Quick tip: Review the prerequisites before you run the lab

End Lab 00:31:45

**Caution:** When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. Learn more.

Open Google Cloud console

Username
student-01-213db0bb56e9

Password
i5tfs7gIus0D

Project ID
qwiklabs-gcp-02-fec0c2c

Region
us-central1

# Generative AI with Vertex AI: Prompt Design

Lab · 45 minutes · No cost · Introductory

This lab may incorporate AI tools to support your learning.

## GSP1151

Google Cloud Self-Paced Labs



## Overview

This lab explores prompt engineering and best practices for designing effective prompts to improve the quality of your LLM-generated responses. You'll learn how to craft prompts that are concise, specific, and well-defined, focusing on one task at a time. The lab also covers advanced techniques like turning generative tasks into classification tasks and using examples to enhance response quality. For further exploration, refer to the official documentation on prompt design.

### Gemini

Gemini is a family of powerful generative AI models developed by Google DeepMind, capable of understanding and generating various forms of content, including text, code, images, audio, and video.

**Gemini API in Vertex AI**

models. This allows developers to easily integrate these powerful AI capabilities into their applications. For the most up-to-date details and specific features of the latest versions, please refer to the official Gemini documentation.

**Gemini Models**

- Gemini Pro: Designed for complex reasoning, including:
  - Analyzing and summarizing large amounts of information.
  - Sophisticated cross-modal reasoning (across text, code, images, etc.).
  - Effective problem-solving with complex codebases.
- Gemini Flash: Optimized for speed and efficiency, offering:
  - Sub-second response times and high throughput.
  - High quality at a lower cost for a wide range of tasks.
  - Enhanced multimodal capabilities, including improved spatial understanding, new output modalities (text, audio, images), and native tool use (Google Search, code execution, and third-party functions).

Before starting this lab, you should be familiar with:

- Basic Python programming.
- General API concepts.
- Running Python code in a Jupyter notebook on Vertex AI Workbench.

## Objectives

In this lab, you will learn how to:

- Get started with prompt engineering using the Google Gen AI SDK
- Apply best practices for prompt design, including conciseness, specificity, and task definition
- Explore various text generation use cases with the Google Gen AI SDK, such as:
  - Question answering
  - Text classification
  - Text extraction
  - Text summarization

## Setup and requirements

### Before you click the Start Lab button

Read these instructions. Labs are timed and you cannot pause them. The timer, which starts when you click **Start Lab**, shows how long Google Cloud resources are made available to you.

This hands-on lab lets you do the lab activities in a real cloud environment, not in a

To complete this lab, you need:

- Access to a standard internet browser (Chrome browser recommended).

**Note:** Use an Incognito (recommended) or private browser window to run this lab. This prevents conflicts between your personal account and the student account, which may cause extra charges incurred to your personal account.

- Time to complete the lab—remember, once you start, you cannot pause a lab.

**Note:** Use only the student account for this lab. If you use a different Google Cloud account, you may incur charges to that account.

### How to start your lab and sign in to the Google Cloud console

following:

- The Open Google Cloud console button
- Time remaining
- The temporary credentials that you must use for this lab
- Other information, if needed, to step through this lab

2. Click **Open Google Cloud console** (or right-click and select **Open Link in Incognito Window** if you are running the Chrome browser).

   The lab spins up resources, and then opens another tab that shows the Sign in page.

   ***Tip:*** Arrange the tabs in separate windows, side-by-side.

   **Note:** If you see the **Choose an account** dialog, click **Use Another Account**.

```
student-01-213db0bb56e9@qwiklabs.net
```

   You can also find the Username in the Lab Details pane.

4. Click **Next**.

5. Copy the **Password** below and paste it into the **Welcome** dialog.

```
i5tfs7gIus0D
```

You can also find the Password in the Lab Details pane.

6. Click **Next**.

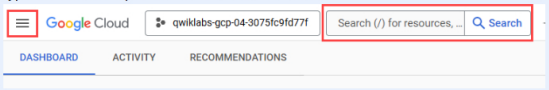
> **Important:** You must use the credentials the lab provides you. Do not use your Google Cloud account credentials.

---

charges.

7. Click through the subsequent pages:
    - Accept the terms and conditions.
    - Do not add recovery options or two-factor authentication (because this is a temporary account).
    - Do not sign up for free trials.

After a few moments, the Google Cloud console opens in this tab.

> **Note:** To access Google Cloud products and services, click the **Navigation menu** or type the service or product name in the **Search** field.
>
> 
> 

---

## Task 1. Open the notebook in Vertex AI Workbench

1. In the Google Cloud console, on the **Navigation menu** (☰), click **Vertex AI > Workbench**.
2. Find the `vertex-ai-jupyterlab` instance and click on the **Open JupyterLab** button.

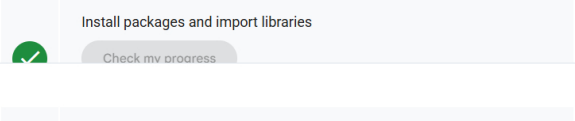
The JupyterLab interface for your Workbench instance opens in a new browser tab.

---

1. Open the `intro_prompt_design` file.
2. In the **Select Kernel** dialog, choose **Python 3** from the list of available kernels.
3. Run through the **Getting Started** and the **Import libraries** sections of the notebook.
    - For **Project ID**, use `qwiklabs-gcp-02-fec0c2c8ac8f`, and for **Location**, use `us-central1`.

> **Note:** You can skip any notebook cells that are noted *Colab only*. If you experience a 429 response from any of the notebook cell executions, wait 1 minute before running the cell again to proceed.

Click **Check my progress** to verify the objective.



## Task 3. Prompt engineering best practices

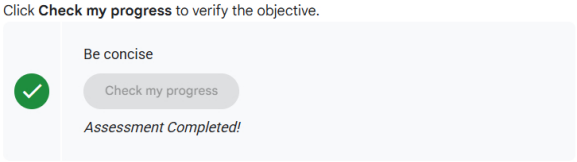
Prompt engineering is all about how to design your prompts so that the response is what you were indeed hoping to see. The idea of using "unfancy" prompts is to minimize the noise in your prompt to reduce the possibility of the LLM misinterpreting the intent of the prompt. Below are a few guidelines on how to engineer "unfancy" prompts.

In this section, you'll cover the following best practices when engineering prompts:

- Be concise
- Be specific, and well-defined
- Ask one task at a time
- Improve response quality by including examples

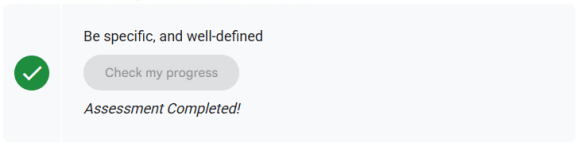
1. Run through the **Be concise** section of the notebook.

Click **Check my progress** to verify the objective.

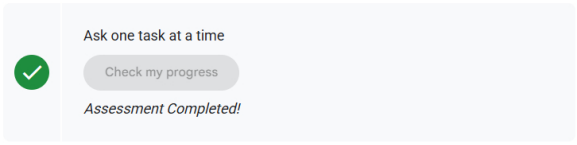



2. Run through the **Be specific, and well-defined** section of the notebook.

Click **Check my progress** to verify the objective.

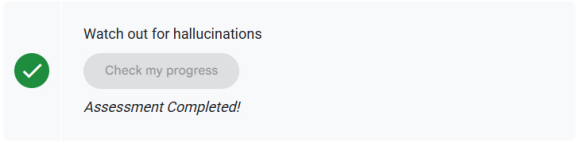



3. Run through the **Ask one task at a time** section of the notebook.




4. Run through the **Watch out for hallucinations** section of the notebook.

Click **Check my progress** to verify the objective.

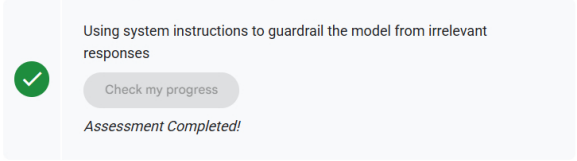



## Task 4. Reduce Output Variability

How can you attempt to reduce the chances of irrelevant responses and hallucinations? One way is to provide the LLM with system instructions. In this section, you will see how system instructions works and how you can use them to reduce hallucinations or irrelevant questions for a travel chatbot.

1. Run through the **Using system instructions to guardrail the model from irrelevant responses** section of the notebook.

Click **Check my progress** to verify the objective.




2. Run through the **Turn generative tasks into classification tasks to reduce**

Click **Check my progress** to verify the objective.

Generative tasks lead to higher output variability

Check my progress

*Assessment Completed!*

3. Run through the **Classification tasks reduces output variability** section of the notebook.

Click **Check my progress** to verify the objective.

Classification tasks reduces output variability

Check my progress

*Assessment Completed!*

## Task 5. Improve Response Quality by Including Examples

Another way to improve response quality is to add examples in your prompt. The LLM learns in-context from the examples on how to respond. Typically, one to five examples (shots) are enough to improve the quality of responses. Including too many examples can cause the model to over-fit the data and reduce the quality of responses.

Similar to classical model training, the quality and distribution of the examples is very important. Pick examples that are representative of the scenarios that you need the model to learn, and keep the distribution of the examples (e.g. number of examples per class in the case of classification) aligned with your actual distribution.

1. Run through the **Improve response quality by including examples** section of the notebook.

Click **Check my progress** to verify the objective.

Improve response quality by including examples

*Assessment Completed!*

## Congratulations!

Congratulations! In this lab you learned prompt engineering best practices using Generative AI with Google Gemini. You explored use cases which follow the best practices of being concise, specific, well-define, providing examples and asking one at a time when using LLMs to generate responses.

### Next steps / learn more

Check out the following resources to learn more about Gemini:

- Generative AI on Vertex AI Documentation
- Generative AI on YouTube
- Explore the Vertex AI Cookbook for a curated, searchable gallery of notebooks for Generative AI.
- Explore other notebooks and samples in the Google Cloud Generative AI repository.

### Google Cloud training and certification

...helps you make the most of Google Cloud technologies. Our classes include technical skills and best practices to help you get up to speed quickly and continue your learning journey. We offer fundamental to advanced level training, with on-demand, live, and virtual options to suit your busy schedule. Certifications help you validate and prove your skill and expertise in Google Cloud technologies.

**Manual Last Updated May 15th, 2025**

**Lab Last Tested May 15th, 2025**