Machine Learning in the Enterprise

Upgrade available
Machine Learning in the Enterprise navigate_next Vertex AI Pipelines

Running Pipelines on Vertex AI 2.5

1 hour Free

Overview

In this lab, you learn how to utilize Vertex AI Pipelines to execute a simple Kubeflow Pipeline SDK derived ML Pipeline.

Objectives

In this lab, you perform the following tasks:

- Set up the Project Environment
- Inspect and Configure Pipeline Code
- Execute the AI Pipeline

Setup and requirements

Before you click the Start Lab button

Note: 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 will be made available to you.

This Qwiklabs hands-on lab lets you do the lab activities yourself in a real cloud environment, not in a simulation or demo environment. It does so by giving you new, temporary credentials that you use to sign in and access Google Cloud for the duration of the lab.

What you need

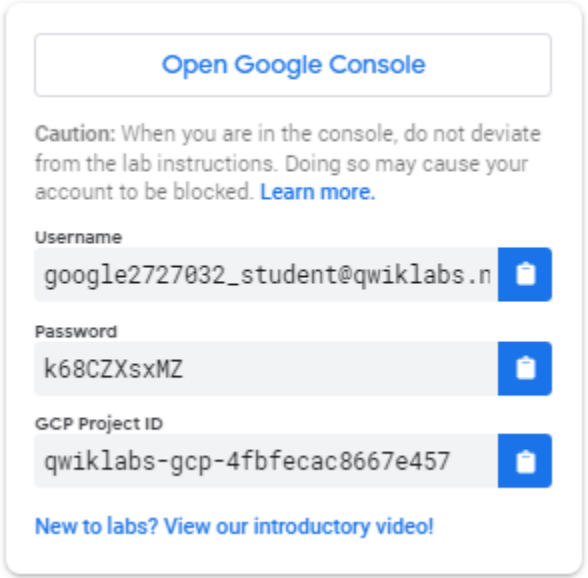
To complete this lab, you need:

- Access to a standard internet browser (Chrome browser recommended).
- Time to complete the lab.

Note: If you already have your own personal Google Cloud account or project, do not use it for this lab. **Note:** If you are using a Pixelbook, open an Incognito window to run this lab.

How to start your lab and sign in to the Console

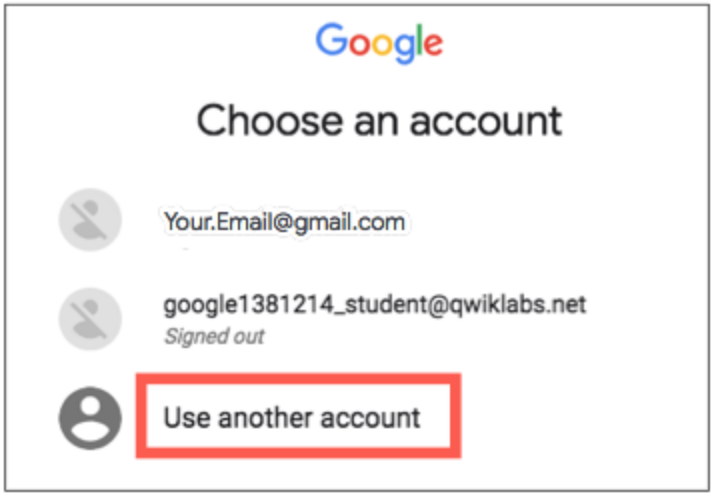
1. Click the **Start Lab** button. If you need to pay for the lab, a pop-up opens for you to select your payment method. On the left is a panel populated with the temporary credentials that you must use for this lab.



2. Copy the username, and then click **Open Google Console**. The lab spins up resources, and then opens another tab that shows the **Choose an account** page.

   Note: Open the tabs in separate windows, side-by-side.

3. On the Choose an account page, click **Use Another Account**. The Sign in page opens.



4. Paste the username that you copied from the Connection Details panel. Then copy and paste the password.

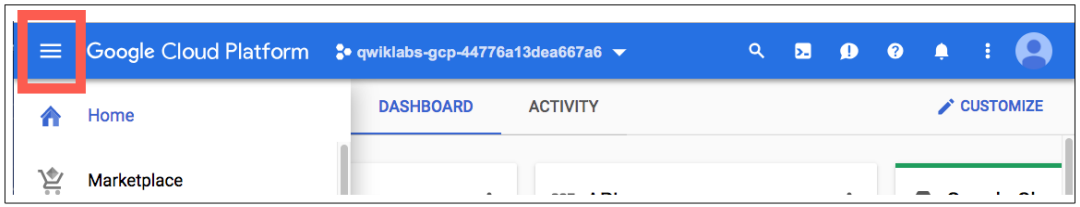
Note: You must use the credentials from the Connection Details panel. Do not use your Google Cloud Skills Boost credentials. If you have your own Google Cloud account, do not use it for this lab (avoids incurring charges).

5. 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 Cloud console opens in this tab.

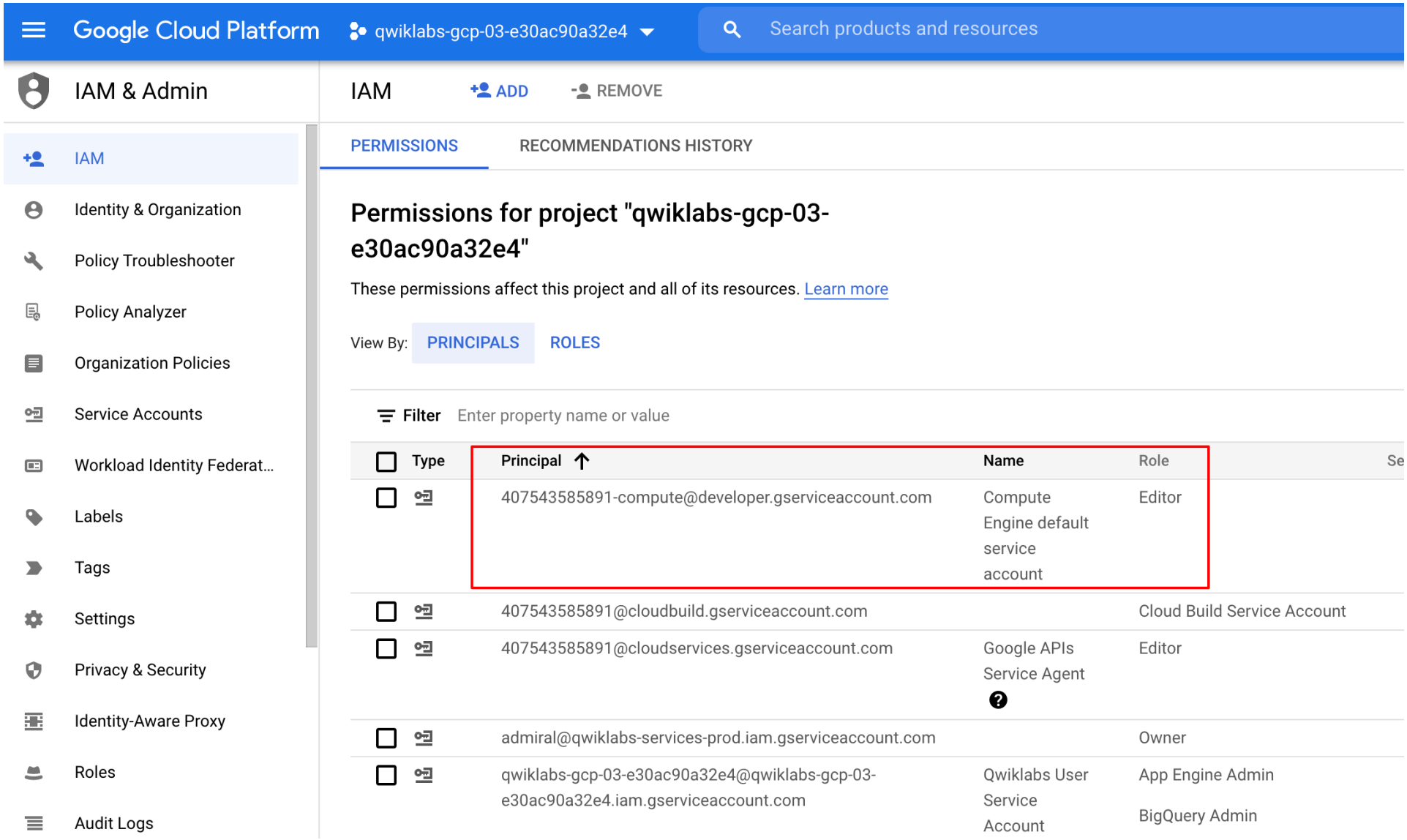
Note: You can view the menu with a list of Google Cloud Products and Services by clicking the **Navigation menu** at the top-left.



Check project permissions

Before you begin your work on Google Cloud, you need to ensure that your project has the correct permissions within Identity and Access Management (IAM).

1. In the Google Cloud console, on the **Navigation menu** (☰), select **IAM & Admin** > **IAM**.
2. Confirm that the default compute Service Account `{project-number}-compute@developer.gserviceaccount.com` is present and has the `editor` role assigned. The account prefix is the project number, which you can find on **Navigation menu** > **Home**.



Note: If the account is not present in IAM or does not have the `editor` role, follow the steps below to assign the required role.

1. In the Google Cloud console, on the **Navigation menu**, click **Home**.
2. Copy the project number (e.g. `729328892908`).
3. On the **Navigation menu**, select **IAM & Admin** > **IAM**.
4. At the top of the **IAM** page, click **Add**.
5. For **New principals**, type:

{project-number}-compute@developer.gserviceaccount.com

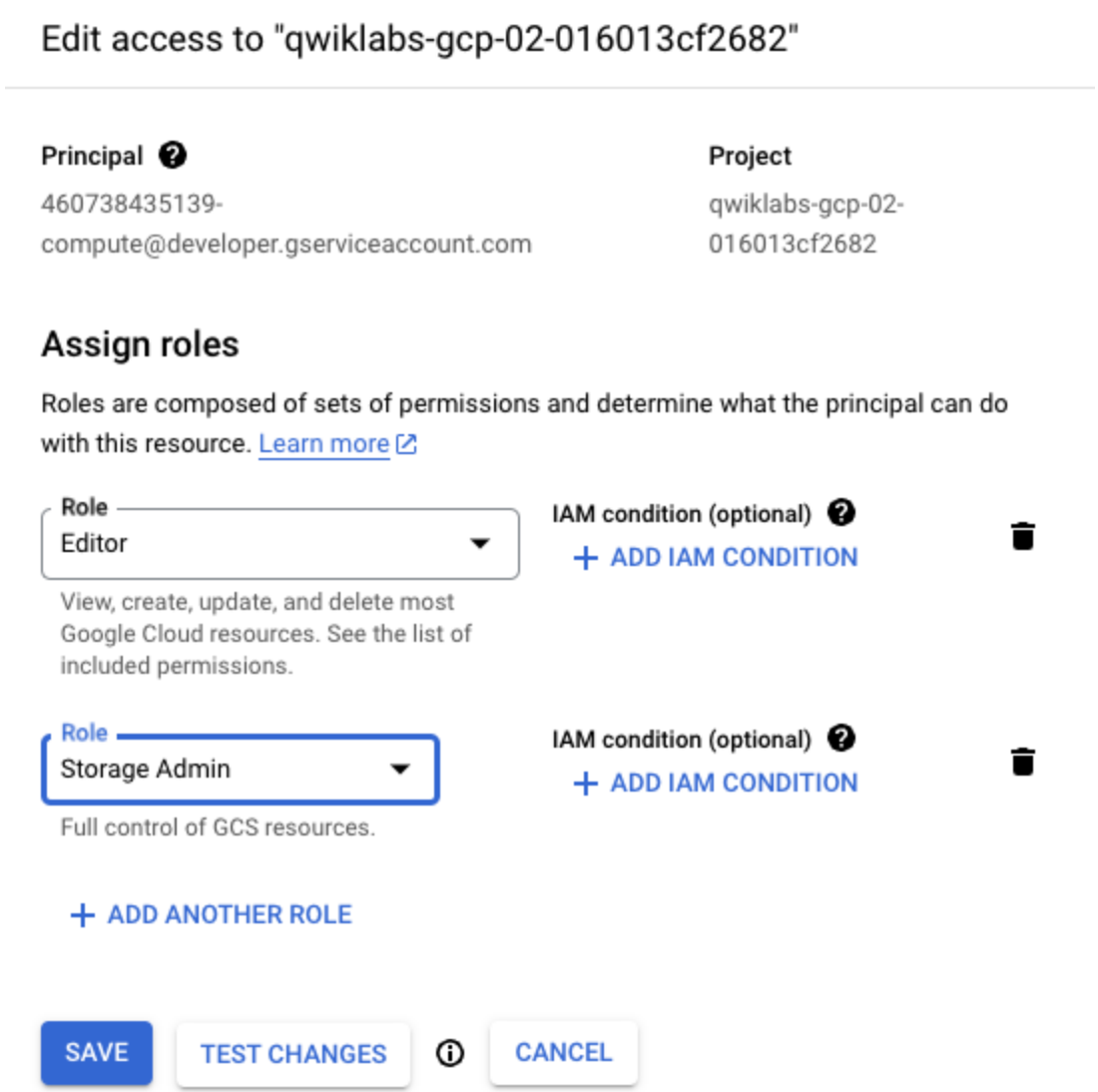
6. Replace `{project-number}` with your project number.
7. For **Role**, select **Project** (or Basic) > **Editor**.
8. Click **Save**.

Task 1. Set up the project environment

Vertex AI Pipelines run in a serverless framework whereby pre-compiled pipelines are deployed on-demand or on a schedule. In order to facilitate smooth execution some environment configuration is required.

For the seamless execution of Pipeline code in a Qwiklabs environment the Compute Service Account needs elevated privileges on Cloud Storage.

1. In the Google Cloud console, on the **Navigation menu** (☰), click **IAM & Admin** > **IAM**.
2. Click the pencil icon for default compute Service Account `{project-number}-compute@developer.gserviceaccount.com` to assign the **Storage Admin** role.
3. On the slide-out window, click **Add Another Role**. Type **Storage Admin** in the search box. Select **Storage Admin** with **Full control of GCS resources** from the results list.
4. Click **Save** to assign the role to the Compute Service Account.



Artifacts will be accessed on ingest and export as the Pipeline executes.

5. Run this code block in the Cloud Shell to create a bucket in your project and two folders each with an empty file:

gcloud storage buckets create gs://{{{primary_project.project_id|Project ID}}} touch emptyfile1 touch emptyfile2 gcloud storage cp emptyfile1 gs://{{{primary_project.project_id|Project ID}}}/pipeline-output/emptyfile1 gcloud storage cp emptyfile2 gs://{{{primary_project.project_id|Project ID}}}/pipeline-input/emptyfile2

The Pipeline has already been created for you and simply requires a few minor adjustments to allow it to run in your Qwiklabs project.

6. Download the AI Pipeline from the lab assets folder:

wget https://storage.googleapis.com/cloud-training/dataengineering/lab_assets/ai_pipelines/basic_pipeline.json

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

Task 2. Configure and inspect the Pipeline code

The Pipeline code is a compilation of two AI operations written in Python. The example is very simple but demonstrates how easy it is orchestrate ML procedures written in a variety of languages (TensorFlow, Python, Java, etc.) into an easy to deploy AI Pipeline. The lab example performs two operations, concatenation and reverse, on two string values.

1. First you must make an adjustment to update the output folder for the AI Pipeline execution. In the Cloud Shell use the Linux **Stream EDitor (sed)** command to adjust this setting:

sed -i 's/PROJECT_ID/{{{primary_project.project_id|Project ID}}}/g' basic_pipeline.json

sed -i 's/PROJECT_ID/qwiklabs-gcp-00-81e02d074f39/g' basic_pipeline.json

sed -i basic_pipeline.json

2. Inspect **basic_pipeline.json** to confirm that the output folder is set to your project:

tail -20 basic_pipeline.json

The key sections of code in **basic_pipeline.json** are the **deploymentSpec** and **command** blocks. Below is the first command block, the job that concatenates the input strings. This is **Kubeflow Pipeline SDK (kfp)** code that is designated to be executed by the Python 3.7 engine. You will not change any code, the section is shown here for your reference:

"program_path=$(mktemp -d)\nprintf \"%s\" \"$0\" > \"$program_path/ephemeral_component.py\"\npython3 -m kfp.v2.components.executor_main --component_module_path \"$program_path/ephemeral_component.py\" \"$@\"\n", "\nimport kfp\nfrom kfp.v2 import dsl\nfrom kfp.v2.dsl import *\nfrom typing import *\n\ndef concat(a: str, b: str) -> str:\n return a + b\n\n"], "image": "python:3.7"

3. You can explore the entire file by issuing the command below:

more basic_pipeline.json **Note:** Press the spacebar to advance through the file until its end. If you wish to close the file early, type **q** to close the **more** command.

4. Next, move the updated **basic_pipeline.json** file to the Cloud Storage bucket created earlier so that it can be accessed to run an AI Pipeline job:

gcloud storage cp basic_pipeline.json gs://{{{primary_project.project_id|Project ID}}}/pipeline-input/basic_pipeline.json

gcloud storage cp basic_pipeline.json gs://qwiklabs-gcp-00-81e02d074f39/pipeline-input/basic_pipeline.json

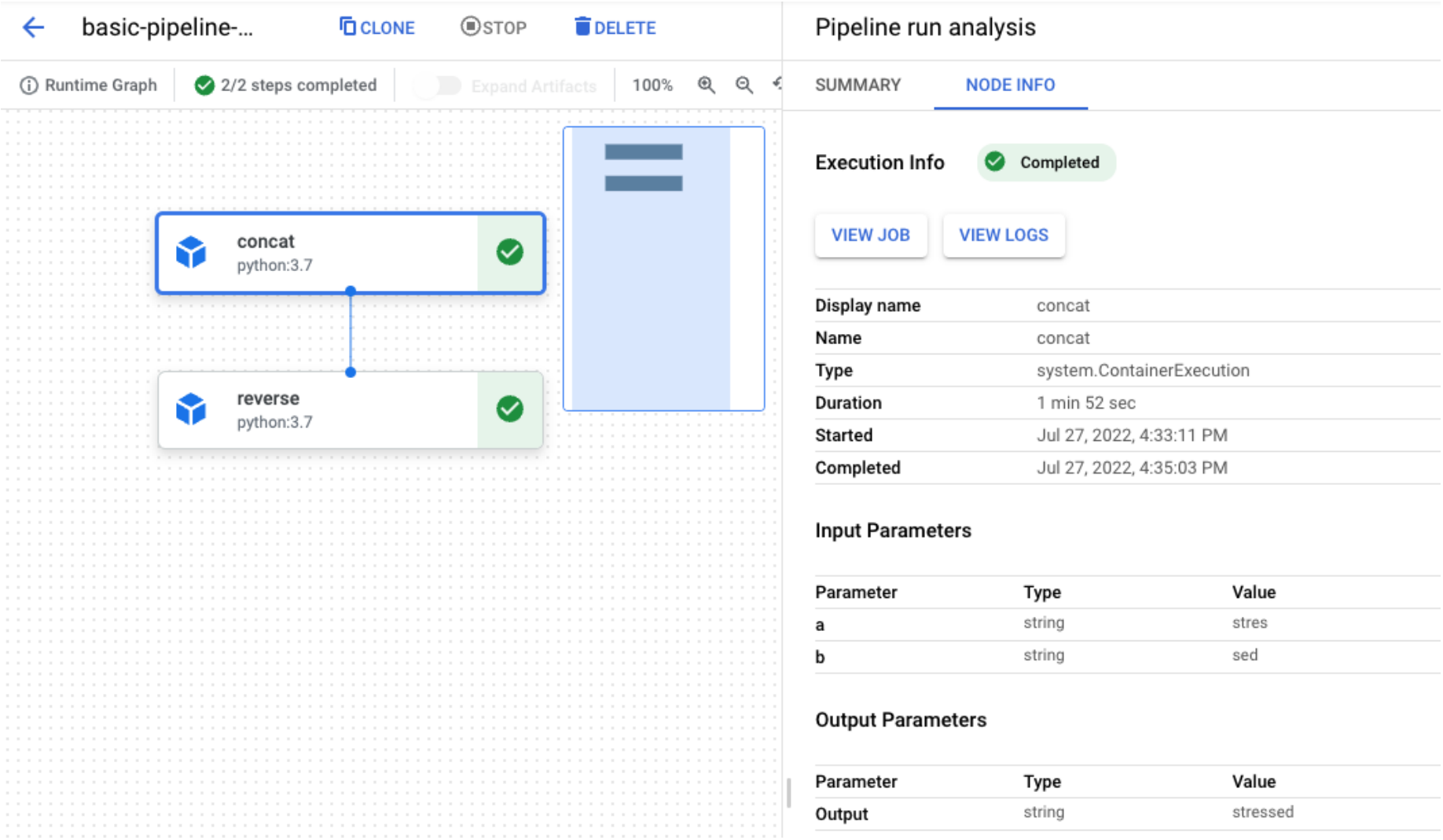
Click **Check my progress** to verify the objective. Deploy Pipeline

Task 3. Execute the AI Pipeline

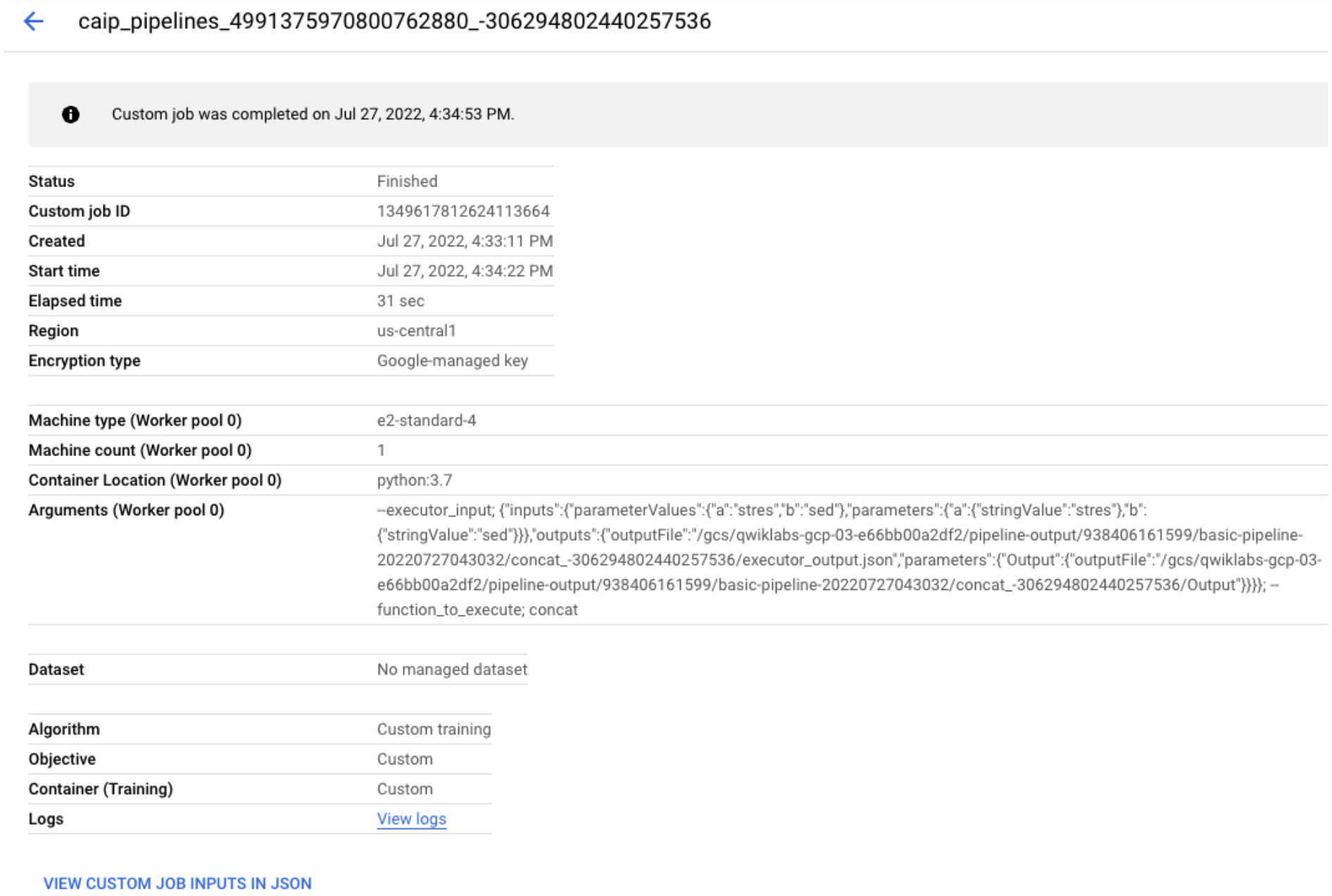
1. From the Console, open the **Navigation menu (☰)**, under **Artificial Intelligence** click **Vertex AI**.
2. Click the blue **Enable all recommended API**.
3. Once the API is enabled, click **Pipelines** in the left menu.
4. Click **Create Run** on the top menu.
5. From **Run detail**, select **Import from Cloud Storage** and for **Cloud Storage URL** browse to the **pipeline-input** folder you created inside your project's cloud storage bucket. Select the **basic_pipeline.json** file.
6. Click **Select**.
7. Leave the remaining default values, click **Continue**.

You may leave the default values for **Runtime configuration**. Notice that the Cloud Storage Output Directory is set to the bucket folder created in an earlier step. The Pipeline Parameters are pre-filled from the values in the **basic_pipeline.json** file but you have the option of changing those at runtime via this wizard.

8. Click **Submit** to start the Pipeline execution.
9. You will be returned to the Pipeline dashboard and your run will progress from **Pending** to **Running** to **Succeeded**. The entire run will take between 3 and 6 minutes.
10. Once the status reaches Succeeded, click on the run name to see the execution graph and details.



11. A graph element exists for each step. Click on the **concat** object to see the details for the job.
12. Click on the **View Job** button. A new tab will open with the Vertex AI Custom Job that was submitted to the backend to satisfy the pipeline request.



Feel free to explore more details on the Pipeline execution.

Congratulations!

You have successfully used Vertex AI Pipelines to execute a simple Kubeflow Pipeline SDK derived ML Pipeline.

Manual Last Updated May 8, 2023

Lab Last Tested May 8, 2023

- Overview
- Setup and requirements
- Task 1. Set up the project environment
- Task 2. Configure and inspect the Pipeline code
- Task 3. Execute the AI Pipeline
- Congratulations!