

## Part 1 :

Part 1 takes care of creating a VM and installing the flask application on the VM. It also modifies the firewall rules such that inbound requests can access the flask application running on port 5000. First I created an image(f1 micro image) in the us-west-1b zone. I'm also using this image to spin off an instance that gets created. I'm reading the shell script file in my create\_instance.py and in the shell script file I'm taking care of installing the flask application(cloning the flask application and installing flask on specific path) on port number 5000 using the nohup command.

I'm also attaching the network tag to my instance using the setTags API. I'm also adding a firewall rule "allow-5000" to allow all external requests to access our flask app running on port number 5000.

## Part 2 :

Part 2 first takes care of creating a snapshot of the current VM instance that was currently spun off. I give the projectId, the zone and the name of the persistent disk that I want to snapshot and specify the name of the snapshot in the snapshot body and use the disks.createSnapshot() api library call in python to create a snapshot of the current VM instance. A snapshot is basically the current state of the machine.

From the snapshot that was created I'm creating 3 cloned instances. (I'm specifying the snapshot source in the sourceSnapshot field in initializeParams. For each of the cloned instances I'm setting a starttime and an endtime before and after the code to check the time taken to create an instance. I'm outputting that into another file TIMING.md to compare times taken to spin off the VM instances.

## Part 3 :

Part 3 starts off by copying the json credentials file, the file to create an instance and the shell script file which has the code to install the flask application to the storage bucket that is present in the first VM instance. The initial shell script also takes care of running the part3-init.py file to spin off the first vm instance. Part3-init.py file takes care of authenticating our service account credentials( with the service account we created) to google cloud and also takes care of creating the first VM instance given the projectId, Zone and instance name. Once the first VM gets created and the bucket has the necessary files, the part3.py file present inside the part3-inter file gets executed which reads off the shell script ( startup-script.sh which contains the code to install the flask application on port 5000) and this file is responsible for spinning of the second vm instance, which will contain the flask application.