**Challenge #1**

A 3 tier environment is a common setup. Use a tool of your choosing/familiarity create these resources. Please remember we will not be judging on the outcome but more on the approach, style and reproducibility. A 3 tier environment is a common setup. Use a tool of your choosing/familiarity create these resources. Please remember we will not be judging on the outcome but more on the approach, style and reproducibility.

terraform {

required\_providers {

google = {

source = "hashicorp/google"

}

}

}

provider "google" {

region = "us-central1"

zone = "us-central1-c"

}

resource "google\_compute\_instance" "terraform" {

count = 3

name = "prod\_VM${count.index + 1}"

machine\_type = "n1-standard-2"

boot\_disk {

initialize\_params {

image = "debian-cloud/debian-11"

}

}

network\_interface {

network = "default"

access\_config {

}

}

}

**Challenge #2**

Summary

We need to write code that will query the meta data of an instance within AWS and provide a json formatted output. The choice of language and implementation is up to you.

Pip install ec2-metadata

>>> from ec2\_metadata import ec2\_metadata

>>> print(ec2\_metadata.region) us-east-1

**Challenge #3**

We have a nested object, we would like a function that you pass in the object and a key and get back the value. How this is implemented is up to you.

Example Inputs

object = {“a”:{“b”:{“c”:”d”}}}

key = a/b/c

object = {“x”:{“y”:{“z”:”a”}}}

key = x/y/z

value = a

Ans :- def findValue(obj,key):

keys = key.split (' / ');

value = " ";

obj= value;

print(value)

if \_name\_=="\_main\_":

obj = {"a" :{"b":{"c":"d"}}};

key = 'a/b/c';

findValue(obj,key);