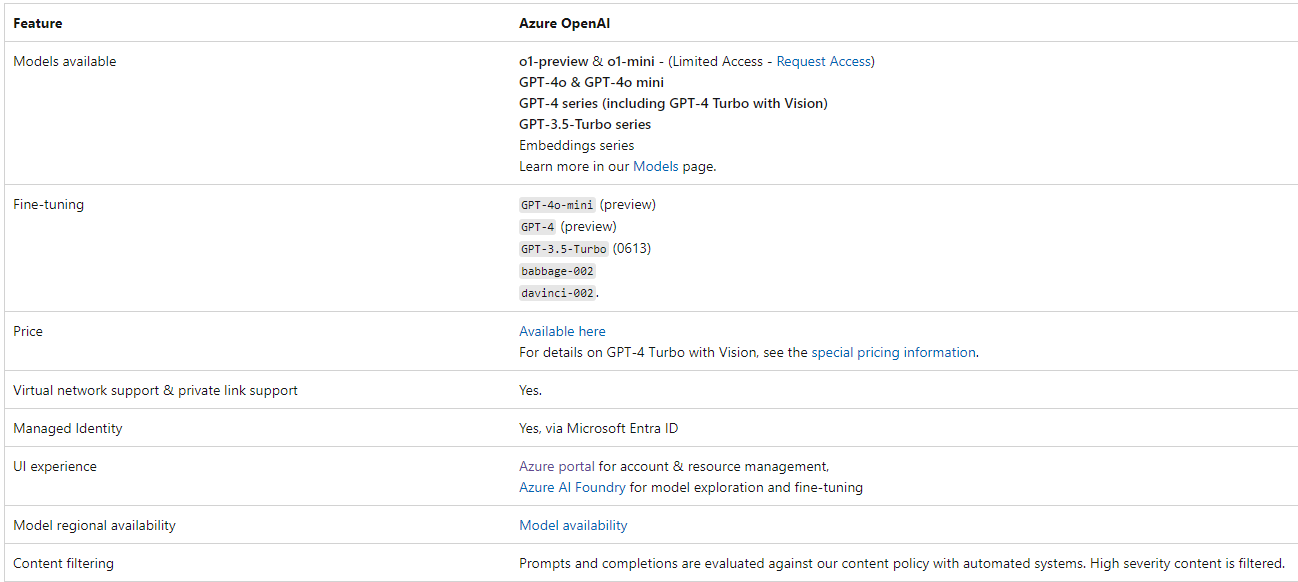
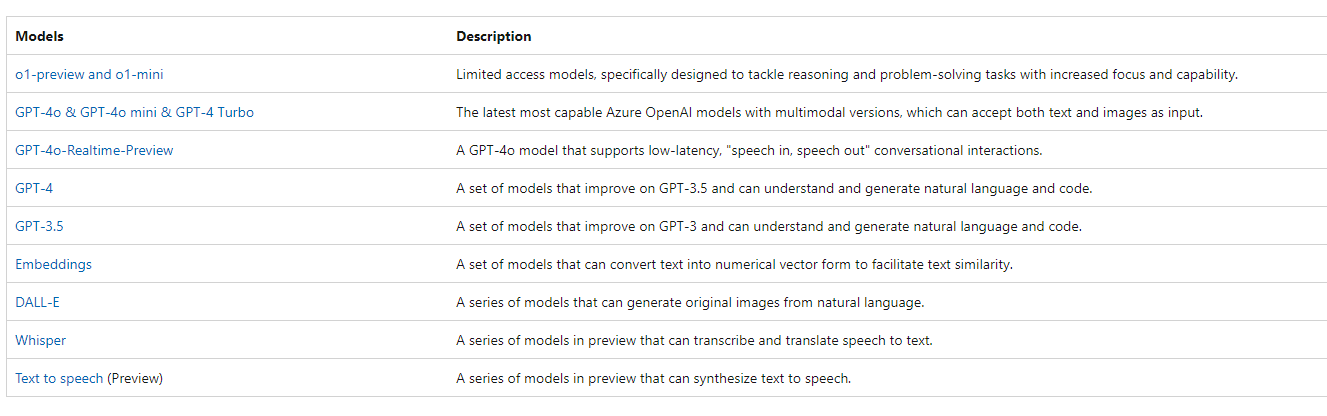
**Azure OpenAI Service**

Azure OpenAI Service provides REST API access to OpenAI's powerful language models including o1-preview, o1-mini, GPT-4o, GPT-4o mini, GPT-4 Turbo with Vision, GPT-4, GPT-3.5-Turbo, and Embeddings model series. These models can be easily adapted to your specific task including but not limited to content generation, summarization, image understanding, semantic search, and natural language to code translation. Users can access the service through REST APIs, Python SDK, or in the [Azure AI Foundry](https://ai.azure.com/).

**Features overview**





**Get started with Azure OpenAI Service**

To get started with Azure OpenAI Service, you need to create an Azure OpenAI Service resource in your Azure subscription.

Start with the [Create and deploy an Azure OpenAI Service resource](https://learn.microsoft.com/en-us/azure/ai-services/openai/how-to/create-resource) guide.

1. You can create a resource via Azure portal, Azure CLI, or Azure PowerShell.
2. When you have an Azure OpenAI Service resource, you can deploy a model such as GPT-4o.
3. When you have a deployed model, you can:
   * Try out the Azure AI Foundry portal playgrounds to explore the capabilities of the models.
   * You can also just start making API calls to the service using the REST API or SDKs.

For example, you can try [real-time audio](https://learn.microsoft.com/en-us/azure/ai-services/openai/realtime-audio-quickstart) and [assistants](https://learn.microsoft.com/en-us/azure/ai-services/openai/assistants-quickstart) in the playgrounds or via code.

**Comparing Azure OpenAI and OpenAI**

Azure OpenAI Service gives customers advanced language AI with OpenAI GPT-4, GPT-3, Codex, DALL-E, Whisper, and text to speech models with the security and enterprise promise of Azure. Azure OpenAI co-develops the APIs with OpenAI, ensuring compatibility and a smooth transition from one to the other.

With Azure OpenAI, customers get the security capabilities of Microsoft Azure while running the same models as OpenAI. Azure OpenAI offers private networking, regional availability, and responsible AI content filtering.

**Key concepts**

**Prompts & completions**

The completions endpoint is the core component of the API service. This API provides access to the model's text-in, text-out interface. Users simply need to provide an input **prompt** containing the English text command, and the model generates a text **completion**.

Here's an example of a simple prompt and completion:

**Prompt**: """ count to 5 in a for loop """

**Completion**: for i in range(1, 6): print(i)

**Tokens**

**Text tokens**

Azure OpenAI processes text by breaking it down into tokens. Tokens can be words or just chunks of characters. For example, the word “hamburger” gets broken up into the tokens “ham”, “bur” and “ger”, while a short and common word like “pear” is a single token. Many tokens start with a whitespace, for example “ hello” and “ bye”.

The total number of tokens processed in a given request depends on the length of your input, output, and request parameters. The quantity of tokens being processed will also affect your response latency and throughput for the models.

**Image tokens**

Azure OpenAI's image processing capabilities with GPT-4o, GPT-4o mini, and GPT-4 Turbo with Vision models uses image tokenization to determine the total number of tokens consumed by image inputs. The number of tokens consumed is calculated based on two main factors: the level of image detail (low or high) and the image’s dimensions. Here's how token costs are calculated:

* **Low resolution mode**
  + Low detail allows the API to return faster responses for scenarios that don't require high image resolution analysis. The tokens consumed for low detail images are:
    - **GPT-4o and GPT-4 Turbo with Vision**: Flat rate of **85 tokens per image**, regardless of size.
    - **GPT-4o mini**: Flat rate of **2833 tokens per image**, regardless of size.
  + **Example: 4096 x 8192 image (low detail)**: The cost is a fixed 85 tokens with GPT-4o, because it's a low detail image, and the size doesn't affect the cost in this mode.
* **High resolution mode**
  + High detail allows the API to analyze images in more detail. Image tokens are calculated based on the image's dimensions. The calculation involves the following steps:
    - * **Image resizing**: The image is resized to fit within a 2048 x 2048 pixel square. If the shortest side is larger than 768 pixels, the image is further resized so that the shortest side is 768 pixels long. The aspect ratio is preserved during resizing.
      * **Tile calculation**: Once resized, the image is divided into 512 x 512 pixel tiles. Any partial tiles are rounded up to a full tile. The number of tiles determines the total token cost.
      * **Token calculation**:
        + **GPT-4o and GPT-4 Turbo with Vision**: Each 512 x 512 pixel tile costs **170 tokens**. An extra **85 base tokens** are added to the total.
        + **GPT-4o mini**: Each 512 x 512 pixel tile costs **5667 tokens**. An extra **2833 base tokens** are added to the total.
  + **Example: 2048 x 4096 image (high detail)**:
    - * The image is initially resized to 1024 x 2048 pixels to fit within the 2048 x 2048 pixel square.
      * The image is further resized to 768 x 1536 pixels to ensure the shortest side is a maximum of 768 pixels long.
      * The image is divided into 2 x 3 tiles, each 512 x 512 pixels.
      * **Final calculation**:
        + For GPT-4o and GPT-4 Turbo with Vision, the total token cost is 6 tiles x 170 tokens per tile + 85 base tokens = 1105 tokens.
        + For GPT-4o mini, the total token cost is 6 tiles x 5667 tokens per tile + 2833 base tokens = 36835 tokens.

**Models**

The service provides users access to several different models. Each model provides a different capability and price point.

The DALL-E models (some in preview; see [models](https://learn.microsoft.com/en-us/azure/ai-services/openai/concepts/models#dall-e)) generate images from text prompts that the user provides.

The Whisper models can be used to transcribe and translate speech to text.

The text to speech models, currently in preview, can be used to synthesize text to speech.

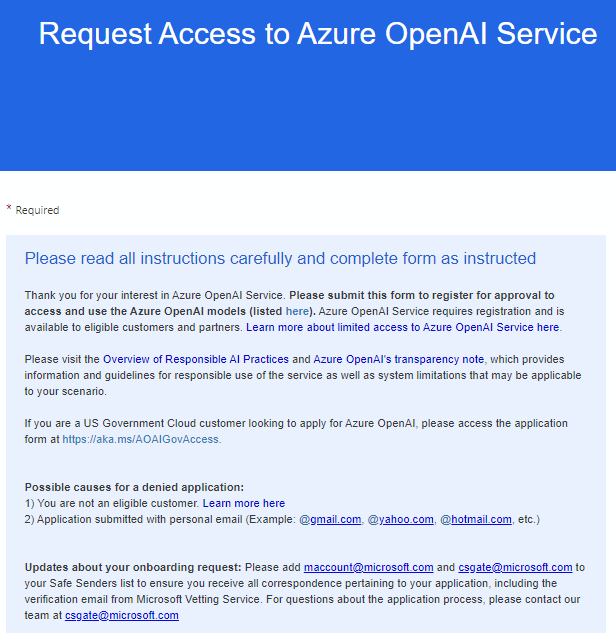
**Azure AI Foundry**

Azure AI Foundry is Microsoft’s all-in-one platform designed to simplify the development, deployment, and management of AI applications. It consists of two main components:

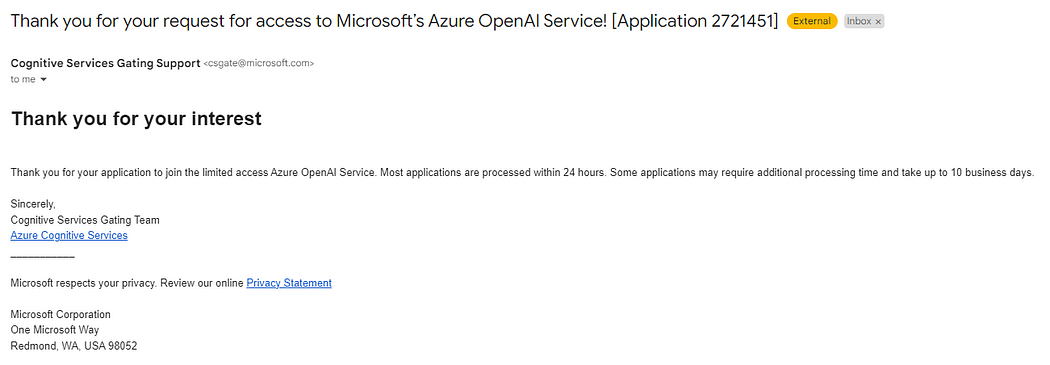
* A GUI portal, formerly known as Azure AI Studio
* An Azure AI Foundry SDK which enables developers to work with robust pre-built components and leverage pre-built app templates (we will see it more in detail later on in the article).

**Built a customer-facing Chatbot using Azure Open AI Studio**

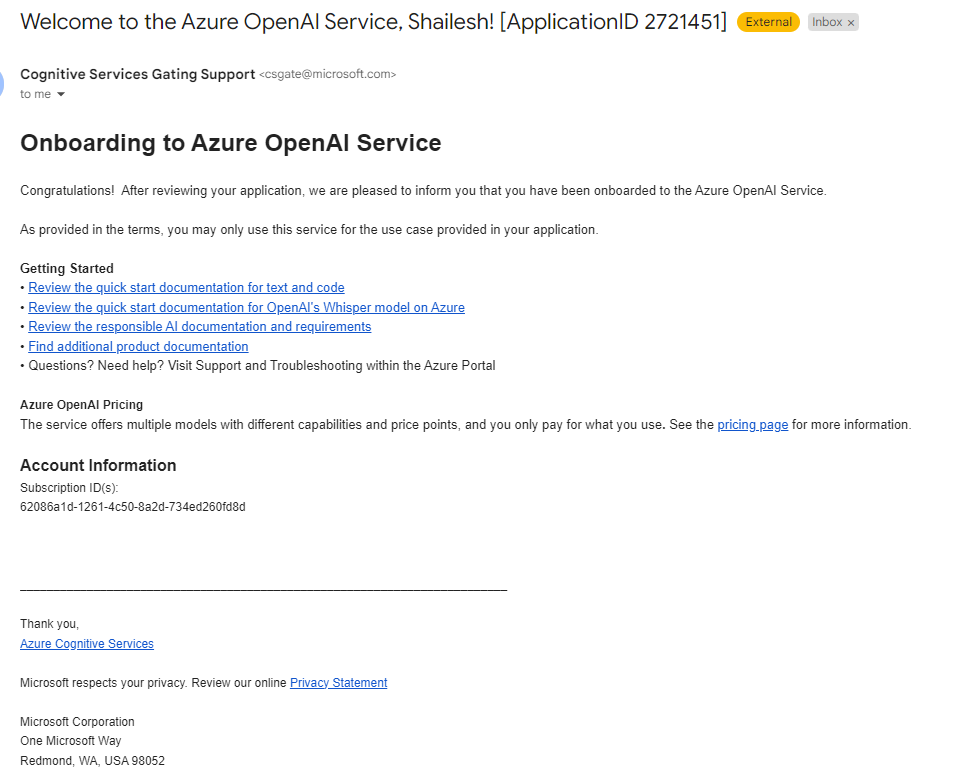
**Pre-requisites : Request Access to Azure OpenAI Service**



**Email sent to Azure Cognitive Services Team**



**Email received from Azure Cognitive Services Team**



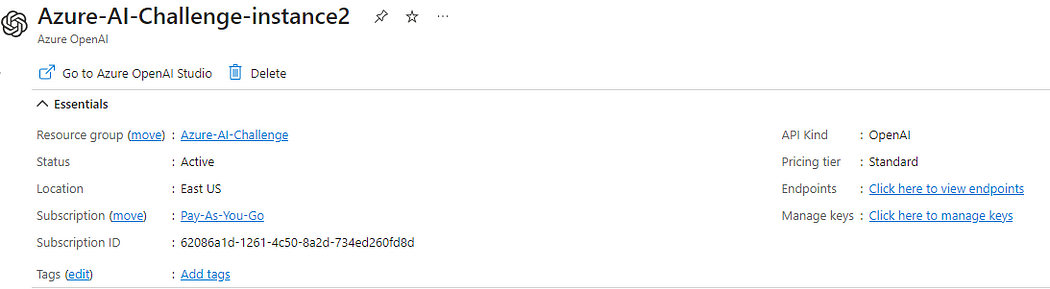
Let’s proceed...!!

**Step 1 :** Setup Azure Open AI Resource

Go to Azure AI Services → Azure Open AI

Create an **Azure OpenAI** resource with the following settings

**Subscription**: Select an Azure subscription that has been approved for access to the Azure OpenAI service  
**Resource group**: Choose or create a resource group  
**Region**: Make a random choice from any of the following regions\*  
Australia East  
Canada East  
East US  
East US 2  
France Central  
Japan East  
North Central US  
Sweden Central  
Switzerland North  
UK South  
**Name**: A unique name of your choice  
**Pricing tier**: Standard S0

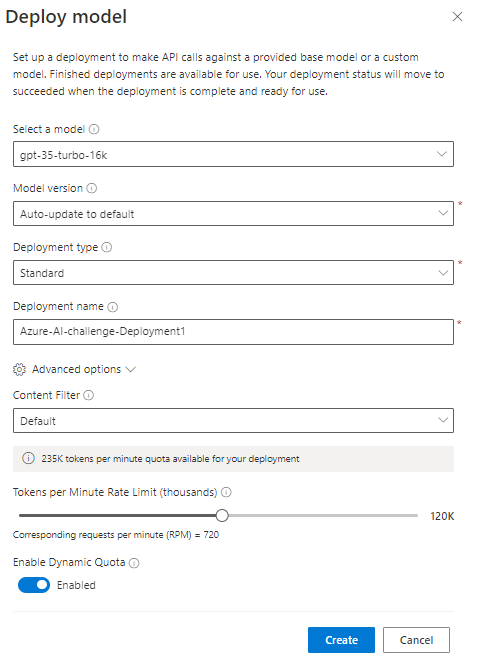


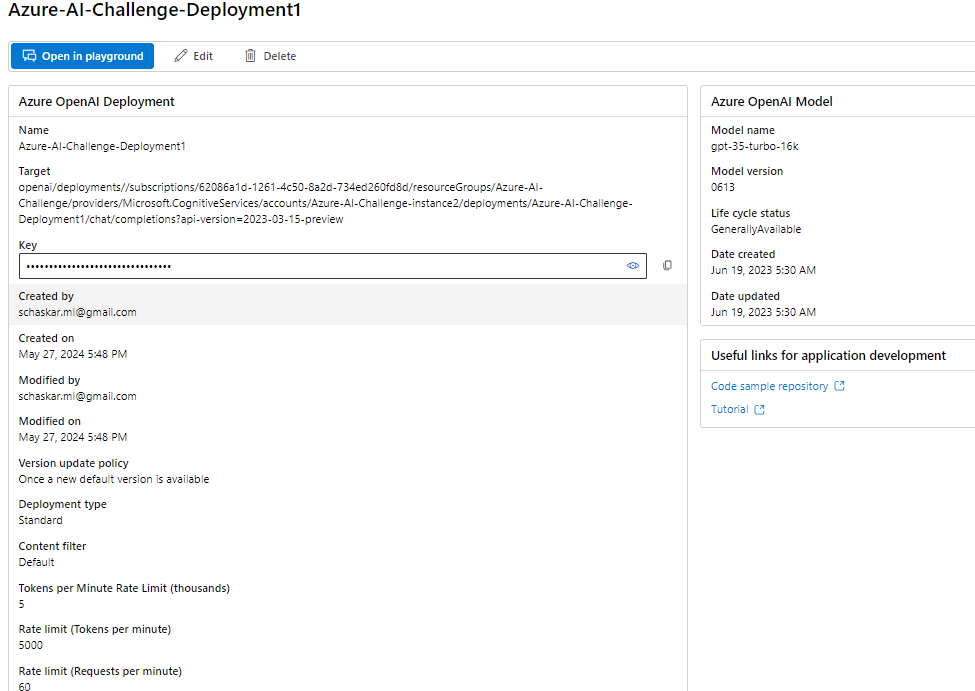
**Step 2 :** Deploy a Model

Go to Azure AI Services → Azure OpenAI Studio -> Deployments

**Model**: gpt-35-turbo-16k  
**Model version**: Auto-update to default  
**Deployment name**: A unique name of your choice

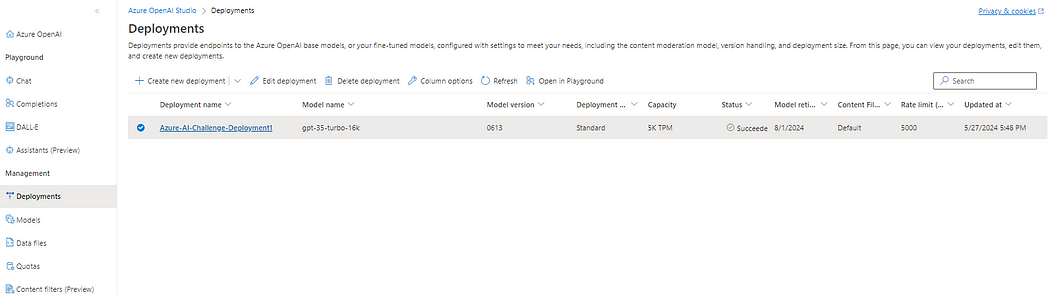
Advanced options  
**Content filter**: Default  
**Deployment type**: Standard  
**Tokens per minute rate limit**: 5K\*  
**Enable dynamic quota**: Enabled

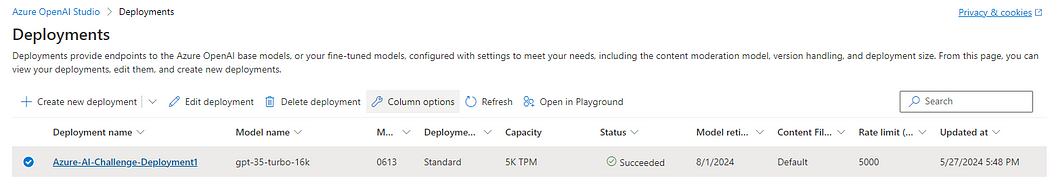




**Step 3 :**Verify Deployment is successfully

Azure AI Services → Azure OpenAI Studio -> Deployments





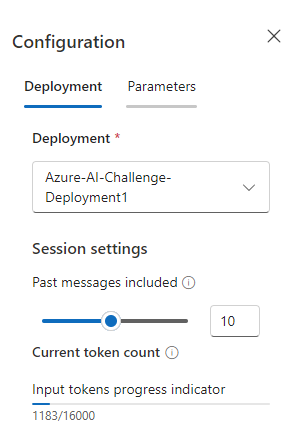
A screenshot of a computer

Description automatically generated

**Step 4 :**Verify configuration on Chat Playground

Go to → Azure OpenAI Studio -> Chat





Step 5 — Let’s test the model in Chat Playground

