User Guide Azure Open AI Studio

Introduction

Data: Please refer to the data folder in the GitHub repository shared with you during the event. Please refer to the Readme file in the GitHub repository for the data description and the problem statement

Overview: We'll build a chatbot powered by Azure AI services such as Azure AI Search, Azure Open AI, and Azure Open AI Studio. This chatbot will be capable of ingesting any document, responding to questions based on only the information available in the documents, and handling multi-turn conversations. Assuming this is our first AI chatbot project, we will detail the steps. We'll use Azure AI Studio to develop this solution and deploy a sample interactive web application.

If time permits, we'll also investigate how to change prompts and parameters like top_k and their impact on responses. We will also try to give you a glimpse at other developer features, such as prompt flow and evaluation flow.

Note: In this step-by-step guide, we have used the resource configuration in table-1. You will receive your temporary values at the venue for this hackathon including a username.
Please follow the naming convention in Table 2 to create the names of the remaining Azure resources.

S No	Configuration	Name used in the documentation	Description
1	User name	Thailand01	Azure User name provided
2	Azure subscription	Tiger analytics-01	Name of Azure Subscription
3	Location	Australia East	Location to create Azure Resources
4	Azure OpenAI	openai-tigeranalytics-01	Name of Azure OpenAI service

Table - 1

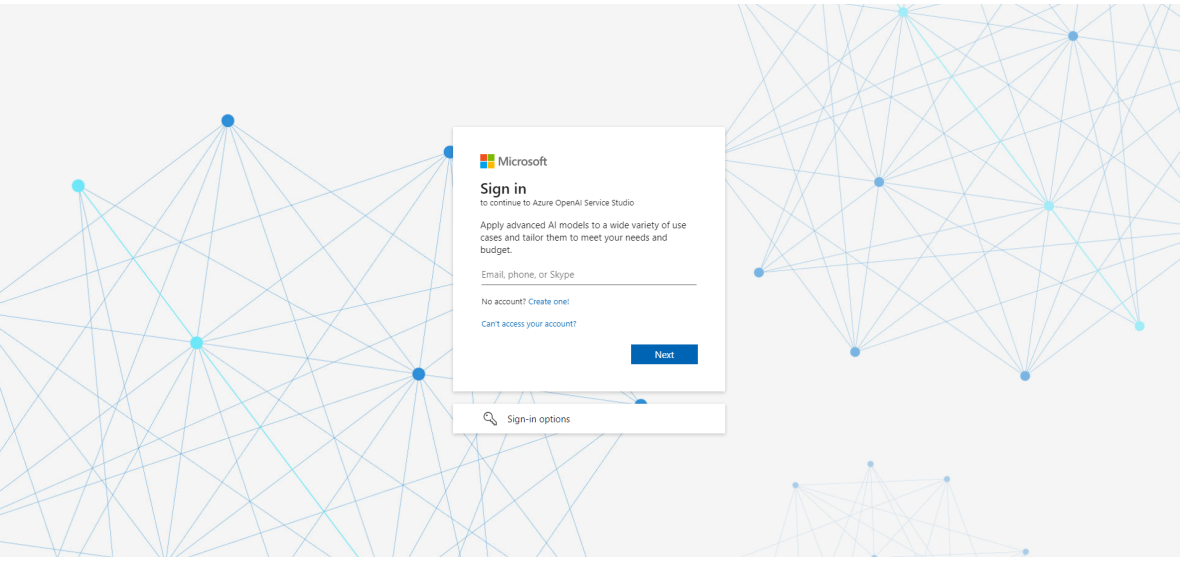
S No	Configuration	Naming Convention	Name used in documentation
1	Resource group name	rg-<User name>ai	rg-Thailand01ai
2	Azure AI search	<User name>aisearch	Thailand01aisearch
3	Connection storage name	<User name>docs	Thailand01docs

4	Index name	ai-build-<User name>-index-<version>	ai-build-Thailand01-index-v8
5	Web app name	<User name>-web app	Thailand01-web app

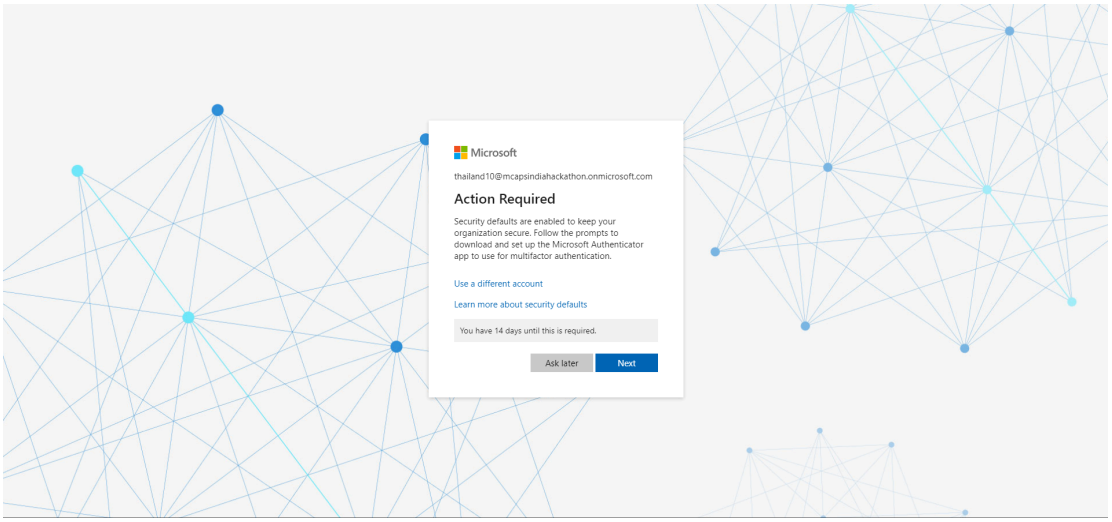
Table 2

1. Log in to the Azure Open AI Studio

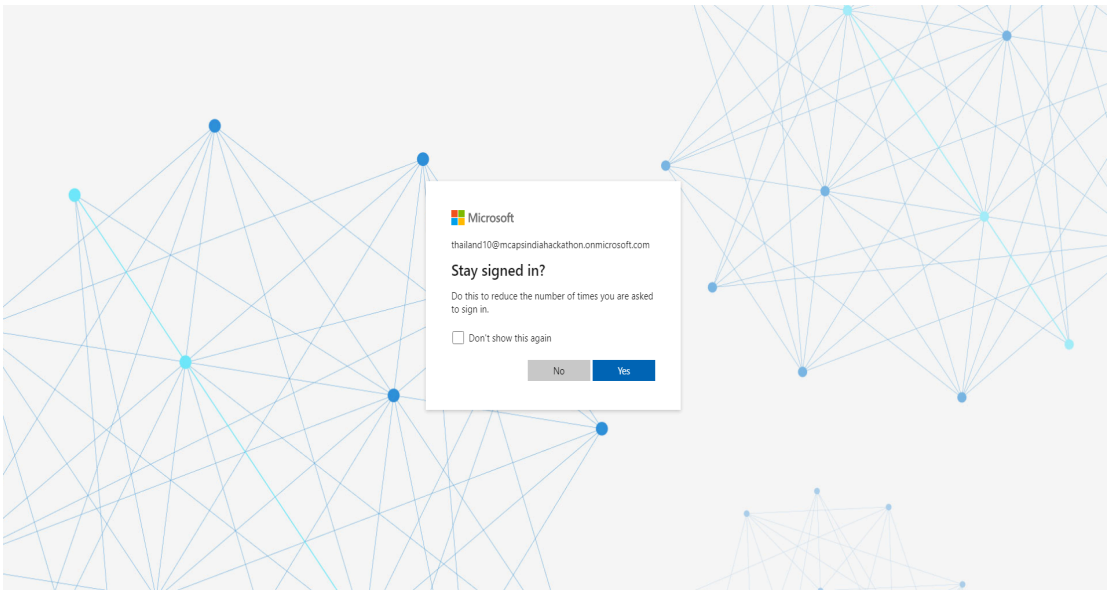
1. Please go to AOAI Studio from your browser (please use Incognito/ private mode to avoid clashes with your existing work profile)
2. Click on the Sign-in Button and enter the credentials shared with you at the registration desk. (Please keep your slip with you till the end of the event)



3. After successful login, **if you are prompted to change the password**, please do so.
4. Setup MFA/authenticator:Skip this step by clicking **“Ask Later.”** (If **authenticator is a mandatory** requirement, please choose “other method” for login and provide your phone number, you can sign-in through OTP**)**

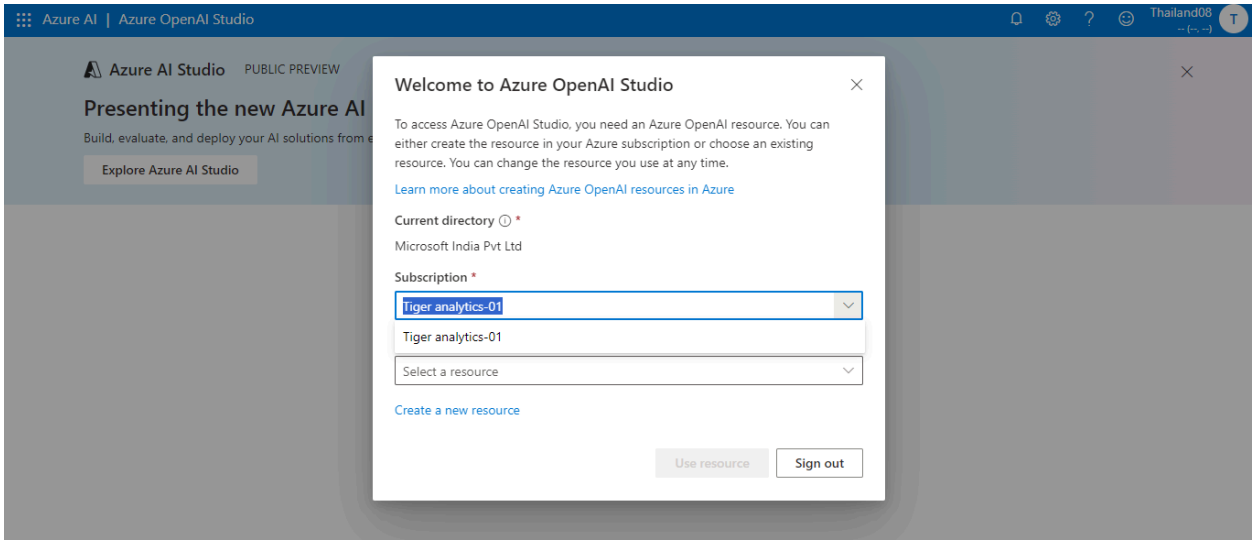


5. Click on **Yes** in the next step.

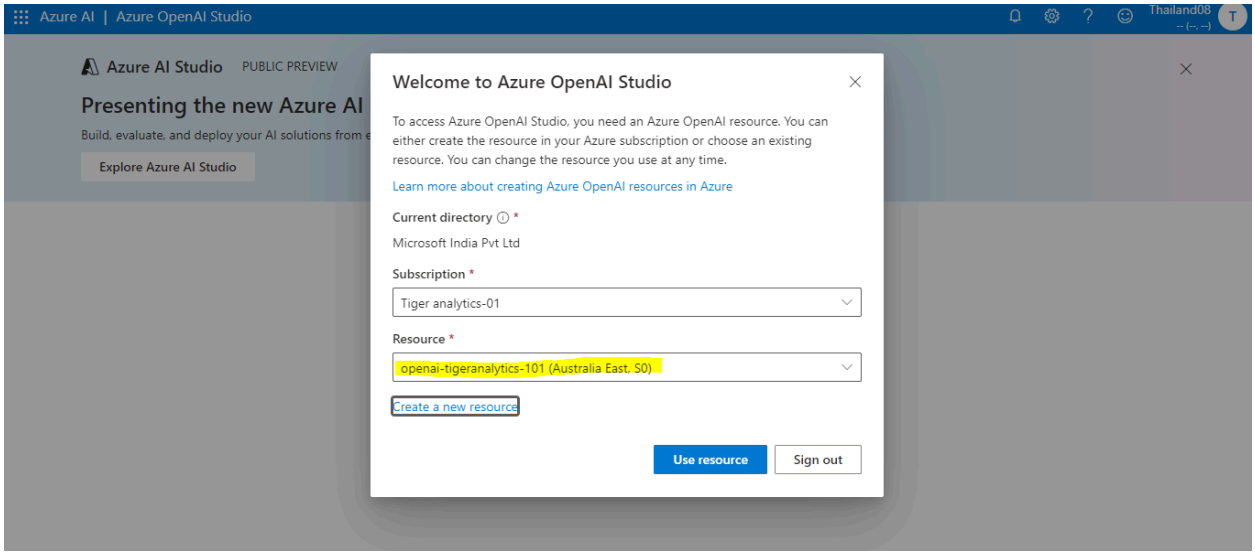


2. Welcome Page

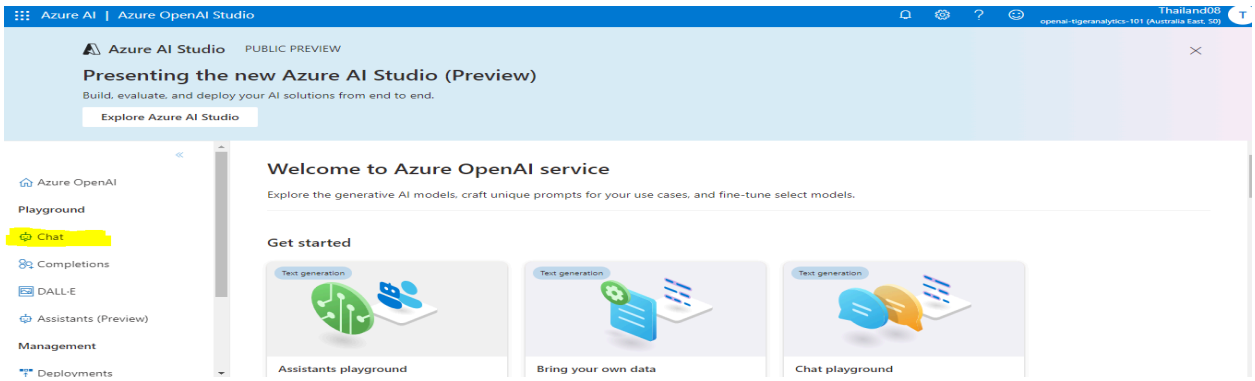
1. Under **Subscription**, select the default option. It will look like **Tiger analytics-<number>**, e.g. Tiger analytics-01



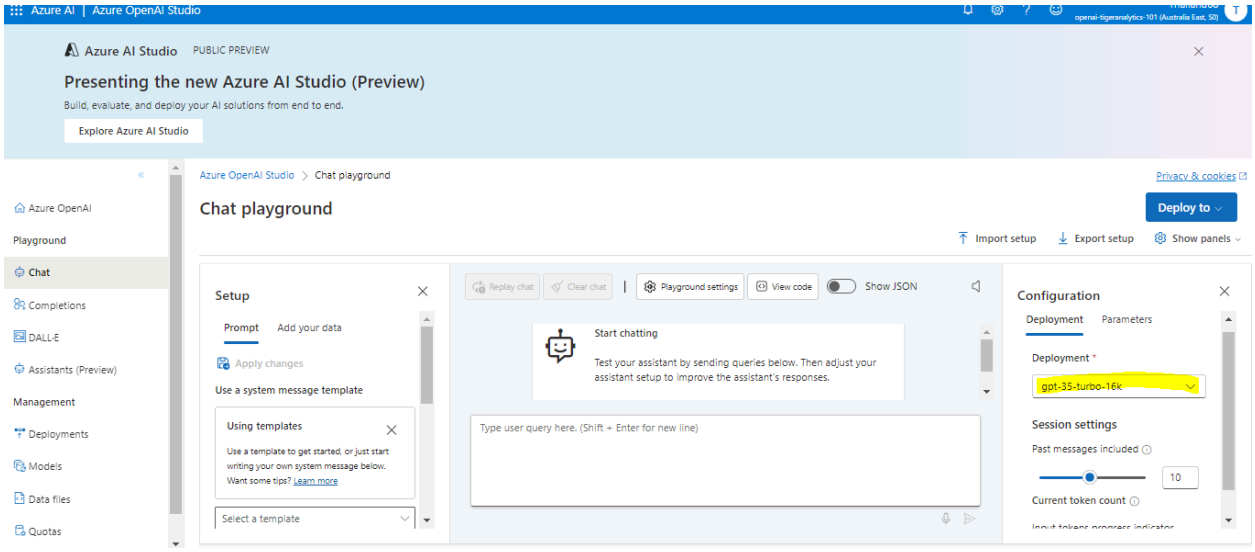
2. Under **Resource**, select the default option. It will look like **openai - Tiger analytics-<number>**, e.g. **open ai Tiger analytics-101**



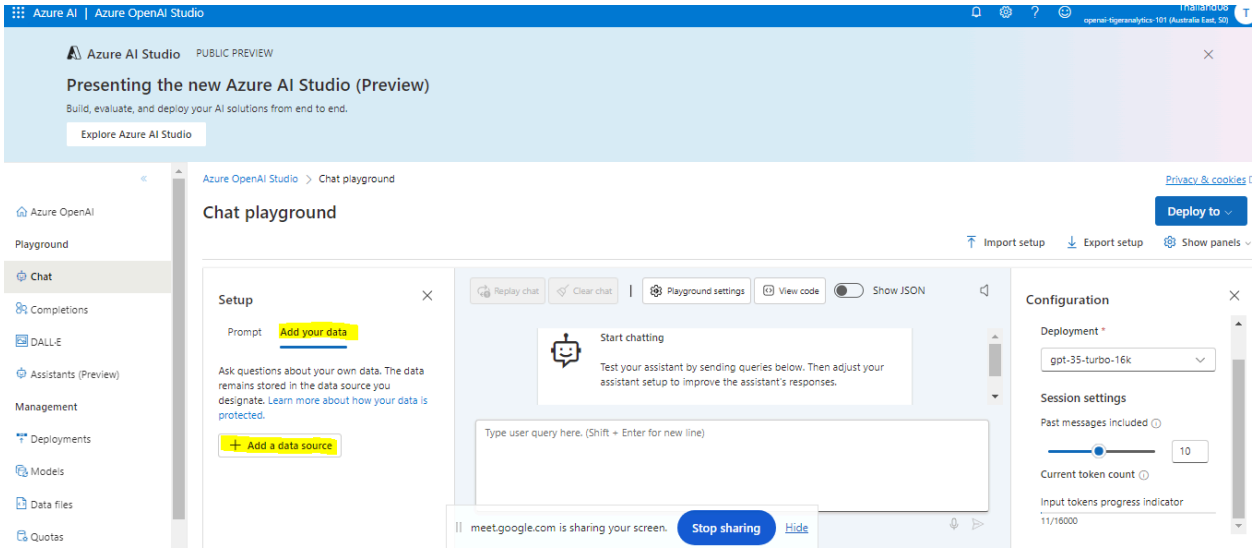
3. Select “chat” option in the AOAI studio home page



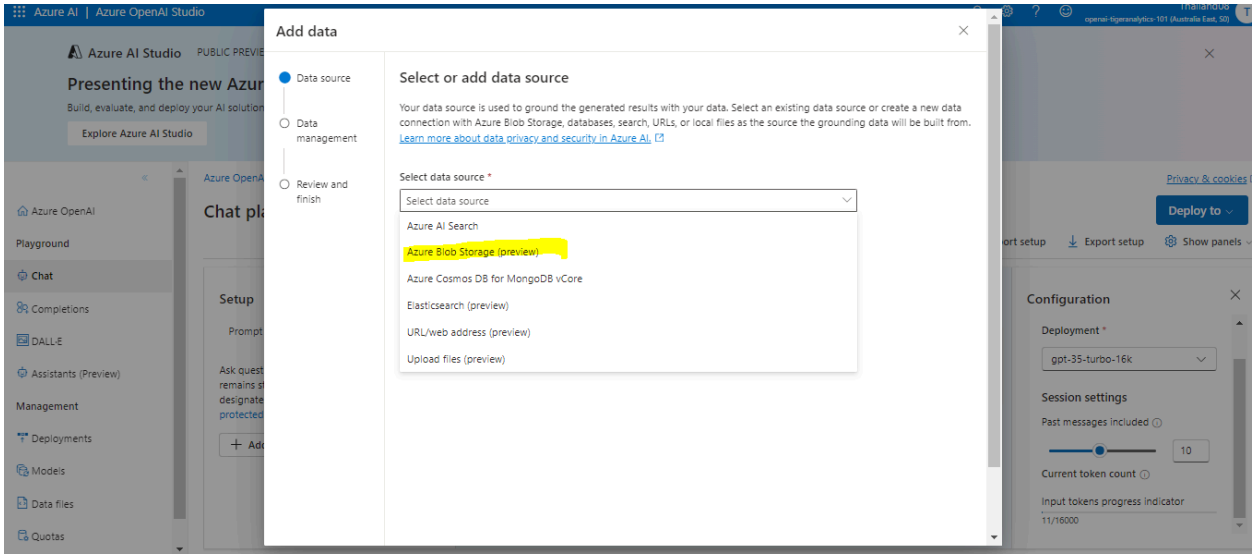
4. You will land up in the Playground page. Let's confirm if we are connected to the Azure OpenAI Service. On the right-hand side panel of the project playground, select the model from the dropdown **gpt-35-turbo16k,** and change the **past messages to include 10**.



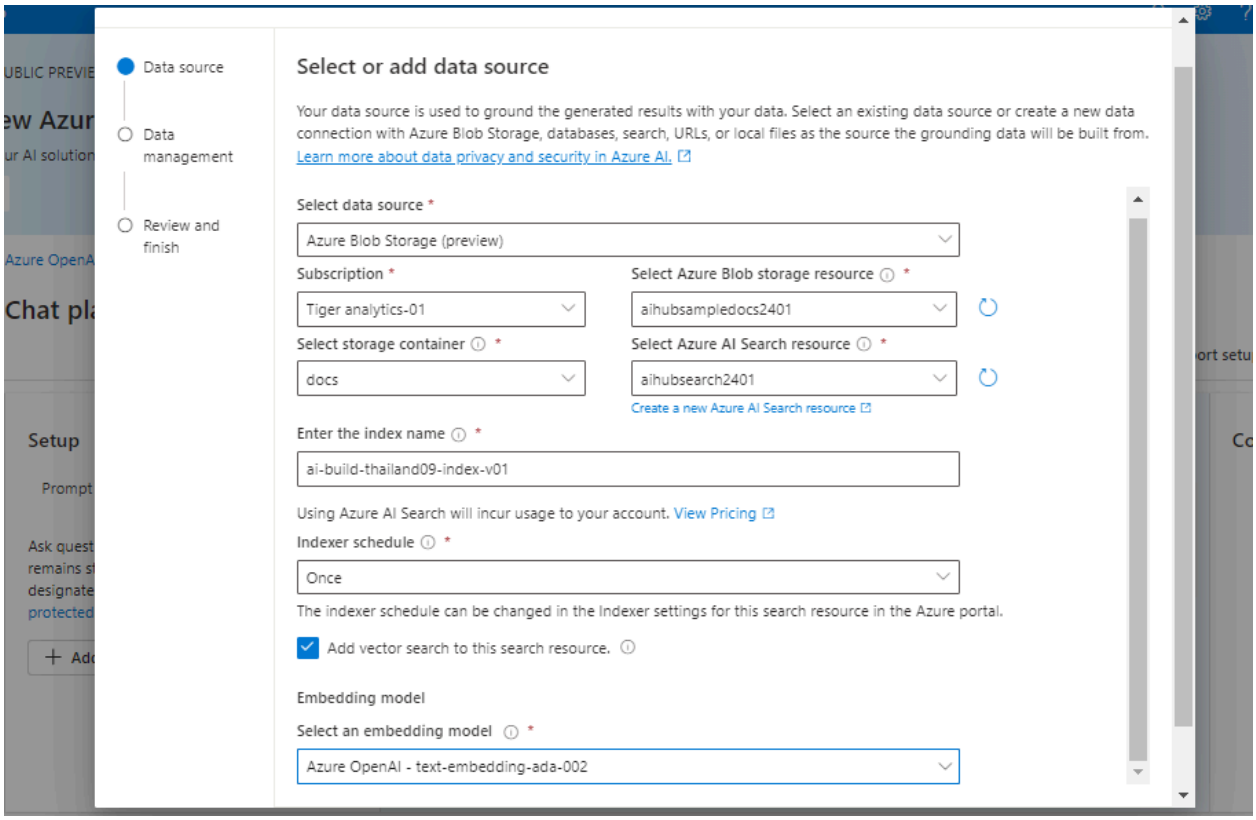
5. Now, let's add our data. On the left-hand side of the playground, go to **Add your Data** and click on **Add your Data.**



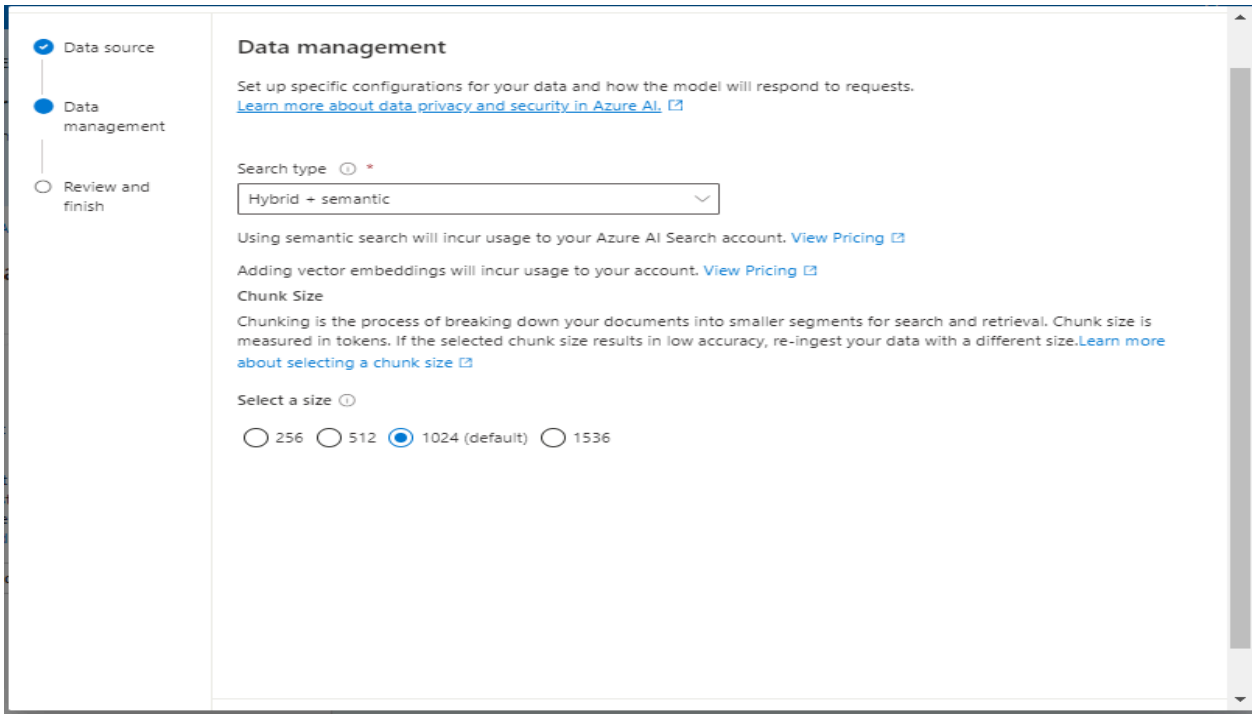
6. Select your "data source" and choose "**Azure Blob storage**"



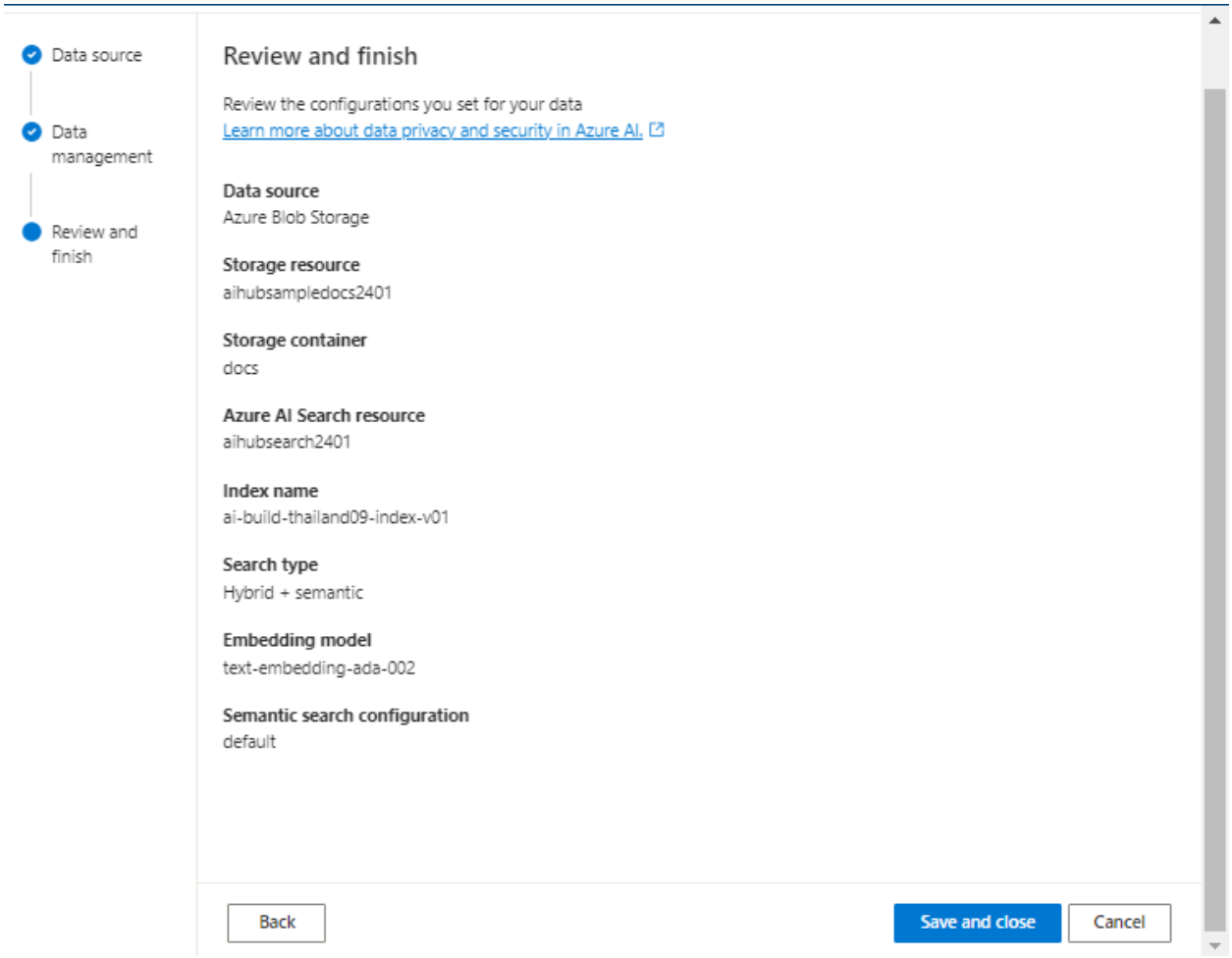
7. Fill in the “data source details”. The **subscription ID** will be populated by default. Select the **storage account,** which will be in the format **aihubsampledocs<numericid>** and **blob container** as **docs.** For Azure AI search resource, choose the default option**.** Indexer schedule should be selected as **“once”.** Select **“add vector search tick mark”.** Choose the default **embedding model** from the drop down**.** In the **Index Settings,** use the index name as follows: “**ai-build-<userid>-index-<version>.** Example **ai-build-Thailand01-index-v8.**



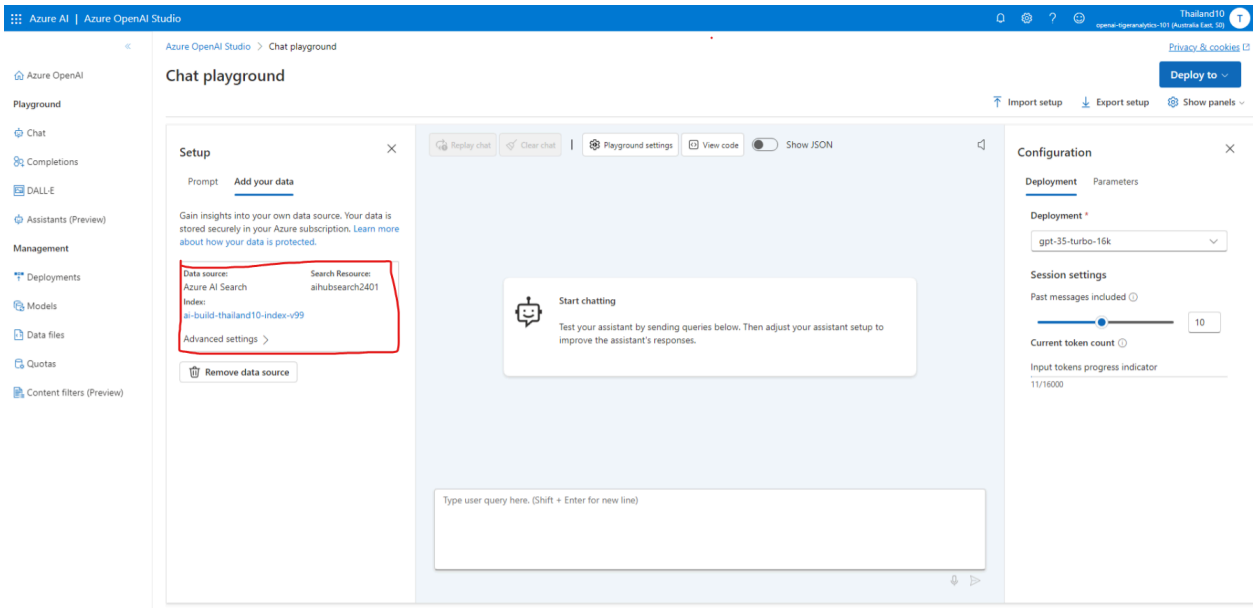
8. **Update the Data Management screen as below.** Ensure the search type is selected as “Hybrid + Semantic”. Keep the other settings as default.



9. **Review & Finish** - Validate your inputs and save

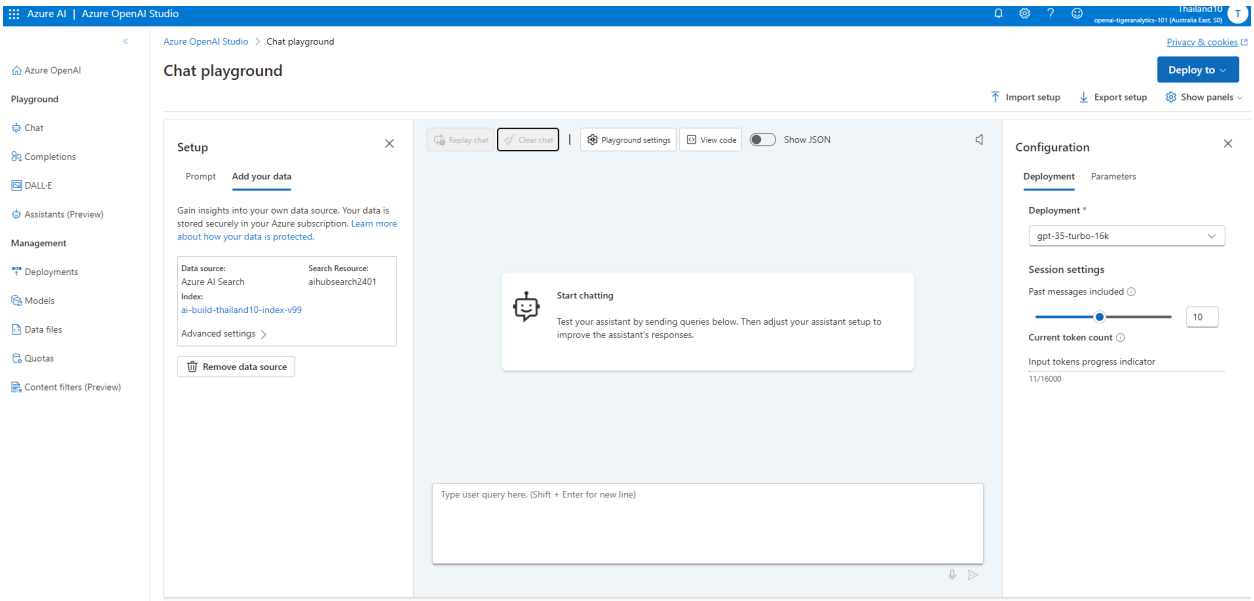


10. You will be redirected to the playground page with the data index created.



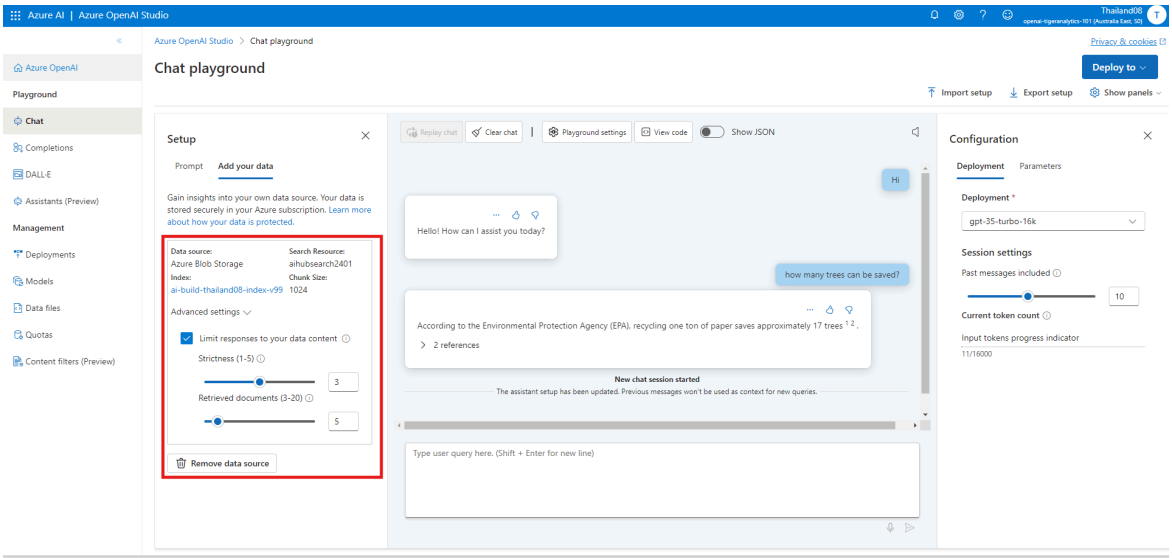
3. Start asking questions

1. Check on the responses from the data and ask questions. Some sample questions on the data can be found here (Please clone the repository or download the zip and navigate to the sample QnA spreadsheet in the **Data** folder to view sample questions).
2. Please click the **Clear Chat** button in the chat box before asking an unrelated follow-up question.

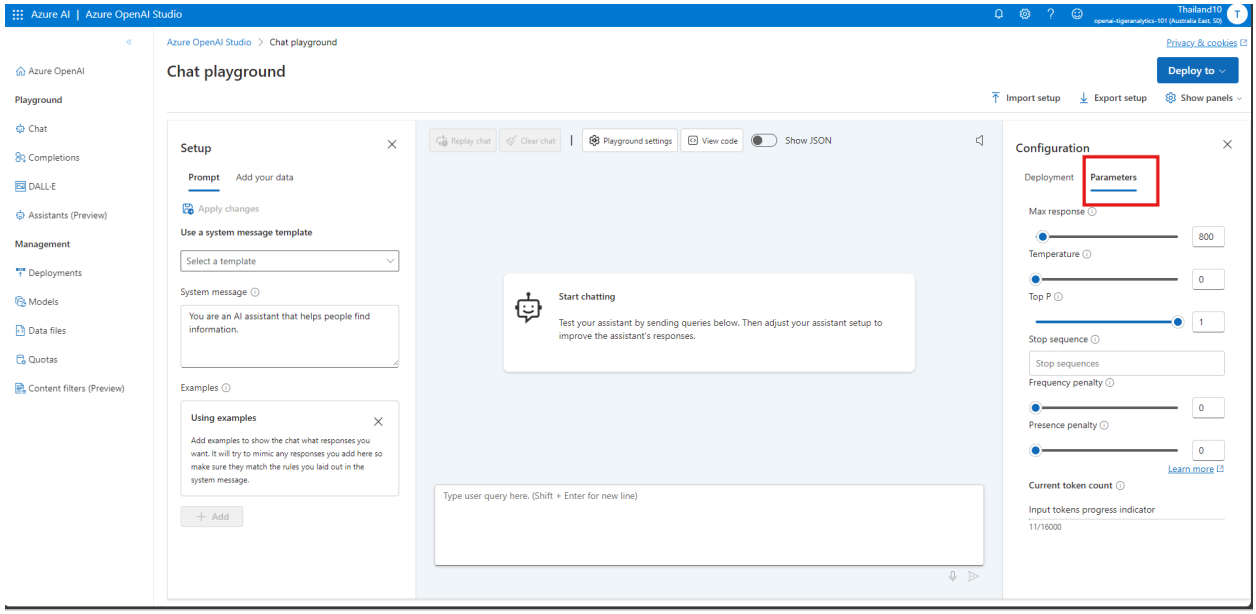


4. Tweak the parameters

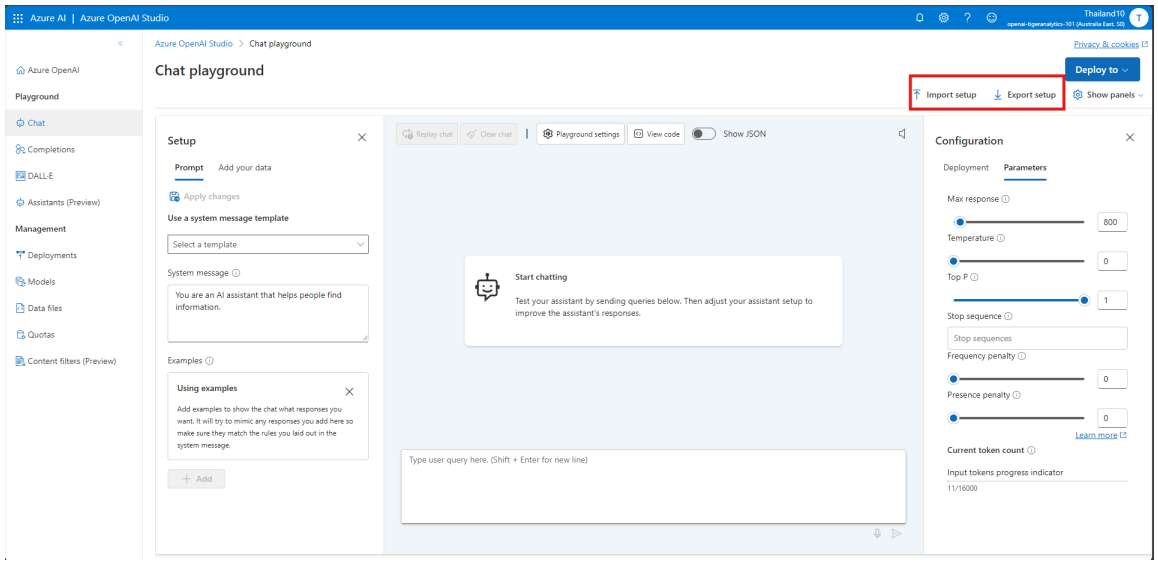
1. Suppose you want the responses to come not only from the source data alone but also from the external knowledge of the Azure OpenAI models. In that case, you can click on **Advanced Settings** under **Add your data** tab and uncheck **Limit responses to your data content.**
2. **Strictness** helps you set the threshold for relevant documents, and **Retrieved documents** help you set the number of chunks/documents that will be retrieved to answer the questions.



3. Please check this guide on [how to generate text with Azure OpenAI Service - Azure OpenAI | Microsoft Learn]
4. Tweak the model parameters from the **Parameters** tab. You can find the definition of the parameters [here].
5. Make changes to the prompts to change the response to how you like (for example, we can add to give the response in bullet points or a happy tone). Remember to click on **Apply Changes** after changing the prompt. You can click the **Replay chat** button to get responses to the existing questions in the updated prompt.



6. You can also **import** and **export** the chat settings (prompts, examples, parameters) to a JSON. You can try the same using this JSON. This feature helps share your workspace settings while working as a team.



7. You can also click on View Code to get executable code in the programming language of your choice with the Azure OpenAI, data, and index connections already set.