

---

## Building reusable prompts

Prompt engineering to find effective prompts for a model takes time and effort. Stretch the benefits of your work by building prompts that you can reuse and share with others.

A great way to add flexibility to a prompt is to add *prompt variables*. A prompt variable is a placeholder keyword that you include in the static text of your prompt at creation time and replace with text dynamically at run time.

Note: You cannot define prompt variables in chat mode.

---

## Using variables to change prompt text dynamically

Foundation model input variables help you to generalize a prompt so that it can be reused more easily.

For example, a prompt for a generative task might contain the following static text:

**Write a story about a dog.**

If you replace the text *dog* with a variable that is named `{animal}`, you add support for dynamic content to the prompt.

**Write a story about a {animal}.**

With the variable `{animal}`, the text can still be used to prompt the model for a story about a dog. But now it can be reused to ask for a story about a cat, a mouse, or another animal, simply by swapping the value that is specified for the `{animal}` variable.

---

## Creating prompt variables

To create a prompt variable, complete the following steps:

1. From the Prompt Lab, review the text in your prompt for words or phrases that, when converted to a variable, will make the prompt easier to reuse.
2. Click the **Prompt variables** icon `{#}` at the start of the page.

The *Prompt variables* panel is displayed where you can add variable name-and-value pairs.

**Note:** The *Prompt variables* panel is not available in chat mode.

3. Click **New variable**.
4. Click to add a variable name, tab to the next field, and then add a default value.

The variable name can contain alphanumeric characters or an underscore `_`, but cannot begin with a number.

The default value for the variable is a fallback value; it is used every time that the prompt is submitted, unless someone overwrites the default value by specifying a new value for the variable.


5. Repeat the previous step to add more variables.

The following table shows some examples of the types of variables that you might want to add.

Table 1. Example variable name and value pairs


Variable name	Default value
country	Ireland
city	Boston
project	Project X
company	IBM

6. Replace static text in the prompt with your variables.

Select the word or phrase in the prompt that you want to replace, and then click the **Prompt variables** icon  within the text box to see a list of available variables. Click the variable that you want to use from the list.

The variable replaces the selected text. It is formatted with the syntax `{variable name}`, where the variable name is surrounded by braces.

If your static text already contains variables that are formatted with braces, they are ignored unless prompt variables of the same name exist.

You can click the **Text** icon  to see the full prompt text with variables replaced by values.

7. To specify a value for a variable at run time, open the *Prompt variables* panel, click **Preview**, and then add a value for the variable.

You can also change the variable value from the edit view of the *Prompt variables* panel, but the value you specify will become the new default value.

When you find a set of prompt static text, prompt variables, and prompt engineering parameters that generates the results you want from a model, save the prompt as a prompt template asset. After you save the prompt template asset, you can reuse the prompt or share it with collaborators in the current project.

## Examples of reusing prompts

The following examples help illustrate ways that using prompt variables can add versatility to your prompts.

- [Thank you note example](#)
- [Devil's advocate example](#)

### Thank you note example

Replace static text in the *Thank you note generation* built-in sample prompt with variables to make the prompt reusable.

To add versatility to a built-in prompt, complete the following steps:

1. From the Prompt Lab, click **Sample prompts** to list the built-in sample prompts. From the *Generation* section, click **Thank you note generation**.

The input for the built-in sample prompt is added to the prompt editor and the `flan-ul2-20b` model is selected.

**Write a thank you note for attending a workshop.**

**Attendees:** `interns`  
**Topic:** `codefest, AI`  
**Tone:** `energetic`

2. Review the text for words or phrases that make good variable candidates.

In this example, if the following words are replaced, the prompt meaning will change:


- workshop
  - interns
  - codefest
  - AI
  - energetic
3. Click the **Prompt variables** icon  and create a variable to represent each word in the list. Add the current value as the default value for the variable.

Table 2. Variables for the thank you note sample

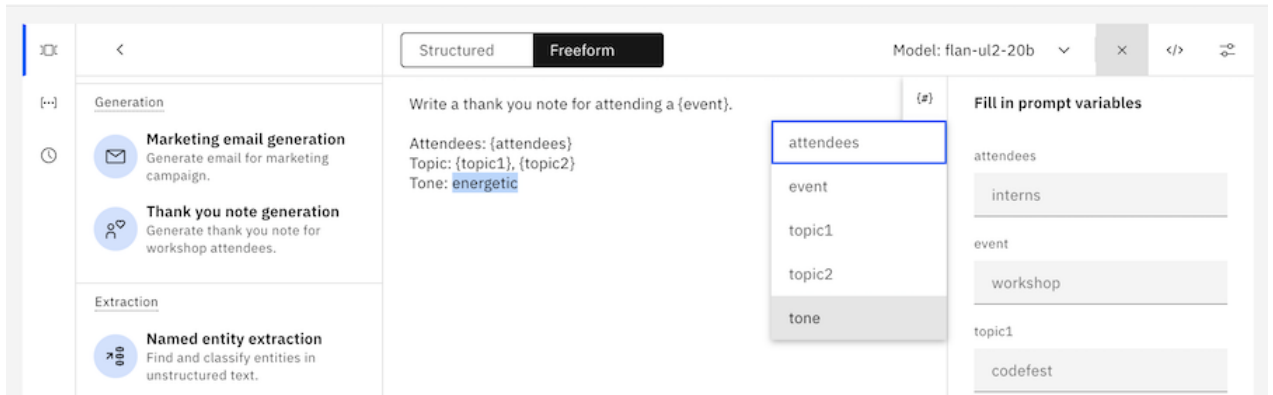
Variable name	Value
event	workshop
attendees	interns
topic1	codefest
topic2	AI

Variable name	Value
tone	energetic

- Click **Preview** to review the variables that you added.
- Update the static prompt text to use variables in place of words.

**Write a thank you note for attending a {event}.**

**Attendees:** {attendees}  
**Topic:** {topic1}, {topic2}  
**Tone:** {tone}



The original meaning of the prompt is maintained.

- Now, change the values of the variables to change the meaning of the prompt.

From the *Fill in prompt variables* view of the prompt variables panel, add values for the variables.

Table 3. New values for the variables

Variable name	Value
event	human resources presentation
attendees	expecting parents
topic1	resources for new parents
topic2	parental leave
tone	supportive

You effectively converted the original prompt into the following prompt:

**Write a thank you note for attending a human resources presentation.**

**Attendees:** expecting parents  
**Topic:** resources for new parents, parental leave  
**Tone:** supportive

Click **Generate** to see how the model responds.

- Swap the values for the variables to reuse the same prompt again to generate thank you notes for usability test attendees.

Table 4. Another set of values for the variables

Variable name	Value
event	usability test
attendees	user volunteers
topic1	testing out new features
topic2	sharing early feedback
tone	appreciative

Click **Generate** to see how the model responds.

## Devil's advocate example

Use prompt variables to reuse effective examples that you devise for a prompt.

You can guide a foundation model to answer in an expected way by adding a few examples that establish a pattern for the model to follow. This kind of prompt is called a *few-shot prompt*. Inventing good examples for a prompt requires imagination and testing and can be time-consuming. If you successfully create a few-shot prompt that proves to be effective, you can make it reusable by adding prompt variables.

Maybe you want to use the granite-13b-instruct-v1 model to help you consider risks or problems that might arise from an action or plan under consideration.

For example, the prompt might have the following instruction and examples:

You are playing the role of devil's advocate. Argue against the proposed plans. List 3 detailed, unique, compelling reasons why moving forward with the plan would be a bad choice. Consider all types of risks.

Plan we are considering:

Extend our store hours.

Three problems with this plan are:

1. We'll have to pay more for staffing.
2. Risk of theft increases late at night.
3. Clerks might not want to work later hours.

Plan we are considering:

Open a second location for our business.

Three problems with this plan are:

1. Managing two locations will be more than twice as time-consuming than managed just one.
2. Creating a new location doesn't guarantee twice as many customers.
3. A new location means added real estate, utility, and personnel expenses.

Plan we are considering:

Refreshing our brand image by creating a new logo.

Three problems with this plan are:

You can reuse the prompt by completing the following steps:

1. Replace the text that describes the action that you are considering with a variable.

For example, you can add the following variable:

Table 5. Variable that represents the possible action

Variable name	Default value
plan	Refreshing our brand image by creating a new logo.

2. Replace the static text that defines the plan with the {plan} variable.

You are playing the role of devil's advocate. Argue against the proposed plans. List 3 detailed, unique, compelling reasons why moving forward with the plan would be a bad choice. Consider all types of risks.

Plan we are considering:

Extend our store hours.

Three problems with this plan are:

1. We'll have to pay more for staffing.
2. Risk of theft increases late at night.
3. Clerks might not want to work later hours.

Plan we are considering:

Open a second location for our business.

Three problems with this plan are:

1. Managing two locations will be more than twice as time-consuming than managed just one.
2. Creating a new location doesn't guarantee twice as many customers.
3. A new location means added real estate, utility, and personnel expenses.

Plan we are considering:

{plan}

Three problems with this plan are:

Now you can use the same prompt to prompt the model to brainstorm about other actions.

3. Change the text in the **{plan}** variable to describe a different plan, and then click **Generate** to send the new input to the