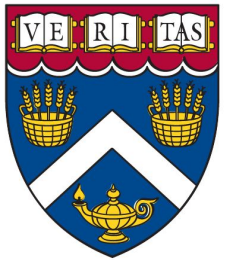


HARVARD
Extension School

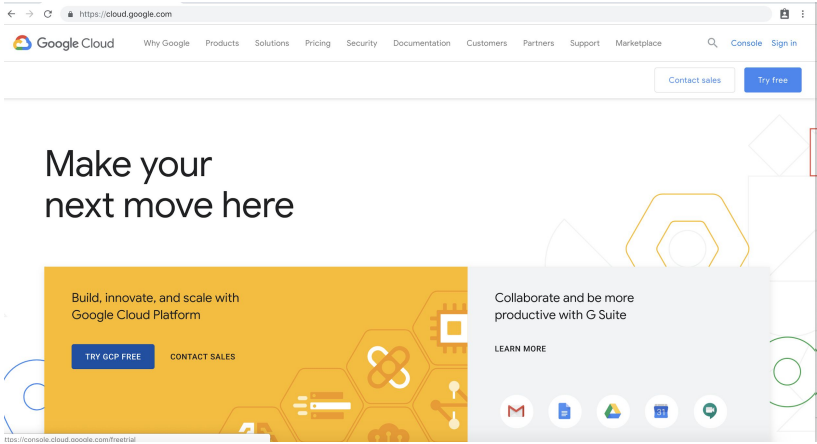
Week 12
Google Cloud Platform (GCP)

Students will create a Google Cloud account and build a computation environment similar to the one they created with the AWS using the console and the API call.

- Create Google account
- Build a computing environment
  - Google Computing
  - Google VPC
  - Google Database
  - Google Function
- Managing resources and credentials
- Google SDK

Google Cloud Platform (GCP) Account

1. Go to https://cloud.google.com
2. Click on “Try GCP Free” blue button
3. Sign in to your google account
4. This only works the first time GCP users. The free trial is not available if you tried it before.
5. Set your country,
6. Click Agree and Continue
7. Select the account type “Individual”. Fill in your information and provide a credit card. You won't be billed until and unless you go over $300
8. Navigate to the Google Cloud Console:https://console.cloud.google.com/



Groups of Services

- Compute:
  - Virtual Machines
  - Container and Container Management
  - Functions and Lambdas
- Security and Identity
- Managing and Monitoring
- Storage
  - Files
  - Archival Storage
  - Persistent Disks for VMs
- Networking and Machine Learning
- Developer tools: https://cloud.google.com/docs/overview/developer-and-admin-tools

Map AWS services to Google Cloud Platform

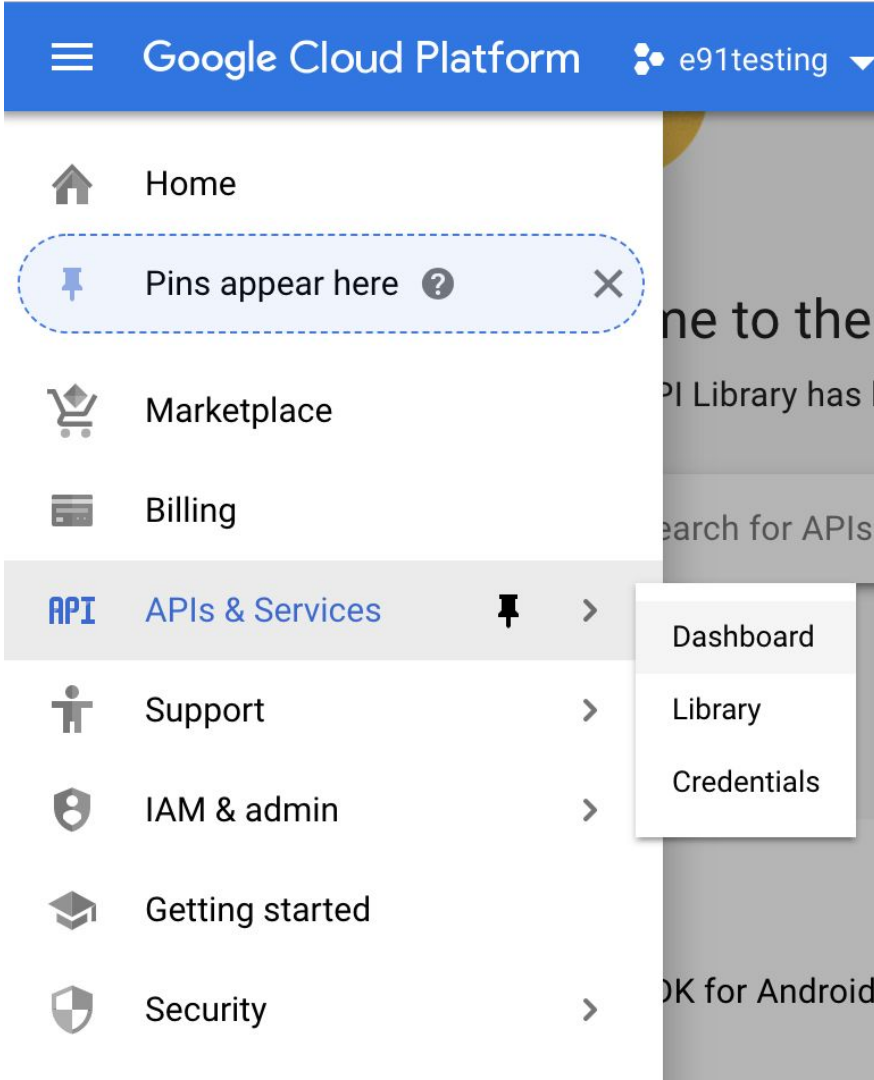
https://cloud.google.com/docs/compare/aws/

Compute	
Amazon Elastic Compute Cloud	Compute Engine
Serverless Functions	
AWS Lambda	Cloud Functions
Network	
Virtual Networks	
Amazon Virtual Private Cloud	Virtual Private Cloud
Load Balancer	
Elastic Load Balancer	Cloud Load Balancing
Domains and DNS	
Amazon Route 53	Google Domains, Cloud DNS
Storage	
Object Storage	
Amazon Simple Storage Service	Cloud Storage
Block Storage	
Amazon Elastic Block Store	Persistent Disk
Archival Storage	
Amazon Glacier	Cloud Storage Coldline

Database	
RDBMS	
Amazon Relational Database Service, Amazon Aurora	Cloud SQL, Cloud Spanner
Management Services	
Monitoring/Logging	
Amazon CloudWatch Logs	Stackdriver Logging/Monitoring
Machine Learning	
Speech	
Amazon Transcribe	Cloud Speech-to-Text

Google Cloud Console

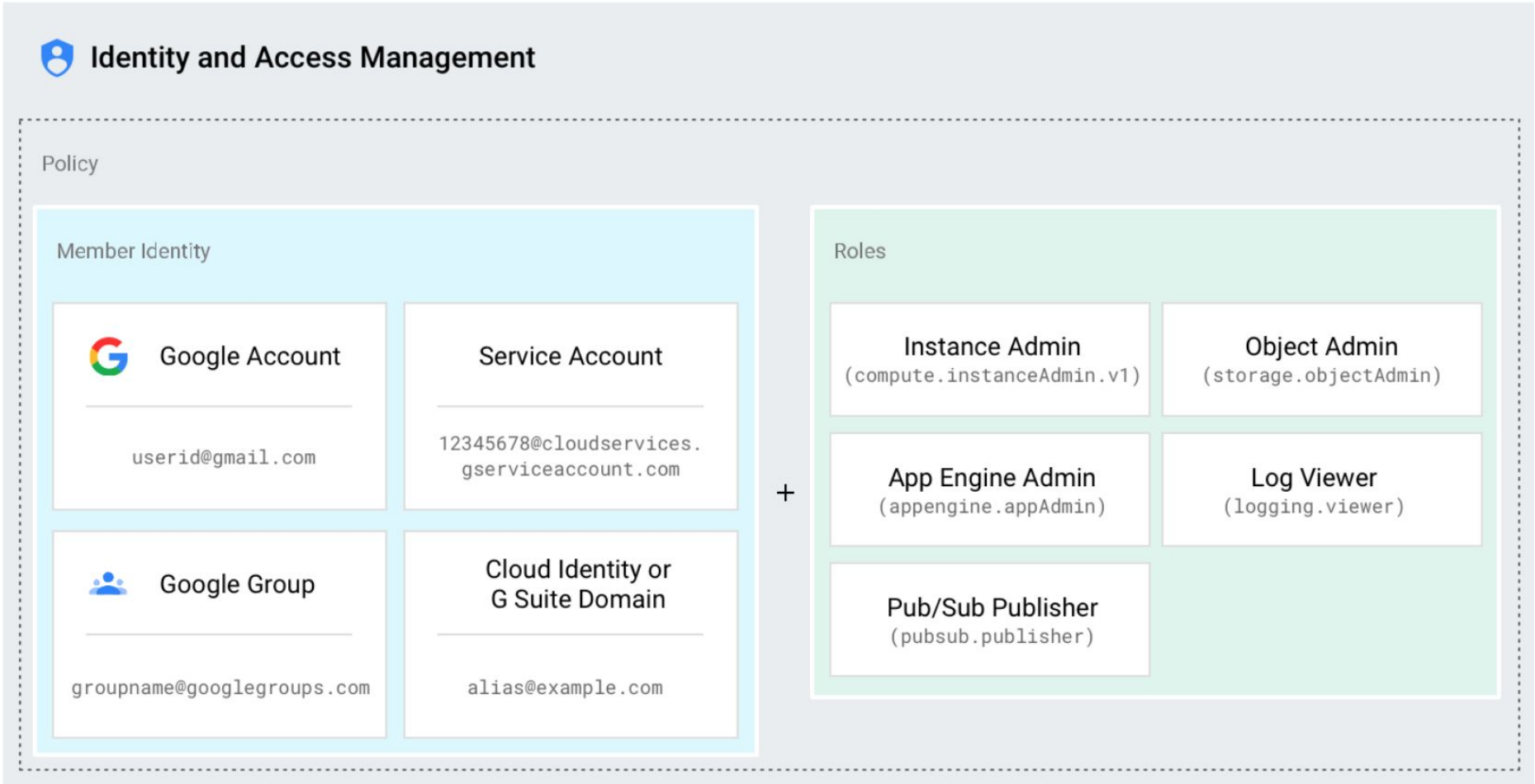
1. A project contains enabled services and at the level of each service, you set the location
2. GCP projects are identified by:
   a. the ID,
   b. the name
   c. the number.
3. Regions and Zones: https://cloud.google.com/compute/docs/regions-zones/
4. The services menu is customized based on services API being disabled or enabled



Security and IAM

- IAM: (for people) to add members and control permissions per project
- Service accounts: (for services) to add service accounts and control add keys
- Quotas: to control spending and avoid billing surprises per service
  (Not available for free accounts)

https://cloud.google.com/iam/docs/overview


Identity and Access Management
Policy
Member Identity
Google Account
userid@gmail.com
Service Account
12345678@cloudservices.gserviceaccount.com
Google Group
groupname@googlegroups.com
Cloud Identity or G Suite Domain
alias@example.com
+
Roles
Instance Admin (compute.instanceAdmin.v1)
Object Admin (storage.objectAdmin)
App Engine Admin (appengine.appAdmin)
Log Viewer (logging.viewer)
Pub/Sub Publisher (pubsub.publisher)

Google Compute Engine (GCE)

1. In the GCP Console, go to the VM Instances page.
2. Click Create instance.
3. In the Boot disk section, click Change to begin configuring your boot disk.
4. In the OS images tab, choose Debian 9.
5. Click Select.
6. In the Firewall section, select Allow HTTP traffic.
7. Click Manage disks, Networking and SSH Keys
    a. User data
    b. Ssh keys
8. Click Create to create the instance.

Google Compute Engine (GCE)

Create instance with user data

```
#!/bin/bash
# Yum Update
 yum update -y
# Install httpd
 yum install httpd -y
# Start and enable service
systemctl start httpd
systemctl enable httpd
sudo groupadd www
sudo usermod -a -G www fadwa
sudo chown -R fadwa:www /var/www
sudo chmod 2775 /var/www
find /var/www -type d -exec chmod 2775 {} \;
find /var/www -type f -exec chmod 0664 {} \;
echo My New Instance on GCP  >> /var/www/html/index.html
```

User data (Startup script)

Re-run user data

You can login to the instance and force rerun userdata script

```
> sudo google_metadata_script_runner --script-type startup
```

Check the output

```
> sudo journalctl -u google-startup-scripts.service
```

User data in a file (will talk about gcloud later)

```
gcloud compute instances create example-instance --metadata-from-file
startup-script=/PATH/TO/YOUR/SCRIPT.sh
```

Instance template and groups

- An instance template is an API resource used create a managed instance group or to create individual VM instances groups with identical configurations.

  https://console.cloud.google.com/compute/instanceTemplates/

- Defines
  - the machine type,
  - boot disk image or container image,
  - zone,
  - labels,
  - other instance properties.
- A managed instance group contains identical instances that you can manage as a single entity.

Google Storage

- Go to https://console.cloud.google.com
- Navigation menu
- Storage
- Create Bucket
  - Name
  - Multiregional
  - US
  - Create
  - Upload files
- To make the file public
  - Select and edit permissions
    - Entity : User
    - Name: allUsers
    - Access: Reader

GCP VPC

- VPC networks, including their associated routes and firewall rules, are global resources. They are *not* associated with any particular region or zone.
- *Subnets* are regional resources. Each subnet defines a range of IP addresses.Traffic to and from instances can be controlled with network firewall rules.
- Resources within a VPC network can communicate with one another using internal (private) IPv4 addresses.
- Instances with internal IP addresses can communicate with Google APIs and services.
- Network administration can be secured using Identity and Access Management (IAM) roles.
- More on VPC https://cloud.google.com/vpc/docs/vpc

Create VPC

- Go to https://console.cloud.google.com
- Navigation menu
- VPC Networks
- Create VPC Network
    - Name :
    - Description
    - Subnets
        - Subnet Creation mode : custom
        - New Subnet
            - Name : pub
            - Region: us-east4
            - IP Address range : 10.10.0.0/16
    - Dynamic routing mode : Regional

Google Database

- Go to https://console.cloud.google.com
- Navigation menu
- SQL
- Create instance
  - MySQL (or Postgresql)
  - Choose second generation
  - Instance ID : e91sql
  - Root password : PASSWORD
  - Location
  - Region
  - Create

Connect to your database

Using MysQL Client

$ apt install mariadb-client OR yum install mariadb-client

$ mysql -u root -h DATABASE-IP

Using gcloud API (will talk about gcloud api later in the slides)

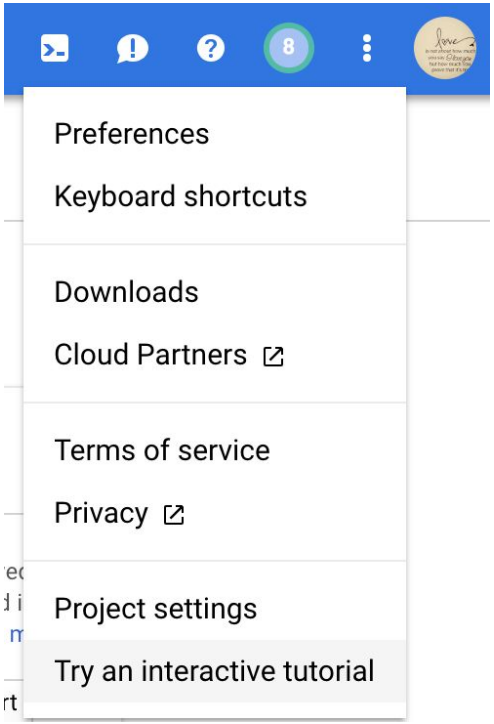
$ gcloud sql connect e91sql --user=root --quiet

Google Function

- Go to https://console.cloud.google.com
- Navigation menu
- Cloud Function
  - Name
  - Memory allocated
  - Trigger
  - Source code
  - Runtime
  - Function to execute
  - Create

Using the tutorial : GCE Kubernetes Example

- Google offers wizard like tutorials to build some services.
- From the console, at the top right corner, click on the three dotted menu and select Interactive Tutorial
- Select Try Kubernetes Engine which will build a Hello World App
- Follow the wizard steps



Command Line Tool

1. Method1: Using the Google Cloud SDK installer: https://cloud.google.com/sdk/docs/downloads-interactive
2. Method2: Installing from versioned archives
   a. Download GCP SDK https://cloud.google.com/sdk/install
   b. Extract and run Install script to add Cloud SDK tools to your path

- On Linux or macOS:
  ./google-cloud-sdk/install.sh
- On Windows:
  .\google-cloud-sdk\install.bat

   c. Initialize the SDK: ./google-cloud-sdk/bin/gcloud init

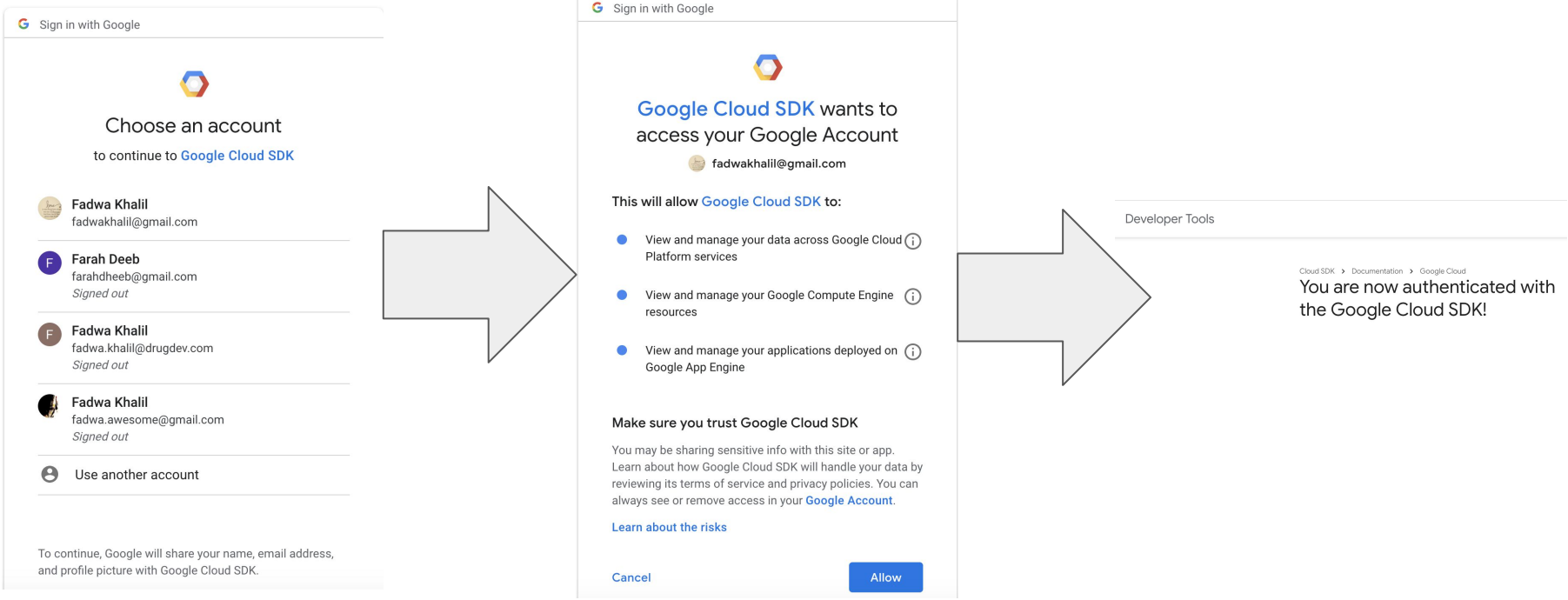
3. Restart Terminal and check version:

```
Fadwas-MacBook-Pro:~ fadwa$ gcloud version
Google Cloud SDK 225.0.0
bq 2.0.37
core 2018.11.09
gsutil 4.34
```

gcloud authentication

From terminal authenticate to gcloud: $ gcloud init

```
Fadwas-MacBook-Pro:~ fadwa$ gcloud init
Welcome! This command will take you through the configuration of gcloud.

Your current configuration has been set to: [default]

You can skip diagnostics next time by using the following flag:
  gcloud init --skip-diagnostics

Network diagnostic detects and fixes local network connection issues.
Checking network connection...done.
Reachability Check passed.
Network diagnostic passed (1/1 checks passed).

You must log in to continue. Would you like to log in (Y/n)?  Y

Your browser has been opened to visit:
```




```
You are logged in as: [fadwakhalil@gmail.com].

Pick cloud project to use:
 [1] e91testing
 [2] Create a new project
Please enter numeric choice or text value (must exactly match list
item):
```

Common Commands

$ gcloud auth login ## To authenticate again

$ gcloud info --show-log ## To check logs

$ gcloud config set project <project-id> ## To set project

$ gcloud init <project-name> ## To set project

Gcloud command help:
$ gcloud --help

gcloud cheat sheet:
https://gist.github.com/pydevops/cffbd3c694d599c6ca18342d3625af97

Google Deployment Manager

From Console: https://console.cloud.google.com -> Navigation menu -> Deployment Manager

Using gcloud :

create a yaml file of resources “for example centos.yaml”

$ gcloud deployment-manager deployments create centos-template --config centos.yaml

$ gcloud deployment-manager deployments describe centos-template

From the Console or the gcloud API check your instances

$ gcloud compute instance list

To delete the template

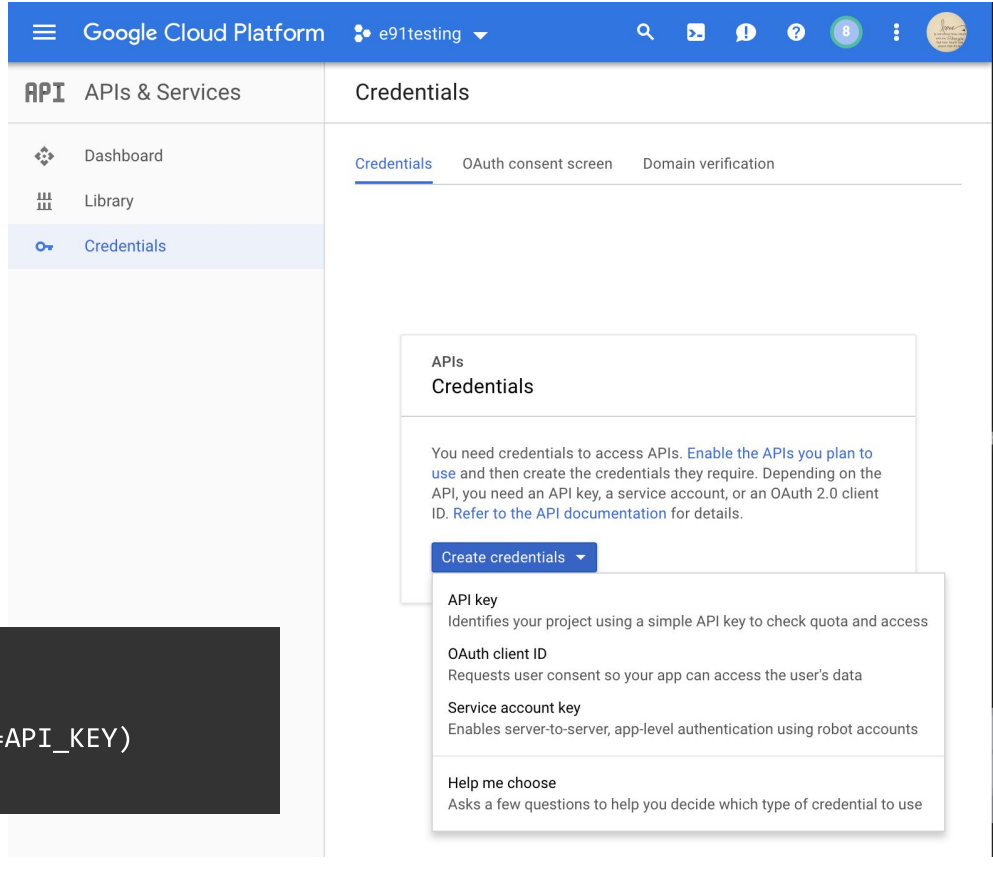
$ gcloud deployment-manager deployments delete centos-template

```
# centos.yaml

# Replace cscie91-222920 with your project ID
# Replace sourceImage with the image of your desire.
# List of Images: https://cloud.google.com/compute/docs/images#os-compute-support
resources:
- type: compute.v1.instance
  name: CentOS-7-VM
  properties:
    # The properties of the resource depend on the type of resource.
    zone: us-central1-f
    machineType:
https://www.googleapis.com/compute/v1/projects/cscie91-222920/zones/us-central1-f/machineTypes/f1-micro
    disks:
    - deviceName: boot
      type: PERSISTENT
      boot: true
      autoDelete: true
      initializeParams:
        sourceImage: https://www.googleapis.com/compute/v1/projects/centos-cloud/global/images/family/centos-7
    networkInterfaces:
    - network: https://www.googleapis.com/compute/v1/projects/cscie91-222920/global/networks/default
      # Access Config required to give the instance a public IP address
      accessConfigs:
      - name: External NAT
        type: ONE_TO_ONE_NAT
```

Google cloud python

- Install the google-api-python-client library.
  $ pip install --upgrade google-api-python-client
- To keep projects secure, the first step is to authorize access.
- Create API Key
- Reference the API Key in python code
- git clone https://github.com/GoogleCloudPlatform/python-docs-samples
- Use your generated key to authenticate

```
from apiclient import discovery
API_KEY = # copied from project credentials page
SERVICE = discovery.build(API, VERSION, developerKey=API_KEY)
```



Example

Objectives and steps:

- Use client secret key,
- List managed virtual machine instances in your project,
- Run gcloud auth login from command line
- Create new Client ID from Console > API > Credentials
- verify your results.

```
#!/usr/bin/env python
from googleapiclient import discovery

def get_authentication():
      API_KEY = # copied from project credentials page
      SERVICE = discovery.build('compute', 'v1', credentials=API_KEY)
    return service

def get_managed_instance_groups(project='my-project'):
    service = get_authentication()
    instance_groups_manager = service.instanceGroupManagers()
    aggregated_list_request = instance_groups_manager.aggregatedList(project=project)
    response = aggregated_list_request.execute()
    return response['items']

groups = get_managed_instance_groups()

assets = groups['zones/us-east4-a']['instanceGroupManagers']

for asset in assets:
    print(asset)
```