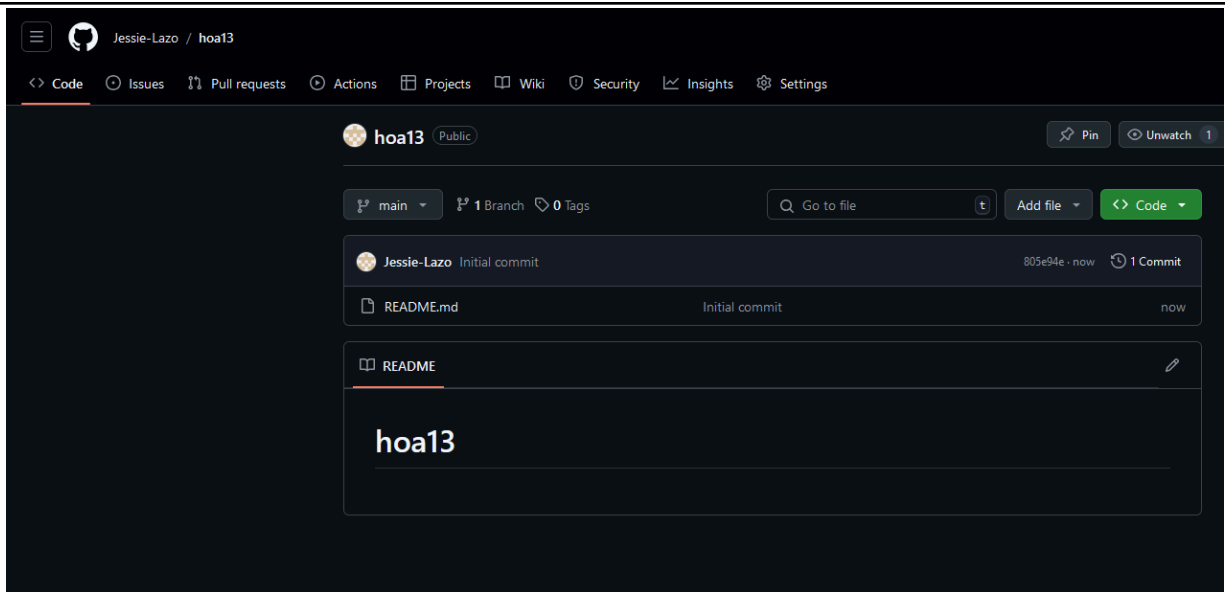


<b>Name: Jessie Robert Lazo</b>	<b>Date Performed:11/27/24</b>
<b>Course/Section:CPE 212-CPE31S2</b>	<b>Date Submitted:12/2/24</b>
<b>Instructor: Engr. Robin Valenzuela</b>	<b>Semester and SY:</b>
<b>Activity 13: OpenStack Prerequisite Installation</b>	
<b>1. Objectives</b>	
Create a workflow to install OpenStack using Ansible as your Infrastructure as Code (IaC).	
<b>2. Intended Learning Outcomes</b>	
<ol style="list-style-type: none"> <li>1. Analyze the advantages and disadvantages of cloud services</li> <li>2. Evaluate different Cloud deployment and service models</li> <li>3. Create a workflow to install and configure OpenStack base services using Ansible as documentation and execution.</li> </ol>	
<b>3. Resources</b>	
<p>Oracle VirtualBox (Hypervisor)</p> <p>1x Ubuntu VM or Centos VM</p>	
<b>4. Tasks</b>	
<ol style="list-style-type: none"> <li>1. Create a new repository for this activity.</li> <li>2. Create a playbook that converts the steps in the following items in <a href="https://docs.openstack.org/install-guide/">https://docs.openstack.org/install-guide/</a> <ol style="list-style-type: none"> <li>a. NTP</li> <li>b. OpenStack packages</li> <li>c. SQL Database</li> <li>d. Message Queue</li> <li>e. Memcached</li> <li>f. Etcd</li> <li>g. Create different plays in installing per server type (controller, compute etc.) and identify it as a group in Inventory file.</li> <li>h. Add, commit and push it to your GitHub repo.</li> </ol> </li> </ol> <p>TASKS: 1.Create a new GitHub repository hoa13</p>	



2. Clone the GitHub repository to the local machine, and set up the Ansible environment with one Ubuntu remote node (since only one machine was asked in the resources section). Make an ansible configuration file and an inventory file needed for the ansible environment.

```
and the repository exists:  
jessielazo@Desktop:~$ git clone https://github.com/Jessie-Lazo/hoa13  
Cloning into 'hoa13'...  
remote: Enumerating objects: 3, done.  
remote: Counting objects: 100% (3/3), done.  
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)  
Unpacking objects: 100% (3/3), done.  
jessielazo@Desktop:~$
```

```
jessielazo@Desktop: ~/hoa13  
File Edit View Search Terminal Help  
GNU nano 2.9.3 ansible.cfg  
[defaults]  
  
inventory = inventory  
host_key_checking = False  
  
deprecation_warning = False  
  
remote_user = jessielazo  
private_key_file = ~/.ssh/
```

```
jessielazo@Desktop: ~/hoa13
File Edit View Search Terminal Help
GNU nano 2.9.3 inventory
[controller]
192.168.56.105 ansible_user=jessieserve ansible_python_interpreter=/usr/bin/py$
```

```
jessielazo@Desktop:~/hoa13$ sudo nano inventory
jessielazo@Desktop:~/hoa13$ ansible all -m ping
192.168.56.105 | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
jessielazo@Desktop:~/hoa13$
```

3. Apply the concept of roles and create the necessary folders, subfolders, and files which will soon be used in the following procedures.

```
jessielazo@Desktop:~/hoa13$ mkdir -p roles/{NTP,OpenStack_packages,SQL_Database,Message_Queue,Memcached,Etc}
jessielazo@Desktop:~/hoa13$ mkdir -p roles/NTP/tasks
jessielazo@Desktop:~/hoa13$ mkdir -p roles/OpenStack_packages/tasks
jessielazo@Desktop:~/hoa13$ mkdir -p roles/SQL_Database/tasks
jessielazo@Desktop:~/hoa13$ mkdir -p roles/Memcached/tasks
jessielazo@Desktop:~/hoa13$ mkdir -p roles/Etc/tasks
jessielazo@Desktop:~/hoa13$ mkdir -p roles/Message_Queue/tasks
jessielazo@Desktop:~/hoa13$ sudo nano roles/NTP/tasks/main.yml
jessielazo@Desktop:~/hoa13$ sudo nano roles/OpenStack_packages/tasks/main.yml
jessielazo@Desktop:~/hoa13$ sudo nano roles/SQL_Database/tasks/main.yml
jessielazo@Desktop:~/hoa13$ sudo nano roles/Message_Queue/tasks/main.yml
jessielazo@Desktop:~/hoa13$ sudo nano roles/Memcached/tasks/main.yml
jessielazo@Desktop:~/hoa13$ sudo nano roles/Etc/tasks/main.yml
jessielazo@Desktop:~/hoa13$
```

```
jessielazo@Desktop: ~/hoa13$ tree
.
├── ansible.cfg
├── inventory
├── README.md
├── roles
│   ├── Etcd
│   │   └── tasks
│   │       └── main.yml
│   ├── Memcached
│   │   └── tasks
│   │       └── main.yml
│   ├── Message_Queue
│   │   └── tasks
│   │       └── main.yml
│   ├── NTP
│   │   └── tasks
│   │       └── main.yml
│   ├── OpenStack_packages
│   │   └── tasks
│   │       └── main.yml
│   └── SQL_Database
│       └── tasks
│           └── main.yml
└── 13 directories, 9 files
jessielazo@Desktop: ~/hoa13$
```

4. Create a playbook 'openstock1.yml' which contains basic repository updates for TyUbuntu followed by the actual program with the concept of roles.

```

jessielazo@Desktop: ~/hoa13
File Edit View Search Terminal Help
GNU nano 2.9.3 openstock1.yml

---
- hosts: all
  become: true
  pre_tasks:

    - name: Ensure dpkg is configured (Ubuntu)
      raw: sudo dpkg --configure -a
      ignore_errors: yes
      changed_when: false
      when: ansible_distribution == "Ubuntu"

    - name: install update and repositories (Ubuntu)
      tags: always
      apt:
        upgrade: yes
        update_cache: yes
        cache_valid_time: 86400
      changed_when: false
      when: ansible_distribution == "Ubuntu"

- hosts: controller
  become: true
  roles:
```

es

Terminal

Wed 08:48

jessielazo@Desktop: ~/hoa13

File Edit View Search Terminal Help

GNU nano 2.9.3openstock1.yml

```
tags: always
apt:
  upgrade: yes
  update_cache: yes
  cache_valid_time: 86400
  changed_when: false
  when: ansible_distribution == "Ubuntu"

- hosts: controller
  become: true
  roles:
    - NTP
    - OpenStack_packages
    - SQL_Database
    - Message_Queue
    - Memcached
    - Etc
```

5. Copy and paste the following codes for setting up the OpenStack in Ubuntu, and its necessary configurations. Role: NTP

jessielazo@Desktop: ~/hoa13/roles/NTP/tasks

File Edit View Search Terminal Help

GNU nano 2.9.3

main.yml

```
- name: install chrony
  shell:
    sudo apt install chrony -y

- name: add ip address as server to /etc/chrony.conf
  lineinfile:
    path: /etc/chrony/chrony.conf
    line: 'server 192.168.56.105 iburst'
    insertafter: '^#.*allow.*$'

- name: add allow line to /etc/chrony.conf
  lineinfile:
    path: /etc/chrony/chrony.conf
    line: 'allow 192.168.56.1/24'
    insertafter: '^#.*allow.*$'

- name: restart chrony service
  service:
    name: chrony
    state: restarted
```

Role: OpenStack\_packages

```

jessielazo@Desktop: ~/hoa13/roles/OpenStack_packages/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml

- name: installs the nova compute component of openstock
  shell:
    sudo apt install nova-compute -y

- name: installs the openstock client
  shell:
    sudo apt install python3-openstackclient -y

- name: add openstock yoga repository
  shell:
    sudo add-apt-repository cloud-archive:yoga -y

```

Role: SQL\_Database

```

jessielazo@Desktop: ~/hoa13/roles/SQL_Database/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml

- name: install mariadb and mysql
  apt:
    name:
      - mariadb-server
      - python3-pymysql
    state: present

- name: create and config 99-openstack.cnf
  template:
    src: 99-openstack.cnf.j2
    dest: /etc/mysql/mariadb.conf.d/99-openstack.cnf

- name: restart mysql service
  service:
    name: mysql
    state: restarted

```



99-openstack.cnf.j2

```
jessielazo@Desktop: ~/hoa13/roles
File Edit View Search Terminal Help
GNU nano 2.9.3 99-openstack.cnf.j2

# /etc/mysql/mariadb.conf.d/99-openstack.cnf

[mysqld]
bind-address = 192.168.56.105

default-storage-engine = innodb
innodb_file_per_table = on
max_connections = 4096
collation-server = utf8_general_ci
character-set-server = utf8
```

Role: Message\_Queue

```
jessielazo@Desktop: ~/hoa13/roles/Message_Queue/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml

- name: install the package
  shell:
    sudo apt install rabbitmq-server -y

- name: add the openstack user
  shell:
    sudo rabbitmqctl add_user openstack RABBIT_PASS

- name: Permit configuration, write, and read access for the openstack user
  shell:
    sudo rabbitmqctl set_permissions openstack ".*" ".*" ".*"
```

Role: Memcached

```
jessielazo@Desktop: ~/hoa13/roles/Memcached/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml

- name: install memcached and its py3 client lib
  apt:
    name:
      - memcached
      - python3-memcache
    state: present

- name: update the memcached conf file
  lineinfile:
    path: /etc/memcached.conf
    regexp: '^(-l\s+)127.0.0.1'
    line: '\g<1>192.168.56.105'

- name: restart memcached service
  service:
    name: memcached
    state: restarted
```

Role: Etcd

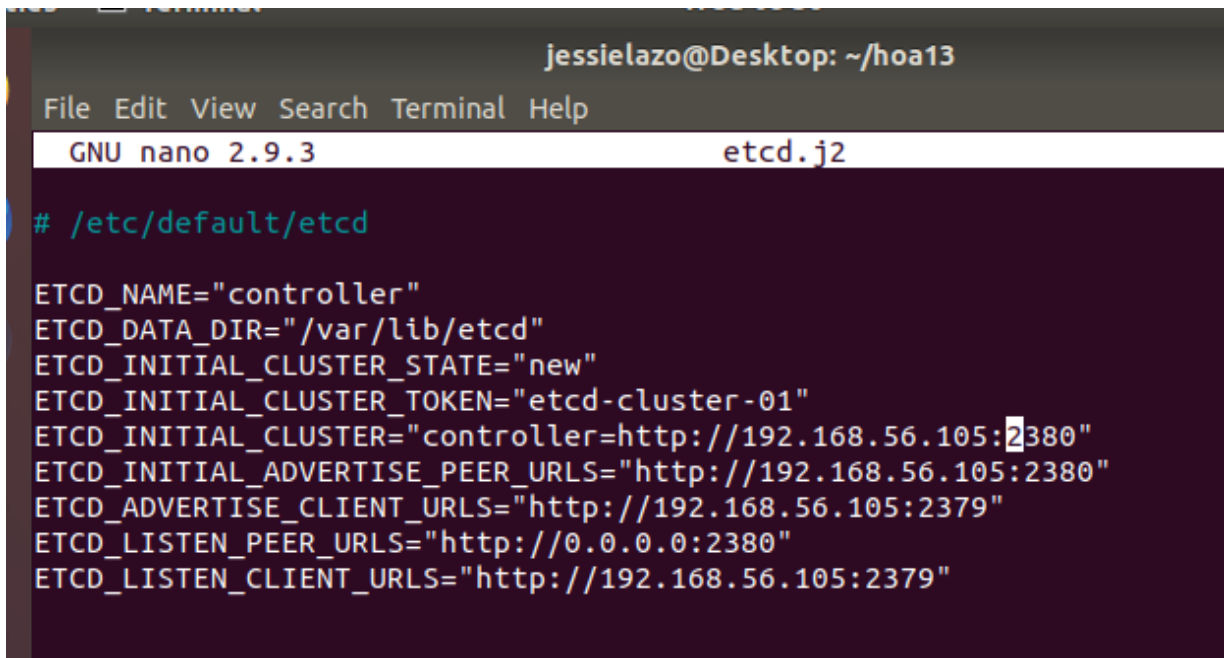
```
jessielazo@Desktop: ~/hoa13/roles/Etcd/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml

- name: install the etcd package
  shell:
    sudo apt install etcd -y

- name: copy etcd conf template
  template:
    src: etcd.j2
    dest: /etc/default/etcd

- name: restart memcached service
  service:
    name: etcd
    state: restarted
```

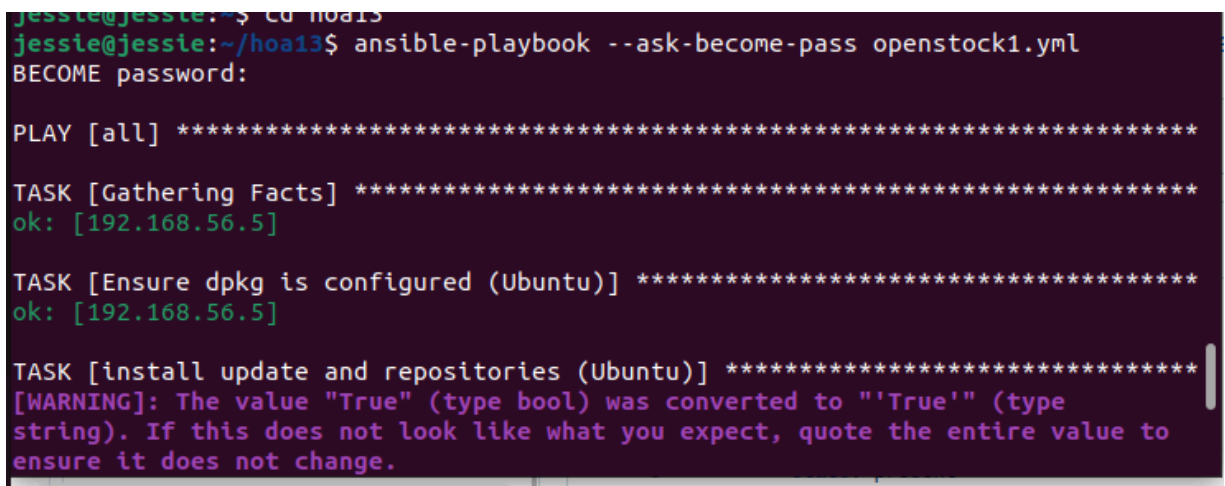
etcd.j2



```
jessielazo@Desktop: ~/hoa13
File Edit View Search Terminal Help
GNU nano 2.9.3 etcd.j2
# /etc/default/etcd

ETCD_NAME="controller"
ETCD_DATA_DIR="/var/lib/etcd"
ETCD_INITIAL_CLUSTER_STATE="new"
ETCD_INITIAL_CLUSTER_TOKEN="etcd-cluster-01"
ETCD_INITIAL_CLUSTER="controller=http://192.168.56.105:2380"
ETCD_INITIAL_ADVERTISE_PEER_URLS="http://192.168.56.105:2380"
ETCD_ADVERTISE_CLIENT_URLS="http://192.168.56.105:2379"
ETCD_LISTEN_PEER_URLS="http://0.0.0.0:2380"
ETCD_LISTEN_CLIENT_URLS="http://192.168.56.105:2379"
```

## 5. Output (screenshots and explanations)



```
jessie@jessie:~$ cd hoa13
jessie@jessie:~/hoa13$ ansible-playbook --ask-become-pass openstock1.yml
BECOME password:

PLAY [all] *****

TASK [Gathering Facts] *****
ok: [192.168.56.5]

TASK [Ensure dpkg is configured (Ubuntu)] *****
ok: [192.168.56.5]

TASK [install update and repositories (Ubuntu)] *****
[WARNING]: The value "True" (type bool) was converted to "'True'" (type
string). If this does not look like what you expect, quote the entire value to
ensure it does not change.
```

```
PLAY [controller] *****

TASK [Gathering Facts] *****
ok: [192.168.56.5]

TASK [NTP : install chrony] *****
[WARNING]: Consider using 'become', 'become_method', and 'become_user' rather
than running sudo
changed: [192.168.56.5]

TASK [NTP : add ip address as server to /etc/chrony.conf] *****
ok: [192.168.56.5]

TASK [NTP : add allow line to /etc/chrony.conf] *****
ok: [192.168.56.5]

TASK [NTP : restart chrony service] *****
changed: [192.168.56.5]
```

```
TASK [NTP : install chrony] *****
*
[WARNING]: Consider using 'become', 'become_method', and 'become_user' rather
than running sudo
changed: [192.168.56.105]

TASK [NTP : add ip address as server to /etc/chrony.conf] *****
*
changed: [192.168.56.105]

TASK [NTP : add allow line to /etc/chrony.conf] *****
*
changed: [192.168.56.105]

TASK [NTP : restart chrony service] *****
*
changed: [192.168.56.105]

TASK [OpenStack_packages : installs the nova compute component of openstock] **
*
changed: [192.168.56.105]

TASK [OpenStack_packages : installs the openstock client] *****
*

```

```
TASK [OpenStack_packages : installs the nova compute component of openstock] ***
changed: [192.168.56.5]

TASK [OpenStack_packages : installs the openstock client] *****
changed: [192.168.56.5]

TASK [OpenStack_packages : add openstock Zed repository] *****
changed: [192.168.56.5]

TASK [SQL_Database : install mariadb and mysql] *****
changed: [192.168.56.5]

TASK [SQL_Database : create and config 99-openstock.cnf] *****
changed: [192.168.56.5]

TASK [SQL_Database : restart mysql service] *****
changed: [192.168.56.5]
```

```
TASK [SQL_Database : install mariadb and mysql] *****
changed: [192.168.56.5]

TASK [SQL_Database : create and config 99-openstock.cnf] *****
changed: [192.168.56.5]

TASK [SQL_Database : restart mysql service] *****
changed: [192.168.56.5]

TASK [Message_Queue : install the package] *****
changed: [192.168.56.5]

TASK [Message_Queue : add the openstack user] *****
changed: [192.168.56.5]

TASK [Message_Queue : Permit configuration, write, and read access for the opens
tack user] ***
changed: [192.168.56.5]

TASK [Memcached : install memcached and its py3 client lib] *****
changed: [192.168.56.5]

TASK [Memcached : update the memcached conf file] *****
changed: [192.168.56.5]
```

```
jessie@jessie: ~/hoa13

TASK [Message_Queue : Permit configuration, write, and read access for the opens
tack user] ***
changed: [192.168.56.5]

TASK [Memcached : install memcached and its py3 client lib] *****
changed: [192.168.56.5]

TASK [Memcached : update the memcached conf file] *****
changed: [192.168.56.5]

TASK [Memcached : restart memcached service] *****
changed: [192.168.56.5]

TASK [Etcd : install the etcd package] *****
changed: [192.168.56.5]

TASK [Etcd : copy etcd conf template] *****
changed: [192.168.56.5]

TASK [Etcd : restart memcached service] *****
changed: [192.168.56.5]

PLAY RECAP *****
192.168.56.5 : ok=23 changed=17 unreachable=0 failed=0 s
kipped=0 rescued=0 ignored=0

jessie@jessie: ~/hoa13$
```

Running 'sudo mysql\_secure\_installation' on server1:

```
jessielazo@jessielazo:~$ sudo mysql_secure_installation
[sudo] password for jessielazo:
sudo: mysql_secure_installation: command not found
jessielazo@jessielazo:~$ sudo mysql_secure_installation
```

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB  
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current  
password for the root user. If you've just installed MariaDB, and  
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none):  
OK, successfully used password, moving on...

Setting the root password or using the unix\_socket ensures that nobody  
can log into the MariaDB root user without the proper authorisation.

You already have your root account protected, so you can safely answer 'n'.

```
jessielazo@jessielazo: ~
```

By default, MariaDB comes with a database named 'test' that anyone can  
access. This is also intended only for testing, and should be removed  
before moving into a production environment.

Remove test database and access to it? [Y/n] y  
- Dropping test database...  
... Success!  
- Removing privileges on test database...  
... Success!

Reloading the privilege tables will ensure that all changes made so far  
will take effect immediately.

Reload privilege tables now? [Y/n] y  
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB  
installation should now be secure.

Thanks for using MariaDB!  
jessielazo@jessielazo:~\$

Proofs:

Thanks for using NetLab.

```
jessielazo@jessielazo:~$ chronyc sources
```

```
\MS Name/IP address          Stratum Poll Reach LastRx Last sample
=====
^- prod-ntp-5.ntp1.ps5.cano>    2   6     3   41    +20ms[ +20ms] +/-  134ms
^* prod-ntp-3.ntp1.ps5.cano>    2   6   377   53   +8667us[ +21ms] +/-  149ms
^- alphyn.canonical.com        2   6   203   42   -6855us[-6855us] +/-  149ms
^- prod-ntp-4.ntp1.ps5.cano>    2   6     3   43   -3467us[-3467us] +/-  161ms
^? 222.127.1.21                 0   8     0   -    +0ns[ +0ns] +/-    0ns
^? 222.127.1.24                 0   8     0   -    +0ns[ +0ns] +/-    0ns
^? 222.127.1.25                 0   8     0   -    +0ns[ +0ns] +/-    0ns
^? 222.127.1.27                 0   8     0   -    +0ns[ +0ns] +/-    0ns
^? 222.127.1.23                 0   7     0   -    +0ns[ +0ns] +/-    0ns
^? 222.127.1.22                 0   8     0   -    +0ns[ +0ns] +/-    0ns
^? port.iwiphil.com             0   8     0   -    +0ns[ +0ns] +/-    0ns
^? jessielazo                   0   7   177   -    +0ns[ +0ns] +/-    0ns
jessielazo@jessielazo:~$ \
```

```
jessielazo@jessielazo:~$
```

```
jessielazo@jessielazo:~$ sudo systemctl status nova-compute
```

```
● nova-compute.service - OpenStack Compute
   Loaded: loaded (/lib/systemd/system/nova-compute.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2024-12-02 14:35:30 CST; 12min ago
     Main PID: 52957 (nova-compute)
        Tasks: 1 (limit: 2271)
       Memory: 148.7M
          CPU: 2.099s
      CGroup: /system.slice/nova-compute.service
              └─52957 /usr/bin/python3 /usr/bin/nova-compute --config-file=/etc/nova/nova.conf
```

```
12月 02 14:35:30 jessielazo systemd[1]: Started OpenStack Compute.
```

```
12月 02 14:35:31 jessielazo nova-compute[52957]: Modules with known eventlet mo>
```

```
lines 1-12/12 (END)
```



```
lines 1-12/12 (END)
jessielazo@jessielazo:~$ sudo systemctl status mariadb
● mariadb.service - MariaDB 10.6.18 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor prese>
   Active: active (running) since Mon 2024-12-02 14:43:52 CST; 4min 41s ago
     Docs: man:mariadb(8)
           https://mariadb.com/kb/en/library/systemd/
  Main PID: 56255 (mariabdd)
    Status: "Taking your SQL requests now..."
     Tasks: 9 (limit: 14990)
  Memory: 63.4M
     CPU: 523ms
    CGroup: /system.slice/mariadb.service
            └─56255 /usr/sbin/mariabdd

12月 02 14:43:52 jessielazo mariabdd[56255]: Version: '10.6.18-MariaDB-0ubuntu0>
12月 02 14:43:52 jessielazo systemd[1]: Started MariaDB 10.6.18 database server.
12月 02 14:43:52 jessielazo /etc/mysql/debian-start[56270]: Upgrading MySQL tab>
12月 02 14:43:52 jessielazo /etc/mysql/debian-start[56273]: Looking for 'mariad>
12月 02 14:43:52 jessielazo /etc/mysql/debian-start[56273]: Looking for 'mariad>
12月 02 14:43:52 jessielazo /etc/mysql/debian-start[56273]: This installation o>
12月 02 14:43:52 jessielazo /etc/mysql/debian-start[56273]: There is no need to>
12月 02 14:43:52 jessielazo /etc/mysql/debian-start[56273]: You can use --force>
```

```
jessielazo@jessielazo:~$ sudo systemctl status mysql
● mariadb.service - MariaDB 10.6.18 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor prese>
   Active: active (running) since Mon 2024-12-02 14:43:52 CST; 5min ago
     Docs: man:mariadb(8)
           https://mariadb.com/kb/en/library/systemd/
  Main PID: 56255 (mariabdd)
    Status: "Taking your SQL requests now..."
     Tasks: 9 (limit: 14990)
  Memory: 63.4M
     CPU: 527ms
    CGroup: /system.slice/mariadb.service
            └─56255 /usr/sbin/mariabdd

12月 02 14:43:52 jessielazo mariabdd[56255]: Version: '10.6.18-MariaDB-0ubuntu0>
12月 02 14:43:52 jessielazo systemd[1]: Started MariaDB 10.6.18 database server.
12月 02 14:43:52 jessielazo /etc/mysql/debian-start[56270]: Upgrading MySQL tab>
12月 02 14:43:52 jessielazo /etc/mysql/debian-start[56273]: Looking for 'mariad>
12月 02 14:43:52 jessielazo /etc/mysql/debian-start[56273]: Looking for 'mariad>
12月 02 14:43:52 jessielazo /etc/mysql/debian-start[56273]: This installation o>
12月 02 14:43:52 jessielazo /etc/mysql/debian-start[56273]: There is no need to>
12月 02 14:43:52 jessielazo /etc/mysql/debian-start[56273]: You can use --force>
```

```
jessielazo@jessielazo:~$ sudo systemctl status rabbitmq-server
● rabbitmq-server.service - RabbitMQ Messaging Server
   Loaded: loaded (/lib/systemd/system/rabbitmq-server.service; enabled; vend
   Active: active (running) since Mon 2024-12-02 14:44:47 CST; 4min 31s ago
     Main PID: 56715 (beam.smp)
        Tasks: 28 (limit: 2271)
      Memory: 90.6M
         CPU: 12.834s
       CGroup: /system.slice/rabbitmq-server.service
               └─56715 /usr/lib/erlang/erts-12.2.1/bin/beam.smp -W w -MBas ageffc
                 └─56727 erl_child_setup 65536
                   └─56786 inet_gethost 4
                     └─56787 inet_gethost 4
                       └─56791 /bin/sh -s rabbit_disk_monitor

12月 02 14:44:32 jessielazo systemd[1]: Starting RabbitMQ Messaging Server...
12月 02 14:44:47 jessielazo systemd[1]: Started RabbitMQ Messaging Server.
lines 1-16/16 (END)
```

```
jessielazo@jessielazo:~$ sudo systemctl status etcd
● etcd.service - etcd - highly-available key value store
   Loaded: loaded (/lib/systemd/system/etcd.service; enabled; vendor preset:
   Active: active (running) since Mon 2024-12-02 14:45:21 CST; 4min 11s ago
     Docs: https://etcd.io/docs
           man:etcd
    Main PID: 57820 (etcd)
         Tasks: 9 (limit: 2271)
        Memory: 5.9M
           CPU: 1.770s
       CGroup: /system.slice/etcd.service
               └─57820 /usr/bin/etcd

12月 02 14:45:21 jessielazo etcd[57820]: 9710ad5d610c7c5a received MsgVoteResp >
12月 02 14:45:21 jessielazo etcd[57820]: 9710ad5d610c7c5a became leader at term>
12月 02 14:45:21 jessielazo etcd[57820]: raft.node: 9710ad5d610c7c5a elected le>
12月 02 14:45:21 jessielazo etcd[57820]: setting up the initial cluster version>
12月 02 14:45:21 jessielazo etcd[57820]: set the initial cluster version to 3.3
12月 02 14:45:21 jessielazo etcd[57820]: enabled capabilities for version 3.3
12月 02 14:45:21 jessielazo etcd[57820]: ready to serve client requests
12月 02 14:45:21 jessielazo etcd[57820]: published {Name:controller ClientURLs:>
12月 02 14:45:21 jessielazo etcd[57820]: serving insecure client requests on 19>
12月 02 14:45:21 jessielazo systemd[1]: Started etcd - highly-available key val>
lines 1-22/22 (END)
```

### Reflections:

Answer the following:

1. What are the benefits of implementing OpenStack?

Implementing OpenStack on a controller provides for central, scalable management of the infrastructure of the cloud. Organizations can efficiently orchestrate and allocate computing, storage, and networking while making available

flexibility and responsiveness to ever-changing demands. Such control from a single source results in streamlined operations with decreased administrative overhead, leading to rapid provisioning of resources. Thus, OpenStack on a controller enables organizations to design an agile and powerful cloud environment.  
enhancing overall productivity and flexibility in the constantly changing face of modern computing

**Conclusions:**

The activity was mainly focused on tackling the OpenStack Prerequisite Installation, and our objective is to create a workflow to install OpenStack using Ansible as Infrastructure as Code (IaC). In the installation guide URL, I found the pages I was looking for which were the guide for installing the NTP, OpenStack packages, SQL Database, Message Queue, Memcached, and Etcd. After compiling the procedures for all of them, then I began creating a playbook to automate the process. Thankfully through the process of playing the playbook, I have encountered just a few errors that I was able to debug quickly, then evaluate different Cloud deployment and service models, then create a workflow to install and configure OpenStack-based services using Ansible as documentation and execution.