Name: Gabiano, Chris leonard A.	**Date Performed: 10/05/2023**
Course/Section: CPE31s6	**Date Submitted: 10/05/2023**
Instructor: Eng. Jonathan Taylar	**Semester and SY: 2023 - 24**

Activity 7: Managing Files and Creating Roles in Ansible

1. **Objectives:**

1.1 Manage files in remote servers

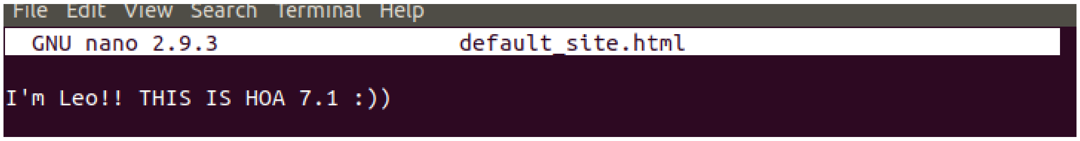
1.2 Implement roles in ansible

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

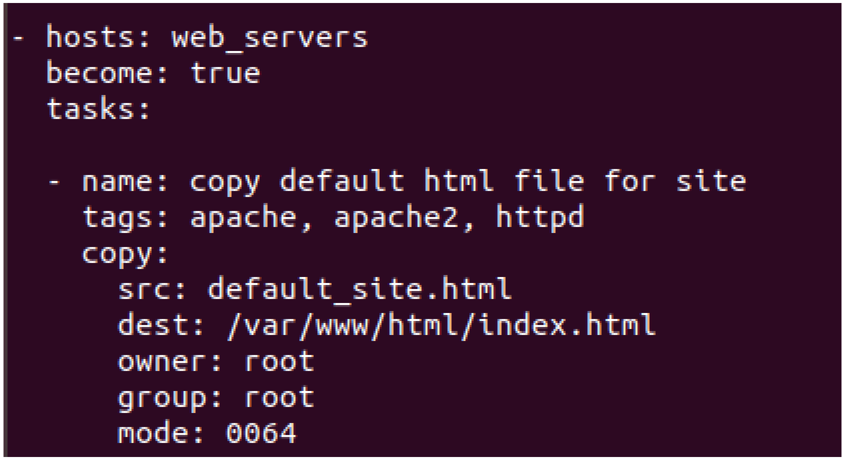
1. Using the previous directory we created, create a directory, and named it "*files*." Create a file inside that directory and name it "*default_site.html*." Edit the file and put basic HTML syntax. Any content will do, as long as it will display text later. Save the file and exit.



```
  GNU nano 2.9.3                    default_site.html

I'm Leo!! THIS IS HOA 7.1 :))
```

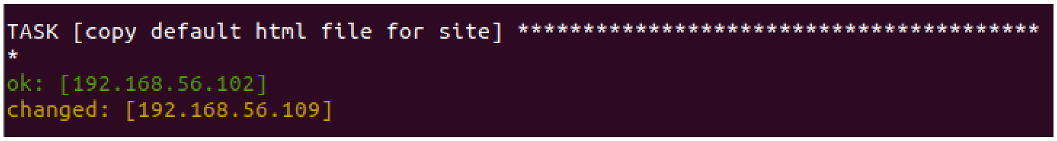
2. Edit the *site.yml* file and just below the *web_servers* play, create a new file to copy the default html file for site:

```
- 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
```



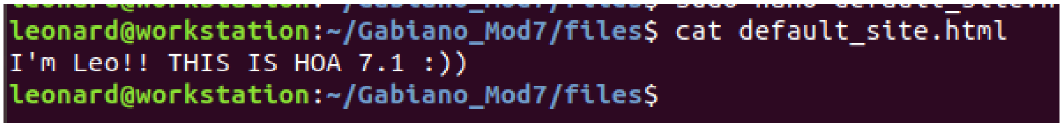
```
- hosts: web_servers
  become: true
  tasks:

  - 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: 0064
```

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

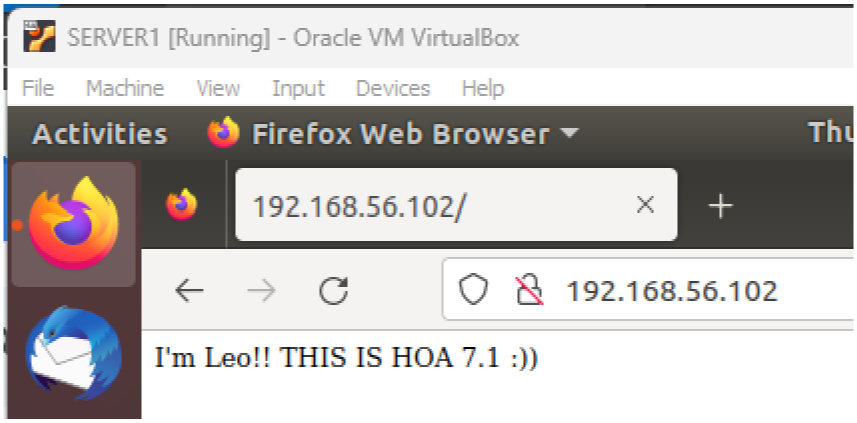


```
TASK [copy default html file for site] ****************************************
*
ok: [192.168.56.102]
changed: [192.168.56.109]
```

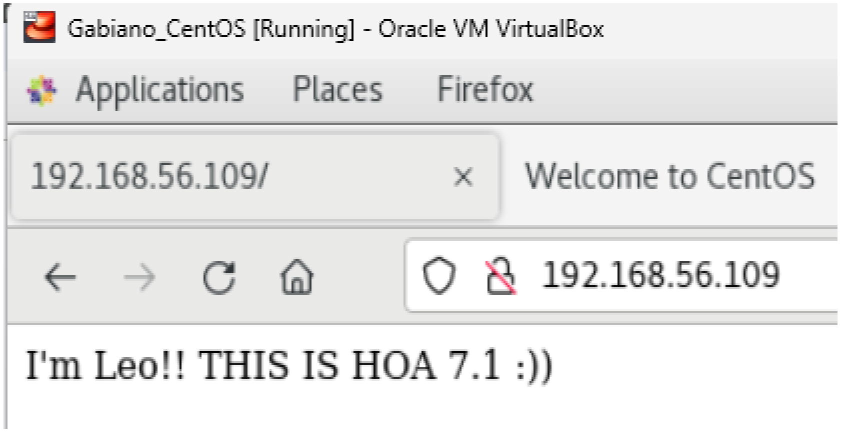
4. Go to the remote servers (*web_servers*) listed in your inventory. Use cat command to check if the index.html is the same as the local repository file (*default_site.html*). Do both for Ubuntu and CentOS servers. On the CentOS server, go to the browser and type its IP address. Describe the output.



```
leonard@workstation:~/Gabiano_Mod7/files$ cat default_site.html
I'm Leo!! THIS IS HOA 7.1 :))
leonard@workstation:~/Gabiano_Mod7/files$
```

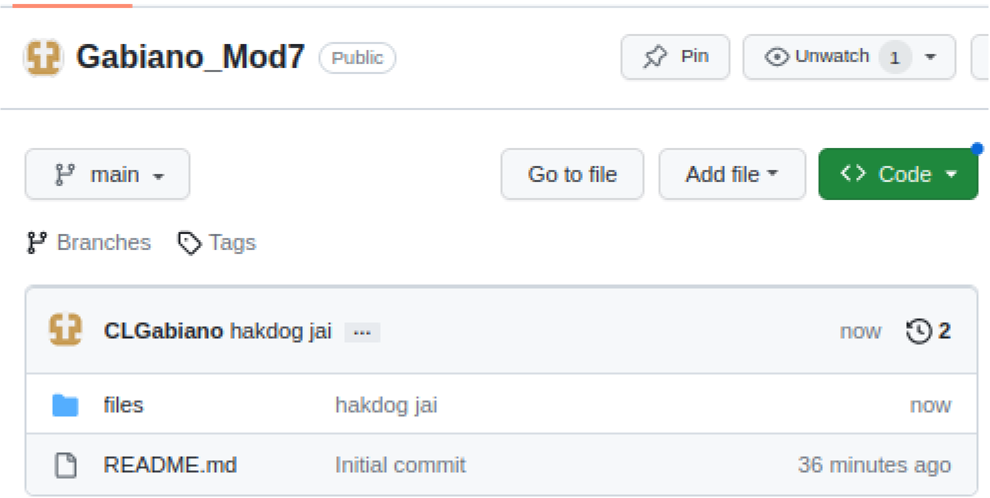


I'm Leo!! THIS IS HOA 7.1 :))



I'm Leo!! THIS IS HOA 7.1 :))

5. Sync your local repository with GitHub and describe the changes.



CLGabiano hakdog jai ··· now 2

files	hakdog jai	now
README.md	Initial commit	36 minutes ago

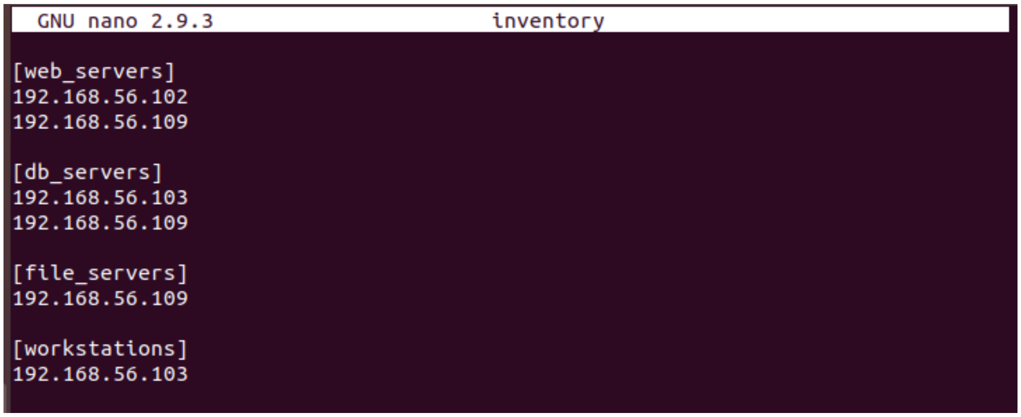
Task 2: Download a file and extract it to a remote server

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
```

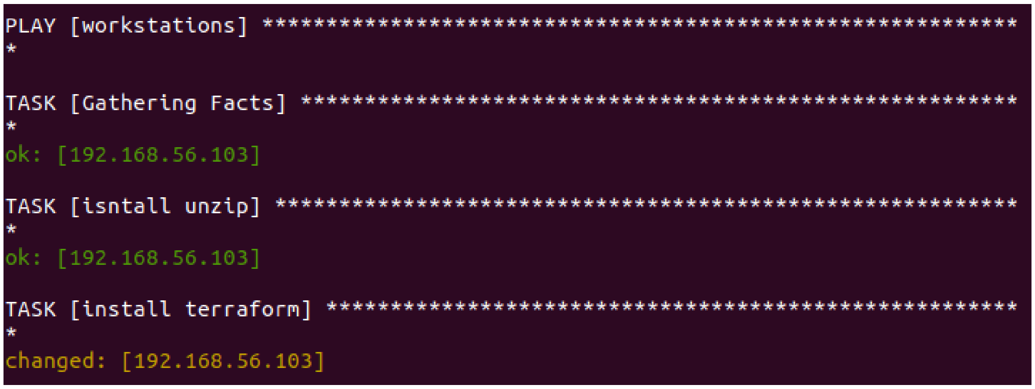
```
dest: /usr/local/bin
remote_src: yes
mode: 0755
owner: root
group: root
```

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

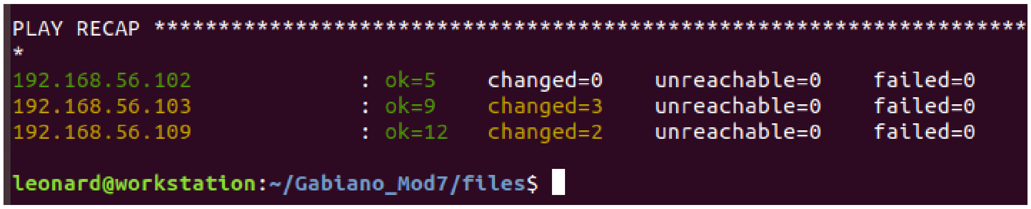


```
  GNU nano 2.9.3                          inventory

[web_servers]
192.168.56.102
192.168.56.109

[db_servers]
192.168.56.103
192.168.56.109

[file_servers]
192.168.56.109

[workstations]
192.168.56.103
```

3. Run the playbook. Describe the output.

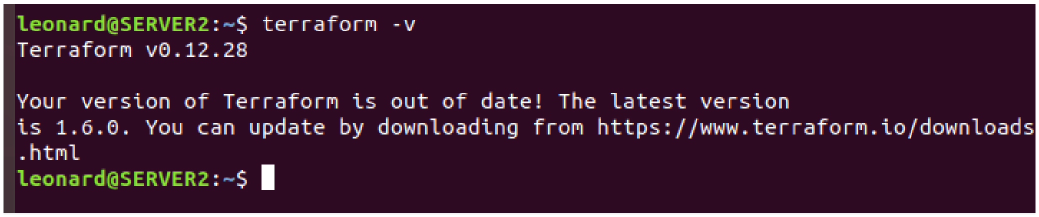


```
PLAY [workstations] ***********************************************************
*

TASK [Gathering Facts] ********************************************************
*
ok: [192.168.56.103]

TASK [isntall unzip] **********************************************************
*
ok: [192.168.56.103]

TASK [install terraform] ******************************************************
*
changed: [192.168.56.103]
```



```
PLAY RECAP ********************************************************************
*
192.168.56.102             : ok=5    changed=0    unreachable=0    failed=0
192.168.56.103             : ok=9    changed=3    unreachable=0    failed=0
192.168.56.109             : ok=12   changed=2    unreachable=0    failed=0

leonard@workstation:~/Gabiano_Mod7/files$
```

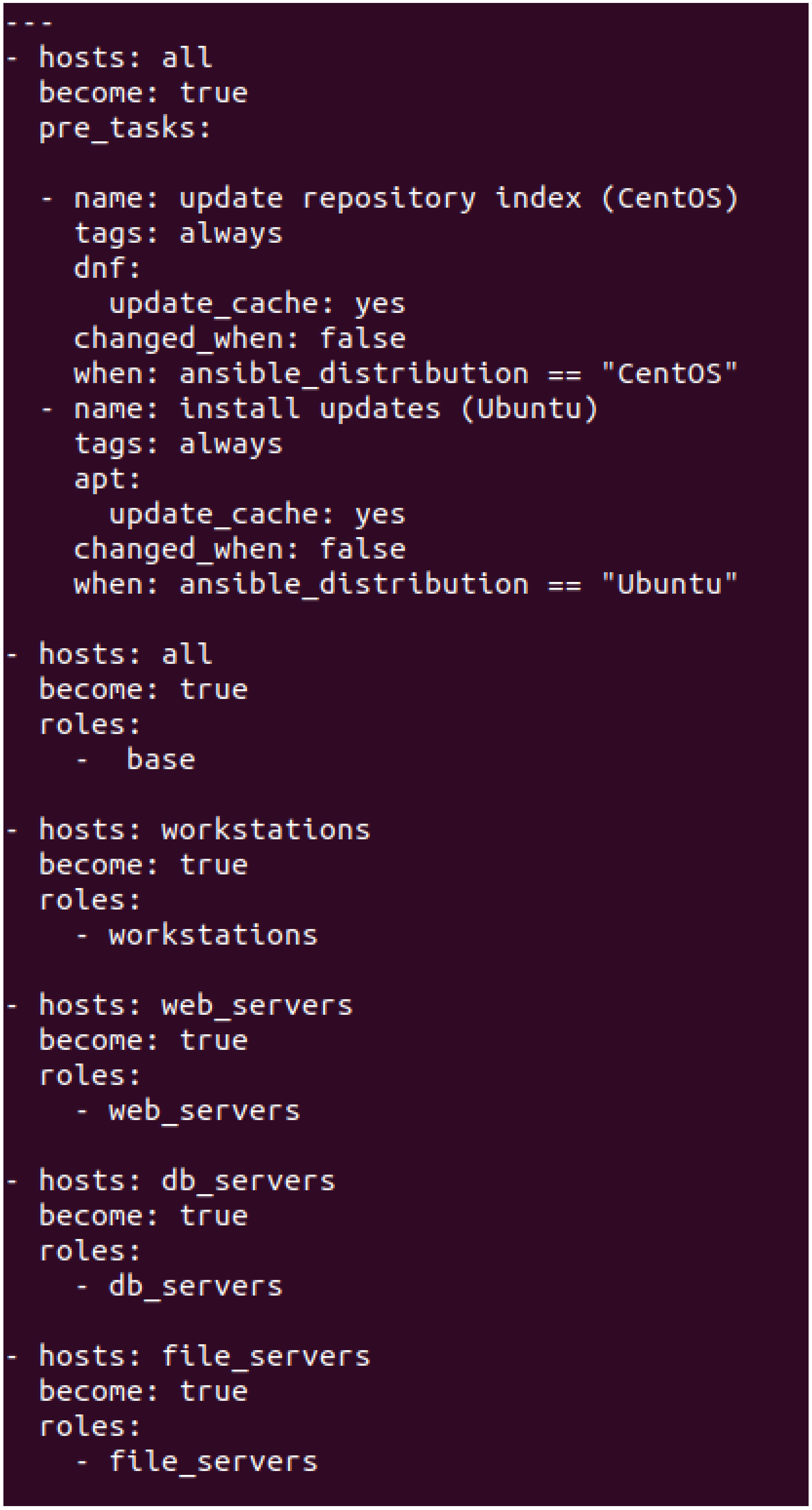
4. On the Ubuntu remote workstation, type terraform to verify installation of terraform. Describe the output.



```
leonard@SERVER2:~$ terraform -v
Terraform v0.12.28

Your version of Terraform is out of date! The latest version
is 1.6.0. You can update by downloading from https://www.terraform.io/downloads
.html
leonard@SERVER2:~$
```

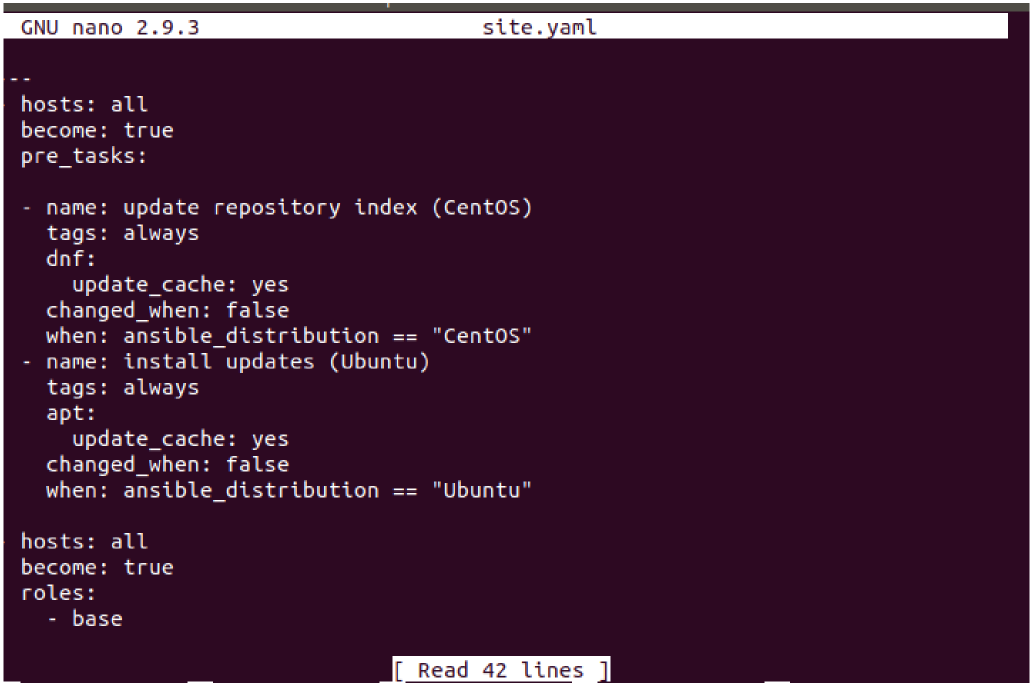
Task 3: Create roles

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)

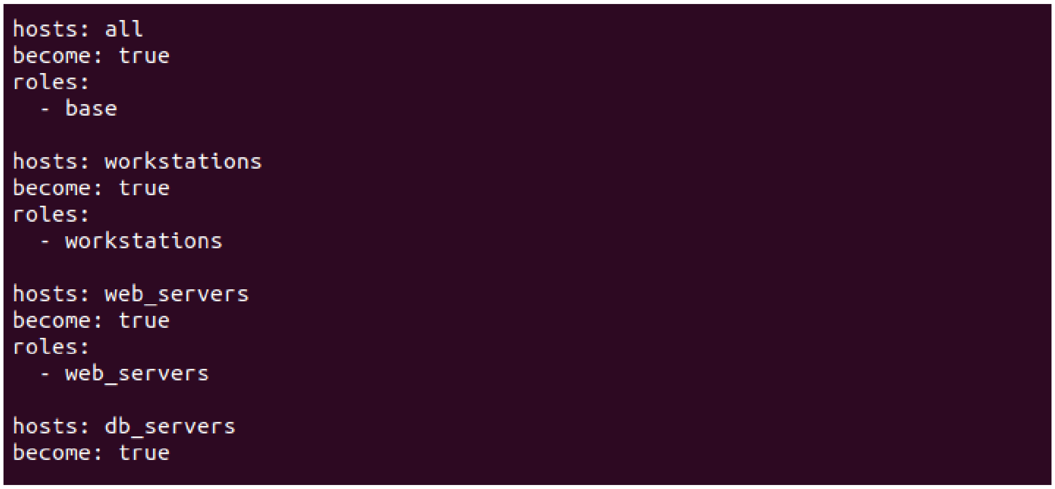


```
---
- hosts: all
  become: true
  pre_tasks:

  - 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.

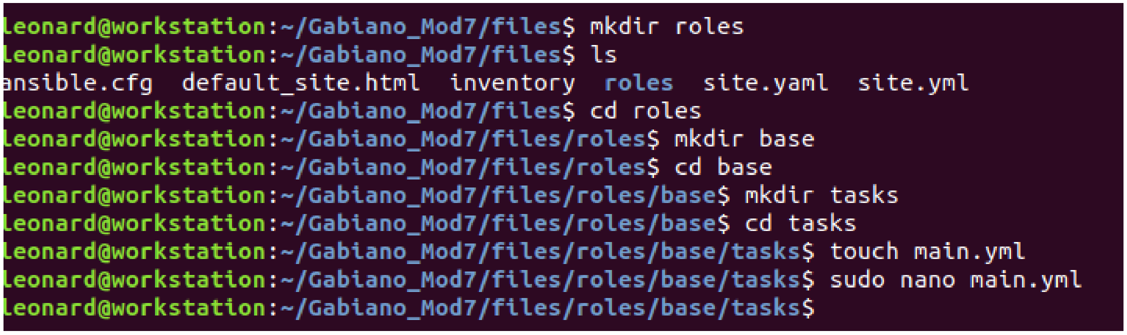


```
 GNU nano 2.9.3                       site.yaml

--
 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 42 lines ]
```



```
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
```

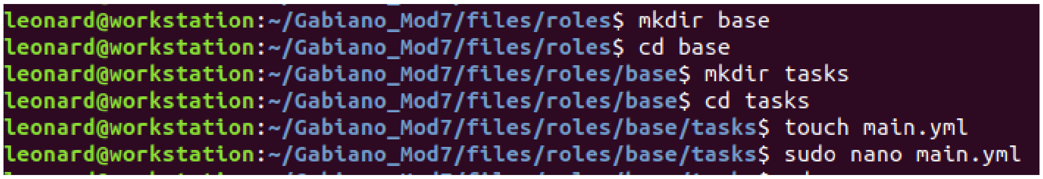
2. Under the same directory, create a new directory and name it roles. Enter the roles directory and create new directories: base, web_servers, file_servers, db_servers and workstations. For each directory, create a directory and name it tasks.



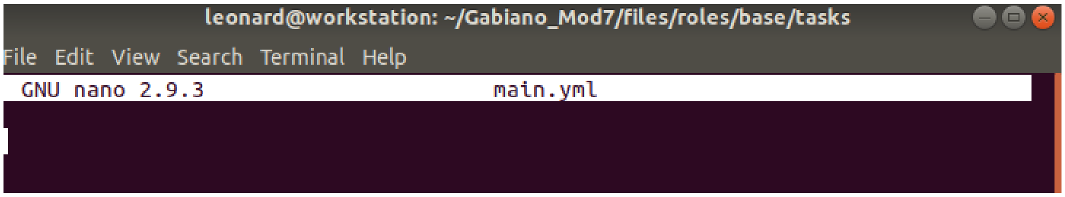
```
leonard@workstation:~/Gabiano_Mod7/files$ mkdir roles
leonard@workstation:~/Gabiano_Mod7/files$ ls
ansible.cfg  default_site.html  inventory  roles  site.yaml  site.yml
leonard@workstation:~/Gabiano_Mod7/files$ cd roles
leonard@workstation:~/Gabiano_Mod7/files/roles$ mkdir base
leonard@workstation:~/Gabiano_Mod7/files/roles$ cd base
leonard@workstation:~/Gabiano_Mod7/files/roles/base$ mkdir tasks
leonard@workstation:~/Gabiano_Mod7/files/roles/base$ cd tasks
leonard@workstation:~/Gabiano_Mod7/files/roles/base/tasks$ touch main.yml
leonard@workstation:~/Gabiano_Mod7/files/roles/base/tasks$ sudo nano main.yml
leonard@workstation:~/Gabiano_Mod7/files/roles/base/tasks$
```

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.

base:



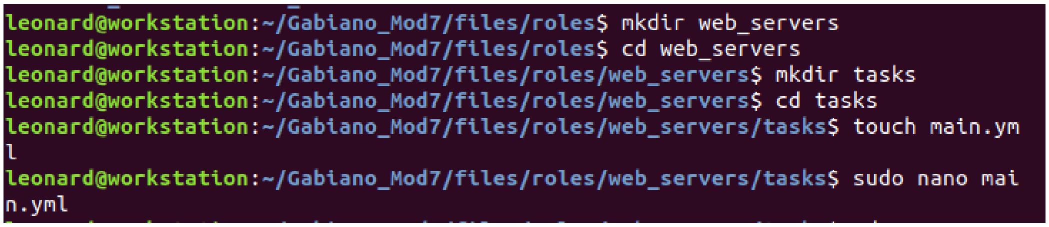
```
leonard@workstation:~/Gabiano_Mod7/files/roles$ mkdir base
leonard@workstation:~/Gabiano_Mod7/files/roles$ cd base
leonard@workstation:~/Gabiano_Mod7/files/roles/base$ mkdir tasks
leonard@workstation:~/Gabiano_Mod7/files/roles/base$ cd tasks
leonard@workstation:~/Gabiano_Mod7/files/roles/base/tasks$ touch main.yml
leonard@workstation:~/Gabiano_Mod7/files/roles/base/tasks$ sudo nano main.yml
```



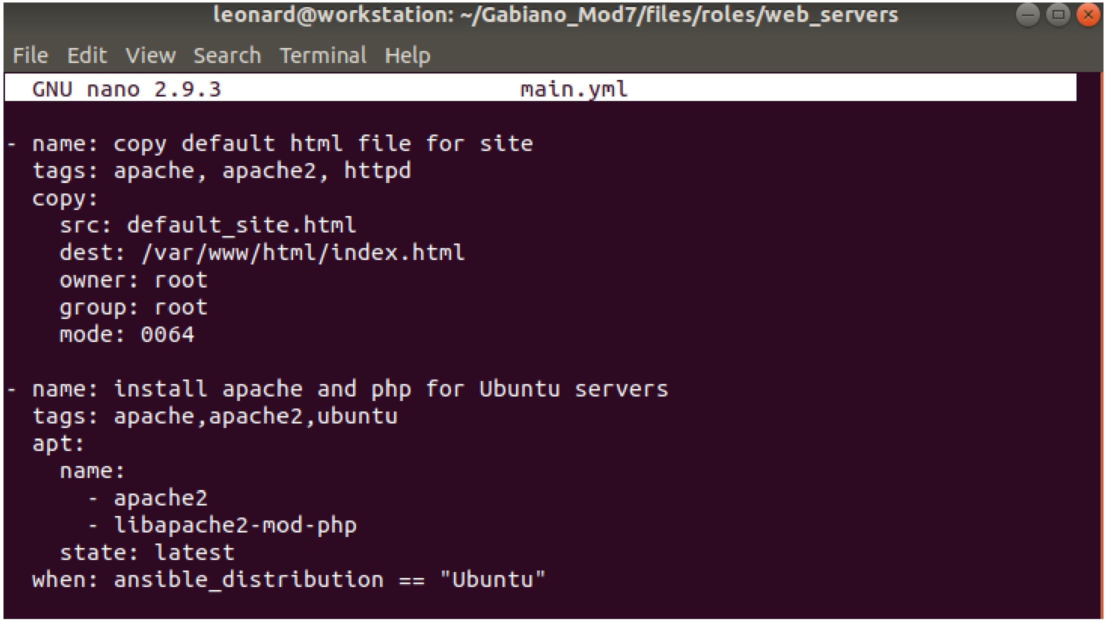
leonard@workstation: ~/Gabiano_Mod7/files/roles/base/tasks
File Edit View Search Terminal Help

```
 GNU nano 2.9.3                          main.yml
```

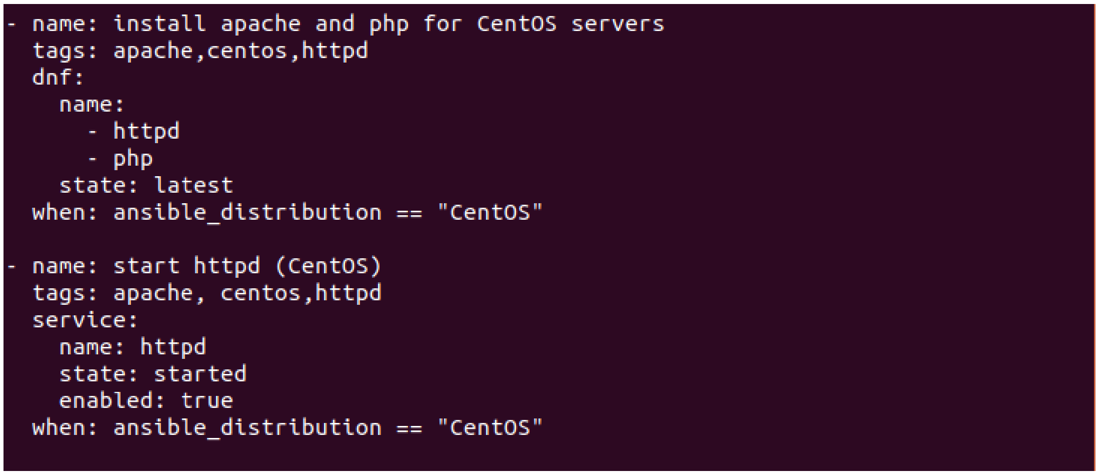
web_servers:



```
leonard@workstation:~/Gabiano_Mod7/files/roles$ mkdir web_servers
leonard@workstation:~/Gabiano_Mod7/files/roles$ cd web_servers
leonard@workstation:~/Gabiano_Mod7/files/roles/web_servers$ mkdir tasks
leonard@workstation:~/Gabiano_Mod7/files/roles/web_servers$ cd tasks
leonard@workstation:~/Gabiano_Mod7/files/roles/web_servers/tasks$ touch main.ym
l
leonard@workstation:~/Gabiano_Mod7/files/roles/web_servers/tasks$ sudo nano mai
n.yml
```

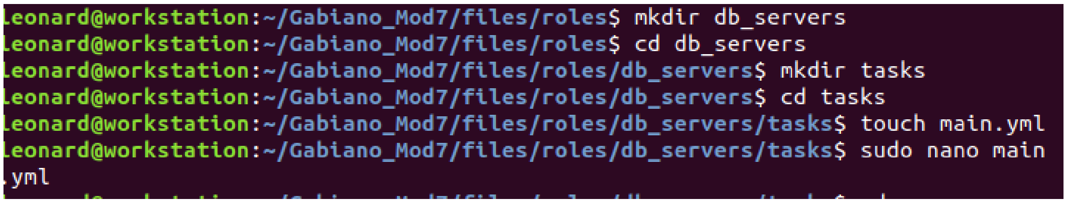

leonard@workstation: ~/Gabiano_Mod7/files/roles/web_servers
File Edit View Search Terminal Help

```
  GNU nano 2.9.3                         main.yml

- 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: 0064

- name: install apache and php for Ubuntu servers
  tags: apache,apache2,ubuntu
  apt:
    name:
      - apache2
      - libapache2-mod-php
    state: latest
  when: ansible_distribution == "Ubuntu"
```

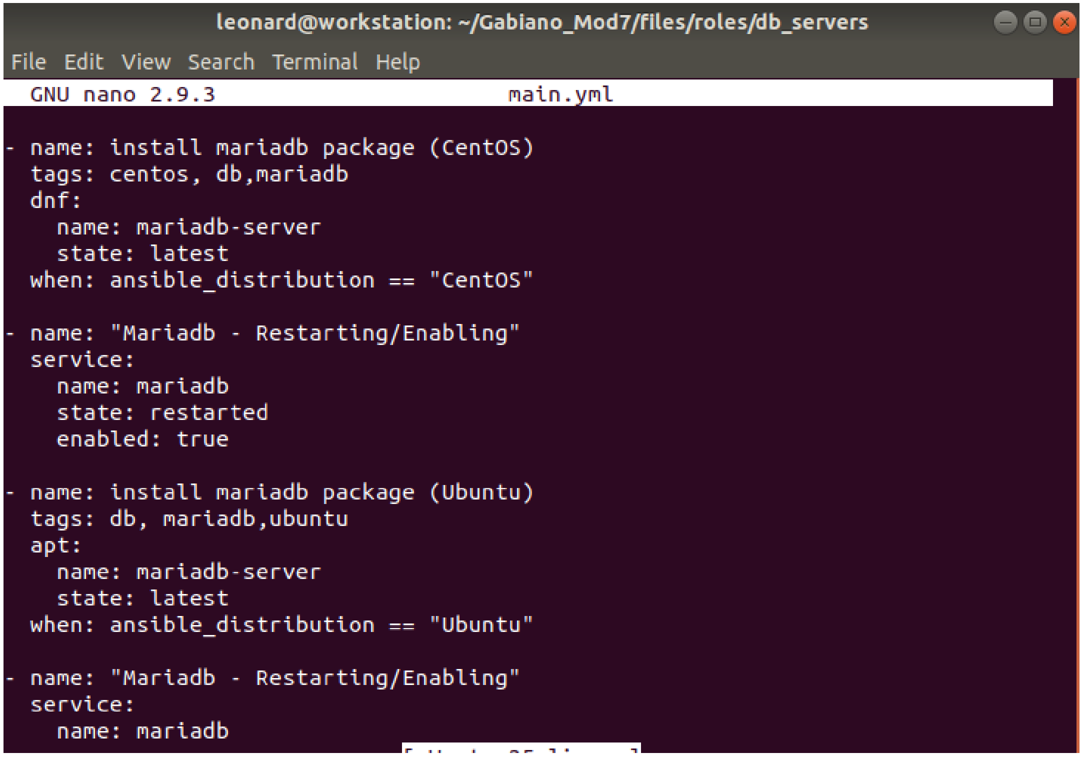


```
- name: install apache and php for CentOS servers
  tags: apache,centos,httpd
  dnf:
    name:
      - httpd
      - php
    state: latest
  when: ansible_distribution == "CentOS"

- name: start httpd (CentOS)
  tags: apache, centos,httpd
  service:
    name: httpd
    state: started
    enabled: true
  when: ansible_distribution == "CentOS"
```

db_servers:

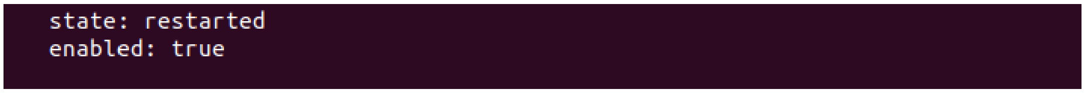


```
leonard@workstation:~/Gabiano_Mod7/files/roles$ mkdir db_servers
leonard@workstation:~/Gabiano_Mod7/files/roles$ cd db_servers
leonard@workstation:~/Gabiano_Mod7/files/roles/db_servers$ mkdir tasks
leonard@workstation:~/Gabiano_Mod7/files/roles/db_servers$ cd tasks
leonard@workstation:~/Gabiano_Mod7/files/roles/db_servers/tasks$ touch main.yml
leonard@workstation:~/Gabiano_Mod7/files/roles/db_servers/tasks$ sudo nano main
.yml
```



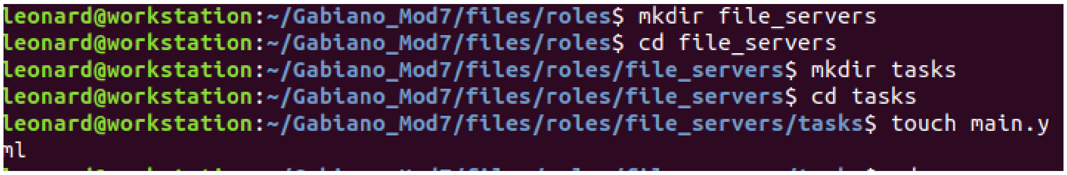
leonard@workstation: ~/Gabiano_Mod7/files/roles/db_servers
File Edit View Search Terminal Help

```
  GNU nano 2.9.3                           main.yml

- name: install mariadb package (CentOS)
  tags: centos, db,mariadb
  dnf:
    name: mariadb-server
    state: latest
  when: ansible_distribution == "CentOS"

- name: "Mariadb - Restarting/Enabling"
  service:
    name: mariadb
    state: restarted
    enabled: true

- name: install mariadb package (Ubuntu)
  tags: db, mariadb,ubuntu
  apt:
    name: mariadb-server
    state: latest
  when: ansible_distribution == "Ubuntu"

- name: "Mariadb - Restarting/Enabling"
  service:
    name: mariadb
```

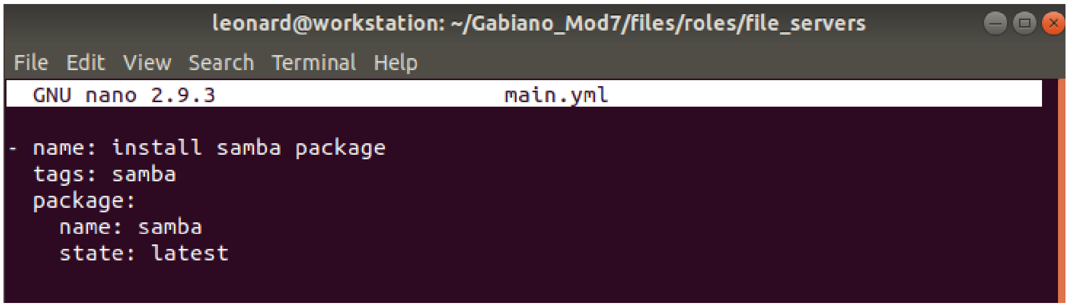


```
    state: restarted
    enabled: true
```

file_servers:



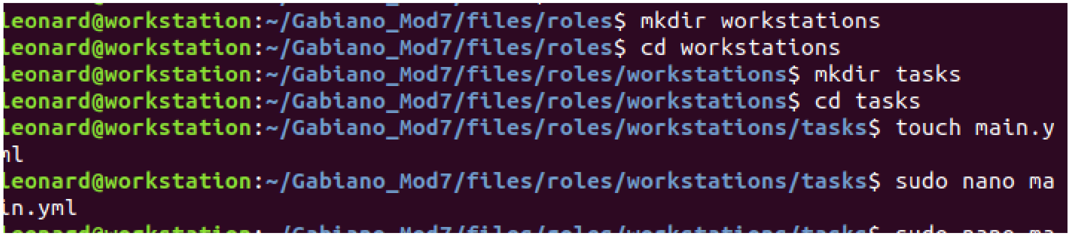
```
leonard@workstation:~/Gabiano_Mod7/files/roles$ mkdir file_servers
leonard@workstation:~/Gabiano_Mod7/files/roles$ cd file_servers
leonard@workstation:~/Gabiano_Mod7/files/roles/file_servers$ mkdir tasks
leonard@workstation:~/Gabiano_Mod7/files/roles/file_servers$ cd tasks
leonard@workstation:~/Gabiano_Mod7/files/roles/file_servers/tasks$ touch main.y
ml
```



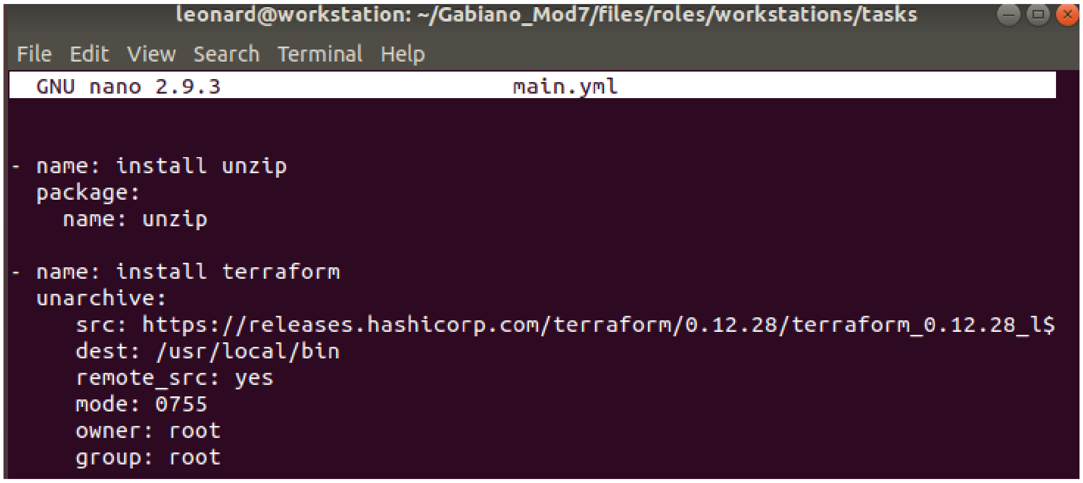
leonard@workstation: ~/Gabiano_Mod7/files/roles/file_servers
File Edit View Search Terminal Help

```
  GNU nano 2.9.3                           main.yml

- name: install samba package
  tags: samba
  package:
    name: samba
    state: latest
```

workstation:

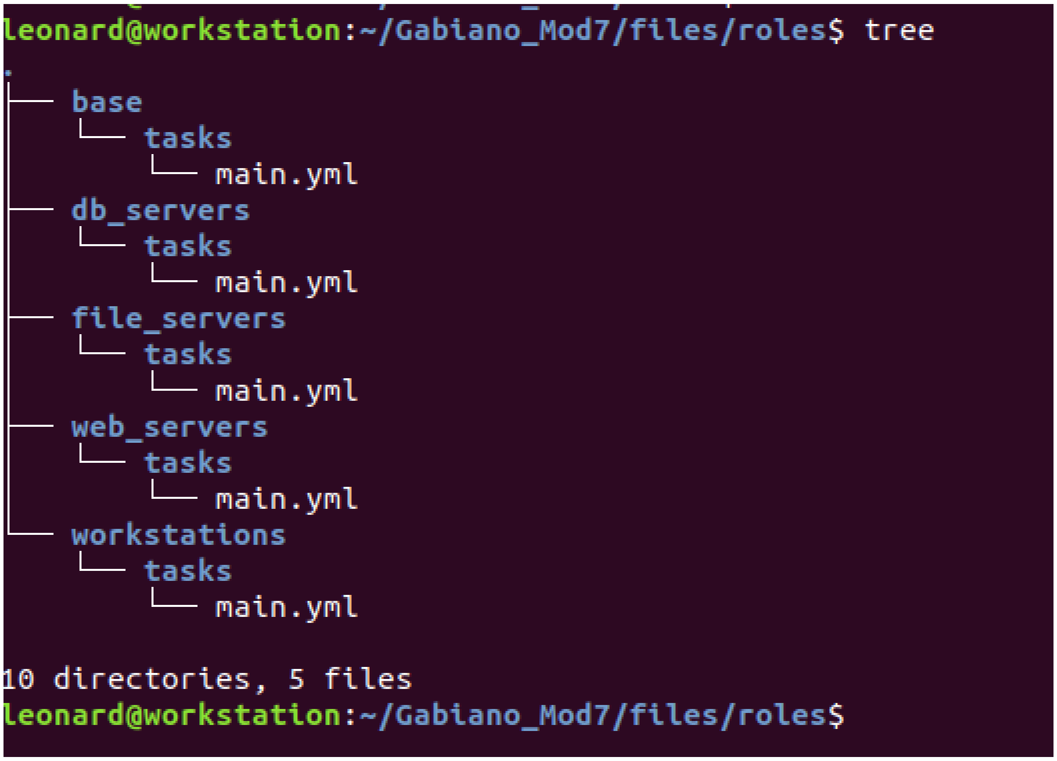


```
leonard@workstation:~/Gabiano_Mod7/files/roles$ mkdir workstations
leonard@workstation:~/Gabiano_Mod7/files/roles$ cd workstations
leonard@workstation:~/Gabiano_Mod7/files/roles/workstations$ mkdir tasks
leonard@workstation:~/Gabiano_Mod7/files/roles/workstations$ cd tasks
leonard@workstation:~/Gabiano_Mod7/files/roles/workstations/tasks$ touch main.y
ml
leonard@workstation:~/Gabiano_Mod7/files/roles/workstations/tasks$ sudo nano ma
in.yml
leonard@workstation:~/Gabiano_Mod7/files/roles/workstations/tasks$ sudo nano ma
```



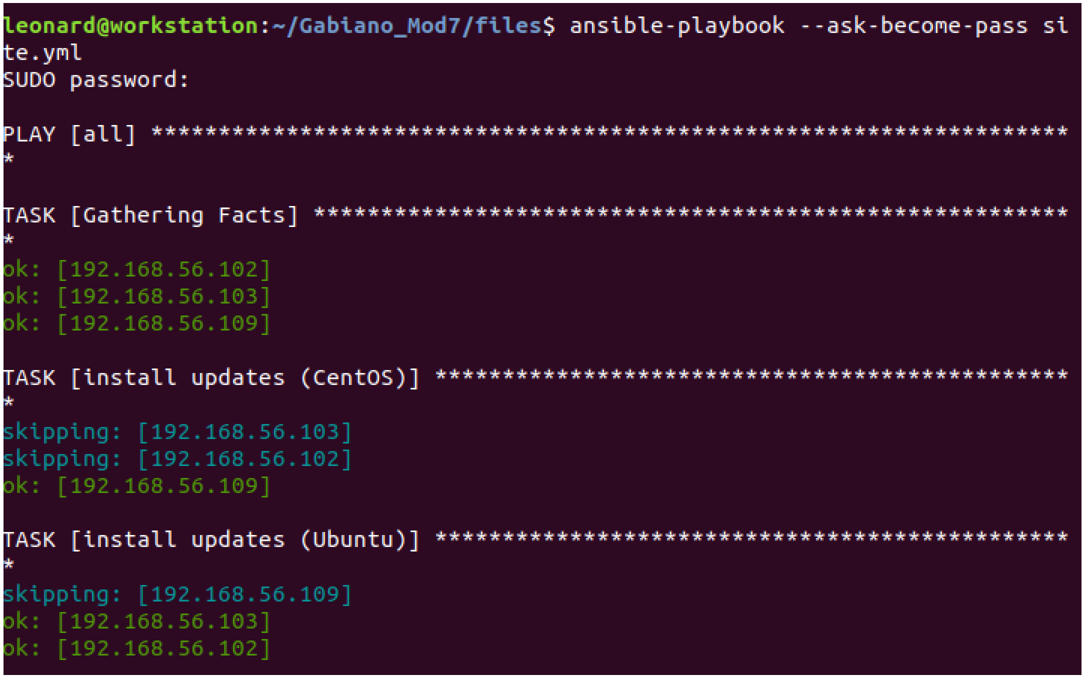
leonard@workstation: ~/Gabiano_Mod7/files/roles/workstations/tasks
File Edit View Search Terminal Help

```
  GNU nano 2.9.3                         main.yml


- name: install unzip
  package:
    name: unzip

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

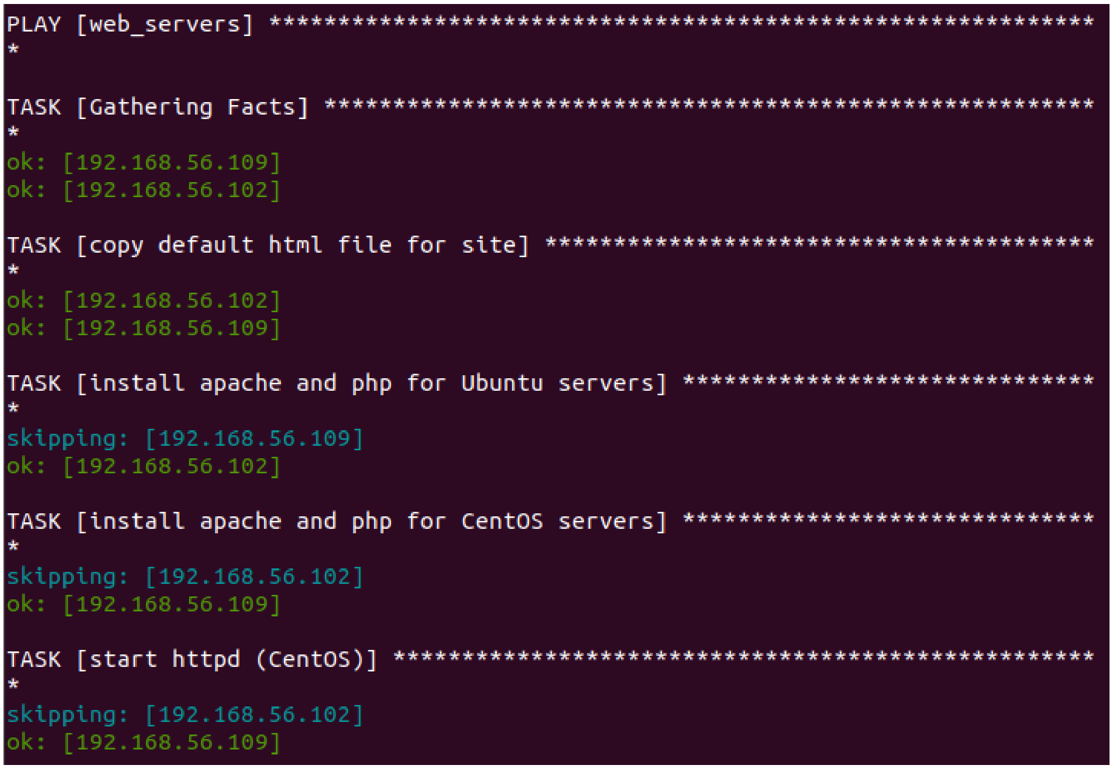


```
leonard@workstation:~/Gabiano_Mod7/files/roles$ tree
.
├── base
│   └── tasks
│       └── main.yml
├── db_servers
│   └── tasks
│       └── main.yml
├── file_servers
│   └── tasks
│       └── main.yml
├── web_servers
│   └── tasks
│       └── main.yml
└── workstations
    └── tasks
        └── main.yml

10 directories, 5 files
leonard@workstation:~/Gabiano_Mod7/files/roles$
```

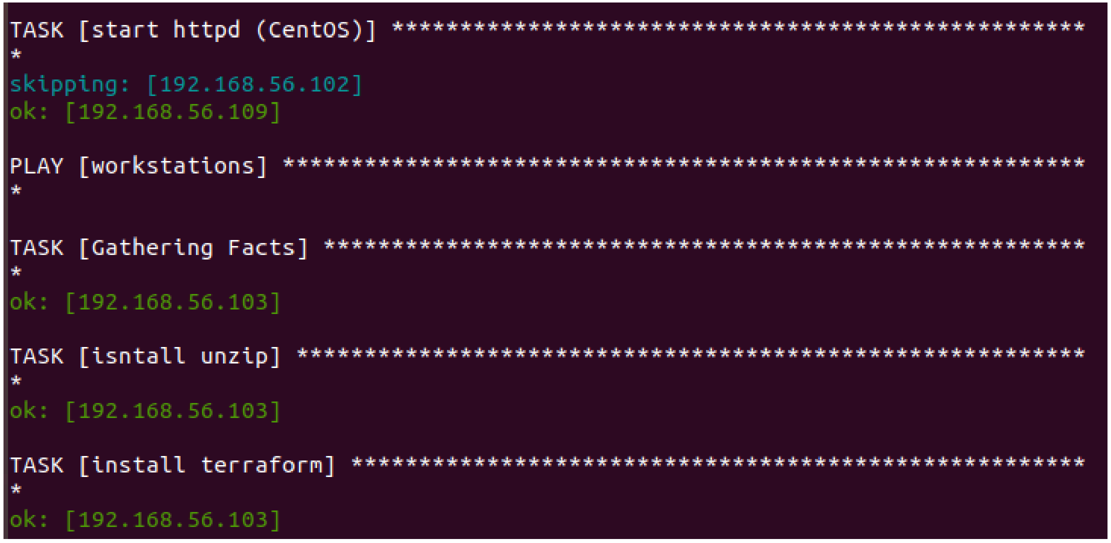
4. Run the site.yml playbook and describe the output.



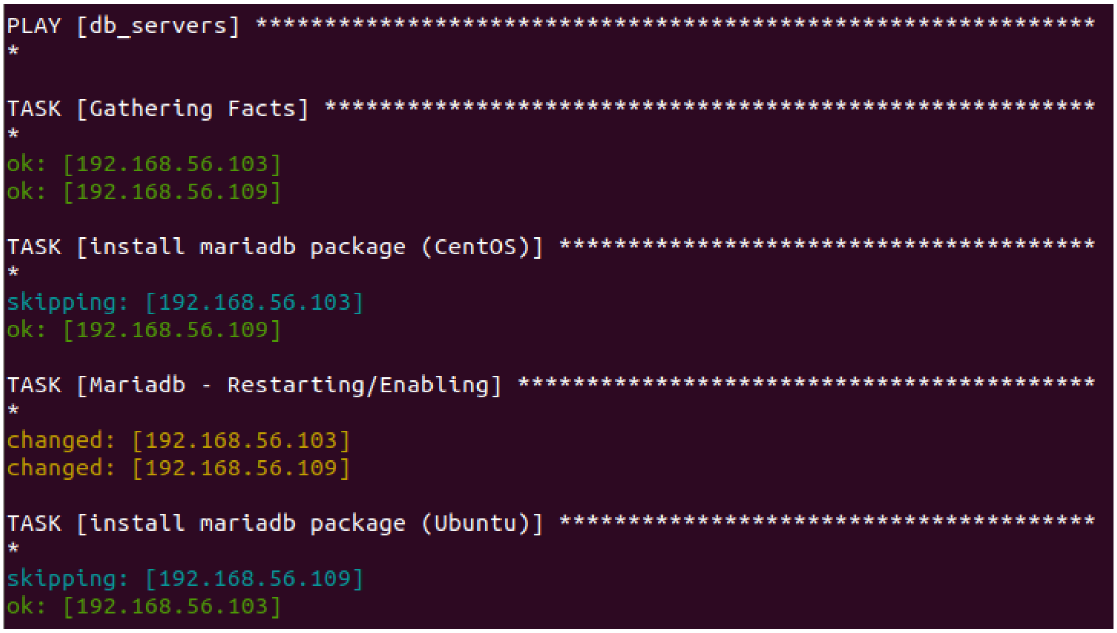
```
leonard@workstation:~/Gabiano_Mod7/files$ ansible-playbook --ask-become-pass si
te.yml
SUDO password:

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

TASK [Gathering Facts] *********************************************************
*
ok: [192.168.56.102]
ok: [192.168.56.103]
ok: [192.168.56.109]

TASK [install updates (CentOS)] ************************************************
*
skipping: [192.168.56.103]
skipping: [192.168.56.102]
ok: [192.168.56.109]

TASK [install updates (Ubuntu)] ************************************************
*
skipping: [192.168.56.109]
ok: [192.168.56.103]
ok: [192.168.56.102]
```



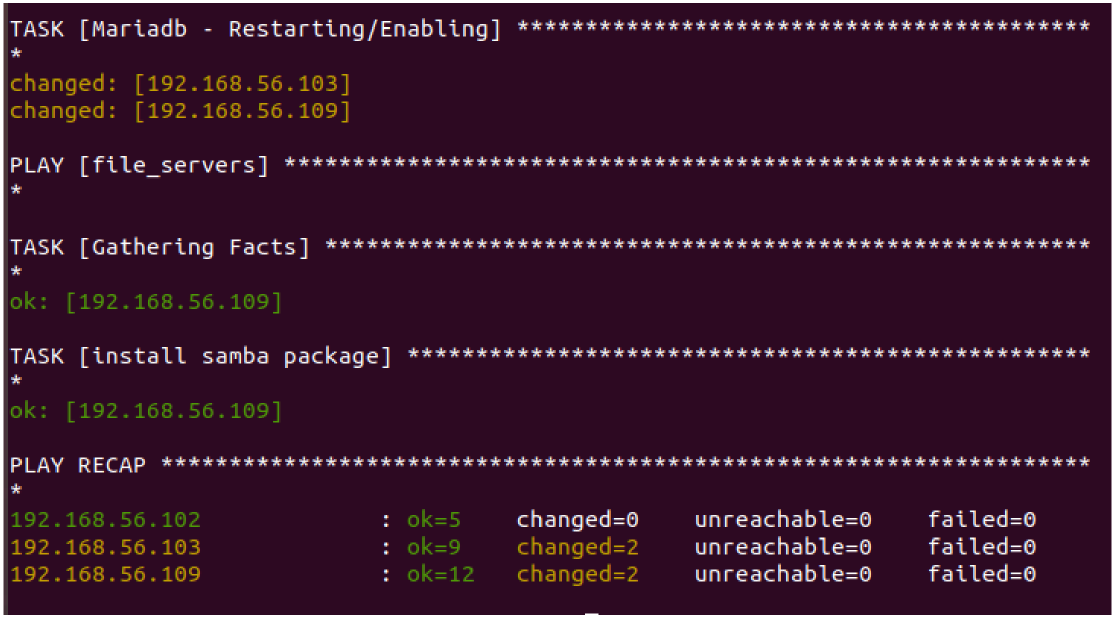
```
PLAY [web_servers] *************************************************************
*

TASK [Gathering Facts] *********************************************************
*
ok: [192.168.56.109]
ok: [192.168.56.102]

TASK [copy default html file for site] *****************************************
*
ok: [192.168.56.102]
ok: [192.168.56.109]

TASK [install apache and php for Ubuntu servers] *******************************
*
skipping: [192.168.56.109]
ok: [192.168.56.102]

TASK [install apache and php for CentOS servers] *******************************
*
skipping: [192.168.56.102]
ok: [192.168.56.109]

TASK [start httpd (CentOS)] ****************************************************
*
skipping: [192.168.56.102]
ok: [192.168.56.109]
```



```
TASK [start httpd (CentOS)] ****************************************************
*
skipping: [192.168.56.102]
ok: [192.168.56.109]

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

TASK [Gathering Facts] *********************************************************
*
ok: [192.168.56.103]

TASK [isntall unzip] ***********************************************************
*
ok: [192.168.56.103]

TASK [install terraform] *******************************************************
*
ok: [192.168.56.103]
```



```
PLAY [db_servers] **************************************************************
*

TASK [Gathering Facts] *********************************************************
*
ok: [192.168.56.103]
ok: [192.168.56.109]

TASK [install mariadb package (CentOS)] ****************************************
*
skipping: [192.168.56.103]
ok: [192.168.56.109]

TASK [Mariadb - Restarting/Enabling] *******************************************
*
changed: [192.168.56.103]
changed: [192.168.56.109]

TASK [install mariadb package (Ubuntu)] ****************************************
*
skipping: [192.168.56.109]
ok: [192.168.56.103]
```



```
TASK [Mariadb - Restarting/Enabling] ******************************************
*
changed: [192.168.56.103]
changed: [192.168.56.109]

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

TASK [Gathering Facts] ********************************************************
*
ok: [192.168.56.109]

TASK [install samba package] **************************************************
*
ok: [192.168.56.109]

PLAY RECAP ********************************************************************
*
192.168.56.102             : ok=5    changed=0    unreachable=0    failed=0
192.168.56.103             : ok=9    changed=2    unreachable=0    failed=0
192.168.56.109             : ok=12   changed=2    unreachable=0    failed=0
```

Reflections:

Answer the following:

1. What is the importance of creating roles?

   Creating roles in a system or organization is important because it helps define clear responsibilities and expectations for individuals or teams. Roles provide structure and organization, ensuring that tasks are allocated efficiently and that everyone understands their specific duties, which can lead to improved productivity, accountability, and overall effectiveness in achieving goals and objectives. Additionally, roles can facilitate specialization and expertise development, allowing individuals to focus on their strengths and contribute to the organization's success in a more strategic manner.

2. What is the importance of managing files?

   Managing files is essential for organizing, accessing, and maintaining information efficiently. Proper file management ensures data integrity, reduces clutter, aids in quick retrieval of information, and helps prevent data loss. It also facilitates collaboration and streamlines workflows, making it a fundamental aspect of digital organization and productivity.

conclusion:

The objectives of this activity were to learn how to manage files on remote servers and implement Ansible roles. Through the discussion and practical exercise, I gained a practical understanding of copying files to a server using Ansible, which is a valuable skill for tasks like customizing websites or configuring systems. Additionally, the incorporation of roles helped streamline and organize our Ansible plays for more efficient server management.