Microsoft Power Virtual Agents in a Day Preview

Lab 03: Authoring 101 in Unified Authoring

Hands-on Lab Step-by-Step

November 2022

PVAIAD Preview

Contents

Power Virtual Agents

This lab is subject to the Terms of Use on page 31 of this document.

Goals for this lab

After this lab you will be able to: • Understand the core capabilities of the message and question node. • Become familiar with the system topics and their behavior • Understand the fundamentals of entities and slot filling • How to use variables in Power Virtual Agent • Understand extensibility capability including PowerFx and extended node properties	• The time to complete this lab is [60] minutes.

The scenario below is used as context for the examples shared in the Labs. You are encouraged to use your own fictious company example or continue using the Contoso Coffee example from Lab 01 if you complete the planning module or do not wish to use your own fictious company. It is important to continue to build upon your scenarios as you go (in reality, business analysis would be done over a longer period for a real project).

Scenario: Contoso Coffee customer support

You can use the following scenario throughout the labs to build your bot around. You can also use your own scenarios if you so wish.

Welcome to Contoso Coffee! At Contoso Coffee, we are passionate about providing efficient, environmentally sustainable, and innovative coffee products, barista experiences, and services. With you as our newest Customer Support team member, we have no doubt that together we will reach higher levels of success and innovation!

Consumer sentiment is at an all-time high, but with increasing competition, it is more important than ever to minimize costs and focus on customer satisfaction so that we can maintain our competitive edge in having a robust customer base, while continuing to invest back into the customer experience and maximize our customer lifetime value.

Contoso Coffee are very interested in providing customers with the capability of asking questions about their order status, delivery time, and even the ability to be able to modify their order and cancel it. This will unlock a lot of capability for their call center staff and be able to help with deflecting common questions that take a lot of time on the phone, making both customers and agents happier and have more time. They also highlighted briefly that in B2B scenarios where they often

work with other manufacturers, they would like to provide access for those partners the ability to view and ask questions around latest invoices or payments or be escalated to live agents if they are in a specific tier of manufacturer.

Customer support costs have continued to rise as the customer base has grown, and the company can no longer continue to expand the Customer Support team. You have been tasked with finding a solution using Power Virtual Agents.

Prerequisites

Please note that some labs, especially later labs, do reference previous labs in reference to capabilities and previous tasks. The labs have been designed so as long you have access to a Power Virtual Agent trial which has Unified Authoring Preview available, you can get started from any lab without having to complete the previous module to be able to move forward.

For *Lab 03 – Authoring 101* in PVA Unified Authoring, we recommend you completing the following labs to get the best experience:

- Lab 00 – Overview
- Lab 01 – Planning your first chatbot
- Lab 02 – Building your first chatbot

Preview Material Notice

Please note that this material is provided is under continuous development and is targeted at the Preview version of Power Virtual Agents, Unified Authoring, released in public preview on November 10th. Please expect the product to change as the preview period continues and so some of the screenshots or text descriptions may not be what you experience in the latest version of preview in your environment.

We would appreciate your feedback on Power Virtual Agents Unified Authoring and on this hands-on-lab, such as the quality of documentation and the usefulness of the learning experience.

Please use the survey at https://aka.ms/pvaiadpreviewsurvey to share your feedback.

You may provide feedback for each module as you complete it or at the end once you've completed all the modules. Thank you!

Exercise 0: Setup

Power Virtual Agents Unified Authoring, currently in Preview at time of writing, has evolved the bot authoring experience to help make it easier and faster to give authors more tools when creating conversational experiences. The result should be that more customers using Power Virtual Agents will be able to have delightful and engaging conversations with a virtual agent, where it meets the goals set out by the business.

In this lab, we will build upon a basic topic from Lab 02 and layer additional experiences you can author, starting with entities all the way to rich text options. This is to demonstrate a basic conversational experience and then how you can use the latest features to further enhance your bot.

Task 0: Create your Custom Topic

Important Note: *If you are completing this lab from Lab 02 – Planning your first chatbot, you should already have everything created. Review the items below and if you already have your topic, you can skip this task.*

This task is to create your own custom topic, test and publish it to create a 'base' for you to work with.

1. Create a custom topic with a title and trigger phrases.
2. Create a question node within the authoring canvas using 'users' full response' to identify the response data
3. Create a message node after the question node
4. Add an 'end conversation' node at the end of the flow.
5. Save your topic and test.
6. Publish your bot!

If you require guidance to complete the above tasks, please see Lab 02 – Building your first chatbot which covers the steps on how to create each item.

Exercise 1: Entities in Power Virtual Agents

Understanding and using entities is the ability for Power Virtual Agents to understand what the user is saying. For example, if the user says, "I tried to use my gift card, but it doesn't work," the bot is able to route the user to the topic related to gift cards not working – even if that exact phrase isn't listed as a trigger phrase.

Natural language understanding helps the bot identify entities in a user's input. An entity represents a real-world subject, such as a phone number, zip code, city, or even a person's name. Your bot can smartly recognize the relevant information from user input and save it for later use.

For example, if the user types "I want fifty red coffee machines", the AI can understand that:

- "fifty" is the number "50" and is also the number of products to purchase.
- "red" is a color and is the color of the products to purchase; and
- "coffee machine" is the product the person wants to purchase.

In Power Virtual Agents, some of these (such as numbers and colors) have already been taught to the AI for every Power Virtual Agents chatbot; others (such as the fact that "coffee machines" is a product or that this red is specifically the color of a product) have to be specified by the bot author, as we will show in this lab. There are two types of entities:

- **Pre-built entities** represent the most used information, such as age, colors, numbers, and names. Power Virtual Agents bots recognize these automatically.
- **Custom entities** are entities you make. While the pre-built entities cover commonly used information types, sometimes you'll need to teach the bot's language understanding model some domain-specific knowledge. For instance, you may need to create a custom entity for your product types.

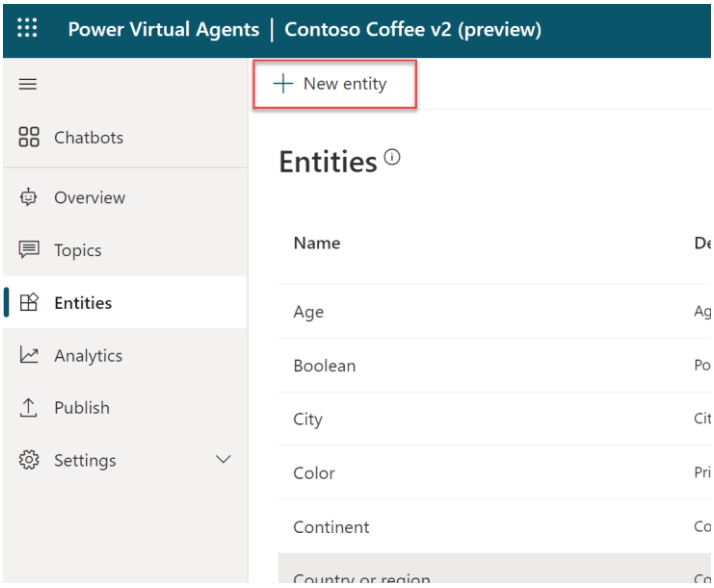
Smart match and synonyms can make your bot even smarter:

- **Smart match** provides the flexibility to let the bot match the user's input to an entity that is a near match but not perfect. Specifically, it lets the bot autocorrect misspellings and expands the matching logic semantically, such as automatically matching "softball" to "baseball". You can turn this off if you need a match to be perfect, such as if the entity contains model numbers or error codes
- **Synonyms** allow you to recognize that something the user typed matches an option you provided. For example, for "free shipping" you can add "complimentary shipping" as a synonym. For "expedited shipping", you can add "2 day shipping" or "overnight shipping" as synonyms. If the user types any of these, they will be matched appropriately.

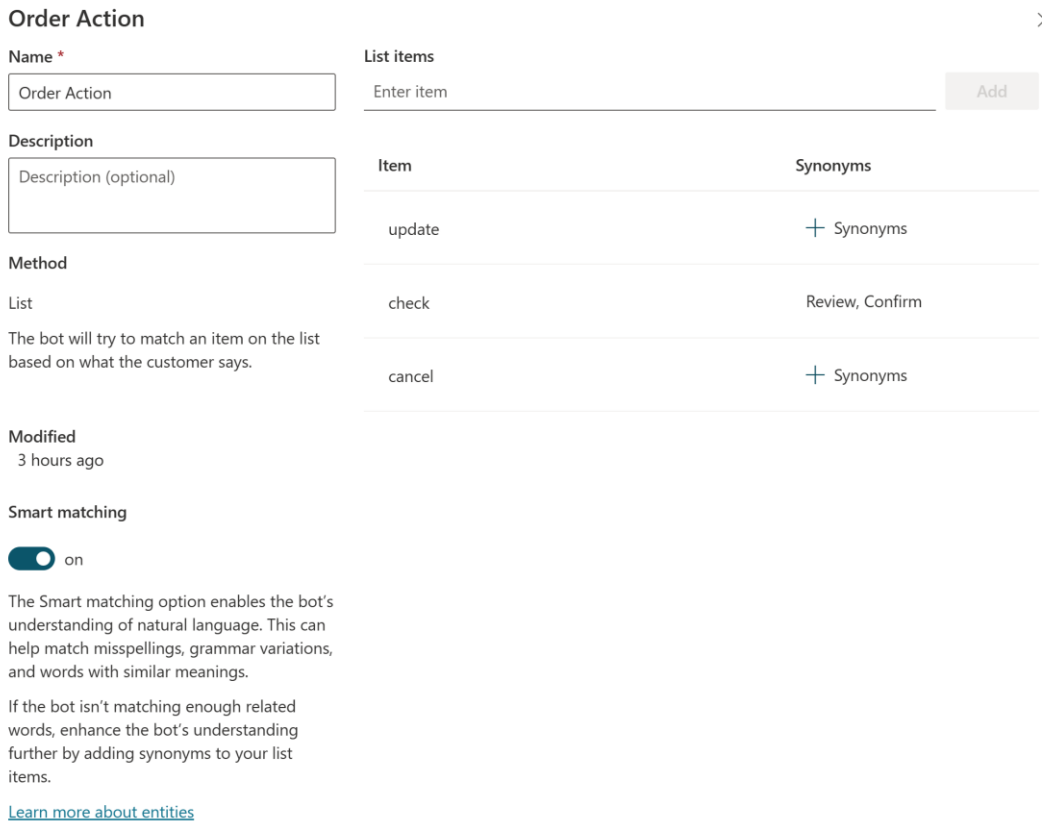
Let's briefly look at how to create a custom entity and use an entity within the topic you already have created.

Task 1: Using Entities and Slot Filling

1. With your bot open, on your left-hand side navigation screen, click on **'Entities'** within your topic and click **'New Entity'.** If your using the demo scenarios provided within the lab, call it 'Order Action'

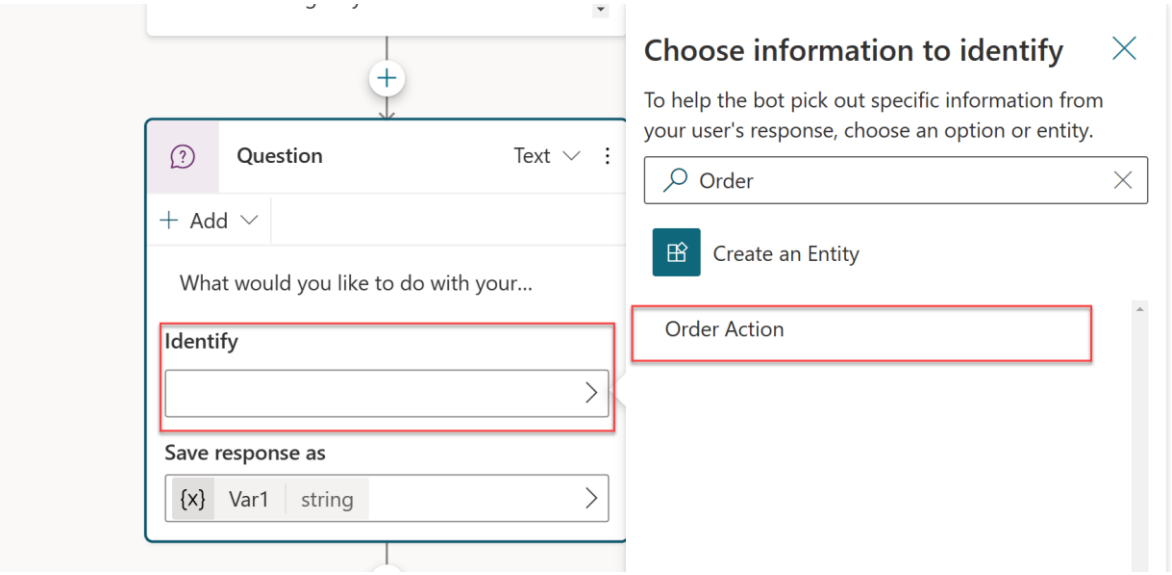


2. Select **'Closed List'**
3. Add three options for **Update, Check, and Cancel**
4. You can also choose to add synonyms by selecting synonyms to each option (Optional for this task)
5. Turn Smart Matching on

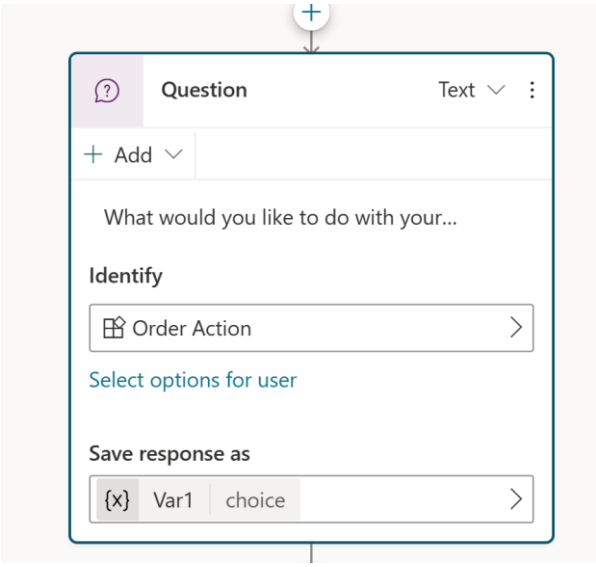


This will now create an Entity for your Order Action to be used with the Question node in your topic to place 'users' entire response' with 'Order Action'

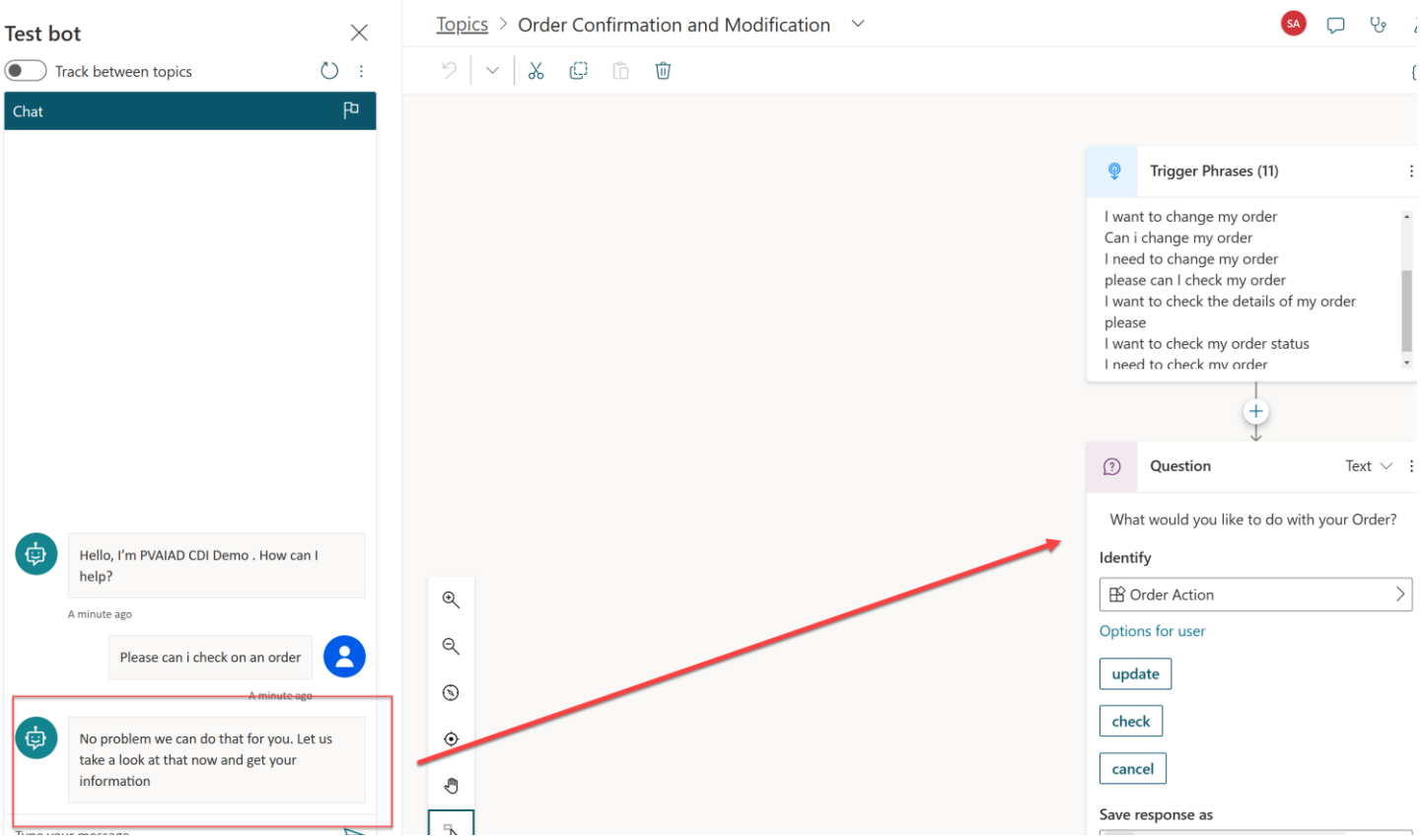
6. Navigate back to your topic you created in Lab 02 or in Exercise 0 and select the Question Node you had created that was originally identifying the user's full response. Click 'Identify' and you will see a slide out menu on the right-hand side to select an Entity. Search for your custom Entity you made in the previous step called 'Order Action' and select it



7. Now it is selected, click on 'Select Options' and select all options to display to the individual using your bot



8. You have successfully configured a Custom Entity for your Question. The default behavior is if the variable the question response is stored in has a value, the question to be skipped. Save your topic and use the test pane to try out how Entities and Slot filling work by typing one of your trigger phrases such as 'Please can I check on an order?' (if your using the demo scenario and data) and you will see that the question is *not even asked! This is because you have used entities and slot filling to retrieve the information from the first question the user has asked*. This avoids asking the user a question which they have already provided the information for.



Exercise 2: Variables

You are beginning to enhance the topic you created in Lab 02. In the previous section you used Entities and Slot Filling to automatically detect the data from a user's sentence and store specific data in a variable. Let's now take a look at how we can use the data we obtained from the question in a variable and display it within a message using Power Virtual Agents Unified Authoring.

Task 1: Understanding Variable Types

Variables let you save responses from your customers to help guide the conversation (such as to determine whether to provide different instructions for returns based on purchase price of the item) and can be used directly in the conversational response from the bot ("I can help you return the {variable_ProductName}."). By default, a variable's value can only be used in the topic where the variable is created. However, if you want the bot to use the same value across other topics, you can choose to make it a global variable (you might know this concept from other systems as a global variable). This means that when the conversation moves to a different topic, the bot can remember and use variable values filled in previous topics in the conversation. In Power Virtual Agent Unified Authoring, variables can also be configured using PowerFx functions and outside of a question node.

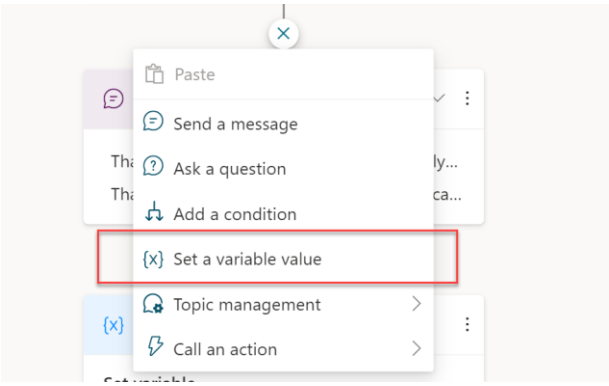
There are different types of variables in Power Virtual Agents:

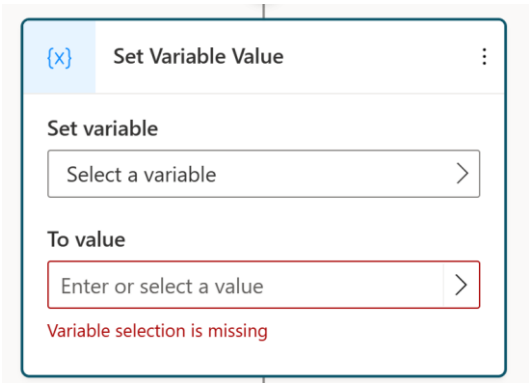
- **System Variables:** These are variables which are normally populated with system data which can be used in conditions. System variables are not user made and are part of the platform. System variables can be accessed in the authoring canvas from the variable selection under 'System'.
- **Topic Variables:** Topic variables are user made variables from either in a 'Set Variable Value' node or 'Question' node. These variables are by default, limited in scope and available only in the topic, which is being authored, and not other topics. There are two options to expand this scope for topic variables if they can receive values **from** other topics and return values **to** other topics. By setting these options, a topic variable is no longer limited to only being used in the topic but can be used by other topics. Topic variables can be accessed in the authoring canvas from the variable selection under 'Custom'.
- **Global Variables:** Global variables are available from any topic, and they are a good way to store data used by multiple topics to be able to assist in the 'conversation', regardless of how many topics get triggered within it. Topic variables can be accessed in the authoring canvas from the variable selection under 'Custom'.

Variables can be used in several places, including **'Questions', 'Conditions'** and the **'Set Variable Value'** nodes. The variable can be either a custom value using PowerFX, a user entered value, a response from a question, or system variable values.

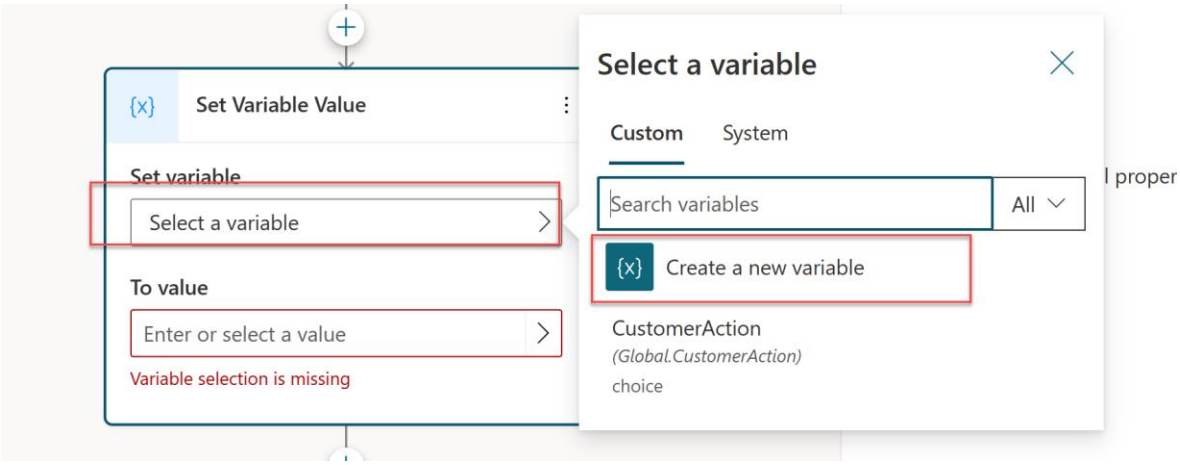
Take this first exercise to familiarize yourself with the *'Set Variable'* node and to review the different types of variables. This task will involve creating a new node, creating a new variable, renaming the variable, and seeing what other variables are within Power Virtual Agents to use at the system level. At the end of this task, you will then delete this node.

1. In your topic, create a 'Set Variable' node anywhere in the canvas by selecting the add node button, and clicking **'Set a variable value'**

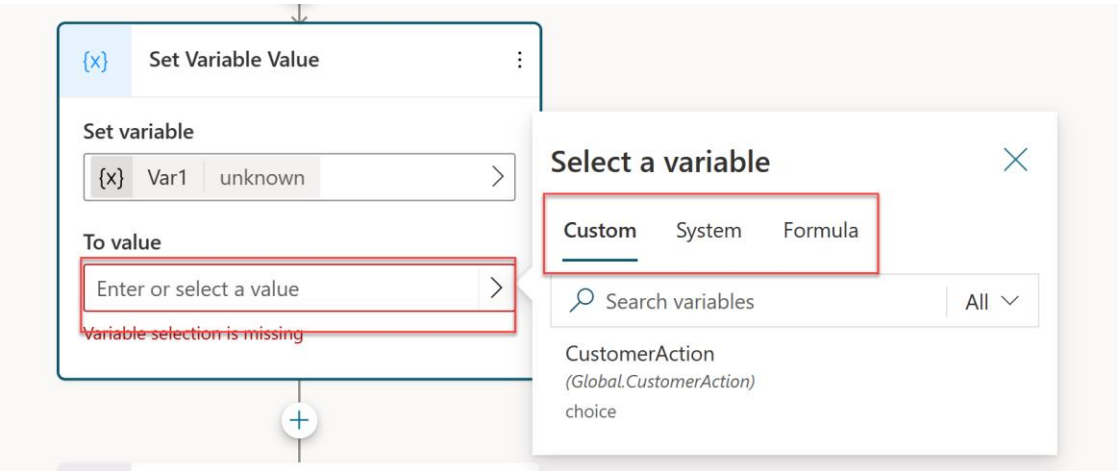




2. A new 'Variable' node is created. Click on 'Select a variable' as shown in the 'Set Variable' area in the node, shown below. This opens a flyout panel on the right-hand side. Select 'Create a new variable.'

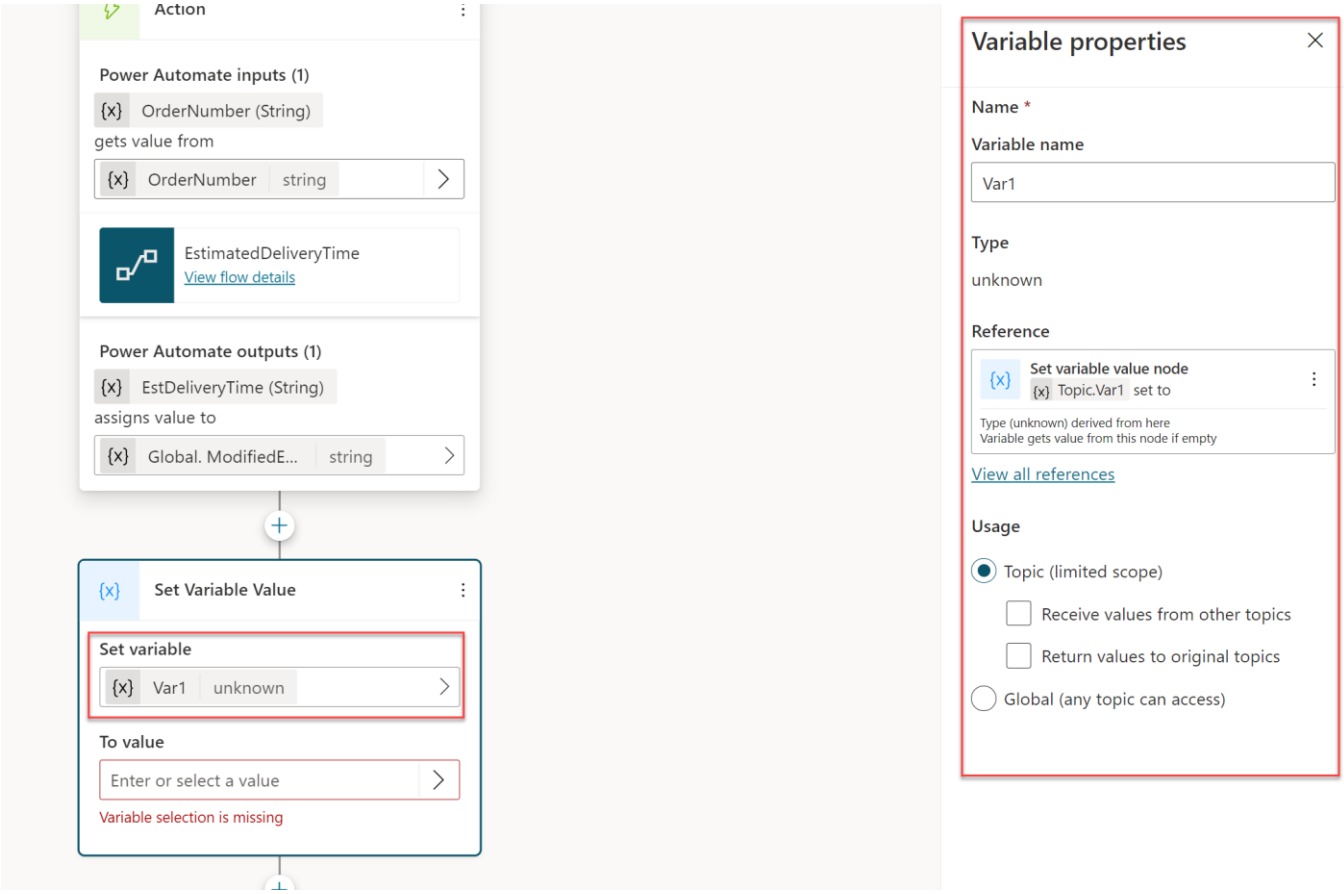


3. By clicking 'Create a new variable' your new variable is made and by default called 'Var1' . (Or a different number if you have already created a variable with this name, like *Var2* or *Var3*).



Click on the name of the variable to open the *Variable Properties* panel on the right-hand side of the screen, as shown below. This is where a variable can be renamed and also the scope of a variable can be changed from Topic or Global.

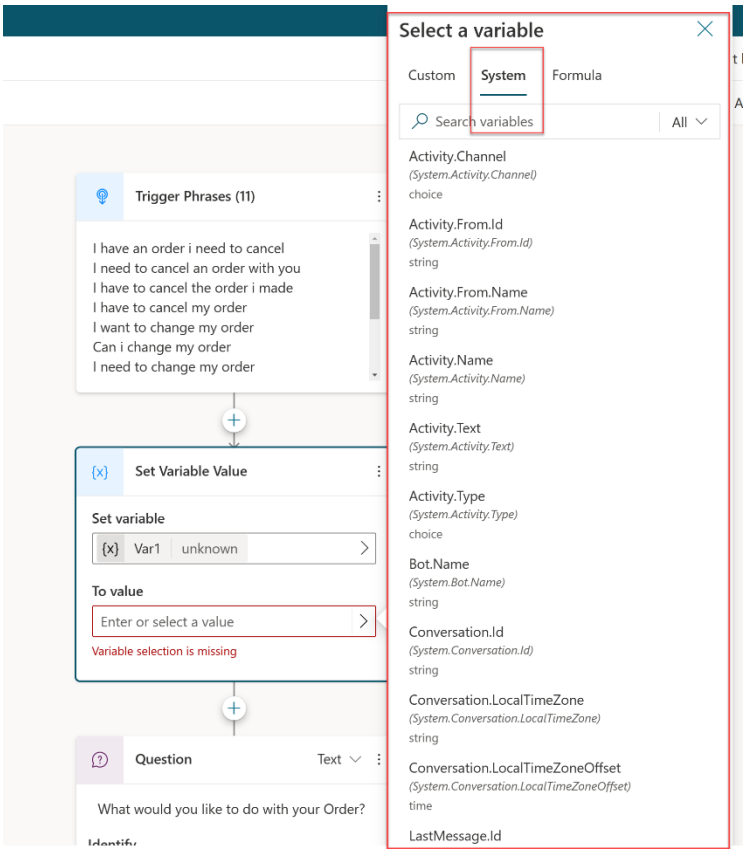
Pro Tip! It is best practice to ensure you name your variables to something descriptive based on the data that is being stored. This not only helps you in the future, but also other makers!



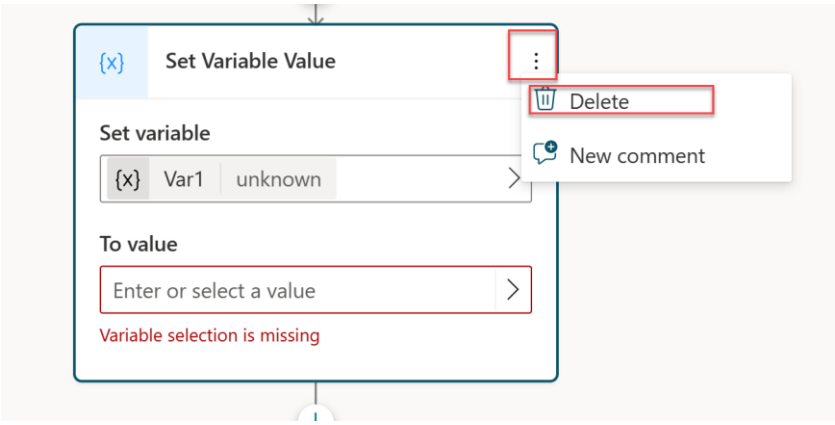
4. Click 'x' at the top of the Variable panel to close it. Now look to see what data you can use to store in the variable. You can use other variables you have created in your authoring canvas, or you can use system variables, or even formulas. On your 'Set Variable Value' node, select 'To value' and a flyout panel will appear where you can see separate headings named **'Custom'** variables**, 'System'** variables, or **Formula** (using PowerFx, covered later in this Lab).

   Click on the 'System' heading

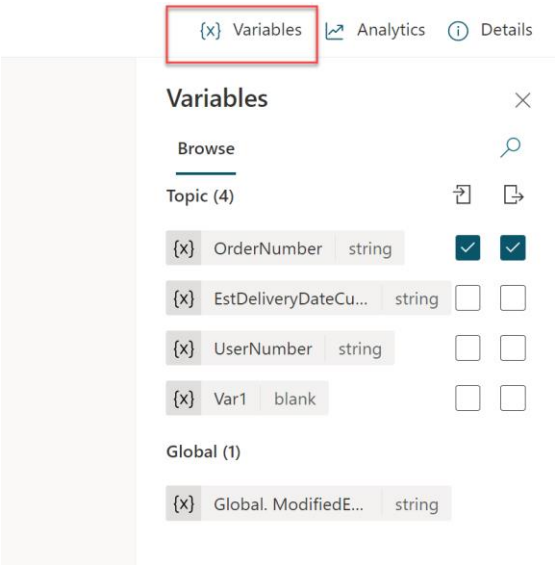
5. This opens the 'System Variable' list so you can now see all of the variables used by Power Virtual Agent. These variables contain data populated by Power Virtual Agents and can be used in your own variables also, as shown below. Review these options so you understand what is available by default.



6. Now you have reviewed the options you have available within the 'Set Variable' node, you can remove this node by clicking on the extended menu options and click 'Delete' as shown below (We will not be using it in the next exercises).



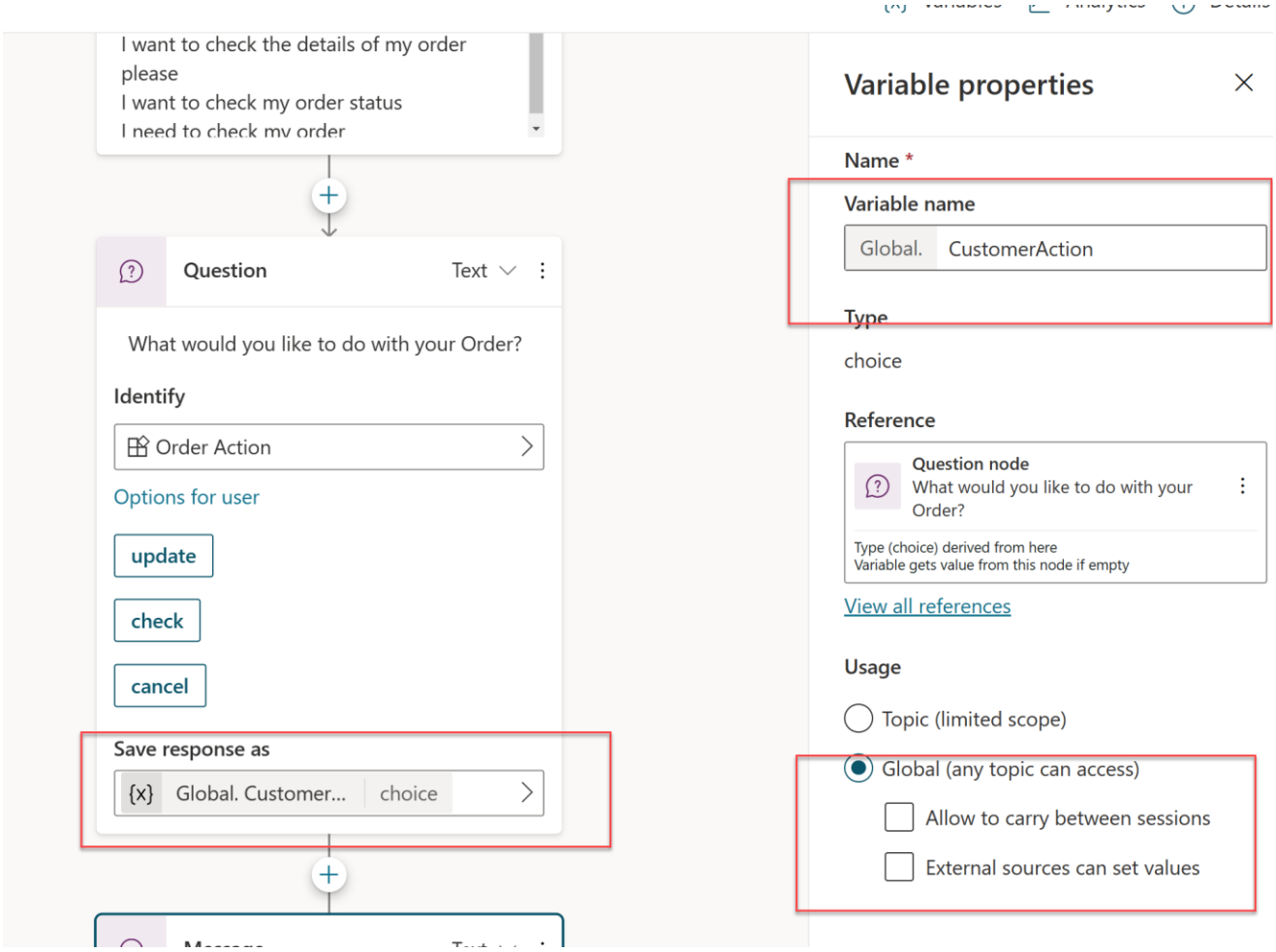
7. From anywhere within the authoring canvas itself, you can also use the **'Variables'** window to review all variables within the topic, including global variables. Navigate to the top right of the screen and select 'Variables', as shown in the screenshot below to be able to view this panel. This is a good place to go to review all variables within a topic, especially large topics.



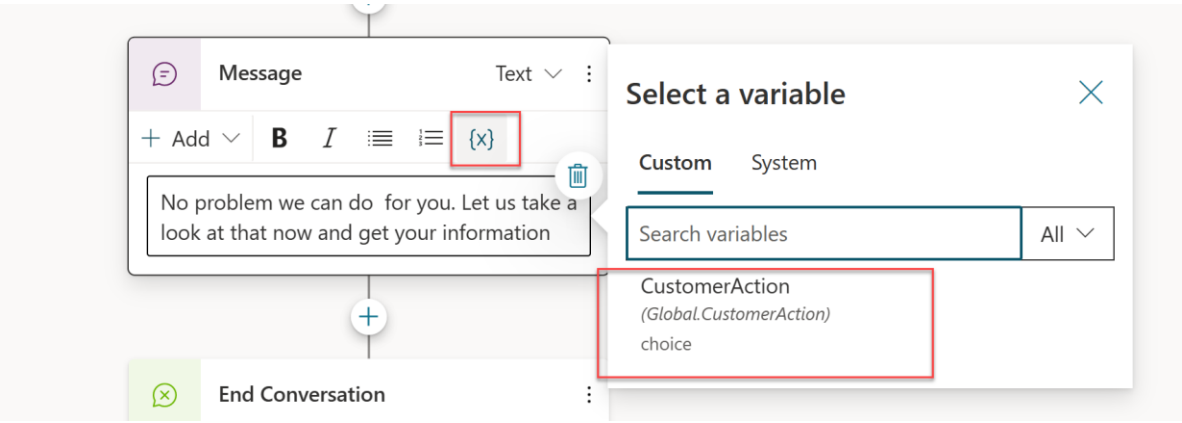
Task 2: Using Variables in a Message Node

Now let's look at how we can use the data from the previous task in Exercise 1. At this point, you should have the question node in your topic linked to an entity.

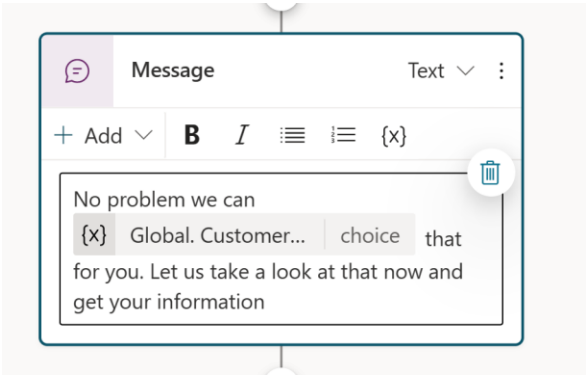
1. Open your question node and click on the variable name. This should be still 'Var1' or similar if you have not yet updated it.
2. In the right-hand side panel that opens, update the Variable Name with **'CustomerAction'** and click on **'Global'** to change the scope from local (topic) to Global, so it can be accessed by other topics, as shown below.



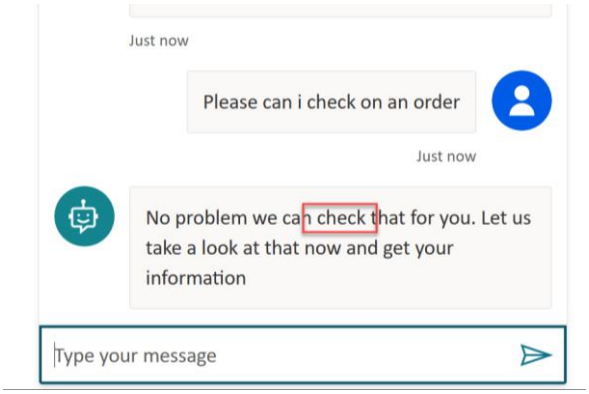
3. Now we have the variable completed from the question or trigger phrase, we want to use this in the message as dynamic data. Click on the message node, and where insert it within the message phrase where it is appropriate by clicking in your message text where you want the variable to display, then click on the '{X}' variable icon, and select the variable you just made - 'Global.CustomerAction'. This is action is common to insert a variable in place of the literal word, making it dynamic based on the user or customers question.



4. You should now see your message node with your text but also the variable within the sentence. This allows for your message to be dynamic depending on what the customer has asked for in the topic and was saved in the variable using slot filling. In the screenshot below, we replaced the word 'do' in the message node with the variable.



5. Use 'Test your bot' to test the behavior of the bot and the changes you made by triggering the topic with a trigger phrase and seeing it in action, as shown below:



Congratulations on completing this task. You have reviewed variables, what they are and the different types in Power Virtual Agents.

Exercise 3: Using Variables in Conditions

Your topic has currently been updated from Lab 02 with Entities and Slot Filling capabilities and it is using dynamic data to store variables and reuse those variables in messages to provide a dynamic authoring experience. Let's now use the same variable in a condition statement in Power Virtual Agent.

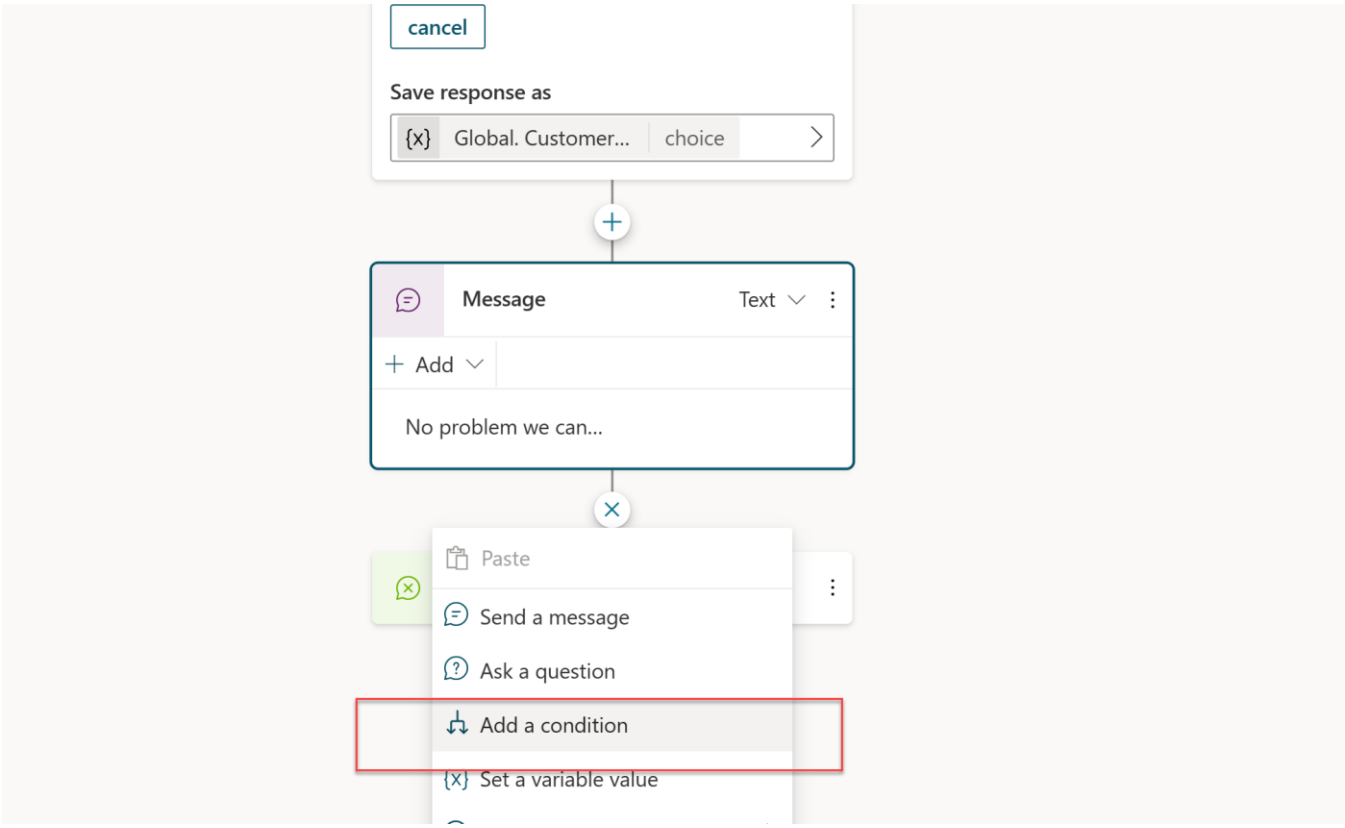
Condition statements in Power Virtual Agents allow a Bot Author to be able to determine behavior of the bot on certain conditions that can be true, false or something else (e.g. if it is blank). They allow and promote flexibility in the authoring canvas, allowing you to provide great customer or user experiences based on the needs, whilst limiting having to create lots of similar topics. Once you begin to use conditions, you create 'branches' which create separate flows that the person using the bot can be directed to. These branches, like a tree, can have their own conditions and so on, depending on what behavior you wish to create.

For more information on conditions in Power Virtual Agents, check out the Microsoft Docs Article here.

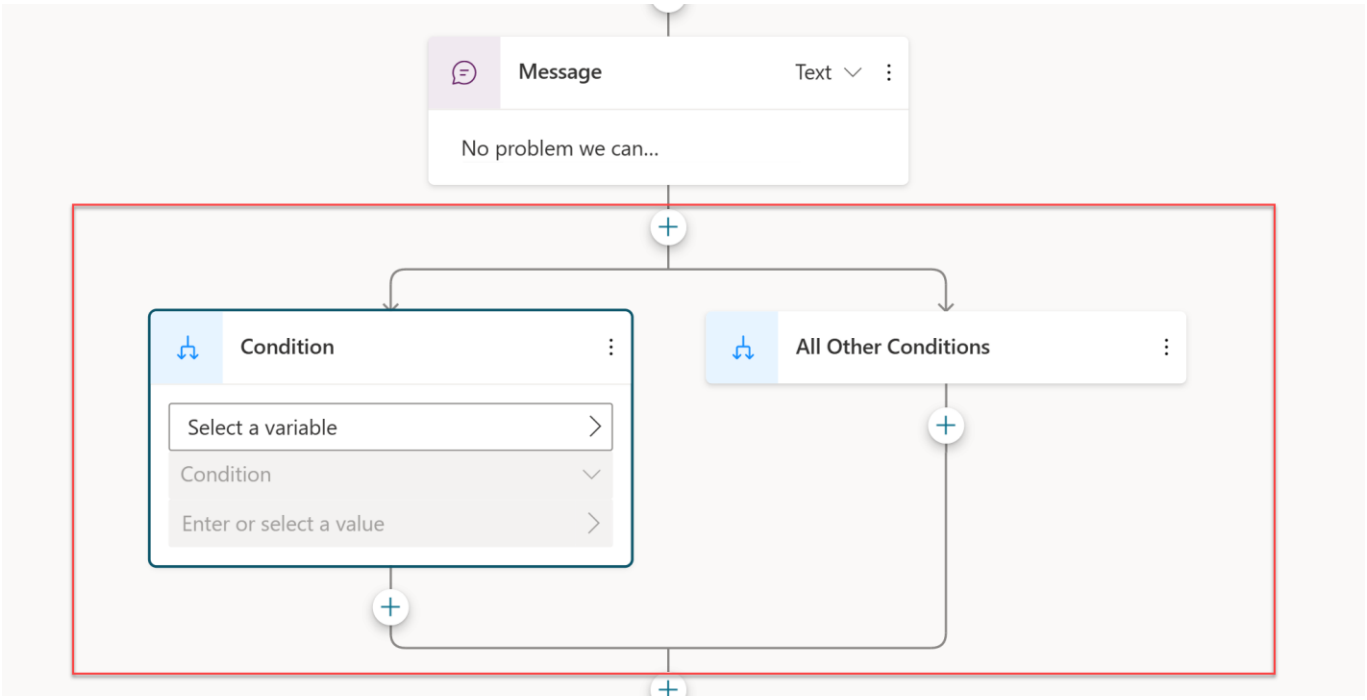
Task 1: Creating a Condition using Variables

Let's go ahead and create a simple condition based on the three variable options we created from the Entity in the first exercise.

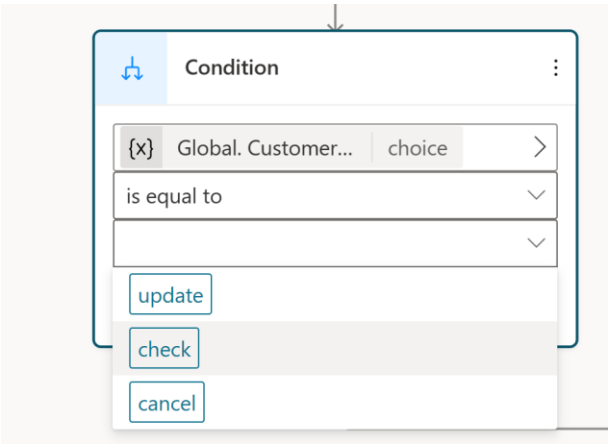
1. In your Authoring Canvas, under the message node you modified in the last task, click the new node + icon and select **'Add a Condition'.**



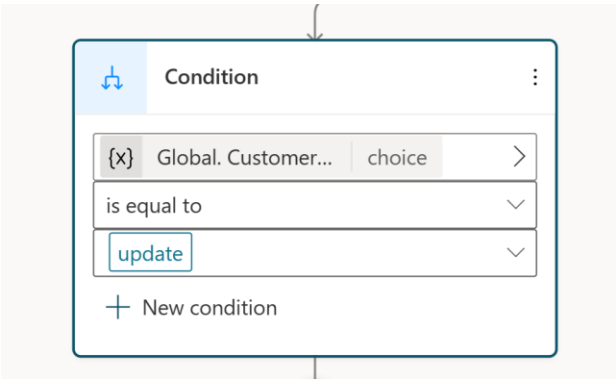
2. You will see two new nodes appear, one is your condition, and another is an exception for *'all other conditions'*. In your 'Condition' node, click 'Select a variable' and select your Global Variable 'Customer Action.'



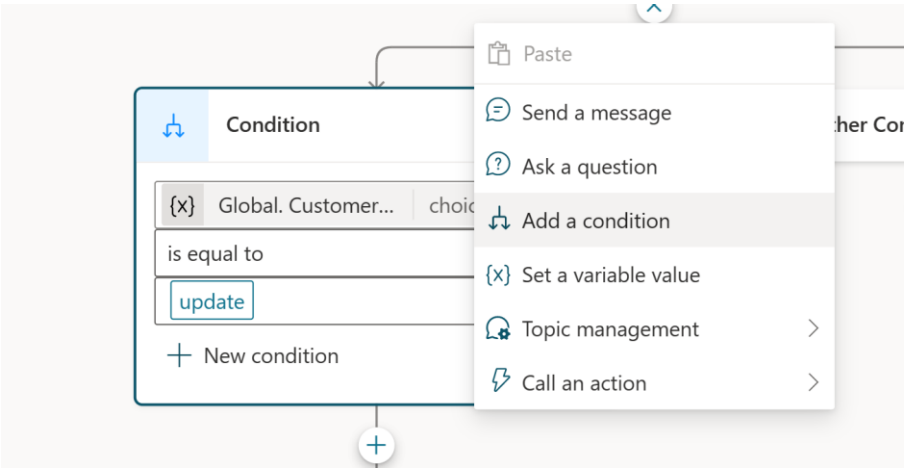
3. Keep the condition operator as *'is equal to'* and select the empty box underneath to display the three options available from the selected variable. Select the **'update'** option.



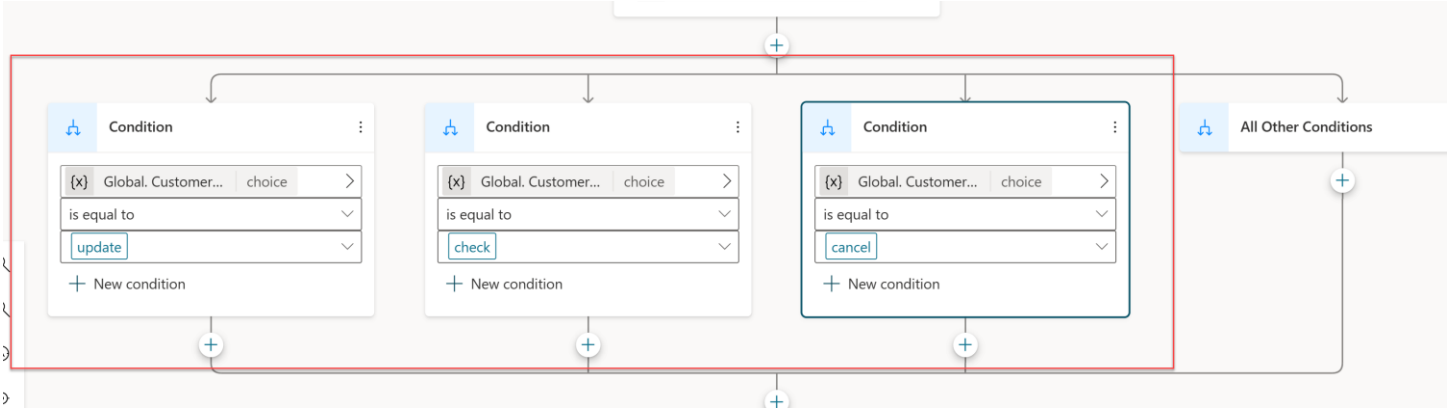
4. You should now have a complete condition where if the CustomerAction value is 'update', as shown below.



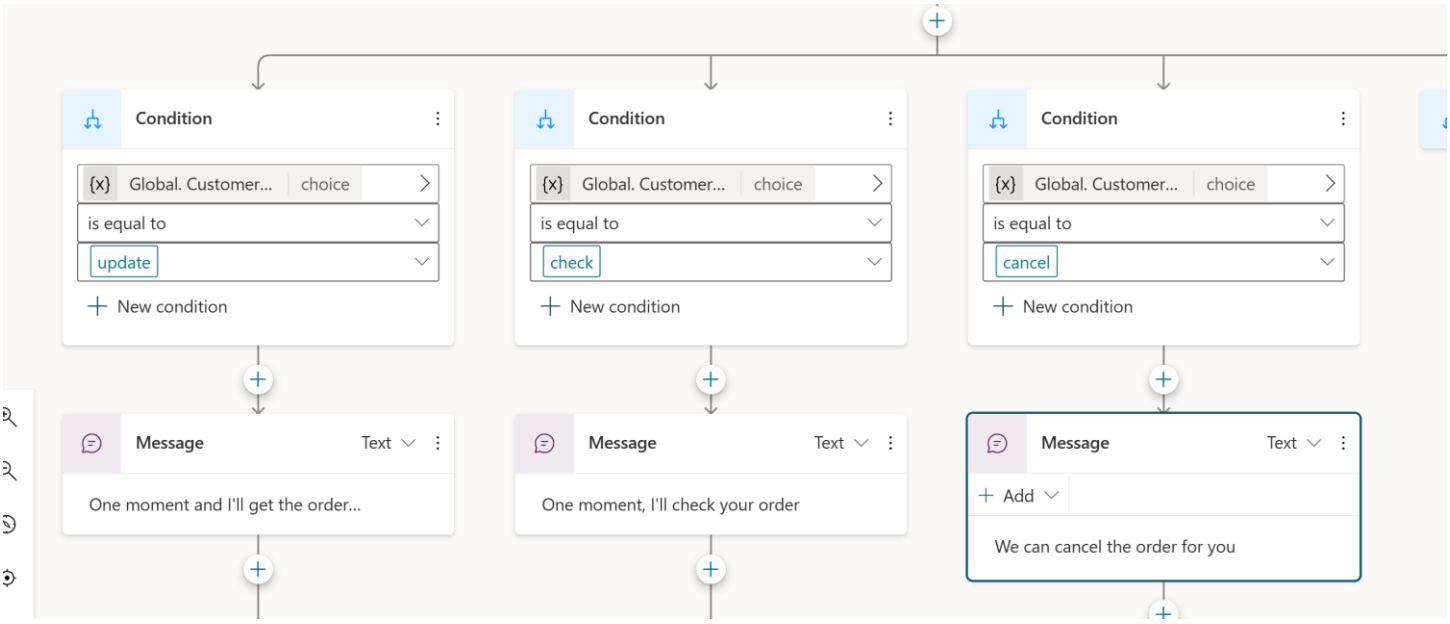
5. We now create two more conditions in this branch for the other two options for the Customer Action variable (check and cancel). Click on the 'add node' icon above the condition and select 'Add Condition' to add another condition



6. Repeat the previous steps by selecting your 'Global.CustomerAction' variable and select the Check and Cancel options in two other conditions so you end up with a conditional branch with three options (plus 'all other conditions') as shown below:



7. Under each of the condition nodes, add a Message Node which displays different text depending on the condition, as shown below:



8. Now save your topic, and use 'Test your bot ' to try out the different trigger phrases and conditions which lead you to the different message outcomes.

Congratulations! You have completed the basics of Conditions and using Variables as parameters within them.

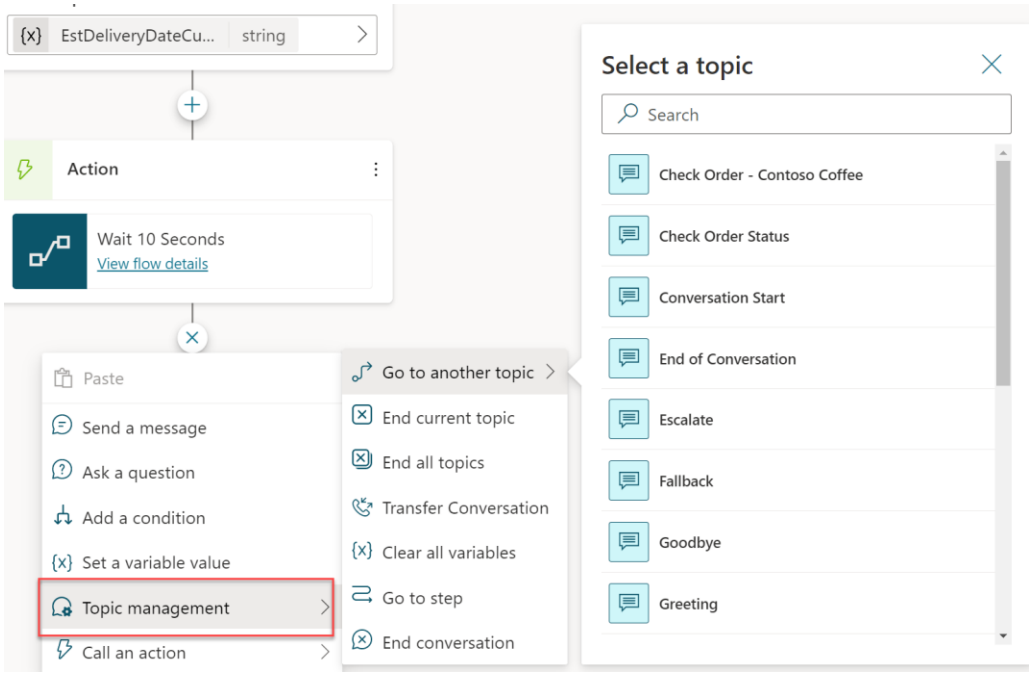
Exercise 4: Using Topic Nodes

Within Power Virtual Agent it has always been possible to redirect to another topic, such as Escalate. In Power Virtual Agents Preview, the capability to link to other topics has been expanded to allow for greater flexibility and easier access to topics, as well as looping in the same topic. Remember that topics are configured to trigger normally based on users typing or saying something to the bot which will trigger an intent match, or based on events such as escalation, conversation start, and other system topics with event driven triggers. Users can also direct to other topics within a conversation where it makes sense to do so or even perform loops within the current topic to another node.

Task 1: Becoming familiar with Topic type nodes

1. In the topic you have been working on previous exercises for this lab, click the **add new node button**, and select **'Topic Management'**. You will see a list of available commands as shown in the screenshot below.

Let's review what each of these options refer to.



- **Go to another topic:** This node has an extended flyout menu where you can then go to another topic that needs to be selected

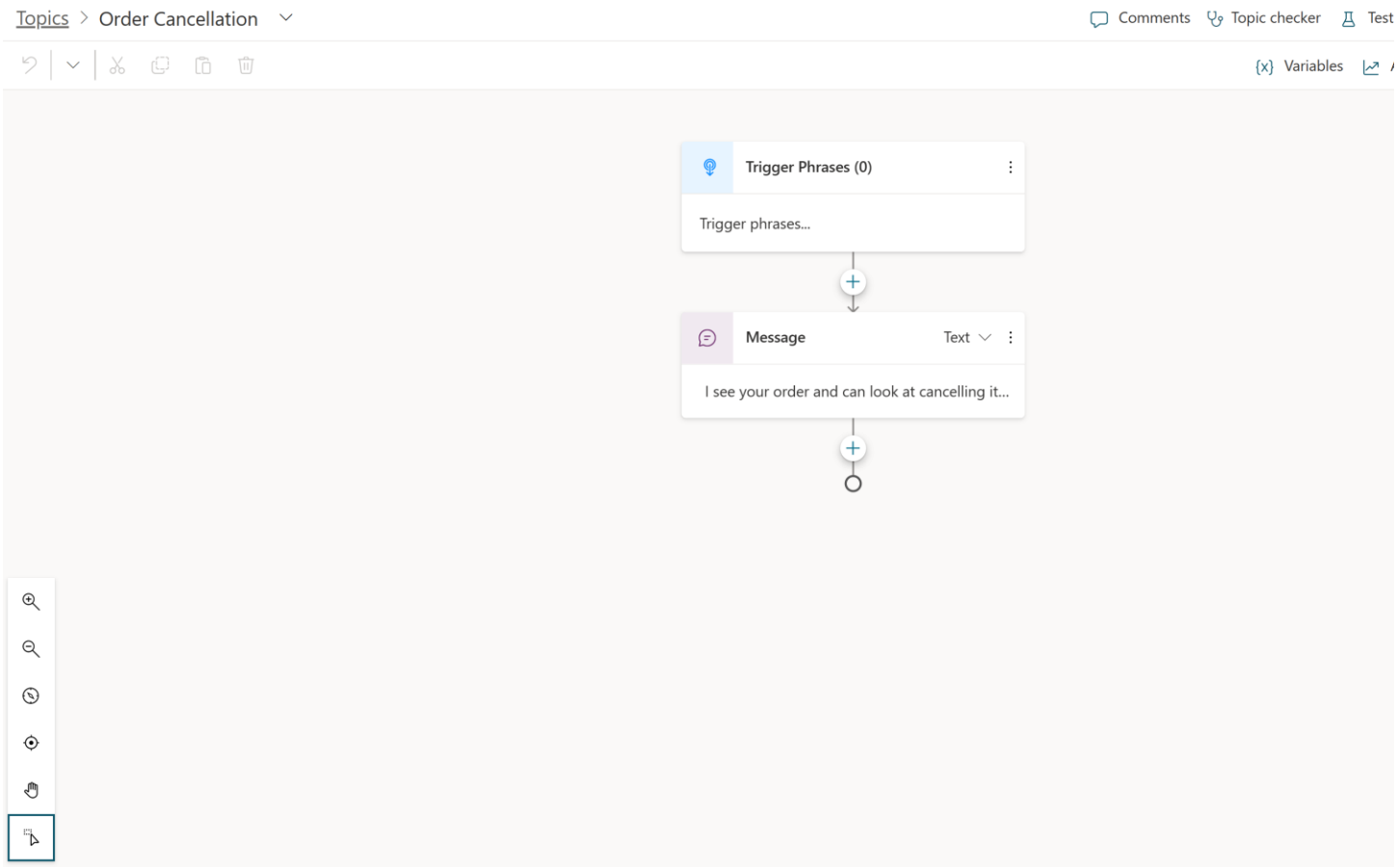
- **End current topic:** This ends the current topic and it is used normally where the topic was called from another topic. It would be returned to where it was originally called from. This can also be used in branching conditions. If this is used to end an entire topic of branch, the behavior will operate like 'End all topics' below.

- **End all topics:** End all active topics and the next message from the user will be as a new conversation, and the bot will trigger the most appropriate topic based on that user message, starting a new active topic.
- **Transfer Conversation** is used to transfer to an agent and send a contextual message. *(Not Supported in Preview)*
- **Clear all variables** clears all variable values. This is useful if you want to begin or loop back into the same topic but take new values, especially if you have configured question behavior properties where a question could be skipped if it already had a value.
- **Go to Step** allows a bot author to navigate to another node in the open topic. This is useful for looping scenarios or if you want to gather more information from the user.
- **End Conversation** confirms the customer has their query answered.

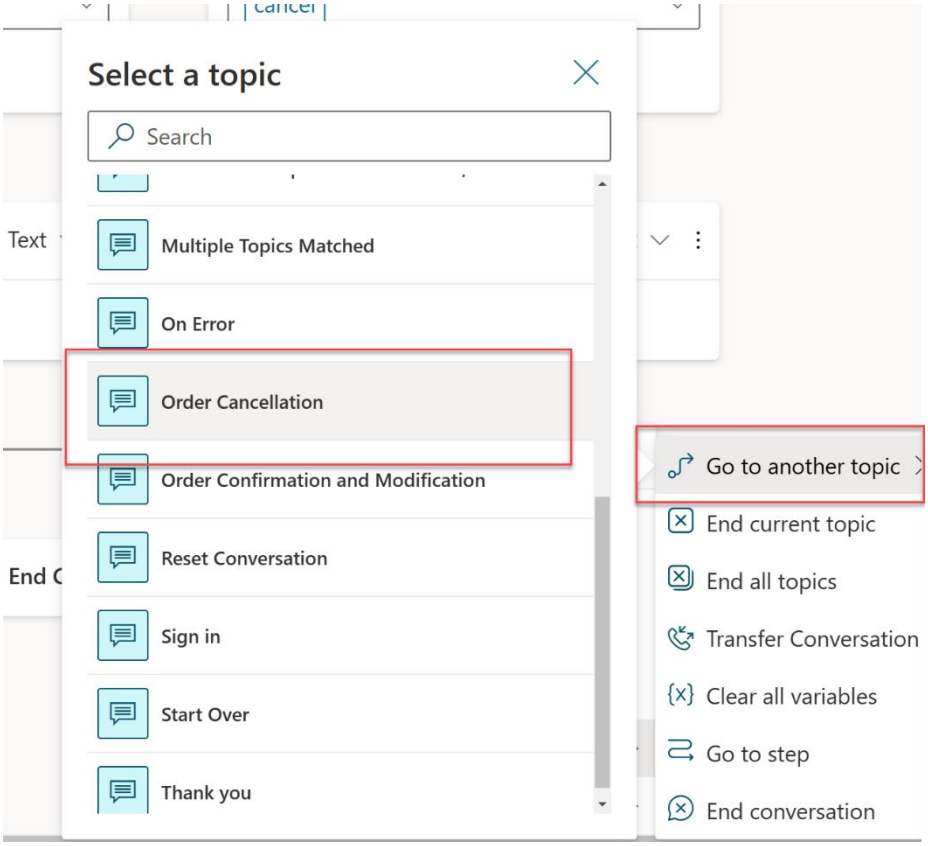
Now you are familiar with the basics of the topic management functions, **let's practice using 'Go to Another Topic' for the next task**. The 'Go to Another Topic' is useful to be able to leverage other topics from conditions based on what the user has asked for in the dialog.

Task 1: Using 'Go to another Topic' node

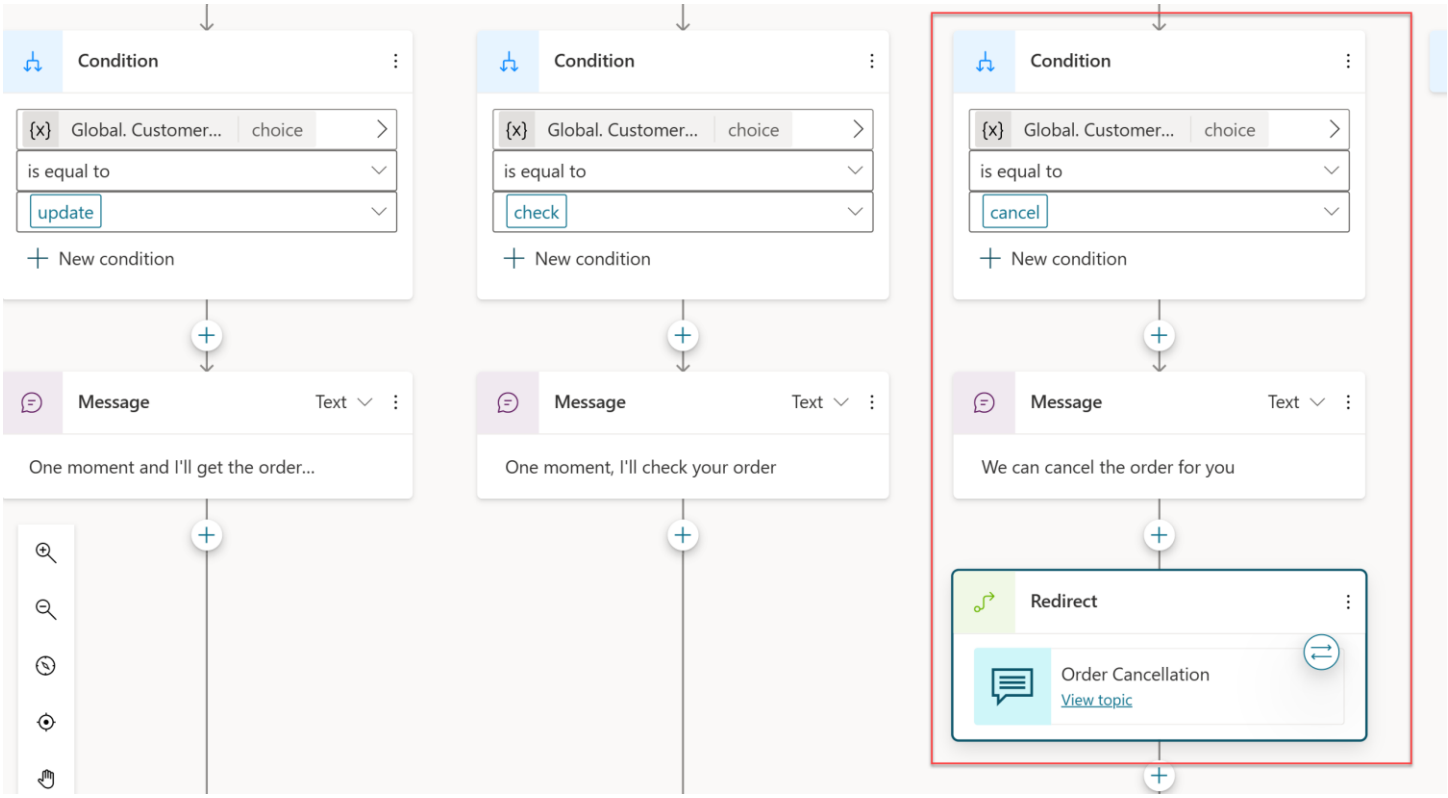
1. In your topic, ensure you have completed the last task and have a condition node in your authoring canvas.
2. Navigate back to the 'Topics' page where you see all of your Topics. Create a brand-new Topic called 'Order Cancellation'. Do not add any Trigger Phases currently and add a single message node acknowledging the cancellation. *(Normally the topic would be fully built out but for the purposes of this lab, we will not complete this, however you are welcome to!)*



3. Now go back to your original topic you created and within the Condition branch that had the Condition as 'Cancel' create a new node underneath and click 'Topic Management', then in the flyout menu, click 'Go to another topic' and find your 'Order Cancellation' topic in the list and select it



4. You should now have your three conditions from the previous task, and your 'Cancel' condition should send an acknowledgement message and redirect to the new topic you just created. Go ahead and test the behavior out by saving your topic and using 'Test your Bot'



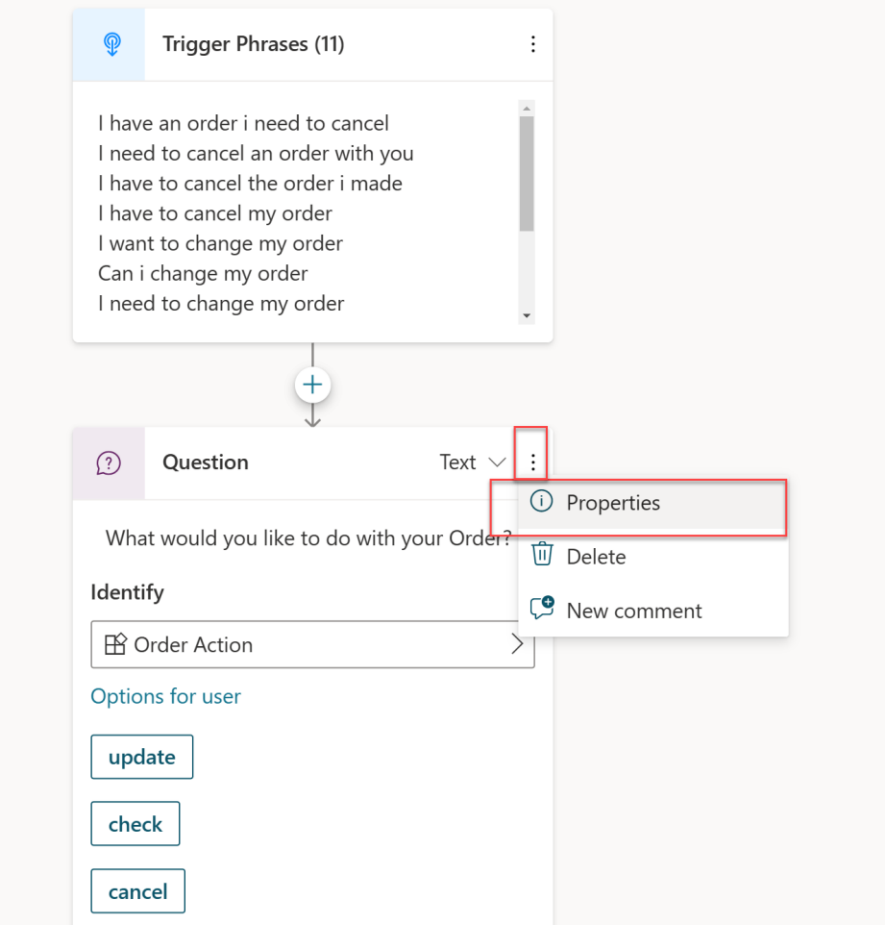
Congratulations! You are now familiar with the available actions under Topic Management. It would be useful to review some of the other options under Topic Management before continuing. This is not essential to move on to the next exercise.

Exercise 5: Question Node Behavior

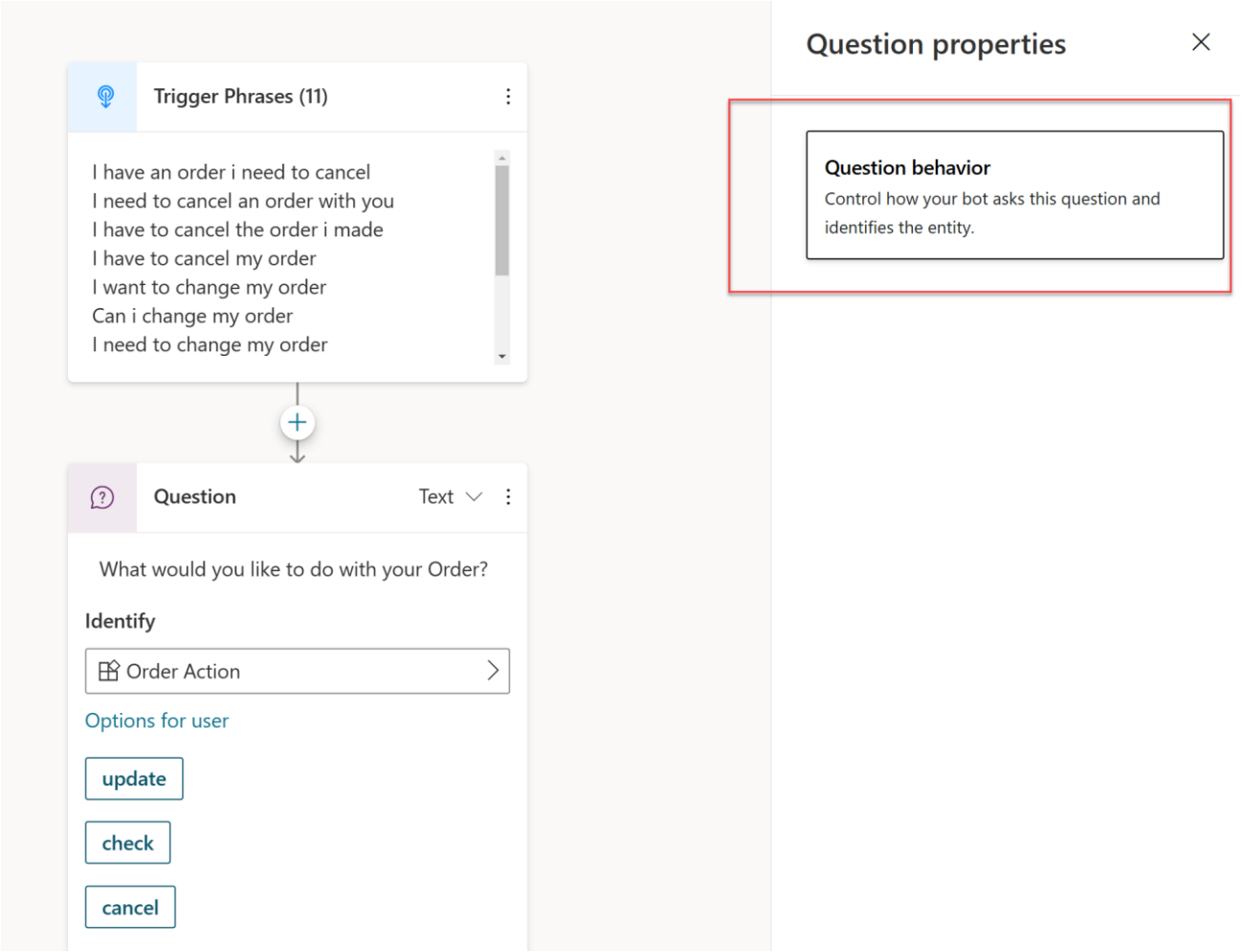
Task 1: Understand Question Node Behavior Options

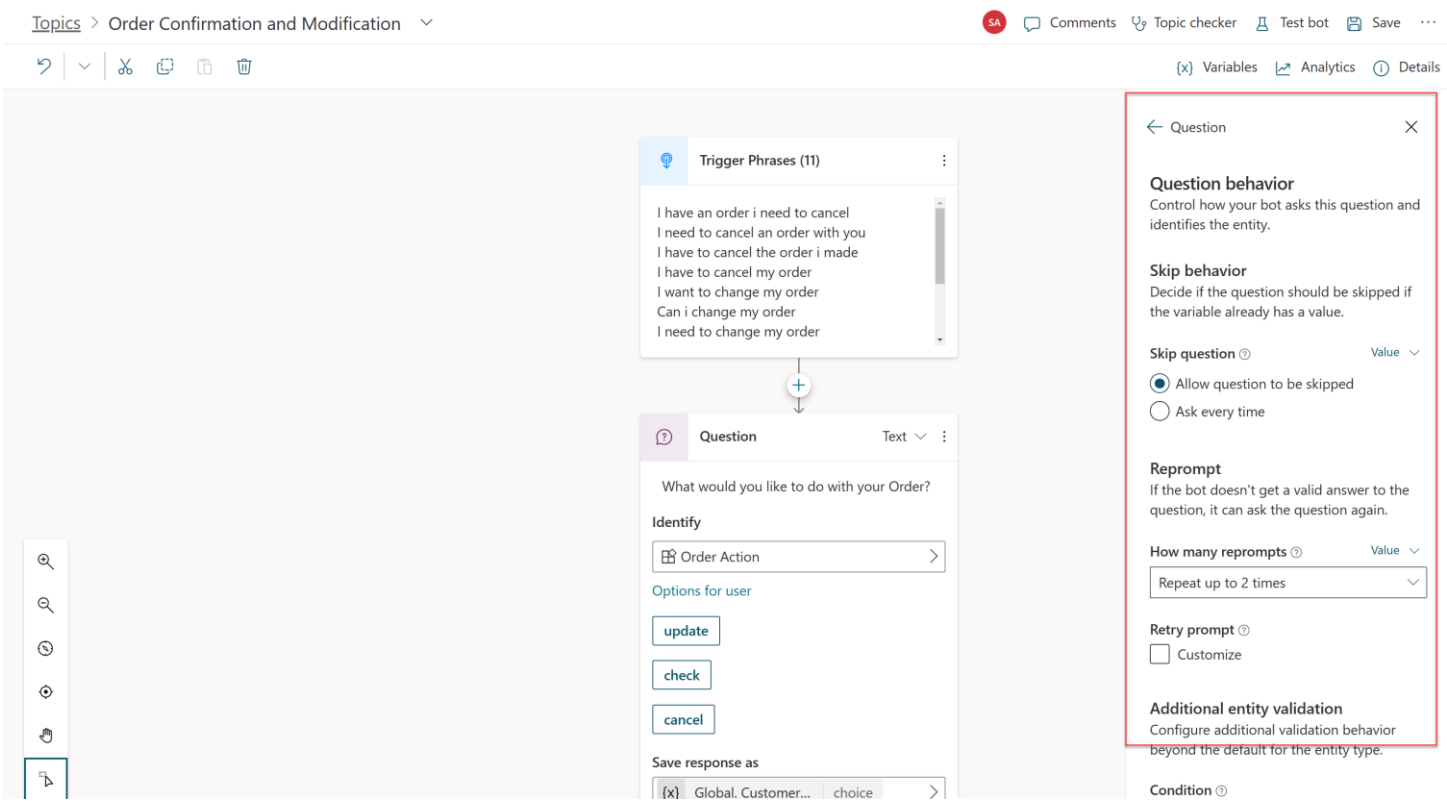
The basics of the Question Node were covered in Lab 2 and was built upon in this lab using Entities and Slot Filling. In addition to being able to store a user's response, the Question Node has additional behavior options that can be configured. One of these options is being able to skip asking the question if the variable it is linked to already contains a value. You saw this in action in a previous exercise where the question was skipped when asked to check an Order. This is because using entities and slot filling, it allowed Power Virtual Agent to retrieve the data from the sentence the user asked it and store it within the variable. Once the question node was reached by Power Virtual Agents, it already had data, so it did not need to ask it again. This is efficient and also expected user or customer experience when talking to bots.

1. Click on your Question Node in your main topic that you have been working on in this lab and click the vertical extended menu on the question node, as shown in the screenshot below, and clicking 'Properties' to open the right-hand side panel.



2. Select **'Question behavior'** on the right-hand side panel that appears.





The question node has several configurable options to be able to better identify what the user's response is to the question you are asking. This is particularly important when developing conversational applications because regardless of the AI behind the scenes managing the natural language responses, a user may answer the question with something unexpected or unidentifiable. By being able to handle the behavior of the bot in those circumstances, it helps to provide an improved customer experience. An example of this is also the same in real life, when you ask a question to another person, like a team member, and they don't understand the question. It is important to rephrase or act differently than just repeat the same question that was not originally understood for the best experience and conversation.

Let's briefly review the question behavior controls that you have available from top to bottom from opening the **Question Behavior** property window:

- **Skip Behavior: Skip Question:** This option allows a bot author to skip the question if the variable already has a value. The variable in the question could have a value from being set somewhere else in the topic, another topic or through slot filling and using entities. This behavior allows the question to be skipped, or even if the variable has a value to ask the question anyway. Additional options here are using PowerFX to create a condition and if that condition is true, to skip the question.
- **Re-prompt: How many re-prompts:** You can configure the behavior to repeat the question a specific number of times and you can select from the dropdown to not repeat, repeat once, or repeat up to 2 times. In the same way as the skip question, you can also use PowerFX to set the condition for this behavior to occur. You can modify the Retry Prompt, which will only occur if you have retries selected to repeat the question, and when selected you can add a different message to reword the question, and even add message validations here to make it sound more natural and be more helpful to the customer or user.

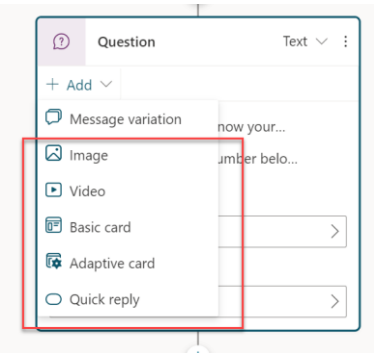
- **Additional Entity Validation**: **Condition**: This behavior is important if you have specific conditions to validate if an entity is valid for it to be slot filled and is dependent on the entity type. You also have the same prompt behavior to change the prompt if the conditions are not met to encourage the user to provide a different input.
- **No Valid Entity Found: Action if no entity found:** If no entity is found, rather than skip the question you can specify the behavior such as leave the variable empty, set the variable to something specific or dynamic (using PowerFx) or even call the escalate system topic.
- **Interruptions:** You can configure if a customer should be able to switch to a different topic than the current topic that the question node is in. This is useful if a customer is likely to answer a question with another question and to continue the conversation, without having to handle all exceptions within a single question node.

Now you are aware of the core functionality of the Question node and associated behavior. Let's take a deeper look at the Rich Text Responses in both the message and question nodes.

Exercise 5: Rich Text Options for Message and Question Nodes

Task 1: Understanding Rich Text Options in Message and Question Nodes

Power Virtual Agents preview has several extended capabilities to author bots and provide positive conversational experiences to customers. One of the core features is the Rich Text Authoring capabilities available on both Message and **Question** node.



Let's take a brief look at the types of options you have available.

A. **Image:** You can add an image which will be displayed on the card. Add the URL of the image and optionally a title.

B. **Video**: You can add a video URL which needs to be a publicly available MP4 or a YouTube video URL

C. **Basic Cards:** Are simple cards that provide adaptive cards like visuals however require standard input such as Title, Messages and the ability to add 'Buttons' with basic actions.

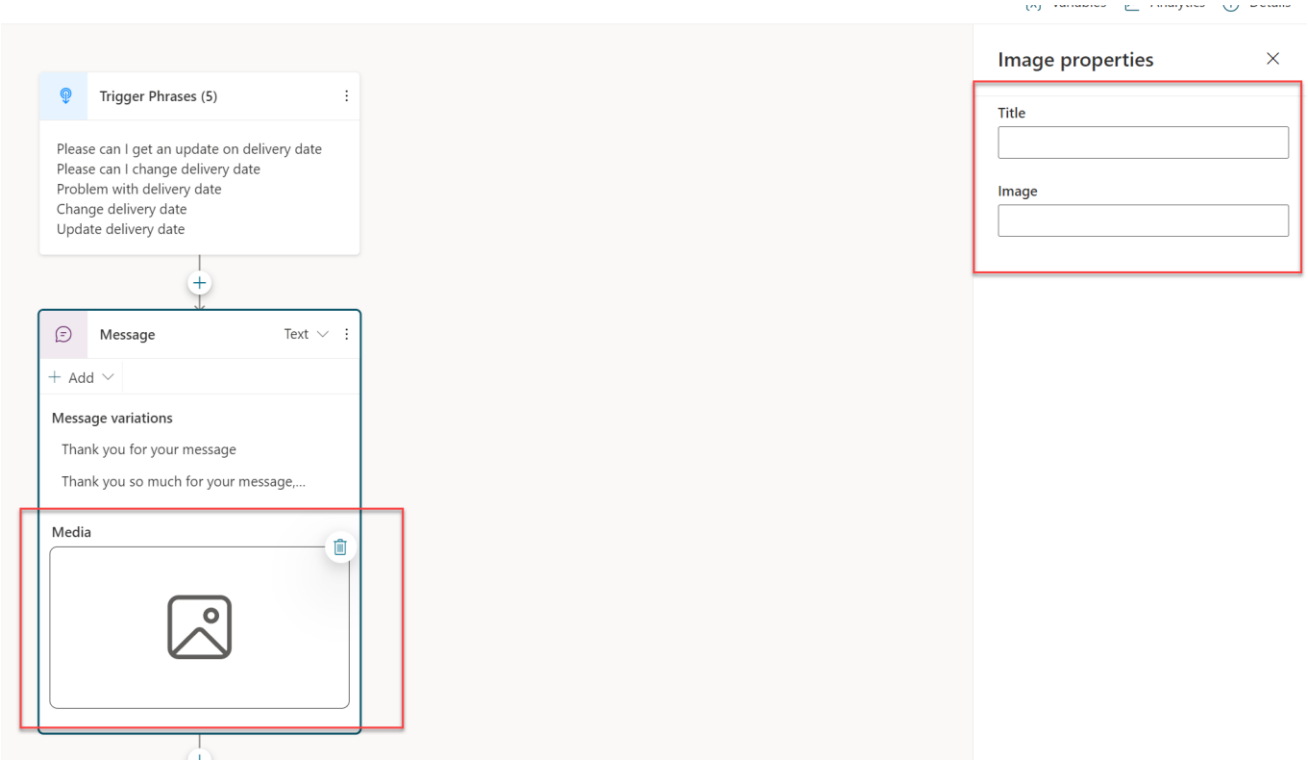
D. **Adaptive Cards:** Can be added which are platform agnostic cards designed to be flexible to suit the need at the time, including requesting action, displaying information so it is displayed with emphasis on specific information or more. Power Virtual Agents preview supports Adaptive Cards v.1.3 at the time of writing these labs.

E. **Quick Replies** allow users to select from specific options rather than have to type the response out in text based scenarios. They are optional so a user can still type or speak their own response and should be used to provide common suggestions or to help give the user ideas on the type of information being asked.

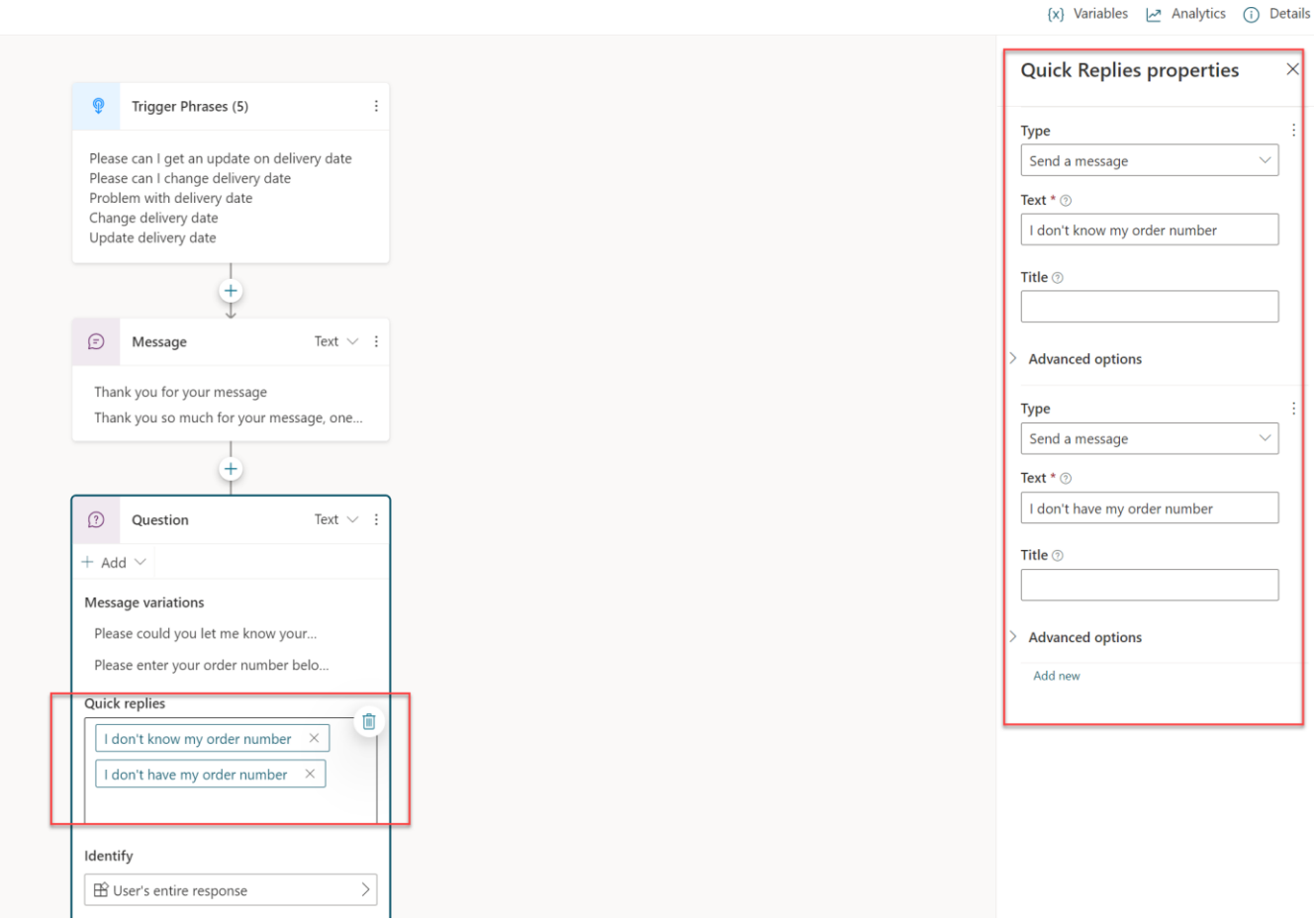
Now let's configure a few of these options so you can become familiar with their behavior.

Task 2: Use Rich Text Capabilities in a Message Node

1. In the topic you have open from the previous task, create a new Message node, click 'Add' and select **'Image'**. Click on the image block that has appeared, shown below, and you will see the Title and Image text boxes appear. You need to enter the Image URL to a publicly available image, and you can name the image using the Title field.



2. Repeat the previous step, and instead of Image, select Quick Replies and you will see a 'Quick replies' box appear within the message node. Enter at least 2 options to your question and you will begin to see options appear on the right-hand side for those quick replies to configure the associated message behavior, such as send message text or open a URL, text, and title. **(Note: The options 'Make a Call' and 'Send a hidden message' are currently not supposed in preview)**



Next Steps: Take some time to repeat this process with all the different Rich Response types to become familiar with the different properties before moving on to the next task. There is extended documentation to each of those response types in the **Microsoft Docs Website.**

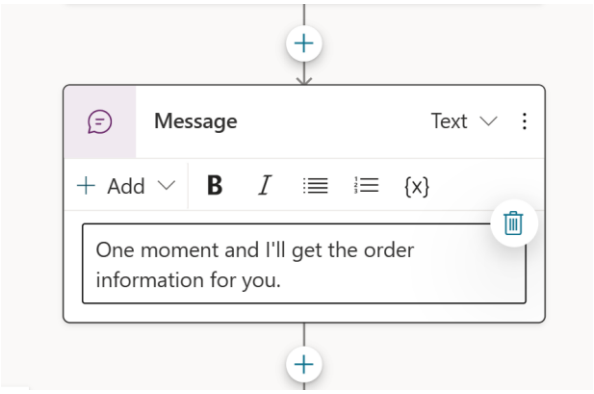
Exercise 5: Message Variations

Task 1: Using Message Variations

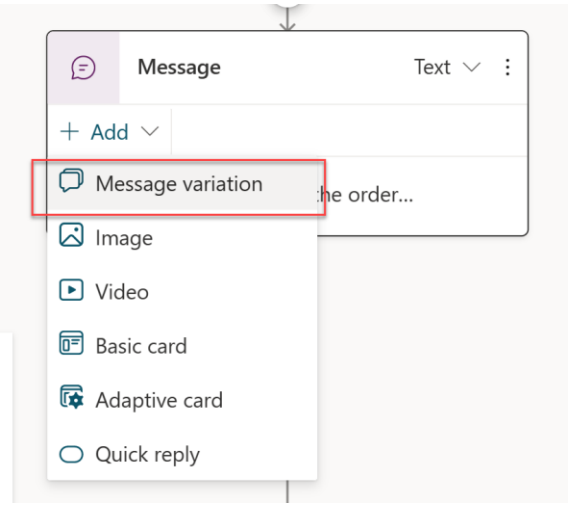
The 'Message' node is used to display a message to the user and the fundamentals were covered in the previous Lab (Lab 02). Messages also support Message Variations. Message Variations allow you to add up to 15 messages in a single message node so that when the bot is triggered, it randomly selects a message variation. Message variations allow a bot author to add different styles of sentence in the same message node, where at runtime Power Virtual Agents will randomly select one message variation when the topic is triggered. This provides authors with the ability to create natural sounding bots and customers to have a more natural sounding experience when interacting with them over time.

Pro Tip! Since message variations are selected randomly when a topic is triggered, you can add multiple versions of the same message if you want to provide an experience which leans on a certain style, whilst providing smaller degrees of probability, but still offer differentiation on some occasions.

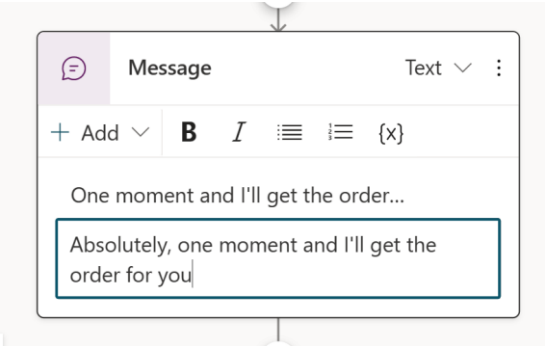
Let's try adding a Message Variation on an existing Message from a previous task.



1. Open your topic you have been working on in this Lab. Select your Message Node that you want to add a Message Variation to, such as the one shown above in the Condition Branch.
2. On the Message Node, click 'Add' as shown below



3. Add at least one message variation of your choosing, or use the examples below, to see how message variations stack in the message node:



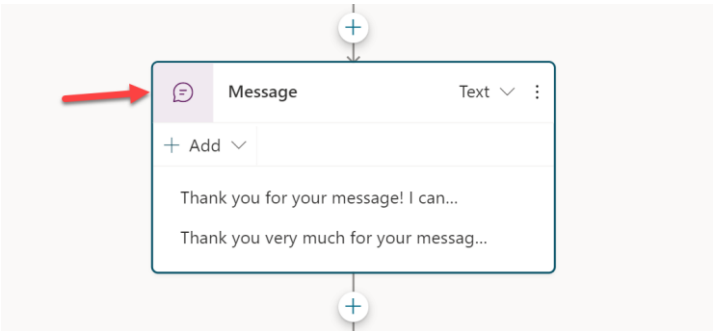
Congratulations you have completed the basics of using the message node. Currently there are no properties to modify the message node. Go ahead and test out your message variations using 'Test your bot' and trigger the condition multiple times to see it working!

Exercise 5: Productivity Features

Task 1: Productivity Features

Power Virtual Agents now can use familiar operations such as copy and paste within the authoring canvas. This can help speed up authoring experiences and make it easier to build logic for more complex topics. Let's try this out in the topic you have opened from the previous task.

1. Select a node within your open topic, so the blue outline appears, as shown below in the screenshot on the next page
2. On the productivity bar near the top left of the authoring canvas, select the 'copy' icon



3. There are two options now you have 'copied' a node:
    - Either don't have any node selected
    - You have a node selected

On the productivity bar, select the 'paste' icon. If you did not have a node selected, the copied node would have been pasted at the bottom of the dialog tree in the canvas. If you did have a node selected, it would have been pasted underneath that node and seamlessly added into the dialog tree.

Congratulations! You have completed the basics on the latest authoring capabilities in Power Virtual Agents! You should now be familiar with the core authoring nodes and types of responses you can use to interact with your customers and users.

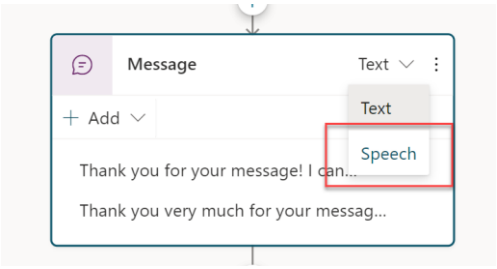
Exercise 6: Speech Authoring

Now within the web authoring canvas in Power Virtual Agent, using SSML tags in Message and Question nodes to be able to extend the behavior when using Power Virtual Agent for speech enabled bots. It is critical to understand that Power Virtual Agents can be used for both text authoring and speech authoring. By default, on voice-enabled channels, the message text entered in the message node will be used for both text display and voice. You can override this behavior by providing different behavior for text and speech. Sometimes this is the case where you may want to provide more emphasis on certain areas of a sentence, or on an image message, you would want to provide an alternative description that can easily be read out.

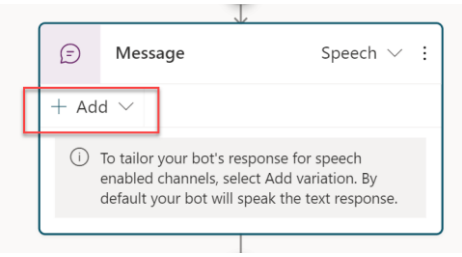
Let's review the options for adding SSML within a message node. This task will walk you through the steps to add SSML, however does not test or show the bot using voice at this time.

Task 1: Becoming familiar with adding SSML

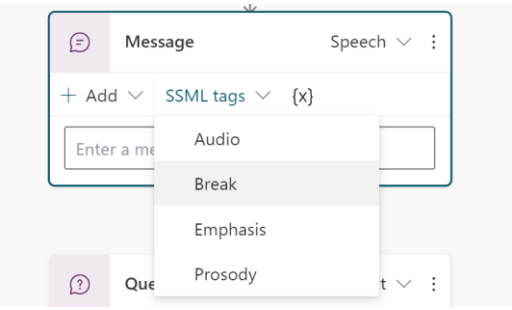
1. Within your topic, create a new Message node.
2. On the message, click the **'Text'** dropdown and select **'Speech'** as shown in the screenshot below:



3. You will see a message appear as shown in the screenshot below. Click the **'Add'** to add a new Message Variation as we covered in Lab 01. Add what you would want the bot to say, and you can add multiple message variations for speech in the same way you can add them for text.



4. When using SSML, you can even configure how the text is converted to synthesized speech to ensure it sounds like natural speech. When SSML is being used you can use SSML tags, such as Audio, Break, Emphasis and Prosody to change the behavior of how your sentence is spoken. In this lab we are not creating a speech enabled bot and only reviewing the available options, but you can review the options below for the purpose of this lab.



- **Audio:** Add prerecorded audio
- **Break:** Insert pauses or breaks between words
- **Emphasis:** Add levels of stress to words to phrases
- **Prosody:** Specify changes to pitch, contour, range, rate and volume.

Congratulations! You have completed the exercise to cover extended functionality now available within the web authoring canvas for Power Virtual Agents. The following section is recommended, but optional.

Exercise 7: Code View and PowerFX

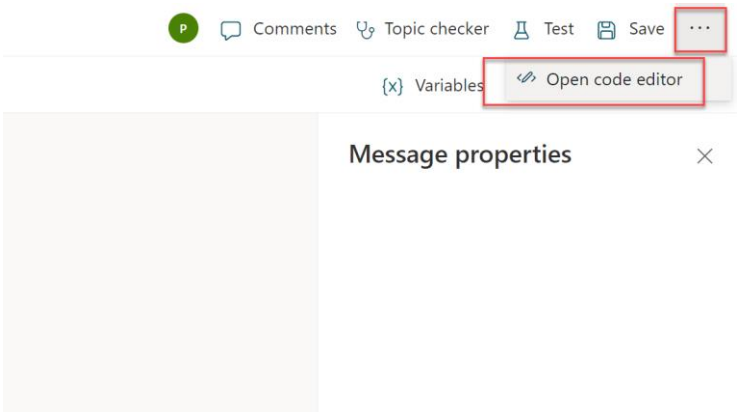
Now you understand the fundamentals of authoring in Power Virtual Agents, let's take a brief look at some extended capabilities that can be used to configure and customize the bot experience even further. The two items this section will review are a brief look at code view for pro developers and PowerFX (for Power Platform makers and also pro developers).

Power Virtual Agents now has the capability to view the code behind a topic. This is incredibly useful for pro code developers where they can view and edit the syntax directly within the web browser, and when saved this is immediately visible in the graphical authoring canvas. This makes creating topics faster and easier, giving the capability to developers to extend beyond the authoring canvas if required.

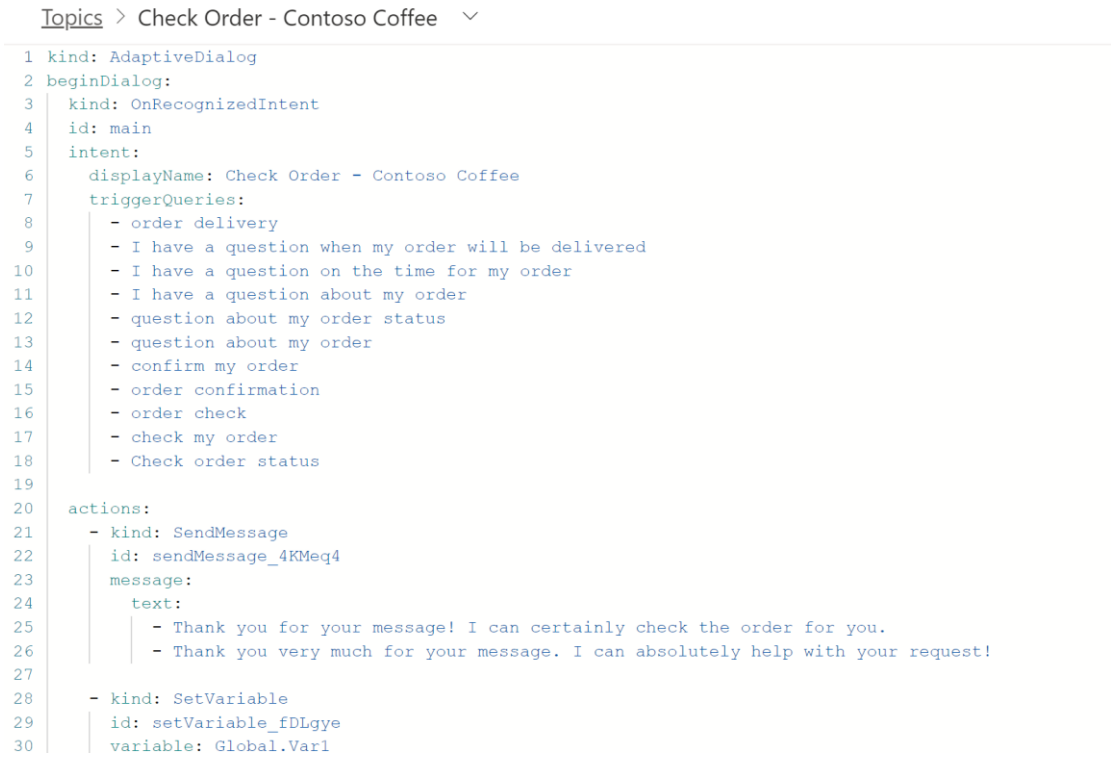
Let's take a brief look how a user would access this code editor.

Task 1: Accessing the code editor

1. Open a topic you have been working on previously in this lab.
2. At the top right of the topic, next to the save icon, click the extended menu '... and click 'Open Code Editor' as shown below:



3. The code editor should open, and you can now view your dialog in the code view.



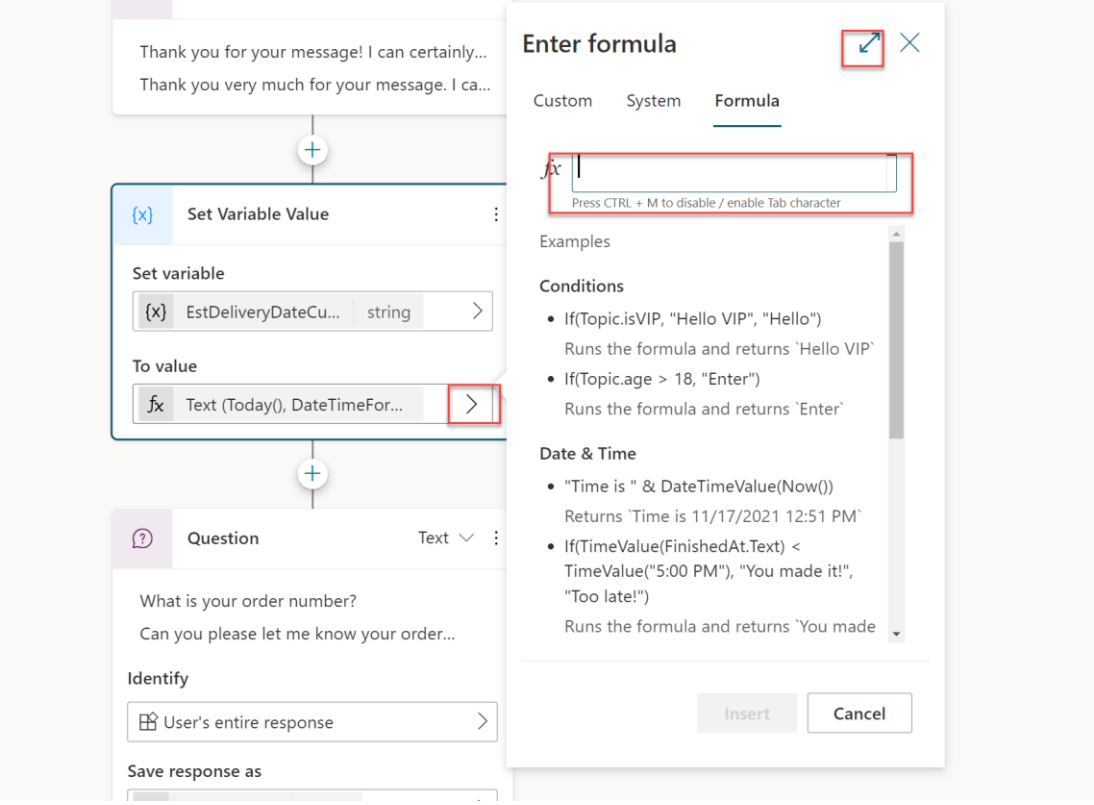
Task 2: Using PowerFx across Power Virtual Agent Unified Authoring

PowerFx is now available in the Preview release of Power Virtual Agent to be able to add functions, just like makers currently use in Power Apps Canvas Apps, within the Power Virtual Agent authoring canvas. PowerFX can be used in message and question nodes, when using the variable node and in other areas like question behavior and in adaptive cards. This gives both authors the capability to be able to have greater control over the data that is displayed to customers and users within the conversational interface and allows common operations to be performed in the runtime of Power Virtual Agents.

Let's walk through a basic scenario using PowerFx within a variable and displaying the value to the user.

Task 1: Use PowerFx to modify how the date is displayed

1. Open a Topic you have been working on during these labs
2. Create a new **'Set Variable'** node and create a new variable, as covered in the previous tasks.
3. On the **'Set Variable'** node, under **'To Value'** click the fly out menu and click **'Formula'** to open the formula panel. Click the expand icon, as highlighted below in the screenshot:

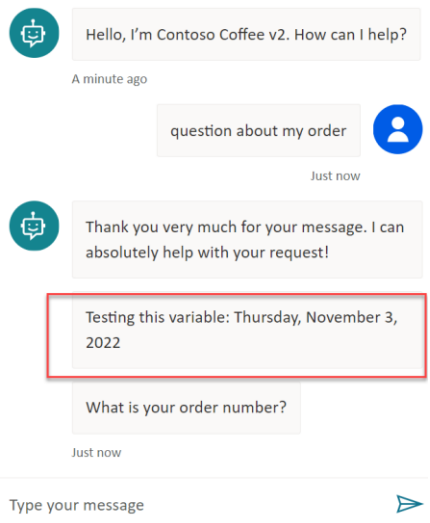


4. In the formula bar, type the following function and click **Insert**

```
Text(Now(), DateTimeFormat.LongDate)
```

This function takes today's date and time, that has a specific date and time format (9/01/2022 08:00 AM) and stores it as a datetime value type, and it converts it using the DateTimeFormat function to a be a long date (e.g. Monday the 1st of September) and displays it as text. This is particularly important if you want to display simple date formats that are customer or user friendly or just store the date as a string in text format.

5. For demo purposes, you can add a new message node, and using the variable feature, add the variable name into the message node to display the output of the function in the bot conversation so you can see the output when testing your bot. Test your bot by opening the testing pane, trigger the topic and follow the topic prompts to reach your message node, as shown below:



Congratulations you have successfully worked through all of the labs on core authoring features introduced in Power Virtual Agents Unified Authoring!

Summary

Thank you for completing Lab 03 'Authoring 101 in Unified Authoring. You have successfully:

- Understood the latest updates to the message and question nodes
- Used the new rich text options available including image and adaptive cards
- Reviewed the behavior of the system topics and how you can use them in conversation flow
- Understood how variables can be used and the different types including system, topic and global
- Reviewed how to add speech to your bot for voice enabled bot using SSML options
- Used PowerFx to modify the date format and display this in a message to the user
- How to change between graphical and code views directly in the authoring canvas

Lab survey

We would appreciate your feedback on Power Virtual Agents and on this hands-on-lab, such as the quality of documentation and the usefulness of the learning experience.

Please use the survey at https://aka.ms/pvaiadpreviewsurvey to share your feedback.

You may provide feedback for each module as you complete it or at the end once you've completed all the modules. Thank you!

Terms of Use

By using this document, in whole or in part, you agree to the following terms:

Notice

Information and views expressed in this document, including (without limitation) URL and other Internet Web site references, may change without notice. Examples depicted herein, if any, are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred. This document does not provide you with any legal rights to any intellectual property in any Microsoft product.

Use Limitations

Copying or reproduction, in whole or in part, of this document to any other server or location for further reproduction or redistribution is expressly prohibited. Microsoft provides you with this document for purposes of obtaining your suggestions, comments, input, ideas, or know-how, in any form, ("Feedback") and to provide you with a learning experience. You may use this document only to evaluate its content and provide feedback to Microsoft. You may not use this document for any other purpose. You may not modify, copy, distribute, transmit, display, perform, reproduce, publish, license, create derivative works from, transfer, or sell this document or any portion thereof. You may copy and use this document for your internal, reference purposes only.

Feedback

If you give Microsoft any Feedback about this document or the subject matter herein (including, without limitation, any technology, features, functionality, and/or concepts), you give to Microsoft, without charge, the right to use, share, and freely commercialize Feedback in any way and for any purpose. You also give third parties, without charge, the right to use, or interface with, any Microsoft products or services that include the Feedback. You represent and warrant that you own or otherwise control all rights to such Feedback and that no such Feedback is subject to any third-party rights.

DISCLAIMERS

CERTAIN SOFTWARE, TECHNOLOGY, PRODUCTS, FEATURES, AND FUNCTIONALITY (COLLECTIVELY "CONCEPTS"), INCLUDING POTENTIAL NEW CONCEPTS, REFERENCED IN THIS DOCUMENT ARE IN A SIMULATED ENVIRONMENT WITHOUT COMPLEX SET-UP OR INSTALLATION AND ARE INTENDED FOR FEEDBACK AND TRAINING PURPOSES ONLY. THE CONCEPTS REPRESENTED IN THIS DOCUMENT MAY NOT REPRESENT FULL FEATURE CONCEPTS AND MAY NOT WORK THE WAY A FINAL VERSION MAY WORK. MICROSOFT ALSO MAY NOT RELEASE A FINAL VERSION OF SUCH CONCEPTS. YOUR EXPERIENCE WITH USING SUCH CONCEPTS IN A PHYSICAL ENVIRONMENT MAY ALSO BE DIFFERENT.

THIS DOCUMENT, AND THE CONCEPTS AND TRAINING PROVIDED HEREIN, IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, WHETHER EXPRESS, IMPLIED, OR STATUTORY, INCLUDING (WITHOUT LIMITATION) THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, AND NONINFRINGEMENT. MICROSOFT DOES NOT MAKE ANY ASSURANCES OR REPRESENTATIONS WITH REGARD TO THE ACCURACY OF THE RESULTS, THE OUTPUT THAT DERIVES FROM USE OF THIS DOCUMENT OR THE CONCEPTS, OR THE SUITABILITY OF THE CONCEPTS OR INFORMATION CONTAINED IN THIS DOCUMENT FOR ANY PURPOSE.

MICROSOFT POWER VIRTUAL AGENTS (1) IS NOT INTENDED OR MADE AVAILABLE AS A MEDICAL DEVICE FOR THE DIAGNOSIS OF DISEASE OR OTHER CONDITIONS, OR IN THE CURE, MITIGATION, TREATMENT OR PREVENTION OF DISEASE, OR OTHERWISE TO BE USED AS A COMPONENT OF ANY CLINICAL OFFERING OR PRODUCT, AND NO LICENSE OR RIGHT IS GRANTED TO USE MICROSOFT POWER VIRTUAL AGENTS FOR SUCH PURPOSES, (2) IS NOT DESIGNED OR INTENDED TO BE A SUBSTITUTE FOR PROFESSIONAL MEDICAL ADVICE, DIAGNOSIS, TREATMENT, OR JUDGMENT AND SHOULD NOT BE USED AS A SUBSTITUTE FOR, OR TO REPLACE, PROFESSIONAL MEDICAL ADVICE, DIAGNOSIS, TREATMENT, OR JUDGMENT, AND (3) SHOULD NOT BE USED FOR EMERGENCIES AND DOES NOT SUPPORT EMERGENCY CALLS. ANY CHATBOT YOU CREATE USING MICROSOFT POWER VIRTUAL AGENTS IS YOUR OWN PRODUCT OR SERVICE, SEPARATE AND APART FROM MICROSOFT POWER VIRTUAL AGENTS. YOU ARE SOLELY RESPONSIBLE FOR THE DESIGN, DEVELOPMENT, AND IMPLEMENTATION OF YOUR CHATBOT (INCLUDING INCORPORATION OF IT INTO ANY PRODUCT OR SERVICE INTENDED FOR MEDICAL OR CLINICAL USE) AND FOR EXPLICITLY PROVIDING END USERS WITH APPROPRIATE WARNINGS AND DISCLAIMERS PERTAINING TO USE OF YOUR CHATBOT. YOU ARE SOLELY RESPONSIBLE FOR ANY PERSONAL INJURY OR DEATH THAT MAY OCCUR AS A RESULT OF YOUR CHATBOT OR YOUR USE OF MICROSOFT POWER VIRTUAL AGENTS IN CONNECTION WITH YOUR CHATBOT, INCLUDING (WITHOUT LIMITATION) ANY SUCH INJURIES TO END USERS.