

Using Terraform to deploy GCP resources

Requirement: Deploy VM instances along with firewall rules, to servers in US and EU using Terraform

Environment: GCP Console, GCP Shell, Terraform files and Terraform Plugin

Key results: Login using your GCP subscription, and follow the below

Confirm Terraform is installed using below cmd

```
terraform --version
```

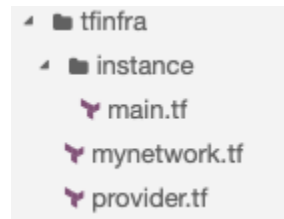
You need Terraform v0.12.2 and later. I had v1.0 installed in my GCP Environment

```
mkdir tfinfra
```

From Cloud Shell, cd to tfinfra

Open Code Editor

Using the editor, create new 3 TF files as shown in figure below



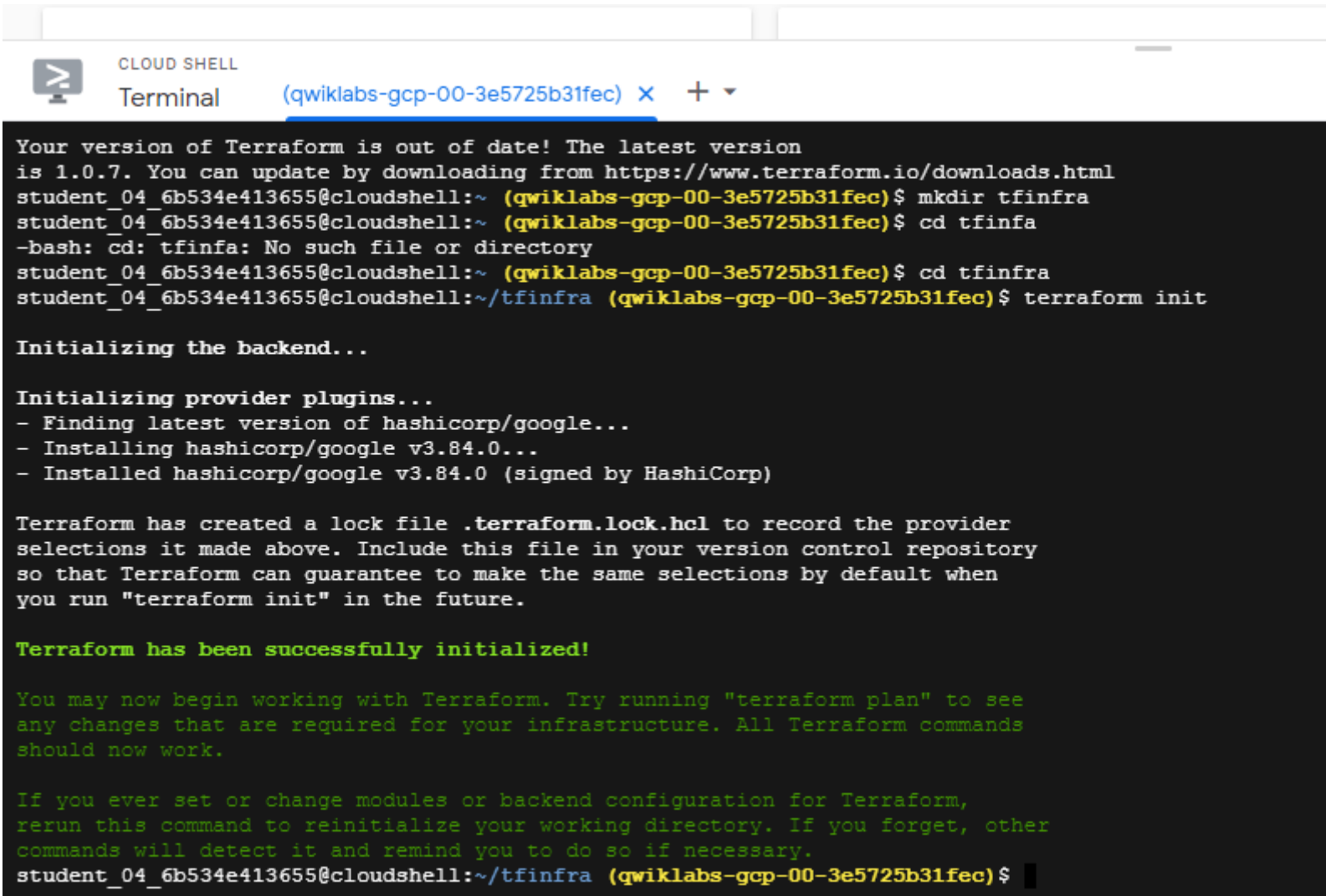
Copy the file contents as below (see that your Editor does not show YAML syntax errors):

PROVIDER.TF

```
provider "google" {}
```

After the Provider.tf is saved, return to shell and initialize Terraform

```
cd tfinfra
terraform init
```



The screenshot shows a Cloud Shell terminal window with the title "Terminal (qwiklabs-gcp-00-3e5725b31fec)". The terminal output shows the user navigating to the 'tfinfra' directory and running 'terraform init'. Terraform displays a message about its version (1.0.7) and provides a link to update. It then proceeds to initialize the backend and provider plugins, specifically finding and installing the HashiCorp/google provider v3.84.0. A lock file '.terraform.lock.hcl' is created. The process concludes with a green message: "Terraform has been successfully initialized!". A final green message encourages the user to run 'terraform plan' to see required changes. The terminal ends with the prompt 'student_04_6b534e413655@cloudshell:~/tfinfra (qwiklabs-gcp-00-3e5725b31fec)\$'.

```
Cloud Shell
Terminal (qwiklabs-gcp-00-3e5725b31fec) x + v

Your version of Terraform is out of date! The latest version
is 1.0.7. You can update by downloading from https://www.terraform.io/downloads.html
student_04_6b534e413655@cloudshell:~ (qwiklabs-gcp-00-3e5725b31fec)$ mkdir tfinfra
student_04_6b534e413655@cloudshell:~ (qwiklabs-gcp-00-3e5725b31fec)$ cd tfinfra
-bash: cd: tfinfra: No such file or directory
student_04_6b534e413655@cloudshell:~ (qwiklabs-gcp-00-3e5725b31fec)$ cd tfinfra
student_04_6b534e413655@cloudshell:~/tfinfra (qwiklabs-gcp-00-3e5725b31fec)$ terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/google...
- Installing hashicorp/google v3.84.0...
- Installed hashicorp/google v3.84.0 (signed by HashiCorp)

Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
student_04_6b534e413655@cloudshell:~/tfinfra (qwiklabs-gcp-00-3e5725b31fec)$
```

MYNETWORK.TF

Create the mynetwork network

```
resource "google_compute_network" "mynetwork" {  
  
  name          = "mynetwork"  
  
  auto_create_subnetworks = true  
  
}
```

Add a firewall rule to allow HTTP, SSH, RDP, and ICMP traffic on mynetwork

```
resource "google_compute_firewall" "mynetwork-allow-http-ssh-rdp-icmp" {  
  
  name = "mynetwork-allow-http-ssh-rdp-icmp"  
  
  network = google_compute_network.mynetwork.self_link  
  
  allow {  
  
    protocol = "tcp"  
  
    ports = ["22", "80", "3389"]  
  
  }  
  
  allow {  
  
    protocol = "icmp"  
  
  }  
  
}
```

MAIN.TF

```
variable "instance_name" {}

variable "instance_zone" {}

variable "instance_type" {
  default = "n1-standard-1"
}

variable "instance_network" {}

resource "google_compute_instance" "vm_instance" {
  name      = "${var.instance_name}"
  zone      = "${var.instance_zone}"
  machine_type = "${var.instance_type}"
  boot_disk {
    initialize_params {
      image = "debian-cloud/debian-9"
    }
  }
}

network_interface {
  network = "${var.instance_network}"
}
```

```
access_config {  
    # Allocate a one-to-one NAT IP to the instance  
}  
  
}  
  
}  
  
//
```

Rewrite the Terraform configuration files to a canonical format

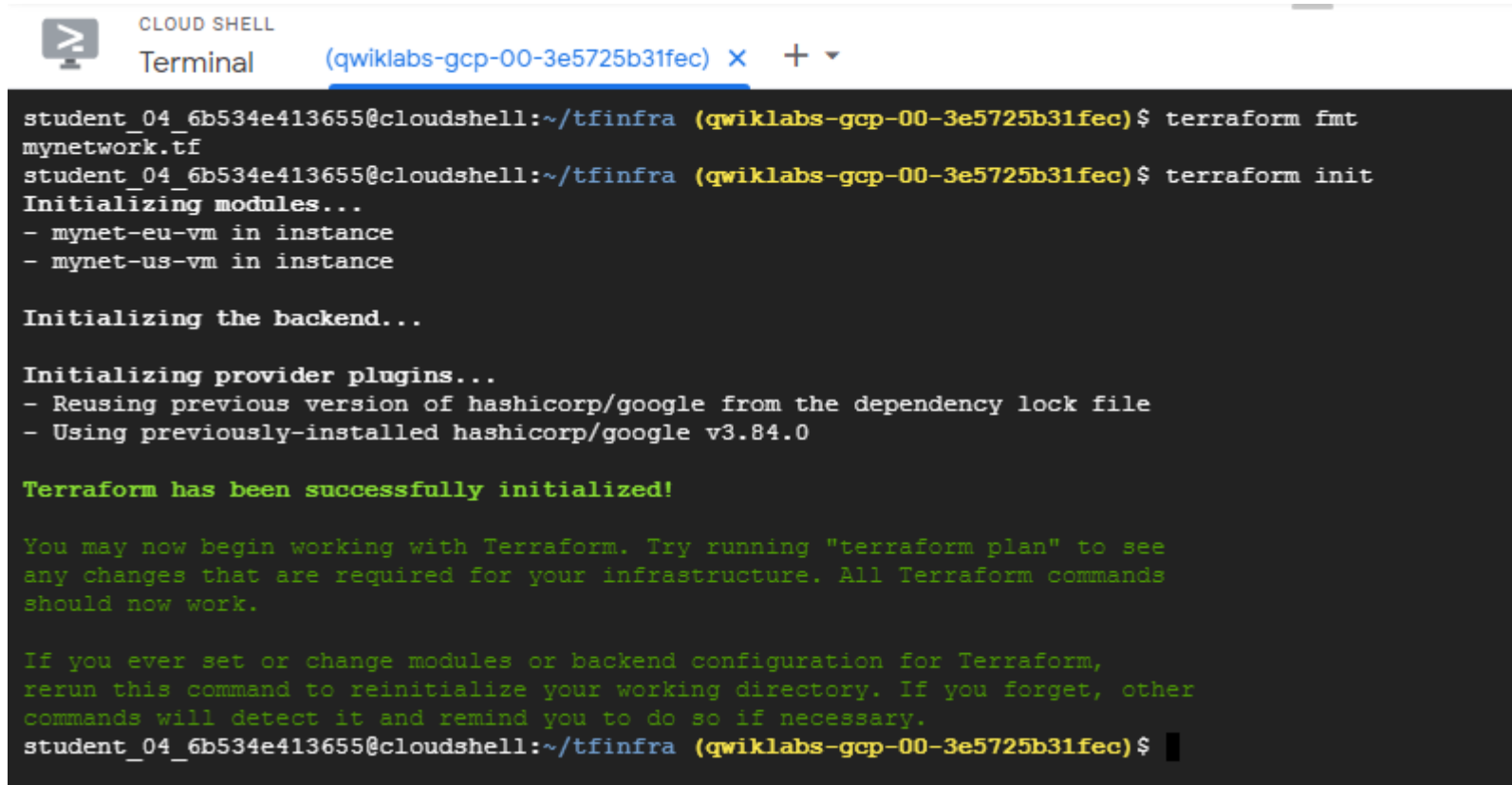
From Shell run cmd:

terraform fmt

To initialize Terraform, run the following command

terraform init

Expected Results:



A screenshot of a Cloud Shell terminal window. The title bar at the top shows 'CLOUD SHELL' and 'Terminal' with a tab labeled '(qwiklabs-gcp-00-3e5725b31fec)'. The terminal content shows the execution of 'terraform fmt mynetwork.tf' followed by 'terraform init'. The output indicates that modules are being initialized, including 'mynet-eu-vm' and 'mynet-us-vm', and that the backend is being initialized. It also shows provider plugins being initialized, reusing previous versions of hashicorp/google. A green message states 'Terraform has been successfully initialized!' and provides instructions on how to begin working with Terraform, including running 'terraform plan'.

```
student_04_6b534e413655@cloudshell:~/tfinfra (qwiklabs-gcp-00-3e5725b31fec)$ terraform fmt mynetwork.tf
student_04_6b534e413655@cloudshell:~/tfinfra (qwiklabs-gcp-00-3e5725b31fec)$ terraform init
Initializing modules...
- mynet-eu-vm in instance
- mynet-us-vm in instance

Initializing the backend...

Initializing provider plugins...
- Reusing previous version of hashicorp/google from the dependency lock file
- Using previously-installed hashicorp/google v3.84.0

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
student_04_6b534e413655@cloudshell:~/tfinfra (qwiklabs-gcp-00-3e5725b31fec)$
```

To create an execution plan, run the following command

terraform plan

Expected results:



CLOUD SHELL

Terminal

(qwiklabs-gcp-00-3e5725b31fec) X + ▾

```
+ access_config {
  + nat_ip      = (known after apply)
  + network_tier = (known after apply)
}

+ reservation_affinity {
  + type = (known after apply)

  + specific_reservation {
    + key      = (known after apply)
    + values = (known after apply)
  }
}

+ scheduling {
  + automatic_restart = (known after apply)
  + min_node_cpus     = (known after apply)
  + on_host_maintenance = (known after apply)
  + preemptible       = (known after apply)

  + node_affinities {
    + key      = (known after apply)
    + operator = (known after apply)
    + values   = (known after apply)
  }
}
}
```

Plan: 4 to add, 0 to change, 0 to destroy.

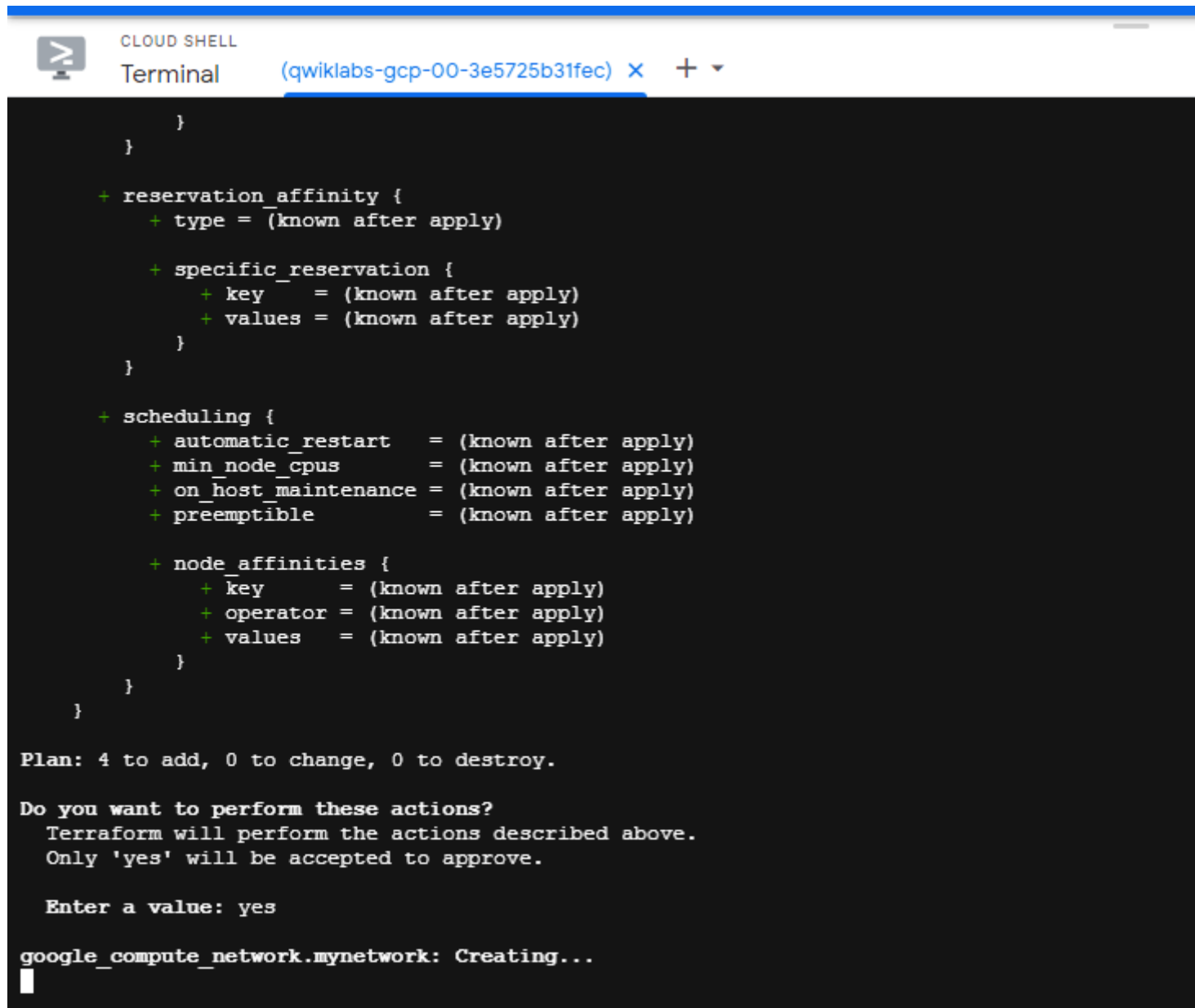
To apply the desired changes, run the following command

terraform apply

To confirm the planned actions, type

yes

Expected results:



A screenshot of a Cloud Shell terminal window. The title bar at the top says "CLOUD SHELL" and "Terminal (qwiklabs-gcp-00-3e5725b31fec)". The terminal output shows a Terraform plan for a Google Cloud project. It lists changes for a reservation affinity, scheduling, and node affinities, all marked as "(known after apply)". The plan summary shows 4 additions, 0 changes, and 0 destructions. A prompt asks for confirmation to perform the actions, and the user has entered "yes". The output then shows the start of creating a network resource.

```
    }  
  }  
  
+ reservation_affinity {  
  + type = (known after apply)  
  
  + specific_reservation {  
    + key    = (known after apply)  
    + values = (known after apply)  
  }  
}  
  
+ scheduling {  
  + automatic_restart    = (known after apply)  
  + min_node_cpus        = (known after apply)  
  + on_host_maintenance = (known after apply)  
  + preemptible          = (known after apply)  
  
  + node_affinities {  
    + key      = (known after apply)  
    + operator = (known after apply)  
    + values   = (known after apply)  
  }  
}  
}  
  
Plan: 4 to add, 0 to change, 0 to destroy.  
  
Do you want to perform these actions?  
Terraform will perform the actions described above.  
Only 'yes' will be accepted to approve.  
  
Enter a value: yes  
  
google_compute_network.mynetwork: Creating...
```


After all resources are created, please see if you get this result,

```
Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

google_compute_network.mynetwork: Creating...
google_compute_network.mynetwork: Still creating... [10s elapsed]
google_compute_network.mynetwork: Still creating... [20s elapsed]
google_compute_network.mynetwork: Still creating... [30s elapsed]
google_compute_network.mynetwork: Still creating... [40s elapsed]
google_compute_network.mynetwork: Creation complete after 42s [id=projects/qwiklabs-gcp-00-3e5725b31fec/global/networks/mynetwork]
module.mynet-eu-vm.google_compute_instance.vm_instance: Creating...
module.mynet-us-vm.google_compute_instance.vm_instance: Creating...
google_compute_firewall.mynetwork-allow-http-ssh-rdp-icmp: Creating...
module.mynet-eu-vm.google_compute_instance.vm_instance: Still creating... [10s elapsed]
module.mynet-us-vm.google_compute_instance.vm_instance: Still creating... [10s elapsed]
google_compute_firewall.mynetwork-allow-http-ssh-rdp-icmp: Still creating... [10s elapsed]
google_compute_firewall.mynetwork-allow-http-ssh-rdp-icmp: Creation complete after 11s [id=projects/qwiklabs-gcp-00-3e5725b31fec/global/firewalls/mynet-allow-http-ssh-rdp-icmp]
module.mynet-eu-vm.google_compute_instance.vm_instance: Creation complete after 18s [id=projects/qwiklabs-gcp-00-3e5725b31fec/zones/europe-west1-a/instances/mynet-eu-vm]
module.mynet-us-vm.google_compute_instance.vm_instance: Creation complete after 19s [id=projects/qwiklabs-gcp-00-3e5725b31fec/zones/us-central1-a/instances/mynet-us-vm]

Apply complete! Resources: 4 added, 0 changed, 0 destroyed.
student_04_6b534e413655@cloudshell:~/tfinfra (qwiklabs-gcp-00-3e5725b31fec)$
```

Verify your deployment, see if your Console has the resources created, my examples

Google Cloud Platform

qwiklabs-gcp-00-3e5725b31fec

Search products and resources

VPC network

VPC networks

External IP addresses

Bring your own IP

Firewall

Routes

VPC network peering

Shared VPC

VPC networks

+ CREATE VPC NETWORK

REFRESH

australia-southeast2	default				10.192.0.0/20	10.192.0.1
▼ mynetwork		28	1460	Auto		1
us-central1	mynetwork				10.128.0.0/20	10.128.0.1
europa-west1	mynetwork				10.132.0.0/20	10.132.0.1
us-west1	mynetwork				10.138.0.0/20	10.138.0.1
asia-east1	mynetwork				10.140.0.0/20	10.140.0.1
us-east1	mynetwork				10.142.0.0/20	10.142.0.1
asia-northeast1	mynetwork				10.146.0.0/20	10.146.0.1
asia-southeast1	mynetwork				10.148.0.0/20	10.148.0.1

CLOUD SHELL

Google Cloud Platform

qwiklabs-gcp-00-3e5725b31fec

Search products and resources

VPC network

VPC networks

External IP addresses

Bring your own IP

Firewall

Routes

VPC network peering

Shared VPC

Firewall

+ CREATE FIREWALL RULE

REFRESH

CONFIGURE LOGS

DELETE

<input type="checkbox"/>	default-allow-internal	Ingress	Apply to all	IP ranges: 10	tcp:0-65535 udp:0-65535 icmp	Allow	65534	default	Off
<input type="checkbox"/>	default-allow-rdp	Ingress	Apply to all	IP ranges: 0.0	tcp:3389	Allow	65534	default	Off
<input type="checkbox"/>	default-allow-ssh	Ingress	Apply to all	IP ranges: 0.0	tcp:22	Allow	65534	default	Off
<input type="checkbox"/>	mynetwork-allow-http-ssh-rdp-icmp	Ingress	Apply to all	IP ranges: 0.0	tcp:22, 80, 3389 icmp	Allow	1000	mynetwork	Off

Google Cloud Platform

qwiklabs-gcp-00-3e5725b31fec

Search products and resources

Compute Engine

Virtual machines

VM instances

Instance templates

Sole-tenant nodes

Marketplace

Release Notes

VM instances

CREATE INSTANCE

IMPORT VM

OPERATIONS

HELP ASSISTANT

SHOW INFO PANEL

INSTANCES

INSTANCE SCHEDULE

Filter

Enter property name or value

	Status	Name ↑	Zone	Recommendations	In use by	Internal IP	External IP	Connect
<input type="checkbox"/>	✓	mynet-eu-vm	eu-west1-d			10.132.0.2 (nic0)	35.187.114.246	SSH
<input type="checkbox"/>	✓	mynet-us-vm	us-central1-a			10.128.0.2 (nic0)	35.224.133.16	SSH

```
ssh.cloud.google.com/projects/qwiklabs-gcp-00-3e5725b31fec/zones/us-central1-a/instances/mynet-us-vm?authuser=2&hl=en_US&p
Connected, host fingerprint: ssh-rsa 0 4C:D5:99:5B:5F:A9:C8:C3:7A:31:FB:81:FA:CF
:3F:B8:38:CB:49:E1:E9:0E:19:C2:FA:BC:52:B0:86:30:D9:77
Linux mynet-us-vm 4.9.0-16-amd64 #1 SMP Debian 4.9.272-2 (2021-07-19) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Creating directory '/home/student-04-6b534e413655'.
student-04-6b534e413655@mynet-us-vm:~$ ping -c 3 10.132.0.2
PING 10.132.0.2 (10.132.0.2) 56(84) bytes of data.
64 bytes from 10.132.0.2: icmp_seq=1 ttl=64 time=100 ms
64 bytes from 10.132.0.2: icmp_seq=2 ttl=64 time=98.7 ms
64 bytes from 10.132.0.2: icmp_seq=3 ttl=64 time=100 ms

--- 10.132.0.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 98.737/99.835/100.738/0.828 ms
student-04-6b534e413655@mynet-us-vm:~$
```

Disclaimer: All the above content is part of the Google Cloud Platform and used here for study and demonstration purpose only. Prepared and executed by Bhadale IT Pvt Ltd in GCP

END