Name: Rio Marie G. Suzuki	**Date Performed:** 10/05/2023
Course/Section: CPE232 S6	**Date Submitted:**10/05/2023
Instructor: Dr. Jonathan Taylar	**Semester and SY:** 1st sem 2023-2024

Activity 7: Managing Files and Creating Roles in Ansible

1. **Objectives:**

1.1 Manage files in remote servers

1.2 Implement roles in ansible

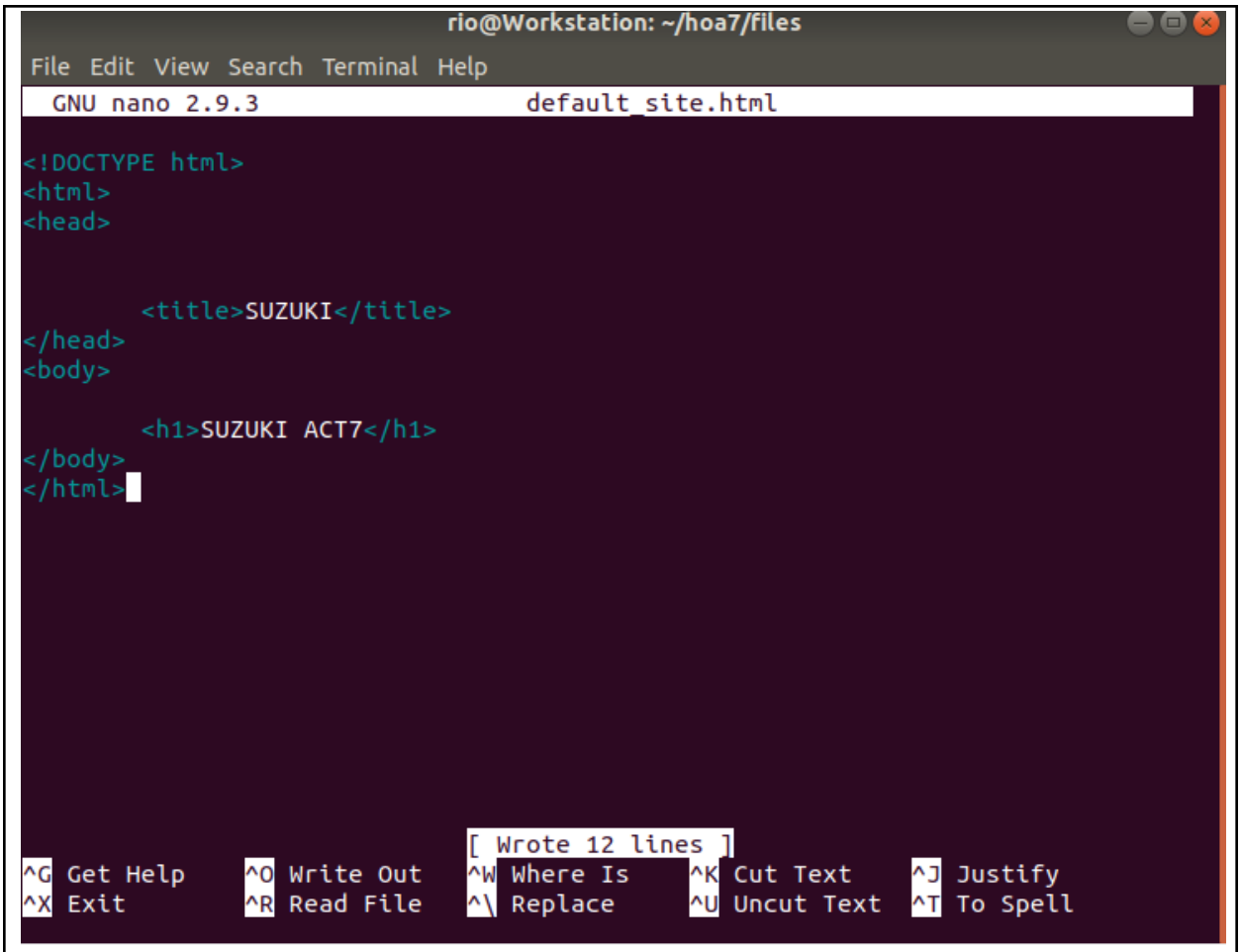
2. **Discussion**:

In this activity, we look at the concept of copying a file to a server. We are going to create a file into our git repository and use Ansible to grab that file and put it into a particular place so that we could do things like customize a default website, or maybe install a default configuration file. We will also implement roles to consolidate plays.

Task 1: Create a file and copy it to remote servers

1. Using the previous directory we created, create a directory, and named it "*files*." Create a file inside that directory and name it "*default_site.html*." Edit the file and put basic HTML syntax. Any content will do, as long as it will display text later. Save the file and exit.

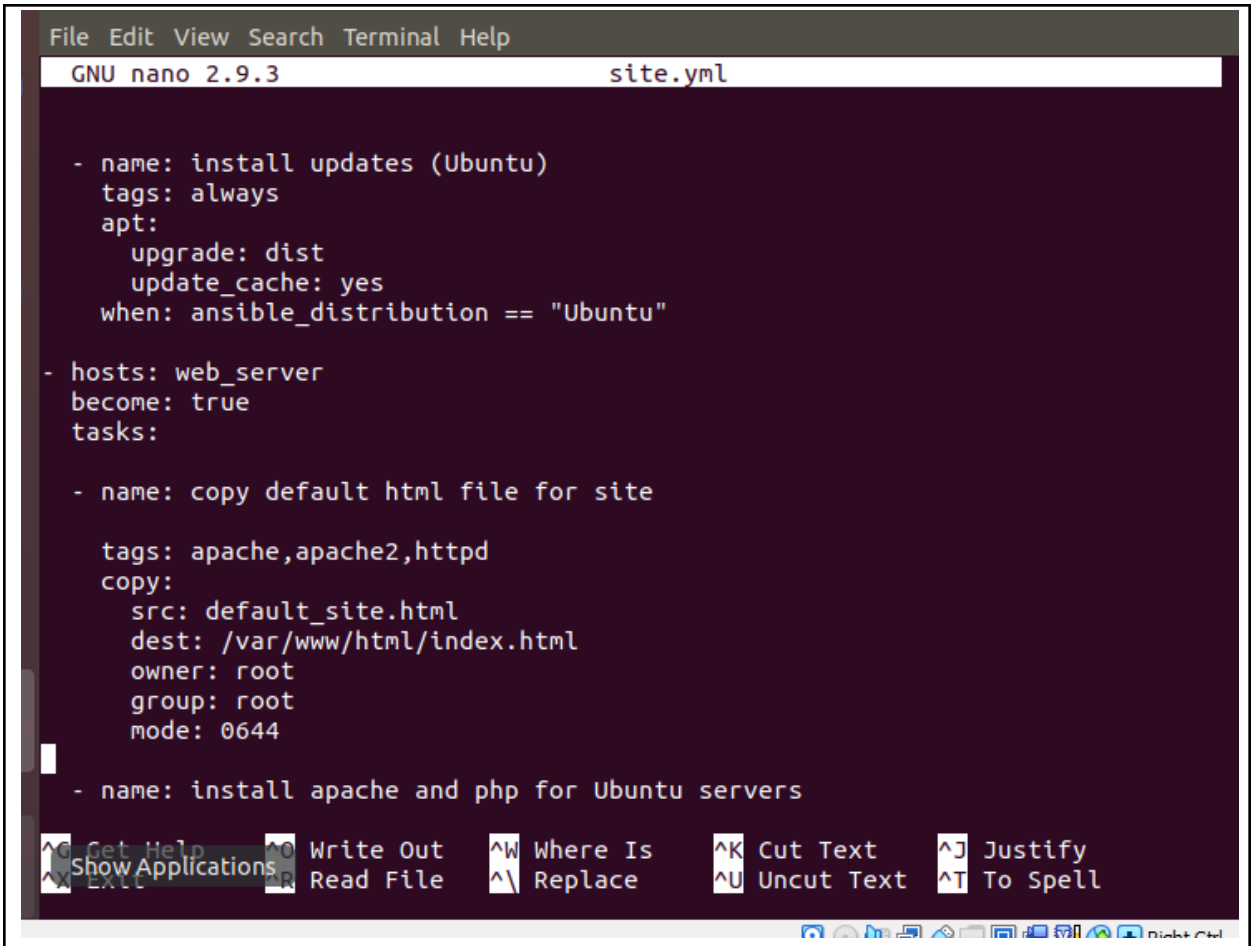
```
rio@Workstation:~$ mkdir hoa7
rio@Workstation:~$ ls
'{'              examples.desktop  'id_rsa\.pub'    Suzuki_PrelimExam
 ansible.cfg     files              inventory       Templates
 config.yaml     HOA6               Music           token
 CPE232_suzuki   hoa7               Pictures        Videos
 Desktop         id_rsa             playbook.yaml   Workstation
 Documents      'id_rsa\'           Public
 Downloads       id_rsa.pub         site.yml
rio@Workstation:~$ cd hoa7
rio@Workstation:~/hoa7$ mkdir files
rio@Workstation:~/hoa7$ cd files
rio@Workstation:~/hoa7/files$
```



2. Edit the *site.yml* file and just below the *web_servers* play, create a new file to copy the default html file for site:
   - name: copy default html file for site

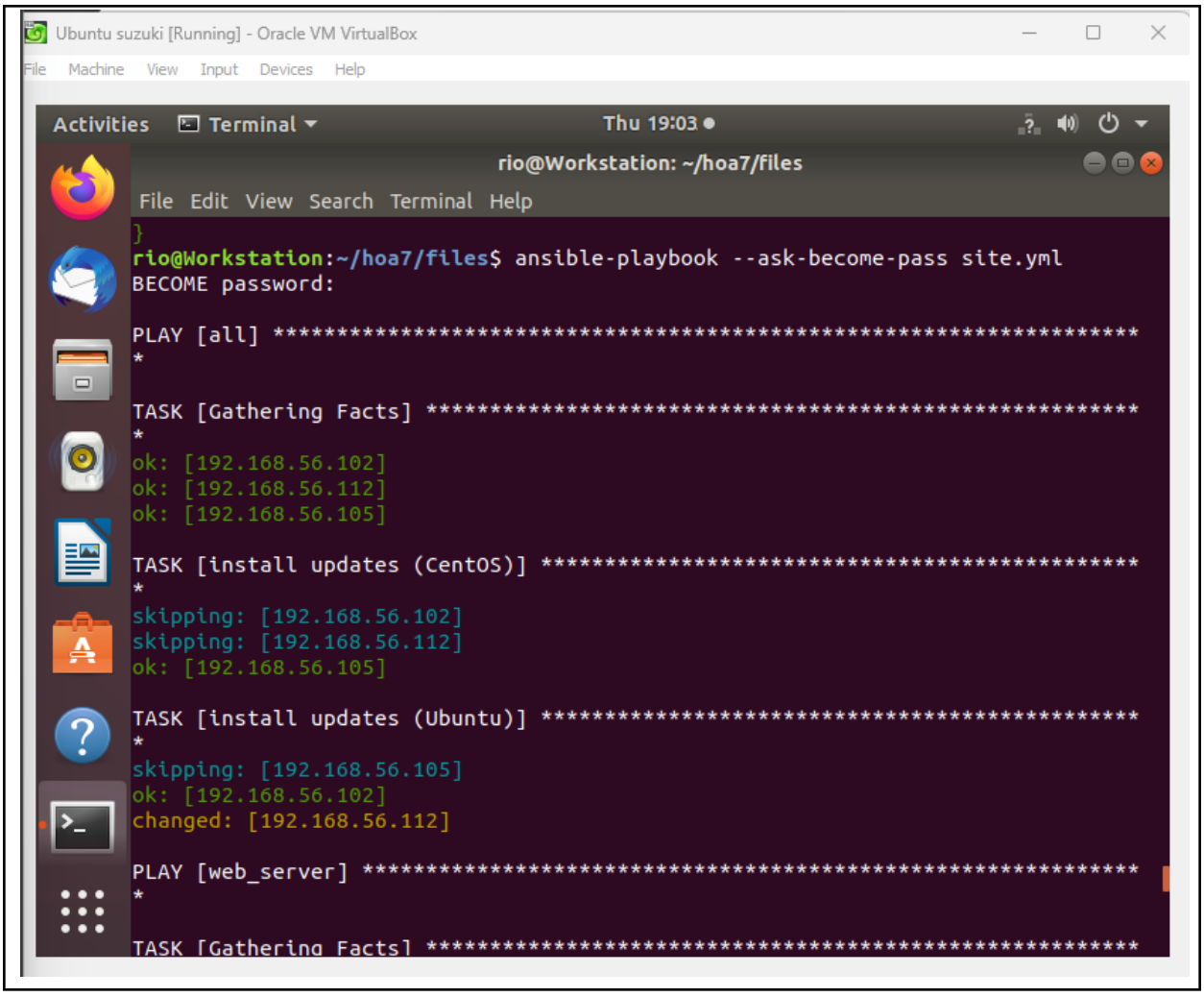
     tags: apache, apache2, httpd
     copy:
       src: default_site.html
       dest: /var/www/html/index.html
       owner: root
       group: root
       mode: 0644

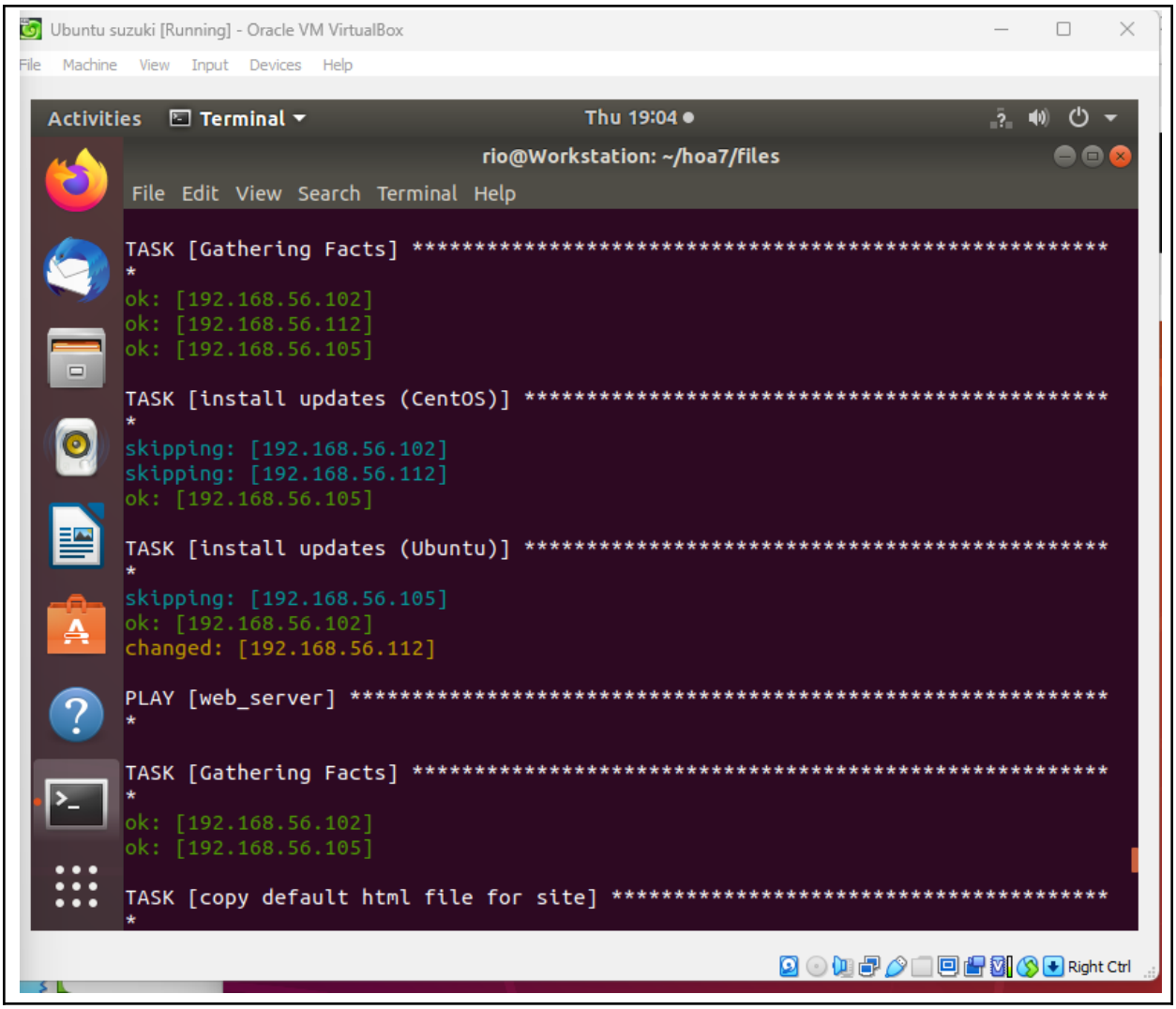
Input:

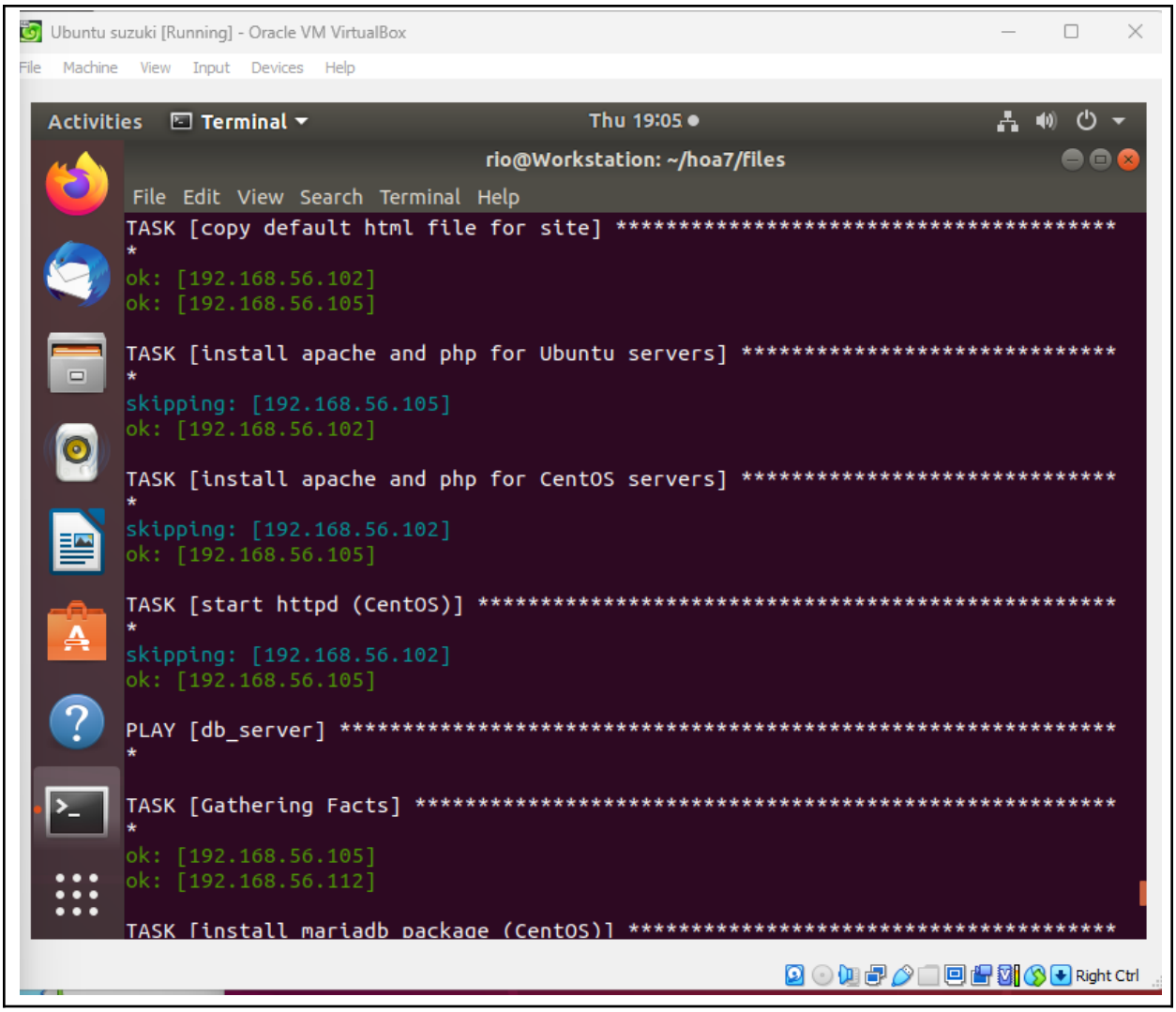


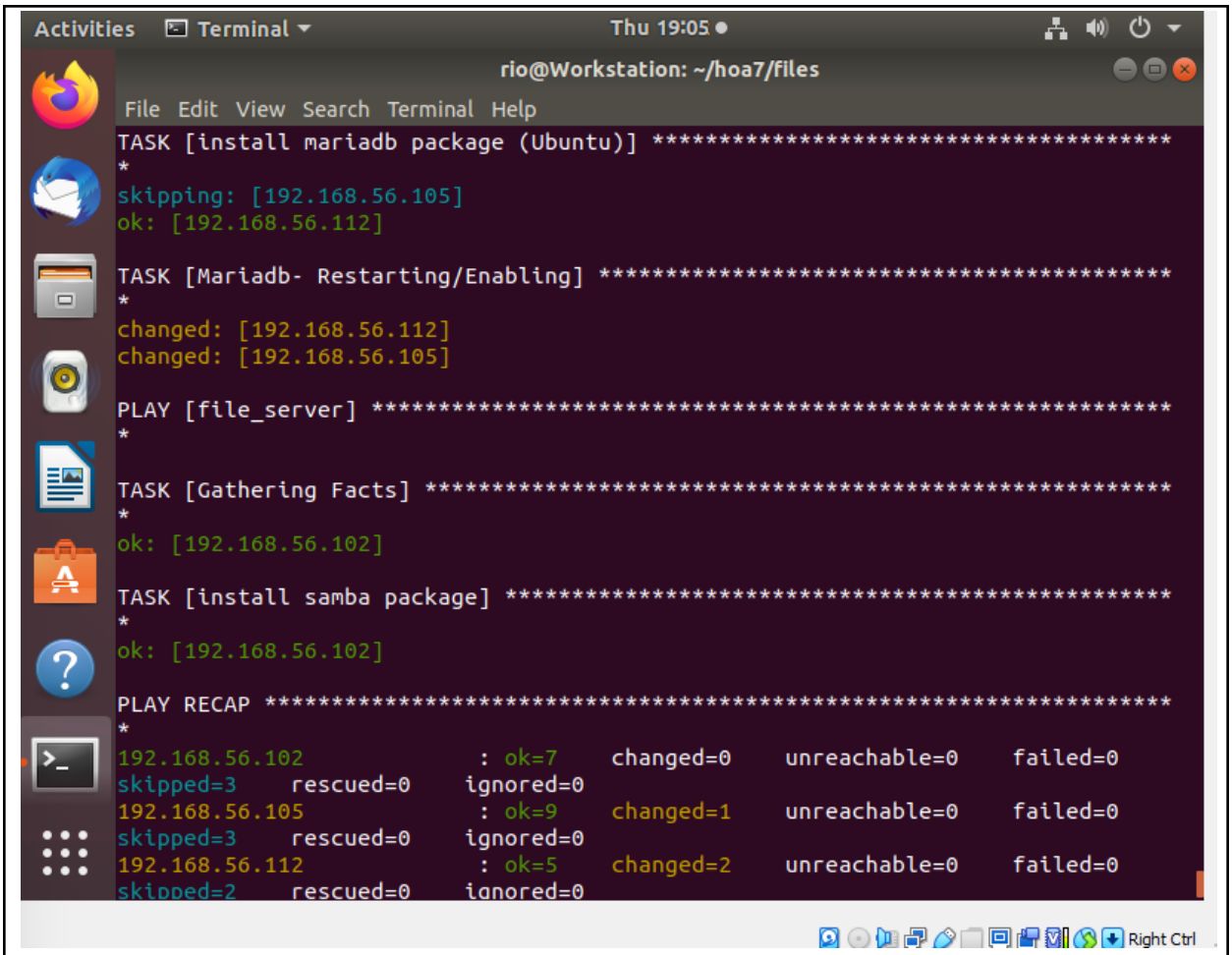
3. Run the playbook *site.yml*. Describe the changes.

Output:

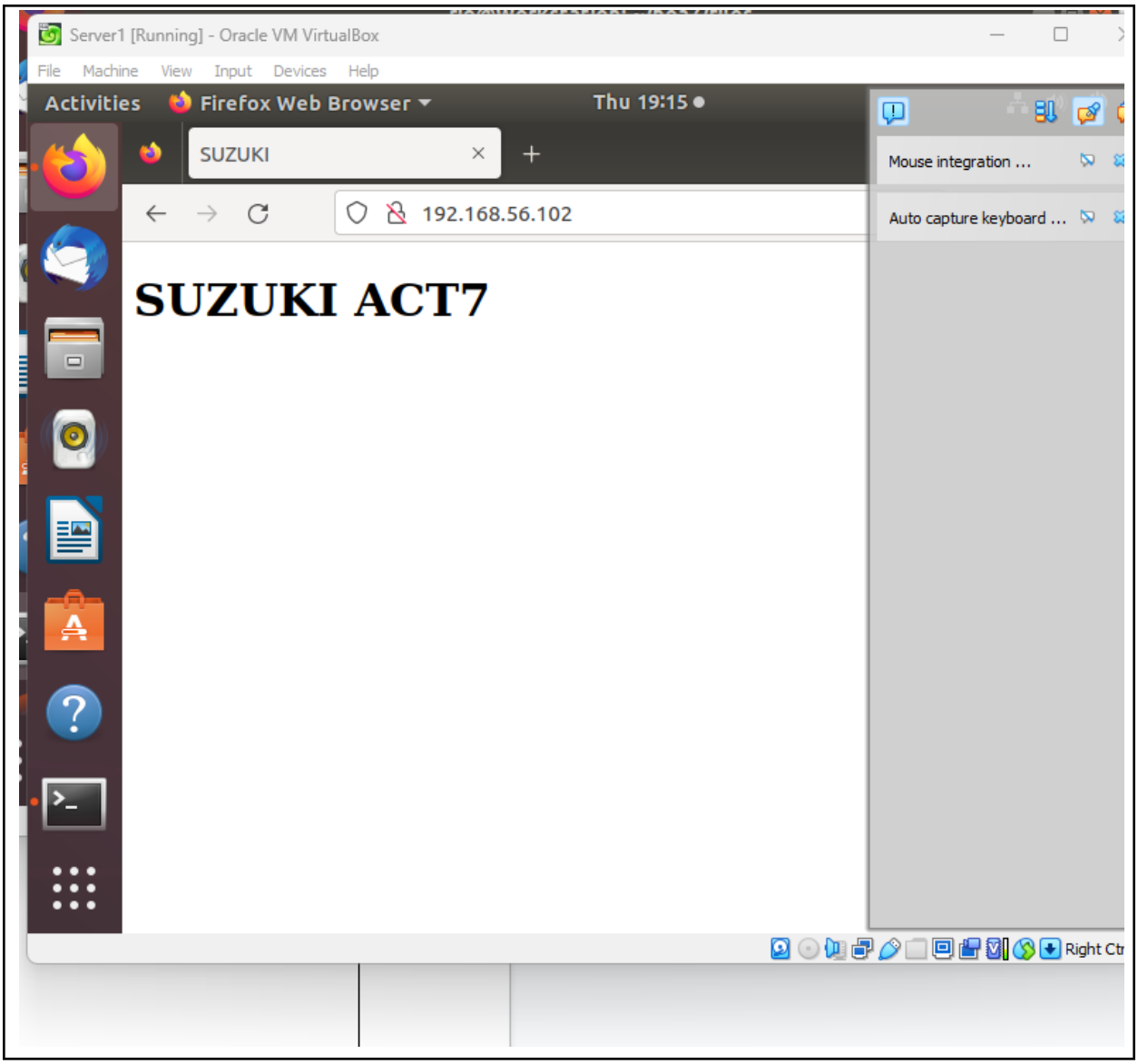

Ubuntu suzuki [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Terminal
Thu 19:03
rio@Workstation: ~/hoa7/files
File Edit View Search Terminal Help
}
rio@Workstation:~/hoa7/files$ ansible-playbook --ask-become-pass site.yml
BECOME password:
PLAY [all] ***
*
TASK [Gathering Facts] ***
*
ok: [192.168.56.102]
ok: [192.168.56.112]
ok: [192.168.56.105]
TASK [install updates (CentOS)] **
*
skipping: [192.168.56.102]
skipping: [192.168.56.112]
ok: [192.168.56.105]
TASK [install updates (Ubuntu)] **
*
skipping: [192.168.56.105]
ok: [192.168.56.102]
changed: [192.168.56.112]
PLAY [web_server] **
*
TASK [Gathering Facts] ***


Ubuntu suzuki [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Terminal
Thu 19:04
rio@Workstation: ~/hoa7/files
File Edit View Search Terminal Help
TASK [Gathering Facts] *********
*
ok: [192.168.56.102]
ok: [192.168.56.112]
ok: [192.168.56.105]
TASK [install updates (CentOS)] *********
*
skipping: [192.168.56.102]
skipping: [192.168.56.112]
ok: [192.168.56.105]
TASK [install updates (Ubuntu)] *********
*
skipping: [192.168.56.105]
ok: [192.168.56.102]
changed: [192.168.56.112]
PLAY [web_server] *********
*
TASK [Gathering Facts] *********
*
ok: [192.168.56.102]
ok: [192.168.56.105]
TASK [copy default html file for site] *********
*
Right Ctrl


Ubuntu suzuki [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Terminal
Thu 19:05
rio@Workstation: ~/hoa7/files
File Edit View Search Terminal Help
TASK [copy default html file for site] **
*
ok: [192.168.56.102]
ok: [192.168.56.105]
TASK [install apache and php for Ubuntu servers] ******************************
*
skipping: [192.168.56.105]
ok: [192.168.56.102]
TASK [install apache and php for CentOS servers] ******************************
*
skipping: [192.168.56.102]
ok: [192.168.56.105]
TASK [start httpd (CentOS)] ***
*
skipping: [192.168.56.102]
ok: [192.168.56.105]
PLAY [db_server] **
*
TASK [Gathering Facts] **
*
ok: [192.168.56.105]
ok: [192.168.56.112]
TASK [install mariadb package (CentOS)] ***************************************
Right Ctrl



4. Go to the remote servers (*web_servers*) listed in your inventory. Use cat command to check if the index.html is the same as the local repository file (*default_site.html*). Do both for Ubuntu and CentOS servers. On the CentOS server, go to the browser and type its IP address. Describe the output.


Server1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Firefox Web Browser
Thu 19:15
SUZUKI
192.168.56.102
SUZUKI ACT7
Mouse integration ...
Auto capture keyboard ...
Right Ctr

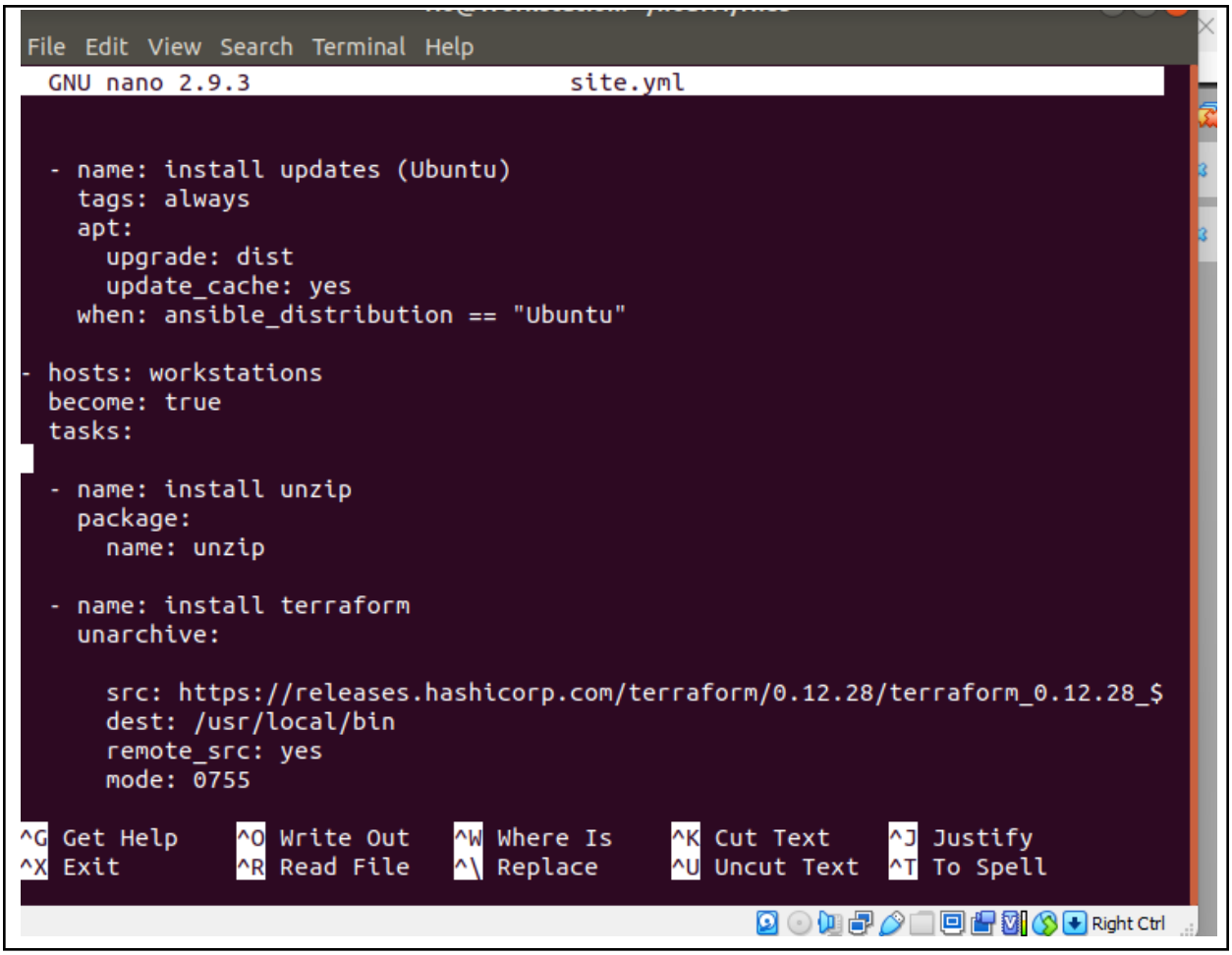
5. Sync your local repository with GitHub and describe the changes.
https://github.com/RioMariee/hoa7.1.git

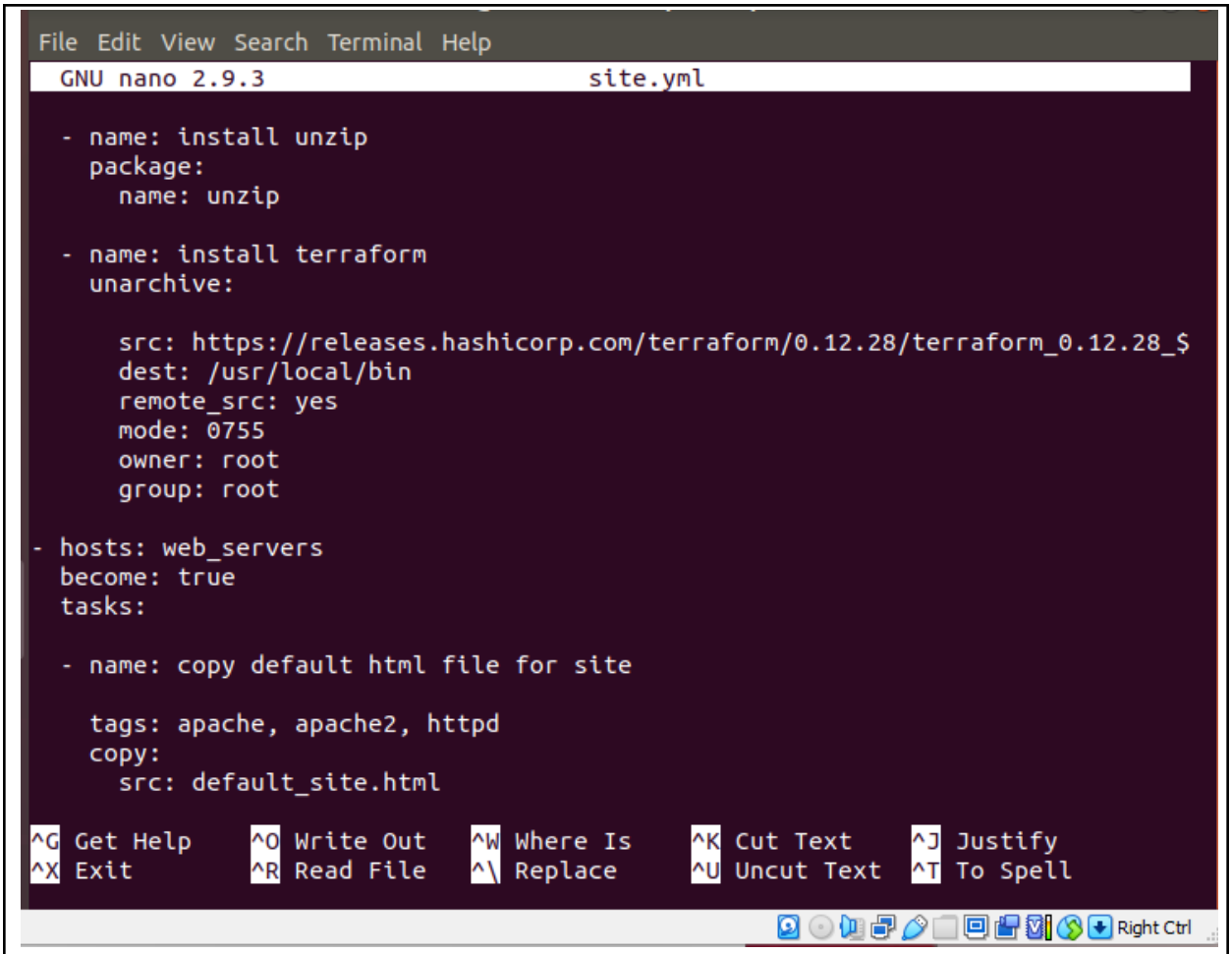
Task 2: Download a file and extract it to a remote server

1. Edit the site.yml. Just before the web_servers play, create a new play:

```
- hosts: workstations
  become: true
  tasks:

  - name: install unzip
    package:
      name: unzip

  - name: install terraform
    unarchive:
      src: https://releases.hashicorp.com/terraform/0.12.28/terraform_0.12.28_linux_amd64.zip
      dest: /usr/local/bin
      remote_src: yes
      mode: 0755
      owner: root
      group: root
```

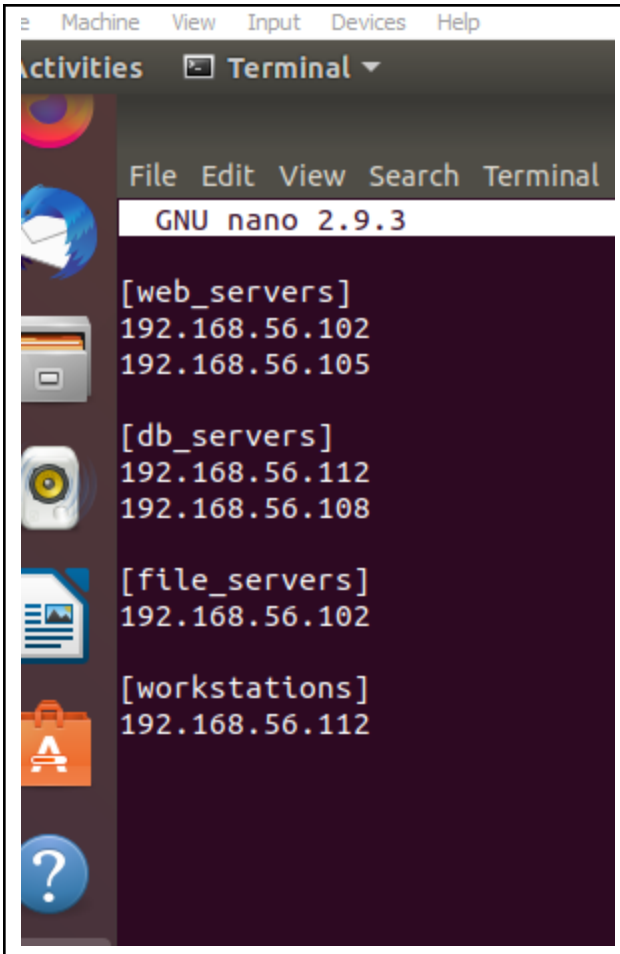
Input:


File Edit View Search Terminal Help
GNU nano 2.9.3 site.yml
- name: install updates (Ubuntu)
tags: always
apt:
upgrade: dist
update_cache: yes
when: ansible_distribution == "Ubuntu"
- hosts: workstations
become: true
tasks:
- name: install unzip
package:
name: unzip
- name: install terraform
unarchive:
src: https://releases.hashicorp.com/terraform/0.12.28/terraform_0.12.28_$
dest: /usr/local/bin
remote_src: yes
mode: 0755
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell
Right Ctrl



2. Edit the inventory file and add workstations group. Add any Ubuntu remote server. Make sure to remember the IP address.

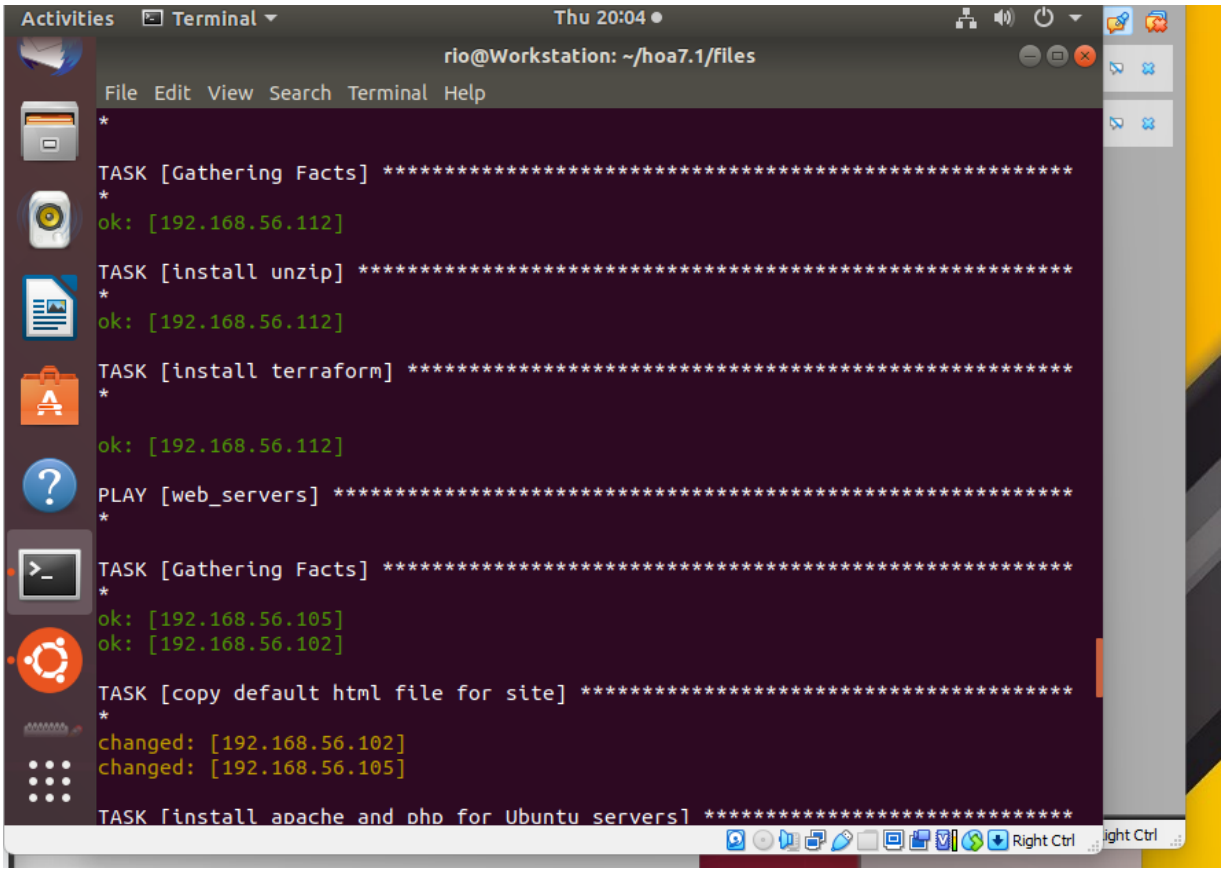
Input:

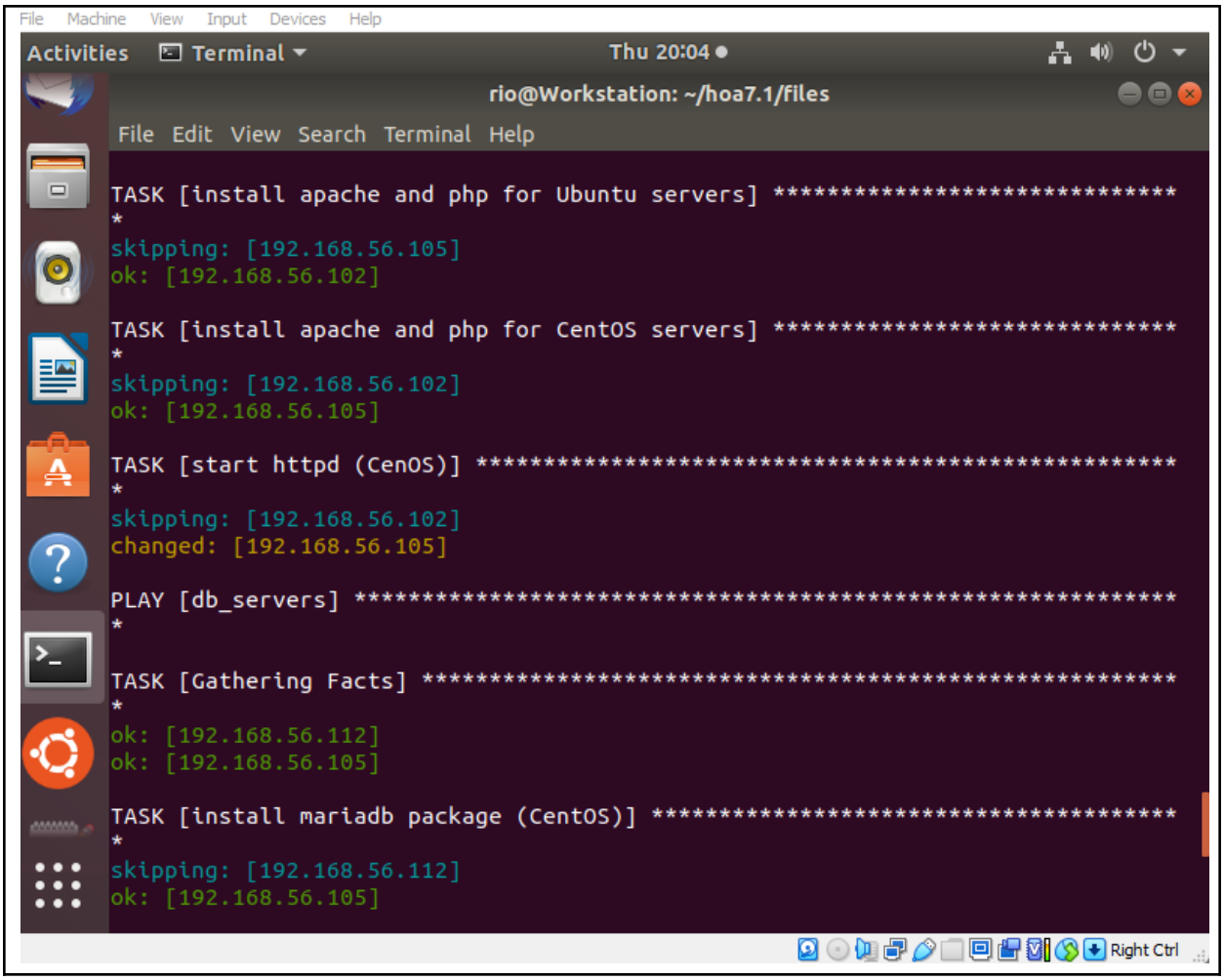


3. Run the playbook. Describe the output.

Output:

```
rio@Workstation:~/hoa7.1/files$ ansible-playbook --ask-become-pass site.yml
BECOME password:

PLAY [all] *********************************************************************
*

TASK [Gathering Facts] *********************************************************
*
ok: [192.168.56.112]
ok: [192.168.56.105]
ok: [192.168.56.102]

TASK [install updates (CentOS)] ************************************************
*
skipping: [192.168.56.102]
skipping: [192.168.56.112]
ok: [192.168.56.105]

TASK [install updates (Ubuntu)] ************************************************
*
skipping: [192.168.56.105]
ok: [192.168.56.112]
ok: [192.168.56.102]

PLAY [workstations] ************************************************************
*
```




File Machine View Input Devices Help
Activities
Terminal
Thu 20:04
rio@Workstation: ~/hoa7.1/files
File Edit View Search Terminal Help
TASK [install apache and php for Ubuntu servers] ********************************
*
skipping: [192.168.56.105]
ok: [192.168.56.102]
TASK [install apache and php for CentOS servers] ********************************
*
skipping: [192.168.56.102]
ok: [192.168.56.105]
TASK [start httpd (CenOS)] **
*
skipping: [192.168.56.102]
changed: [192.168.56.105]
PLAY [db_servers] ***
*
TASK [Gathering Facts] **
*
ok: [192.168.56.112]
ok: [192.168.56.105]
TASK [install mariadb package (CentOS)] ***
*
skipping: [192.168.56.112]
ok: [192.168.56.105]
Right Ctrl

```
ok: [192.168.56.105]

TASK [Mariadb- Restarting/Enabling] ****************************************
*
changed: [192.168.56.112]
changed: [192.168.56.105]

TASK [install mariadb package (Ubuntu)] ************************************
*
skipping: [192.168.56.105]
ok: [192.168.56.112]

PLAY [file_servers] ********************************************************
*

TASK [Gathering Facts] *****************************************************
*
ok: [192.168.56.102]

TASK [install samba package] ***********************************************
*
ok: [192.168.56.102]

PLAY RECAP *****************************************************************
*
192.168.56.102             : ok=7    changed=1    unreachable=0    failed=0
skipped=3    rescued=0    ignored=0
192.168.56.105             : ok=9    changed=3    unreachable=0    failed=0
skipped=3    rescued=0    ignored=0
```

4. On the Ubuntu remote workstation, type terraform to verify installation of terraform. Describe the output.

```
rio@Workstation:~/hoa7.1/files$ terraform

Command 'terraform' not found, but can be installed with:

sudo snap install terraform

rio@Workstation:~/hoa7.1/files$
```

```
       --classic.
rio@Workstation:~/hoa7.1/files$ sudo snap install terraform --classic
Download snap "terraform" (552) from channel "stable"        10% 1.09MB/s 22.2s
```

```
       --classic.
rio@Workstation:~/hoa7.1/files$ sudo snap install terraform --classic
terraform 1.6.0 from Jon Seager (jnsgruk) installed
```

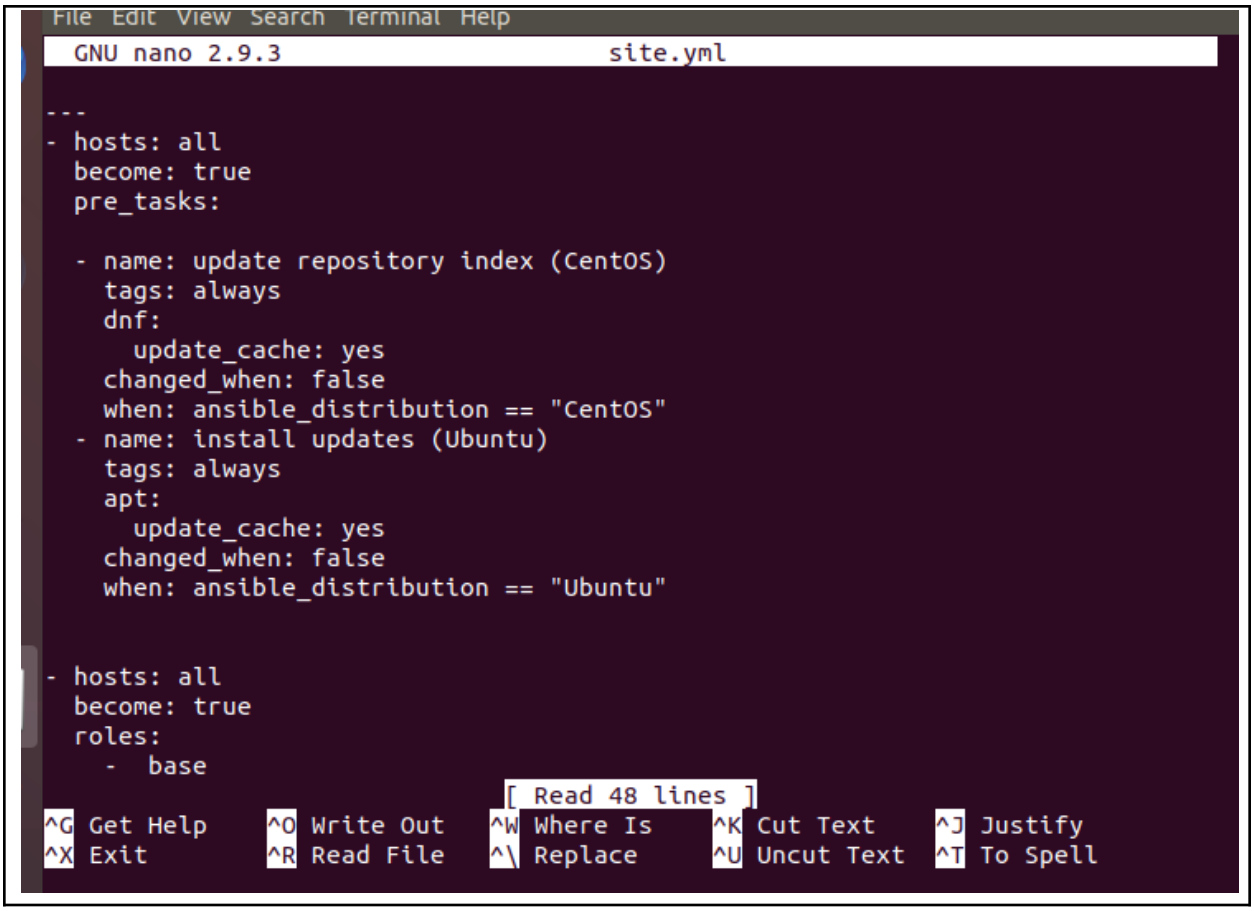
Task 3: Create roles

1. Edit the site.yml. Configure roles as follows: (make sure to create a copy of the old site.yml file because you will be copying the specific plays for all groups)

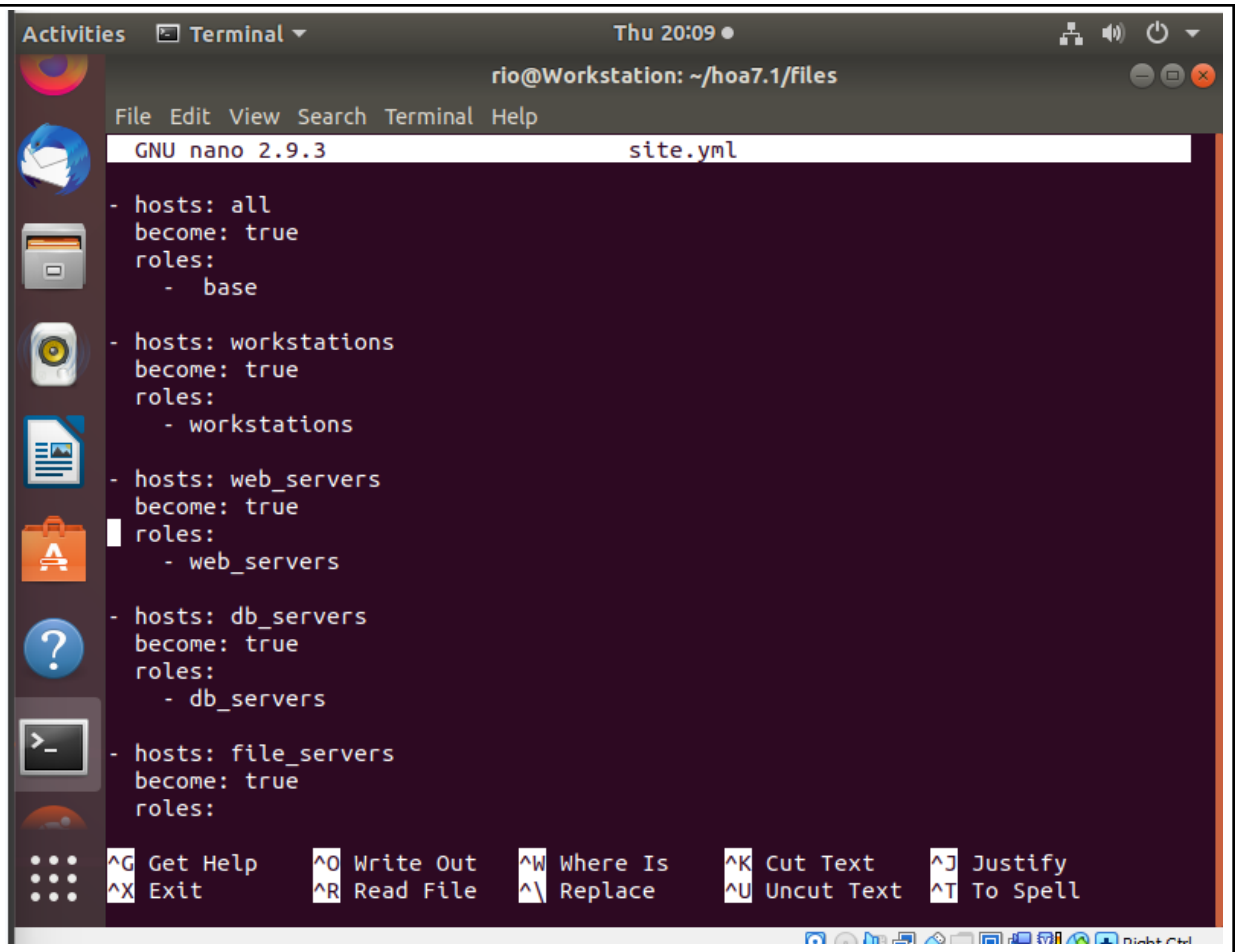
```
---
- hosts: all
  become: true
  pre_tasks:

  - name: update repository index (CentOS)
    tags: always
    dnf:
      update_cache: yes
    changed_when: false
    when: ansible_distribution == "CentOS"
  - name: install updates (Ubuntu)
    tags: always
    apt:
      update_cache: yes
    changed_when: false
    when: ansible_distribution == "Ubuntu"

- hosts: all
  become: true
  roles:
    -  base

- hosts: workstations
  become: true
  roles:
    - workstations

- hosts: web_servers
  become: true
  roles:
    - web_servers

- hosts: db_servers
  become: true
  roles:
    - db_servers

- hosts: file_servers
  become: true
  roles:
    - file_servers
```

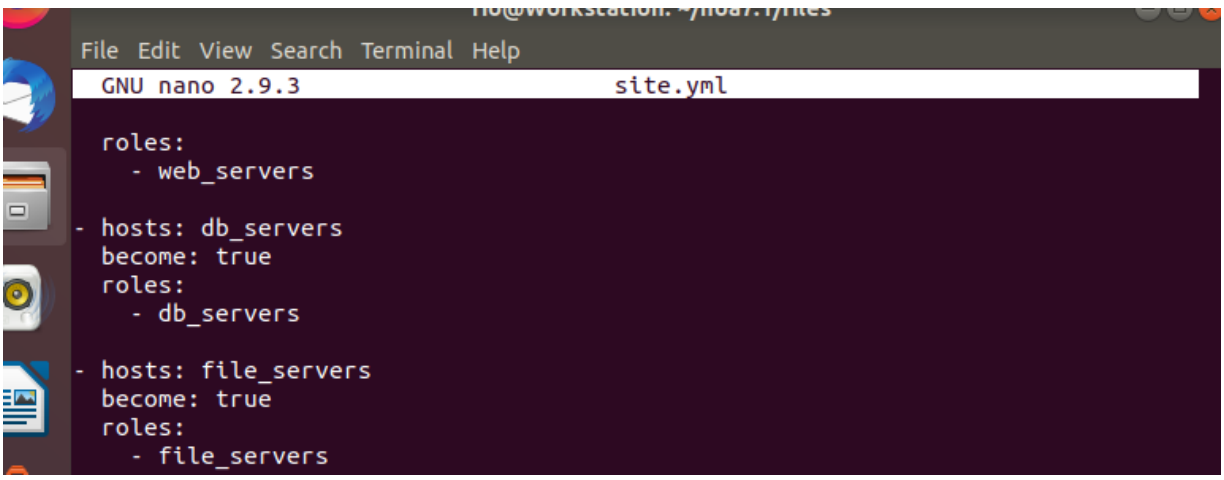
Save the file and exit.

Input:


File Edit View Search Terminal Help
GNU nano 2.9.3 site.yml

- hosts: all
become: true
pre_tasks:
- name: update repository index (CentOS)
tags: always
dnf:
update_cache: yes
changed_when: false
when: ansible_distribution == "CentOS"
- name: install updates (Ubuntu)
tags: always
apt:
update_cache: yes
changed_when: false
when: ansible_distribution == "Ubuntu"
- hosts: all
become: true
roles:
- base
[Read 48 lines]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell





2. Under the same directory, create a new directory and name it roles. Enter the roles directory and create new directories: base, web_servers, file_servers, db_servers and workstations. For each directory, create a directory and name it tasks.

Input:

```
rio@Workstation:~/hoa7.1/files$ mkdir roles
rio@Workstation:~/hoa7.1/files$ mkdir roles
rio@Workstation:~/hoa7.1/files$ ls
ansible.cfg  default_site.html  inventory  roles  site.yml
rio@Workstation:~/hoa7.1/files$ cd roles
0 directories, 0 files
rio@Workstation:~/hoa7.1/files/roles$ mkdir base
rio@Workstation:~/hoa7.1/files/roles$ mkdir db_servers
rio@Workstation:~/hoa7.1/files/roles$ mkdir file_servers
rio@Workstation:~/hoa7.1/files/roles$ mkdir web_servers
rio@Workstation:~/hoa7.1/files/roles$ mkdir workstations
rio@Workstation:~/hoa7.1/files/roles$
```

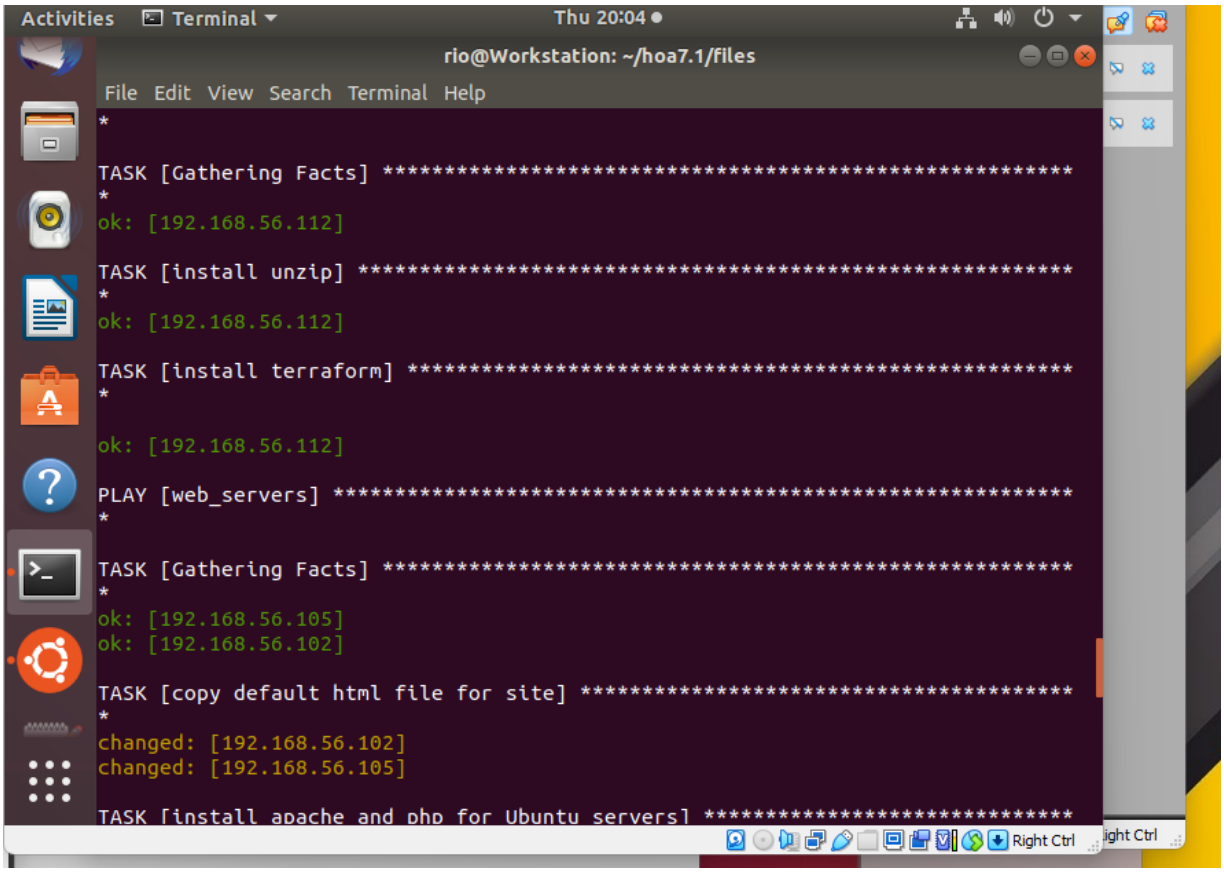
3. Go to tasks for all directory and create a file. Name it main.yml. In each of the tasks for all directories, copy and paste the code from the old site.yml file. Show all contents of main.yml files for all tasks.

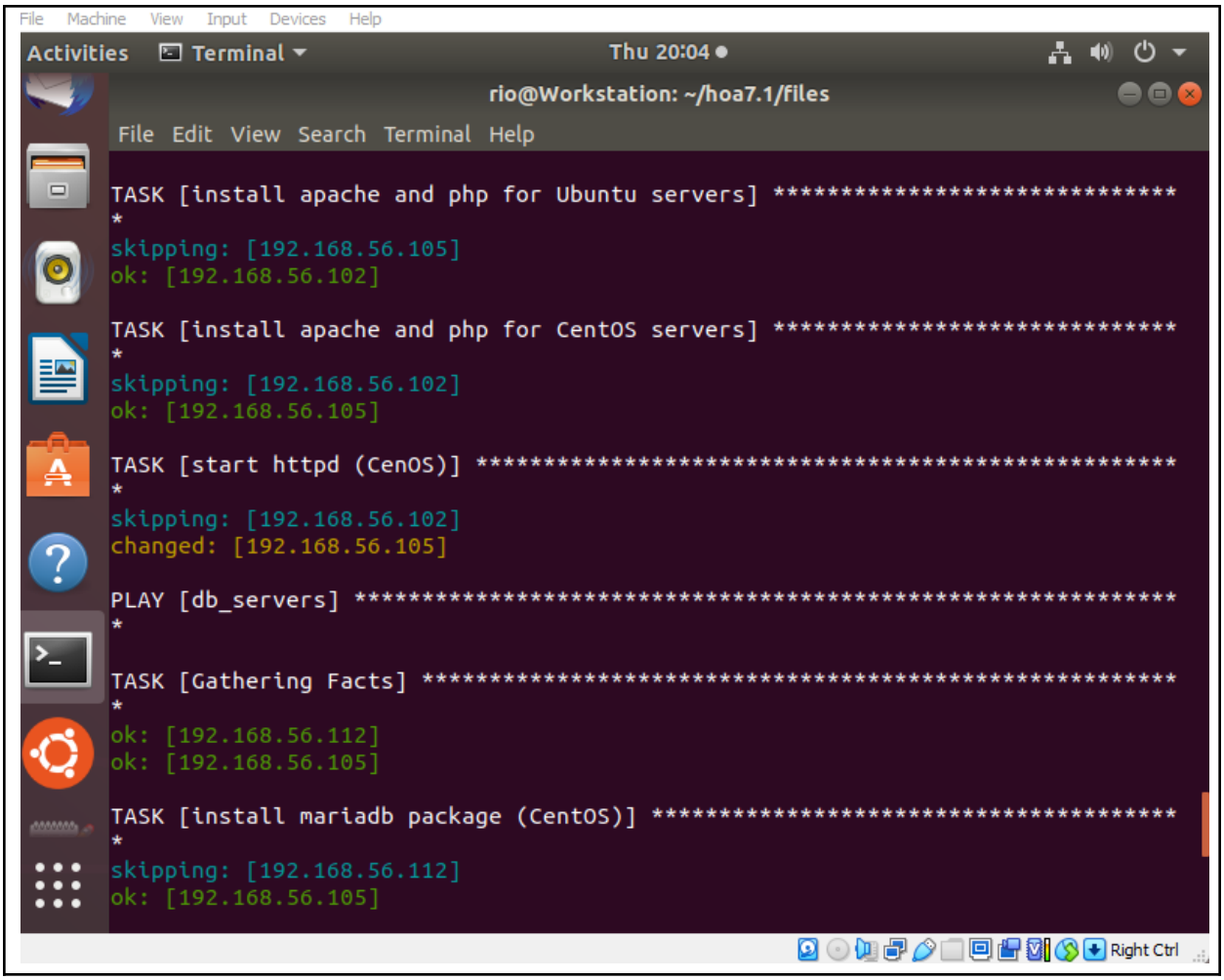
```
rio@Workstation:~/hoa7.1/files/roles$ tree
.
├── base
│   └── tasks
│       └── main.yml
├── db_servers
│   └── tasks
│       └── main.yml
├── file_servers
│   └── tasks
│       └── main.yml
├── web_servers
│   └── tasks
│       └── main.yml
└── workstations
    └── tasks
        └── main.yml

10 directories, 5 files
rio@Workstation:~/hoa7.1/files/roles$
```

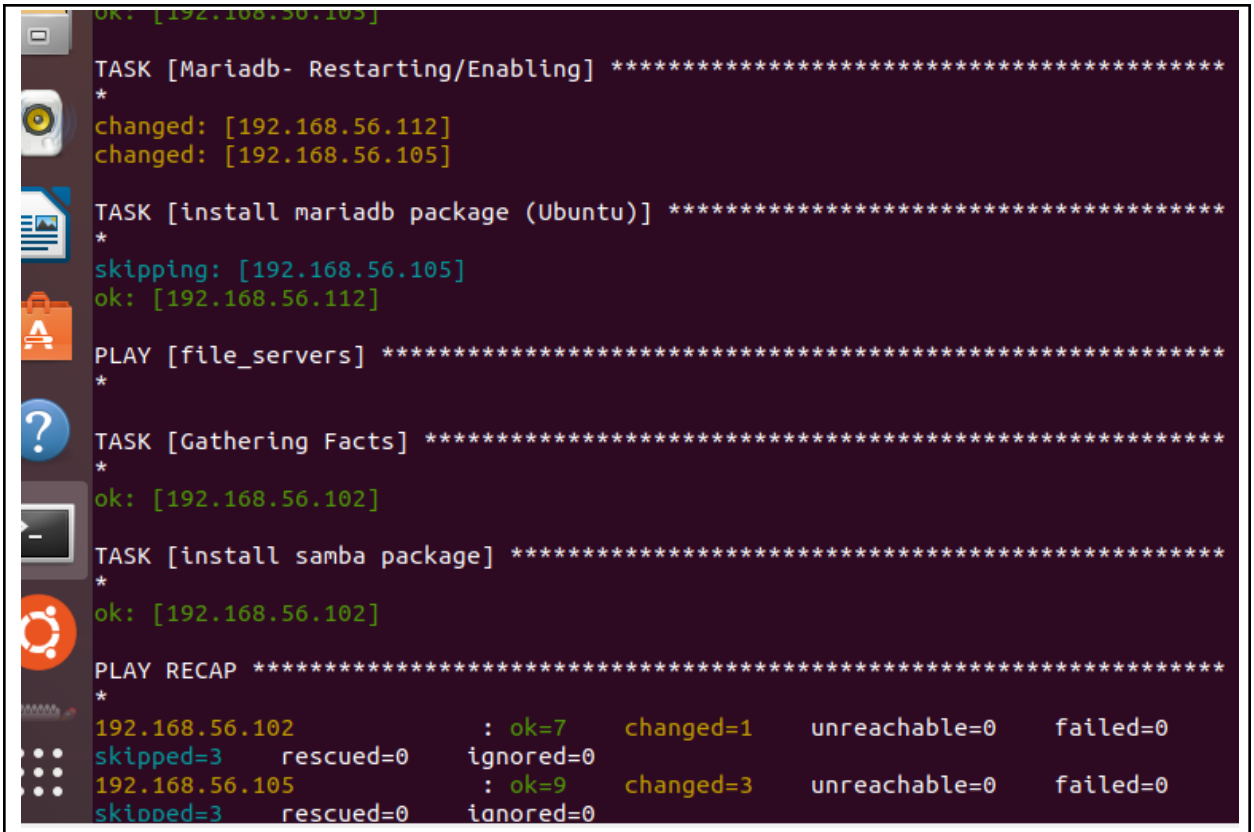
4. Run the site.yml playbook and describe the output.

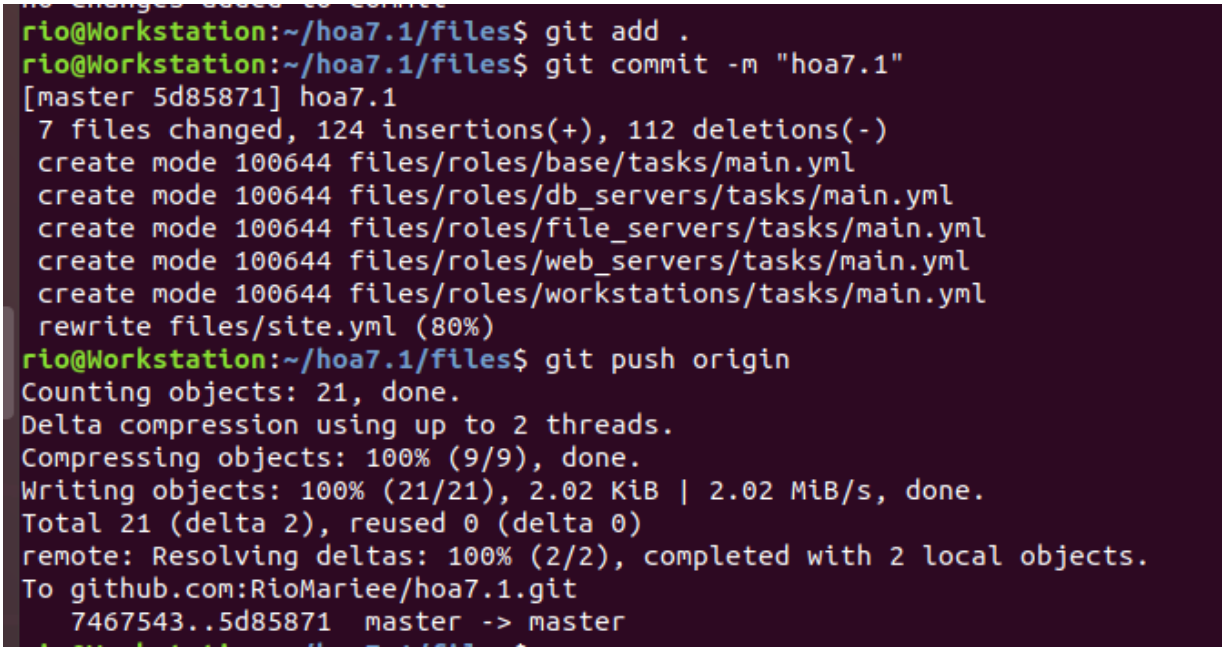
Output:

```
rio@Workstation:~/hoa7.1/files$ ansible-playbook --ask-become-pass site.yml
BECOME password:

PLAY [all] *********************************************************************
*

TASK [Gathering Facts] *********************************************************
*
ok: [192.168.56.112]
ok: [192.168.56.105]
ok: [192.168.56.102]

TASK [install updates (CentOS)] ************************************************
*
skipping: [192.168.56.102]
skipping: [192.168.56.112]
ok: [192.168.56.105]

TASK [install updates (Ubuntu)] ************************************************
*
skipping: [192.168.56.105]
ok: [192.168.56.112]
ok: [192.168.56.102]

PLAY [workstations] ************************************************************
*
```




File Machine View Input Devices Help
Activities
Terminal
Thu 20:04
rio@Workstation: ~/hoa7.1/files
File Edit View Search Terminal Help
TASK [install apache and php for Ubuntu servers] ********************************
*
skipping: [192.168.56.105]
ok: [192.168.56.102]
TASK [install apache and php for CentOS servers] ********************************
*
skipping: [192.168.56.102]
ok: [192.168.56.105]
TASK [start httpd (CenOS)] **
*
skipping: [192.168.56.102]
changed: [192.168.56.105]
PLAY [db_servers] ***
*
TASK [Gathering Facts] **
*
ok: [192.168.56.112]
ok: [192.168.56.105]
TASK [install mariadb package (CentOS)] ***
*
skipping: [192.168.56.112]
ok: [192.168.56.105]
Right Ctrl





Github repository link: https://github.com/RioMariee/hoa7.1.git

Reflections:

Answer the following:

1. What is the importance of creating roles?
    - Creating roles is important because it defines what different users can and cannot do on your computer. By creating roles, you control who has what kind of access and permissions on your computer. This helps maintain security, ensures that tasks are performed correctly, and prevents unauthorized changes that could cause problems. In simple terms, creating roles in Ubuntu helps you manage who can do what on your computer, which is crucial for keeping it safe and organized.

2. What is the importance of managing files?

    - Managing files is important because it is crucial to keep your computer tidy and helps you find what you need easily. When you put things in the right place, you can quickly locate them when you need them. It also prevents your computer from becoming messy, slowing down, or running out of space. So, managing files in Ubuntu helps you stay organized, find your stuff easily, and keeps your computer running smoothly.

Conclusion

To conclude, when we handle files on faraway computers and set up specific tasks in Ansible, it's like having two key ingredients for making automation and computer management work really well. Think of it as making sure the right instructions and data go where they should, so everything runs smoothly without hiccups. On top of that, if we organize these tasks neatly, like putting them into labeled boxes, it makes it easier to reuse them later. This simplifies the job of keeping our computer systems organized and flexible, even when they are complex. When we put these practices together, they give organizations the ability to manage their computers effectively, boosting productivity, reducing mistakes, and adapting quickly to changes. Ansible's skills in handling files and tasks are crucial for building strong, automated systems that make managing computers easier and more adaptable in today's fast-changing tech world

Assessment Rubrics

Hands-On Rubirc (3)

Criteria	Ratings					Pts
Completeness This criterion specifies the analysis of the student of the task given.	**5 pts** **Excellent** Components of all tasks are present in the documentation and execution.	**4 pts** **Good** Components of most of the tasks are present in the documentation and execution.	**3 pts** **Ok** Components of half of the tasks are present in documentation and execution.	**2 pts** **Poor** Components some tasks are present in documentation and execution	**1 pts** **Bad** Components of all tasks lacks data in documentation and execution.	5 pts
Design This criterion measures the components and engineering of the Hands-on activity.	**5 pts** **Execellent** Design is robust and acceptable in the industry	**4 pts** **Good** Design is acceptable in the industry but can be improved.	**3 pts** **Ok** Design is a satisfactory level in the industry.	**2 pts** **Poor** Design is poorly architected and engineered needs improvement.	**1 pts** **Bad** Design is badly architectured and engineered needs revisiting and rework.	5 pts
Documentation This criterion measures the context and completeness of artifacts of the activity.	**5 pts** **Excellent** The context of documentation is precise and understandable to readers.	**4 pts** **Good** The context of documentation is acceptable for readers.	**3 pts** **Ok** The documentation is satisfactory, has the main components needed, and grammar is acceptable.	**2 pts** **Poor** The documentation needs grammar checks but the content is complete.	**1 pts** **Bad** Documentation needs revisions from grammar to contexts.	5 pts
						Total Points: 15