**Streaming Data Processing: Publish Streaming Data into PubSub**

**Task 1: Preparation**

You will be running a sensor simulator from the training VM. There are several files and some setup of the environment required.

Open the SSH terminal and connect to the training VM

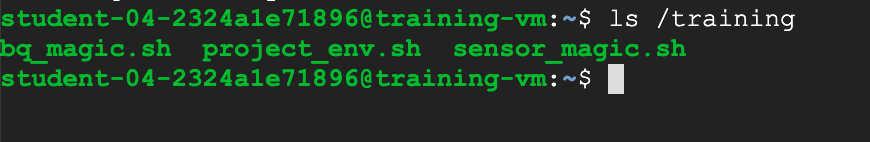
1. In the Console, on the **Navigation menu** ( ), click **Compute Engine** > **VM instances**.
2. Locate the line with the instance called **training-vm**.
3. On the far right, under **Connect**, click on **SSH** to open a terminal window.
4. In this lab, you will enter CLI commands on the **training-vm**.

**Verify initialization is complete**

1. The **training-vm** is installing some software in the background. Verify that setup is complete by checking the contents of the new directory.

ls /training

The setup is complete when the result of your list (ls) command output appears as in the image below. If the full listing does not appear, wait a few minutes and try again. **Note**: It may take 2 to 3 minutes for all background actions to complete.



**Download Code Repository**

1. Next you will download a code repository for use in this lab.

git clone https://github.com/GoogleCloudPlatform/training-data-analyst

Identify a project

One environment variable that you will set is **$DEVSHELL\_PROJECT\_ID** that contains the Google Cloud project ID required to access billable resources.

1. In the Console, on the **Navigation menu** ( ), click **Home**. In the panel with Project Info, the **Project ID** is listed. You can also find this information in the Qwiklabs tab under Connection Details, where it is labeled **GCP Project ID**.
2. On the **training-vm** SSH terminal, set the DEVSHELL\_PROJECT\_ID environment variable and export it so it will be available to other shells. The following command obtains the active Project ID from the Google Cloud environment.

export DEVSHELL\_PROJECT\_ID=$(gcloud config get-value project)

Task 2: Create Pub/Sub topic and subscription

1. On the **training-vm** SSH terminal, navigate to the directory for this lab.

cd ~/training-data-analyst/courses/streaming/publish

Verify that the Pub/Sub service is accessible and working using the gcloud command.

1. Create your topic and publish a simple message.

gcloud pubsub topics create sandiego

1. Publish a simple message.

gcloud pubsub topics publish sandiego --message "hello"

1. Create a subscription for the topic.

gcloud pubsub subscriptions create --topic sandiego mySub1

1. Pull the first message that was published to your topic.

gcloud pubsub subscriptions pull --auto-ack mySub1

Do you see any result? If not, why?

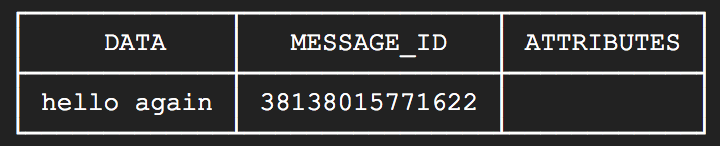
1. Try to publish another message and then pull it using the subscription.

gcloud pubsub topics publish sandiego --message "hello again"

gcloud pubsub subscriptions pull --auto-ack mySub1

Did you get any response this time?

Output:



1. In the **training-vm** SSH terminal, cancel your subscription.

gcloud pubsub subscriptions delete mySub1

Task 3: Simulate traffic sensor data into Pub/Sub

1. Explore the python script to simulate San Diego traffic sensor data. **Do not make any changes to the code.**

cd ~/training-data-analyst/courses/streaming/publish

nano send\_sensor\_data.py

Look at the simulate function. This one lets the script behave as if traffic sensors were sending in data in real time to Pub/Sub. The speedFactor parameter determines how fast the simulation will go. Exit the file by pressing **Ctrl+X**.

1. Download the traffic simulation dataset.

./download\_data.sh

Simulate streaming sensor data

1. Run the **send\_sensor\_data.py**.

./send\_sensor\_data.py --speedFactor=60 --project $DEVSHELL\_PROJECT\_ID

This command simulates sensor data by sending recorded sensor data via Pub/Sub messages. The script extracts the original time of the sensor data and pauses between sending each message to simulate realistic timing of the sensor data. The value **speedFactor** changes the time between messages proportionally. So a **speedFactor** of 60 means "60 times faster" than the recorded timing. It will send about an hour of data every 60 seconds.

Leave this terminal open and the simulator running.

Task 4: Verify that messages are received

Open a second SSH terminal and connect to the training VM

1. In the Console, on the **Navigation menu** ( ), click **Compute Engine** > **VM instances**.
2. Locate the line with the instance called **training-vm**.
3. On the far right, under **Connect**, click on **SSH** to open a second terminal window.
4. Change into the directory you were working in:

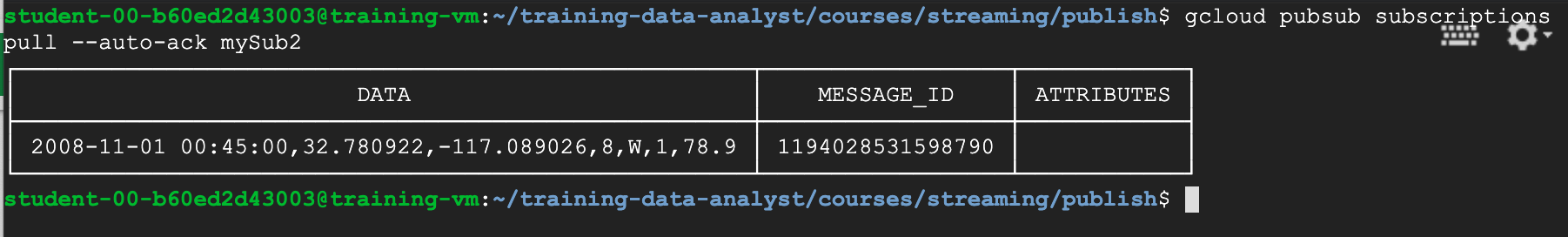
cd ~/training-data-analyst/courses/streaming/publish

1. Create a subscription for the topic and do a pull to confirm that messages are coming in (note: you may need to issue the 'pull' command more than once to start seeing messages):

gcloud pubsub subscriptions create --topic sandiego mySub2

gcloud pubsub subscriptions pull --auto-ack mySub2

Confirm that you see a message with traffic sensor information.



1. Cancel this subscription.

gcloud pubsub subscriptions delete mySub2

1. Close the second terminal.

exit

Stop the sensor simulator

1. Return to the first terminal.
2. Interrupt the publisher by typing **Ctrl+C** to stop it.
3. Close the first terminal.

exit