ARQUITETURA DA SOLUÇÃO

AUTOMAÇÃO NA GOOGLE CLOUD UTILIZANDO PYTHON – IAM SERVICE ACCOUNTS – PARTE 1
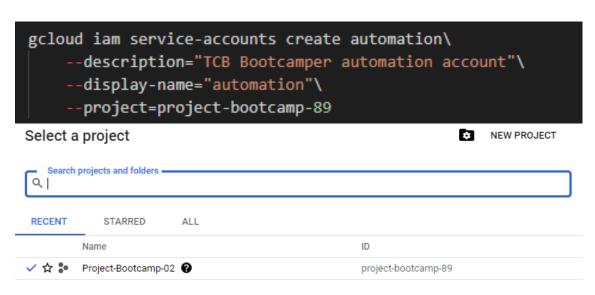
BOOTCAMP GOOGLE CLOUD

Requirements
- Install Python
- Install SDK
  After install SDK update with commands below

  ➢ pip install - upgrade google-cloud-storage
  ➢ pip install - upgrade google-cloud-compute
  ➢ pip install - upgrade google-api-python-client
  ➢ pip install - upgrade oauth2client

- Creation Project in GCP



```
gcloud iam service-accounts create automation\
    --description="TCB Bootcamper automation account"\
    --display-name="automation"\
    --project=project-bootcamp-89
```

Select a project                                     NEW PROJECT

Search projects and folders
🔍 |

RECENT     STARRED     ALL

| Name | ID |
| --- | --- |
| ✓ ☆ Project-Bootcamp-02 ❓ | project-bootcamp-89 |

- Creation IAM Service and change roles

```
gcloud iam service-accounts create create automation \
    --description="TCB Bootcamper automation account" \
    --display-name="automation1"

gcloud iam roles describe roles/storage.admin
gcloud projects add-iam-policy-binding bootcamp-gcp-03 \
--member="serviceAccount:automation@project-bootcamp-89.iam.gserviceaccount.com"\
--role="roles\storage.admin"
```

- Creation key and download.

```
# get full email id of new service account
emailId=$(gcloud iam service-accounts list --filter="email ~ ^automation" --format='value(email)')
# download key for service account
gcloud iam service-accounts keys create automation.json --iam-account $emailId
```

- Creation, Upload and List Bucket

```
#Create bucket storage
gsutil mb gs://iam-service-account-2

Validando o bucket criado anteriormente
##Copy
gsutil cp "FileName"
##List
gsutil ls gs://iam-service-account-2
```
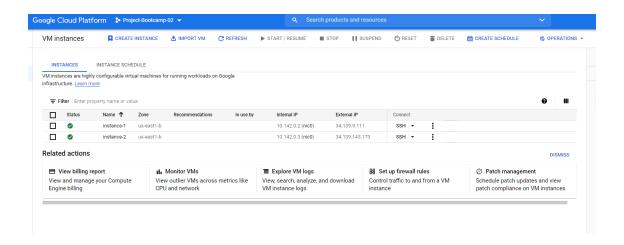
```
FelipePiton@Intrepid MINGW64 ~/Desktop/Bootcamp/GCP/Missão-1
$ python ./lista_storage.py bootcamp-gcp-felipepiton-efbcce1a4110.json
[<Bucket: iam-service-account-1>]
```

# Desafio SDK



After completing the module 2 Hands-on Project, your manager calls you to propose an extra challenge and he asks: "Would it be possible for this python script, making a small adjustment, to read all the virtual machines that are created there in the Google Cloud?" And now? Is it possible to read all virtual machines created on Google Cloud? How could it be that by making a small adjustment to the file, it would be possible to complete this challenge? Tip for creating the virtual machines: In the contents, Herbert teaches you how to create 3virtual machines with just one command using gcloud and you can use this information to create the virtual machines .SDK Challenge.

```python
# Some prerequisites are needed
# To execute in your gcloud environment run the line below
# python gcp_list_vms.py <your-key.json>
from pprint import pprint
from googleapiclient import discovery
from googleapiclient.discovery import build
from oauth2client.client import GoogleCredentials
from six.moves import input

import googleapiclient.discovery
import argparse
import os
os.environ["GOOGLE_APPLICATION_CREDENTIALS"] = "C:/Users/FelipePiton/Desktop/Felipe/Curso/Bootcamp/GCP/Missão-1/project-bootcamp-89-4c5ee838c44e.json"

# Project ID for this request.
project = "project-bootcamp-89"
zone = "us-east1-b"
compute = googleapiclient.discovery.build('compute', 'v1')
space = ' '

def list_instances(compute, project, zone):
    result = compute.instances().list(project=project, zone=zone).execute()

    print ("{:<7} {:<11} {:<13} {:<12} {:<12} {:<15} {:<0}".format('NAME','ZONE','MACHINE_TYPE','PREEMPTIBLE','INTERNAL_IP','EXTERNAL_IP','STATUS'))

    for item in result['items']:
        if item['status'] == "RUNNING":

            var_name = item['name']
            var_zone = item['zone']
            var_machine_type = item['machineType']
            var_preemptible = item['scheduling']['preemptible']
            var_internal_ip = item['networkInterfaces'][0]['networkIP']
            var_external_ip = '{:<14}'.format(item['networkInterfaces'][0]['accessConfigs'][0]['natIP'])
            var_status = item['status'].rjust(0)

            print ("{:<7} {:<11} {:<13} {:<12} {:<12} {:<15} {:<0}".format(var_name, var_zone.rsplit('/', 1)[-1], var_machine_type.rsplit('/', 1)[-1], str(var_preemptible),var_internal_ip, var_external_ip,var_status))
    return result['items'] if 'items' in result else None
list_instances(compute, project, zone)
```

```
FelipePiton@Intrepid MINGW64 ~/Desktop/Felipe/Curso/Bootcamp/GCP/Missão-1
$ py ./gcp_list_vms.py
NAME       ZONE      MACHINE_TYPE  PREEMPTIBLE  INTERNAL_IP  EXTERNAL_IP     STATUS
instance-1 us-east1-b e2-medium    False        10.142.0.2   34.139.9.111    RUNNING
instance-2 us-east1-b e2-medium    False        10.142.0.3   34.139.143.173  RUNNING

FelipePiton@Intrepid MINGW64 ~/Desktop/Felipe/Curso/Bootcamp/GCP/Missão-1
$
```