Name: Santos, Emmanuelle Dave G.	Date Performed: 10/03/25
Course/Section: CPE212/ CPE31S2	Date Submitted: 10/15/25
Instructor: Engr. Robin Valenzuela	Semester and SY: 1st sem 2025

Activity 7: Managing Files and Creating Roles in Ansible

1. **Objectives:**

1.1 Manage files in remote servers

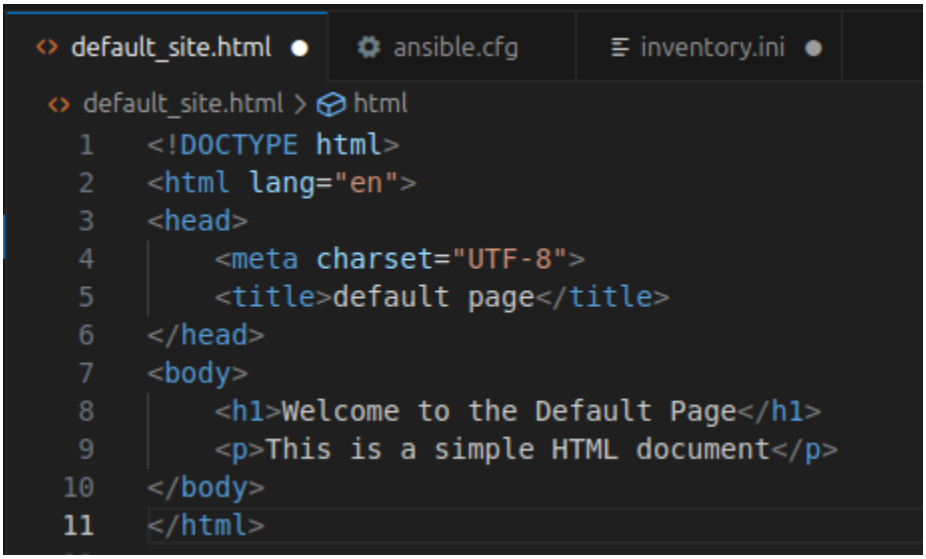
1.2 Implement roles in ansible

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.



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: 0644
```

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

- **It copies my default html to index.html**

```
chloe@Workstation:~/HOA7_Santos$ ansible-playbook --ask-become-pass site.yml
TASK [copy default html file for site] ***************************************************
changed: [192.168.56.102]
changed: [192.168.56.103]

TASK [install apache and php for Ubuntu servers] *****************************************
ok: [192.168.56.102]
ok: [192.168.56.103]

TASK [install apache and php for CentOS servers] *****************************************
skipping: [192.168.56.102]
skipping: [192.168.56.103]

TASK [start httpd service on CentOS] *****************************************************
skipping: [192.168.56.102]
skipping: [192.168.56.103]

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

TASK [Gathering Facts] *******************************************************************
ok: [192.168.56.101]

TASK [install mariadb package (CentOS)] **************************************************
skipping: [192.168.56.101]

TASK [install mariadb package (Ubuntu)] **************************************************
ok: [192.168.56.101]

TASK [start MariaDB service on CentOS] ***************************************************
skipping: [192.168.56.101]

TASK [start MariaDB service on Ubuntu] ***************************************************
changed: [192.168.56.101]
```

```
chloe@Workstation:~/HOA7_Santos$ ansible-playbook --ask-become-pass site.yml

TASK [start MariaDB service on CentOS] ****************************************************************
skipping: [192.168.56.101]

TASK [start MariaDB service on Ubuntu] ****************************************************************
changed: [192.168.56.101]

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

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

TASK [install samba package] **************************************************************************
ok: [192.168.56.105]

PLAY RECAP ********************************************************************************************
192.168.56.101             : ok=8    changed=1    unreachable=0    failed=0    skipped=3    rescued=0
ignored=0
192.168.56.102             : ok=5    changed=1    unreachable=0    failed=0    skipped=3    rescued=0
ignored=0
192.168.56.103             : ok=5    changed=1    unreachable=0    failed=0    skipped=3    rescued=0
ignored=0
192.168.56.105             : ok=4    changed=0    unreachable=0    failed=0    skipped=1    rescued=0
ignored=0
```

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.

```
chloe@Workstation:~/HOA7_Santos$ ssh chloe@192.168.56.102

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Wed Oct 15 06:08:26 2025 from 192.168.56.101
chloe@Server1:~$ cat /var/www/html/index.html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>default page</title>
</head>
<body>
    <h1>Welcome to the Default Page</h1>
    <p>This is a simple HTML document</p>
</body>
</html>
```

```
chloe@Workstation:~/HOA7_Santos$ ssh chloe@192.168.56.103

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Wed Oct 15 06:08:26 2025 from 192.168.56.101
chloe@Server2:~$ cat /var/www/html/index.html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>default page</title>
</head>
<body>
    <h1>Welcome to the Default Page</h1>
    <p>This is a simple HTML document</p>
</body>
</html>
```

```
chloe@Workstation:~/HOA7_Santos$ ssh chloemargarette@192.168.56.105
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Wed Oct 15 15:49:09 2025 from 192.168.56.101
chloemargarette@localhost:~$ cat /var/www/html/index.html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>default page</title>
</head>
<body>
    <h1>Welcome to the Default Page</h1>
    <p>This is a simple HTML document</p>
</body>
</html>
chloemargarette@localhost:~$ S
```

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

```
chloe@Workstation:~/HOA7_Santos$ git push --force origin main
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 3 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (9/9), 1.44 KiB | 1.44 MiB/s, done.
Total 9 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:ChloeMargarettePronebo/HOA7_Santos.git
 + e7faa91...588b7cf main -> main (forced update)
```

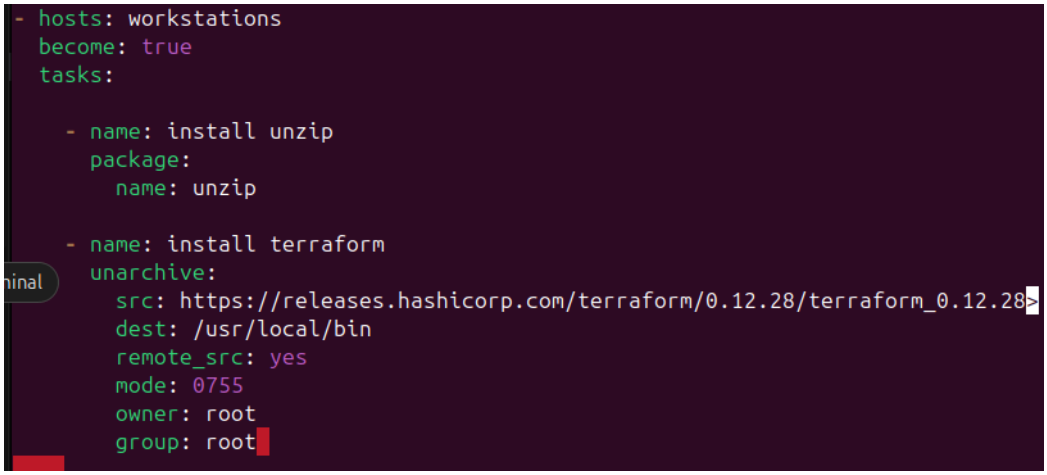
-

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



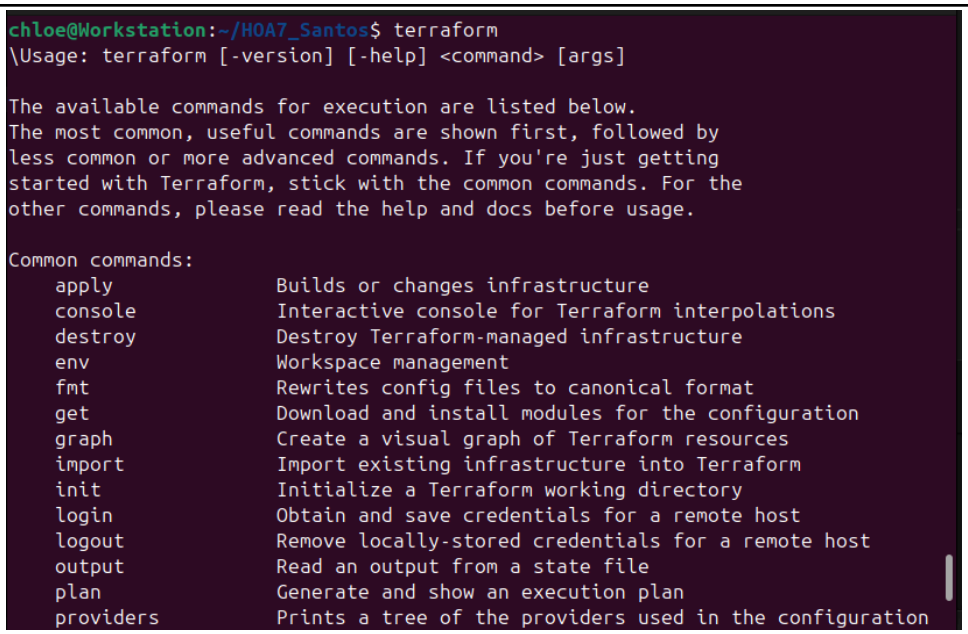
2. Edit the inventory file and add workstations group. Add any Ubuntu remote server. Make sure to remember the IP address.

```
 GNU nano 7.2                          inventory.ini
[web_servers]
192.168.56.102 ansible_user=chloe
192.168.56.103 ansible_user=chloe
192.168.56.105 ansible_user=chloemargarette

[workstations]
192.168.56.101 ansible_user=chloe

[db_servers]
192.168.56.101 ansible_user=chloe

[file_servers]
192.168.56.105 ansible_user=chloemargarette

                        [ Read 14 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^/ Go To Line
```

3. Run the playbook. Describe the output.

```
TASK [start MariaDB service on CentOS] ************************************
skipping: [192.168.56.101]

TASK [start MariaDB service on Ubuntu] ************************************
changed: [192.168.56.101]

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

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

TASK [install samba package] **********************************************
ok: [192.168.56.105]

PLAY RECAP ****************************************************************
192.168.56.101             : ok=8    changed=2    unreachable=0    failed=0    s
kipped=3    rescued=0    ignored=0
192.168.56.102             : ok=5    changed=1    unreachable=0    failed=0    s
kipped=3    rescued=0    ignored=0
192.168.56.103             : ok=5    changed=1    unreachable=0    failed=0    s
kipped=3    rescued=0    ignored=0
192.168.56.105             : ok=8    changed=2    unreachable=0    failed=0    s
kipped=2    rescued=0    ignored=0
```

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



Task 3: Create roles

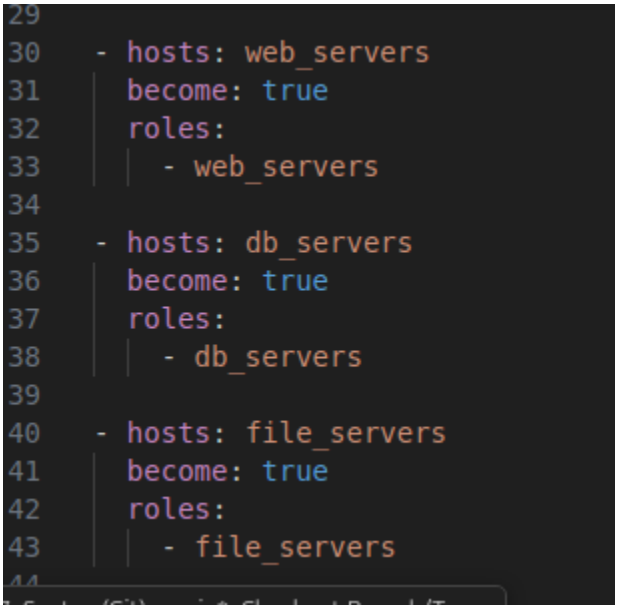
1. Edit the site.yml. Configure roles as follows: (make sure to create a copy of the old site.yml file because you will be copying the specific plays for all groups)

```
---
- 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.

new_site.yml

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

    - name: update repository index(CentOS)
      tags: always
      dnf:
        update_only: yes
        update_cache: 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: worksations
  become: true
  roles:
    - worksations

- hosts: web_servers
  become: true
  roles:
    - web_servers

```



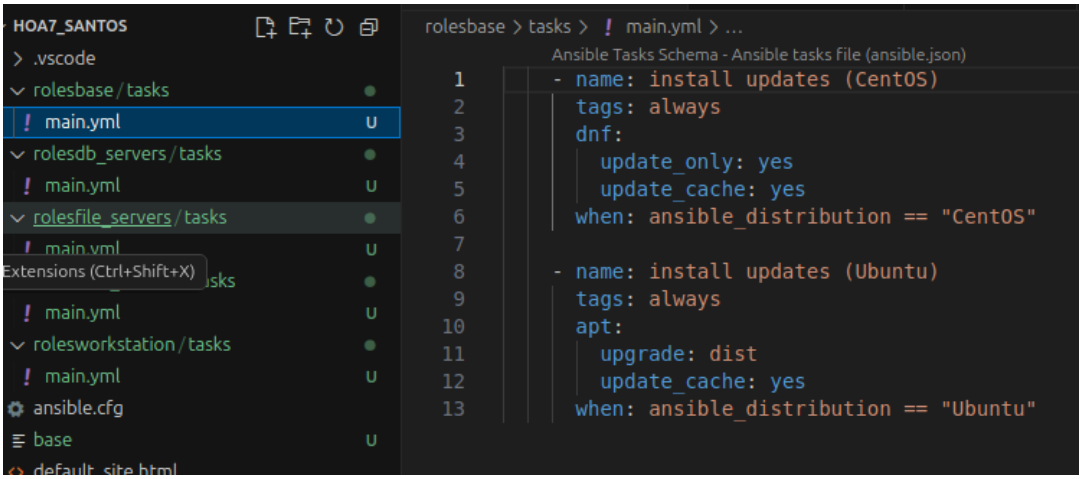
```yaml
- hosts: web_servers
  become: true
  roles:
    - web_servers

- hosts: db_servers
  become: true
  roles:
    - db_servers

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

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.

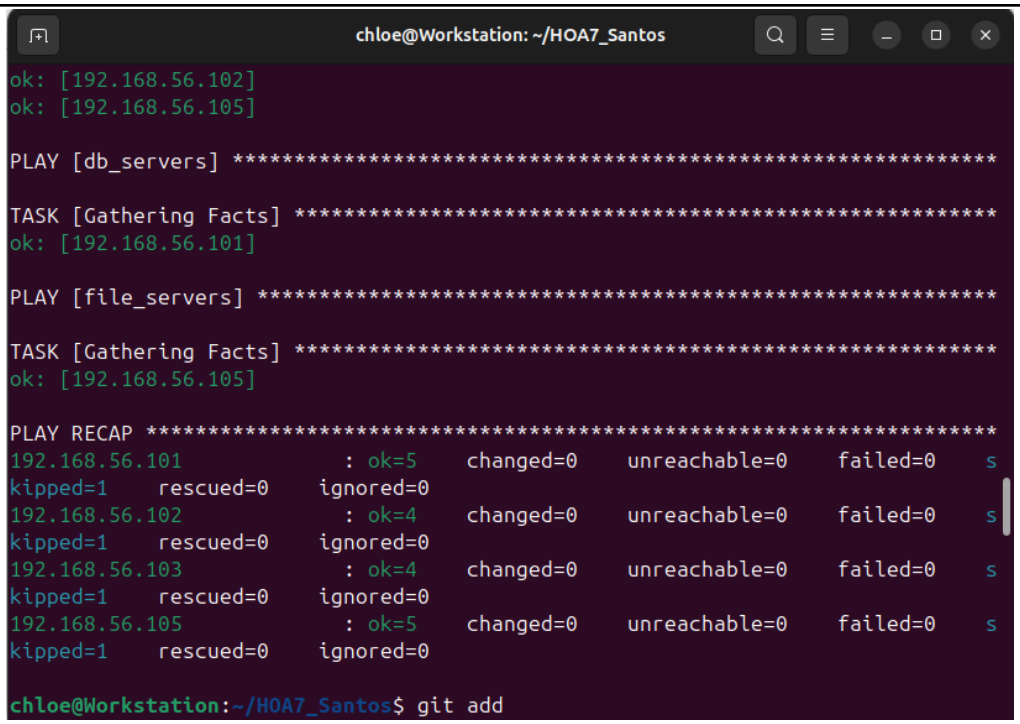
```
chloe@Workstation:~$ cd HOA7_Santos
chloe@Workstation:~/HOA7_Santos$ mkdir -p roles{base,web_servers,file_serve
_servers,db_servers,workstation}/tasks
chloe@Workstation:~/HOA7_Santos$ touch mkdir roles{base,web_servers,file_se
,db_servers,db_servers,workstation}/tasks/main.yml
chloe@Workstation:~/HOA7_Santos$
```

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.



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

5.



Reflections:

Answer the following:

1. What is the importance of creating roles?

   Ansible roles help organize playbooks by dividing them into functional sections. This makes tasks easier to manage, reuse, and update. A clear structure also simplifies troubleshooting and helps maintain an organized codebase.

2. What is the importance of managing files?

   Using Ansible to manage files helps keep your systems neat and organized. It ensures all your configurations are consistent, lets you track changes to files over time, and helps prevent mistakes when you're automating tasks or applying updates.