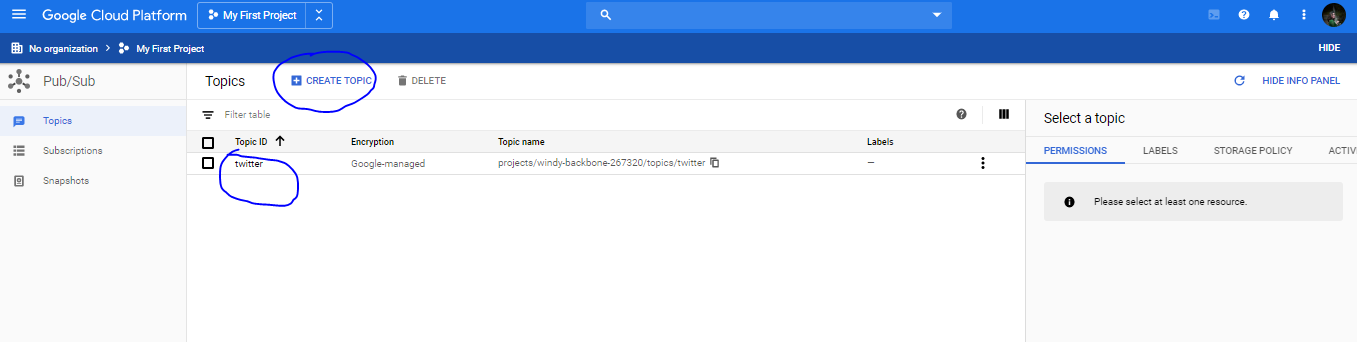
**Goal: Publish twitter messages to pub/sub using dockerized image on GCP. From pub/sub the tweets get stored in google cloud storage.**

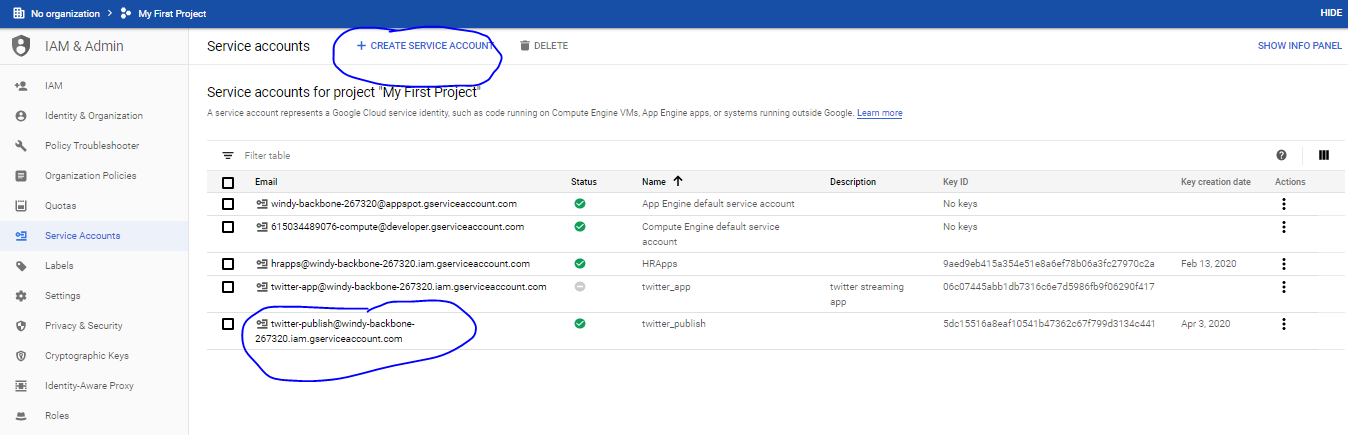
Step 1: Pub/Sub on GCP

Go to google cloud platform and create a new topic where the twitter messages are going to publish. Below example has ‘twitter’ as the topic name.



Step 2: IAM & Admin on GCP

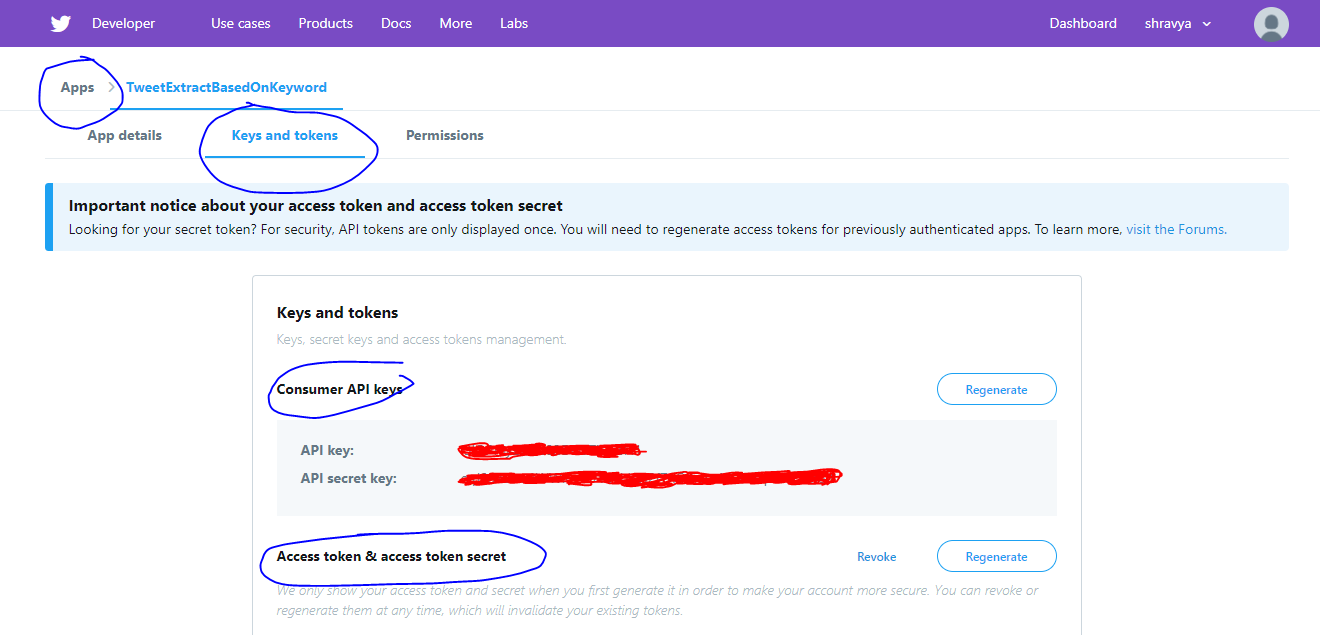
Create a new service account with Pub/sub editor role. Create key and download the json key file



Step 3: Create a new folder called ‘my\_new\_docker\_build’ on local machine.

Step 4: Copy the json key file to the new folder - ‘my\_new\_docker\_build’

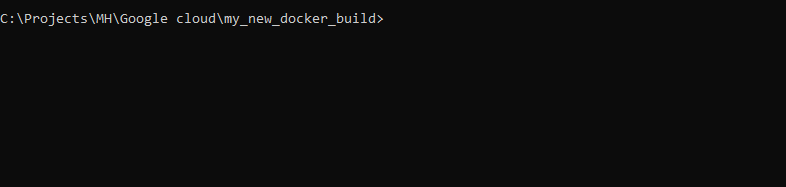
Step 5: Create a developer account on twitter and obtain consumer key, consumer secret, access token and access token secret



Step 6: Create a python script for streaming twitter messages and publishing it to pub/sub topic – ‘my\_script\_stream.py’ and copy it to the folder – ‘my\_new\_docker\_build’

Step 7: Create a ‘Dockerfile’ without any extension. It should contain the pointer to trigger python script, pointer to the json key and all python packages to install. No extension should be given to ‘Dockerfile’. Save it as ‘all files’ type.

Step 8: Open command prompt and navigate to ‘my\_new\_docker\_build’ folder.

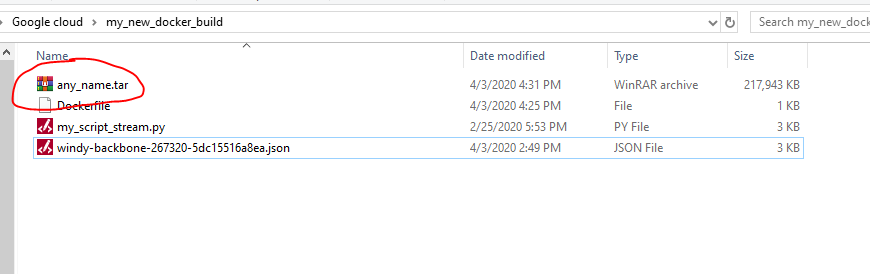


Step 9: Build the docker image by running command : docker build -t any\_name .

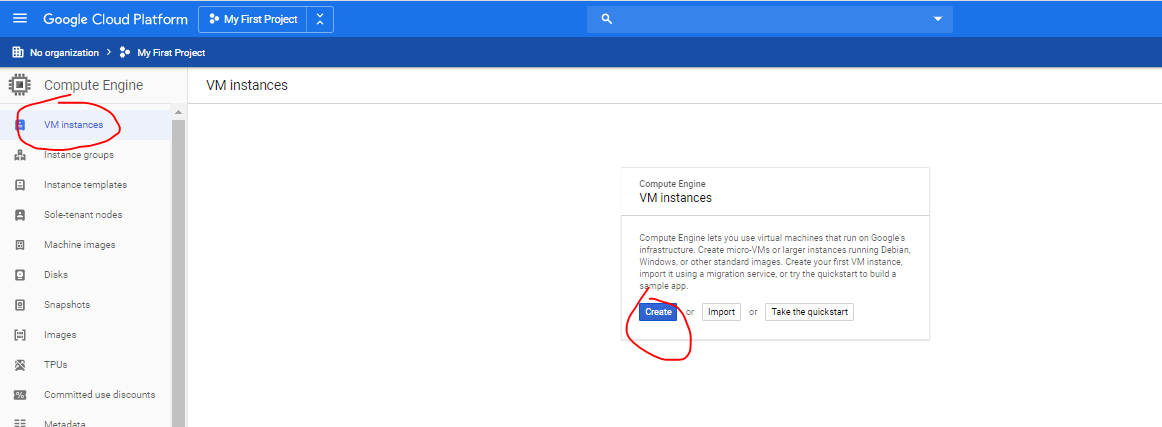
*Prerequisite : Docker has to be installed in desktop*

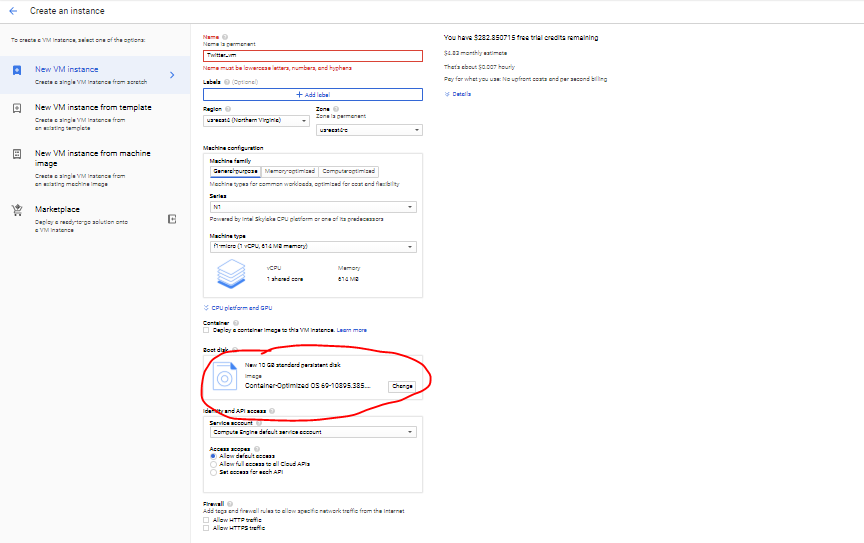
Step 10: Test by running the docker file: docker run any\_name

Step 11: Package the docker file: docker save -o any\_name.tar any\_name

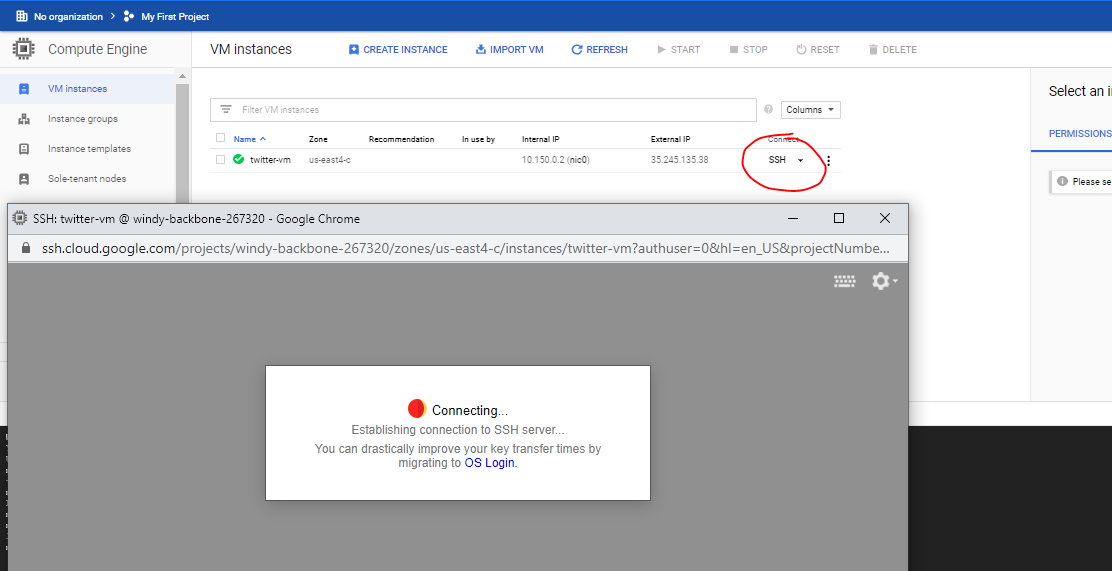


Step 12: Create a new VM instance in Compute Engine GCP

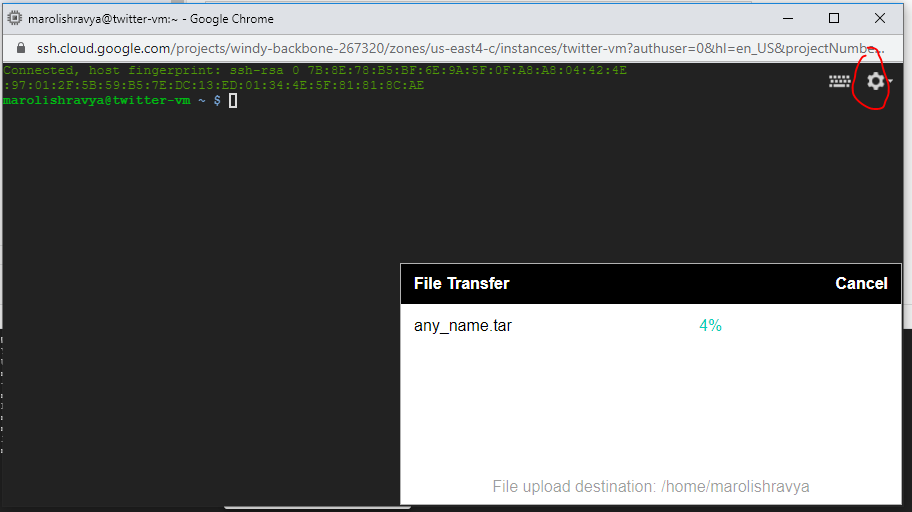




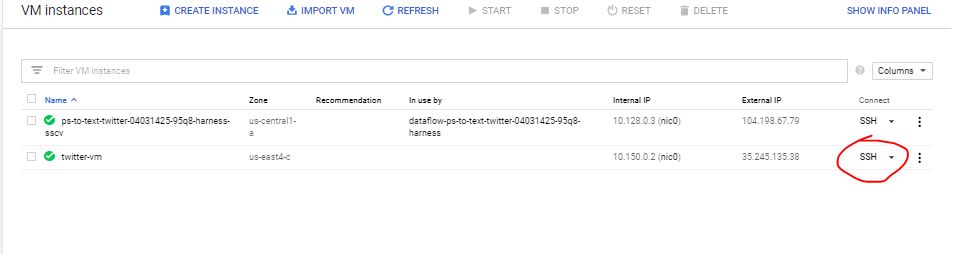
Step 13: Connect to VM using SSH



Step 14: Upload the tar file once in



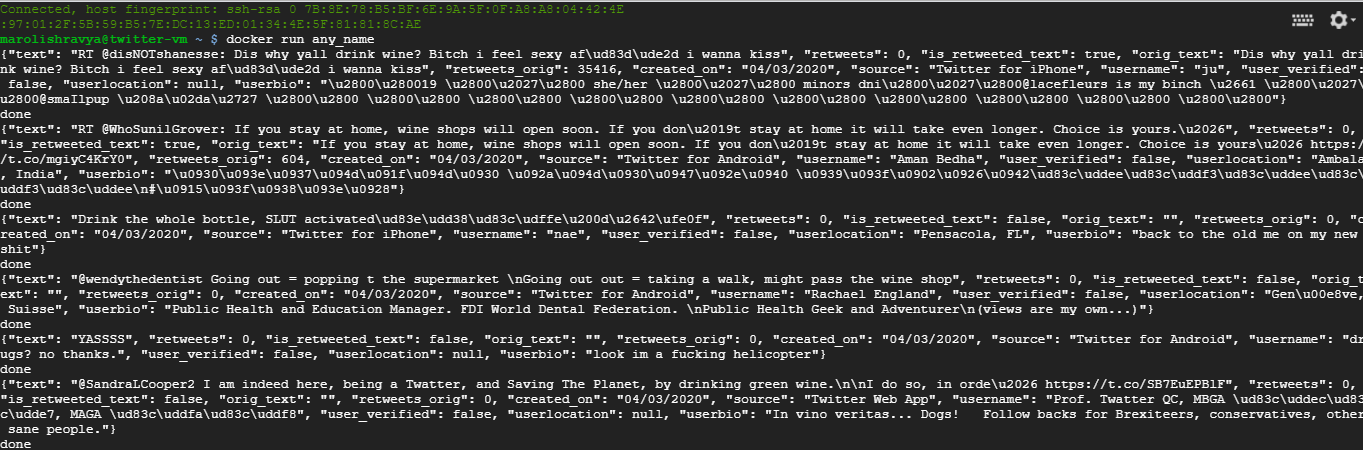
Step 15: SSH into the VM



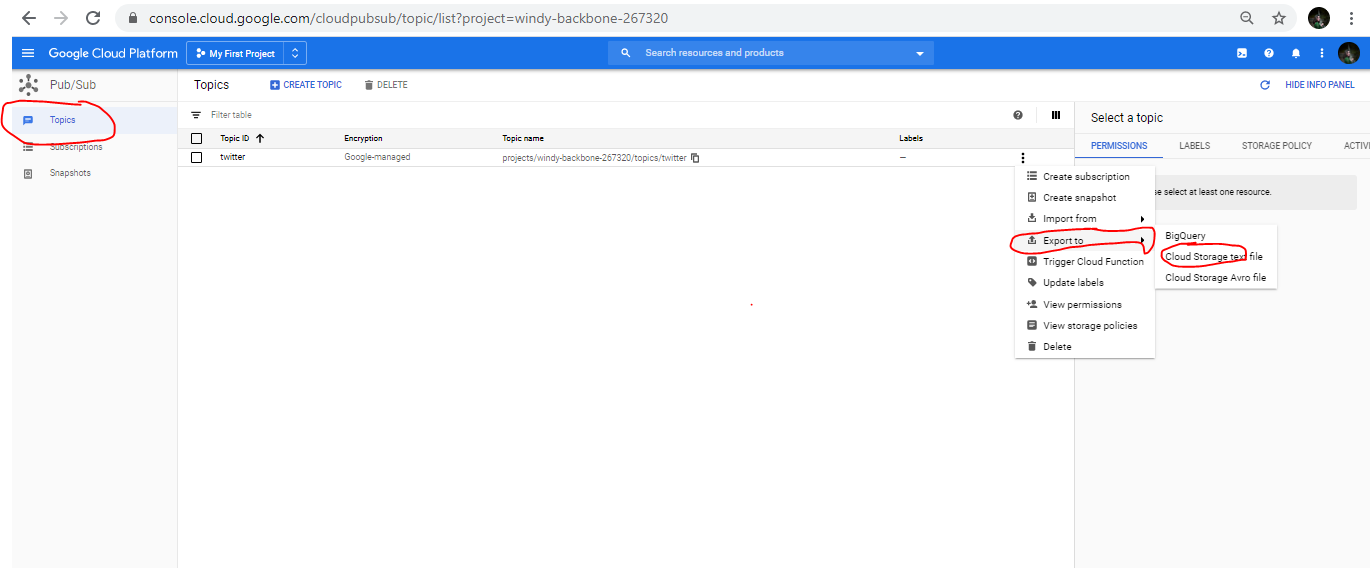
Step 16: Load the tar file - docker load -i any\_name.tar

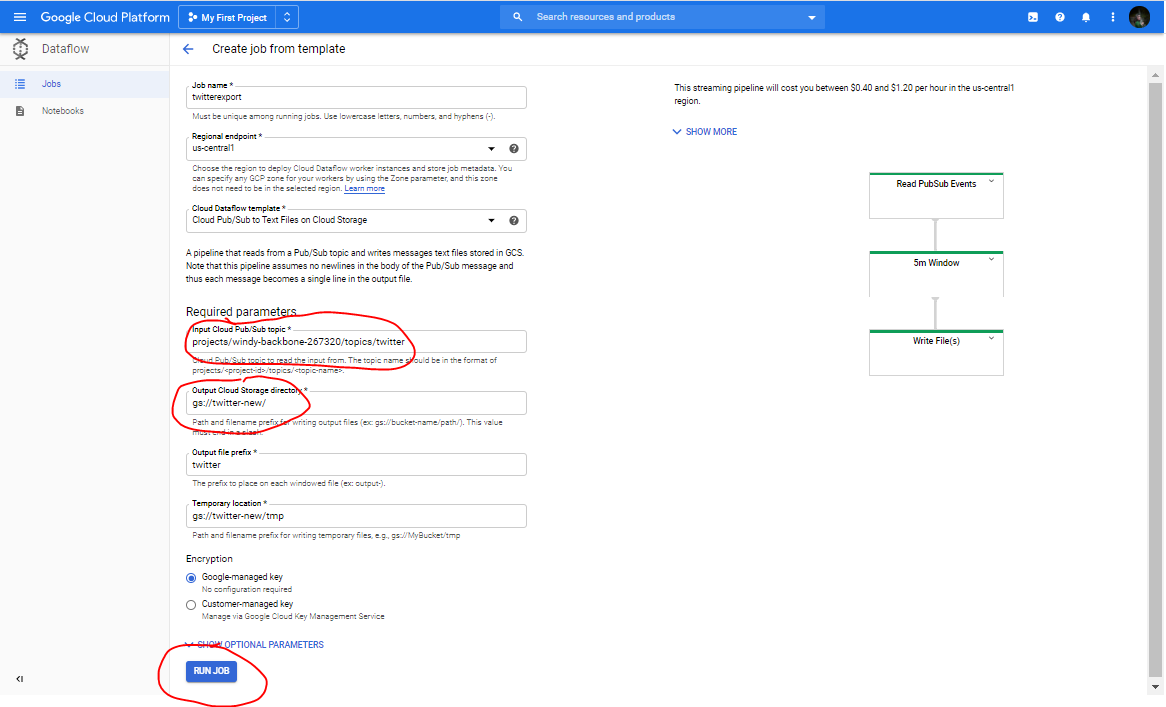
Step 17: Run the docker file – docker run any\_name

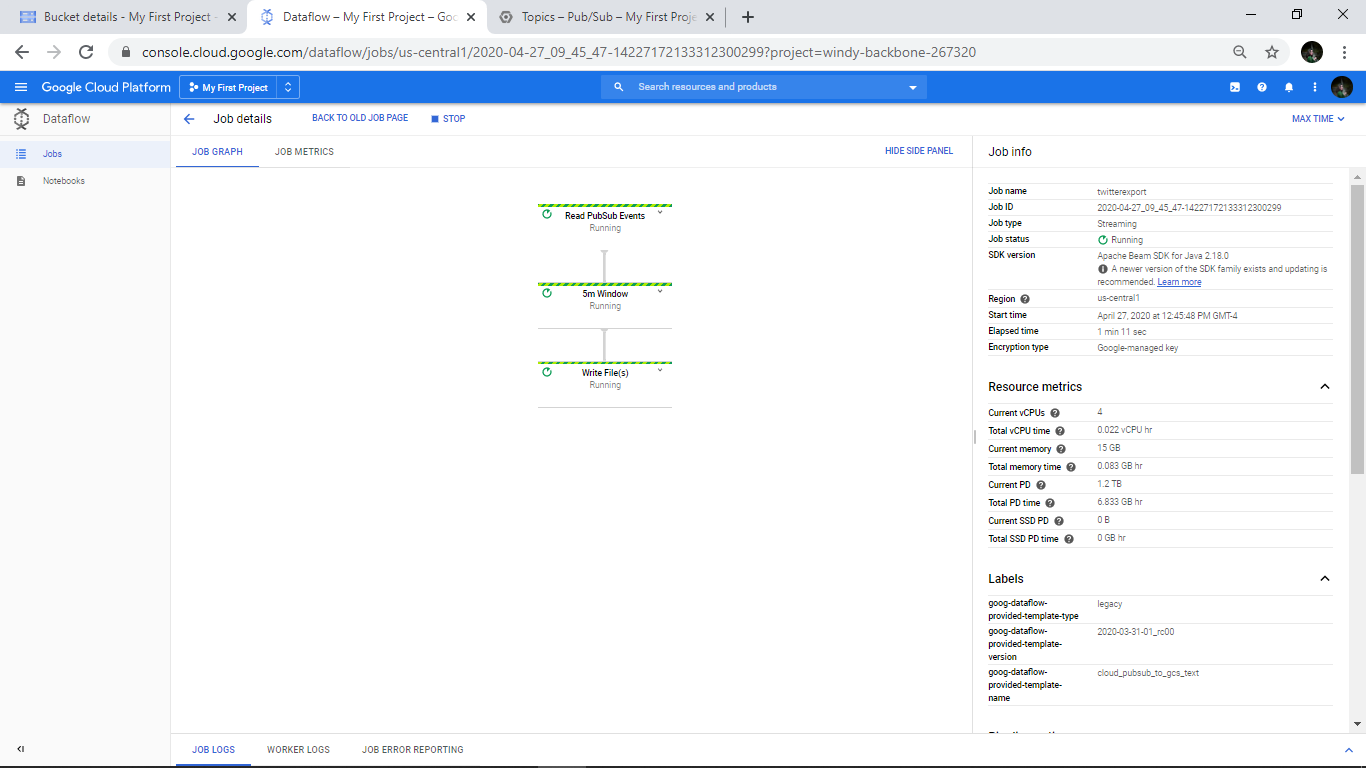
The tweets are streaming in the VM

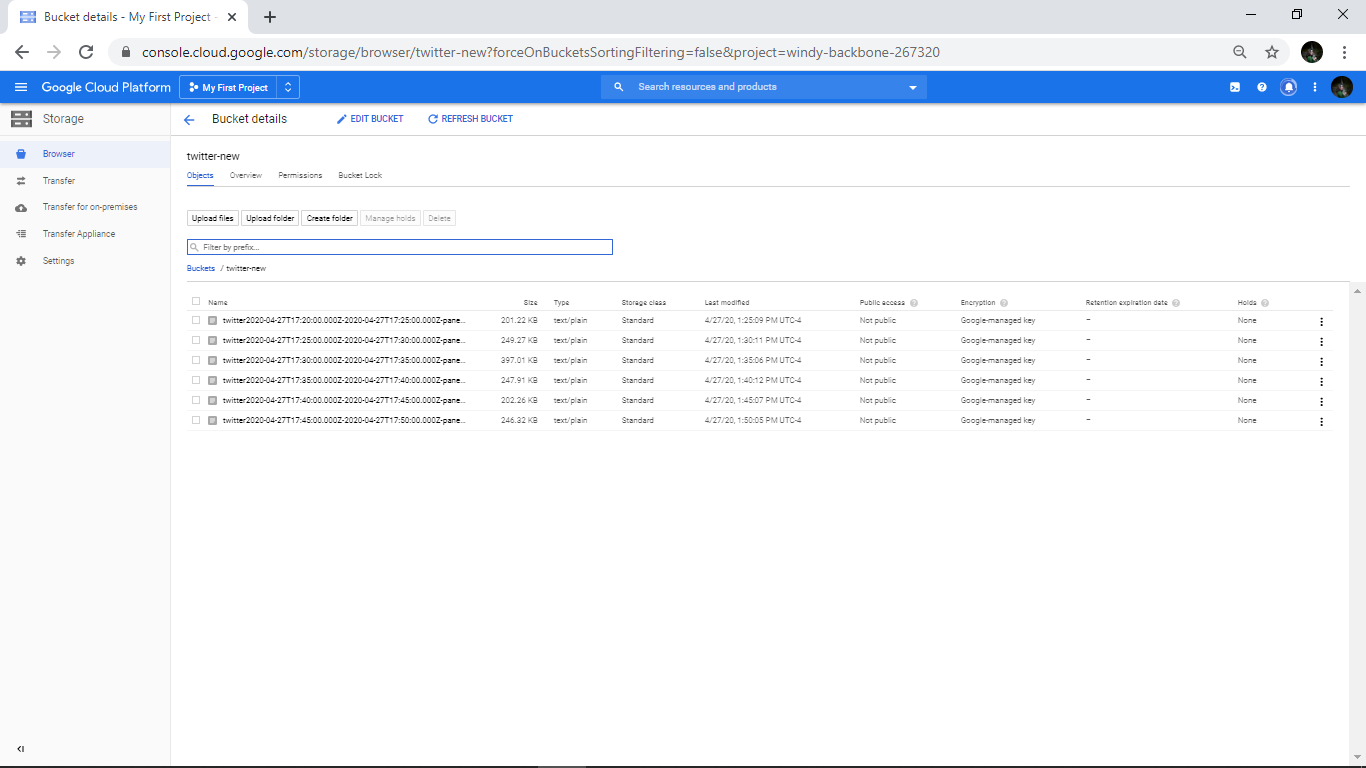


Step 18: Set up the pub/sub topic to export to cloud storage bucket so that the tweets can be viewed.







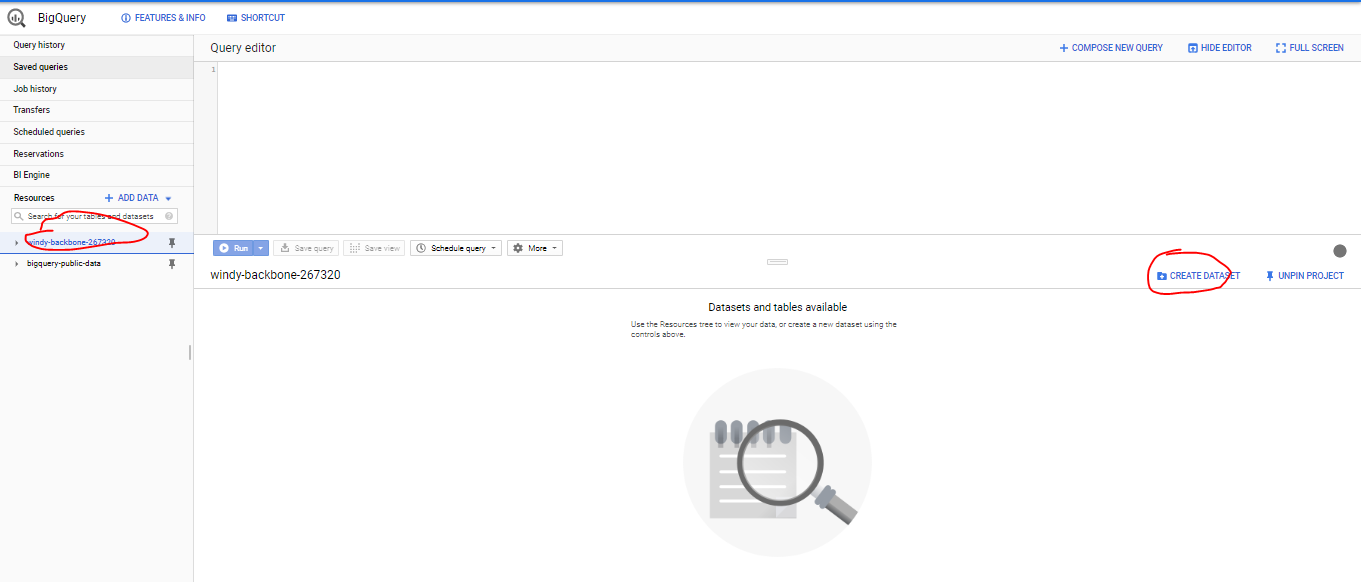


Step 19: Stop the VM if needed to stop streaming (Closing the SSH doesn’t stop streaming)

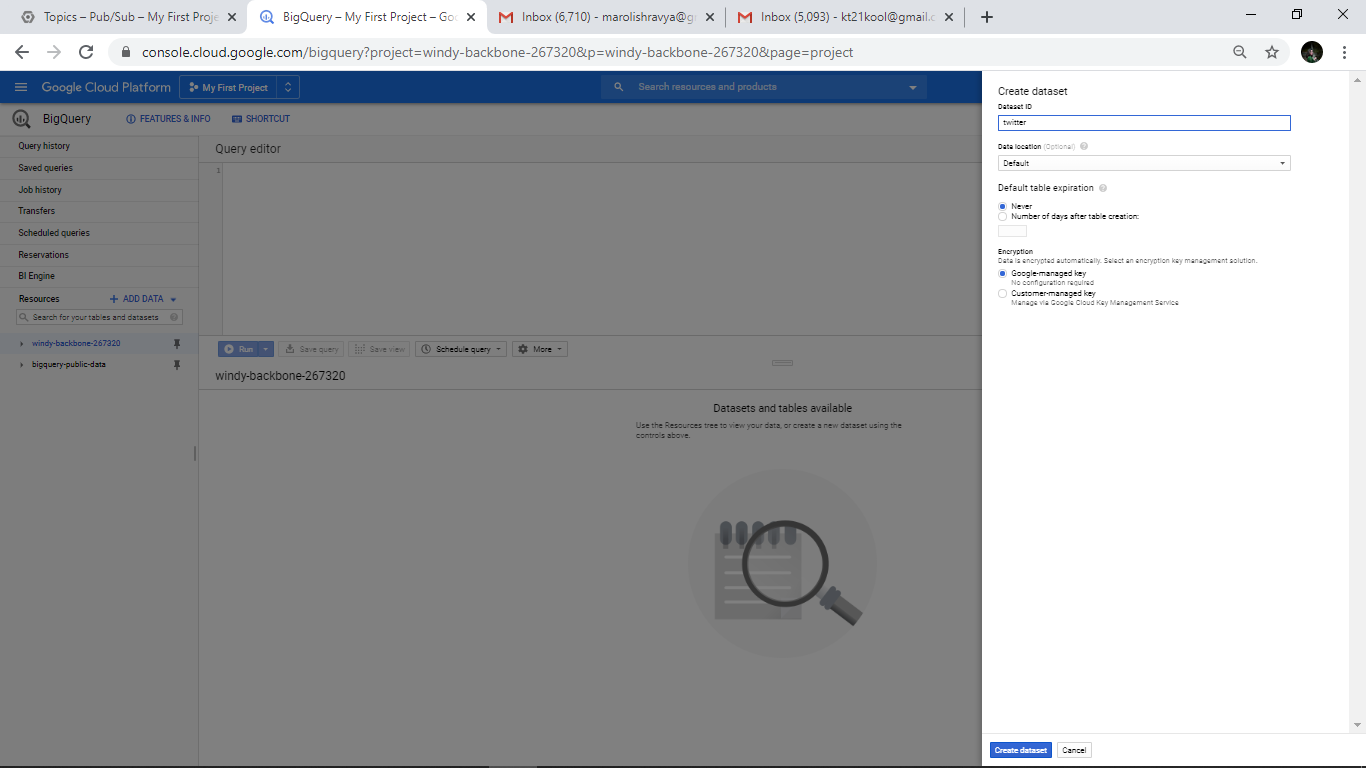
------------------------------------------------------------------------------------------------------------------------------------------

**Goal: Publish twitter messages to pub/sub using dockerized image on GCP. From pub/sub the tweets get stored in Big Query. We create an additional field called ‘wordcount’ which counts the number of words in each tweet.**

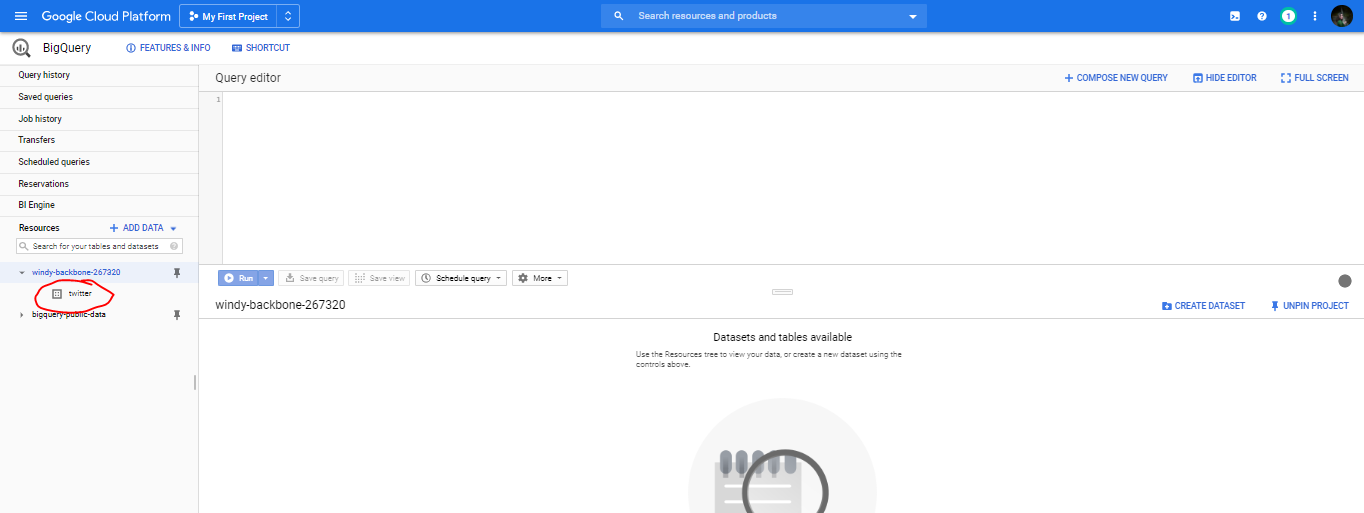
Step 1: Click on ‘Create dataset’ under the project.



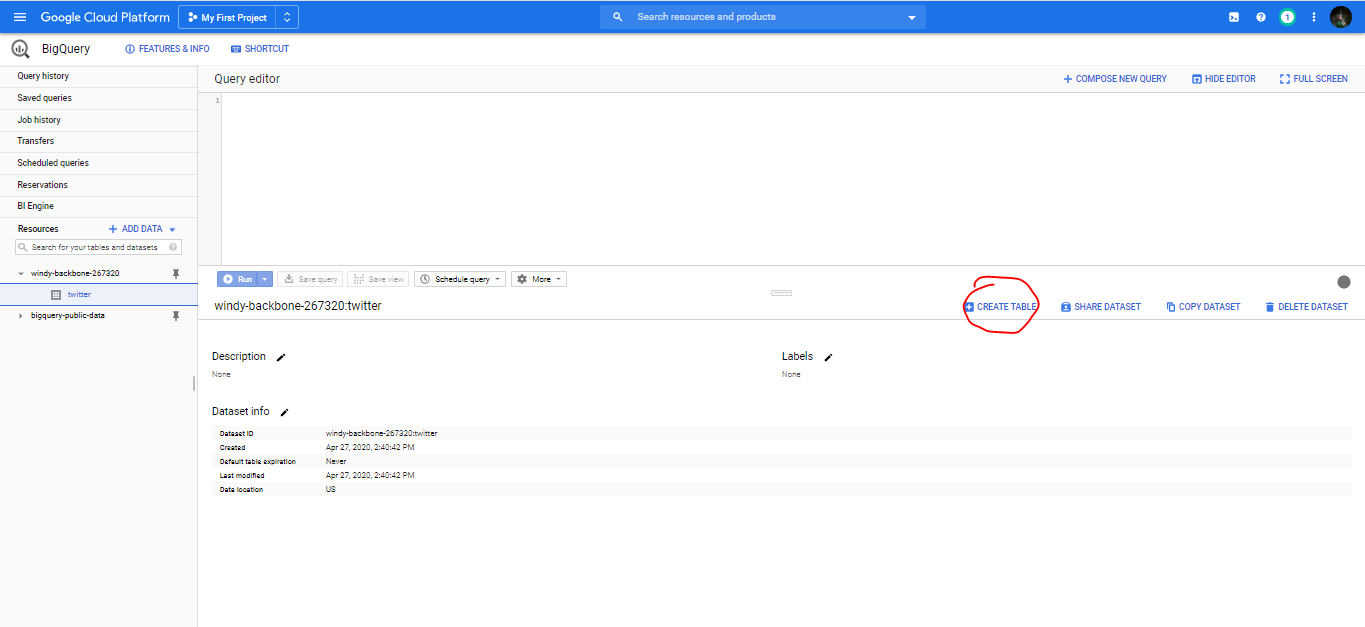
Step 2: Provide dataset name as ‘twitter’ and click on ‘Create dataset’.



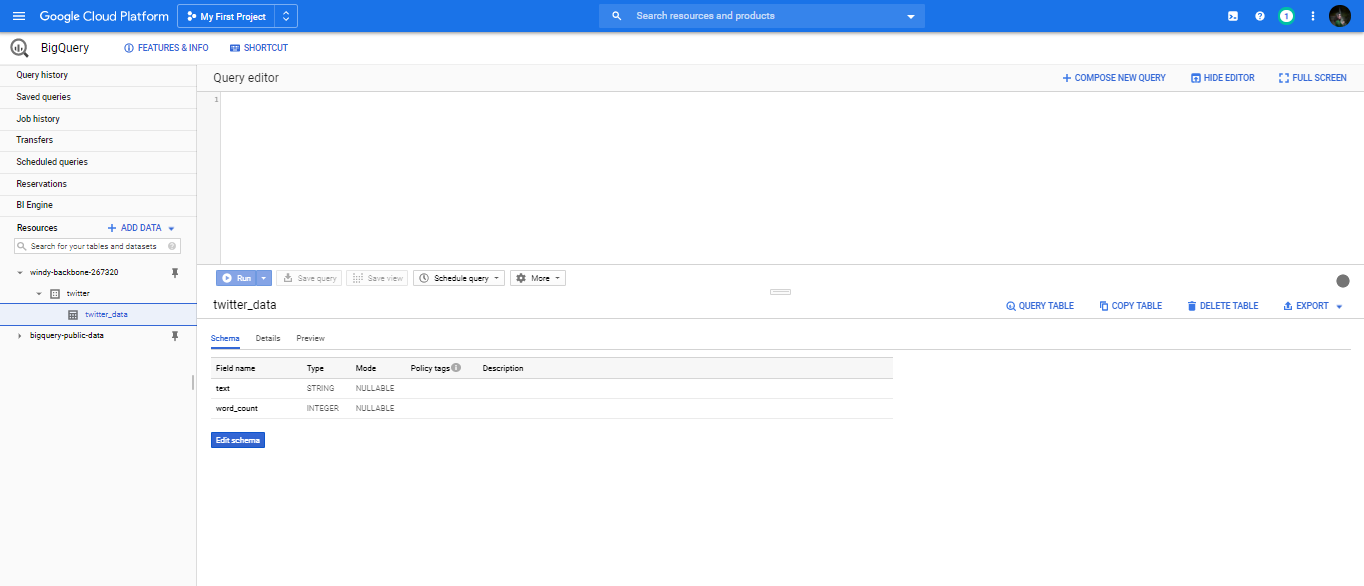
Step 3: A new dataset is created called ‘twitter’ as seen below.



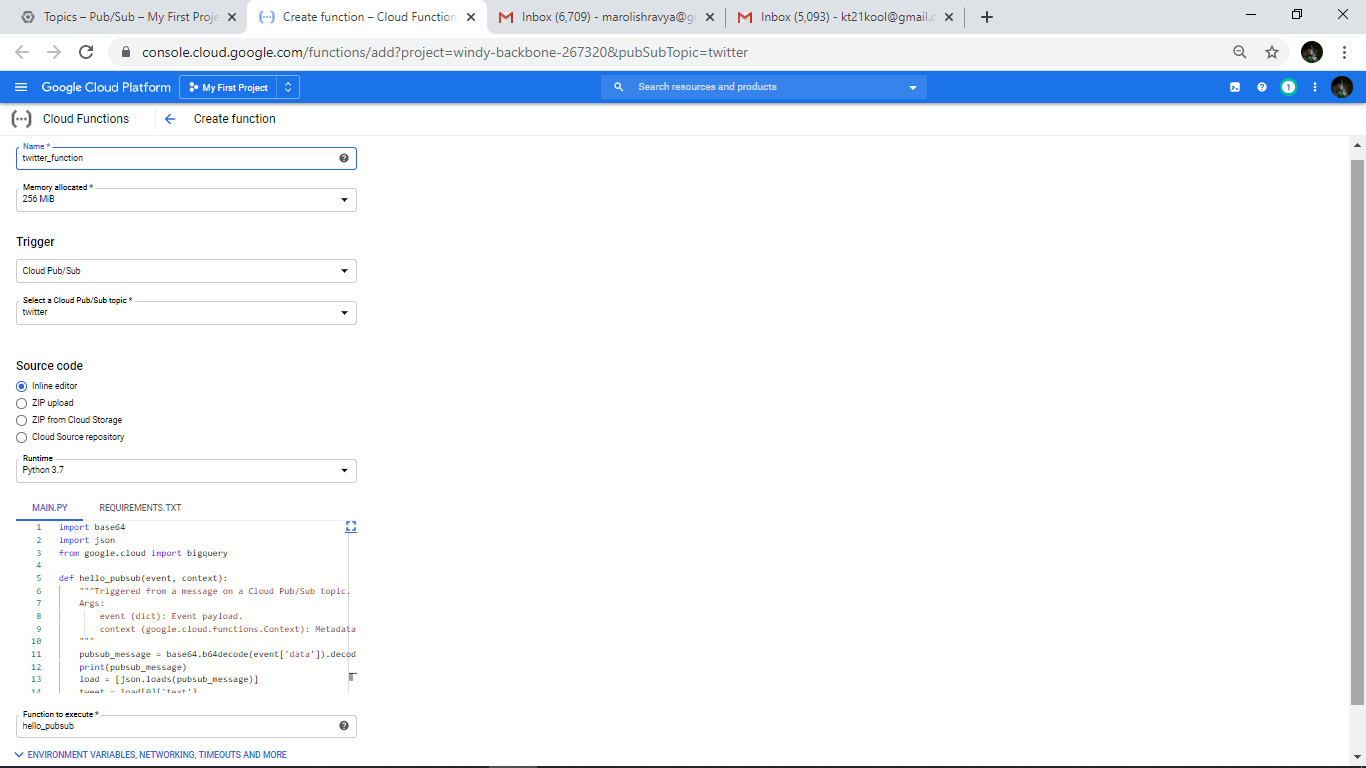
Step 4: Create a table ‘twitter\_data’ under the dataset ‘’.



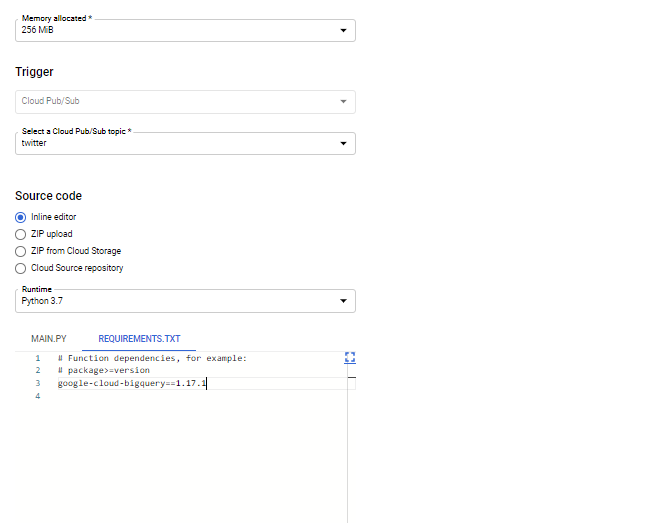
Step 5: Give names of all fields extracted which you want to capture. In the below example, we are trying to extract the twitter text and the word count in each tweet.



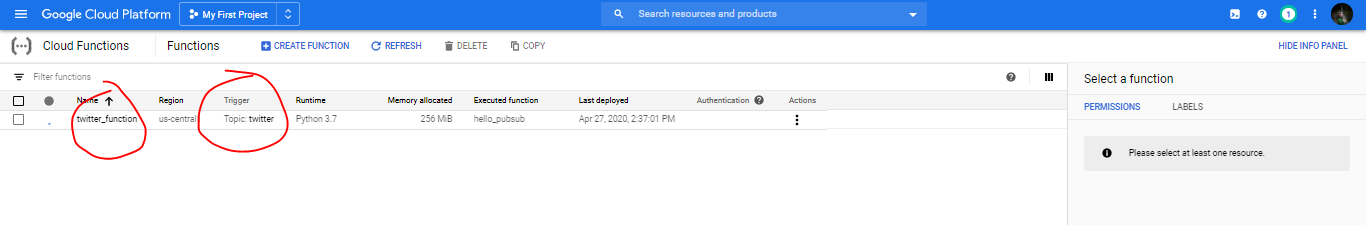
Step 6: Under topic ‘twitter’ click on ‘Trigger cloud function’. Write a python code ‘cloud\_function\_twitter.py’ to add word count field to each tweet so that it is then stored in Big Query.



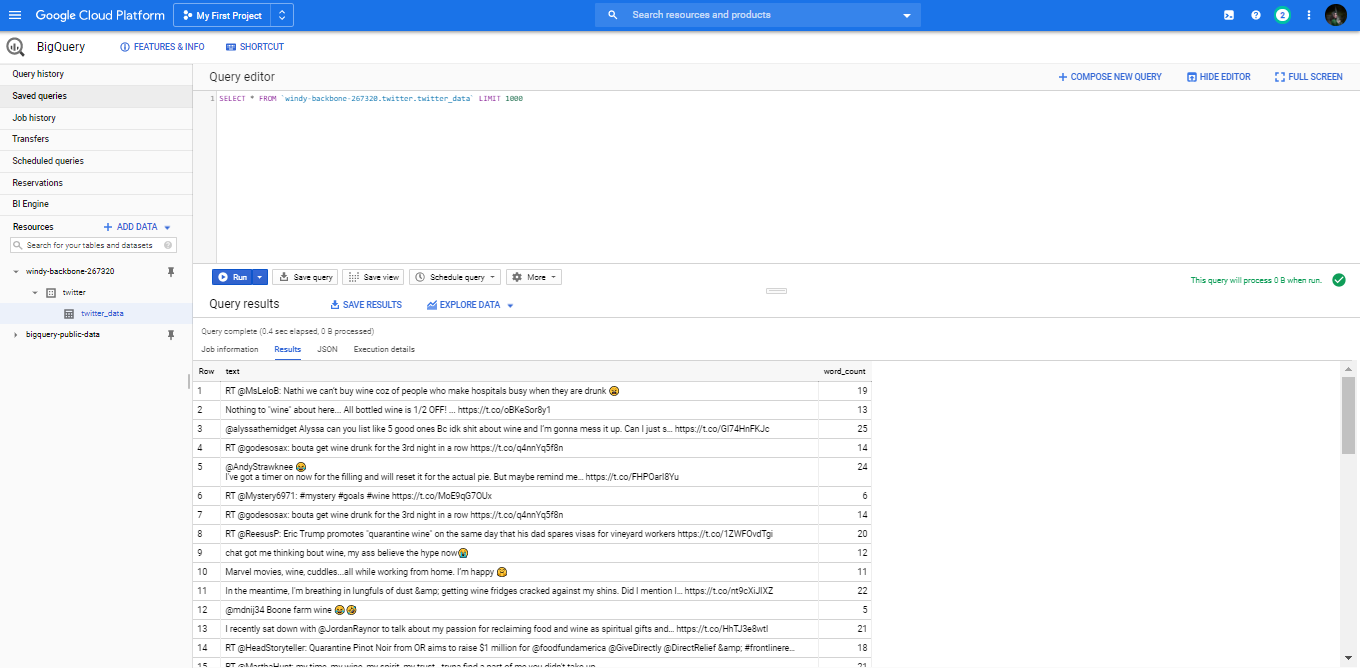
Step 7: Under requirements.txt add package details for google-cloud-bigquery as below.



Step 8: The cloud function is created as seen below. Every time there is data published to ‘twitter’ topic, the cloud function gets triggered.



Step 9: Repeat the steps to ssh to VM and run dockerized image. Verify that the tweets are stored in respective fields in Big Query.



Step 10: Stop the VM if needed to stop streaming (Closing the SSH doesn’t stop streaming)

------------------------------------------------------------------------------------------------------------------------------------------