



Project #1

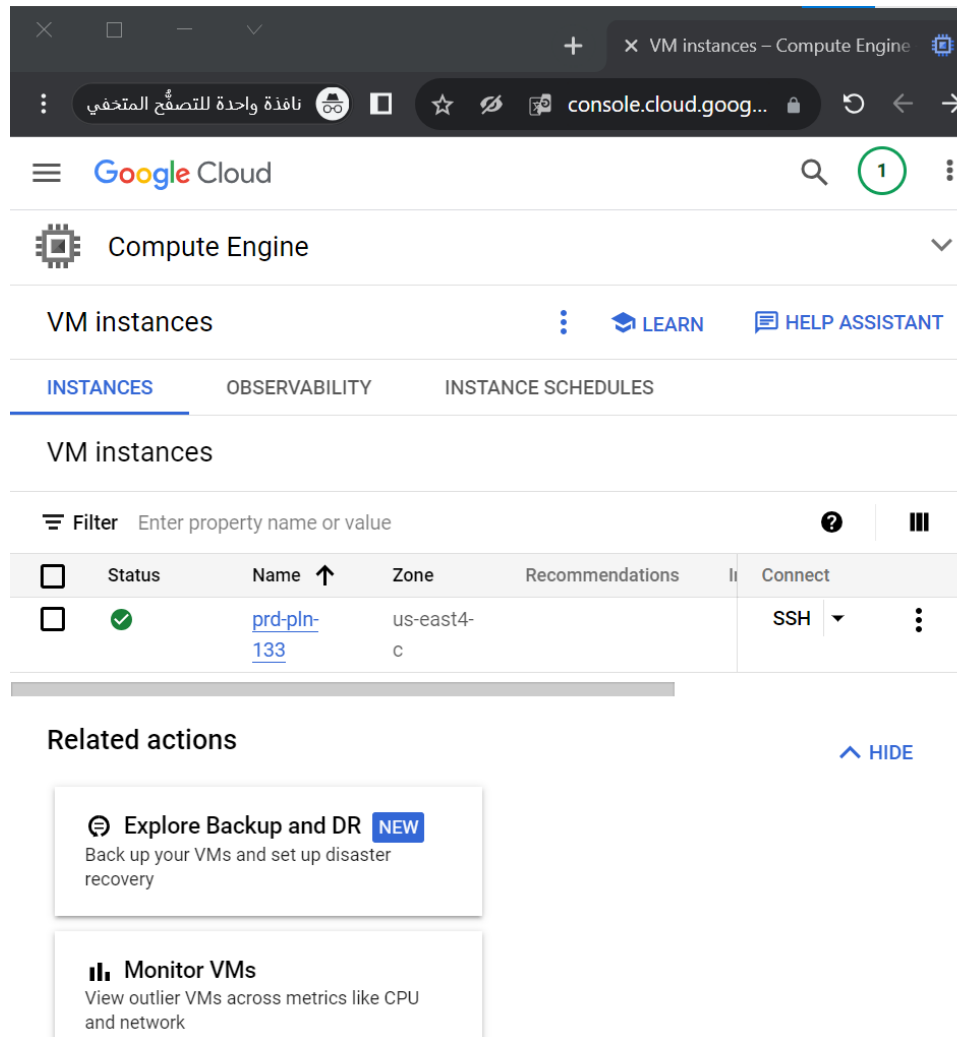
Apache or NGINX Web Server on Google Cloud

Nsreen Hujjatullah



Task 1. Create a Linux VM instance

Create a Linux virtual machine, name it Instance name and specify the zone as Compute zone.



The screenshot shows the Google Cloud Console interface for VM instances. The browser address bar shows 'console.cloud.goog...'. The page title is 'VM instances - Compute Engine'. The main navigation bar includes 'Google Cloud', a search icon, and a notification badge with the number '1'. The 'Compute Engine' section is selected, and the 'VM instances' page is displayed. The 'INSTANCES' tab is active, showing a table of VM instances. The table has columns for Status, Name, Zone, Recommendations, and Connect. One instance is listed with the name 'prd-pln-133' and zone 'us-east4-c'. Below the table, there are 'Related actions' including 'Explore Backup and DR' and 'Monitor VMs'.

Status	Name	Zone	Recommendations	Connect
✓	prd-pln-133	us-east4-c		SSH

Related actions

- Explore Backup and DR** NEW
Back up your VMs and set up disaster recovery
- Monitor VMs**
View outlier VMs across metrics like CPU and network

Task 2. Enable public access to VM instance

- While creating the Linux instance, make sure to apply the appropriate firewall rules so that potential customers can find your new product.

```
ssh.cloud.google.com/v2/ssh/projects/qwiklabs-gcp-04-d9b5580cbd6a/zones/us-east4-...
SSH-in-browser
Setting up apache2-bin (2.4.56-1-deb11u2) ...
Setting up apache2 (2.4.56-1-deb11u2) ...
Enabling module mpm_event.
Enabling module authz_core.
Enabling module authz_host.
Enabling module authn_core.
Enabling module auth_basic.
Enabling module access_compat.
Enabling module authn_file.
Enabling module authz_user.
Enabling module alias.
Enabling module dir.
Enabling module autoindex.
Enabling module env.
Enabling module mime.
Enabling module negotiation.
Enabling module setenvif.
Enabling module filter.
Enabling module deflate.
Enabling module status.
Enabling module reqtimeout.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service →
/lib/systemd/system/apache2.service.
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean
an.service → /lib/systemd/system/apache-htcacheclean.service.
Processing triggers for man-db (2.9.4-2) ...
Processing triggers for libc-bin (2.31-13+deb11u6) ...
student-02-1f24af7497dd6prd-pln-133:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor pr
   Active: active (running) since Sat 2023-07-29 23:24:22 UTC; 2min 4s ago
     Docs: https://httpd.apache.org/docs/2.4/
    Main PID: 1978 (apache2)
       Tasks: 55 (limit: 4691)
      Memory: 11.1M
         CPU: 87ms
    CGroup: /system.slice/apache2.service
            └─1978 /usr/sbin/apache2 -k start
            └─1979 /usr/sbin/apache2 -k start
            └─1981 /usr/sbin/apache2 -k start
Jul 29 23:24:22 prd-pln-133 systemd[1]: Starting The Apache HTTP Server...
Jul 29 23:24:22 prd-pln-133 systemd[1]: Started The Apache HTTP Server.
lines 1-15/15 (END)
```

```
SSH-in-browser
Enabling module mime.
Enabling module negotiation.
Enabling module setenvif.
Enabling module filter.
Enabling module deflate.
Enabling module status.
Enabling module reqtimeout.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service →
/lib/systemd/system/apache2.service.
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean
an.service → /lib/systemd/system/apache-htcacheclean.service.
Processing triggers for man-db (2.9.4-2) ...
Processing triggers for libc-bin (2.31-13+deb11u6) ...
student-02-1f24af7497dd6prd-pln-133:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor pr
   Active: active (running) since Sat 2023-07-29 23:24:22 UTC; 2min 4s ago
     Docs: https://httpd.apache.org/docs/2.4/
    Main PID: 1978 (apache2)
       Tasks: 55 (limit: 4691)
      Memory: 11.1M
         CPU: 87ms
    CGroup: /system.slice/apache2.service
            └─1978 /usr/sbin/apache2 -k start
            └─1979 /usr/sbin/apache2 -k start
            └─1981 /usr/sbin/apache2 -k start
Jul 29 23:24:22 prd-pln-133 systemd[1]: Starting The Apache HTTP Server...
Jul 29 23:24:22 prd-pln-133 systemd[1]: Started The Apache HTTP Server.
lines 1-15/15 (END)
```

Google Cloud
Essential Skills:
Challenge Lab

Task 2. Enable public access to VM instance

0/100

- While creating the Linux instance, make sure to apply the appropriate firewall rules so that potential customers can find your new product.

Click **Check my progress** to verify the objective.

✓

Create a Compute Engine instance, add necessary firewall rules.

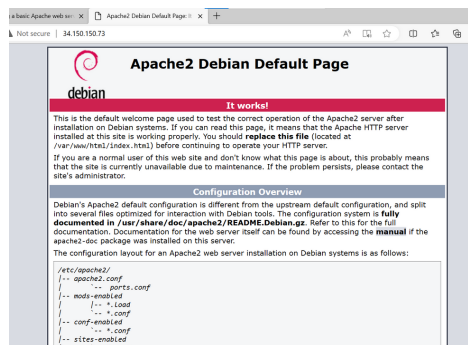
Check my progress

Assessment Completed!

Task 3. Running a basic Apache Web Server

A virtual machine instance on Compute Engine can be controlled like any standard Linux server.

- Deploy a simple Apache web server (a placeholder for the new product site) to learn the basics of running a server on a virtual machine instance.

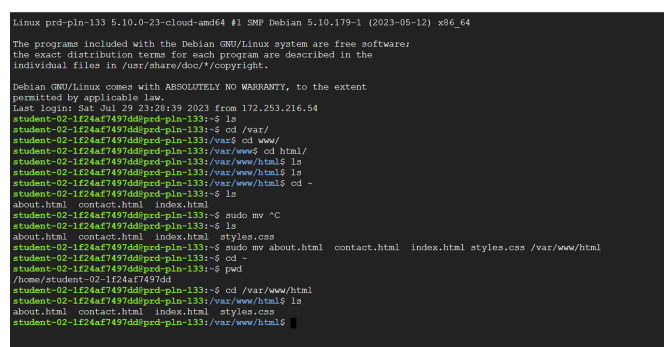
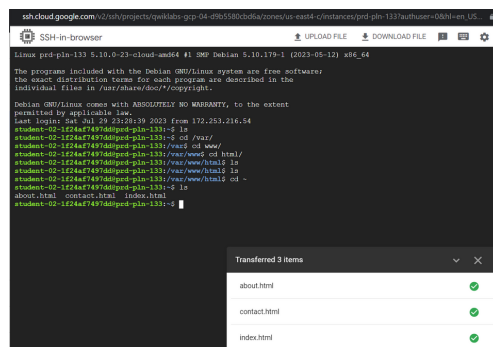
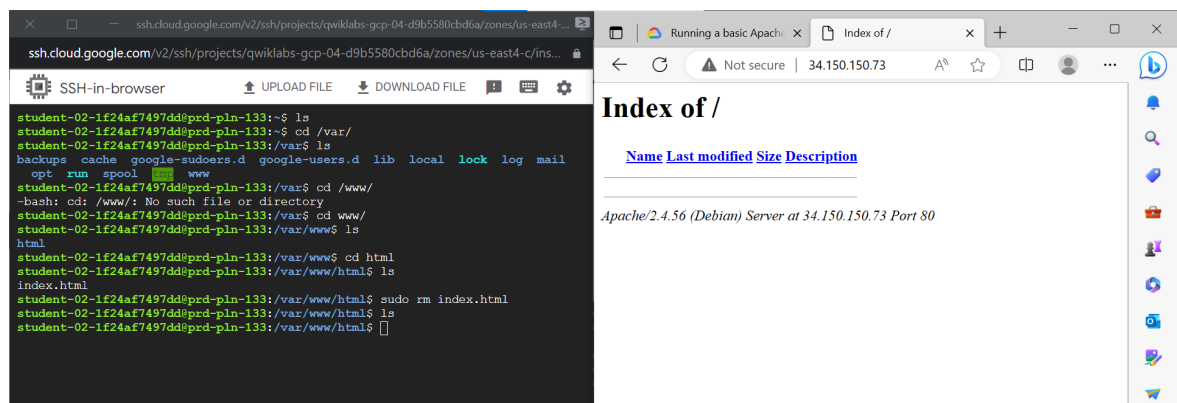
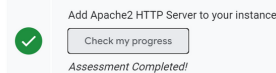


Task 3. Running a basic Apache Web Server

A virtual machine instance on Compute Engine can be controlled like any standard Linux server.

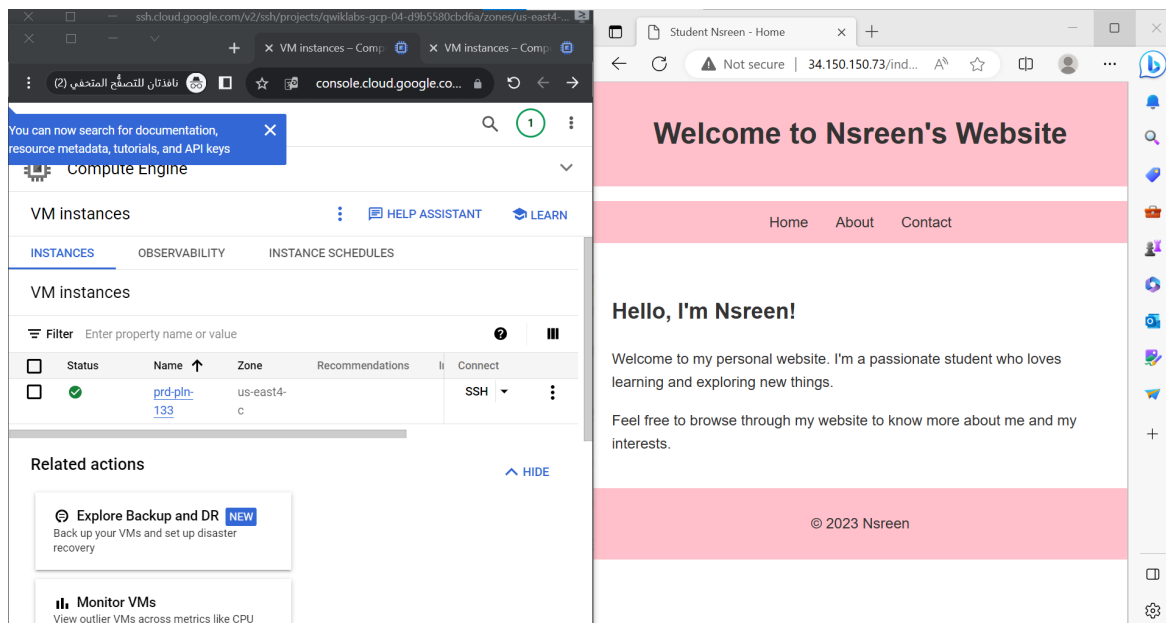
- Deploy a simple Apache web server (a placeholder for the new product site) to learn the basics of running a server on a virtual machine instance.

Click **Check my progress** to verify the objective.



Task 4. Test your server

- Test that your instance is serving traffic on its external IP.

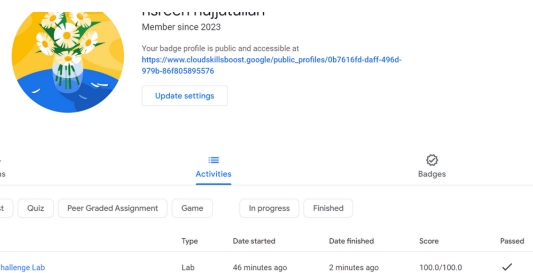
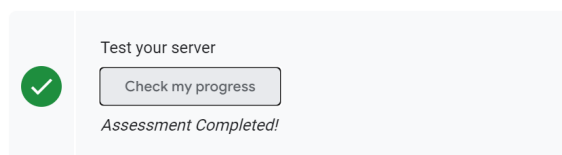


Task 4. Test your server

- Test that your instance is serving traffic on its external IP.

You should see the "Hello World!" page (a placeholder for the new product site).

Click **Check my progress** to verify the objective.





<http://34.16.0.20/index.html>

