**LAB 1: Generative AI with Vertex AI: Prompt Design**

**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**](https://cloud.google.com/vertex-ai/generative-ai/docs/learn/prompts/introduction-prompt-design)**.**

**Gemini**

[**Gemini**](https://deepmind.google/technologies/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**

**The Gemini API in Vertex AI provides a unified interface for interacting with Gemini 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**](https://cloud.google.com/vertex-ai/generative-ai/docs/learn/models#gemini-models)**.**

**Gemini Models**

* [**Gemini Pro**](https://deepmind.google/technologies/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**](https://deepmind.google/technologies/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).**

**Prerequisites**

**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**](https://cloud.google.com/vertex-ai/docs/workbench/introduction)**.**

**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:**
  + **Ideation**
  + **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 simulation or demo environment. It does so by giving you new, temporary credentials you use to sign in and access Google Cloud for the duration of the lab.**

**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**

1. **Click the Start Lab button. If you need to pay for the lab, a dialog opens for you to select your payment method. On the left is the Lab Details pane with the 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.**

1. **If necessary, copy the Username below and paste it into the Sign in dialog.**

**"Username"**

**Copied!**

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

1. **Click Next.**
2. **Copy the Password below and paste it into the Welcome dialog.**

**"Password"**

**Copied!**

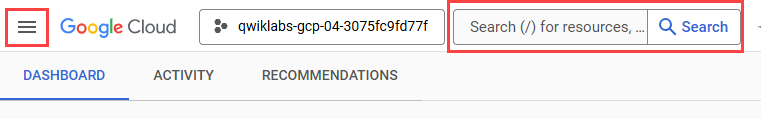
**You can also find the Password in the Lab Details pane.**

1. **Click Next.**

**Important: You must use the credentials the lab provides you. Do not use your Google Cloud account credentials.Note: Using your own Google Cloud account for this lab may incur extra charges.**

1. **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 (Navigation menu icon), click Vertex AI > Workbench.**
2. **Find the Workbench instance name instance and click on the Open JupyterLab button.**

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

**Note: If you do not see notebooks in JupyterLab, please follow these additional steps to reset the instance:**

**1. Close the browser tab for JupyterLab, and return to the Workbench home page.**

**2. Select the checkbox next to the instance name, and click Reset.**

**3. After the Open JupyterLab button is enabled again, wait one minute, and then click Open JupyterLab.**

**Task 2. Set up the notebook**

1. **Open the notebook name 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 Project ID, and for Location, use Region.**

**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.**

**Install packages and import libraries**

**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**
* **Turn generative tasks to classification tasks to improve safety**

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

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

**Be concise**

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

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

**Be specific, and well-defined**

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

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

**Ask one task at a time**

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

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

**Watch out for hallucinations**

**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**](https://cloud.google.com/vertex-ai/generative-ai/docs/learn/prompts/system-instruction-introduction)**. 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.**

**Using system instructions to guardrail the model from irrelevant responses**

1. **Run through the Turn generative tasks into classification tasks to reduce output variability section of the notebook.**

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

**Generative tasks lead to higher output variability**

1. **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**

**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**

**Congratulations!**

**Congratulations! In this lab you learned prompt engineering best practices using Generative AI with 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:**

* [**Gemini Overview**](https://deepmind.google/technologies/gemini/)
* [**Generative AI on Vertex AI Documentation**](https://cloud.google.com/vertex-ai/docs/generative-ai/learn/overview)
* [**Generative AI on YouTube**](https://www.youtube.com/@googlecloudtech/)
* **Explore the Vertex AI**[**Cookbook**](https://cloud.google.com/vertex-ai/generative-ai/docs/cookbook)**for a curated, searchable gallery of notebooks for Generative AI.**
* **Explore other notebooks and samples in the**[**Google Cloud Generative AI repository**](https://github.com/GoogleCloudPlatform/generative-ai)**.**

**Google Cloud training and certification**

**...helps you make the most of Google Cloud technologies.**[**Our classes**](https://cloud.google.com/training)**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**](https://cloud.google.com/certification/)**help you validate and prove your skill and expertise in Google Cloud technologies.**

**Manual Last Updated July 11, 2025**

**Lab Last Tested July 11, 2025**

**Copyright 2025 Google LLC. All rights reserved. Google and the Google logo are trademarks of Google LLC. All other company and product names may be trademarks of the respective companies with which they are associated.**

**++++++**

**LAB 2: Getting Started with Vertex AI Studio**

**Objectives**

In this lab, you learn how to:

* Create applications from prompts.
* Design effective prompts.
* Engineer and manage prompts.
* Use multimodal prompts.

**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 simulation or demo environment. It does so by giving you new, temporary credentials you use to sign in and access Google Cloud for the duration of the lab.

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

1. Click the **Start Lab** button. If you need to pay for the lab, a dialog opens for you to select your payment method. On the left is the Lab Details pane with the 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**.

1. If necessary, copy the **Username** below and paste it into the **Sign in** dialog.

"Username"

Copied!

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

1. Click **Next**.
2. Copy the **Password** below and paste it into the **Welcome** dialog.

"Password"

Copied!

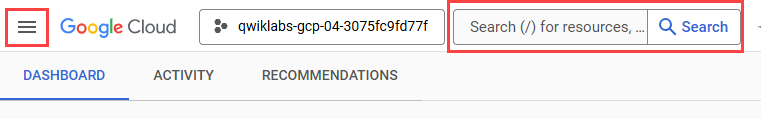
You can also find the Password in the Lab Details pane.

1. Click **Next**.

**Important:**You must use the credentials the lab provides you. Do not use your Google Cloud account credentials.**Note:**Using your own Google Cloud account for this lab may incur extra charges.

1. 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.

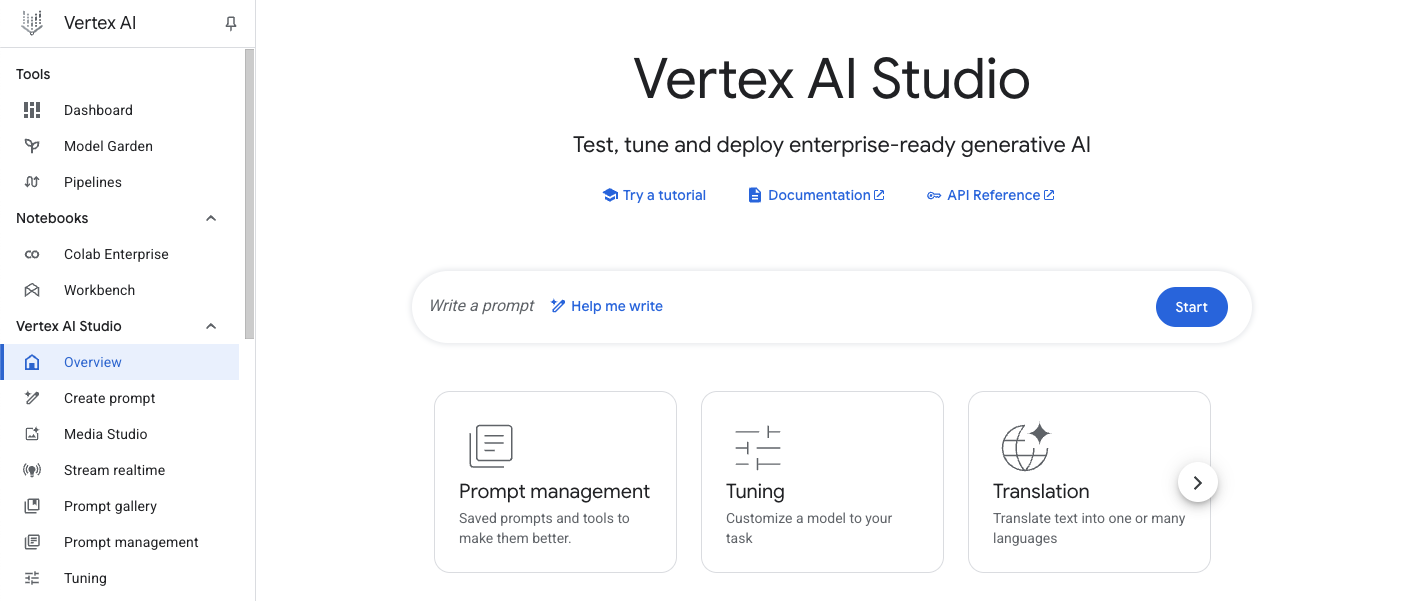
After a few moments, the Google Cloud console opens in this tab.

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

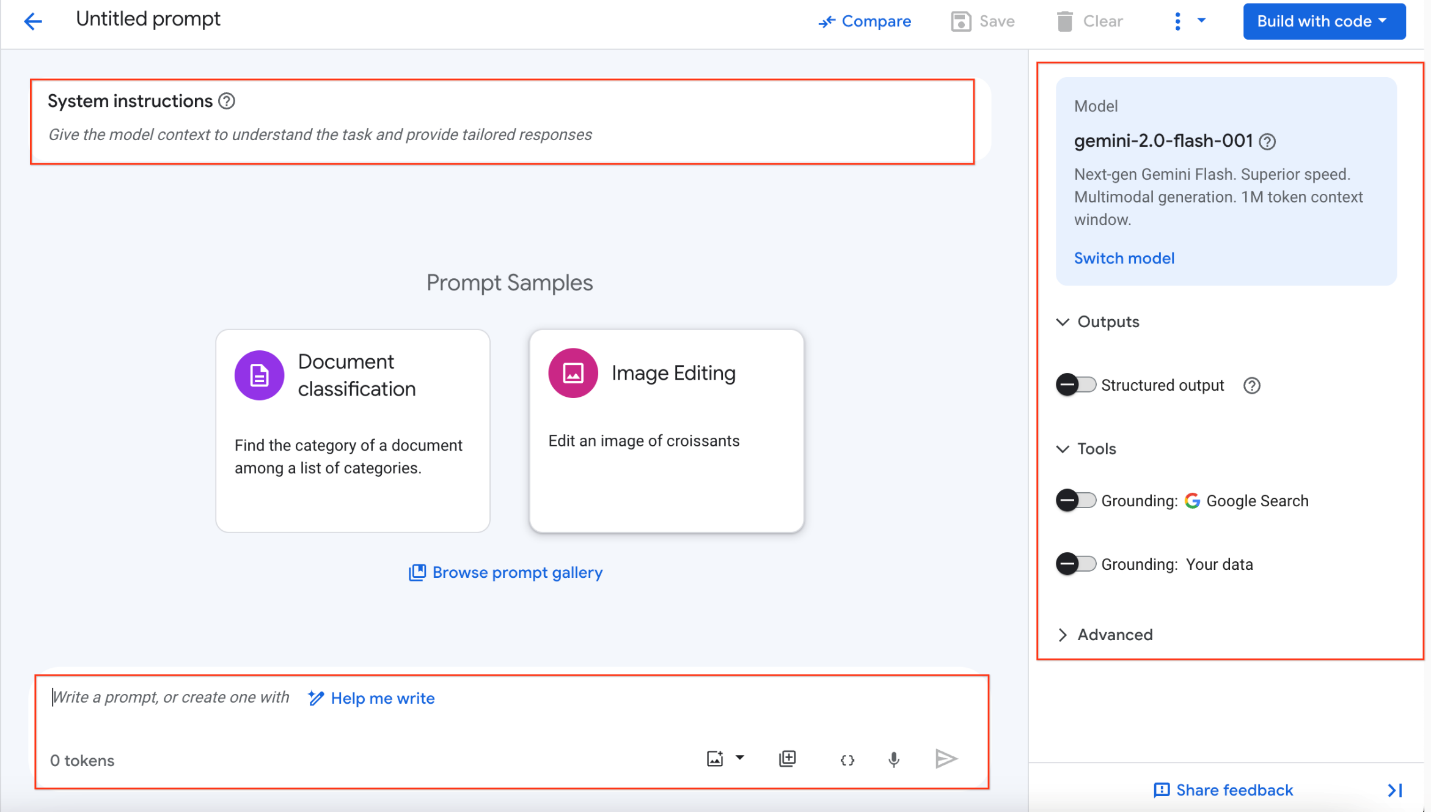
**Task 1. Create applications from prompts**

In this task, you'll see how quickly you can take an idea for a generative AI assistant and turn it into a working prototype using Vertex AI Studio. You'll focus on our insurance use case: creating a prompt that helps an insurance professional summarize client information for a risk analysis report and then prepare this prompt as a simple application.

1. In the Google Cloud console, from the **Navigation menu** (Navigation menu), select **Vertex AI** > **Vertex AI Studio** > **Overview**.



1. From the Vertex AI menu, under **Vertex AI Studio**, select **Create prompt**. This will bring you to the prompt editor page.



The UI contains three main sections:

* **System Instructions** (located at the top): a set of instructions that the model processes before it processes prompts. When a system instruction is set, it applies to the entire request. It works across multiple user and model turns when included in the prompt. We recommend that you use system instructions to tell the model how you want it to behave and respond to prompts.
* **Configuration** (located on the right): This section allows you to select models (including 3rd party models), configure parameters, use Tools (such as grounding), and set advanced options.
* **Prompt** (located at the bottom): Here, you can create a prompt that utilizes multimodal capabilities.

1. Once the new untitled prompt page loads, click on **Untitled Prompt** in the top left corner and rename your prompt to Insurance Risk Summary - Prototype.
2. In the main canvas, click into the **System instructions** text box and enter the following to give your AI assistant a role relevant to our insurance scenario:
3. You are an expert AI assistant for an insurance underwriting department.
4. Your primary goal is to help underwriters by accurately and concisely summarizing client information and highlighting potential risk factors.
5. Maintain a professional and objective tone.

Focus only on the information provided in the prompt. Do not invent details.

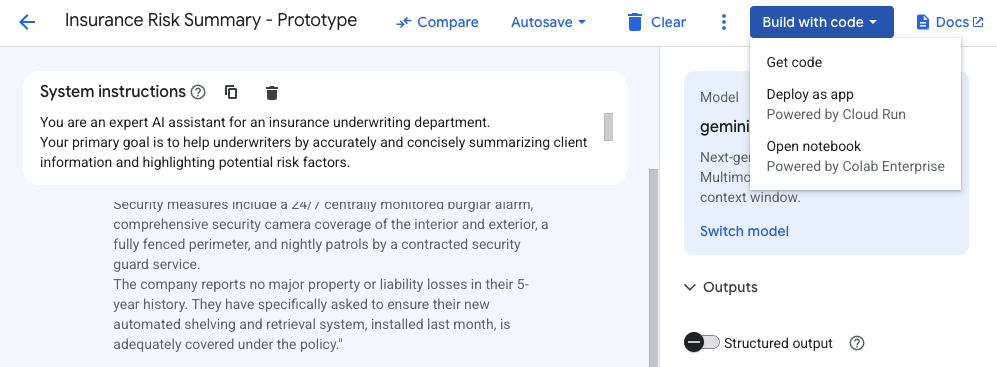
Copied!

1. Below the system instructions, in the main prompt area at the bottom of the page, paste the following:
2. Customer Notes for 'SafeHarbor Warehousing':
3. "The applicant is seeking coverage for their 50,000 sq ft warehouse. The business is 5 years old. The building is a concrete tilt-up structure, originally built in 2010. They store a variety of non-hazardous dry goods.
4. Fire safety measures include a full sprinkler system, a centrally monitored fire alarm, and documented annual inspections by a certified third party.
5. Security measures include a 24/7 centrally monitored burglar alarm, comprehensive security camera coverage of the interior and exterior, a fully fenced perimeter, and nightly patrols by a contracted security guard service.
6. The company reports no major property or liability losses in their 5-year history. They have specifically asked to ensure their new automated shelving and retrieval system, installed last month, is adequately covered under the policy."
7. Your Task:
8. 1. Briefly summarize the key details of the 'SafeHarbor Warehousing' business and its existing safety measures.
9. 2. Based \*only\* on the notes provided, identify any immediate questions an underwriter should ask or potential risk factors they should consider further.

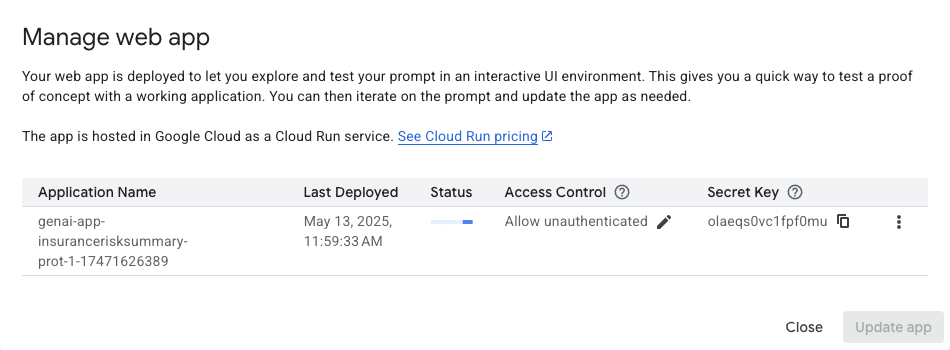
Present the summary first, then the questions/risk factors as bullet points.

Copied!

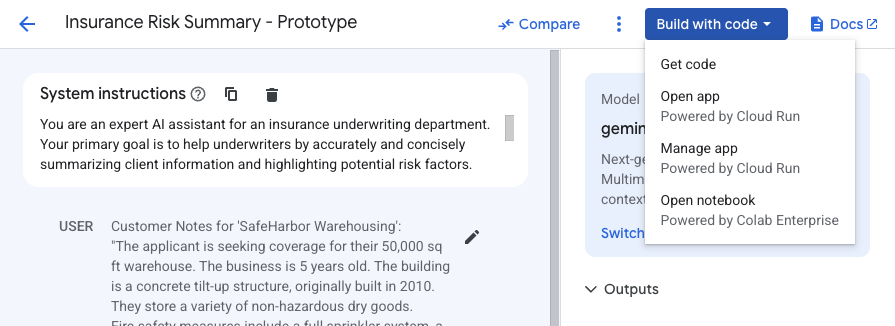
1. In the **Configuration** section on the right:
   * Ensure the **model name** model is selected. You can click **Switch model** if you need to change it.
   * If not already set, toggle the **Advanced** options, and for **Region**, select **Region**.
2. Click the **Submit** arrow button (typically at the bottom right of the prompt input area or by pressing CTRL+Enter). Review the model's response.
3. At the top of the page, click the **Save** button. In the "Save prompt" dialog, the name Insurance Risk Summary - Prototype should be pre-filled. Confirm the **Region** is correct (**Region**) and click **Save**.
4. Now, you'll explore how this drafted prompt can be turned into a prototype application. At the top right of the page, click the **Build with code** button. From the dropdown menu that appears, select **Deploy as app (Powered by Cloud Run)**.



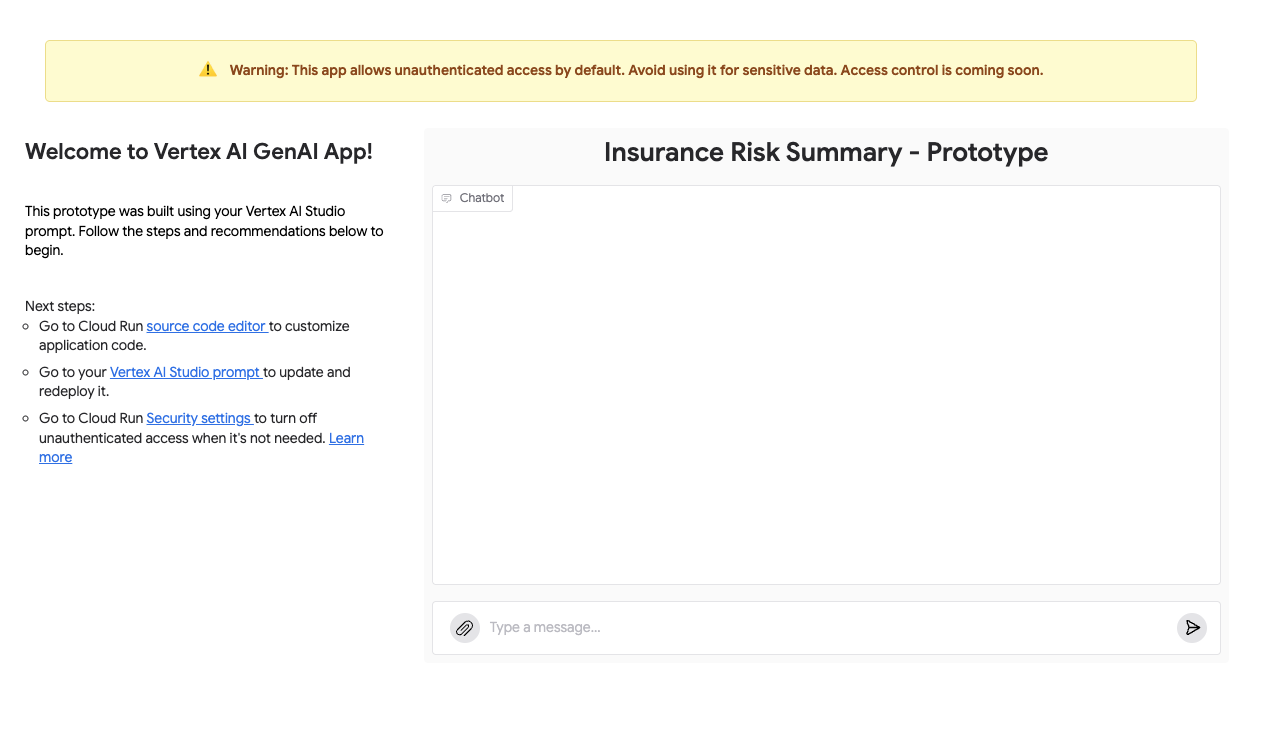
1. In the "Deploy to Cloud Run" dialog that appears:
   * You may need to **Enable services** if prompted (e.g., Cloud Build API, Cloud Run API). Click **Enable** if necessary and wait for the services to be enabled.
   * Check the **Acknowledgement** for publicly deploying your app.
   * Click **Create App**.
2. The deployment process will begin and might take a few minutes. You may see status updates in the UI that resemble the following:



1. Once completed, click the **Close** button in the **Manage web app** box. Then, to open your newly deployed application, click the **Build with code** button again at the top right of the page. From the dropdown menu, now select **Open app (Powered by Cloud Run)**. This will open your "Vertex AI GenAI App" in a new browser tab.



1. You should now see a page titled "Welcome to Vertex AI GenAI App!" with your prompt title, "Insurance Risk Summary - Prototype," displayed.



1. In the "Chatbot" section, in the "Type a message..." input field at the bottom, enter a new test message. For example:
2. New Customer Inquiry:
3. "Applicant 'Coastal Goods Delivery' has a fleet of 10 delivery vans, all equipped with GPS and telematics. They operate within a 100-mile radius of their depot. Drivers undergo annual safety training. They had one minor fender bender last year, no injuries, $1500 damages. What are the primary risk considerations?"

Please summarize key points and identify potential risks.

Copied!

1. Click the submit arrow button to send your message to the application.
2. Observe the response from your deployed GenAI app. It should process your input based on the logic and system instructions you defined in Vertex AI Studio.

**Note:** As indicated by the warning on the app page, this application allows unauthenticated access by default. In a production scenario, you would configure appropriate security settings. For this lab, the default is fine for exploration.

1. You have now completed the full cycle:
   * Designed a prompt in Vertex AI Studio.
   * Deployed it as a serverless application using Cloud Run with a few clicks.
   * Directly opened and interacted with your generative AI model through a web interface. This demonstrates the power of Vertex AI Studio for rapid prototyping and deployment of generative AI capabilities.

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

Create a prompt application with Vertex AI Studio.

**Task 2. Design effective prompts**

In Task 1, you prototyped an initial prompt. Now, you'll dive deeper into refining prompts to get more precise, controlled, and useful outputs from the generative models. This is a core skill in prompt engineering. You will continue with your insurance theme by trying to extract specific information from a claims document or improving the summarization quality.

Zero-shot prompting

You'll start by creating a new prompt to explore detailed prompt design.

1. Ensure you are in the main **Vertex AI Studio** area. If you were viewing your deployed app from Task 1, close that browser tab to return to the Google Cloud console.
2. From the **Navigation menu** (Navigation menu), select **Vertex AI** > **Vertex AI Studio** > **Create prompt**.
3. If you started with a new prompt page (showing "Prompt Samples"), click on **Untitled Prompt** in the top left corner and rename it Insurance Claim Data Extraction.
4. Familiarize yourself with the scenario for this section: *An insurance adjuster often receives unstructured notes or emails about a new claim and needs to quickly extract key pieces of information to enter into their claims management system.*
5. In the **System instructions** box, enter the following:
6. You are an AI assistant specializing in parsing and extracting specific data points from unstructured insurance claim notifications.
7. Your goal is to identify and list key information accurately.
8. If a piece of information is not found, clearly state "Not found".

Output the extracted information in a key: value format, with each key on a new line.

Copied!

1. In the main prompt area, paste the following example of an unstructured claim note:
2. Claim Notification Received:
3. "Hi team, just got a call from Mrs. Eleanor Vance, policy #POL458892. She reported a kitchen fire that occurred on May 12th, 2025, around 3 PM. The main damage seems to be to the oven and surrounding cabinets. She mentioned smoke damage in the kitchen and dining area too. She thinks the total damage might be around $7,500. Her contact is 555-0123. No injuries reported, thankfully."
4. Extract the following:
5. - Policy Number
6. - Claimant Name
7. - Date of Loss
8. - Time of Loss
9. - Type of Loss
10. - Brief Description of Damage
11. - Estimated Loss Amount

- Injuries Reported

Copied!

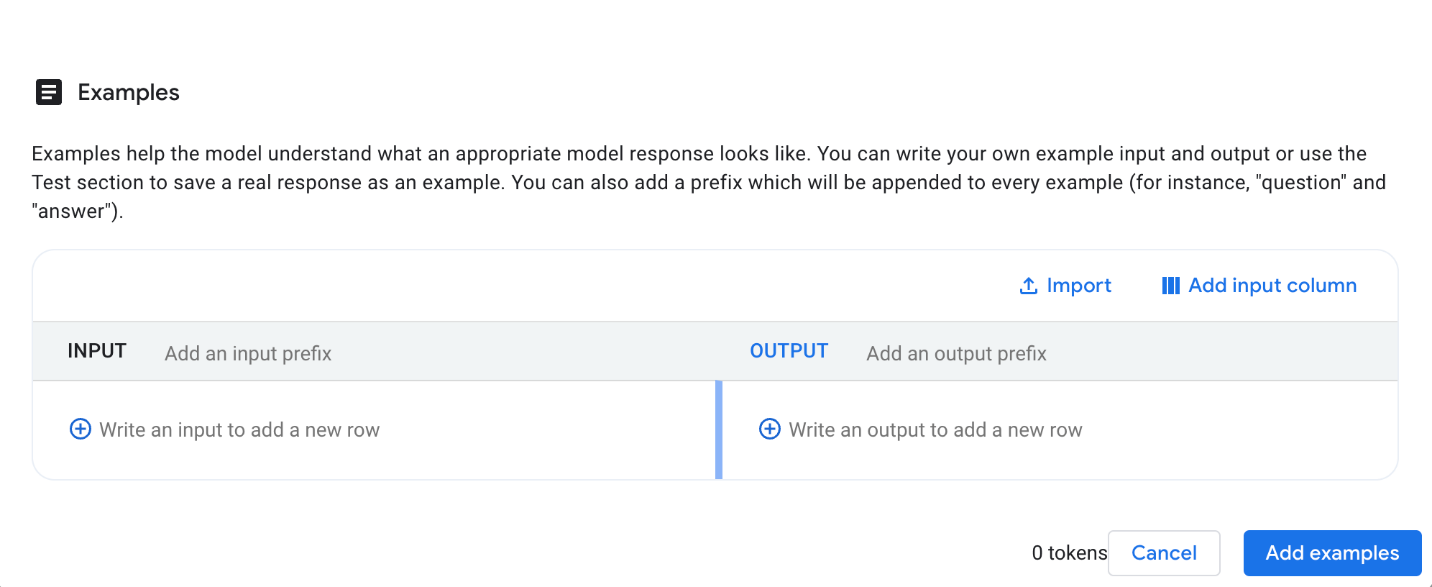
1. In the **Configuration** section on the right:
   * Select the **model name** model.
   * Set **Temperature** to 0.1 (for more factual, less creative extraction).
   * Set **Output token limit** to a reasonable number, like 1024.
   * Ensure the **Region** is **Region**.
2. Click the **Submit** arrow button. Review the output. This first attempt without explicit examples is called **zero-shot prompting**.
3. Once you are done reviewing the response from your zero-shot attempt, click the **Clear** icon on the top toolbar to clear the entire prompt canvas. This is necessary because examples can typically only be added to a fresh prompt.

Few-Shot prompting

Often, providing a few examples (few-shot prompting) can significantly improve the model's performance, especially for specific formatting or nuanced extraction.

1. On the bottom right of the **Prompt** section, click the **Add Examples** (Add Examples button) button.

This will open a new window where you can add examples for the prompt.



1. In the "Examples" interface that appears:
   * For the **INPUT** of your first example, paste the following unstructured note:
2. Claim Notification Received:
3. "Email from John Sterling (policy POL77521) re: water damage at his shop. Happened sometime last night, May 10th, 2025. A pipe burst in the ceiling. Stockroom is flooded, some damage to inventory. He's not sure on the cost yet, maybe $5k-$10k? No one was there, so no injuries."
4. Extract the following:
5. - Policy Number
6. - Claimant Name
7. - Date of Loss
8. - Time of Loss
9. - Type of Loss
10. - Brief Description of Damage
11. - Estimated Loss Amount

- Injuries Reported

Copied!

* + For the **OUTPUT** of your first example, paste the following perfectly formatted extraction:

Policy Number: POL77521

Claimant Name: John Sterling

Date of Loss: May 10th, 2025

Time of Loss: Night

Type of Loss: Water damage

Brief Description of Damage: Pipe burst in ceiling, stockroom flooded, some damage to inventory.

Estimated Loss Amount: $5,000 - $10,000

Injuries Reported: No

Copied!

* + Click the **Add examples** button to save this example and return to the main prompt.

1. **Re-add System Instructions**: Since clearing the prompt also cleared the system instructions, paste them again into the System instructions box at the top:
2. You are an AI assistant specializing in parsing and extracting specific data points from unstructured insurance claim notifications.
3. Your goal is to identify and list key information accurately.
4. If a piece of information is not found, clearly state "Not found".

Output the extracted information in a key: value format, with each key on a new line.

Copied!

1. **Provide the New Input and the Prompt:**
   * In the area labeled **(Input) Write value here**, paste the original claim notification for Mrs. Eleanor Vance that you want the model to process now:
2. Claim Notification Received:

"Hi team, just got a call from Mrs. Eleanor Vance, policy #POL458892. She reported a kitchen fire that occurred on May 12th, 2025, around 3 PM. The main damage seems to be to the oven and surrounding cabinets. She mentioned smoke damage in the kitchen and dining area too. She thinks the total damage might be around $7,500. Her contact is 555-0123. No injuries reported, thankfully."

Copied!

1. In the area labeled **Write your prompt here** (below the Input field), you need to provide the instruction for the model. This tells the model what to do with the (Input) text, using the Examples as a guide. Enter the following:
2. Extract the following data points from the provided claim notification:
3. - Policy Number
4. - Claimant Name
5. - Date of Loss
6. - Time of Loss
7. - Type of Loss
8. - Brief Description of Damage
9. - Estimated Loss Amount

- Injuries Reported

Copied!

1. Ensure your configurations (Model, Temperature, Token Limit, Region) are still set as desired (e.g., Temperature at 0.1).
2. Click the **Submit** arrow button again. Compare this new output to the previous zero-shot attempt. Note if the accuracy or formatting has improved significantly due to the few-shot example and the structured input method.

Experimenting with prompt configurations

Now, you'll explore how different parameters in the **Configuration** panel (on the right) can affect the model's response. Ensure your "Insurance Claim Data Extraction" prompt with the few-shot example is active.

1. Experiment with **Temperature**:
   * **Explanation:** Temperature controls randomness. Lower values (e.g., 0.0-0.2) make the output more focused and deterministic. Higher values (e.g., 0.7-1.0) encourage more diverse or creative responses.
   * **Try it:** Change **Temperature** to 0.7. Click **Submit** and note any changes. Then, change **Temperature** back to 0.1.
2. Experiment with **Output Token Limit**:
   * **Explanation:** This sets the maximum number of tokens (parts of words) the model can generate for its response.
   * **Try it:** Set **Output token limit** to a very small number, like 20. Click **Submit** and observe the truncated output. Reset it to a suitable value (e.g., 1024, or the default).
3. Experiment with **Top-P**:
   * **Explanation:** Top-P (nucleus sampling) also controls randomness. It considers only the most probable tokens whose combined probability mass exceeds the Top-P value. A value of 1.0 considers all tokens. Lowering Top-P (e.g., to 0.8) makes the output more focused, similar to lowering temperature.
   * **Try it:** With Temperature at 0.1 (or slightly higher, like 0.5, to better observe Top-P effects), set **Top-P** to 0.8. Click **Submit**. Then set **Top-P** to 1.0, click **Submit**, and observe if there are subtle differences.
4. Briefly review other settings in the **Advanced** configuration panel:
   * **Safety Filter Settings:** These are active by default to help block harmful content. For this lab, you'll use the default settings.

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

Prompt engineering in Vertex AI Studio.

**Task 3. Engineer and manage prompts**

Once you have a working prompt, you'll often want to experiment with changes to the instructions or model configurations to see if you can improve the response. The "Compare" feature in Vertex AI Studio is designed for this. For this section, you'll use the prompt we just created.

1. Ensure you are in the main **Vertex AI Studio** area. If you were viewing your deployed app from Task 1, close that browser tab to return to the Google Cloud console.
2. Navigate to create a new prompt. From the **Navigation menu** (Navigation menu), select **Vertex AI** > **Vertex AI Studio** > **Create prompt**.
3. Click on **Untitled Prompt** in the top left corner and rename this new prompt to Comparison Base - Restaurant Risks.
4. Set up this simple base prompt:
   * In the **System instructions** box, enter:

You are an insurance risk analyst assistant. Your task is to identify potential risk factors from a given scenario. Be concise.

Copied!

* + In the main prompt area (where you "Write your prompt here"), paste the following:

Scenario:

"The applicant, 'The Fiery Grill,' is a new upscale restaurant specializing in wood-fired oven pizzas and open-flame grilling. They have installed a brand new, custom-built fire suppression system for their cooking area, but it has not yet been certified by a third party. The restaurant plans to feature live acoustic music on weekend evenings and has a small, raised stage area. They also want to offer valet parking."

Based on this scenario, list three primary risk factors an underwriter should consider.

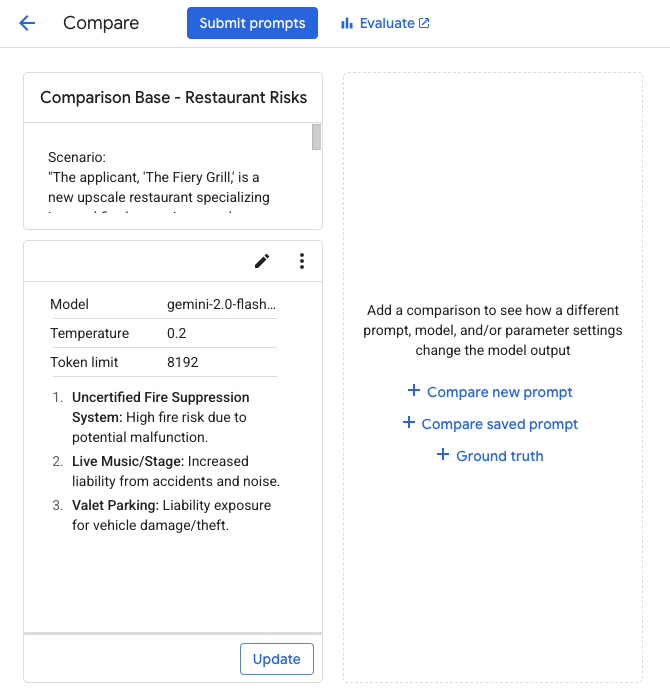
Copied!

* + In the **Configuration** section on the right:
    - Select the **model name** model.
    - Set **Temperature** to 0.2.
    - Ensure the **Region** is **Region**.

1. Click the **Submit** arrow button. Review the model's initial response.
2. Click the **Save** button. Confirm the name Comparison Base - Restaurant Risks and save it. You generally need to save a prompt before you can effectively use it in some comparison workflows, especially if you plan to compare *with a saved prompt* later.
3. Now, with this Comparison Base - Restaurant Risks prompt and its response displayed, click the **Compare** button on the top toolbar.

**Note:** If prompted, **Exit without saving** then click **Continue**.

1. The "Compare" interface will open. Your Comparison Base - Restaurant Risks prompt, its configurations, and its latest response will typically be displayed in a column on the left.



Comparing by modifying prompt instructions

You'll now see how changing the instructions affects the output for "The Fiery Grill." In the **Compare** view, the comparison panes may not have a separate "System Instructions" field; if so, you'll prepend any system-level guidance to the main prompt.

1. In the central area of the **Compare** interface (or to the right of your first prompt), click the **+ Compare new prompt** button.
2. A new prompt editing pane will appear on the right.
3. Configure this new (second) prompt as a variation:
   * In the single large text box for the prompt in this right-hand pane, paste the following combined instructions and scenario:
4. You are an expert insurance risk analyst assistant. Your task is to identify potential risk factors from a given scenario. For each risk factor, also briefly suggest a potential mitigation strategy or question for the underwriter. Be clear and structured.
5. Scenario:
6. "The applicant, 'The Fiery Grill,' is a new upscale restaurant specializing in wood-fired oven pizzas and open-flame grilling. They have installed a brand new, custom-built fire suppression system for their cooking area, but it has not yet been certified by a third party. The restaurant plans to feature live acoustic music on weekend evenings and has a small, raised stage area. They also want to offer valet parking."

Based on this scenario, list three primary risk factors an underwriter should consider.

Copied!

* + **Configurations for this variation:**
    - Ensure the **Model** is the same as the first prompt: **model name**.
    - Keep **Temperature** at 0.2.
    - Keep other configurations (Token Limit, Region, etc.) identical to the first prompt to isolate the effect of the instruction change.
    - Scroll down in the pane and click **Apply**.

1. Once the new prompt variation is set up with its text and configurations in the right-hand pane, click the **Submit prompts** button (usually located at the top of the "Compare" interface).
2. Wait for both prompts to generate responses. Review the two responses side-by-side. Does the second prompt now include mitigation strategies or questions, due to the modified instructions?
3. To save this as a new prompt, Click **Save as new**, then enter your desired *[Prompt name]*, and then click **Save**.

Comparing with a different temperature setting

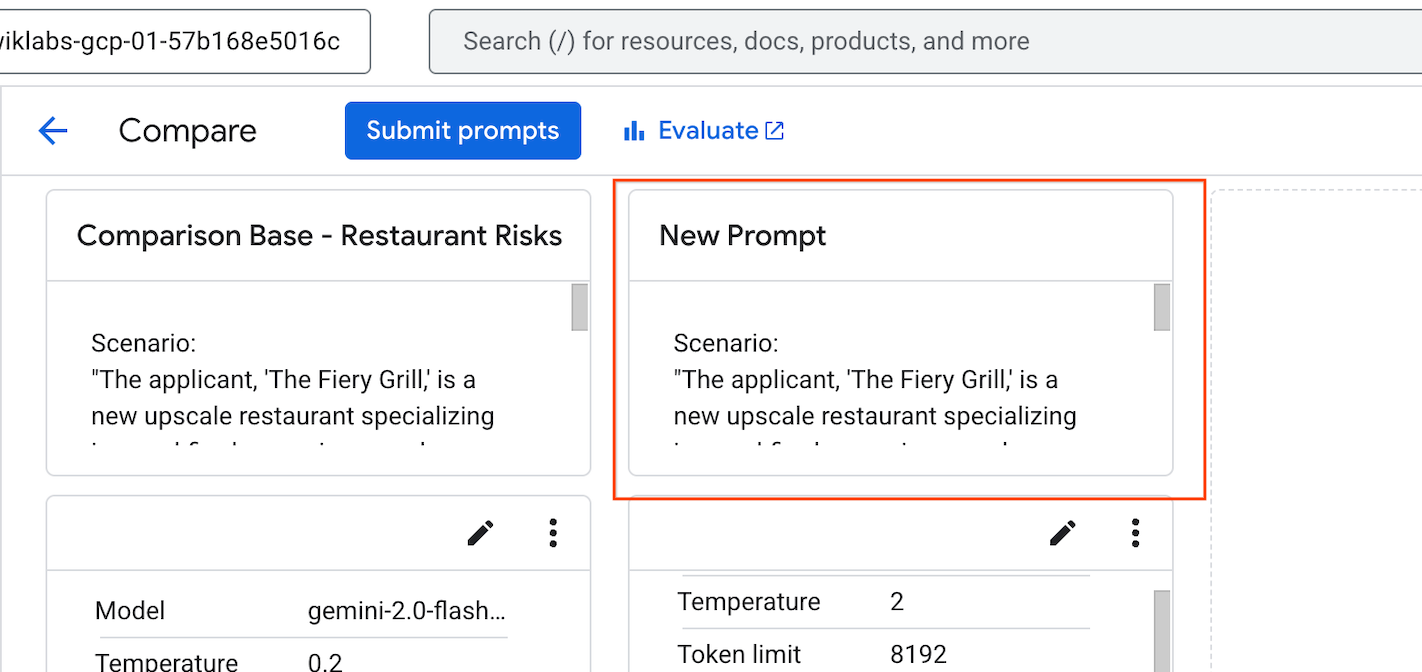
Now, you'll use a comparison pane to test a different temperature.

1. In the text box inside of the second prompt pane (the one on the right),
   * **Prompt Text:** Revert the prompt text to the original, simpler version (without the added system instructions about mitigation). Copy and paste the prompt from your Comparison Base - Restaurant Risks (left pane):
2. Scenario:
3. "The applicant, 'The Fiery Grill,' is a new upscale restaurant specializing in wood-fired oven pizzas and open-flame grilling. They have installed a brand new, custom-built fire suppression system for their cooking area, but it has not yet been certified by a third party. The restaurant plans to feature live acoustic music on weekend evenings and has a small, raised stage area. They also want to offer valet parking."

Based on this scenario, list three primary risk factors an underwriter should consider.

Copied!

\*If the compare view doesn't have a system instruction box for this pane, ensure the system instructions "You are an insurance risk analyst assistant..." are prepended here if they were cleared, or rely on the original system instructions if they apply globally from the left pane.



1. Click the **Edit** (pencil) icon in the second prompt pane below the text box:
   * **Configuration Change:** In the configuration settings for this second prompt pane, change the **Temperature** to 2.0. Ensure the Model is still **model name**.
   * Scroll down in the pane and click **Apply**.
2. Click **Submit prompts** again.
3. Observe the differences in the responses. Does the higher temperature (2.0) in the second prompt cause the list of risk factors to be less focused, more speculative, or significantly different compared to the 0.2 temperature output? **Note:** setting the temperature this high will likely result in less coherent or relevant output, but it demonstrates the extreme effect of the parameter.
4. To update the changes, click the **Update** button located below the second prompt pane.

Comparing different models and configurations

Now you will try comparing your base model with a different model and settings to observe differences in reasoning or output style.

1. In the text box inside of the second prompt pane (the one on the right),
   * **Prompt Text:** Use the same original "Fiery Grill" scenario and request for three risk factors as in your Comparison Base - Restaurant Risks (left pane). *(Again, ensure the base system instruction is effectively present for this pane, either by prepending it if necessary or if the UI carries it over implicitly).*
2. Click the **Edit** (pencil) icon in the second prompt pane below the text box:
   * **Configuration Changes for this variation:**
     + Change the **Model** to **model name** (select this from the model dropdown for this pane).
     + Set the **Temperature** to 0.2.
     + Set the **Output token limit** to 65535 (or the maximum allowed by the model in the UI).
     + Scroll down in the pane and click **Apply**.
3. Click **Submit prompts**.
4. Review the responses. Compare the output from **model name** (left pane) with **model name** (right pane).

Notice any differences in the identified risk factors, the detail provided, the structure of the response, or how it seems to reason through the scenario.

1. Make sure to click on the **Update** button located below the second prompt pane, to store the new changes.

Other comparison options (brief overview)

Vertex AI Studio offers other ways to add prompts for comparison:

1. Notice the **+ Compare saved prompt** button in the area where you added a "new prompt" for comparison.
   * Clicking this would allow you to select one of your previously saved prompts from "Prompt Management" to compare against.**Note:** As indicated in the UI (and to clarify the scope of "conversation history"), prompt comparison has limitations. For instance, it does not support chat prompts, prompts with media, or prompts with conversation history involving **more than one exchange**.
2. You may also see a **+ Ground truth** button.
   * This feature allows you to input an "ideal" or "perfect" response for your prompt. If ground truth is provided, the system might offer more detailed evaluation metrics, which can be useful for advanced prompt testing but is beyond the scope of this introductory lab. For our purposes, visual side-by-side comparison is sufficient.
3. To exit the "Compare" view and return to the main prompt editing interface (for example, to continue working on the version you prefer), click the **back arrow** at the top left of the "Compare" interface.

Prompt Management

As you experiment with different instructions, examples, and configurations, it's crucial to save your work. Saving prompts allows you to:

* Organize your experiments.
* Reuse effective prompts easily.

Vertex AI Studio provides **Prompt Management** for this purpose.

**Saving a prompt**

Assume you've just finished an iteration you want to save, for example, one of the prompt variations from your comparison exercises, or a refined version of your "Insurance Claim Data Extraction" prompt.

1. Ensure the prompt you want to save is active in the main prompt editing interface (e.g., you might have just exited the "Compare" view and have a preferred version open, or you are in the Comparison Base - Restaurant Risks prompt).
2. If you have already saved your prompt, you will notice an **Autosave** button on the top toolbar. Once you initially save your prompt, **Autosave** is on by default. If you would like to disable this feature, you can click the button and select **Turn off Autosave**.

**Accessing Prompt Management**

Once you have saved prompts, you can manage them in **Prompt Management**.

1. Navigate to create a new prompt. From the **Navigation menu** (Navigation menu), select **Vertex AI** > **Vertex AI Studio** > **Prompt Management**.
2. You will see a list of your saved prompts, often including their names, media, model, and the last modified date.
3. From this page, you can view, delete, or export your saved prompts, and access optimization tools to improve your prompts.

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

Compare, evaluate, and manage prompts.

**Task 4. Use multimodal prompts with Gemini**

In this task, you'll use the main prompt interface in Vertex AI Studio with the Gemini model to analyze an image and extract information from it. This demonstrates how to design prompts for various analytical tasks such as description, text extraction, and question answering based on visual content.

1. In the Google Cloud console, from the **Navigation menu** (Navigation menu), select **Vertex AI** > **Vertex AI Studio** > **Create prompt**.
2. At the top left, click **Untitled Prompt** and rename your prompt to Timetable Image Analysis.
3. In the **Configuration** panel on the right:
   * Ensure the **model name** model is selected. You can click **Switch model** if you need to change it.
   * Toggle the **Advanced** options, and for **Region**, select **Region**.
4. Download the sample timetable image to your local machine:
   * **Right-click** the image below and save it: 
5. In the **Prompt** section (at the bottom of the page), click the **Insert Media** (Insert Media icon) button, typically found on the right side of the prompt input area.
6. In the **Select source** menu that appears, click **Upload** and then select the timetable image file you just downloaded from your computer. The image will appear directly in the prompt input area.

**Note:** You'll notice several options for sourcing your media. These typically include direct **Upload**, providing a file **By URL**, importing from **Cloud Storage** or **Google Drive**, and even linking a **YouTube video. For this lab, you'll focus on the direct upload method.**

1. Now, you can ask the model to perform a few tasks on the image. Below the inserted image in the prompt input field, paste the following prompt:

1. Provide a concise title for this image (under 5 words).

2. Describe the image in one or two sentences.

3. Extract all visible text from the image. Present the flight schedule as a clearly formatted list with columns for "Time" and "City".

Copied!

1. Click the **Submit** arrow button (bottom right of the prompt section) and review the model's response.
2. Next, you can ask a question that requires reasoning based on the extracted information. Replace the previous text prompt (leave the image in place) with the following:

Based on the flight schedule shown in the image, what percentage of the listed flights depart before 11:30 AM? Show your calculation if possible.

Copied!

1. Click the **Submit** arrow button and examine the response.
2. Briefly observe the effect of temperature. In the **Configuration** panel on the right:
   * Adjust the **Temperature** to 0.8.
   * Resubmit the *exact same prompt* from step 9 ("Based on the flight schedule... percentage...").
   * Note if the style, confidence, or detail of the explanation changes.
   * After observing, set the **Temperature** back to a lower value like 0.2 for more predictable responses.

**Note:** Temperature controls randomness. Lower values (e.g., 0.0-0.2) are good for factual responses, while higher values (e.g., 0.7+) can lead to more diverse or creative outputs, which might be less suitable for precise data extraction or analysis.

1. Once you're satisfied with your exploration, save your work. Click the **Save** button on the top toolbar.
   * The name Timetable Image Analysis should be pre-filled.
   * Confirm the **Region** is **Region**.
   * Click **Save** in the dialog.

**Note:** After clicking **Save**, allow a few moments for the prompt to be saved before proceeding if you were to navigate away immediately.

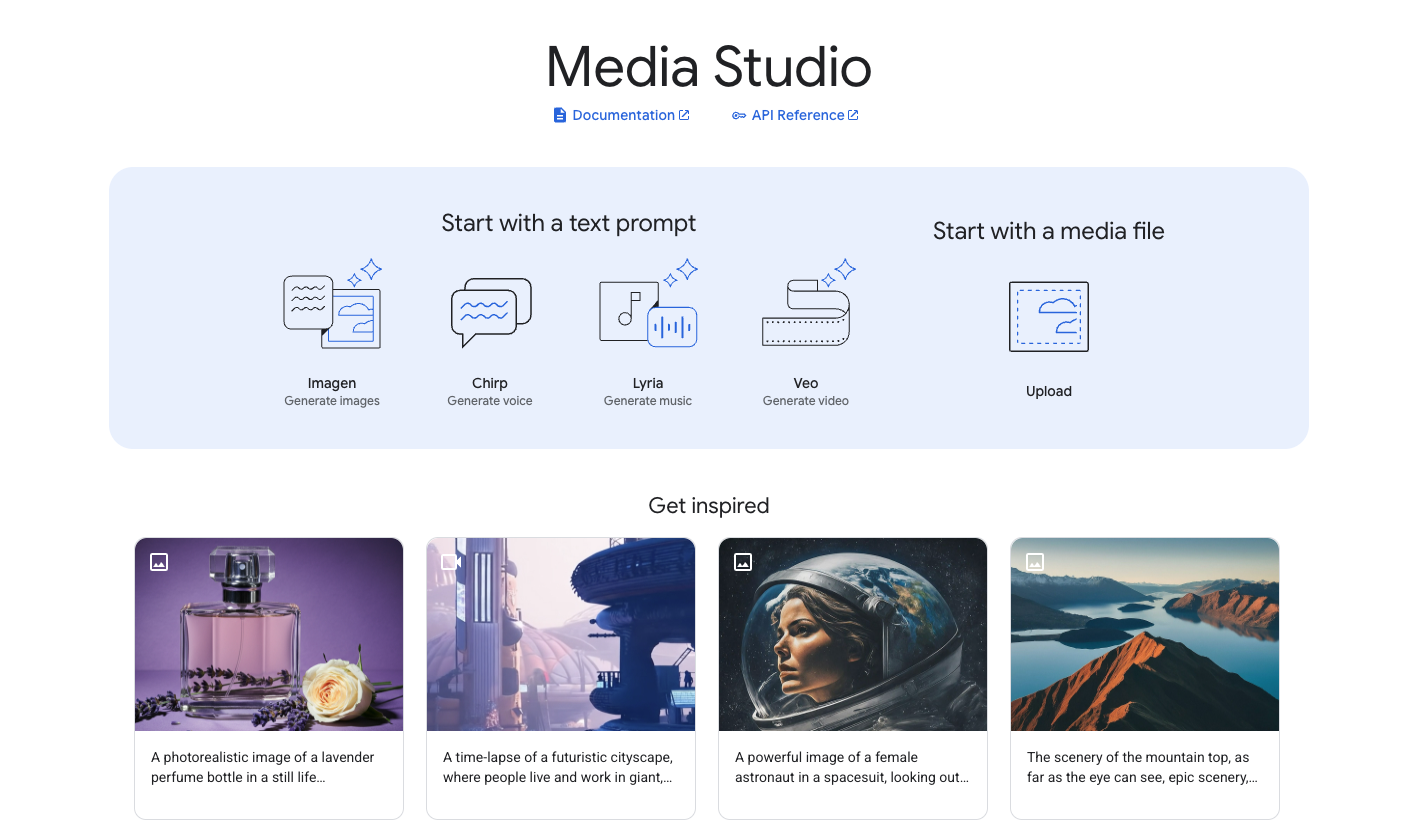
Click **Check my progress** to verify the objectives.

Analyze images with Gemini in Vertex AI Studio.

**Task 5. Explore Vertex AI Media Studio**

Beyond text, Vertex AI Studio offers powerful tools for generating various media types directly from text prompts or by refining existing media. In this task, you'll explore generating images, videos, and voice.

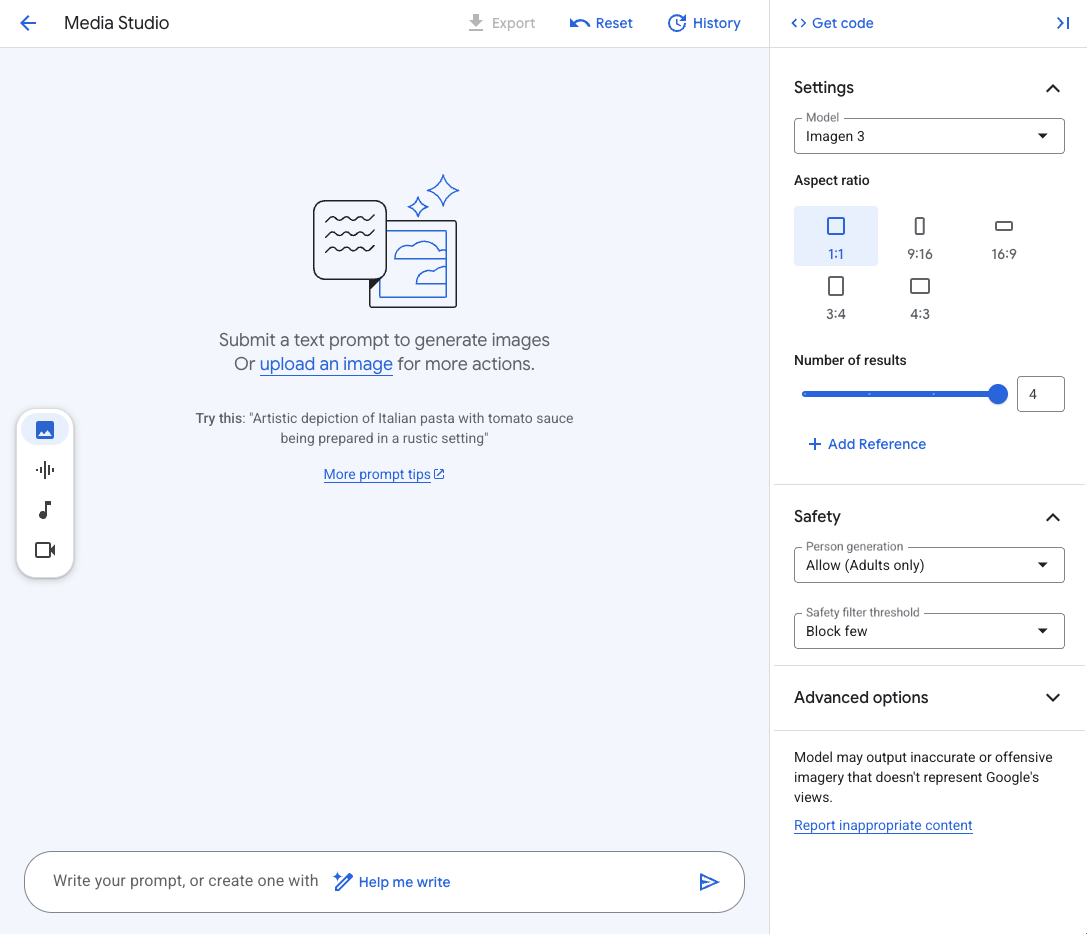
1. Ensure you are in **Vertex AI Studio**. If you're not already there, navigate from the **Navigation menu** (Navigation menu) by selecting **Vertex AI** > **Vertex AI Studio > Media Studio**. The page should resemble the following:



Generating an Image with Imagen

You'll start by generating an image.

1. On the Media Studio landing page, under the "Start with a text prompt" section, click on **Imagen - Generate images**.

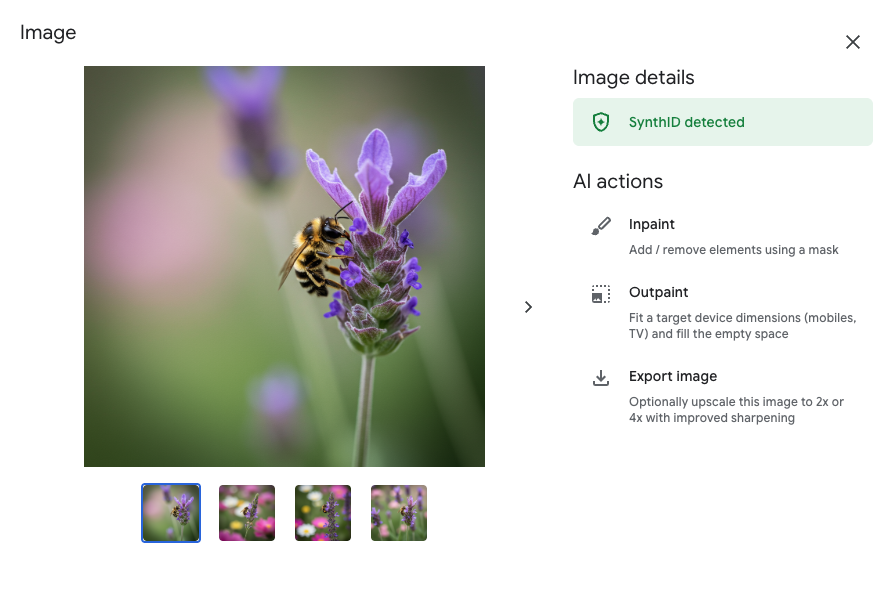


1. In the text prompt area at the bottom, enter a descriptive prompt. For example:

A close-up, photorealistic image of a single honeybee collecting pollen from a vibrant purple lavender flower, with a softly blurred garden background.

Copied!

1. In the **Settings** panel on the right:
   * Ensure the **Model** is set to Imagen 4 (or the latest available Imagen model).
   * For **Aspect ratio**, select 1:1.
   * Set **Number of results** to 4 for this first generation.
   * Review the **Safety** settings (e.g., Person generation, Safety filter threshold) and leave them at their defaults or adjust as preferred for this non-person-focused prompt.
2. Click the **Submit** button at the bottom right of the prompt area.
3. After a few moments, your generated images will appear in the main area.
4. Click on one of the generated image thumbnails to open the detail view.



1. In the **Image details** pane on the right:
   * Observe the available **AI actions** such as Inpaint (to add/remove elements using a mask), Outpaint (to extend the image), and Export image (which may offer upscaling).
   * Notice if **SynthID detected** is displayed with a green checkmark.
   * (Optional): select one of the options Inpaint or Outpaint and play around with them to see the capabilities of Imagen.

**What is SynthID?**

SynthID is a technology developed by Google DeepMind that embeds a digital watermark directly into the pixels of AI-generated images. This watermark is designed to be imperceptible to the human eye but detectable by an algorithm. Its purpose is to help identify images as AI-generated, promoting transparency and responsible AI practices, even if the image is later modified (e.g., compressed, filtered).

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

Explore Vertex AI Media Studio.

Generating Voice with Chirp (Optional)

If you wish to explore AI-generated voice:

1. In the small vertical toolbar on the far left, click the **Audio icon** (it looks like a microphone) to switch to the voice generation tool. This opens the Chirp interface.
2. You might be prompted to enable the **Cloud Text-to-Speech API** if it's not already active for your project. If so, click **Enable** and wait for the API to be enabled (this might take a moment).
3. Once the interface is ready, in the text prompt area at the bottom, enter the text you want to synthesize. For example:

Welcome to the world of generative AI on Google Cloud

Copied!

1. In the **Settings** panel on the right:
   * Select a **Model** (e.g., Chirp 3. HD Voices).
   * Choose your desired **Language** (e.g., English (US)).
   * Select a **Voice** from the dropdown list. You can try a few different ones to hear their characteristics.
   * Explore any **Advanced options** if available and desired.
2. Click the **Submit** button.
3. After processing, you should be able to play the generated audio directly in the interface.

**Congratulations!**

Congratulations! In this lab, you successfully navigated Vertex AI Studio to prototype a generative AI application for an insurance scenario, from initial design and deployment to advanced prompt engineering and comparison. You practiced refining text outputs for specific analytical tasks and explored the exciting multimodal capabilities for generating images, video, and voice. These foundational skills will empower you to build even more sophisticated generative AI solutions on Google Cloud.

Next steps / learn more

* Dive deeper into the capabilities of the generative AI models you explored (like Gemini for text and image analysis, Imagen, Veo, and Chirp) by visiting the official [Vertex AI Model Garden documentation](https://cloud.google.com/vertex-ai/generative-ai/docs/model-garden/explore-models).
* Enhance your prompt engineering skills with Google Cloud's comprehensive [Prompt design strategies guide](https://ai.google.dev/gemini-api/docs/prompting-strategies) for best practices and advanced techniques.
* Learn how to integrate and automate these generative AI functionalities into your applications by exploring the [Vertex AI SDKs documentation](https://cloud.google.com/vertex-ai/docs).
* Continue your learning journey with more [Generative AI labs and courses on Google Cloud Skills Boost](https://www.cloudskillsboost.google/paths).

Google Cloud training and certification

...helps you make the most of Google Cloud technologies. [Our classes](https://cloud.google.com/training) 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](https://cloud.google.com/certification/) help you validate and prove your skill and expertise in Google Cloud technologies.

**Manual Last Updated July 04, 2025**

**Lab Last Tested July 04, 2025**

Copyright 2025 Google LLC. All rights reserved. Google and the Google logo are trademarks of Google LLC. All other company and product names may be trademarks of the respective companies with which they are associated.

**+++++**

**LAB 3: Getting Started with Google Generative AI using GEN AI SDK**

**Overview**

The [Google Gen AI SDK](https://cloud.google.com/vertex-ai/generative-ai/docs/sdks/overview) provides a unified interface to Google's generative AI API services. This SDK simplifies the process of integrating generative AI capabilities into applications and services, enabling developers to leverage Google's advanced AI models for various tasks. In this lab, you explore the Google Gen AI SDK, learning to connect to AI services, send diverse prompts, and fine-tune responses from Gemini. You also get hands-on experience with more advanced techniques to prepare you to leverage the power of generative AI for your own projects.

**Objectives**

In this lab, you learn how to use the Google Gen AI SDK for Python to interact with Google's generative AI services and models, including Gemini. You cover the following:

* Installing the Gen AI SDK.
* Connecting to an API service.
* Sending text and multimodal prompts.
* Setting system instructions.
* Configuring model parameters and safety filters.
* Managing model interactions (multi-turn chat, content streaming, asynchronous requests).
* Using advanced features (token counting, context caching, function calling, batch prediction, text embeddings).

Prerequisites

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](https://cloud.google.com/vertex-ai/docs/workbench/introduction).

**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 simulation or demo environment. It does so by giving you new, temporary credentials you use to sign in and access Google Cloud for the duration of the lab.

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

1. Click the **Start Lab** button. If you need to pay for the lab, a dialog opens for you to select your payment method. On the left is the Lab Details pane with the 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**.

1. If necessary, copy the **Username** below and paste it into the **Sign in** dialog.

"Username"

Copied!

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

1. Click **Next**.
2. Copy the **Password** below and paste it into the **Welcome** dialog.

"Password"

Copied!

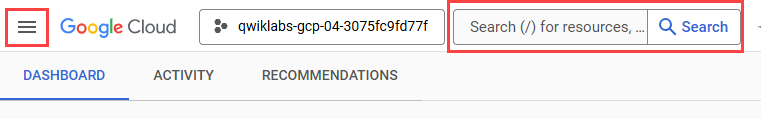
You can also find the Password in the Lab Details pane.

1. Click **Next**.

**Important:**You must use the credentials the lab provides you. Do not use your Google Cloud account credentials.**Note:**Using your own Google Cloud account for this lab may incur extra charges.

1. 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** (Navigation menu icon), click **Vertex AI > Workbench**.
2. Find the Workbench instance name instance and click on the **Open JupyterLab** button.

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

**Note:** If you do not see notebooks in JupyterLab, please follow these additional steps to reset the instance:

1. Close the browser tab for JupyterLab, and return to the Workbench home page.

2. Select the checkbox next to the instance name, and click **Reset**.

3. After the **Open JupyterLab** button is enabled again, wait one minute, and then click **Open JupyterLab**.

**Task 2. Set up the notebook**

Open your notebook file, import your libraries, and choose your model.

1. Open the notebook name file.
2. In the **Select Kernel** dialog, choose **Python 3** from the list of available kernels.
3. Run through these sections of the notebook:
   * **Get started**
   * **Use Google Gen AI SDK**
   * **Connect to a generative AI API service**

For **Project ID**, use Project ID, and for **Location**, use Region.

**Note:** 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.

Import libraries and set up the notebook

**Task 3. Interact with the model**

For more information about all AI models and APIs on Vertex AI, refer to [Google Models](https://cloud.google.com/vertex-ai/generative-ai/docs/learn/models#gemini-models) and [Model Garden](https://cloud.google.com/vertex-ai/generative-ai/docs/model-garden/explore-models).

Choose a model

* Run the **Choose a model** section of the notebook.

Send text prompts

Use the generate\_content method to generate responses to your prompts. You can pass text to generate\_content, and use the .text property to get the text content of the response.

* Run the **Send text prompts** section of the notebook.

Send multimodal prompts

You can include text, PDF documents, images, audio and video in your prompt requests and get text or code responses.

You can also pass the file URL in Part.from\_uri in the request to the model directly.

* Run the **Send multimodal prompts** section of the notebook.

Set the system instructions

The [system instructions](https://cloud.google.com/vertex-ai/generative-ai/docs/learn/prompts/system-instruction-introduction) allow you to control model behavior. Set the system instruction to give the model additional context to understand the task, provide more customized responses, and adhere to guidelines over the user interaction.

* Run the **Set system instruction** section of the notebook.

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

Interact with the model

**Task 4. Configure and control the model**

Configure model parameters

You can include parameter values in each call that you send to a model to control how the model generates a response. To learn more, refer to [experimenting with parameter values](https://cloud.google.com/vertex-ai/generative-ai/docs/learn/prompts/adjust-parameter-values).

* Run the **Configure model parameters** section of the notebook.

Configure safety filters

The Gemini API provides safety filters that you can adjust across multiple filter categories to restrict or allow certain types of content. You can use these filters to adjust what's appropriate for your use case. Refer to the [Configure safety filters](https://cloud.google.com/vertex-ai/generative-ai/docs/multimodal/configure-safety-filters) page for details.

When you make a request to the model, the content is analyzed and assigned a safety rating. You can inspect the safety ratings of the generated content by printing out the model responses, as in this example:

* Run the **Configure safety filters** section of the notebook.

Start a multi-turn chat

The Gemini API enables you to have freeform conversations across multiple turns.

* Run through the **Start a multi-turn chat** section of the notebook.

Control generated output

The [controlled generation](https://cloud.google.com/vertex-ai/generative-ai/docs/multimodal/control-generated-output) capability in Gemini API allows you to constrain the model output to a structured format. You can provide the schemas as Pydantic Models or a JSON string.

You also can define a response schema in a Python dictionary. You can use only the fields below. All other fields are ignored.

* enum
* items
* maxItems
* nullable
* properties
* required

In this example, you instruct the model to analyze product review data, extract key entities, perform sentiment classification (multiple choices), provide additional explanation, and output the results in JSON format.

* Run the **Control generated output** section of the notebook.

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

Configure and control the model

**Task 5. Manage the model interaction**

Generate content stream

By default, the model returns a response after completing the entire generation process. You can also use the generate\_content\_stream method to stream the response as it is being generated. The model returns chunks of the response as they are generated.

* Run the **Generate content stream** section of the notebook.

Send asynchronous requests

You can send asynchronous requests using the client.aio module. This module exposes all the analogous async methods that are available on client.

For example, client.aio.models.generate\_content is the async version of client.models.generate\_content.

* Run the **Send asynchronous requests** section of the notebook.

Count tokens and compute tokens

You can use the count\_tokens method to calculate the number of input tokens before sending a request to the Gemini API. Refer to the [List and count tokens](https://cloud.google.com/vertex-ai/generative-ai/docs/multimodal/list-token) page for details.

**Count tokens**

* Run the **Count tokens** section of the notebook.

**Compute tokens**

* Run the **Compute tokens** section of the notebook.

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

Manage the model interaction

**Task 6. Advanced features**

Function calling

[Function calling](https://cloud.google.com/vertex-ai/generative-ai/docs/multimodal/function-calling) lets you provide a set of tools that it can use to respond to the user's prompt. You create a description of a function in your code, then pass that description to a language model in a request. The response from the model includes the name of a function that matches the description and the arguments to call it with.

For more examples of Function Calling, refer to [this notebook](https://github.com/GoogleCloudPlatform/generative-ai/blob/main/gemini/function-calling/intro_function_calling.ipynb)..

* Run the **Function calling** section of the notebook.

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

Function calling

Use context caching

[Context caching](https://cloud.google.com/vertex-ai/generative-ai/docs/context-cache/context-cache-overview) lets you store frequently used input tokens in a dedicated cache and reference them for subsequent requests. This eliminates the need to repeatedly pass the same set of tokens to a model.

**Note**: Context caching is only available for stable models with fixed versions (for example, gemini-2.0-flash-001). You must include the version postfix (for example, the -001).

**Create a cache**

* Run the **Create a cache** section of the notebook.

**Use a cache**

* Run the **Use a cache** section of the notebook.

**Delete a cache**

* Run the **Delete a cache** section of the notebook.

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

Use context caching

Batch prediction

Different from getting online (synchronous) responses, where you are limited to one input request at a time, [batch predictions for the Gemini API in Vertex AI](https://cloud.google.com/vertex-ai/generative-ai/docs/multimodal/batch-prediction-gemini) allow you to send a large number of requests to Gemini in a single batch request. Then, the model responses asynchronously populate to your storage output location in [Cloud Storage](https://cloud.google.com/storage/docs/introduction) or [BigQuery](https://cloud.google.com/bigquery/docs/storage_overview" \t "_blank).

Batch predictions are generally more efficient and cost-effective than online predictions when processing a large number of inputs that are not latency sensitive.

**Prepare batch inputs**

The input for batch requests specifies the items to send to your model for prediction.

Batch requests for Gemini accept BigQuery storage sources and Cloud Storage sources. You can learn more about the batch input formats in the [Batch text generation](https://cloud.google.com/vertex-ai/generative-ai/docs/multimodal/batch-prediction-gemini#prepare_your_inputs) page.

This lab uses Cloud Storage as an example. The requirements for Cloud Storage input are:

* File format: [JSON Lines (JSONL)](https://jsonlines.org/)
* Located in us-central1
* Appropriate read permissions for the service account

Each request that you send to a model can include parameters that control how the model generates a response. Learn more about Gemini parameters in the [Experiment with parameter values](https://cloud.google.com/vertex-ai/generative-ai/docs/learn/prompts/adjust-parameter-values) page.

This is one of the example requests in the input JSONL file batch\_requests\_for\_multimodal\_input\_2.jsonl:

{"request":{"contents": [{"role": "user", "parts": [{"text": "List objects in this image."}, {"file\_data": {"file\_uri": "gs://cloud-samples-data/generative-ai/image/office-desk.jpeg", "mime\_type": "image/jpeg"}}]}],"generationConfig":{"temperature": 0.4}}}

* Run the **Prepare batch inputs** section of the notebook.

**Prepare batch output location**

When a batch prediction task completes, the output is stored in the location specified in your request.

* The location is in the form of a Cloud Storage or BigQuery URI prefix, for example: gs://path/to/output/data or bq://projectId.bqDatasetId.
* If not specified, gs://STAGING\_BUCKET/gen-ai-batch-prediction is used for Cloud Storage source and bq://PROJECT\_ID.gen\_ai\_batch\_prediction.predictions\_TIMESTAMP is used for BigQuery source.

This lab uses a Cloud Storage bucket as an example for the output location.

You can specify the URI of your Cloud Storage bucket in BUCKET\_URI, or, if it is not specified, a new Cloud Storage bucket in the form of gs://PROJECT\_ID-TIMESTAMP is be created for you.

* Run the **Prepare batch output location** section of the notebook.

**Send a batch prediction request**

To make a batch prediction request, you specify a source model ID, an input source and an output location where Vertex AI stores the batch prediction results.

For more, see the [Batch prediction API](https://cloud.google.com/vertex-ai/generative-ai/docs/model-reference/batch-prediction-api) page. You can also check the status in the console at https://console.cloud.google.com/vertex-ai/batch-predictions

* Run the **Send a batch prediction request** section of the notebook.

**Note:** it may take a few minutes for your batch prediction to complete.

**Retrieve batch prediction results**

When a batch prediction task is complete, the output of the prediction is stored in the location specified in your request. It is also available in batch\_job.dest.bigquery\_uri or batch\_job.dest.gcs\_uri.

Example output:

{"status": "", "processed\_time": "2024-11-13T14:04:28.376+00:00", "request": {"contents": [{"parts": [{"file\_data": null, "text": "List objects in this image."}, {"file\_data": {"file\_uri": "gs://cloud-samples-data/generative-ai/image/gardening-tools.jpeg", "mime\_type": "image/jpeg"}, "text": null}], "role": "user"}], "generationConfig": {"temperature": 0.4}}, "response": {"candidates": [{"avgLogprobs": -0.10394711927934126, "content": {"parts": [{"text": "Here's a list of the objects in the image:\n\n\* \*\*Watering can:\*\* A green plastic watering can with a white rose head.\n\* \*\*Plant:\*\* A small plant (possibly oregano) in a terracotta pot.\n\* \*\*Terracotta pots:\*\* Two terracotta pots, one containing the plant and another empty, stacked on top of each other.\n\* \*\*Gardening gloves:\*\* A pair of striped gardening gloves.\n\* \*\*Gardening tools:\*\* A small trowel and a hand cultivator (hoe). Both are green with black handles."}], "role": "model"}, "finishReason": "STOP"}], "modelVersion": "gemini-2.0-flash-001@default", "usageMetadata": {"candidatesTokenCount": 110, "promptTokenCount": 264, "totalTokenCount": 374}}}

* Run the **Retrieve batch prediction results** section of the notebook.

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

Retrieve batch prediction results

Get text embeddings

You can get text embeddings for a snippet of text by using embed\_content method. All models produce an output with 768 dimensions by default. However, some models give users the option to choose an output dimensionality between 1 and 768. See [Vertex AI text embeddings API](https://cloud.google.com/vertex-ai/generative-ai/docs/embeddings/get-text-embeddings) for details.

* Run the **Get text embeddings** section of the notebook.

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

Get text embeddings

**Congratulations!**

Congratulations! You've successfully explored the Google Gen AI SDK, learning to connect to AI services, send diverse prompts, and fine-tune responses from the Gemini model. You've also got hands-on experience with more advanced techniques like managing interactions, using context caching, and even working with embeddings! Now you're well-equipped to leverage the power of generative AI for your own projects.

Next steps / learn more

Check out the following resources to learn more about Gemini:

* [Gemini Overview](https://deepmind.google/technologies/gemini/)
* [Generative AI on Vertex AI Documentation](https://cloud.google.com/vertex-ai/docs/generative-ai/learn/overview)
* [Generative AI on YouTube](https://www.youtube.com/@googlecloudtech/)
* Explore the Vertex AI [Cookbook](https://cloud.google.com/vertex-ai/generative-ai/docs/cookbook) for a curated, searchable gallery of notebooks for Generative AI.
* Explore other notebooks and samples in the [Google Cloud Generative AI repository](https://github.com/GoogleCloudPlatform/generative-ai).

Google Cloud training and certification

...helps you make the most of Google Cloud technologies. [Our classes](https://cloud.google.com/training) 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](https://cloud.google.com/certification/) help you validate and prove your skill and expertise in Google Cloud technologies.

**Manual Last Updated July 11, 2025**

**Lab Last Tested July 11, 2025**

Copyright 2025 Google LLC. All rights reserved. Google and the Google logo are trademarks of Google LLC. All other company and product names may be trademarks of the respective companies with which they are associated.

**+++++**

**LAB 4: PROMPT DESIGN IN VERTEX AI: CHALLENGE LAB**

**Overview**

**In a challenge lab you’re given a scenario and a set of tasks. Instead of following step-by-step instructions, you will use the skills learned from the labs in the course to figure out how to complete the tasks on your own! An automated scoring system (shown on this page) will provide feedback on whether you have completed your tasks correctly.**

**When you take a challenge lab, you will not be taught new Google Cloud concepts. You are expected to extend your learned skills, like changing default values and reading and researching error messages to fix your own mistakes.**

**To score 100% you must successfully complete all tasks within the time period!**

**This lab is recommended for students who have enrolled in the Prompt Design in Vertex AI course. Are you ready for the challenge?**

**Topics tested**

**Craft effective prompts and use parameters to guide generative AI output in Vertex AI Studio.**

**Apply Gemini models to create product descriptions and taglines in a real-world marketing scenario.**

**Examine and run Python code exported from Vertex AI Studio to gain a basic understanding of generative AI implementation.**

**Use Jupyter Notebooks to test and modify generative AI code.**

**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 simulation or demo environment. It does so by giving you new, temporary credentials you use to sign in and access Google Cloud for the duration of the lab.**

**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.**

**Challenge Scenario**

**You're a member of an educational content startup specializing in engaging learners with the natural world. You've formed a partnership with Cymbal Direct, an online retailer launching a new line of outdoor gear and apparel designed to encourage young people to explore and connect with nature.**

**Cymbal Direct wants to create a marketing campaign for its new product line that leverages the power of generative AI. Your task is to help them develop a set of tools within Google Cloud's Vertex AI platform that will streamline the generation of the following:**

**Evocative Product Descriptions: using image analysis to inspire short, descriptive text that captures the essence of their products and the feeling of being in nature.**

**Catchy Taglines: focused on highlighting product features, the target audience, and the desired emotional response.**

**Task 1. Build a Gemini image analysis tool**

**In this section, you will create a template for analyzing images of Cymbal Direct products using the model name model in Vertex AI Studio. The goal is to generate descriptive text options inspired by the image, from simple details to more evocative, mood-setting phrases.**

**Tasks:**

**Create a prompt in Vertex AI Studio with the model name model to analyze Cymbal Direct's product image (provided in Google Cloud Storage) and generate multiple descriptive text options inspired by the image.**

**The image for this task is located at: image file path.**

**Experiment with different prompts to generate the following:**

**Short, descriptive text inspired by the image.**

**Catchy phrases suitable for advertisements.**

**A poetic description for a nature-focused campaign.**

**Evaluate and Iterate: adjust your prompt and parameters as needed to refine the results.**

**Name your prompt Cymbal Product Analysis.**

**Save Prompt: Once you're happy with the results, click Save, and select the Region region.**

**Note: Ensure you are using the model name model for this task!**

**Build a Gemini image analysis tool.**

**Task 2. Build a Gemini tagline generator**

**In this task, you will create a prompt for generating diverse tagline possibilities using the model name model in Vertex AI Studio. The goal is to develop a prompt that allows for customization of the tagline style, based on product attributes, target audience, and emotional resonance.**

**Tasks:**

**Create a new prompt with the model name model to create a customizable tagline generator for Cymbal Direct's new product line.**

**In the System instructions box, enter the following:**

**Cymbal Direct is partnering with an outdoor gear retailer. They're launching a new line of products designed to encourage young people to explore the outdoors. Help them create catchy taglines for this product line.**

**Copied!**

**Include 2 Examples in your prompt to guide the output style. Add this example input and output for your first example, then use this template to add another example.**

**Input Output**

**Write a tagline for a durable backpack designed for hikers that makes them feel prepared. Consider styles like minimalist. Built for the Journey: Your Adventure Essentials.**

**Design a prompt with parameters to customize taglines based on:**

**Product attributes (e.g., durable, lightweight)**

**Target audience (e.g., young adventurers, families)**

**Emotional resonance (e.g., empowered, connected)**

**Add one input and your prompt, then click the submit button to have Gemini generate a tagline option.**

**Evaluate and Iterate:**

**Experiment with different parameter combinations to see the variety of taglines produced.**

**Based on the results, fine-tune the wording of your prompt, add more parameter options, or adjust the style choices to achieve your desired outcome.**

**Name your prompt Cymbal Tagline Generator Template.**

**Save Prompt: Once you're happy with the results, click Save, and select the Region region. Note: if you prompt is already saved with the Autosave functionality, ensure the name of the prompt is correct and you can see it in the Prompt Management page.**

**Note: Ensure you are using the model name model for this task!**

**Build a Gemini tagline generator.**

**Task 3. Experiment with image analysis code**

**In this task, you will explore the Python code for the image analysis prompt you created. You will then modify the prompt to be more specific and test the new prompt in a notebook.**

**In the Google Cloud console, on the Navigation menu (Navigation menu icon), click Vertex AI > Workbench.**

**Find the Workbench instance name instance and click on the Open JupyterLab button.**

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

**Note: If you do not see notebooks in JupyterLab, please follow these additional steps to reset the instance:**

**1. Close the browser tab for JupyterLab, and return to the Workbench home page.**

**2. Select the checkbox next to the instance name, and click Reset.**

**3. After the Open JupyterLab button is enabled again, wait one minute, and then click Open JupyterLab.**

**Create a new notebook file named image-analysis.ipynb. Set the kernel to Python 3.**

**Explore the image analysis code**

**From the Vertex AI Studio page, navigate to the Prompt Management page. Select the Cymbal Product Analysis prompt you created.**

**On the right side of the toolbar, click Build with Code. Use Python as the language.**

**Note: The first code block uses terminal commands to install the necessary Python packages. You can run these commands in a terminal inside of your Workbench instance by navigating to File > New > Terminal.**

**Run the second code cells in the notebook. Verify that the code executes successfully and produces the expected output.**

**Modify the image analysis prompt**

**Within the code, there will be a line of text between triple quotes ("""). This is your current image analysis prompt. For example, it might be:**

**"""Describe this image with a focus on colors, textures, and the feeling it evokes."""**

**Now you will modify the prompt to be more specific. Change the wording of the prompt in the code cell to make the output less than 10 words.**

**Next, modify the prompt code to encourage the model to produce the most creative, unusual, and unexpected descriptions of the image it can think of.**

**Hint: You will need to adjust one of the parameters in the code to achieve this!**

**Save the changes to your code. Then, rerun the code cell in your notebook to test Gemini with the new prompt.**

**Verify that the new descriptions are shorter and more creative than the previous ones.**

**Note: Ensure you are saving your notebook file to pass the progress check!**

**Experiment with image analysis code.**

**Task 4. Experiment with tagline generation code**

**In this task, you will explore the Python code for the tagline prompt you created. You will then modify the prompt to include a specific keyword and test the new prompt in a notebook.**

**Explore the tagline generator code**

**Inside of your Workbench instance, create a new notebook file named tagline-generator.ipynb. Set the kernel to Python 3.**

**From the Vertex AI Studio page, navigate to the Prompt Management page. Select the Cymbal Tagline Generator Template prompt you created.**

**On the right side of the toolbar, click Build with Code. Use Python as the language.**

**Note: The first code block uses terminal commands to install the necessary Python packages. If you ran the it in the previous section already, you won't need to run it again.**

**Run the second code cells in the notebook. Verify that the code executes successfully and produces the expected output.**

**Modify the tagline generation prompt**

**Within the code, there will be multiple lines of text between triple quotes ("""). This is your current tagline generation prompt.**

**Now you will modify the prompt to include a specific keyword. Modify the last input to specifically request that the tagline includes the keyword nature.**

**Save the changes to your code. Then, rerun the code cell in your notebook to test the language model with the new prompt.**

**Verify that the new tagline includes the keyword nature.**

**Experiment with tagline generation code.**

**Note: Wait for couple of minutes and click the check my progress button if you are not getting the score.**

**Congratulations!**

**Congratulations, you've successfully completed the lab! You've helped Cymbal Direct create a set of tools within Google Cloud's Vertex AI platform that will streamline the generation of evocative product descriptions and catchy taglines for their new product line. You've also explored and modified the image analysis and tagline generation code in a notebook. Great job!**

**Prompt Design in Vertex AI skill badge**

**Next steps / learn more**

**Check out the following resources to learn more about Gemini:**

**Gemini Overview**

**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 July 11, 2025**

**Lab Last Tested July 11, 2025**

**Copyright 2025 Google LLC. All rights reserved. Google and the Google logo are trademarks of Google LLC. All other company and product names may be trademarks of the respective companies with which they are associated.**