

# Build a Chatbot with Azure OpenAI + PVA in less than 15 mins

Microsoft AI Rangers

## What is Azure OpenAI?

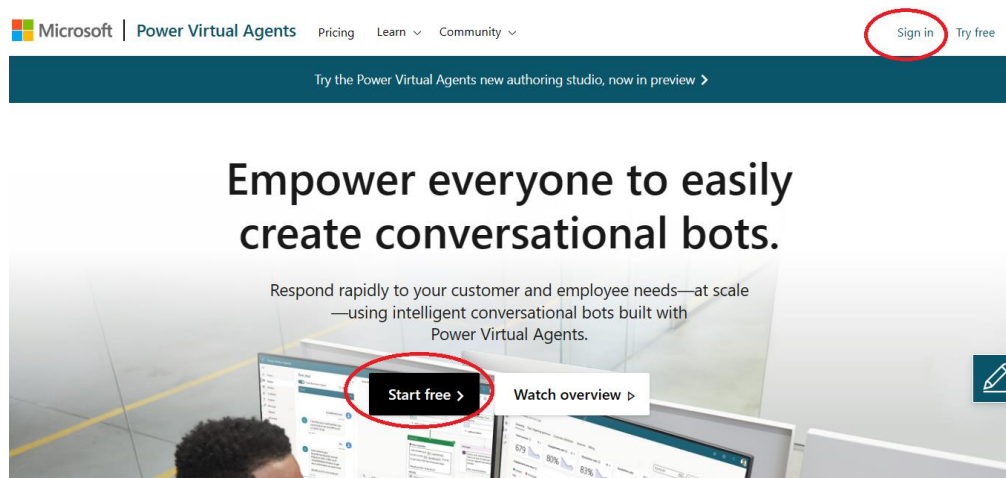
The Azure OpenAI service provides REST API access to OpenAI's powerful language models including the GPT-3, Codex and Embeddings model series. These models can be easily adapted to your specific task including but not limited to content generation, summarization, semantic search, and natural language to code translation. Users can access the service through REST APIs, Python SDK, or our web-based interface in the Azure OpenAI Studio. You can learn more about it here: [What is Azure OpenAI? - Azure Cognitive Services | Microsoft Learn](#).

## What is PVA (Power Virtual Agents)?

Microsoft Power Virtual Agents empowers everyone to create intelligent conversational bots - from citizen developers, business users to professional developers alike – to seamlessly build secure and scalable virtual agents in an integrated building platform. You can learn more about it here: [Overview - Power Virtual Agents | Microsoft Learn](#)

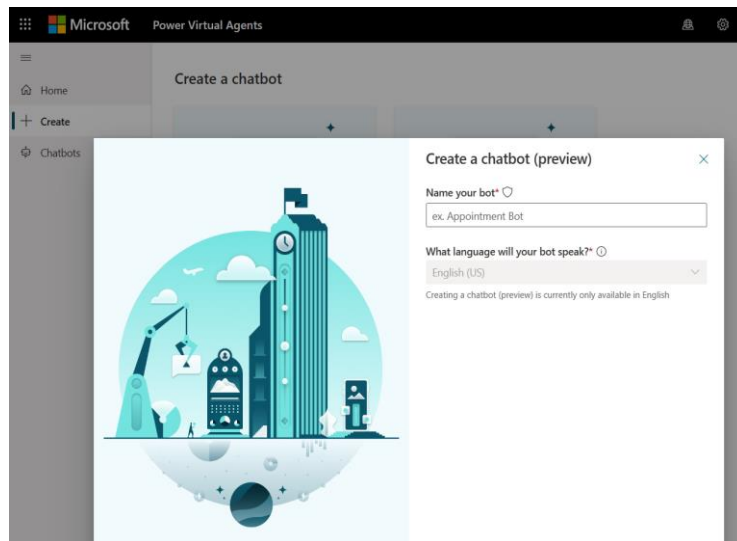
## How to get started:

1. You will first need to secure Azure OpenAI access. To do this, please apply here with your use case:: [Request Access to Azure OpenAI Service \(microsoft.com\)](#).
  - Once you have Azure OpenAI access, create a new resource in your Azure Subscription following this documentation : [How-to - Create a resource and deploy a model using Azure OpenAI - Azure OpenAI | Microsoft Learn](#)
  - Once the Azure OpenAI resource is created, you will need to create a new deployment, which you will use later. Follow this guide for more info: [Deploy a model](#)
  - You can choose any of the text models, for this demo we used the advanced "text-davinci-003"
2. As a next step, you will need to create a Power Virtual Agent account (or use one you may already have). For this, select the "start free" to create a new account for Power Virtual Agents here: [Intelligent Virtual Agents and Bots | Microsoft Power Virtual Agents](#)

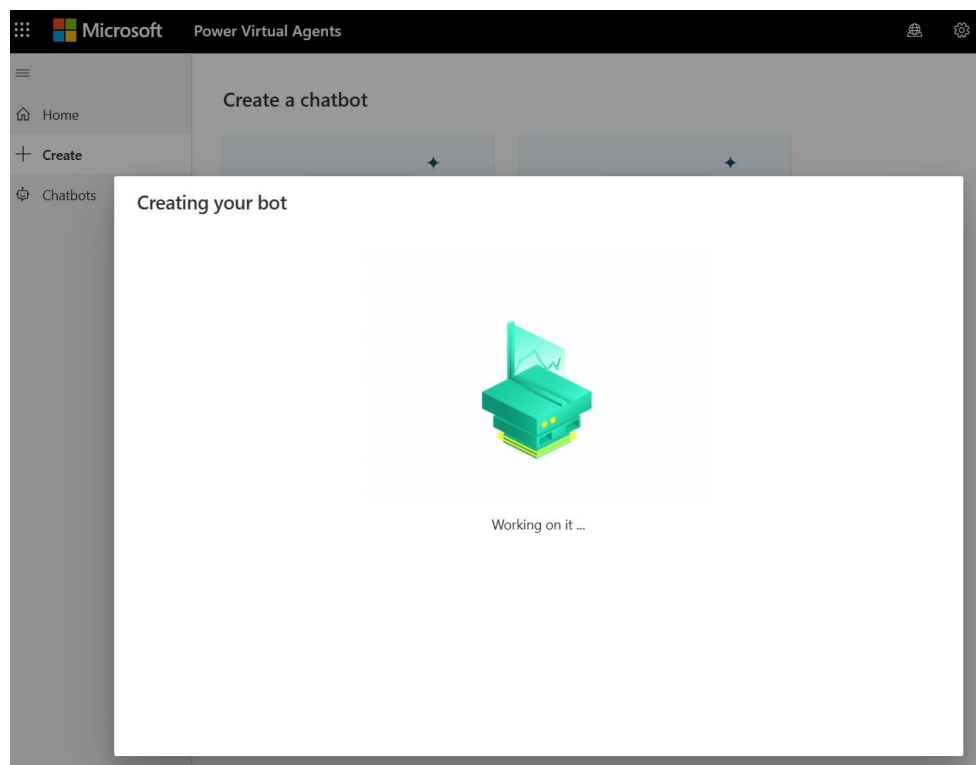


## Now that you are all set-up, let's get started!

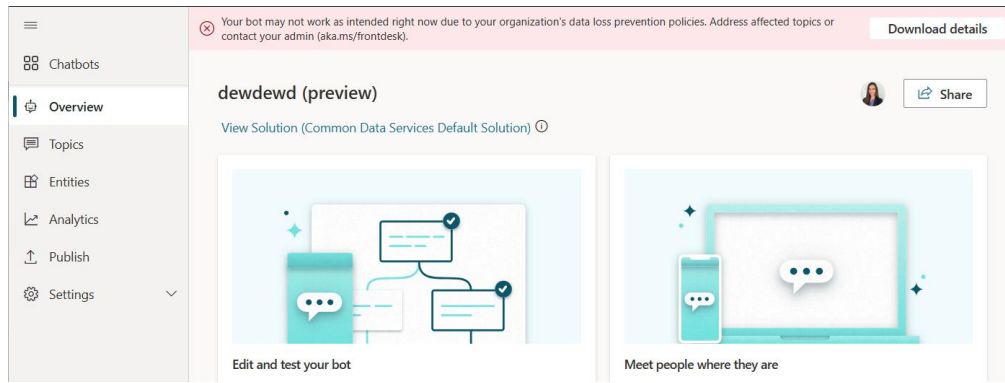
1. Sign-in the Power Virtual Agents page with your existing account.
2. On the next step, click "Create" to create a new bot. Then, "Try the unified canvas (preview)". Provide a name for your bot. E.g. "demo\_bot", then "Create".



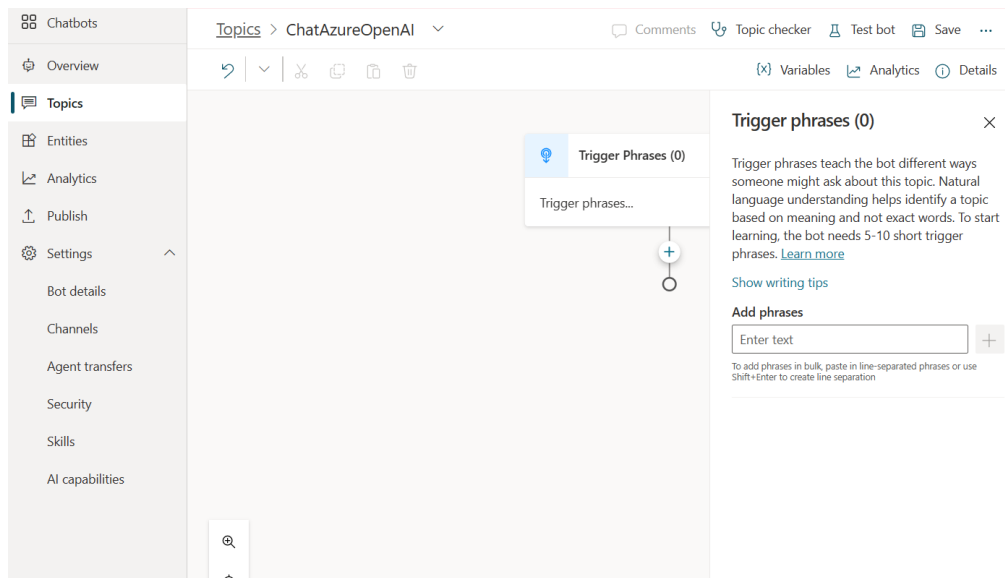
4. Your new bot will be created after a few moments:



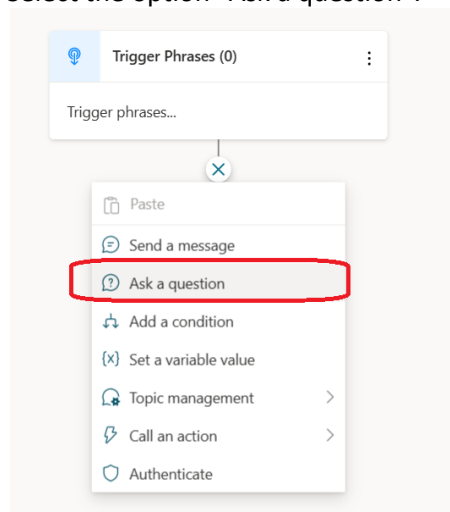
5. Once it's created, you will land on the following page. If you get the error message in red at the top of your page, use the options on the left side menu, namely Settings > Security > Authentication, and select the option to require users to sign in.



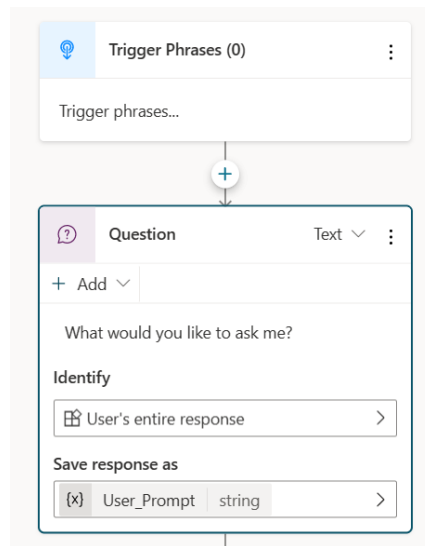
- On the Topics menu, select "+ New topic", and rename your topic on the top of the page, to something like "ChatAzureOpenAI"



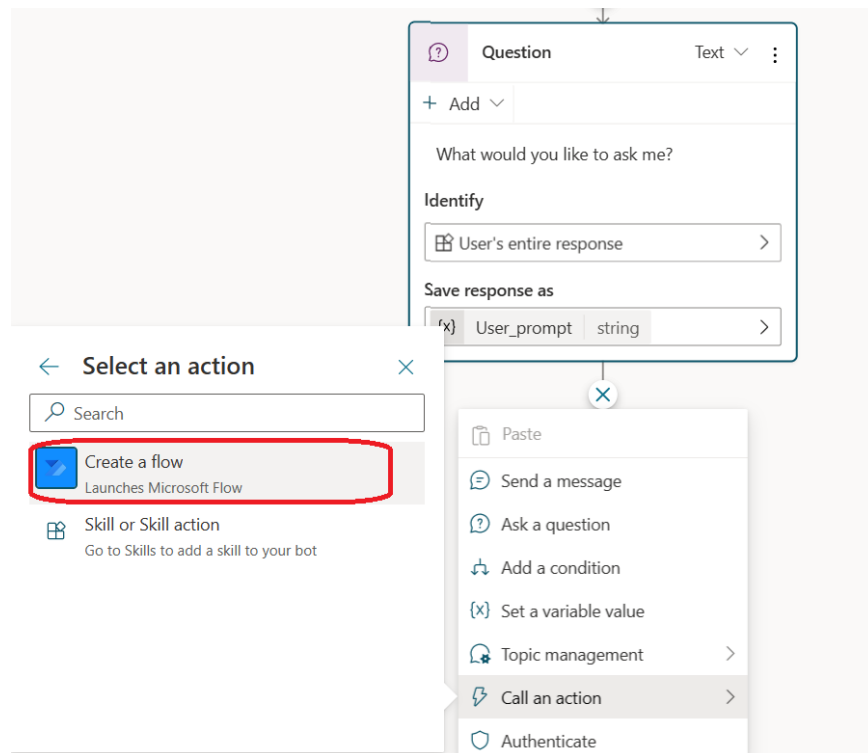
- Click on the "+" icon, to add a new node. Select the option "Ask a question".



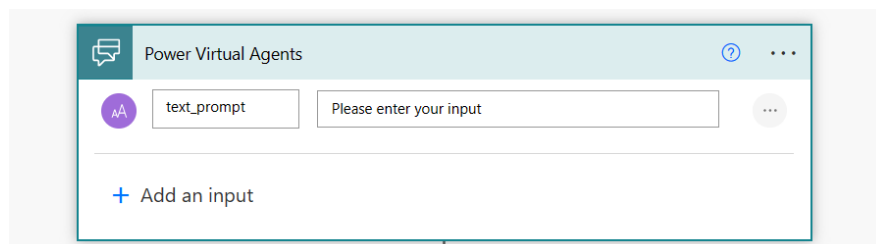
- Add the default question that you would like your bot to ask (See below picture). Choose for "Identify", the value "User's entire response". Assign a new variable name, for the "Save response as", e.g. User\_prompt



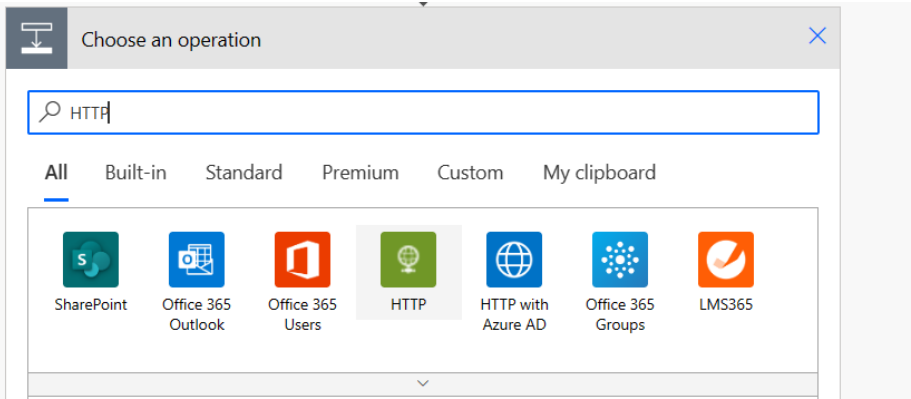
9. Same as before, click on the “+” symbol under the node you just added. Go to “Call an action > Create a flow”.



10. This action will redirect you to Power Automate, a service that helps you create automated workflows between your apps and services to synchronize files, get notifications, collect data, and more (find out more here: [Get started with Power Automate \(contains video\) - Power Automate | Microsoft Learn](#)), in order to create your flow. On the top node "Power virtual Agents", add an input and provide a name for it, e.g. text\_prompt. Leave the value of the variable as is.



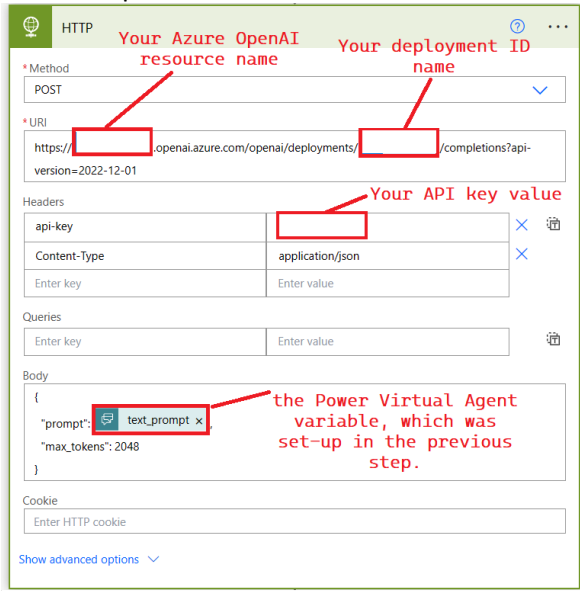
11. Click the + symbol after the node you just modified, select "Add an action" , type "HTTP" and select the simple HTTP option available.



12. Populate with the following with the details as in the table and image below:

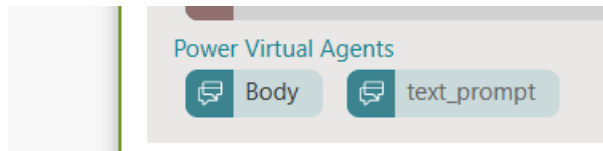
Method	POST
URI	Follow the instructions on this link in order to construct your URI, <a href="#">Azure OpenAI REST API reference - Azure OpenAI   Microsoft Learn</a>
api-key	How to retrieve your APIKEY and Endpoint for OpenAI, see here: <a href="#">Retrieve Azure OpenAI KEY and endpoint   Microsoft Learn</a>
Content-Type	application/json
Body	<pre>{   "prompt": "the-variable-name-with-user-text-prompt",   "max_tokens": 2048 }</pre>

Verify that all info is populated as in the example below.

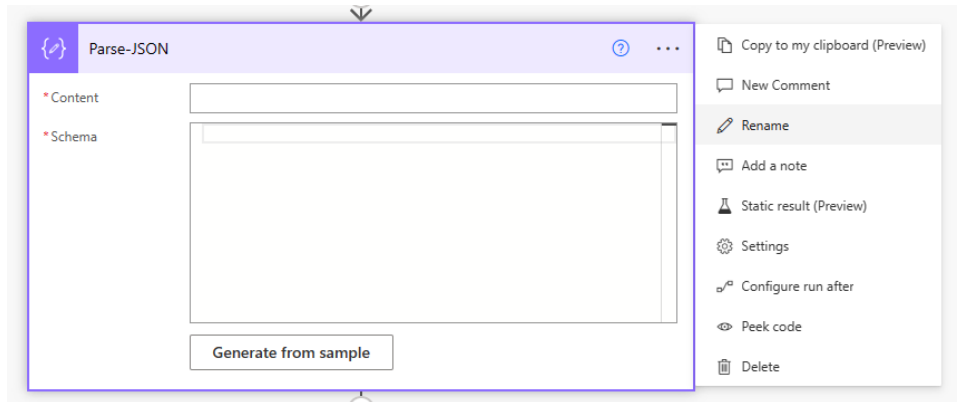


13. Replace the "the-variable-name-with-user-text-prompt" in the body, with the variable name we defined above. To do that, once you click here, there are going to be several options proposed, scroll all the way down until you

find the "Power Virtual Agents" section and select the name of the variable you defined previously -- in this example it's "text\_prompt".



14. Next, click on the "+" icon to add a new action and search for "Parse JSON".  
Rename the action to "Parse-JSON " as shown in the picture below:



As in the previous step, add the "Body" variable in the content. For the Schema, add the following (you don't need to click on "Generate from sample" option):

```
{
  "type": "object",
  "properties": {
    "id": {
      "type": "string"
    },
    "object": {
      "type": "string"
    },
    "created": {
      "type": "integer"
    },
    "model": {
      "type": "string"
    },
    "choices": {
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "text": {
            "type": "string"
          },
          "index": {
            "type": "integer"
          },
          "finish_reason": {
            "type": "string"
          }
        }
      }
    }
  }
}
```

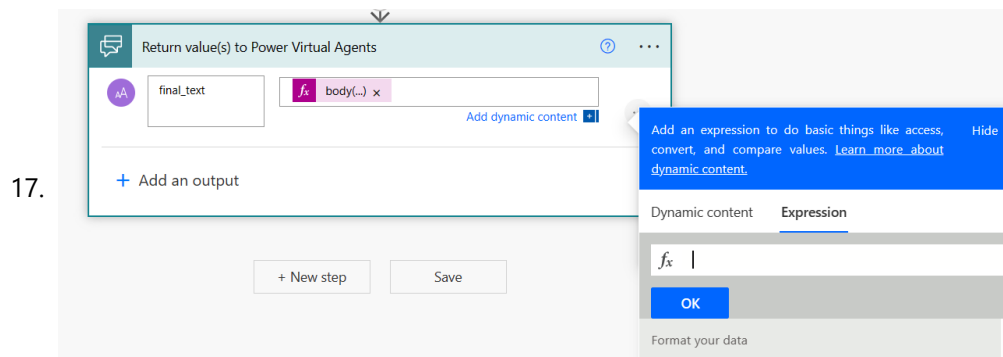
```

    },
    "logprobs": {}
  },
  "required": [
    "text",
    "index",
    "finish_reason",
    "logprobs"
  ]
}
},
"usage": {
  "type": "object",
  "properties": {
    "completion_tokens": {
      "type": "integer"
    },
    "prompt_tokens": {
      "type": "integer"
    },
    "total_tokens": {
      "type": "integer"
    }
  }
}
}
}
}

```

15. The final step is to write the output of the REST API request into a variable, in this case we called it "final\_text". Click on the text field for assigning the value, right next to it. A new window will pop-up,

16.



18.

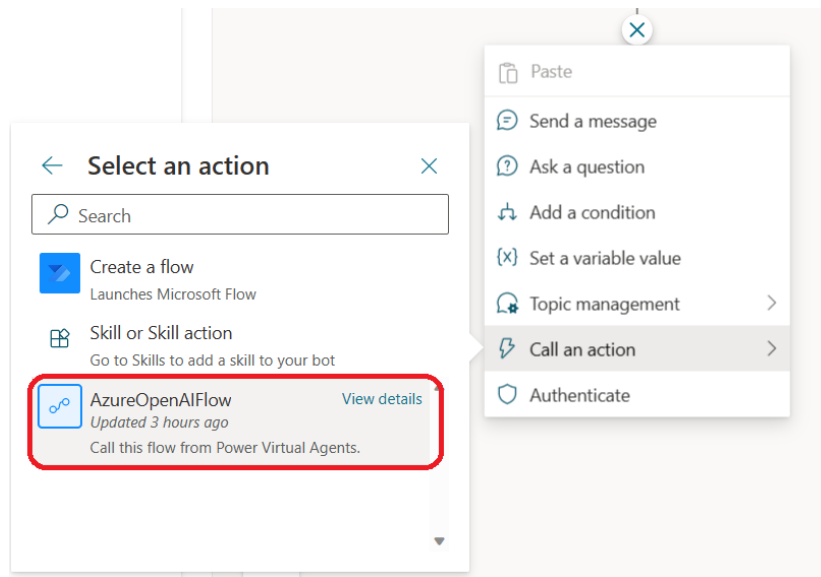
19. choose the "Expression" tab and add the following expression making sure that Parse-JSON is written correctly, then OK:

**body('Parse-JSON')?['choices'][0]?['text']**

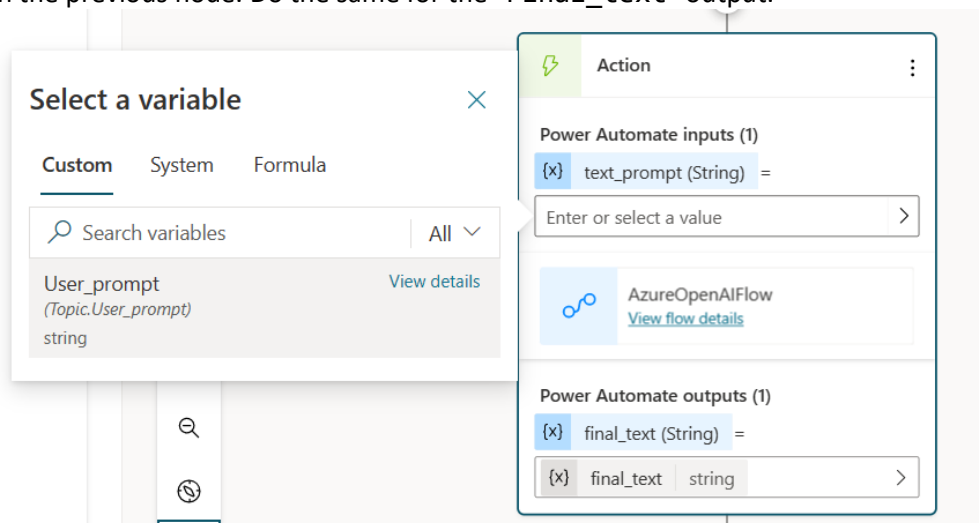
16. Your Power automate flow is now ready, so click Save on the top right of your page. I have saved mine as "AzureOpenAIFlow". You can check the flow with the flow checker, and you can test it to make sure it works fine.

17. Now we are going back to the topic we were working on until step 9. Click on the "+" symbol under the node you just added. Go to "Call an action > Select your new Power Automate Flow name.

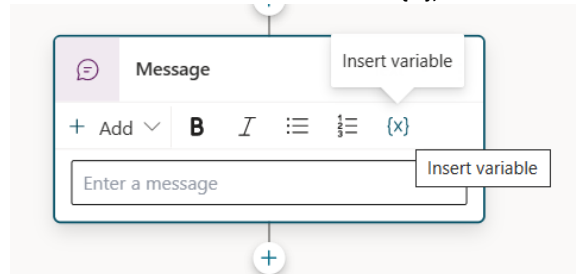




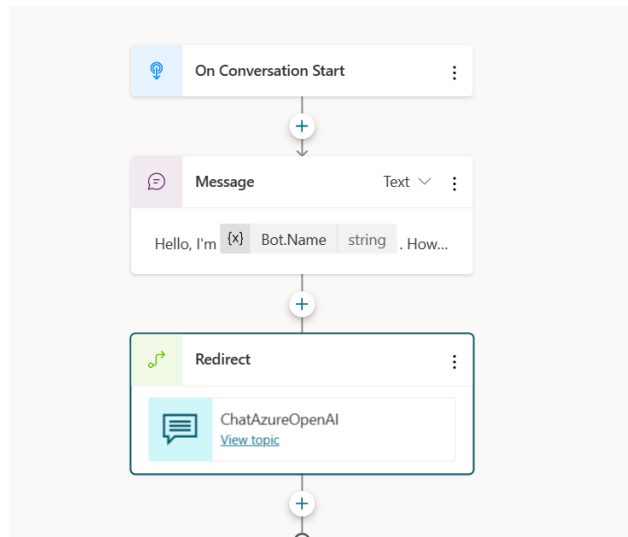
18. Select a variable's value to be assigned to your text\_prompt. In this example, this is the "User\_prompt" variable, which was set-up in the previous node. Do the same for the "Final\_text" output.



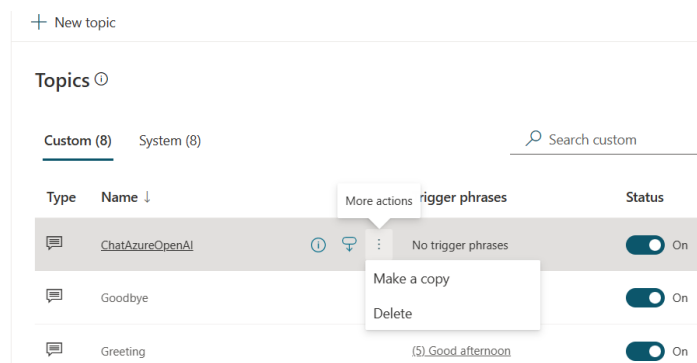
19. Then, let's create a new message back to the user with a print out of our Azure OpenAI response to the user question. Click on "+", select "Send a message" and select the Insert a variable {X}, choose "final\_text". Click Save.



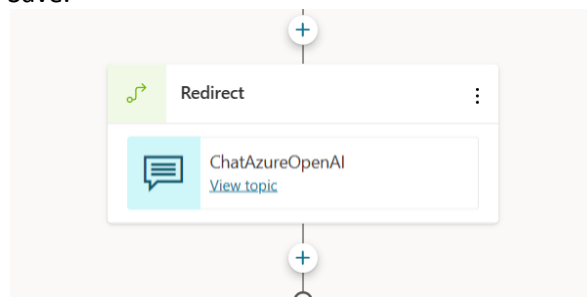
20. We are almost done! Let's now add this just created topic as conversation starter for our bot. Go to Topics > System > Choose "Conversation start". Click on the + after the message node to add a node. Go to Topic Management > Go to another topic > Choose the newly created topic from the previous steps. In this example, it's the "ChatAzureOpenAI". Like the picture below:



21. Replicate the "ChatAzureOpenAI". Got to topics, click on the 3 vertical dots and choose Make a copy.



22. In the new window, rename the topic to something like "ChatAzureOpenAI\_follow\_up". You can edit the questions, like "What else would you like to ask me", etc. You can also add more than one so the bot can randomly pick what to ask. At the end of the flow, add a new node. Go to Topic Management > Go to another topic > Choose the "ChatAzureOpenAI". Once done, Save.

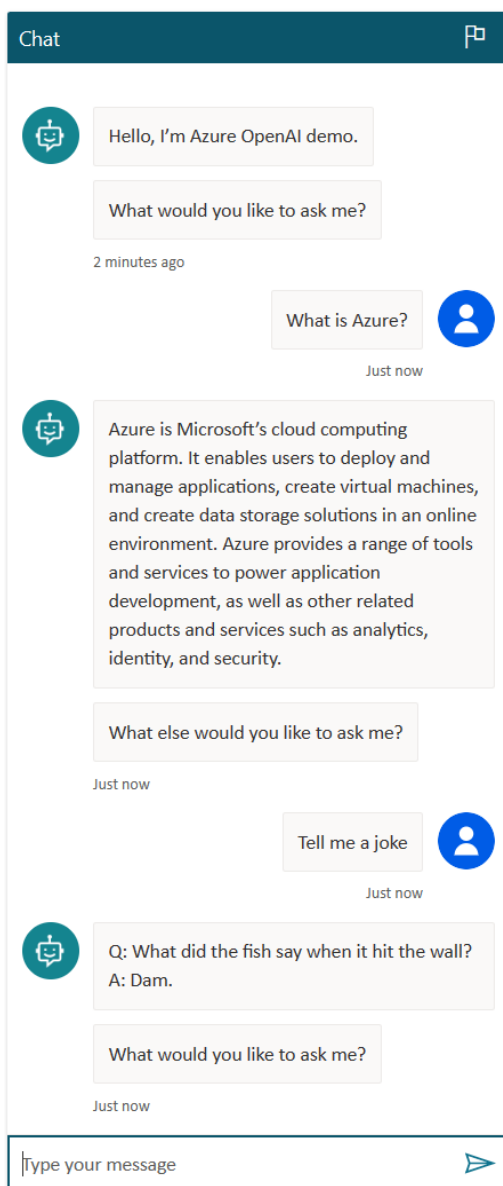


23. [explain what is this final step?] Go to the Topics, go to "ChatAzureOpenAI" topic and at the end of the flow, add a new node. Go to Topic Management > Go to another topic > Choose "ChatAzureOpenAI\_follow\_up". Click Save.

And that it's! A long sequence of steps, but the bot is created and the connection to Azure OpenAI is established. It's time to try the bot.

## Let's try the bot.

Click on the Test your bot option on the bottom left of your screen. (add a screenshot) and ask whatever you want. The following screenshot shows an example of a conversation.



Closing note – that I showed you above was the wiring of Power Virtual Agents to Azure OpenAI, to expose a way to interact with this powerful language model using a chat interface. This should not be confused with ChatGPT, a separate offering from OpenAI that includes other conversational components such as short-term memory. Keep an eye on the Azure blog for updates on the availability of Azure OpenAI ChatGPT.