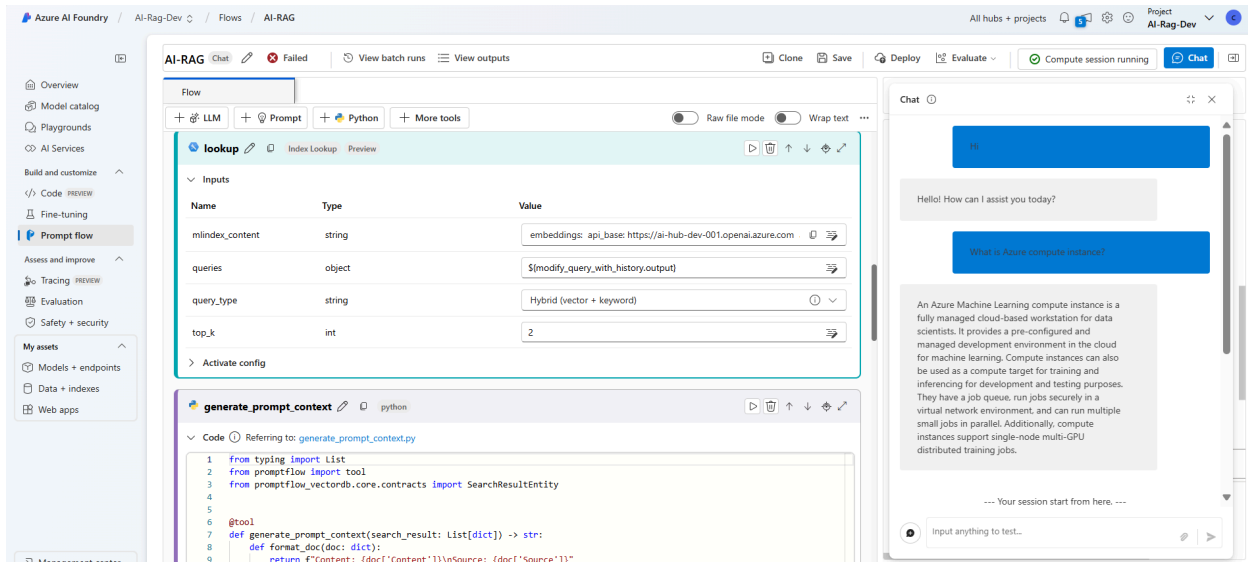


Select the index in previous steps.

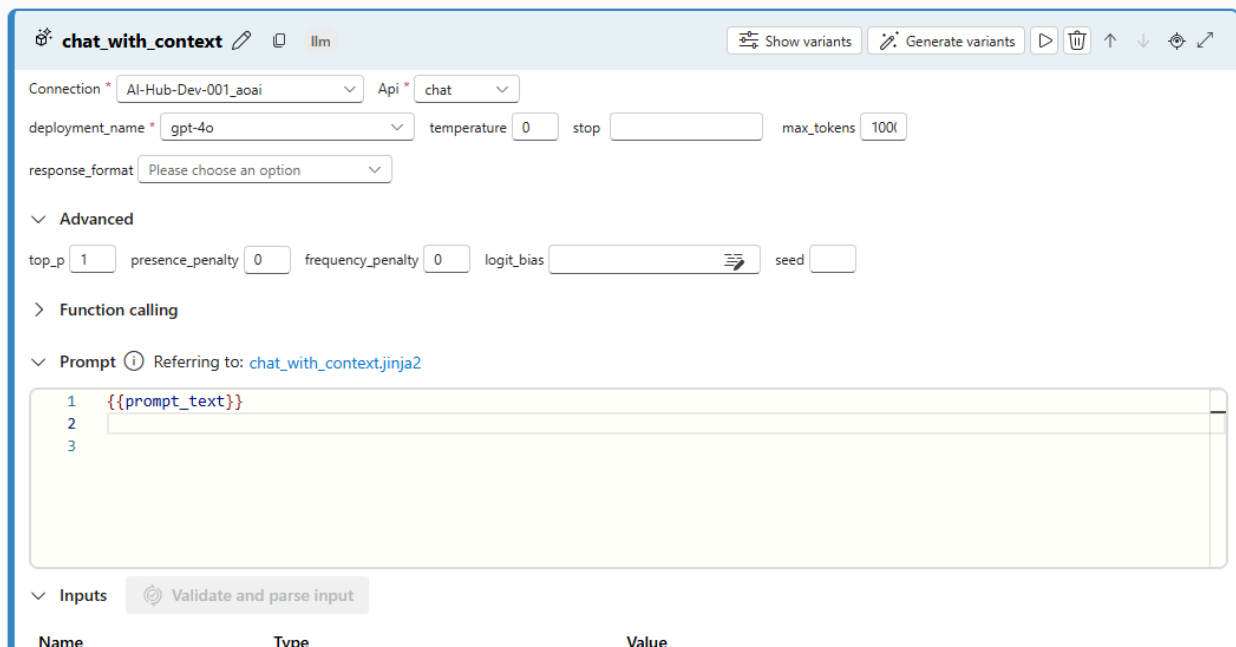


Select query type and select hybrid (vector + keyword)

Click save



In chat with context step set the connection and deployment name (model)



In the modify\_query\_with\_history step set the connection and deployment name (model)





Deploy



Evaluate ▾



Compute session running



Chat



Chat ⓘ



give me an overview of BBH?

BBH (Brown Brothers Harriman) is a financial services firm offering a range of services across investment management, private banking, and global financial administration. The firm provides fund accounting, securities lending, foreign exchange, global custody, and mutual fund services to institutional clients, including asset managers, private banks, and hedge funds (Source: Investment\_Management\_Business\_Client\_Disclosure\_Document\_2024.pdf). BBH also supports private clients, including business owners and high-net-worth individuals, through services like trust administration, estate planning, and multi-family office advisory (Source: Investment\_Management\_Business\_Client\_Disclosure\_Document\_2024.pdf). Additionally, BBH is committed to community engagement, partnering with organizations like Futures and Options to support career development for New York City high school students (Source: FW-2023 Annual Report-ACC-page-spreads.pdf).

Total tokens for generating this: **1322** tokens, time spent: **7.33** sec



Input anything to test...



