

# Unit 2: Homework

## Purpose

The purpose of this exercise is to create and control your own virtual machine using your GCP account.

## Preparation

See [Unit 2 – Notes](#) for how to get started with your own GCP account. You only need to do this once. You do not need to provide a credit card number and you do not need to start a free trial. If you are asked for either of these, STOP, you are going down a wrong path and should start over.

## Assignment

To document the steps below, create a Word document that contains screenshots and brief explanations of what you did. This serves as a notebook for you and a summary for the instructor to grade.

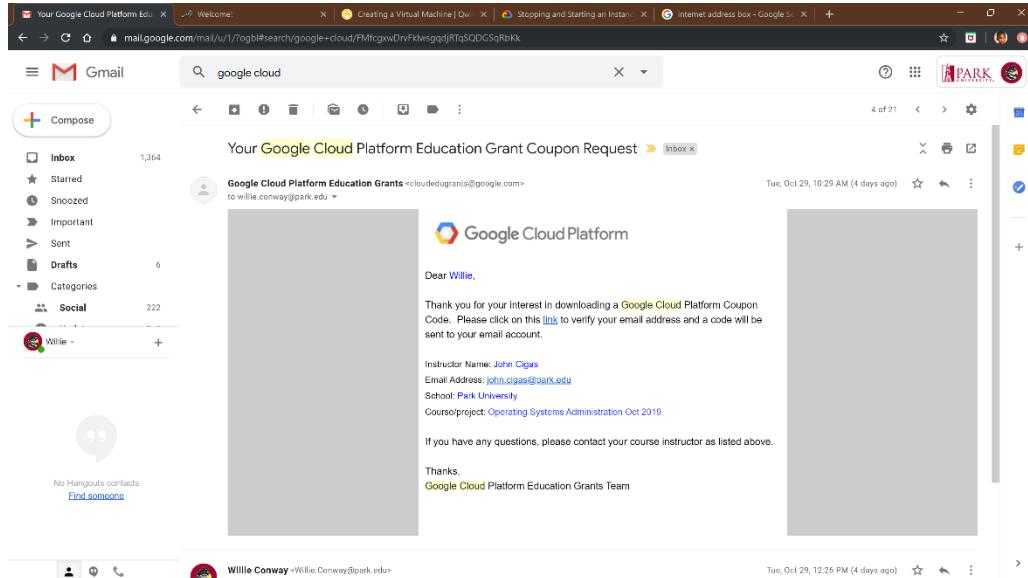
(Summary to screen shots are in red) (keywords are in blue)

1. Set up your GCP account and give your instructor access to your project.

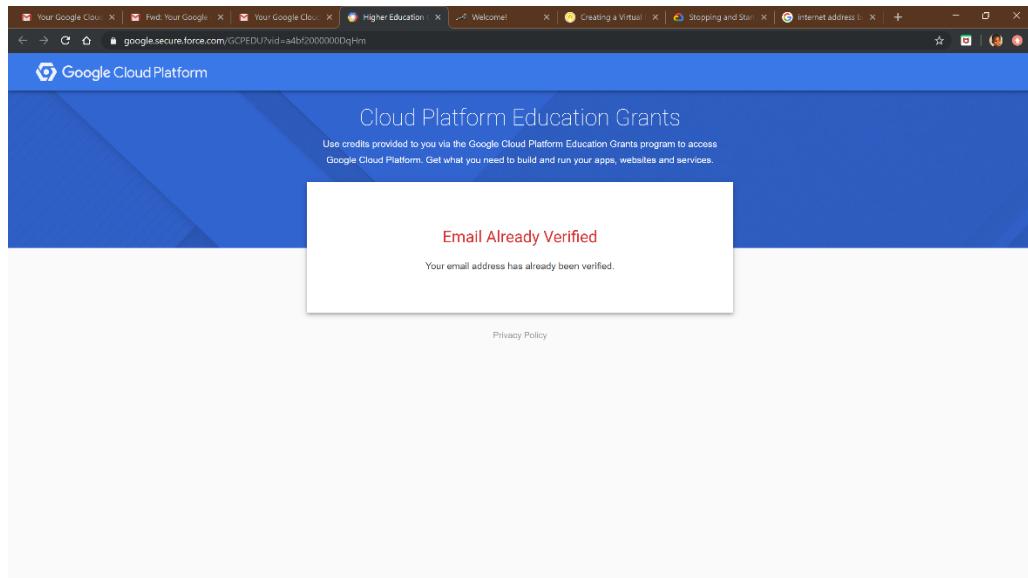
The screenshot shows a web browser window with multiple tabs open. The active tab is titled "Creating a Virtual Machine | Quell" and displays the Google Cloud Platform Education Grants application form. The form is titled "Cloud Platform Education Grants" and asks for personal information: First Name (Willie), Last Name (Conway), and School Email (willie.conway@park.edu). A "Submit" button is at the bottom. Below the form, a small note states: "By clicking "Submit" below, you agree that we may share the following information with your educational institution and course instructor ([john.page@park.edu](#)): (1) personal information that you provide to us on this form and (2) information regarding your use of the coupon and Google Cloud Platform products." A "Privacy Policy" link is also present.

*(After clicking on the link to the coupon, this page then pops up, and I begin entering all of the following information asked in the fields. The First Name, Last Name, School Email, then clicked Submit.)*

# Unit 2: Homework

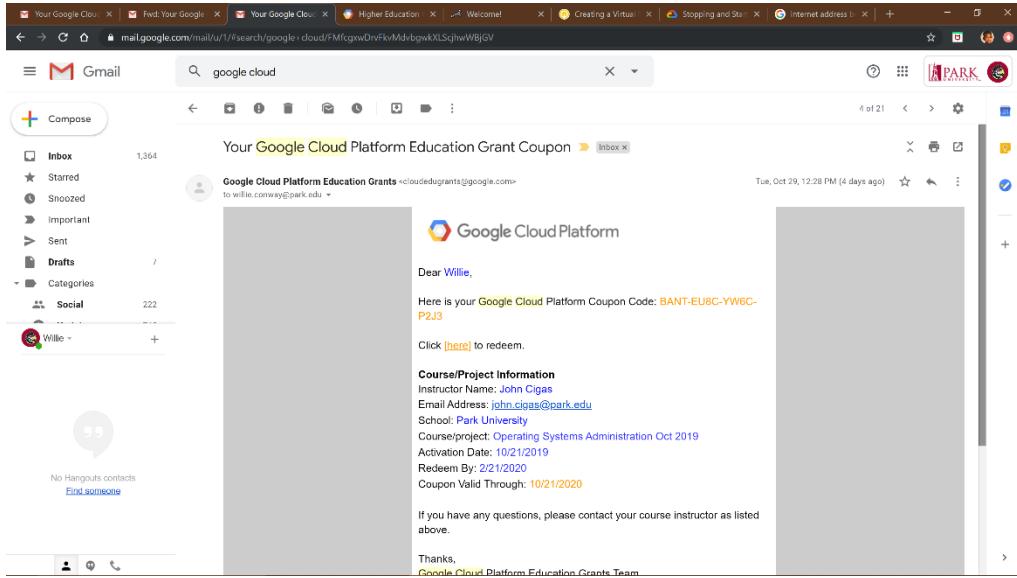


*(After checking my email to , I forward the message to my regular email account I'll be using for this assignment, which is [conway198799@gmail.com](mailto:conway198799@gmail.com). I completely logged out of my Park email account [willie.conway@spark.edu](mailto:willie.conway@spark.edu) . I then logged into [conway198799@gmail.com](mailto:conway198799@gmail.com) to take the next step in verifying my email. So, I clicked on the link to verify.)*

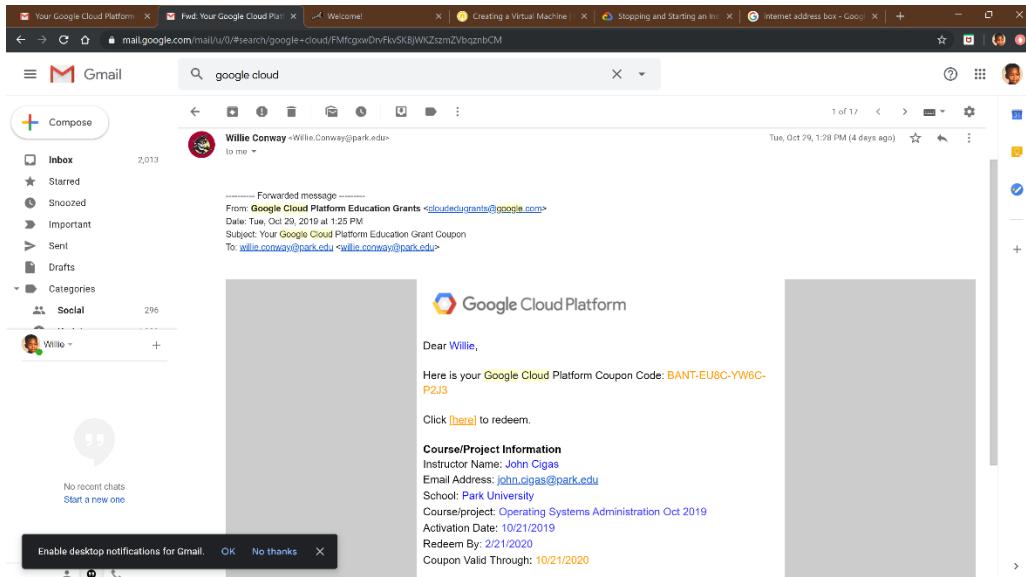


*(After finding the email I clicked verified. Since I done everything prior it already says email already verified.)*

# Unit 2: Homework



*(Since the email has been verified I need to log completely out of my [conway198799@gmail.com](mailto:conway198799@gmail.com) account and log back into my [willie.conway@park.edu](mailto:willie.conway@park.edu) account to check for the email with the coupon code. In the screenshot above you can see the coupon email with the link to redeem. I need to forward this email to the [conway198799@gmail.com](mailto:conway198799@gmail.com) account to take advantage of the coupon.)*



*(After forwarding the coupon code message from my [willie.conway@park.edu](mailto:willie.conway@park.edu) account I completely logged out and logged into my [conway198799@gmail.com](mailto:conway198799@gmail.com) account open the message and open the link to redeem my coupon code.)*

# Unit 2: Homework

Please enter the coupon code provided to you via the Google Cloud Platform Education Grants program to receive credit for Google Cloud Platform. Get what you need to build and run your apps, websites and services.

**Coupon code**

The coupon code you have entered is invalid.

**Terms of Service**

Country of residence

**Google Cloud Platform education grants credits terms and conditions**

By clicking "Accept and continue" below, you, on behalf of yourself and the organization you represent ("You") agree to these terms and conditions:

The credit is valid for Google Cloud Platform products and is subject to Google's acceptance of the applicable Google Cloud Platform terms and conditions and any other applicable terms of service. The credit is non-transferable and may not be sold or bartered. Unused credit expires on the date indicated on the most recent version of the promotion code. The credit may be issued in increments as You use the credit over the period of time during which the credit is valid. Once used, where applicable by law, You represent that you are accepting the promotional credit on behalf of your educational institution and the credit will apply to any purchases made by your educational entity and not your individual. You represent, on behalf of your educational entity, (i) You are authorized to accept this credit; (ii) this credit is consistent with all applicable laws and regulations, including relevant privacy rules and laws; and (iii) the provision of credits will not negatively impact Google's current or future ability to do business.

You agree that we may share the following information with your educational institution and course instructor: (1) personal information that you provide to us during the coupon redemption process and (2) information regarding your use of the coupon and Google Cloud Platform products.

**Accept and continue** **Clear**

(This page pops up. As you can see in the screenshot that it says the coupon code has already been entered, meaning that I already redeemed it. But from here you would simply just click Accept and continue.)

You have 24 projects remaining in your quota. Request an increase or delete projects. [Learn more](#)

**MANAGE QUOTAS**

**Project name \***

Project ID: stoked-flame-257816. It cannot be changed later. [EDIT](#)

**Billing account \***

Any charges for this project will be billed to the account you select here.

**Location \***

[BROWSE](#)

Parent organization or folder

**CREATE** **CANCEL**

(After Accept and continue. I then entered the name of my project and selected the billing account I'll be using the coupon for. Then clicked Create.)

# Unit 2: Homework

The screenshot shows the Google Cloud Platform dashboard for the project "Willie Conway CS369 F2 2019". A red box highlights the "Project info" section, which displays the project name, ID, and number. Below it, a red box highlights the "ADD PEOPLE TO THIS PROJECT" button. The dashboard also includes sections for APIs, Google Cloud Platform status, Error Reporting, and News.

(As you can see this a screenshot of my completed creation of my GCP account. You can also see the project name above where it says dashboard.)

The screenshot shows the Google Cloud Platform IAM & admin page for the same project. A red box highlights the "+ ADD" button. The main area displays the "Permissions for project 'Willie Conway CS369 F2 2019'" table, which lists two users: "conway198799@gmail.com" (Owner) and "cs373JohnCigas@gmail.com" (Editor). The table has columns for Type, Member, Name, Role, and Over granted.

Type	Member	Name	Role	Over granted
User	conway198799@gmail.com	Willie Conway	Owner	On
User	cs373JohnCigas@gmail.com		Editor	On

(In order to give access privileges to the instructor, I had to click on the hamburger menu, then select the IAM & admin, then select the +ADD button. As seen here I've already handled the next task below.)

# Unit 2: Homework

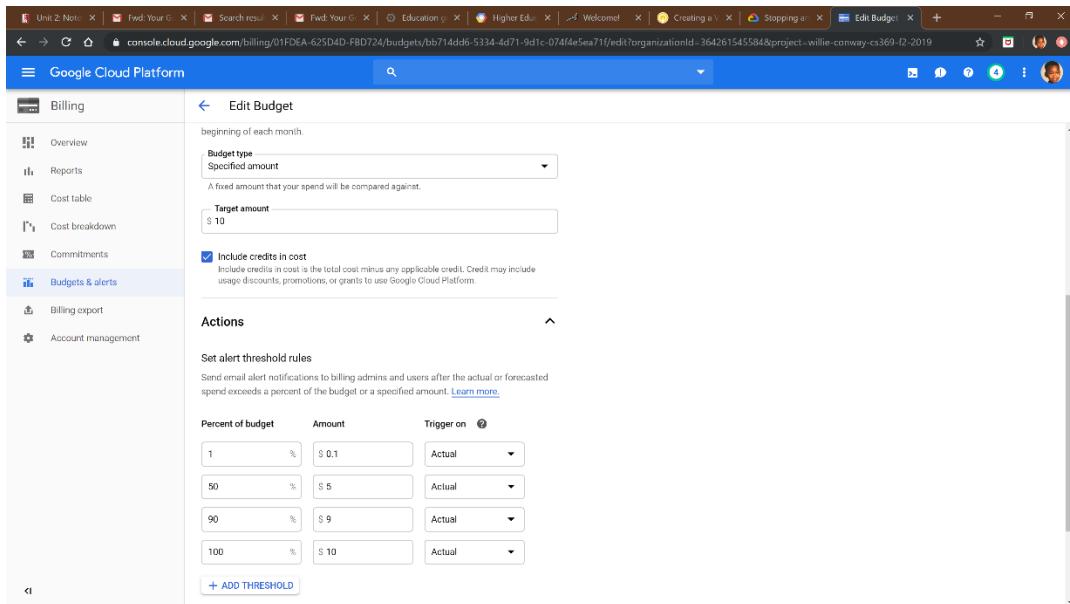
The screenshot shows a browser window with several tabs open, including "Textbooks", "linuxfun.pdf", "how to ren...", "IAM - IAM", and "Linux / Unix". The active tab is "console.cloud.google.com/iam-admin/iam?folder=true&organizationId=true&proj...". The main content area is titled "Add members to 'Willie Conway CS369 F2 2019'". It says "Add members, roles to 'Willie Conway CS369 F2 2019' project". Below that, it says "Enter one or more members below. Then select a role for these members to grant them access to your resources. Multiple roles allowed. [Learn more](#)". A "New members" input field contains "cs373johnigas@gmail.com". A dropdown menu labeled "Role" is set to "Editor". A red oval highlights this "Role" dropdown. Below it is a note: "Edit access to all resources." and a "+ ADD ANOTHER ROLE" button. At the bottom are "SAVE" and "CANCEL" buttons.

(Next, enter the instructor's Gmail account, with role of Editor, then save.)

The screenshot shows a browser window with multiple tabs at the top, including "Unit 2: Note...", "Fwd: Your G...", "Search result...", "Fwd: Your G...", "Education...", "Higher Edu...", "Welcome!", "Creating a...", "Stopping an...", and "Edit Budget". The active tab is "Edit Budget" under the "Billing" section of the Google Cloud Platform navigation menu. The left sidebar shows "Overview", "Reports", "Cost table", "Cost breakdown", "Commitments", "Budgets & alerts" (which is selected), "Billing export", and "Account management". The main content area is titled "CS 369 Budget". It has fields for "Name" (set to "CS 369 Budget") and "Scope". Under "Scope", "Projects" is set to "All projects (1)" and "Products" is set to "All products (179)". Below that is the "Amount" section, which says "Set a monthly budget amount. Budgets begin on the first of the month, and reset at the beginning of each month." It shows "Budget type" as "Specified amount" and "Target amount" as "\$ 10". There is also a checked checkbox for "Include credits in cost".

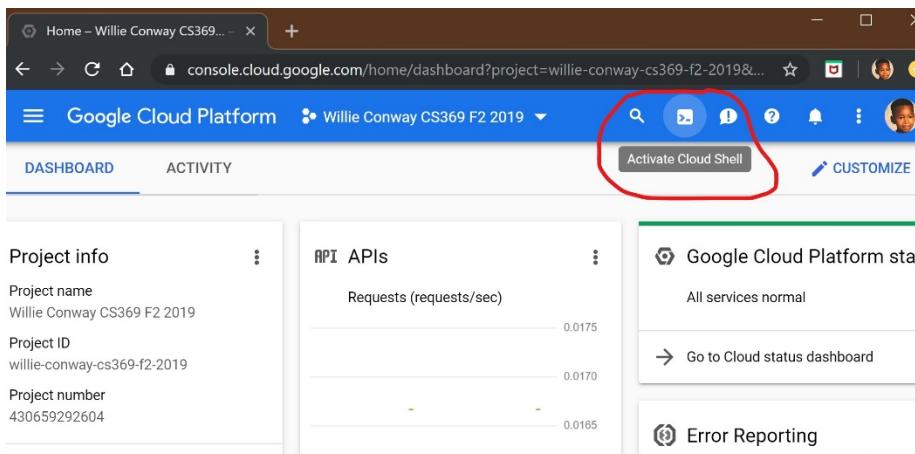
(Next I had to go back to the Navigation menu then select Billing > Budgets and alerts, then click Create Budget button. Form here I filled in the following fields. Picking the budget name CS369 Budget and setting specified amount to \$10.)

# Unit 2: Homework



(Then scrolled down to the Actions area, Under Set Threshold Rules, click Add Item button  
Add 1% item and Save. Then I was all set up to complete the assignment.)

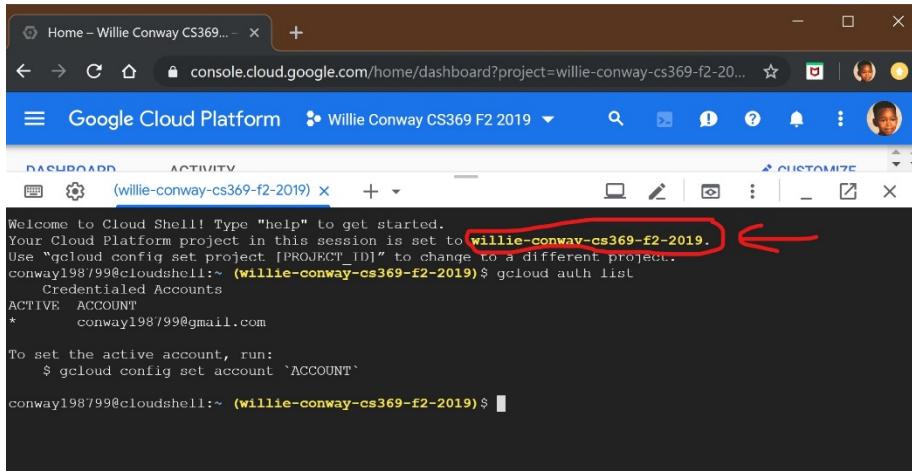
2. Create a virtual machine as you did in the previous Qwiklab exercise. Choose whatever name you like. The only other difference should be that you use a Machine type of “micro (1 shared VCPU)” Using a micro instance will minimize the amount of credits you use, and you will not notice any difference in performance for the work done in this class.



(Referring to last week Qwiklabs exercise, the Google Cloud Shell is a virtual machine that is loaded with development tools that offers a persistent 5GB home directory and runs on the Google Cloud. Google Cloud Shell provides command-line access to your GCP resources. In GCP

# Unit 2: Homework

(console, on the top right toolbar, circled is the Active Cloud Shell button. Click on it, this is the first process to creating a virtual machine.)



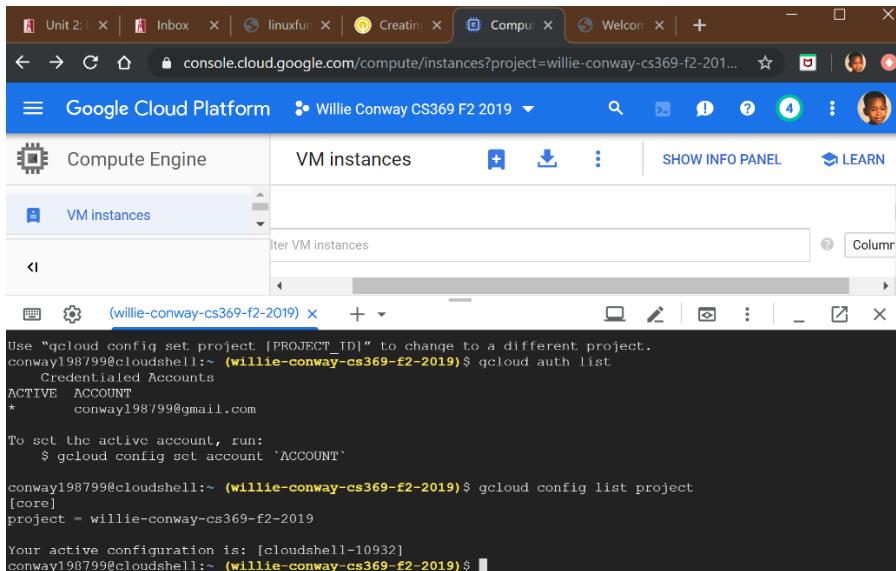
The screenshot shows a browser window for the Google Cloud Platform Cloud Shell. The URL is `console.cloud.google.com/home/dashboard?project=willie-conway-cs369-f2-2019`. The title bar says "Home - Willie Conway CS369...". The main area is a terminal window. The terminal output shows:

```
Welcome to Cloud Shell! Type "help" to get started.  
Your Cloud Platform project in this session is set to willie-conway-cs369-f2-2019.  
Use "gcloud config set project [PROJECT_ID]" to change to a different project.  
conway198799@cloudshell:~ (willie-conway-cs369-f2-2019)$ gcloud auth list  
  Credentialed Accounts  
    ACTIVE  ACCOUNT  
*      conway198799@gmail.com  
  
To set the active account, run:  
  $ gcloud config set account `ACCOUNT`  
  
conway198799@cloudshell:~ (willie-conway-cs369-f2-2019)$
```

A red arrow points to the word "willie-conway-cs369-f2-2019" in the terminal output.

(It takes a few moments to provision and connect to the environment after opening the Active Cloud Shell. When you get connected, you are already authenticated, and the project is set to your PROJECT\_ID which my is willie-conway-cs369-f2-2019.)

(When using the Active Cloud Shell, gcloud is a command-line tool for Google Cloud Platform. It comes pre-installed on Cloud Shell and supports tab-completion. I listed the active account name with the command gcloud auth list. As shown in the screen shot above is information of the authorized user of the account. The project ID could also be listed with the command gcloud config list project. This is shown below in a screenshot.)

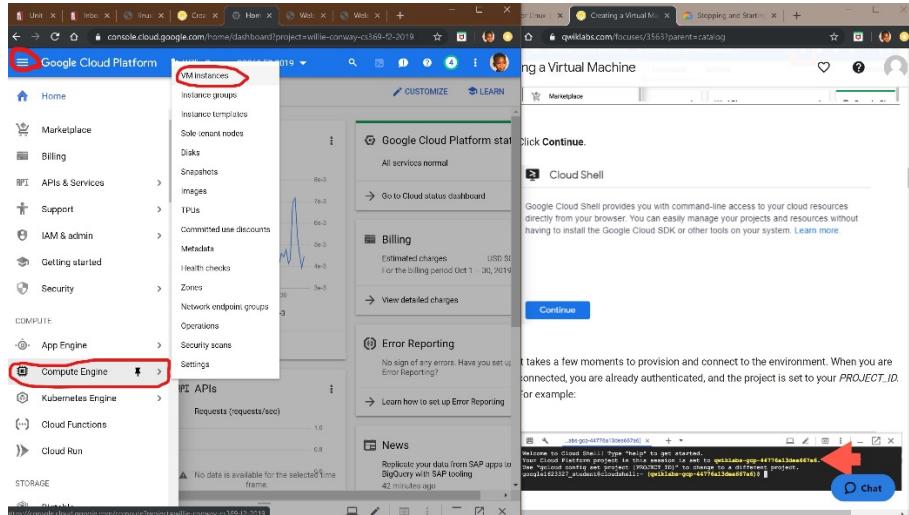


The screenshot shows a browser window for the Google Cloud Platform Compute Engine VM instances page. The URL is `console.cloud.google.com/compute/instances?project=willie-conway-cs369-f2-2019`. The title bar says "Unit 2: |". The main area shows a table of VM instances. The terminal at the bottom shows:

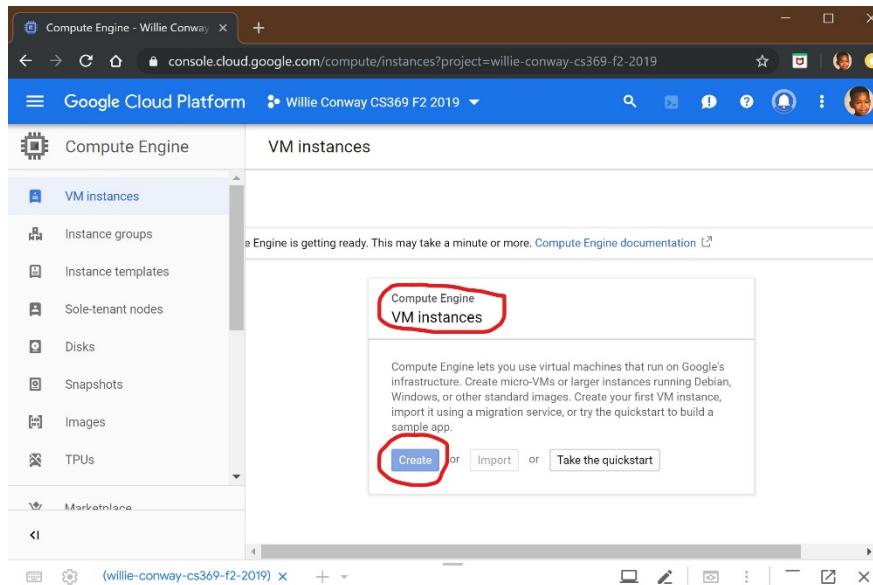
```
Use "gcloud config set project [PROJECT_ID]" to change to a different project.  
conway198799@cloudshell:~ (willie-conway-cs369-f2-2019)$ gcloud auth list  
  Credentialed Accounts  
    ACTIVE  ACCOUNT  
*      conway198799@gmail.com  
  
To set the active account, run:  
  $ gcloud config set account `ACCOUNT`  
  
conway198799@cloudshell:~ (willie-conway-cs369-f2-2019)$ gcloud config list project  
[core]  
project = willie-conway-cs369-f2-2019  
  
Your active configuration is: [cloudshell-10932]  
conway198799@cloudshell:~ (willie-conway-cs369-f2-2019)$
```

(The command gcloud config list project displays the project ID willie-conway-cs369-f2-2019)

# Unit 2: Homework

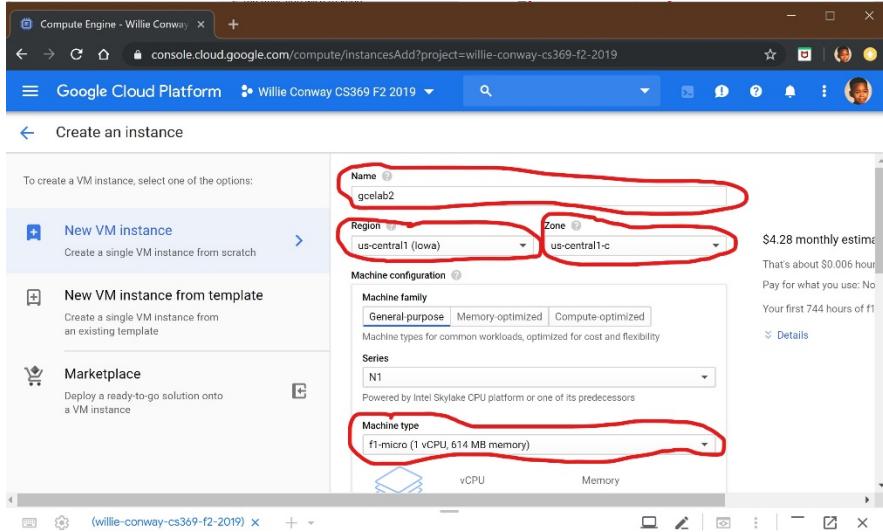


*(Now the next step to creating a Virtual Machine is to create a new instance from the Cloud Console. In this section, we learn how to create a new pre-defined machine types with Google Compute Engine from the Cloud Console. In the GCP Console, on the top left of the screen, select the Navigation menu > Compute Engine > VM Instances. All indications are circled in red. This may take a minute to initialize for the first time.)*



*(After initiating clicking Compute Engine > VM instances, you should be able see the screen above. From here, to create a new instance, I clicked Create.)*

# Unit 2: Homework



(Now, you should see this display page where you will be met with parameters you can configure when creating a new instance. For this project I stuck to the normal guidelines of the first Qwiklabs Assignment. The only change that is configured is the machine type, which for this assignment we must utilize the "f1-micro (1 vCPU, 614 MB memory)." The following parameters are below.)

**Name** – gcelab2

**Region** – us-central1 (Iowa)

**Zone** – us-central1-c

**Machine Type** – f1-micro (1 vCPU, 614 MB memory)

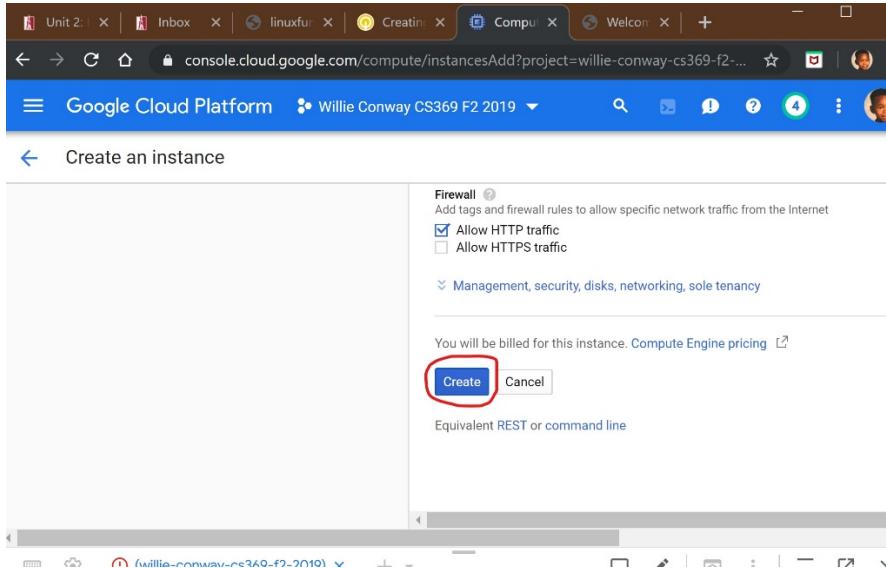
**Boot Disk** – New 10 GB standard persistent disk

OS Image: Debian GNU/Linux 9 (stretch)

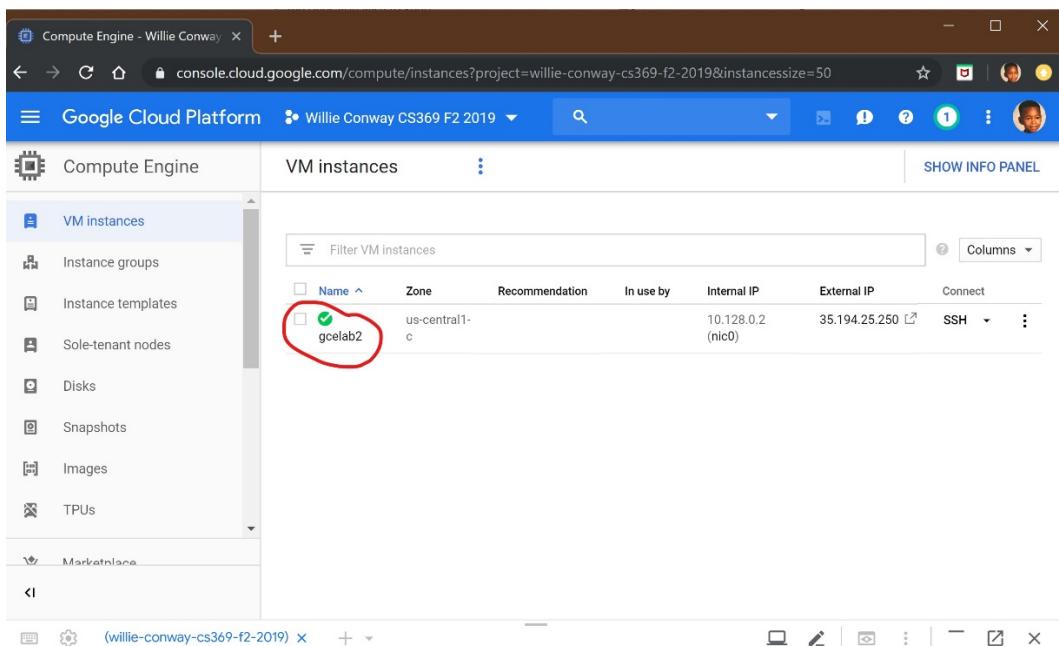
**Firewall** – Check Allow HTTP traffic

(this option gives you access to the webserver that you'll install later. This also will automatically create firewall rule to allow HTTP traffic on port 80.)

# Unit 2: Homework



*(After choosing all the following selections to the parameters, I clicked Create. As shown in the screenshot above, circled in red. Wait for it to finish – it shouldn't take more than a minute.)*



*(Once finished, you should see the new virtual machine in the VM instances page. Circled in red is the Name of my VM instance I just created called gcelab2.)*

3. Connect to your instance using the SSH button, and take a screen shot showing the date and your Gmail name as the prompt:

# Unit 2: Homework

The screenshot shows the Google Cloud Platform Compute Engine interface. On the left, a sidebar lists options like VM instances, Instance groups, Instance templates, Sole-tenant nodes, Disks, Snapshots, Images, and TPUs. The main area displays a table of VM instances. One row is selected, showing details for 'gcelab2'. The 'Connect' column contains an 'SSH' button, which is highlighted with a red box.

Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
gcelab2	us-central1-c			10.128.0.2 (nic0)	35.194.25.250	SSH

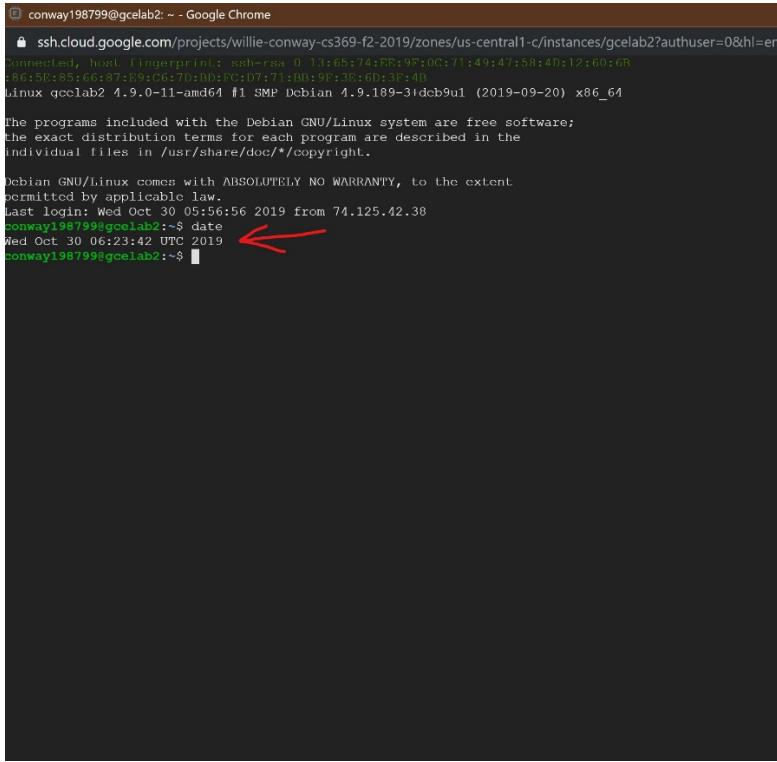
*(To SSH into the virtual machine, I clicked on the SSH on the right-hand side. This launches an SSH client directly from the browser. The SSH is circled in red.)*

```
conway@198799:gcelab2:~ - Google Chrome
ssh.cloud.google.com/projects/willie-conway/cs369-f2-2019/zones/us-central1-c/instances/gcelab2?authuser=0&hl=en_US&projectNum...
connected, host fingerprint: ssh-rsa 0 13:65:74:1e:9f:0c:71:49:47:50:40:12:60:68
0:65:6:0:5:0:0:71:9:CG:7:0:BD:8C:1:D:7:1:B:0:9F:0:5:6:0:3E:4B
Linux gcelab2 4.9.0-11-amd64 #1 SMP Debian 4.9.189-3+deb9u1 (2019-09-20) 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.
conway@198799:gcelab2:~
```

*(After connecting the instance using the SSH button, I was able to see the Gmail name as the prompt along with the name of your VM instance. This is circled in red in the screenshot above.)*

# Unit 2: Homework



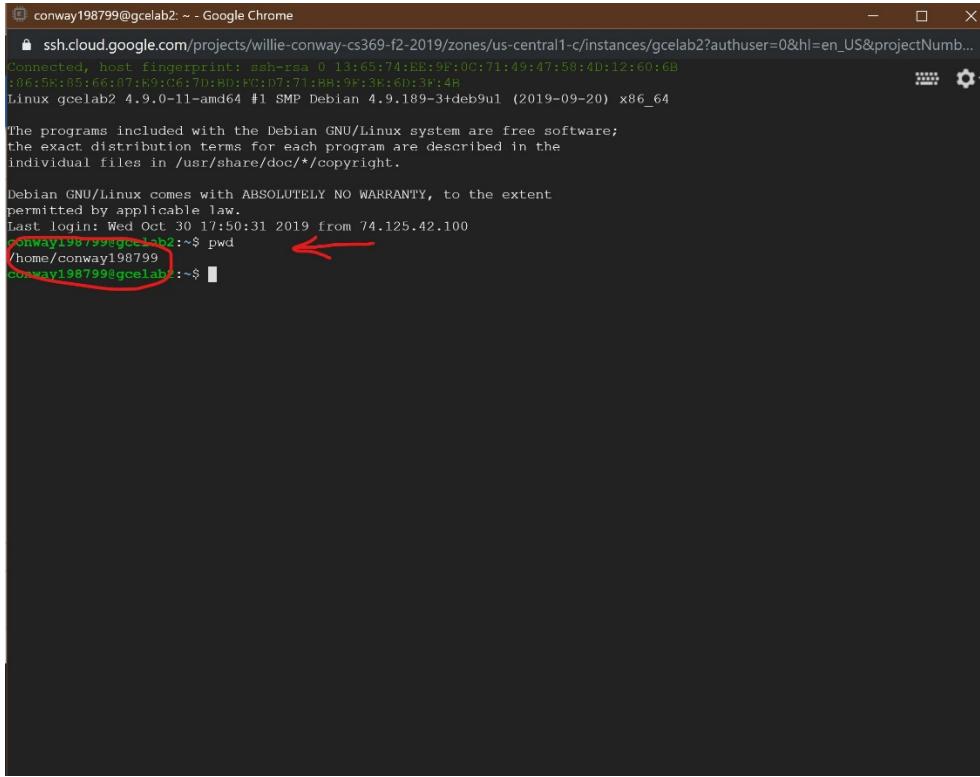
A screenshot of a terminal window titled "conway198799@gcelab2: ~ - Google Chrome". The window shows a shell session on a Debian system. The user runs the "date" command, which outputs "Wed Oct 30 06:23:42 UTC 2019". A red arrow points to the output of the command.

```
conway198799@gcelab2: ~ - Google Chrome
ssh.cloud.google.com/projects/willie-conway-cs369-f2-2019/zones/us-central1-c/instances/gcelab2?authuser=0&hl=en
connected, host fingerprint: ssh-rsa 0 13:65:74:FF:9F:0C:71:49:47:58:40:12:60:6B
:86:5E:85:66:87:E9:C6:7D:BB:FC:D7:71:BB:9F:30:6D:3F:4B
Linux gcelab2 4.9.0-11-amd64 #1 SMP Debian 4.9.189-3+deb9u1 (2019-09-20) 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.
Last login: Wed Oct 30 05:56:56 2019 from 74.125.42.38
conway198799@gcelab2:~$ date
Wed Oct 30 06:23:42 UTC 2019 ←
conway198799@gcelab2:~$
```

(To issue the *date* command in Linux, you must type in the word *date*, then press ENTER. This will then display the output of todays date, including the day of the week, month, date, UTC (Coordinated Universal Time), and year. In the above screenshot you can see that the current date is *Wed Oct 30 06:23:42 UTC 2019*.)

# Unit 2: Homework



A screenshot of a terminal window titled "conway198799@gcelab2: ~ - Google Chrome". The window shows the following text:

```
ssh.cloud.google.com/projects/willie-conway-cs369-f2-2019/zones/us-central1-c/instances/gcelab2?authuser=0&hl=en_US&projectNum...
Connected, host fingerprint: ssh-rsa 0 13:65:74:EE:9F:0C:71:49:47:58:4D:12:60:6B
:06:5E:05:66:07:E9:C6:7D:BD:FC:D7:71:BB:9F:3E:6D:3F:4B
Linux gcelab2 4.9.0-11-amd64 #1 SMP Debian 4.9.189-3+deb9u1 (2019-09-20) 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.

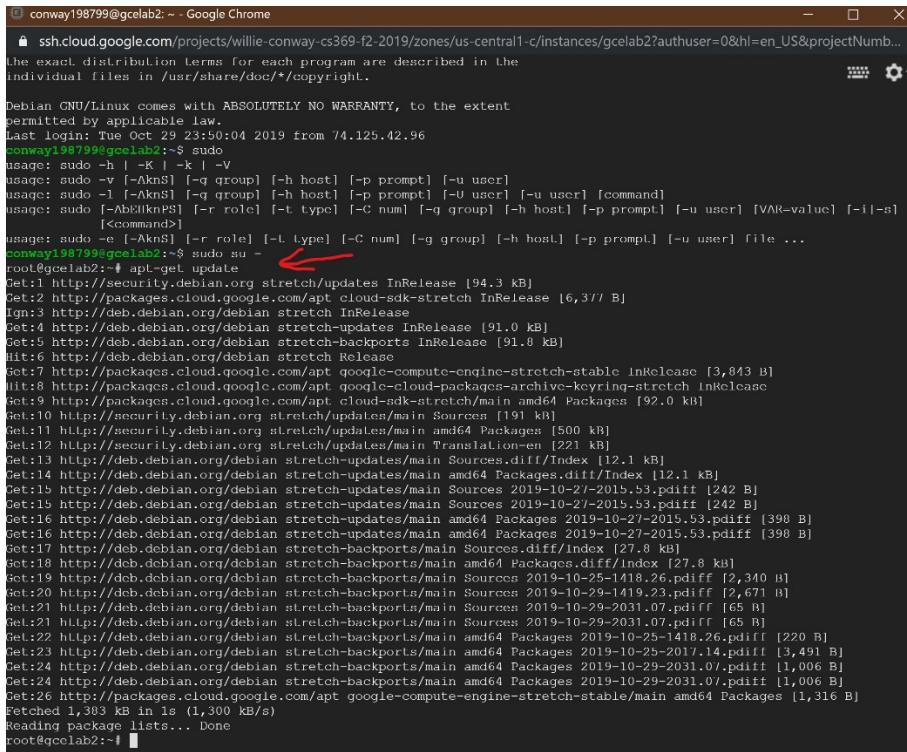
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last Login: Wed Oct 30 17:50:31 2019 from 74.125.42.100
conway198799@gcelab2:~$ pwd
/home/conway198799
conway198799@gcelab2:~$
```

The line "conway198799@gcelab2:~\$ pwd" is highlighted with a red oval, and a red arrow points to the "pwd" command.

*(To assure that I'm in my home directory, I used the `pwd` command to see what my directory name is. As you can see in the screen shot, my home directory is `/home/conway198799`. The `pwd` command is a command line utility for printing the current working directory. It will print the full system path of the current working directory to standard output.)*

4. Install the nginx web server, as you did in the previous Qwiklab.

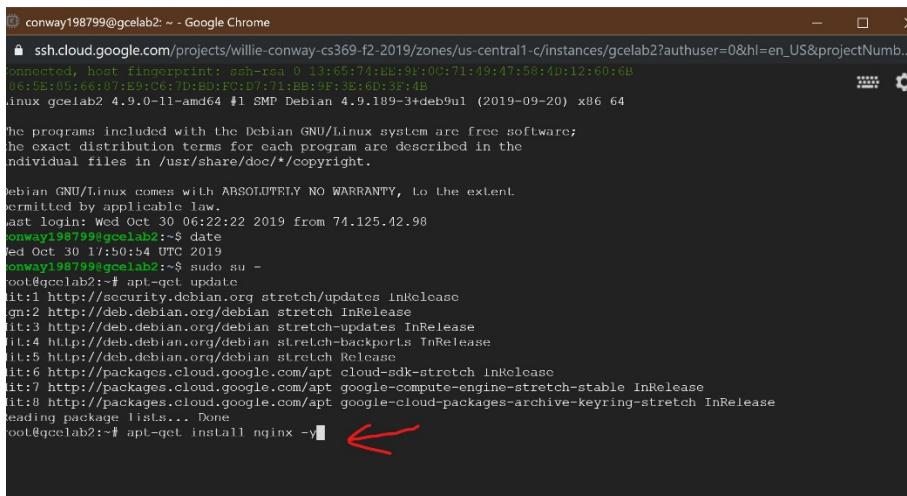
# Unit 2: Homework



```
conway198799@gcelab2: ~ - Google Chrome
ssh.cloud.google.com/projects/willie-conway-cs369-f2-2019/zones/us-central1-c/instances/gcelab2?authuser=0&hl=en_US&projectNum...
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.
Last login: Tue Oct 29 23:50:04 2019 from 74.125.42.96
conway198799@gcelab2:~$ sudo
usage: sudo [-h | -K | -k | -V]
usage: sudo [-v [-AknS] [-g group] [-h host] [-p prompt] [-u user]}
usage: sudo [-l [-AknS] [-g group] [-h host] [-p prompt] [-u user] [-u user] [command]}
usage: sudo [-tAbdklnRSt] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-u user] [-VnR-value] [-i]-s
      [-ccommand>
usage: sudo [-e [-AknS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-u user] file ...
conway198799@gcelab2:~$ sudo su -
root@gcelab2:~# apt-get update
Get:1 http://security.debian.org/debian stretch InRelease [94.3 kB]
Get:2 http://packages.cloud.google.com/apt cloud-sdk-stretch InRelease [6,311 B]
Ign:3 http://deb.debian.org/debian stretch InRelease
Get:4 http://deb.debian.org/debian stretch-updates InRelease [91.0 kB]
Get:5 http://deb.debian.org/debian stretch-backports InRelease [91.8 kB]
Hit:6 http://deb.debian.org/debian stretch Release
Get:7 http://packages.cloud.google.com/apt google-compute-engine-stretch-stable InRelease [3,843 B]
Get:8 http://packages.cloud.google.com/apt google-cloud-packages-archive-keyring-stretch InRelease
Get:9 http://packages.cloud.google.com/apt cloud-sdk-stretch/main amd64 Packages [92.0 kB]
Get:10 http://security.debian.org/debian stretch-updates/main Sources [191 kB]
Get:11 http://security.debian.org/debian stretch-updates/main amd64 Packages [500 kB]
Get:12 http://security.debian.org/debian stretch-updates/main Translation-en [221 kB]
Get:13 http://deb.debian.org/debian stretch-updates/main Sources.diff/Index [12.1 kB]
Get:14 http://deb.debian.org/debian stretch-updates/main amd64 Packages.diff/Index [12.1 kB]
Get:15 http://deb.debian.org/debian stretch-updates/main Sources 2019-10-27-2015.53.pdiff [242 B]
Get:16 http://deb.debian.org/debian stretch-updates/main Sources 2019-10-27-2015.53.pdiff [242 B]
Get:17 http://deb.debian.org/debian stretch-updates/main amd64 Packages 2019-10-27-2015.53.pdiff [398 B]
Get:18 http://deb.debian.org/debian stretch-updates/main amd64 Packages 2019-10-27-2015.53.pdiff [398 B]
Get:19 http://deb.debian.org/debian stretch-backports/main Sources 2019-10-25-1418.26.pdiff [2,340 B]
Get:20 http://deb.debian.org/debian stretch-backports/main Sources 2019-10-29-1419.23.pdiff [2,671 B]
Get:21 http://deb.debian.org/debian stretch-backports/main Sources 2019-10-29-2031.07.pdiff [65 B]
Get:21 http://deb.debian.org/debian stretch-backports/main Sources 2019-10-29-2031.07.pdiff [65 B]
Get:22 http://deb.debian.org/debian stretch-backports/main amd64 Packages 2019-10-25-1418.26.pdiff [220 B]
Get:23 http://deb.debian.org/debian stretch-backports/main amd64 Packages 2019-10-25-2017.14.pdiff [3,491 B]
Get:24 http://deb.debian.org/debian stretch-backports/main amd64 Packages 2019-10-29-2031.07.pdiff [1,006 B]
Get:24 http://deb.debian.org/debian stretch-backports/main amd64 Packages 2019-10-29-2031.07.pdiff [1,006 B]
Get:26 http://packages.cloud.google.com/apt google-compute-engine-stretch-stable/main amd64 Packages [1,316 B]
Fetched 1,383 kB in 1s (1,300 kB/s)
Reading package lists... Done
root@gcelab2:~#
```

(In the process of installing NGINX web server, Once SSH'ed, get root access using sudo. The command sudo is often used to bypass denied permissions. To do this I entered the command sudo su -. This command is essentially the same as just running su in the shell. Instead of telling the system to "switch users" directly, you're telling it to run the "su" command as root. As the root user, update your OS using the command apt-get update. The command apt-get is the command-line tool for handling packages, and may be considered the user's "back-end" to other tools using the APT library. You should then see a series of updates occurring to your system. This can be shown in the screen shot above.)



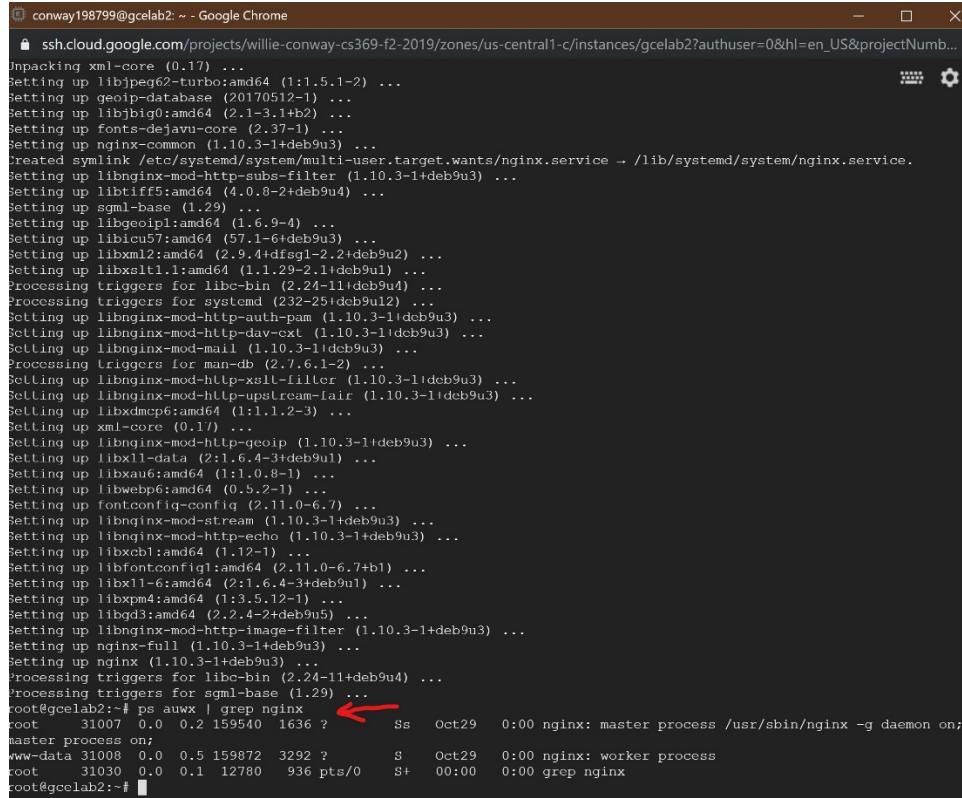
```
conway198799@gcelab2: ~ - Google Chrome
ssh.cloud.google.com/projects/willie-conway-cs369-f2-2019/zones/us-central1-c/instances/gcelab2?authuser=0&hl=en_US&projectNum...
Connected, host fingerprint: ssh-rsa 0:13:65:74:8E:9E:0C:71:49:47:58:4D:12:60:6B
06:5E:05:66:07:E9:C6:D0:FC:D7:71:BB:9F:38:6D:3F:4B
Linux gcelab2 4.9.0-11-amd64 #1 SMP Debian 4.9.199-3+deb9u1 (2019-09-20) 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.
Last login: Wed Oct 30 06:22:22 2019 from 74.125.42.96
conway198799@gcelab2:~$ date
Wed Oct 30 17:50:54 UTC 2019
conway198799@gcelab2:~$ sudo su -
root@gcelab2:~# apt-get update
Get:1 http://security.debian.org/debian stretch InRelease
Get:2 http://deb.debian.org/debian stretch InRelease
Get:3 http://deb.debian.org/debian stretch-updates InRelease
Get:4 http://deb.debian.org/debian stretch-backports InRelease
Get:5 http://deb.debian.org/debian stretch Release
Get:6 http://packages.cloud.google.com/apt cloud-sdk-stretch InRelease
Get:7 http://packages.cloud.google.com/apt google-compute-engine-stretch-stable InRelease
Get:8 http://packages.cloud.google.com/apt google-cloud-packages-archive-keyring-stretch InRelease
Reading package lists... Done
root@gcelab2:~# apt-get install nginx -y
```

# Unit 2: Homework

(Next, we are going to install NGINX utilizing the command `apt-get install nginx -y`. This is shown in the screenshot above. After NGINX web server is being installed you should see the following additional packages being installed. This will be a quick process, so you probably won't catch everything.)



```
conway198799@gcelab2: ~ - Google Chrome
ssh.cloud.google.com/projects/willie-conway-cs369-f2-2019/zones/us-central1-c/instances/gcelab2?authuser=0&hl=en_US&projectNum...
Inpacking xml-core (0.17) ...
Setting up libjpeg62-turbo:amd64 (1:1.5.1-2) ...
Setting up geoip-database (20170512-1) ...
Setting up libbigio:amd64 (2.1-3.1+2) ...
Setting up fonts-dejavu-core (2.37-1) ...
Setting up nginx-common (1.10.3-1+deb9u3) ...
Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service → /lib/systemd/system/nginx.service.
Setting up libnginx-mod-http-subs-filter (1.10.3-1+deb9u3) ...
Setting up libtiff5:amd64 (4.0.8-2+deb9u4) ...
Setting up sgml-base (1.29) ...
Setting up libgeoip1:amd64 (1.6.9-4) ...
Setting up libicu57:amd64 (57.1-6+deb9u3) ...
Setting up libxml2:amd64 (2.9.4+dfsg1-2.1+deb9u2) ...
Setting up libxslt1.1:amd64 (1:1.29-2.1+deb9u1) ...
Processing triggers for libc-bin (2.24-11+deb9u4) ...
Processing triggers for systemd (232-25+deb9u2) ...
Setting up libnginx-mod-http-auth-pam (1:10.3-1+deb9u3) ...
Setting up libnginx-mod-http-dav-ext (1:10.3-1+deb9u3) ...
Setting up libnginx-mod-mail (1:10.3-1+deb9u3) ...
Processing triggers for man-db (2.7.6.1-2) ...
Setting up libnginx-mod-http-xslt-filter (1:10.3-1+deb9u3) ...
Setting up libnginx-mod-http-upstream-fair (1:10.3-1+deb9u3) ...
Setting up libxmlmcp6:amd64 (1:1.1.2-3) ...
Setting up xml-core (0.1/) ...
Setting up libnginx-mod-http-geoip (1:10.3-1+deb9u3) ...
Setting up libxml1-data (2:1.6.4-3+deb9u1) ...
Setting up libxau6:amd64 (1:1.0.8-1) ...
Setting up libwebp6:amd64 (0.5.2-1) ...
Setting up fontconfig-config (2.11.0-6.7) ...
Setting up libnginx-mod-stream (1:10.3-1+deb9u3) ...
Setting up libnginx-mod-http-echo (1:10.3-1+deb9u3) ...
Setting up libxcb1:amd64 (1.12-1) ...
Setting up libfontconfig1:amd64 (2.11.0-6.7+b1) ...
Setting up libx11-6:amd64 (2:1.6.4-3+deb9u1) ...
Setting up libxpm4:amd64 (1:3.5.12-1) ...
Setting up libgd3:amd64 (2.2.4-2+deb9u5) ...
Setting up libnginx-mod-http-image-filter (1:10.3-1+deb9u3) ...
Setting up nginx-full (1:10.3-1+deb9u3) ...
Setting up nginx (1:10.3-1+deb9u4) ...
Processing triggers for libc-bin (2.24-11+deb9u4) ...
Processing triggers for sgml-base (1.29) ...
root@gcelab2:~# ps aux | grep nginx
root      31007  0.0  0.2 159540  1636 ?        Ss   Oct29   0:00 nginx: master process /usr/sbin/nginx -g daemon on;
master process on;
www-data  31008  0.0  0.5 159872  3292 ?        S    Oct29   0:00 nginx: worker process
root      31030  0.0  0.1 12780   936 pts/0     S+  00:00   0:00 grep nginx
root@gcelab2:~#
```

(The next process is checking to see if the actual NGINX web server was installed, to do this I entered the command `ps auwx | grep nginx`. The `ps` (i.e., process status) command is used to provide information about the currently running processes, including their process identification numbers (PIDs). The `grep` command is used to search text or searches the given file for lines containing a match to the given strings or words.)

5. Test your server by connecting to it by the External IP link on the VM instances page in Google Cloud Console. You will see the nginx Welcome page.

# Unit 2: Homework

The screenshot shows the Google Cloud Platform Compute Engine interface. On the left, a sidebar lists options like VM instances, Instance groups, Instance templates, Sole-tenant nodes, Disks, Snapshots, Images, and TPUs. The main area is titled 'VM instances' and displays a table with columns: Name, Zone, Recommendation, In use by, Internal IP, and External IP. The 'External IP' column for the 'gcelab2' instance is circled in red. The IP address shown is 35.194.25.250.

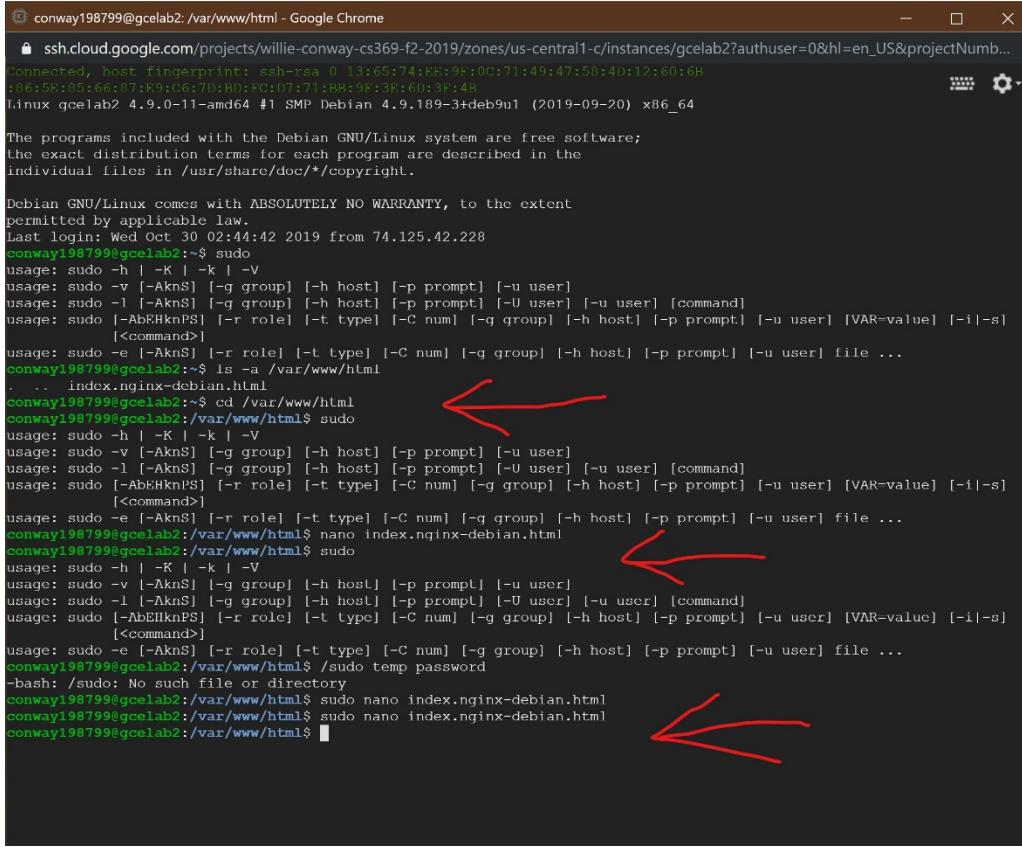
(To assure NGINX is working, I went to the Cloud Console and click the External IP link of the virtual machine instance. You can also see the web page by adding the External IP to [http://EXTERNAL\\_IP/](http://EXTERNAL_IP/) in a new browser window or tab. In the above screenshot, circled in red, you can see the External link IP address is 35.194.25.250.)

The screenshot shows a web browser displaying the 'Welcome to nginx!' default page. The URL in the address bar is 35.194.25.250. The page content includes the heading 'Welcome to nginx!', a message about successful installation, links to online documentation and support, and a thank you note.

(You should see the default web page in the screenshot above.)

6. Using any editor, change the text file found in the directory /var/www/html. Add your name to the welcome message, then save the file. Remember to use sudo to avoid permission errors.

# Unit 2: Homework



```
conway198799@gcelab2:~$ ls -a /var/www/html
.  ..  index.nginx-debian.html
conway198799@gcelab2:~$ cd /var/www/html
conway198799@gcelab2:/var/www/html$ sudo nano index.nginx-debian.html
conway198799@gcelab2:/var/www/html$ nano index.nginx-debian.html
conway198799@gcelab2:/var/www/html$ sudo
conway198799@gcelab2:/var/www/html$ ls -a /var/www/html
.  ..  index.nginx-debian.html
conway198799@gcelab2:/var/www/html$ sudo temp password
conway198799@gcelab2:/var/www/html$ sudo nano index.nginx-debian.html
conway198799@gcelab2:/var/www/html$ sudo nano index.nginx-debian.html
conway198799@gcelab2:/var/www/html$
```

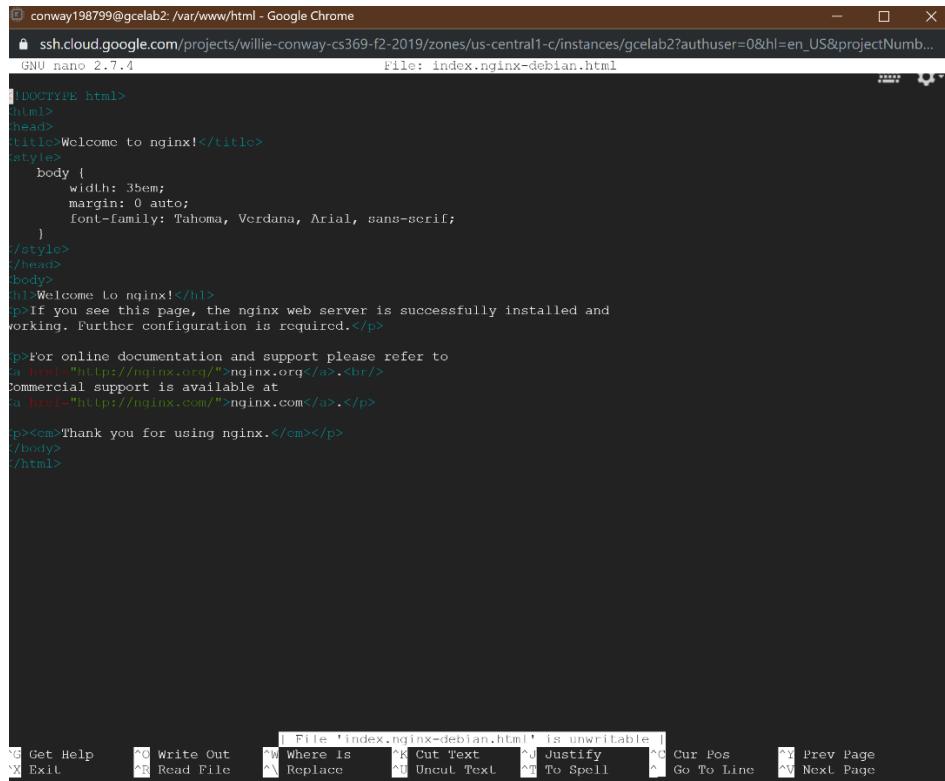
(For this exercise I decided to use the *nano* editor other then the *pico* editor although there are very similar to each other in usage. The *vi/vim* editor appears to be a little more complex. Our first initiative in solving exercise 6 is to find out what file needs to be changed in the */var/www/html* directory. We are currently in our */home/conway198799* directory, but we still can check which file is in the */var/www/html* directory. To do this we need to us the command *ls -a /var/www/html*. The *ls* command is used to get a list of files and directories. Options can be used to get additional information about the files. The *ls -a* option flag lists all files including hidden files. So as seen in the screenshot above, the first arrow is pointing to the only file that happens to be in the */var/www/html* directory, which is *index.nginx-debian.html*.)

(Now that we have found the file that we need to edit, we need access to getting to the file. To do change directories to get to the *index.nginx-debian.html* file in the */var/www/html*. To change directories, we need to utilize the *cd* command. The *cd* command is used to change the current directory. The first arrow points to the command *cd /var/www/html* to initiate the change in directories. You can notice the change from seeing that the */var/www/html* is highlighted in light blue.)

(Now we need to use that *nano* editor to edit the *index.nginx-debian.html* file. In the screenshot above the second arrow displays where we first initiated the command *nano index.nginx-debian.html* to open up the editor. We are then greeted with the screenshot below, which displays all the contents in the file. It seems that the online line that we need to change is line 14.

# Unit 2: Homework

*(This line displays “Welcome to nginx!” we just need to add our name to the welcoming message.)*



```
conway198799@gcelab2: /var/www/html - Google Chrome
ssh.cloud.google.com/projects/willie-conway-cs369-f2-2019/zones/us-central1-c/instances/gcelab2?authuser=0&hl=en_US&projectNumb...
GNU nano 2.7.4
File: index.nginx-debian.html
[...]
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
}
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and working. Further configuration is required.</p>
<p>For online documentation and support please refer to
<a href="http://nginx.org/">http://nginx.org/.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com/.</p>
<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

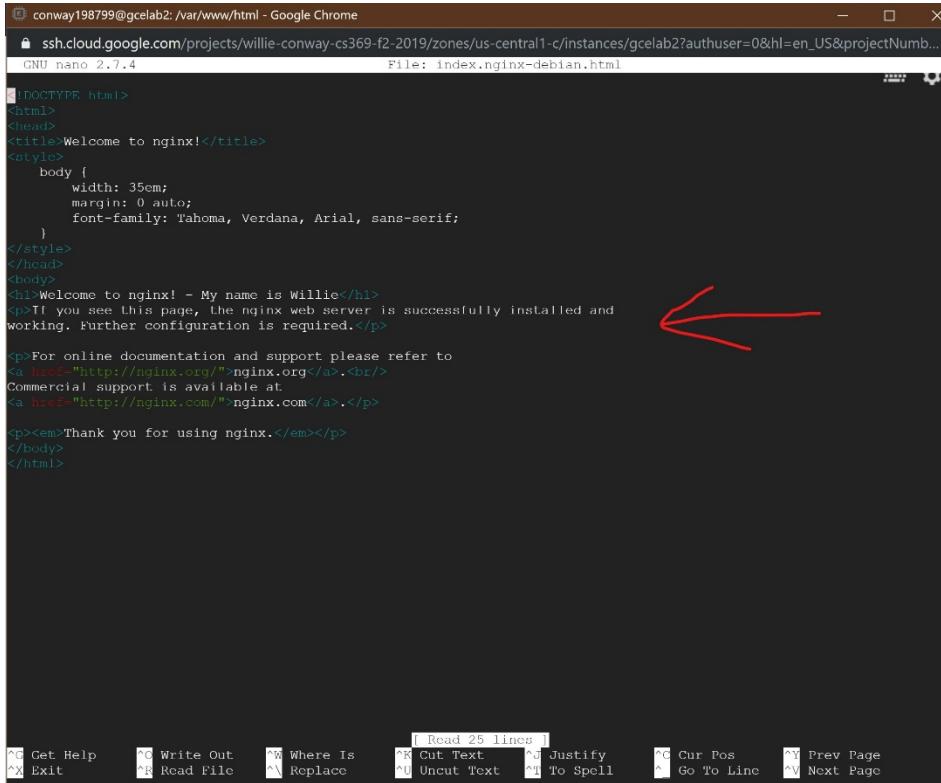
File 'index.nginx-debian.html' is unwriteable!

Get Help Write Out Where Is Cut Text Justify Cur Pos Prev Page  
Exit Read File Replace Undo Text Uncut Text To Spell Go To Line Next Page

*(Looking at the screenshot above, you can see that I edit my name to the welcoming message, but it appears that I can't save the file due to no having permissions to the server, causing the file to be unwriteable. Now, I must exit out of the editor using CTRL+X. I have to find a way to get into the server file by bypassing the permissions. I believe this calls for me to use sudo. I need permissions to the file to edit it.)*

*(Returning back to the first screenshot of exercise 6, the third arrow point to the proper way to bypass the permissions, so that I can edit the index.nginx-debian.html file. I can get access to the file by entering the command sudo nano index.nginx-debian.html.)*

# Unit 2: Homework



```
conway198799@gcelab2: /var/www/html - Google Chrome
ssh.cloud.google.com/projects/willie-conway-cs369-f2-2019/zones/us-central1-c/instances/gcelab2?authuser=0&hl=en_US&projectNum...
GNU nano 2.7.4
File: index.nginx-debian.html

<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
}
</style>
</head>
<body>
<h1>Welcome to nginx! - My name is Willie</h1>
<p>If you see this page, the nginx web server is successfully installed and working. Further configuration is required.</p>
<p><a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at<a href="http://nginx.com/">nginx.com</a>.</p>
<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

(In the screenshot above, now you can see that I have full access to the file as it displays a total of 25 lines that can be read. Now I can edit line 14 by adding "My name is Will" to the "Welcome to nginx!" To save the file we need to select "write out" by pressing **CTRL + O**. If you want to save the changes you've made, press **Ctrl + O**. To exit nano, type **Ctrl + X**. If you ask nano to exit from a modified file, it will ask you if you want to save it, you then press **Enter** to confirm.)

# Unit 2: Homework

A screenshot of a terminal window titled "conway198799@gcelab2: /var/www/html - Google Chrome". The window shows the contents of a file named "index.nginx-debian.html" using the "GNU nano 2.7.4" editor. The file contains HTML code for a basic web page. A red arrow points from the bottom right towards the status bar at the bottom of the terminal window.

```
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
}
</style>
</head>
<body>
<h1>Welcome to nginx! - My name is Willie</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>
<p><a href="http://nginx.org">http://nginx.org</a> .<br/>
Commercial support is available at
<a href="http://nginx.com">http://nginx.com</a> .</p>
<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

File Name to Write: index.nginx-debian.html [ ]

File: index.nginx-debian.html

^G Get Help M-D DOS Format M-A Append M-B Backup File  
^C Cancel M-M Mac Format M-P Prepend ^T To Files

(This screenshot indicates that the file is being saved, we press Enter to save the file.)

A screenshot of a terminal window titled "conway198799@gcelab2: /var/www/html - Google Chrome". The window shows the contents of a file named "index.nginx-debian.html" using the "GNU nano 2.7.4" editor. The file contains the same HTML code as the previous screenshot. The status bar at the bottom shows various keyboard shortcuts for navigating and editing the file.

```
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
}
</style>
</head>
<body>
<h1>Welcome to nginx! - My name is Willie</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>
<p><a href="http://nginx.org">http://nginx.org</a> .<br/>
Commercial support is available at
<a href="http://nginx.com">http://nginx.com</a> .</p>
<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

File Name to Write: index.nginx-debian.html [ ]

File: index.nginx-debian.html

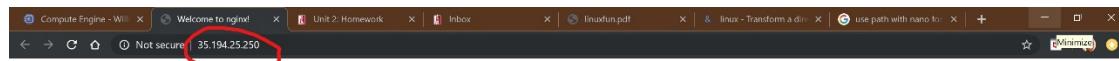
^G Get Help M-D DOS Format M-A Append M-B Backup File  
^C Cancel M-M Mac Format M-P Prepend ^T To Files

^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos ^Y Prev Page  
^X Exit ^R Read File ^A Replace ^U Uncut Text ^T To Spell ^G Go To Line ^V Next Page

# Unit 2: Homework

(This screenshot shows that the file has been saved and it's now good to exit with CTRL + X.)

7. Refresh your nginx welcome page to see your changes. Take a screenshot showing the IP address of the page and your name as part of the welcome.



## Welcome to nginx! - My name is Willie

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](http://nginx.org).  
Commercial support is available at [nginx.com](http://nginx.com).

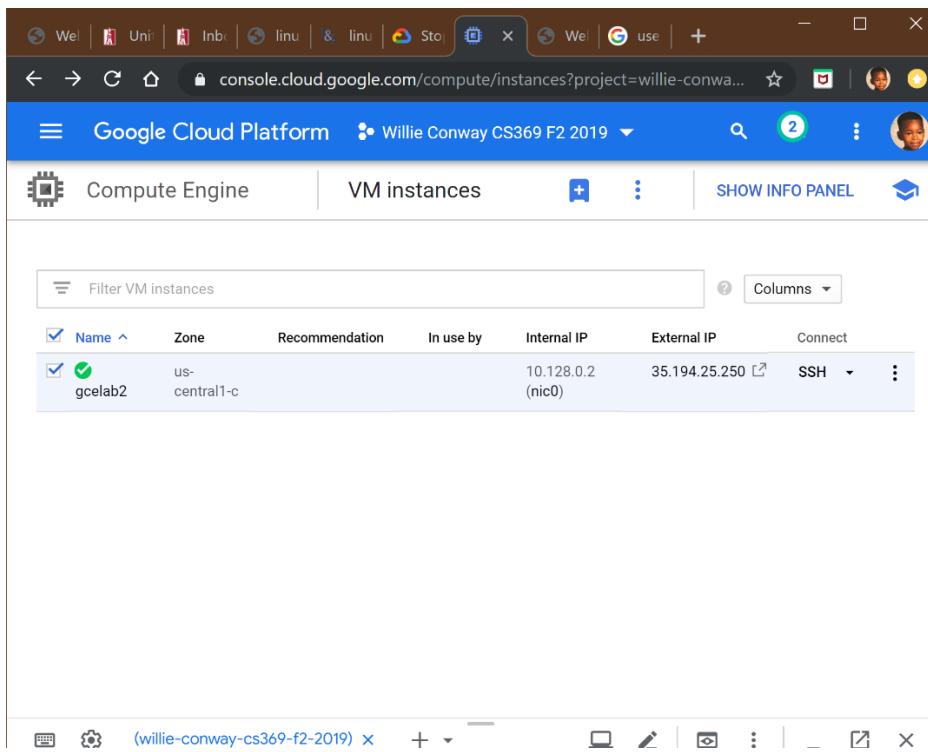
Thank you for using nginx.

(After refreshing the NGINX webserver page, we can see that the welcoming text displays "Welcome to nginx! – My name is Willie." In the address box is the External IP address for the web server which is 35.194.25.250. This is circled in red.)

8. Stop and start your instance from the Cloud Console. Make sure you can still access your web server after a restart. Show screen shots of your instance stopped and running

(To start and stop an instance from the Cloud Console, we have to return to the VM instances page.)

# Unit 2: Homework



The screenshot shows a browser window for the Google Cloud Platform Compute Engine VM instances page. The URL is `console.cloud.google.com/compute/instances?project=willie-conway...`. The page title is "Google Cloud Platform" and the subtitle is "Willie Conway CS369 F2 2019". The main content area displays a table of VM instances. The table has columns: Name, Zone, Recommendation, In use by, Internal IP, External IP, and Connect. A single row is visible, showing "gcelab2" in the Name column, "us-central1-c" in the Zone column, and "willie-conway-cs369-f2-2019" in the In use by column. The Internal IP is 10.128.0.2 (nic0) and the External IP is 35.194.25.250. The Connect dropdown is set to "SSH". There are filter and column settings at the top of the table.

Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
gcelab2	us-central1-c			10.128.0.2 (nic0)	35.194.25.250	SSH

*(Returning to the VM instances page. The instance is currently up and running as you can see that it has a green check mark. If you click on the gcelab2 you can see all the contents to the instance.)*

# Unit 2: Homework

The screenshot shows the Google Cloud Platform Compute Engine VM instances page. A table lists one instance: 'gcelab2' in 'us-central1-c'. The instance has an internal IP of 10.128.0.2 (nic0) and no external IP. An SSH button is present, and a three-dot menu is open next to it, showing options: Start, Stop, Reset, Delete, View network details, and View logs.

Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
gcelab2	us-central1-c			10.128.0.2 (nic0)	None	SSH



(To stop, start, or reset an instance from the GCP, select the instance and locate the three dots next to the SSH button. Click the three dots to display a menu that gives you the options to Start, Stop, Reset, Delete and options to view your network details and view logs. The above screenshot shows the instance being currently stopped. This can take a couple minutes before the process kicks in. As you can see the green checkmark is gone and there is now a gray square icon to simulate the instance is stopped. To start the instance, return to the three-dot menu and select start. The three-dot menu is how you control your instance from the GCP.)



## Welcome to nginx! - My name is Willie

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

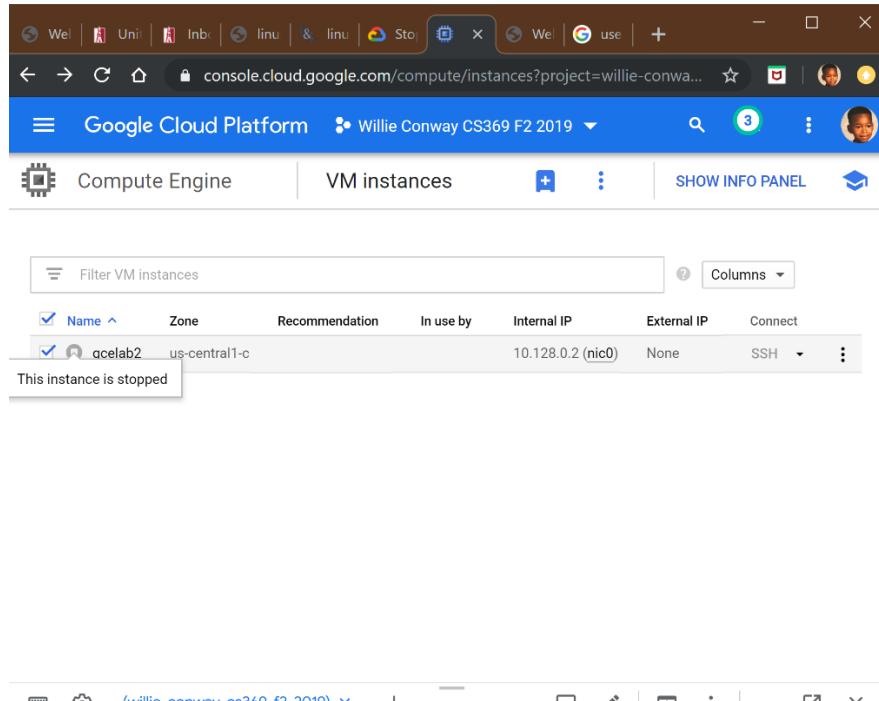
For online documentation and support, please refer to [nginx.org](http://nginx.org).

Commercial support is available at [nginx.com](http://nginx.com).

Thank you for using nginx.

# Unit 2: Homework

*(After restarting the instance in the GCP Console from the VM instance page, I went back to check the NGINX web server, and everything seemed to be working fine as it did before. The screenshot above displays the NGINX webpage after the edit.)*

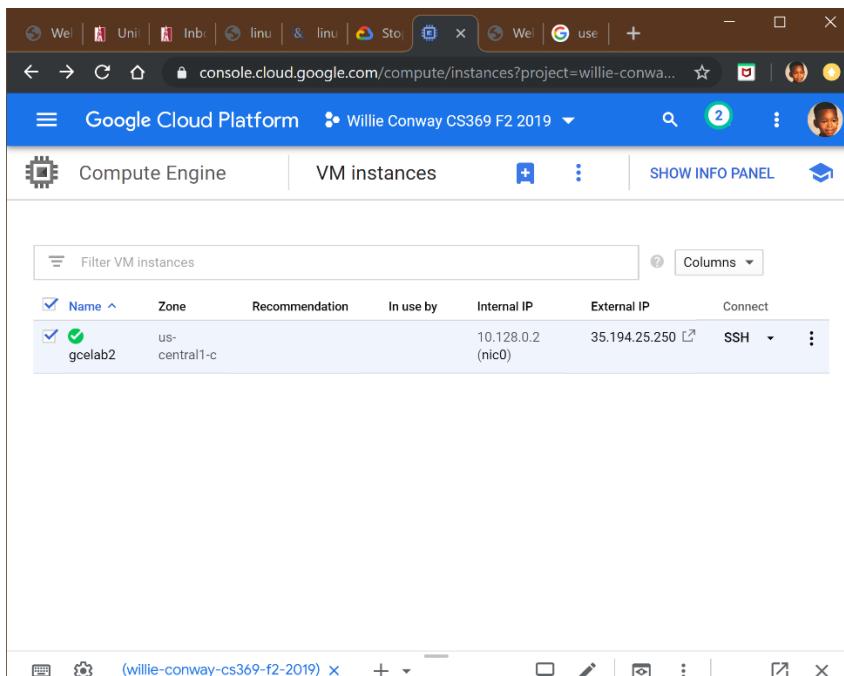


The screenshot shows the Google Cloud Platform Compute Engine VM instances page. The browser address bar shows `console.cloud.google.com/compute/instances?project=willie-conwa...`. The main interface displays a table of VM instances. One instance, named `qcelab2`, is listed with the status `This instance is stopped`.

Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
<code>qcelab2</code>	us-central1-c			10.128.0.2 (nic0)	None	SSH

*(You can also hover your mouse over the icon of your instance once it's stopped. When you do it will display the following instance is stopped.)*

# Unit 2: Homework



The screenshot shows the Google Cloud Platform Compute Engine VM instances page. A single VM instance, 'gcelab2', is listed. The instance has a Name of 'gcelab2', Zone of 'us-central1-c', and an External IP of '35.194.25.250'. The Internal IP is '10.128.0.2 (nic0)'. There is an 'SSH' button next to the IP address. The page includes a filter bar, a table header with columns for Name, Zone, Recommendation, In use by, Internal IP, External IP, and Connect, and a 'SHOW INFO PANEL' button.

Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
gcelab2	us-central1-c			10.128.0.2 (nic0)	35.194.25.250	SSH

*(I tried this with starting the instance, but I didn't get anything when I hovered the mouse over the icon.)*

9. Figure out the commands for stopping and re-starting your instance from the Cloud Shell. Show screen shots of the results of each of these commands.

*(There is a lot to understand when utilizing the Google Cloud Shell. One of which is knowing that you have to have permissions to do task. Controlling an instance from the Console is simple with a couple clicks from the three-dot drop menu from the VM instance page. However, the Active Cloud Shell has a different approach.)*

# Unit 2: Homework

The screenshot shows two browser tabs. The left tab is 'Google Cloud Platform' showing the 'Compute Engine' section with a list of VM instances, including 'gcelab2' in 'us-central1-c'. The right tab is 'cloud.google.com/compute/docs/instances/stop-start-instance' with sections like 'Compute Products', 'Compute Engine', 'Quickstarts', 'How-to guides', and a note about terminated instances.

**Note:** You can review the progression of [instance statuses](#) that describe the process of starting (or restarting) a new or terminated instance, if you are curious.

To reset your instance using gcloud compute:

```
gcloud compute instances start example-instance
```

Restarting an instance that has encrypted disks

Permissions required for this task

If the instance you want to restart uses [customer-supplied encryption keys](#), you must provide those keys when trying to restart the instance.

(When stopping an instance in the Google Cloud Shell we must use the command-line to connect to the instance, from the Shell. To stop an instance from the Google Cloud Shell, we must use the instances stop command to specify one or more instances that you want to stop, make sure the instance is selected in the VM instance page. For this particular instance the command would be gcloud compute instances stop gcelab2. This will then simulate to ask "Did you mean zone /us-east1-b for instance: [gcelab2] (Y/n)? I would select n for no, because the zone that was chosen for the instance was us-central1-c. It will then simulate to find the zone or to see if it exists and prepare to stopping the instance (s). This can take some time, due to simulating. When the instance is stopped will notice the word done next to the instance name in the Google Cloud Shell and the display of the instance on the VM instance page in the stop position.)

The <https://cloud.google.com/compute/docs/instances/stop-start-instance> page notifies that a TERMINATED instance still exists with its configuration settings and instance metadata but loses its in-memory data and virtual machine state. Any resources that are still attached to the terminated instance will remain attached until you manually detach those resources or delete the instance.

# Unit 2: Homework

The screenshot shows two browser tabs. The left tab is 'console.cloud.google.com/compute/instances?project=willie-conwa...' showing a list of VM instances with one named 'gcelab2' selected. The right tab is 'cloud.google.com/compute/docs/instances/stop-start-instance' titled 'Stopping and Starting'.

**VM instances Table:**

Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
gcelab2	us-central1-c			10.120.0.2 (nic0)	35.194.25.250	SSH

**Google Cloud Documentation (Stopping and Starting):**

**TERMINATED STATE:** Whereas methods such as `reset()` and `sudo reboot` only work on instances that are currently running. Most instances can be restarted, as long as the instance is in a TERMINATED state, with the exception of instances with local SSDs attached, which cannot be restarted.

**Note:** You can review the progression of [instance statuses](#) that describe the process of starting (or restarting) a new or terminated instance, if you are curious.

**How-to guides:**

- Creating VM instances
- Connecting to VM instances
- Managing access to VM instances
- Adding storage
- Backing up persistent disks using snapshots
- Building and managing custom images
- Importing custom images and VM instances
- Managing your instances
  - [Stopping and starting an instance](#)
  - Deleting an instance
  - Preventing accidental VM deletion
  - Moving an instance between zones
  - Applying sizing recommendations for

**CONSOLE GCLOUD API**

To reset your instance using `gcloud compute instances stop example-instance`:

```
gcloud compute instances start example-instance
```

*(When starting an instance in the Google Cloud Shell we must use the command-line to connect to the instance, from the Shell. To start an instance from the Goggle Cloud Shell, we must use the instances start command to reset your instance using gcloud compute, make sure the instance is selected in the VM instance page. For this particular instance the command would be gcloud compute instances start gcelab2. This will then simulate to ask "Did you mean zone /us-east1-c/ for instance: /gcelab2/ (Y/n)? I would select n for no, because the zone that was chosen for the instance was us-central1-c. It will then simulate to find the zone or to see if it exists and prepare to starting the instance (s). This can take some time, due to simulating. When the instance is started, you will notice the word done next to the instance name in the Google Cloud Shell and the display of the instance on the VM instance page in the start position.)*

*The <https://cloud.google.com/compute/docs/instances/stop-start-instance> page notifies that The start method restarts an instance in a TERMINATED state, whereas methods such as `reset()` and `sudo reboot` only work on instances that are currently running. Most instances can be restarted, as long as the instance is in a TERMINATED state, with the exception of instances with local SSDs attached, which cannot be restarted.*