Microsoft

Proudly supported by SAP

edunet foundation

Lab Manual

Working with Prompts

Disclaimer: The content is curated from online/offline resources and used for educational purpose only

Disclaimer: The content is curated from online/offline resources and used for educational purpose only

Table of Contents

Microsoft Copilot Studio overview

Introduction

Microsoft Copilot Studio is a graphical, low-code tool for both creating a custom Copilot—including building automation with Power Automate—and extending a Copilot for Microsoft 365 with your own enterprise data and scenarios.

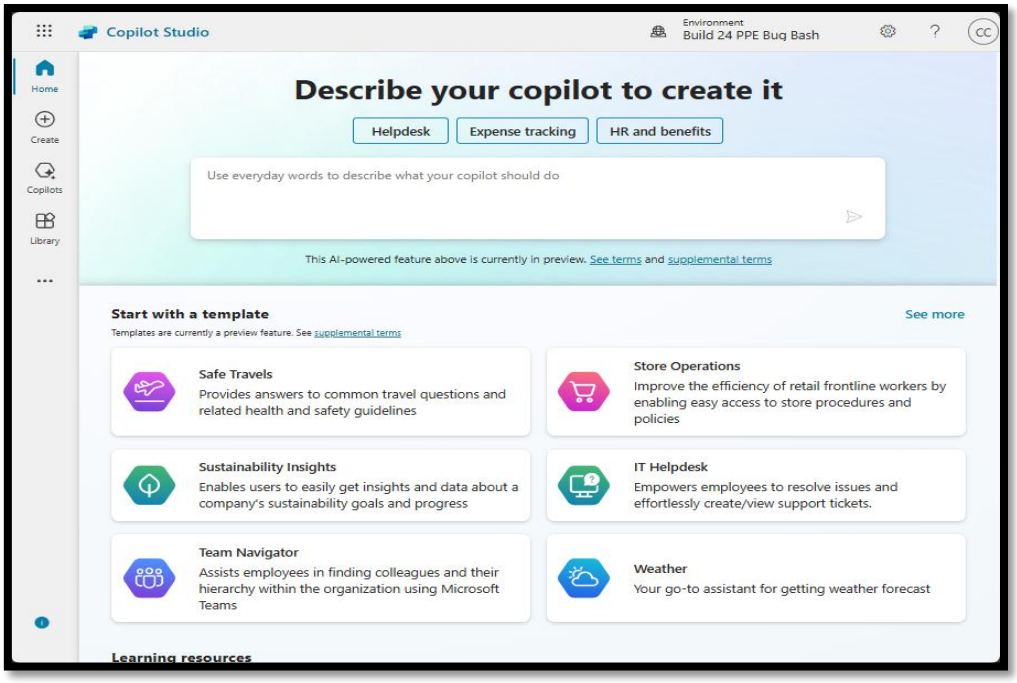

Fig: Copilot Studio Overview

One of the standout features of Copilot Studio is its ability to connect to other data sources using either prebuilt or custom plugins. This flexibility enables users to create and orchestrate sophisticated logic, ensuring that their copilot experiences are both powerful and intuitive.

The platform's low-code experience puts the power of AI at the user's fingertips, making it accessible even to those without extensive technical backgrounds.

A copilot is an AI-powered conversational interface based on large language models (LLMs) and additional sources of knowledge. It's a powerful AI companion that can handle a range of requests, from providing simple responses to common questions to resolving issues requiring complex conversations.

Copilots can engage with customers and employees in multiple languages across websites, mobile apps, Facebook, Microsoft Teams, or any channel supported by the Azure Bot Service.

You can easily create copilots in Copilot Studio without the need for data scientists or developers. Some of the ways you might use copilots include:

- Sales help and support issues.
- Opening hours and store information.
- Employee health and vacation benefits.
- Public health tracking information.
- Common employee questions for businesses.

Copilot Studio is available as both a standalone web app, and as a discrete app within Teams. Most of the functionality between the two is the same. However, there might be different reasons to choose one version or the other based on the ways you want to use Copilot Studio.

Create a custom copilot

Copilot Studio makes creating copilots easy. You only must describe the copilot you want in plain language to create it. Tell Copilot Studio what specific instructions, triggers, knowledge sources, and actions you want for your copilot. Then test your copilot before you deploy. Publish your copilot when you are ready across multiple channels.

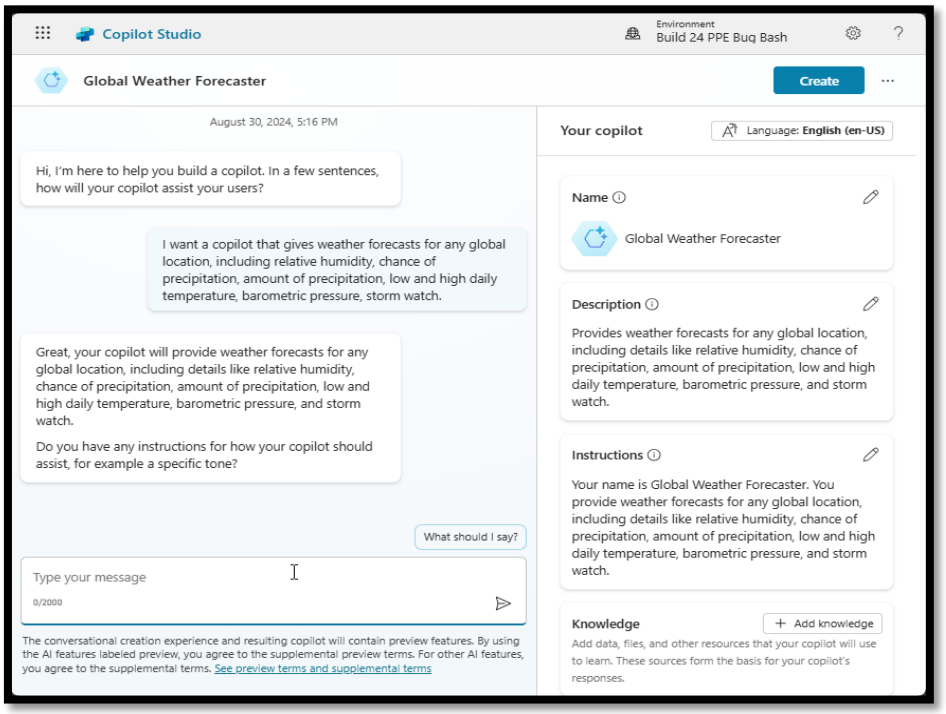

Fig: Custom Copilot

Copilot topics and actions with generative AI

How does generative mode work?

Using generative AI to determine how your copilot responds can make the conversation more natural and fluid for the users.

When a user sends a message, your copilot selects one or more actions or topics to prepare its response. Multiple factors determine the selection. The most important factor is the description of the topics and actions. Other factors include the name of a topic or actions, any input or output parameters, and their names and descriptions. Descriptions make it possible for your copilot to be more accurate when it associates a user's intent with actions and topics.

In generative mode, a copilot can select multiple actions or topics at once, to handle multi-intent queries. Once actions and topics are selected, the copilot generates a plan that determines their execution order.

When you test a copilot that uses generative mode in Copilot Studio, you can open the conversation map to follow the execution of the plan.

Turn on generative mode for a copilot

1. Open your copilot in Copilot Studio.
2. On the top menu bar, select **Settings**.
3. Select the **Generative AI** tab.
4. Choose the **Generative** option. (The default is Classic.)
5. (Optional) Select your preferred level of strictness for content moderation.

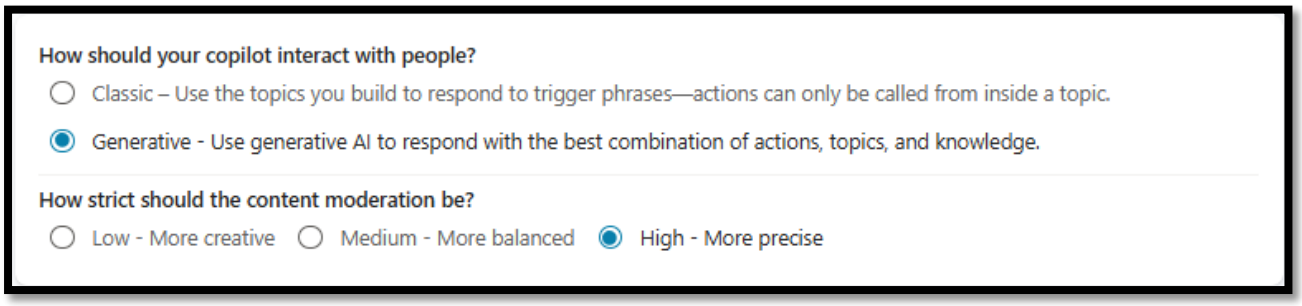

Fig: Turn on Generative Mode

6. Select **Save**.

Open the conversation map

The conversation map is available for copilots configured to respond to users in generative mode.

- To open the conversation map, select the map icon at the top of the test pane.
- To make the conversation map automatically appear when you send a query to your copilot in the test pane, use the down arrow and select **On**.

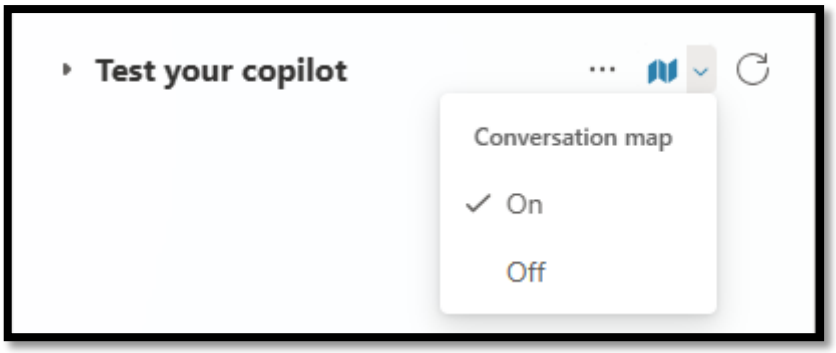

Fig: Test Your Copilot

When you ask your copilot a question in the test pane, the conversation map shows a visual representation of the plan that was generated. Here, the user asked, "what are the current weather conditions?", and you can see the action that was selected to respond and the input that still needs to be collected, along with annotations from the copilot about its decision-making.

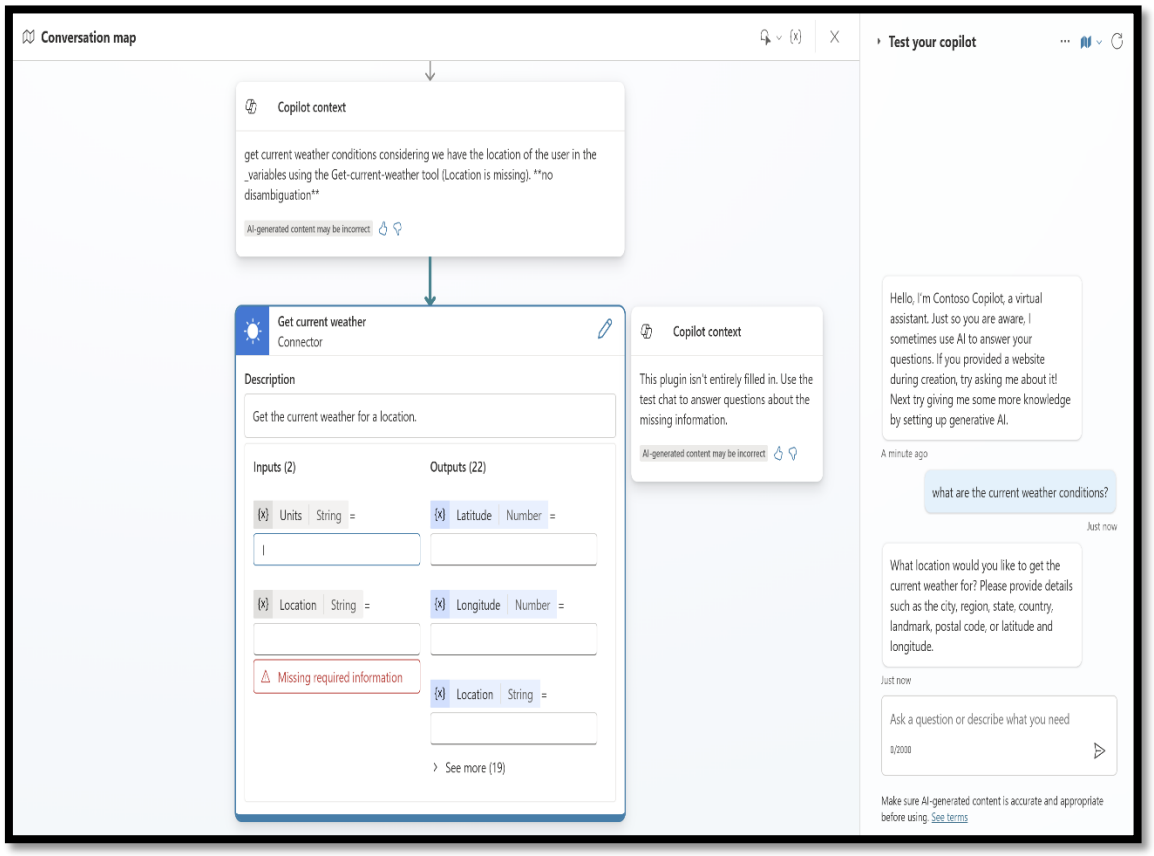

Fig: Conversation Map

When you use the conversation map, the option to track between topics is available at the top of the conversation map. With tracking enabled, when a topic is triggered as part of a plan, the nodes within the topic appear on the conversation map as they are executed, allowing you to keep track of where you are in the conversation.

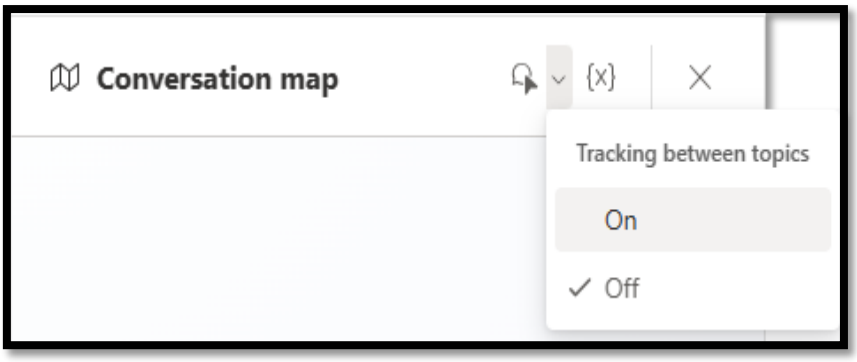

Fig: Conversation Map

Create and deploy a copilot

Microsoft Copilot Studio empowers teams to quickly and easily create powerful copilots using a guided, no-code graphical experience without needing data scientists or developers.

Create a copilot on Web app

When you create a copilot, you can describe what you want your copilot to be and do or add those details as separate pieces of information.

However, you can create your own unique copilot using a publicly available website.

1. After logging in or signing up for Copilot Studio, you land on the Home page. Select Create in the left navigation.

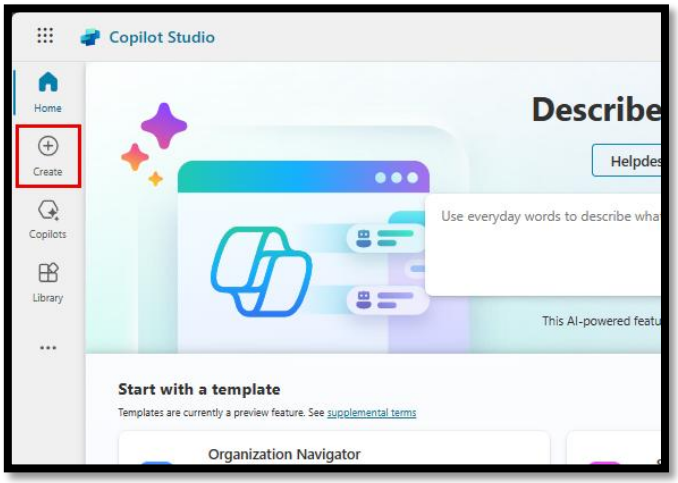

Fig: Click on Copilot

2. On the Create page, select New copilot.
3. Use the chat to describe your copilot, using the provided questions for guidance.

Keep your description simple for now, but make sure you include information about what your copilot helps users do and what conversation style and tone it uses. Copilot Studio uses your responses to fill in the details in name, description, instructions, and knowledge that define your copilot.

4. Add an image to represent your copilot.

    a. Select the copilot icon in the top bar.

    b. Select **Change icon**.

    c. Choose an image from your device. The image file must be in PNG format and less than 30 KB in size.

    d. Select **Save**.

5. After you have a name, icon, description, and instructions, your copilot is ready. Select **Create**. The **Overview** tab for your copilot appears.

You now have a copilot you can start testing! You can chat with your copilot in the **Test your copilot** chat.

Improve your copilot

Now that you have a copilot, you can start testing and improving it.

To open your copilot, in the left navigation select Copilots, then select your copilot.

You need the Overview tab for most of the actions in this section.

Test changes to your copilot

The best way to improve your copilot? Test it. Make some changes. Test it again. Repeat.

In this section, you'll test how changing your copilot's knowledge sources affects how your copilot responds to users.

1. Start by testing how your copilot currently responds in the Test your copilot chat. Ask your copilot a question. For example, ask how to create a copilot.

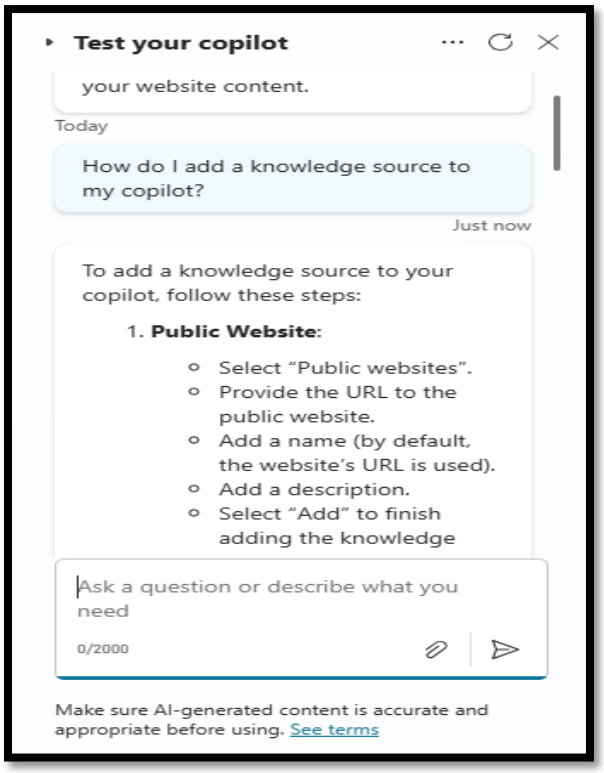

Fig: Test Your Copilot

2. In this example, the copilot's instructions are to *talk to users like a kind, patient teacher.* What if you give your copilot different instructions?

   In the **Details** card, select **Edit.** Change your copilot's instructions to use a different tone, like *talk to users like Jane Austen.*

3. Test your copilot's new instructions with another question. How has the response changed?

Change your copilot's introduction

Help your copilot make a great first impression with a new introductory message. This first message lets users know what your copilot does and encourages them to interact with your copilot.

1. In the Test your copilot chat, select on your copilot's introductory message. This opens the message in the Topics tab.

If you can't see the introductory message, select Refresh at the top of the test chat panel to restart the conversation.

2. In the Message box, select the text of the message.

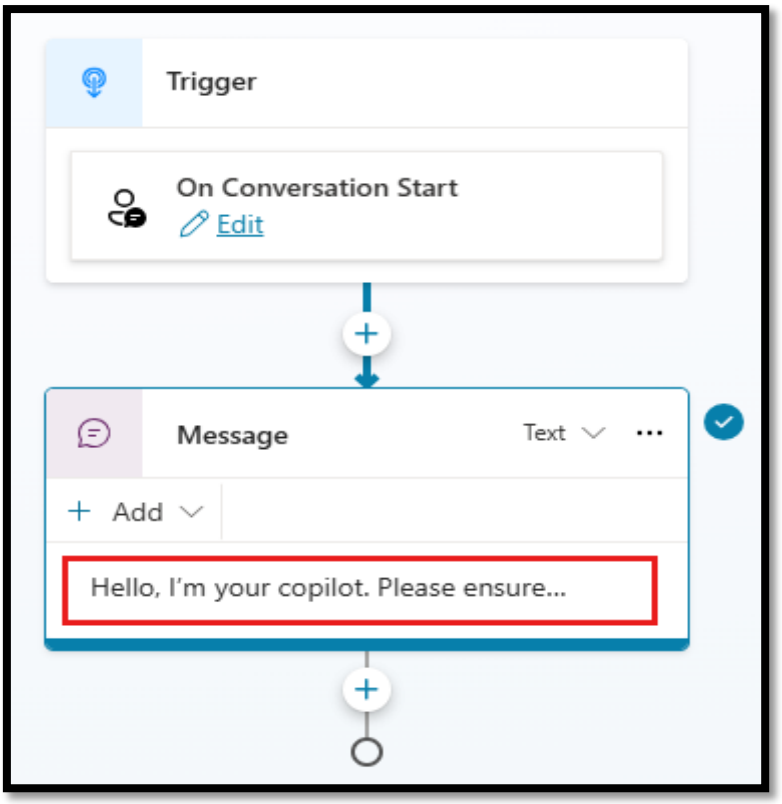

Fig: Default Messsage

3. Delete the default message and add your own. In the introduction message, your copilot should greet users, tell them what your copilot does, and tell them how to start interacting with your copilot. You can also give users an example question or prompt.
4. Select **Save**.

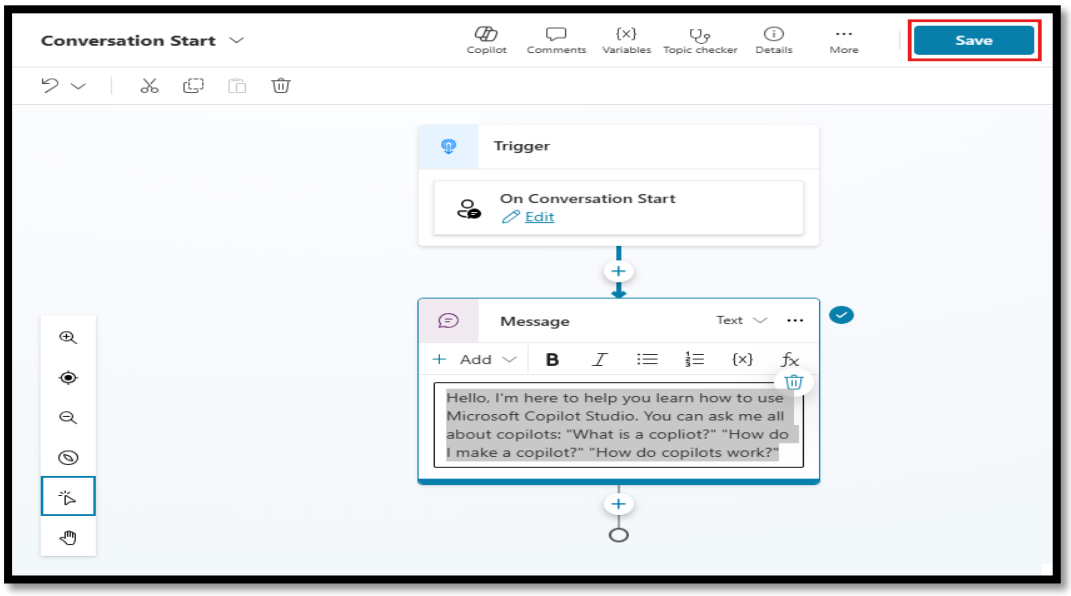

Fig: Save

Edit your copilot's basics

You can change your copilot's name, description, instructions, and knowledge sources after creating it. Remember to test your changes as you go!
To update your copilot's name, description, or instructions:

1. In the **Overview** tab's **Details** section, select **Edit**.

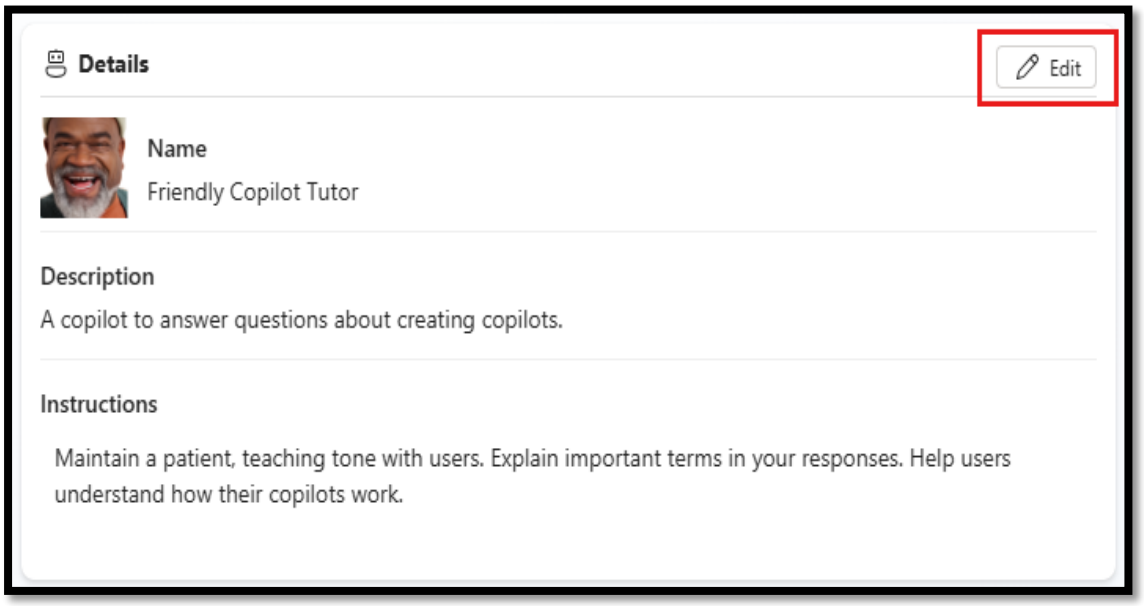

Fig: Fig: Conversation Map

2. Make your changes.
3. Select **Save**. Remember, editing the instructions changes how your copilot engages with users. Make sure to test your changes.

To add a knowledge source:

1. In the **Overview** tab's **Knowledge** section, select **Add knowledge**.

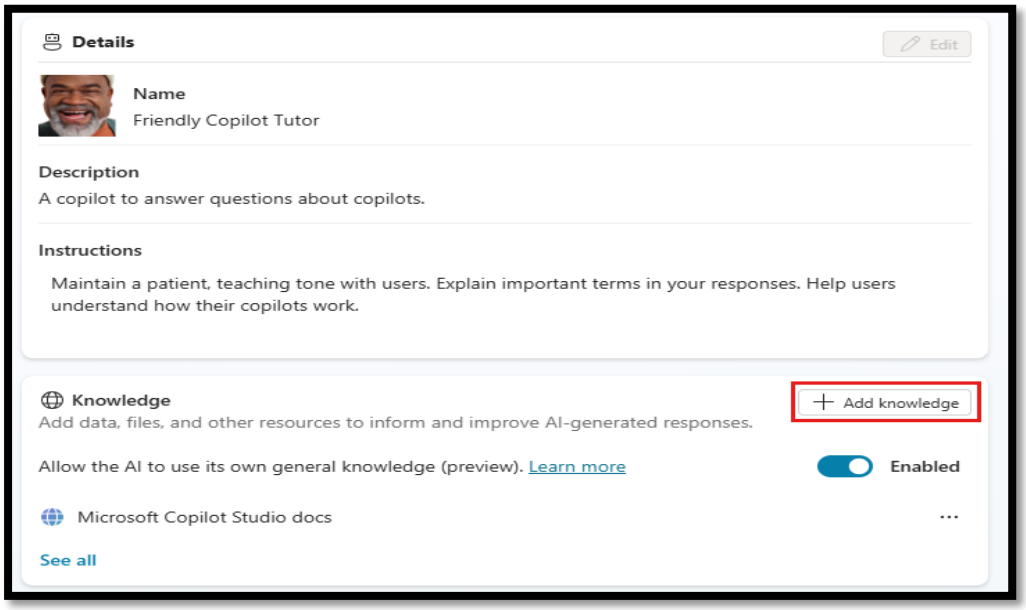

Fig: Add Knowlege

2. Select the type of knowledge. This quickstart uses **Public websites** sources.
3. Enter the URL for the website, then select **Add** beside the textbox.
4. Name and describe the knowledge source so you can keep track of all your copilot's sources.

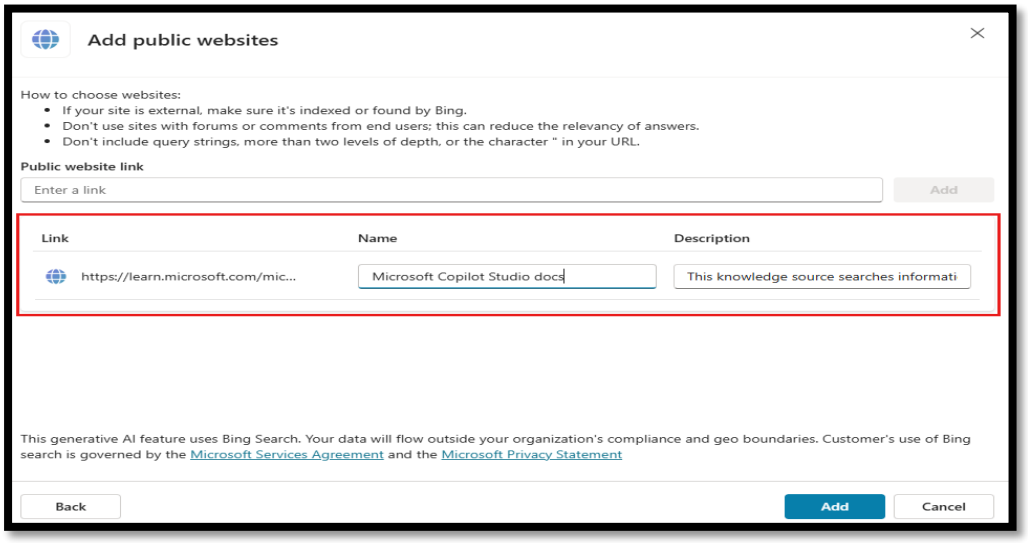

Fig: Add Public Websites

5. Select **Add**.

To change existing knowledge sources:

1. In the **Overview** tab's **Knowledge** section, select the overflow menu.
2. Select **Edit** to change the knowledge source, or **Delete** to remove it from your copilot's sources.

You now have a copilot you can start testing! You can chat with your copilot in the **Test your copilot** chat.

Publish your copilot

Once you are happy with the content authored in your copilot, you can publish your copilot to a website.

1. At the top of the page, select **Publish**, and then select **Publish** again in the **Publish this copilot** confirmation message. If the publish is successful, you see a green banner on the top of the page.

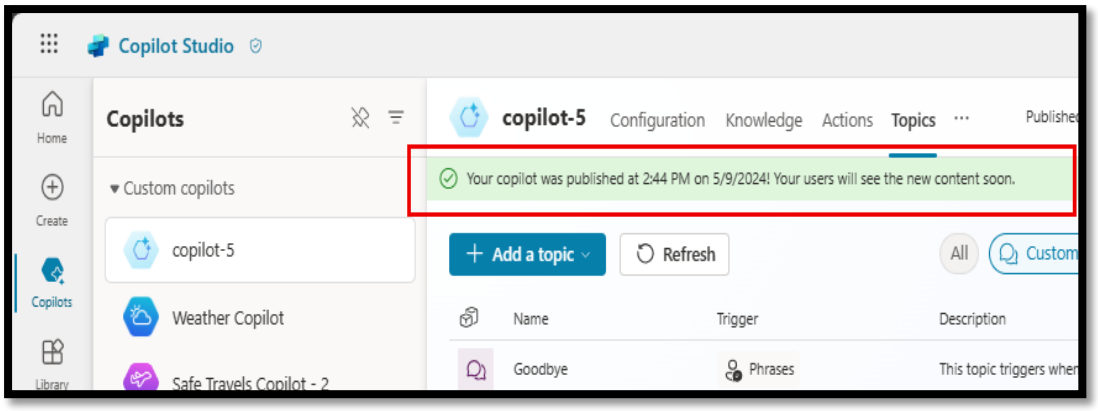

Fig: Publish Your Copilot

2. At the top of the page, select **Go to demo website** in the overflow menu.
3. Send the URL to others to demonstrate it.

Work with variables

You can use variables to save customers' responses and reuse their content later in the conversation.
You can also use variables to create logical expressions that dynamically route the customer down different conversation paths. For example, save a customer's name in a variable called *UserName,* and the copilot can address the customer by name as the conversation continues.

Create a variable
Any node that prompts you to select a variable as an output, such as a **Question** node, automatically creates an output variable of the appropriate type.

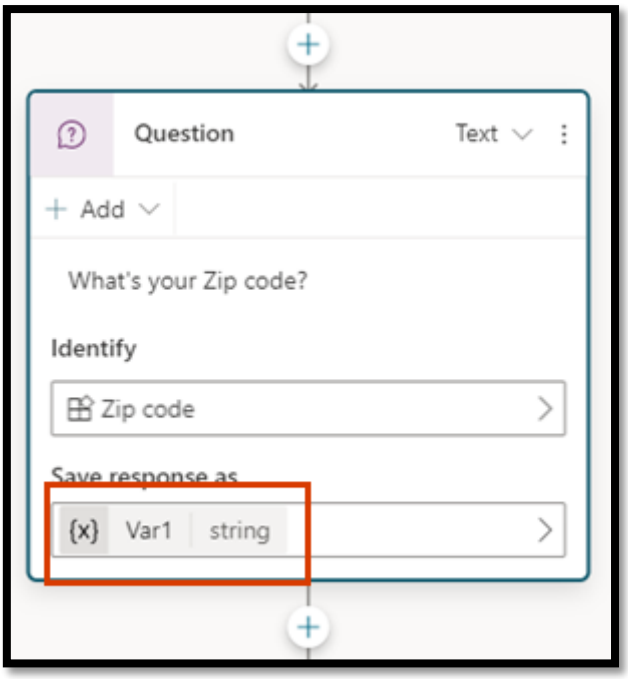

Fig: Create a Variable

Rename a variable

Variables are automatically assigned a name when you create them. A best practice is to give your variables meaningful names to make their purpose clear to anyone else who must maintain your copilot.

1. Select the variable to open it in the Variable properties pane.
2. Under **Variable name**, enter a new name for your variable.

Use entities and slot filling in copilots

A significant part of copilot conversations in Copilot Studio is natural language understanding, which is the ability for the AI to understand a user's intent. For example, the user might say "I tried to use my gift card but it doesn't work" and the copilot knows 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.

One fundamental aspect of natural language understanding is to identify *entities* in a user dialog. Watch the video to get started with entities.

An entity can be thought of as a unit of information that represents a certain type of a real-world subject, like a phone number, zip code, city, or even a person's name. With the knowledge granted by entities, a copilot can smartly recognize the relevant information from a user input and save it for later use.

Prebuilt entities

Entities in copilots let you store information in similar groups. Out of the box, Copilot Studio comes with a set of prebuilt entities, which represent the most used information types in real-world dialogs, such as age, colors, numbers, and

names. With the knowledge granted by entities, a copilot can smartly recognize the relevant information from a user input and save it for later use.

To help understand that notion, the **Money** entity can be used as an example.

1. Open your copilot in Copilot Studio and select **Settings** at the top of the page and then **Entities** in the menu.

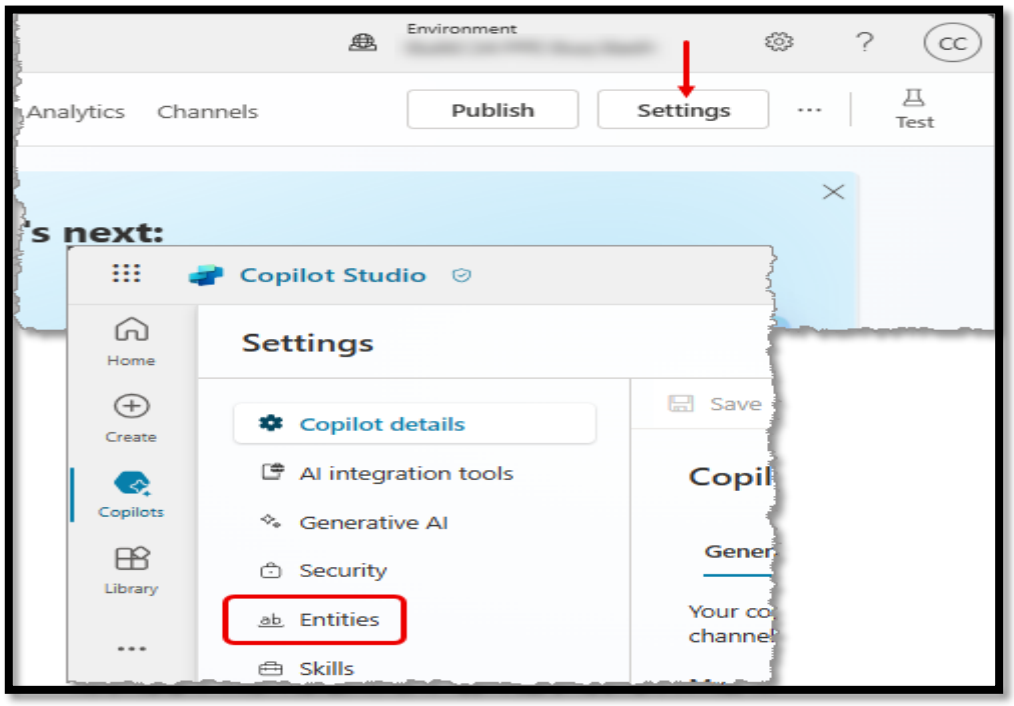

Fig: Select Entities

A list of the prebuilt entities that are available is shown.

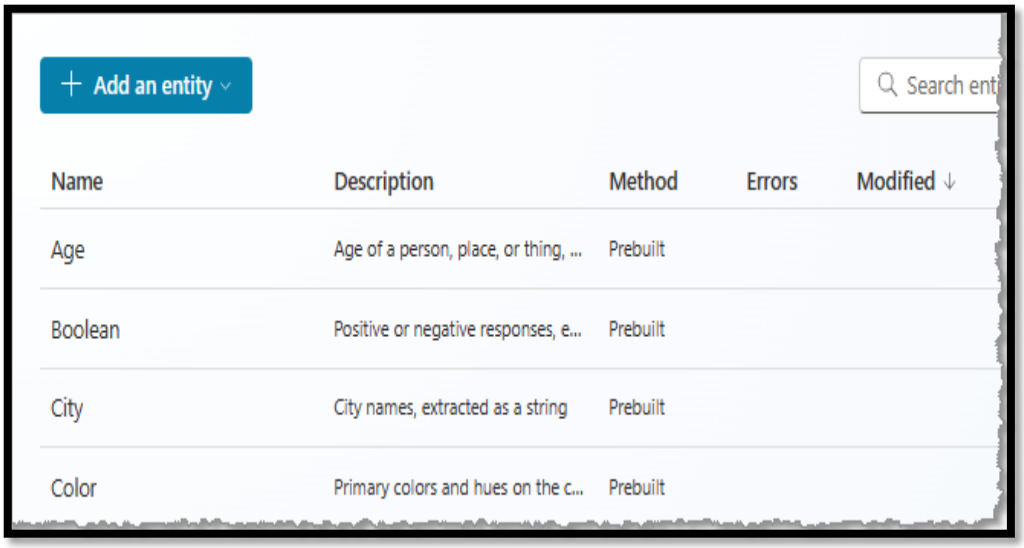

Fig: A list of the prebuilt entities

2. Select the **Money** entity to open the details pane for the entity.

Fig: Select Money

Here you can see an explanation of this entity and the ways it can be used to look for information related to money or currency from a user's input.

For example, when a user inputs "It costs 1000 dollars," using this money entity the copilot knows that "1000 dollars" represents the *money* type of information. When the copilot extracts this entity and saves it to a variable, it saves "1000" as a number even though the surrounding information was text.

Custom entities

The prebuilt entities cover commonly used information types. On some occasions, such as when building a copilot that serves a specific purpose, you might need to teach the copilot's language understanding model domain-specific knowledge.

For instance, let's say you want to build a copilot for an outdoor store. In this case, you need to teach the copilot to acknowledge the "outdoor gears product" category in a dialog.

First, create a custom entity. In this case, you can create an entity that gives the copilot the knowledge of all outdoor product categories.

1. Open your copilot in Copilot Studio and select **Settings** at the top of the page and then **Entities** in the menu.

2. Select **Add an entity** > **New entity**.

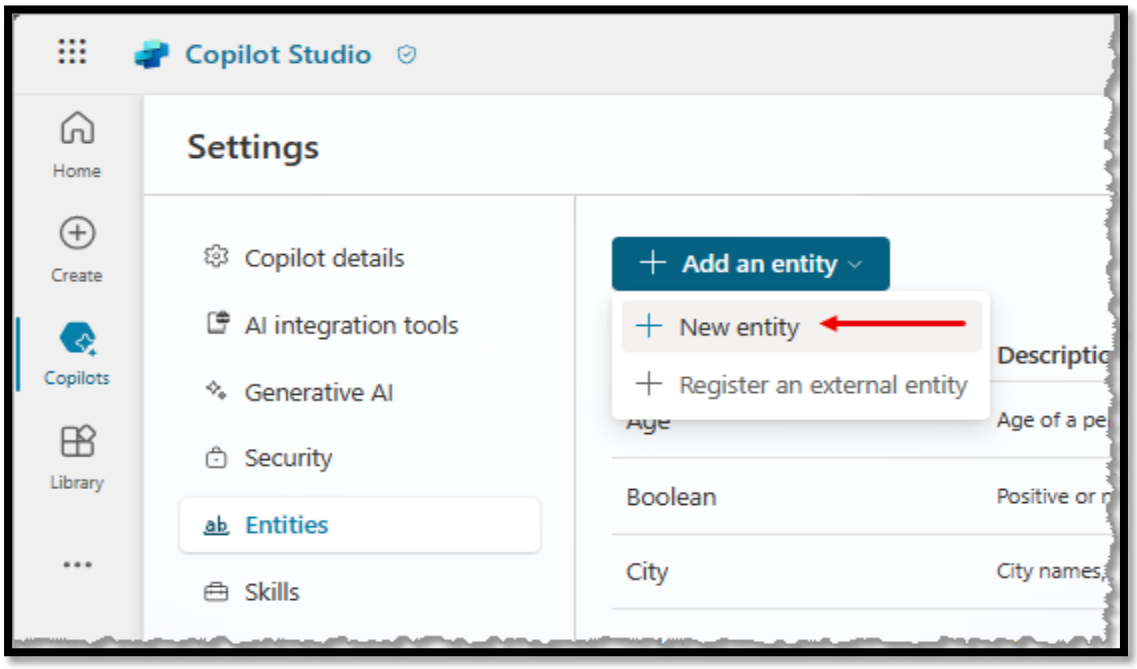

Fig: Add an entity

3. A new pane opens where you can choose the type of entity: either a Closed list entity or a Regular expression (regex) entity.

Closed list entities

Closed list entities let you define a list of items. These entities are best used for small lists that are easy to manage and that have simple item labels.

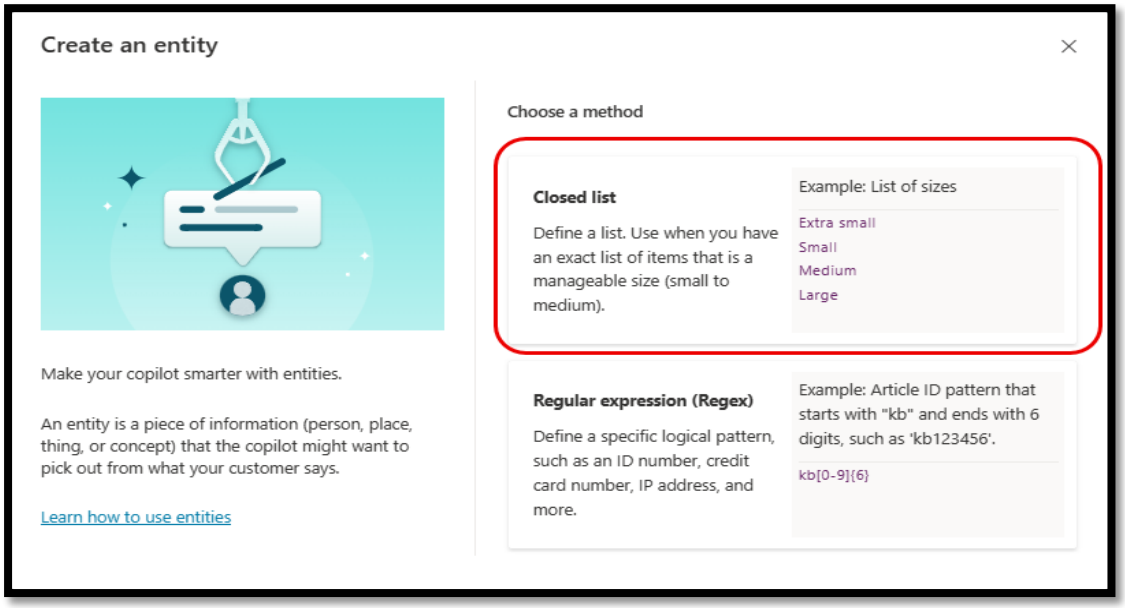

Fig: Create an entity

Selecting this option when creating an entity shows a pane where you can specify the name of the entity, an optional description, and enter the items you want included in the entity.

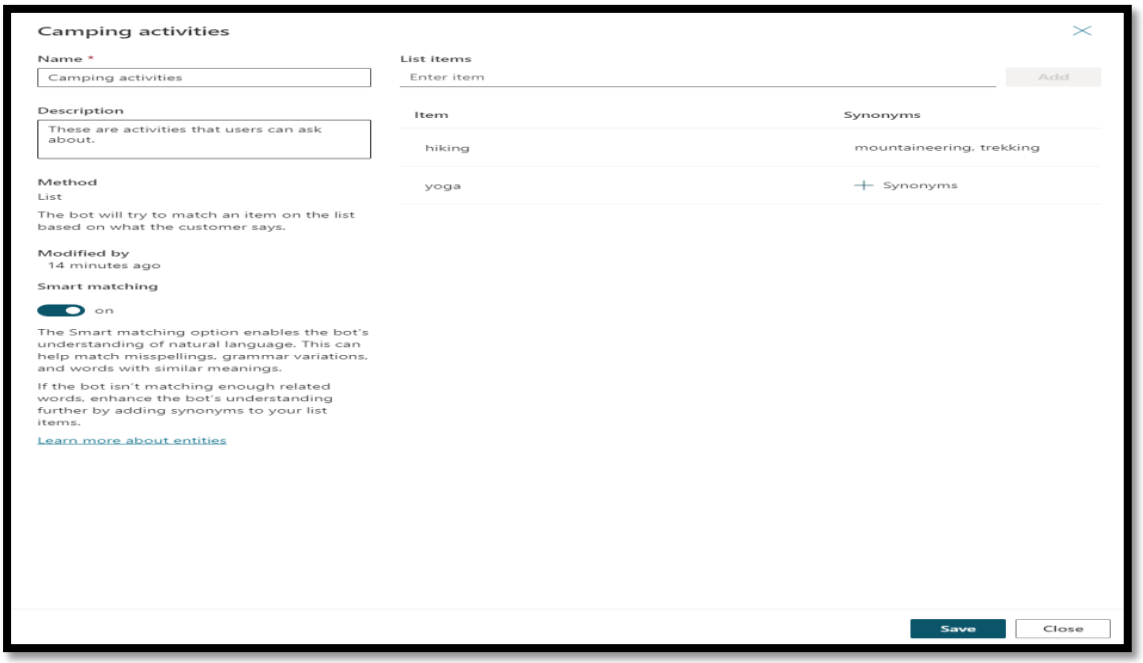

Fig: Add item name

When you enter items, you can:

- Select each item and change its name.
- Select the trash can icon to delete the item from the list.
- Select **+ Synonyms** (or select the listed synonyms if they're already added) to open the **Edit synonyms** pane.
  You can add synonyms to manually expand the matching logic for each item in the entity's list. For example, in the "hiking" item, you can add "trekking" and "mountaineering" as synonyms.

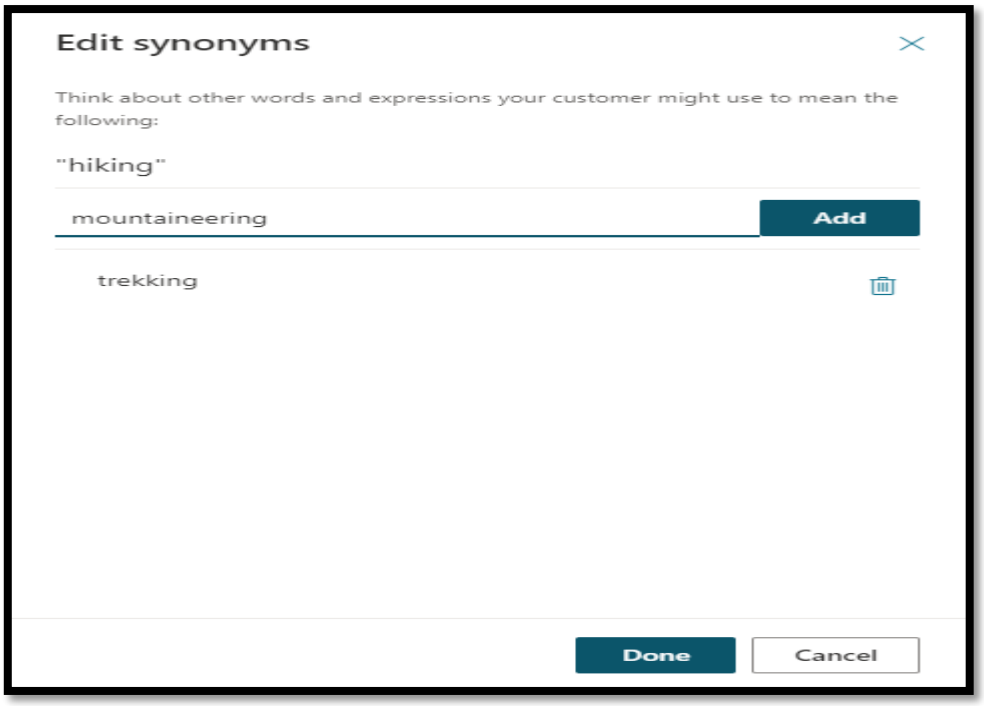

Fig: Add synonyms

For each entity, you can also enable **Smart matching**.

Smart matching is part of the intelligence supported by the copilot's language understanding model. With this option enabled, the copilot will interpret the user's input using fuzzy logic, based on the listed items in the entity.

In particular, the copilot will autocorrect misspellings and expand its matching logic semantically. For example, the copilot can automatically match "softball" to "baseball".

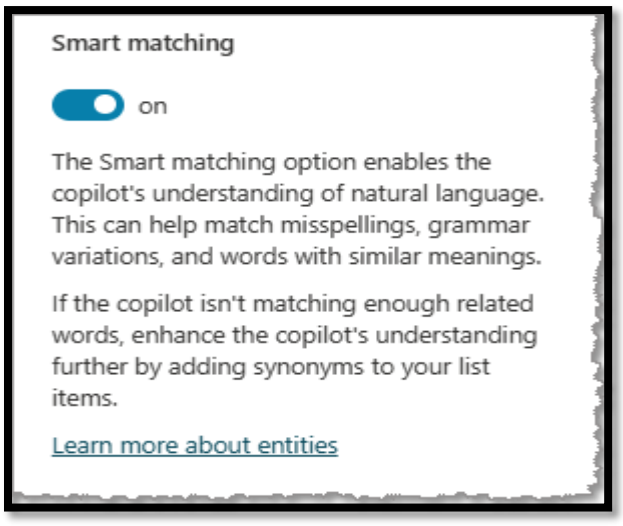

Fig: On Smart Matching

When you're finished creating or editing your entity, select **Save** to save and return to the list of entities. Select **Close** to discard your changes (you'll be given an option to go back to editing the entity in case you accidentally selected the wrong option).

Use entities in a conversation

Now that you've done the work of giving the copilot the knowledge about outdoor gear by creating that product category entity and a few other custom entities, you can start to use them when constructing a copilot conversation.

1. Go to the Topics page for the copilot you want to edit.
2. Open the topic for an entity you want to collect.
3. Select the **Add node** icon and then select **Ask a question**.
4. Under **Identify**, select the entity you created in Custom entities.

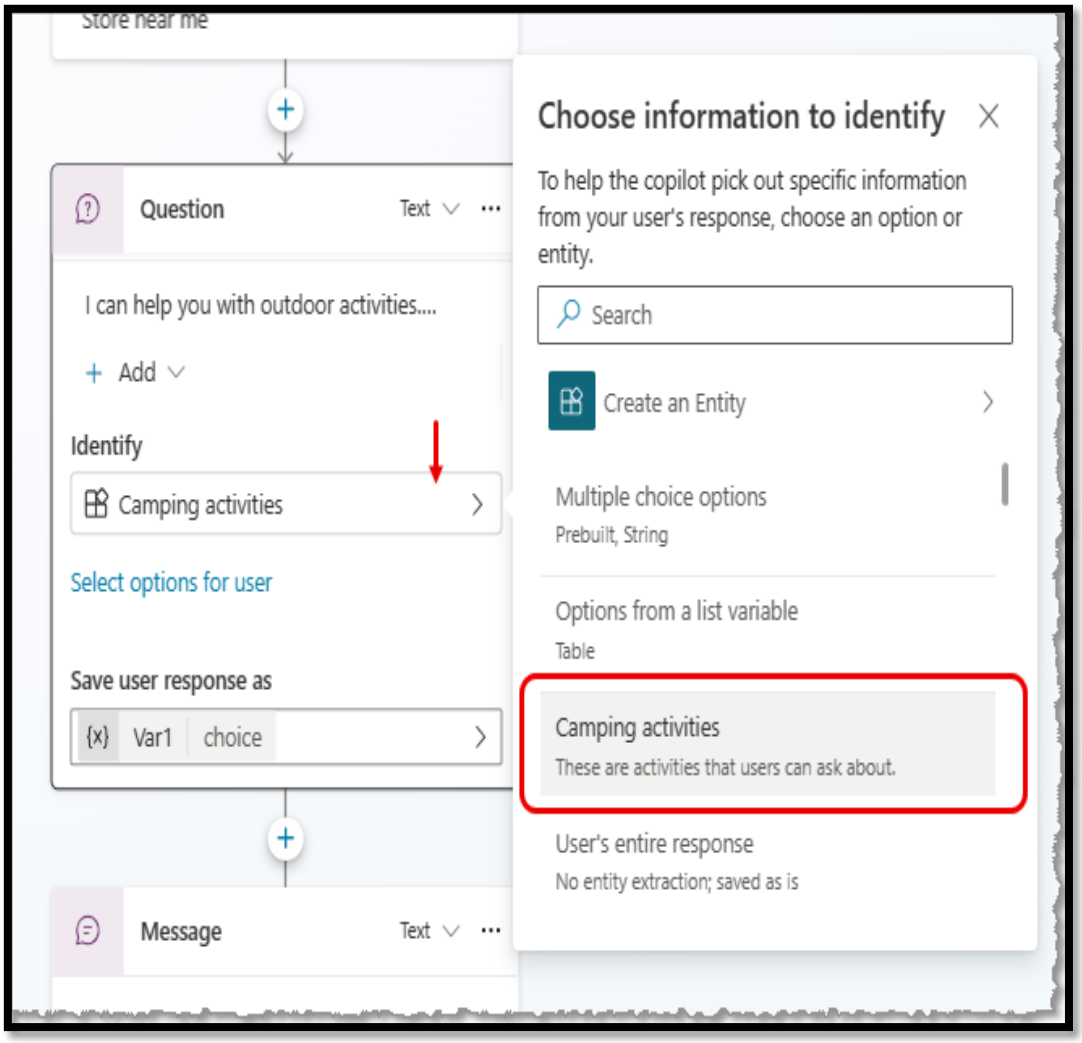

Fig: Choose Identity

5. You can also optionally select items to show as buttons. For example, if you'd like to show some categories as buttons for users to conveniently choose from as their input, you can choose **Select options for user** and then pick them from the list that contains the items you added when you created the custom entity.

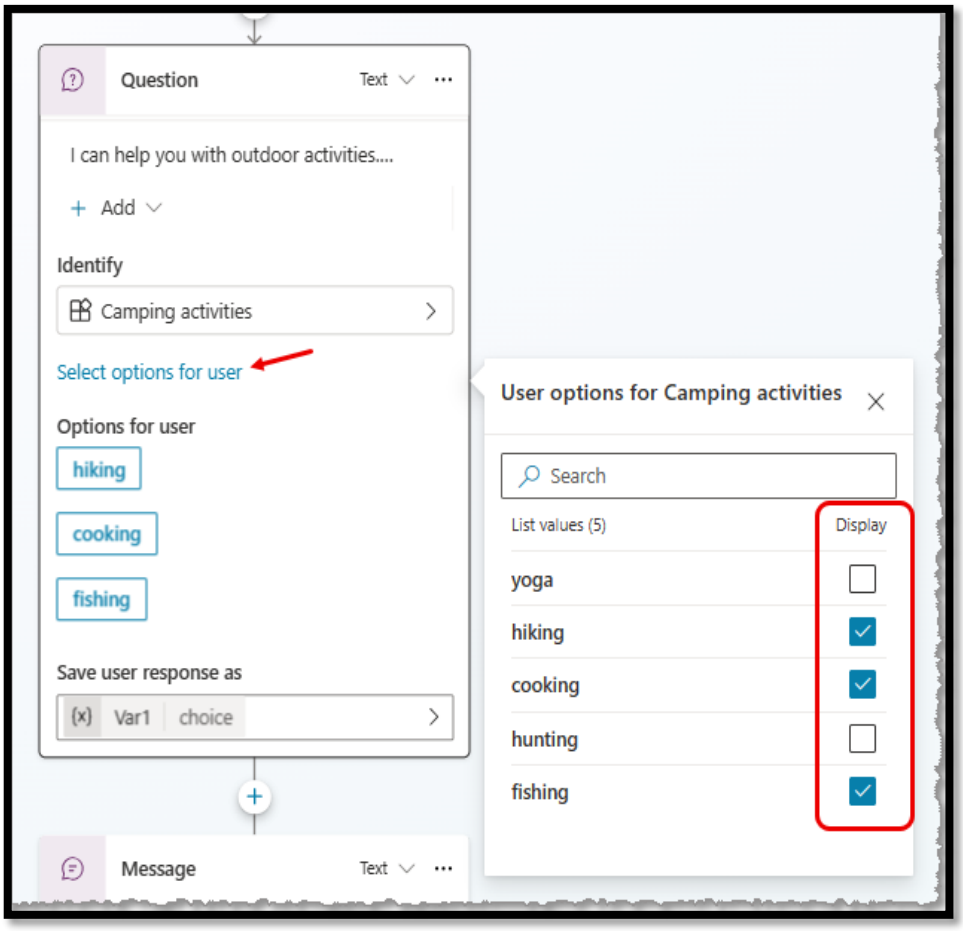

Fig: Select options for users

6. Name the variable for the output of the user's response, if necessary.

Slot filling

Slot filling is a natural language understanding concept that means saving an extracted entity to an object. However, in Copilot Studio, slot filling means placing the extracted entity value into a variable.

We'll continue to use the camping activities topic as an example, which is triggered by typing "I want something to do" in the test chat.

The topic is successfully triggered, and the copilot asks for the type of activity, also showing the button choices specified when authoring the **Ask a question** node. In the dialog tree, tracking between topics also shows the copilot is running to the question node you just edited.

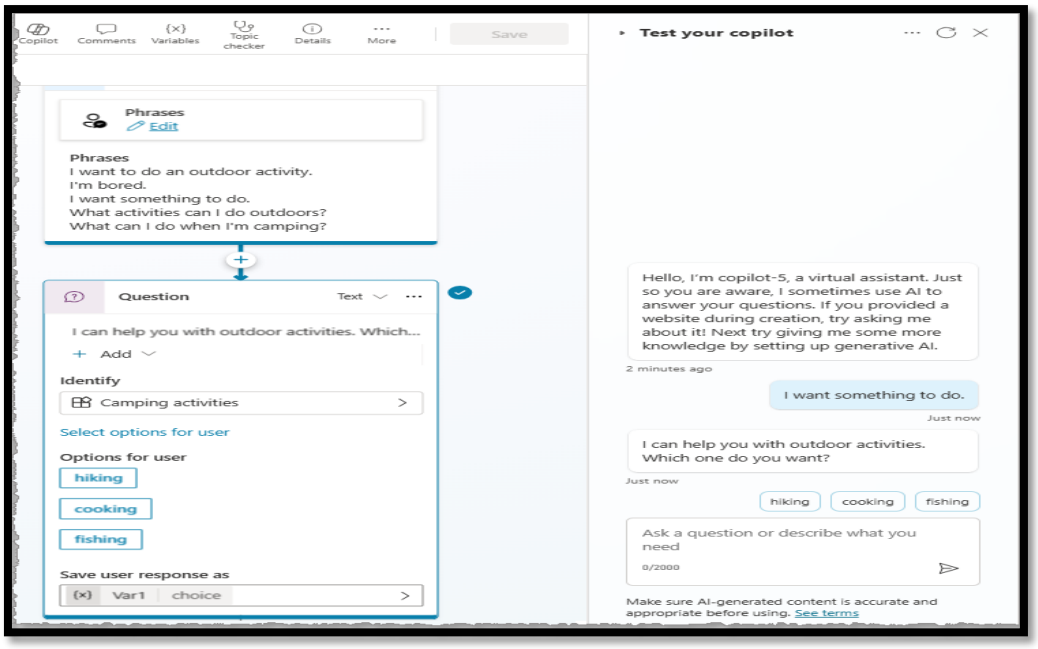

Fig: Ask a question

A user can use the predefined choices by selecting one of the buttons. Alternatively, they can also type in something like "trekking" and see that it maps to "hiking", because those words were defined as synonyms.

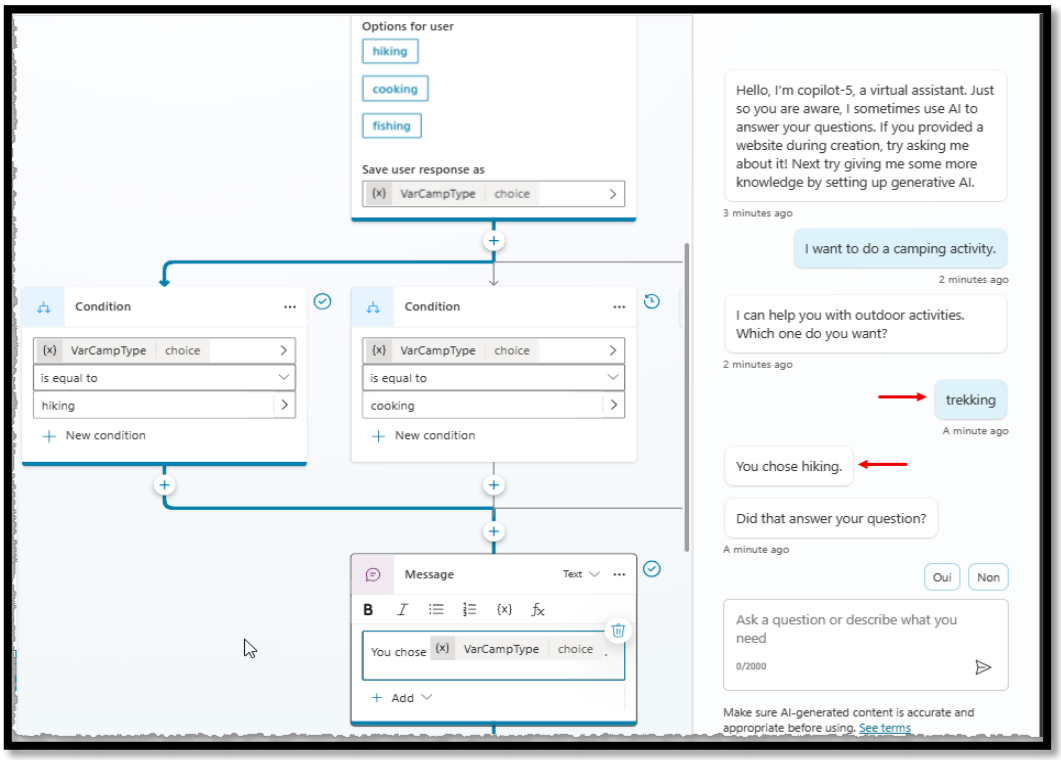

Fig: Add predefined choices

In tracking between topics, it shows the dialog is correctly routed to the path in which the product category value is "Hiking". You can inspect the variable value from the variable watch window at the bottom of the authoring canvas. In the watch window, it shows the variable value is "Hiking".

Essentially, slot filling has happened by inserting the extracted entity "Hiking" into the variable VarCampType.

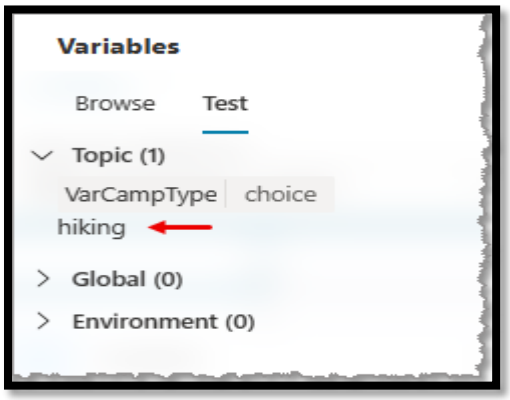

Fig: Inspect variable value

You can also use what is known as "proactive slot filling" where the user can specify multiple pieces of information that map to multiple entities. The copilot can understand what information belongs to which entity automatically. In cases where it is unsure of the intended mapping, it will prompt the user to be more specific by providing choices.

In this example, the user wrote "I'm bored but I like mountaineering." This message includes both the trigger phrase that the user wants help with outdoor activities and provides a second piece of information, "mountaineering", which is an entity-defined synonym for hiking. In this case, the copilot fills in both the entity for choosing an outdoor activity, and for the type of activity.

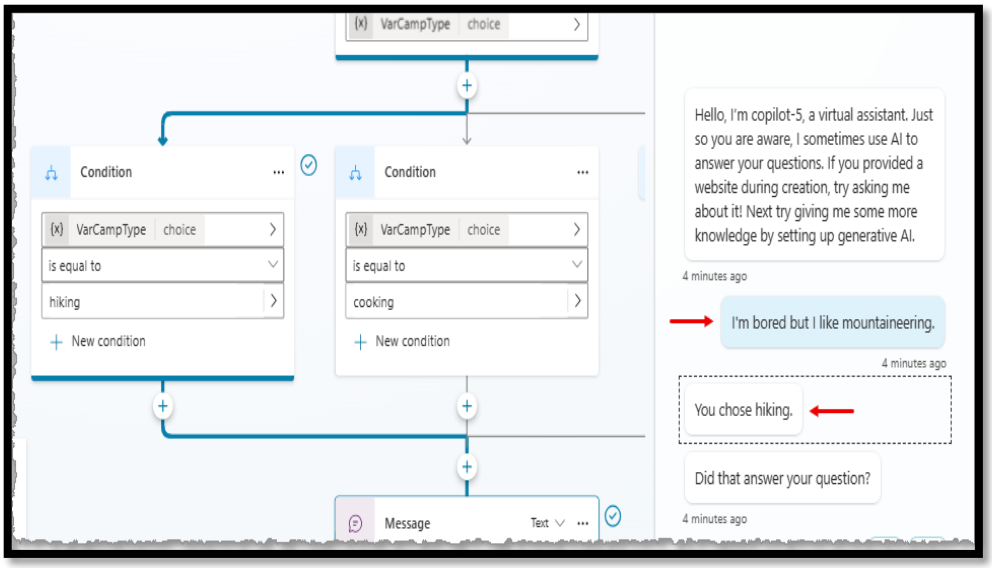

Fig: Entity-defined synonym for hiking

In tracking between topics, you can see the copilot takes in this user input, and intelligently skips the question node asking for the type of activity.

The copilot is always actively listening to the user input, remembering information upfront so it can skip unnecessary steps as appropriate.

Let us restart the testing again and try another case. In this round, you can add a couple more question nodes asking for things like how much time you have for the activity (using the **Duration** entity) as well as the price range (using the **Money** entity).

This time when the product category question is presented, instead of telling the copilot only the product category, the user can say "I want to buy a pair of hiking boots under $100". In this example, the copilot isn't only able to route to the correct hiking product category path, but also actively fill the slots asking for the type of hiking gear and the target price range information.

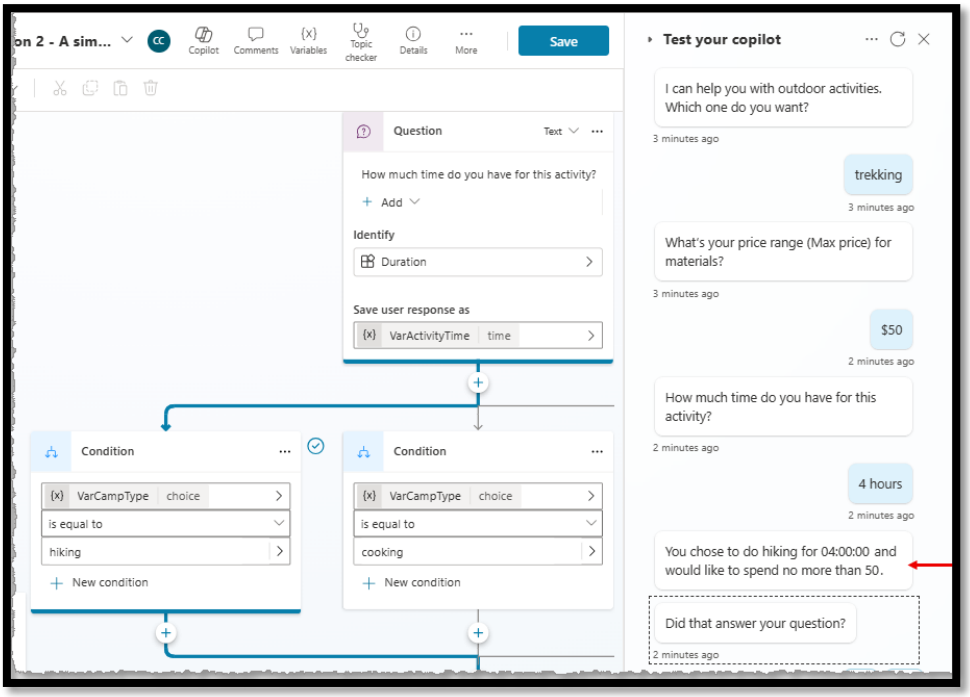

Fig: Proactive Slot

Proactive slot filling can be manually controlled at the node level. If you'd like to always prompt for the question within a specific node, regardless of whether the slot has been filled from previous user responses, you can disable the **Skip question** option for that question node. Perform the following steps to disable the **Skip question** option.

1. Select the **More** icon (**…**) of the **Question** node, and then select **Properties**. The **Question properties** panel appears.

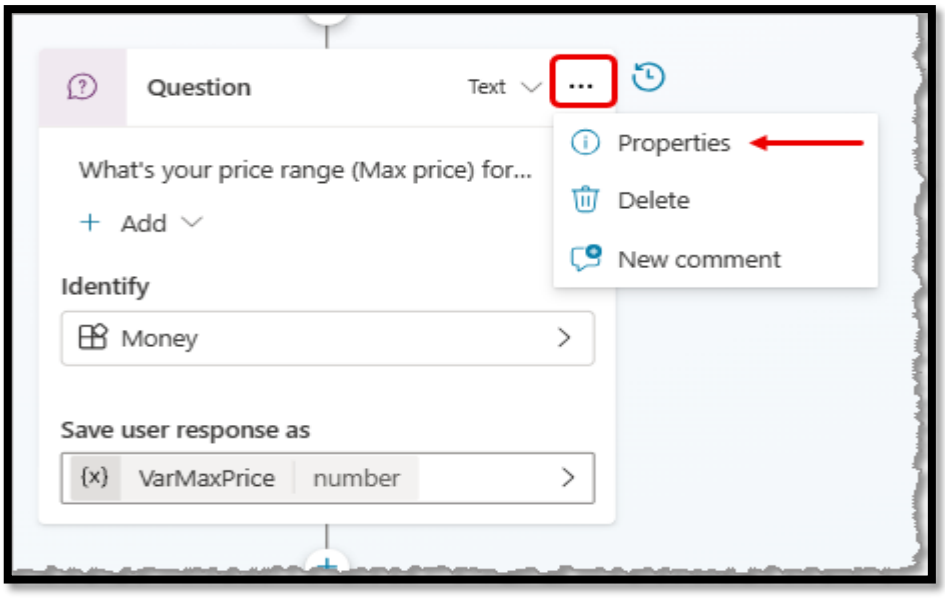

Fig: Properties

2. On the **Question properties** panel, select **Question behavior**.
3. On the **Question behavior** panel, under **Skip question**, select **Ask every time**, and then select **Save** at the top of the page.

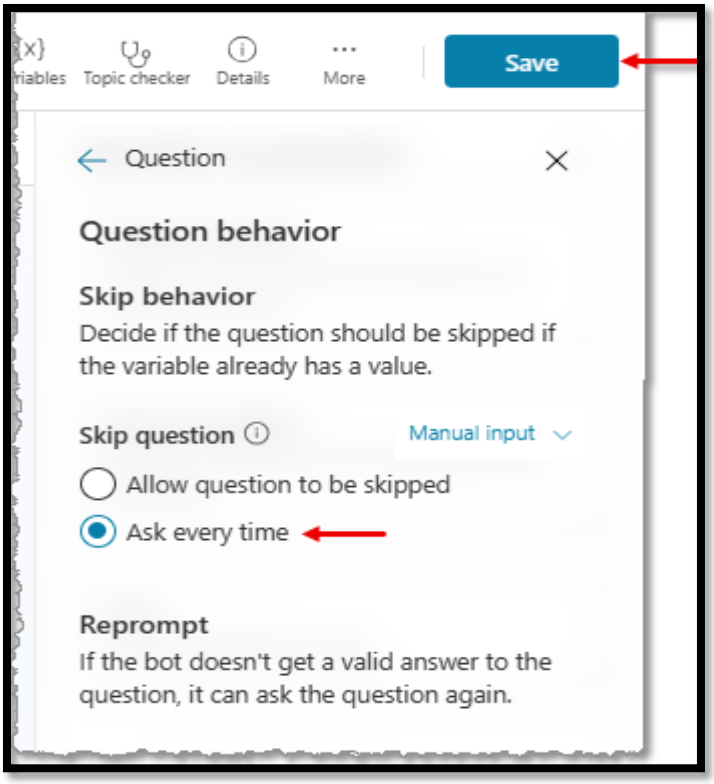

Fig: Question Panel

Create and delete copilots

Copilot Studio lets you create a copilot, using built-in content building blocks containing topics, trigger phrases, and preauthored copilot conversations.

Create a copilot

1. Go to the Copilot Studio home page, select **Create** in the left navigation, then select on the **Copilots** page.
   Alternatively, select **+ New copilot** on the **Copilots** page.
2. Use the chat to describe your copilot, using the provided questions for guidance.
   Alternatively, select **Skip to configure** and fill the form.
3. Select **Create**.

Delete a copilot

You can delete copilots to remove them from your environment.

1. Go to the Copilot Studio home page.
2. On the side navigation pane, select **Copilots**.
3. Select the copilot you want to delete.
4. On the top menu bar, select the **More options** icon (**…**), then select **Delete**.

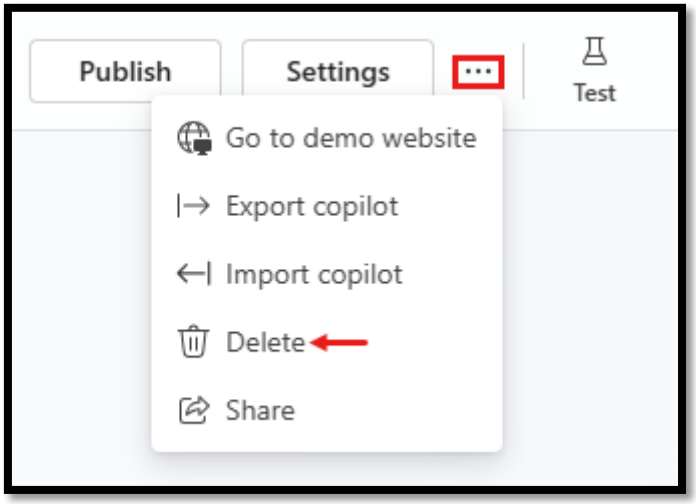

Fig: Delete Copilot

5. Confirm the deletion of the copilot by typing the copilot's name. Once you confirm, all copilot content is deleted after a few minutes.
   If your license is expired, you can delete your copilots by selecting **Permanently delete your copilots**.

Publish a copilot to a live or demo website

When publishing the copilot to the web, you can publish it to a prebuilt demo website (which you can use to share the copilot with your teammates and stakeholders). You can also publish it to your own live website.
You can edit the welcome message and suggested trigger phrases for the demo website. A welcome message helps prompt your teammates for what they should ask the copilot about.

Update the demo website

1. Open your copilot and on the top menu bar, select **Settings**.
2. On the side navigation pane, select **Security**.
3. Select **Authentication** and then **No authentication**, and then select **Save**.
4. Exit **Settings** and on the top menu bar, select **Channels**.
5. Under **Channels**, select **Demo website**.
6. On the **Demo Website** pane, under **Welcome message**, enter the message you want your teammates to see.
7. Enter a list of suggested trigger phrases in the **Conversation starters** text field. Trigger phrases are what initiate specific topics, so you could customize your trigger phrases to specific topics that you want your teammates to try out.
   To share the demo website link, copy the demo website URL and share it with your teammates directly.

Custom website

You can add your copilot to a live website as an iframe. Your live website can be a customer-facing external website or an internal site, like a SharePoint site. You can also add the copilot to your Power Platform admin center.

Add copilot to your website

1. Under the copilot settings, select **Channels**.
2. Select **Custom website** and then select **Copy** to copy it directly to the clipboard, or **Share to email** to open a new email message with the snippet included, in your default email app.

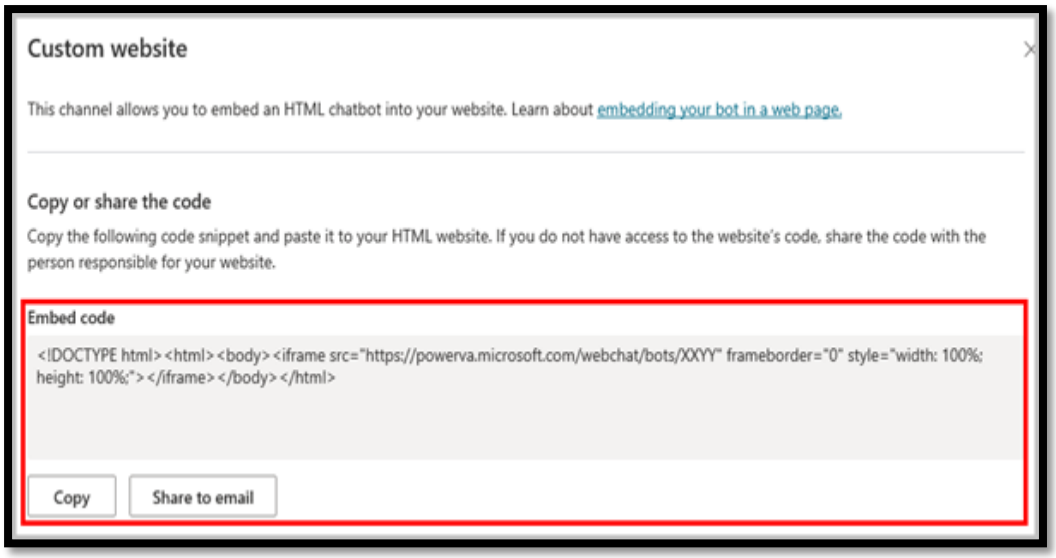

Fig: Custom Website

3. Provide the snippet to your web developer to add the copilot to your website.

Connect a copilot to the Microsoft Teams channel

After publishing your copilot at least once, you can connect it to the Microsoft Teams channel to allow users to chat with it in Teams.

1. Open the configuration panel for the Microsoft Teams channel.
2. Select **Turn on Teams**.

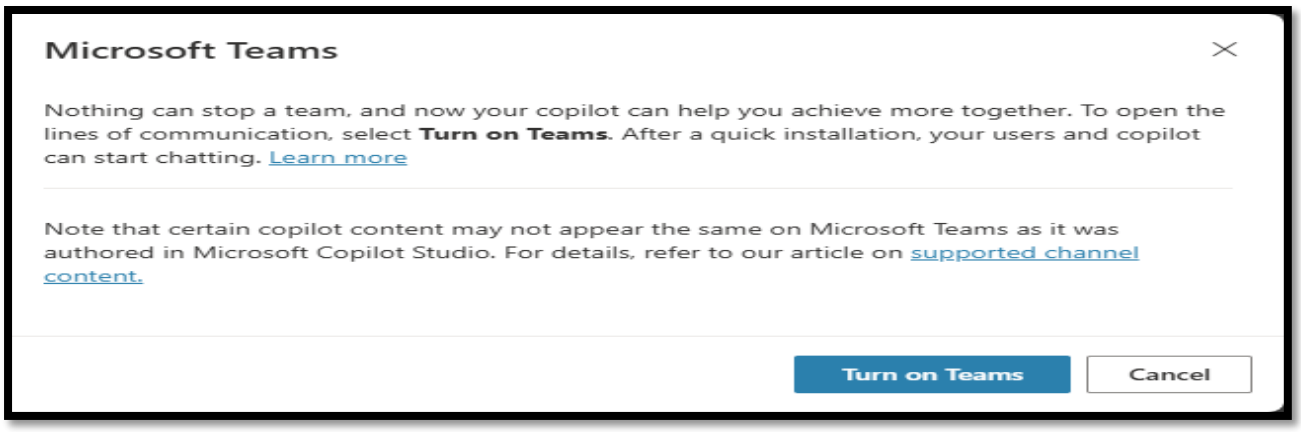

Fig: Turn on Teams

Customize the appearance of a copilot for Teams

Providing the right description and appearance for a copilot before making it available to other users is important as it informs them on the copilot's purpose and branding, when applicable.

1. Open the configuration panel for the Microsoft Teams channel.

2. Select **Edit details** to change the copilot's icon, color, and descriptions. These attributes are visible in the Teams app store and on the app's **About** tab after the user install the copilot. Review the app icon format guidelines for Teams.

3. Select **More** to add the following information:
    - Developer name
    - Website
    - Privacy statement
    - Terms of use

You should provide this information to inform the users about your copilot. For more information, see Create a privacy statement and terms of use in Microsoft Teams.

4. Select **Save**. Your changes to the colour, icon, and short description are now visible on the configuration panel.

Install a copilot as an app in Teams

With your copilot published, you can add the copilot to your own Teams profile directly from Copilot Studio.

> We recommend adding your copilot to your own profile in Teams first, before sharing it with others.

1. Open the configuration panel for the Microsoft Teams channel.

2. Select **Open copilot** to have the installation prompt launch in Teams. Select **Add** to add it as an app in Teams. This operation only installs the copilot for you.