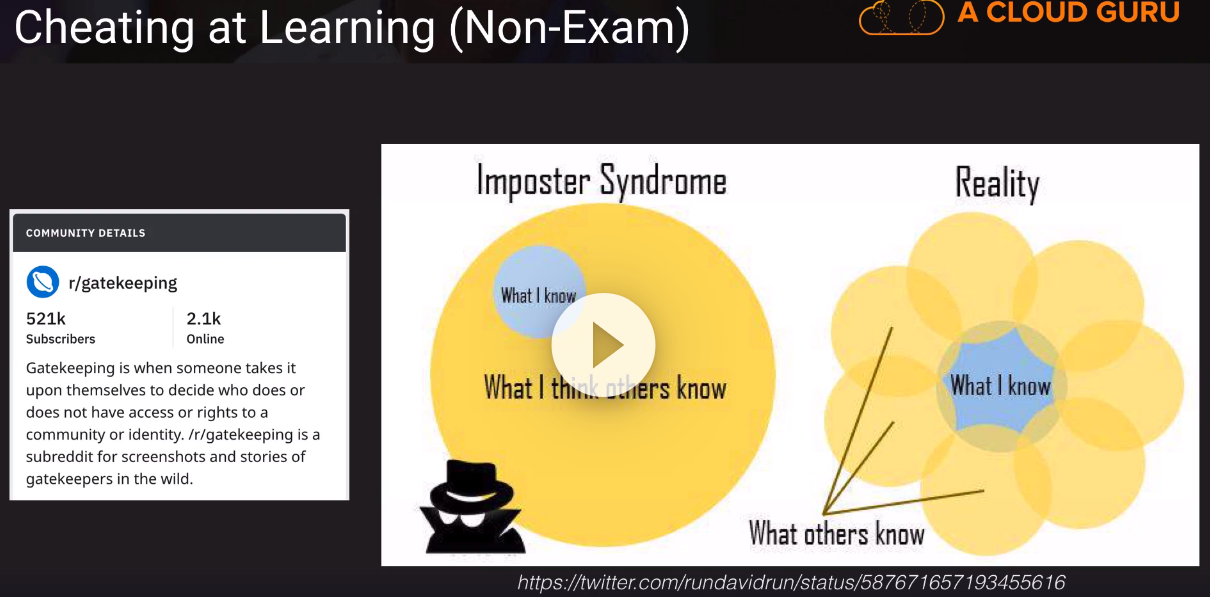
GCP CLOUD GURU



<https://learn.acloud.guru/cloud-playground/cloud-sandboxes>

Cap05 - Milestone - Start

<https://acloud.guru/forums/gcp-certified-associate-cloud-engineer/discussion/-LHq7ia97ot7POrc6Nw7/exam_report_mega-thread>

Cap06 – Budgets

<https://cloud.google.com/billing/docs/how-to/export-data-bigquery>

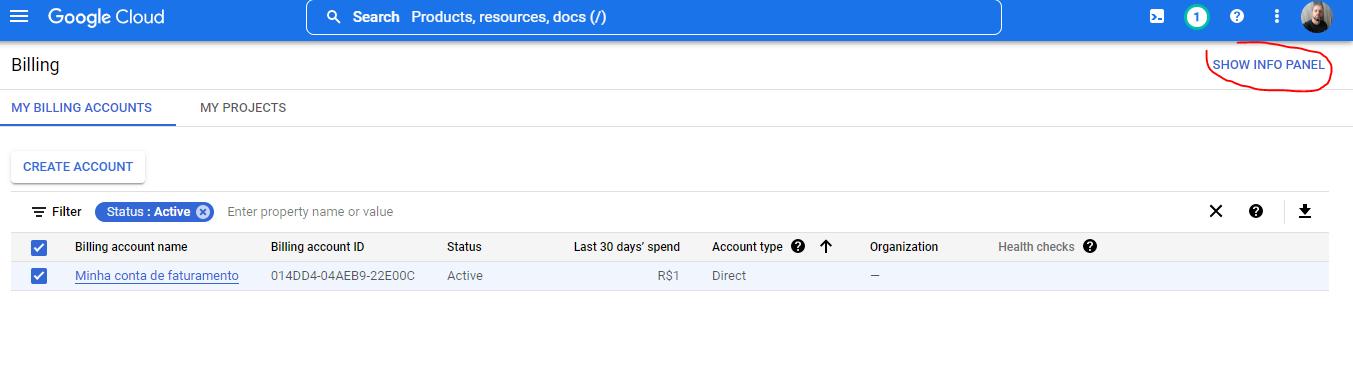
<https://cloud.google.com/billing/docs/how-to/budgets>

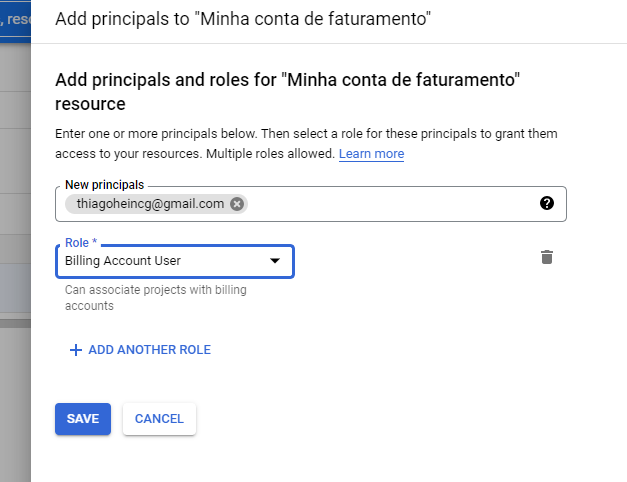
<https://cloud.google.com/billing/docs/how-to/billing-access>

<https://myaccount.google.com/>

<https://support.google.com/a/answer/9807615>

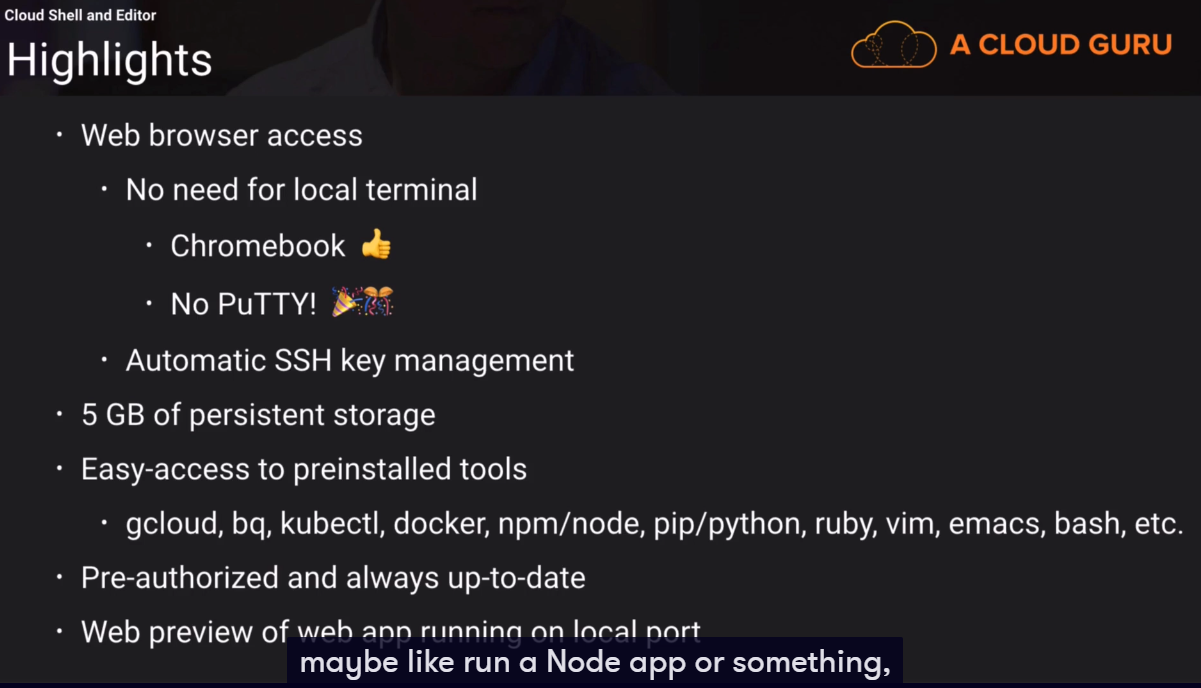
Dar permissão a outra pessoa (billing)





<https://cloud.google.com/shell/>

<https://github.com/ACloudGuru/gcp-cloud-engineer>



Mental Models:

<https://learn.acloud.guru/course/aws-certification-preparation/learn/Learning-Effectively/Mental-Models/watch>

<https://learn.acloud.guru/course/aws-certification-preparation/learn/Learning-Effectively/Mental-Model-Example/watch>

<https://learn.acloud.guru/course/aws-certification-preparation/learn/Learning-Effectively/Zooming-In-and-Out/watch>

Projects:

<https://cloud.google.com/docs/overview/#projects>

***LAB – VIEW SAMPLE BILLING DATA WITH BIGQUERY***

Introduction

Part of working with billing data is working with data exported to BigQuery. In this lab, we are going to view sample billing exports maintained by Google, and conduct queries against a public dataset of billing exports. This will be a fun lab in that we can play around with SQL queries in BigQuery and see what kind of results we can get. Let's get started!

Once you are in your lab project, we will need to get into the BigQuery web console. From the top left menu, scroll down to Big Data, and select BigQuery.

Now that we are in BigQuery, let's look at the sample dataset we are going to work with. We are going to view all columns in our example table to see what fields are included. From the large Query Editor box, copy and paste the following query, then click the Run button:

SELECT \* FROM `cloud-training-prod-bucket.arch\_infra.billing\_data`

The field of cloud-training-prod-bucket.arch\_infra.billing\_data is the public dataset we are working with.

If we click the Results tab underneath, we can view the entire table we are going to work with. Feel free to experiment with other queries such as ordering by cost or usage amount by adding the below string to your query to sort by the column of your choice:

SELECT \* FROM `cloud-training-prod-bucket.arch\_infra.billing\_data` ORDER BY cost DESC

In this query, we are bringing up the entire table contents, but sorting by the highest cost first. You can experiment with other fields as well.

Let's now do some specific queries. In the same Query editor box, delete the existing contents, and enter the below query to find all charges that were more than 3 dollars:

SELECT product, resource\_type, start\_time, end\_time, cost, project\_id, project\_name, project\_labels\_key, currency, currency\_conversion\_rate, usage\_amount, usage\_unit FROM `cloud-training-prod-bucket.arch\_infra.billing\_data` WHERE (cost > 3)

Next let’s find which product had the highest total number of records:

SELECT product, COUNT(\*) FROM `cloud-training-prod-bucket.arch\_infra.billing\_data` GROUP BY product LIMIT 200

Looks like Pub/Sub is pretty popular here...

Finally, let’s see which product most frequently cost more than a dollar:

SELECT product, cost, COUNT(\*) FROM `cloud-training-prod-bucket.arch\_infra.billing\_data` WHERE (cost > 1) GROUP BY cost, product LIMIT 200

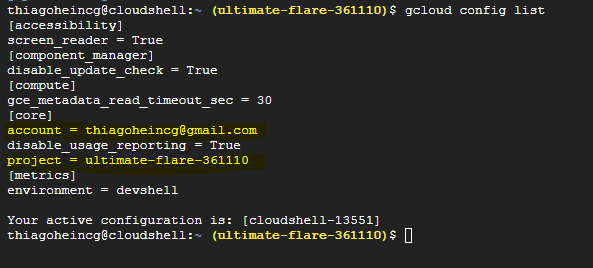
That will conclude this lab. You can quit anytime, however you are also free to experiment with other query types as well. The great thing about working with BigQuery is the flexibility of discovering what you can learn from your data!

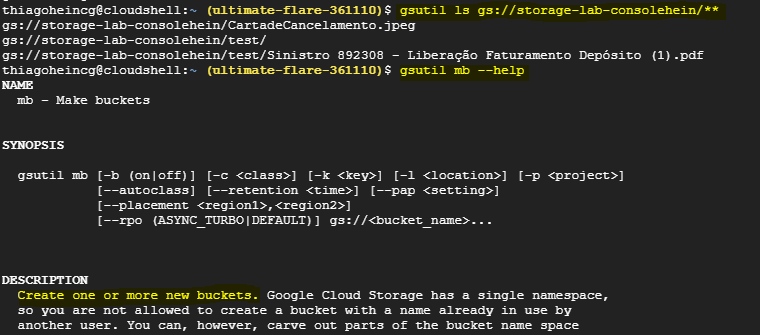
**CHAPTER 7 – BASIC SERVICES**

Google Storage

<https://acloud.guru/forums/gcp-certified-associate-cloud-engineer/discussion/-LU4YI7_daGa6vpcGC4P/Ask%20for%20%22enable%20billing%22%20when%20i%20try%20to%20use%20Cloud%20Storage>

<https://cloud.google.com/storage/docs/access-control/making-data-public>





<https://cloud.google.com/storage/docs/locations>

<https://cloud.google.com/storage/docs/locations>

COMMANDS

pwd

ls

gcloud config list

gsutil ls

gsutil ls gs://storage-lab-console/

gsutil ls gs://storage-lab-console/\*\*

gsutil mb --help

gsutil mb -l northamerica-northeast1 gs://storage-lab-cli

gsutil ls

gsutil label get gs://storage-lab-console/

gsutil label get gs://storage-lab-console/ >bucketlabels.json

cat bucketlabels.json

gsutil label get gs://storage-lab-cli/

gsutil label set bucketlabels.json gs://storage-lab-cli/

gsutil label get gs://storage-lab-cli/

gsutil label ch -l "extralabel:extravalue" gs://storage-lab-cli

gsutil versioning get gs://storage-lab-cli/

gsutil versioning set on gs://storage-lab-cli/

gsutil versioning get gs://storage-lab-cli/

gsutil ls gs://storage-lab-cli/

gsutil cp README-cloudshell.txt gs://storage-lab-cli/

gsutil ls gs://storage-lab-cli/

gsutil ls gs://storage-lab-cli/

gsutil ls -a gs://storage-lab-cli/

gsutil rm gs://storage-lab-cli/README-cloudshell.txt

gsutil ls gs://storage-lab-cli/

gsutil ls -a gs://storage-lab-cli/

gsutil cp gs://storage-lab-console/\*\* gs://storage-lab-cli/

gsutil ls gs://storage-lab-cli/

gsutil ls -a gs://storage-lab-cli/

gsutil acl ch -u AllUsers:R gs://storage-lab-cli/Selfie.jpg

03

gcloud config get-value project

gcloud compute instances list

gcloud services list

gcloud services list -h

gcloud services list --enabled

gcloud services list --available

gcloud services list --available | grep compute

gcloud services -h

gcloud compute instances list

gcloud services list

gcloud compute instances create myvm

gcloud compute instances delete myvm

gcloud compute instances list

Rundown Gcloud

<https://cloud.google.com/sdk/docs/configurations>

<https://cloud.google.com/sdk/docs/properties>

<https://cloud.google.com/sdk/gcloud/reference/>

<https://cloud.google.com/sdk/gcloud/>

GCE In Data Out

<https://cloud.google.com/compute/docs/metadata/overview>

<https://cloud.google.com/sdk/gcloud/reference/topic/filters>

# Check the elected project

gcloud config list

# Show any .ssh folder

pwd

ls

ls -a .ssh

# Get our bearings in Cloud Shell

whoami

hostname

curl api.ipify.org

# Check that we have nothing running

gcloud compute instances list

# Don't create a default VM

# Cancel: gcloud compute instances create myhappyvm

# Look at how to set the machine type

gcloud compute instances create myhappyvm -h

gcloud compute instances create myhappyvm --help

gcloud compute machine-types list

# See how to filter

gcloud topic filters

# Show some free-tier-eligible options

gcloud compute machine-types list --filter="NAME:f1-micro"

gcloud compute machine-types list --filter="NAME:f1-micro AND ZONE~us-west"

# Set our defaults to Los Angeles

gcloud config set compute/zone us-west2-b

gcloud config set compute/region us-west2

# Start our instance

gcloud compute instances create --machine-type=f1-micro myhappyvm

ping -c 3 myhappyvm

ping -c 3 internalipaddress

ping -c 3 externalipaddress

# Connect to the VM

ssh externalipaddress

gcloud compute ssh myhappyvm

# Get our bearings -- Skip?

whoami

hostname

curl api.ipify.org

# Get back to Cloud Shell

exit

curl api.ipify.org

# Look at the Cloud Shell .ssh files

cd .ssh

ls

cat google\_compute\_engine.pub

head -n 10 google\_compute\_engine

# Log back onto the VM

gcloud compute ssh myhappyvm

# See that our key is authorized

cd .ssh

ls

cat authorized\_keys

cd ..

# Check out the metadata

curl metadata.google.internal/computeMetadata/v1/

curl -H "Metadata-Flavor: Google" metadata.google.internal/computeMetadata/v1/

curl -H "Metadata-Flavor: Google" metadata.google.internal/computeMetadata/v1/project/

curl -H "Metadata-Flavor: Google" metadata.google.internal/computeMetadata/v1/project/project-id

curl -H "Metadata-Flavor: Google" metadata.google.internal/computeMetadata/v1/project/attributes/

curl -H "Metadata-Flavor: Google" metadata.google.internal/computeMetadata/v1/project/attributes/ssh-keys

# Look at some instance metadata

curl -H "Metadata-Flavor: Google" metadata.google.internal/computeMetadata/v1/instance/

curl -H "Metadata-Flavor: Google" metadata.google.internal/computeMetadata/v1/instance/name

curl -H "Metadata-Flavor: Google" metadata.google.internal/computeMetadata/v1/instance/service-accounts/default/

curl -H "Metadata-Flavor: Google" metadata.google.internal/computeMetadata/v1/instance/service-accounts/default/email

# See what gcloud knows

gcloud config list

# Look at our buckets

gsutil ls

gsutil ls gs://storage-lab-cli/

# Attempt to delete the VM from within the VM

gcloud compute instances delete myhappyvm

# Exit back to Cloud Shell and actually delete the VM

exit

gcloud compute instances delete myhappyvm

GCP via console

<https://cloud.google.com/compute/docs/access/create-enable-service-accounts-for-instances>

<https://cloud.google.com/compute/docs/instances/startup-scripts>

<https://cloud.google.com/compute/docs/instances/create-use-preemptible>

<https://cloud.google.com/compute/docs/instances/create-start-instance>

**Cap08 - Basic Services Challenge**