Name:	Date Performed:
Course/Section:	Date Submitted:
Instructor:	Semester and SY:

Activity 7: Managing Files and Creating Roles in Ansible

1. **Objectives:**

1.1 Manage files in remote servers

1.2 Implement roles in ansible

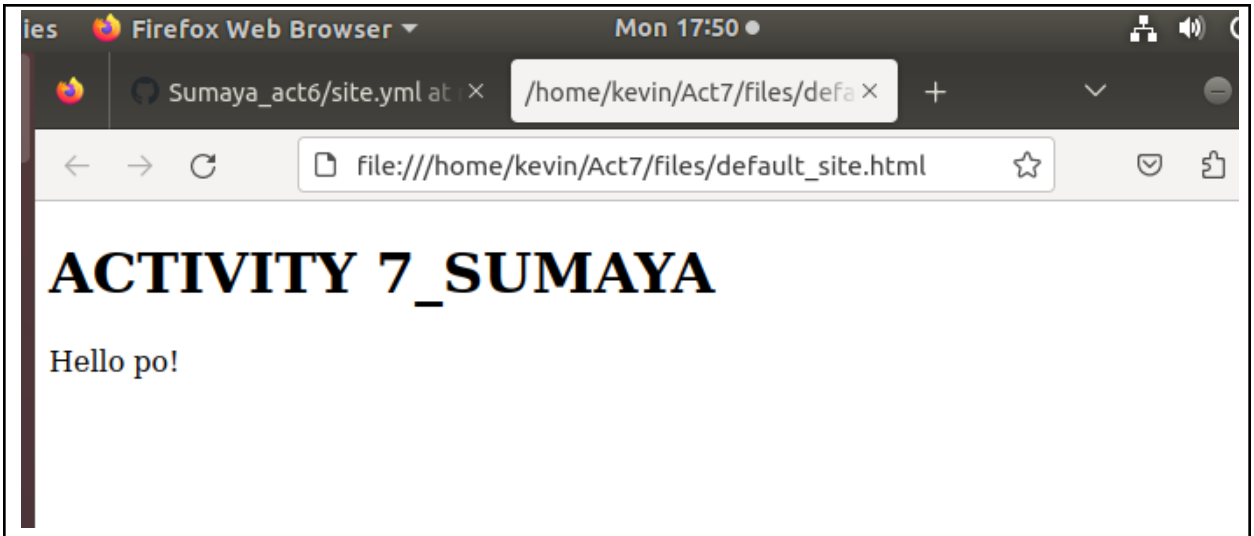
2. **Discussion**:

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

<!DOCTYPE html>
<html>
<body>

<h1> ACTIVITY 7_SUMAYA </h1>

<p> Hello po!  </p>


</body>
</html>
```



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

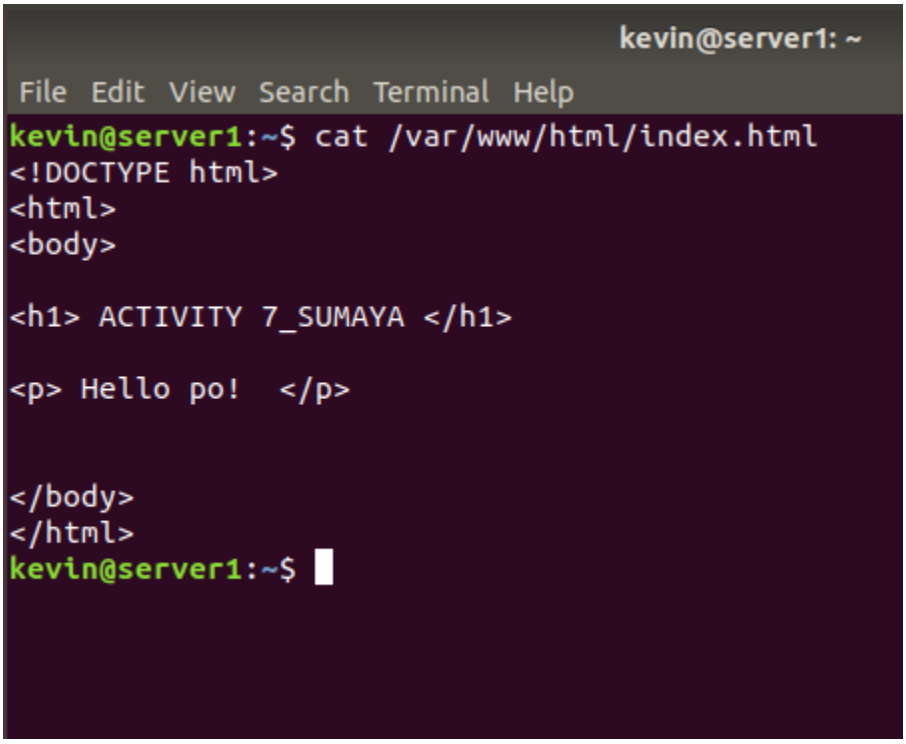
     tags: apache, apache2, httpd
     copy:
       src: default_site.html
       dest: /var/www/html/index.html
       owner: root
       group: root
       mode: 0644

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

  - name: copy default html file for site
    tags: apache, apach2, 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.

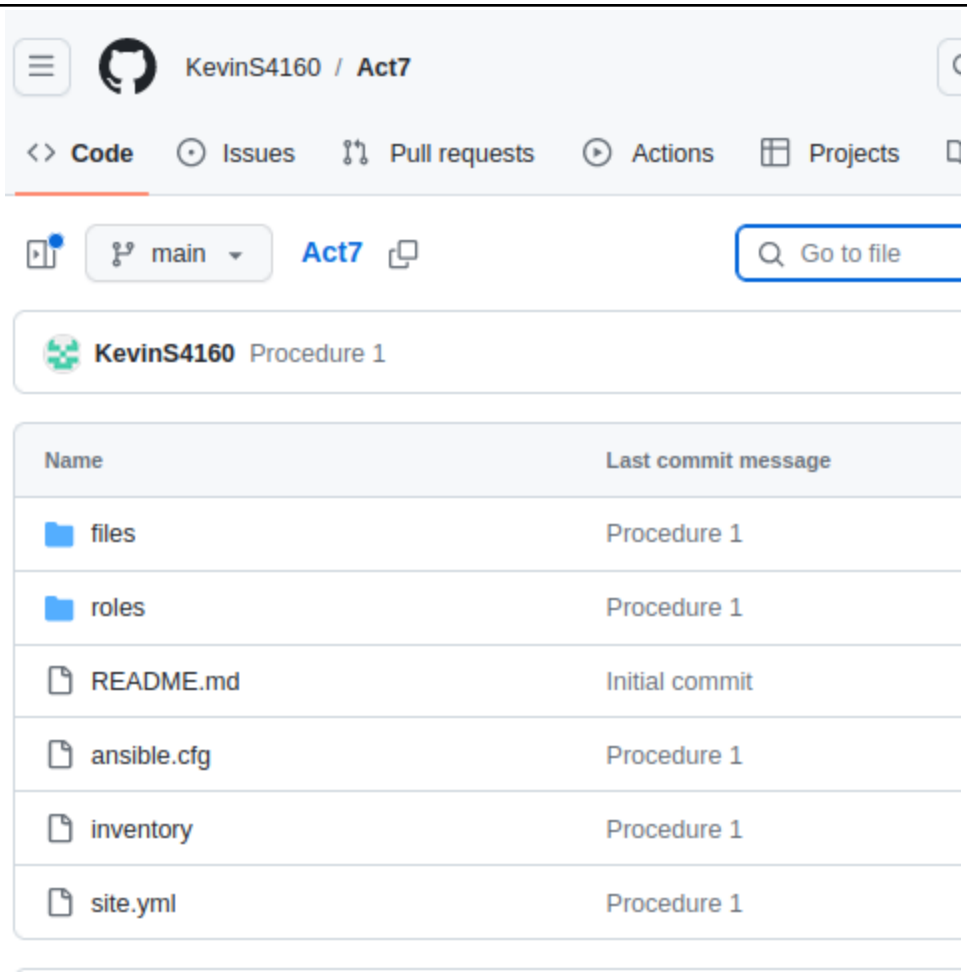
4. Go to the remote servers (*web_servers*) listed in your inventory. Use cat command to check if the index.html is the same as the local repository file (*default_site.html*). Do both for Ubuntu and CentOS servers. On the CentOS

server, go to the browser and type its IP address. Describe the output.



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

```
15 files changed, 125 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 files/.default_site.html.swp
 create mode 100644 files/.default_site.yml.swp
 create mode 100644 files/default_site.html
 create mode 100644 inventory
 create mode 100644 roles/base/tasks/.main.yml.swp
 create mode 100644 roles/base/tasks/main.yml
 create mode 100644 roles/db_servers/tasks/.main.yml.swp
 create mode 100644 roles/db_servers/tasks/main.yml
 create mode 100644 roles/file_servers/tasks/.main.yml.swp
 create mode 100644 roles/file_servers/tasks/main.yml
 create mode 100644 roles/manageNode/.main.yml.swp
 create mode 100644 roles/manageNode/main.yml
 create mode 100644 roles/web_servers/tasks/main.yml
 create mode 100644 site.yml
kevin@Workstation:~/Act7$ git push origin
Username for 'https://github.com': KevinS4160
Password for 'https://KevinS4160 @github.com':
Counting objects: 24, done.
Delta compression using up to 6 threads.
Compressing objects: 100% (18/18), done.
Writing objects: 100% (24/24), 2.25 KiB | 2.25 MiB/s, done.
Total 24 (delta 5), reused 0 (delta 0)
remote: Resolving deltas: 100% (5/5), done.
To https://github.com/KevinS4160/Act7.git
   2590817..4ae3f72  main -> main
```



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

```
kevin@Workstation:~/Act7$ ansible-playbook --ask-become-pass site.yml
BECOME password:

*
192.168.56.102             : ok=5    changed=0    unreachable=0    failed=0
skipped=1    rescued=0    ignored=0
192.168.56.103             : ok=5    changed=1    unreachable=0    failed=0
skipped=2    rescued=0    ignored=0
sumaya@192.168.56.110      : ok=6    changed=1    unreachable=0    failed=0
skipped=2    rescued=0    ignored=0
```

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

```
kevin@Workstation:~/Act7$ ansible-playbook --ask-become-pass site.yml
BECOME password:

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

TASK [copy default html file for site] *********************************
*
changed: [192.168.56.103]

192.168.56.102             : ok=6    changed=0    unreachable=0    failed=0
skipped=3    rescued=0    ignored=0
192.168.56.103             : ok=9    changed=3    unreachable=0    failed=0
skipped=2    rescued=0    ignored=0
sumaya@192.168.56.110      : ok=8    changed=1    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.

```
kevin@server2:~$ terraform -version
Terraform v0.12.28

Your version of Terraform is out of date! The latest version
is 1.5.7. You can update by downloading from https://www.terraform.io/downloads
.html
```

-

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.

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.

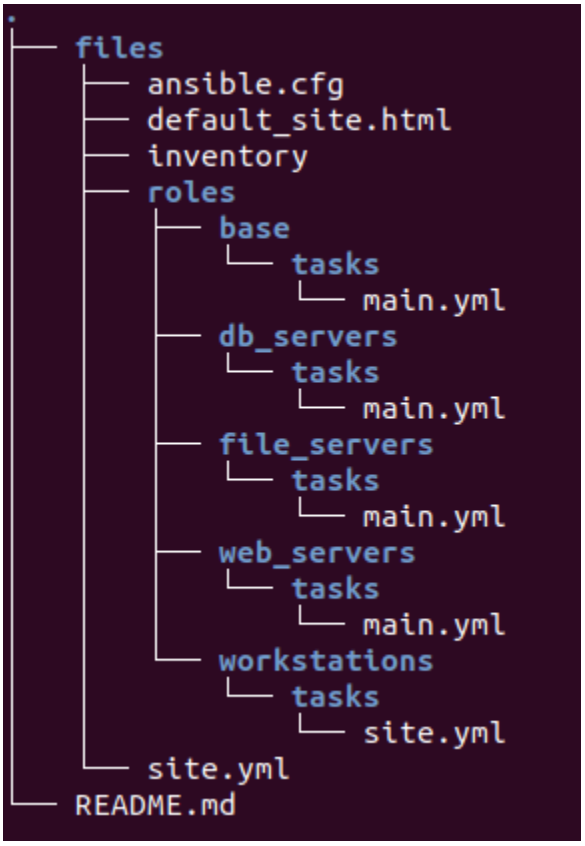
```
kevin@Workstation:~/Act7$ ansible-playbook --ask-become-pass site.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************
*
ok: [192.168.56.102]
ok: [sumaya@192.168.56.110]
ok: [192.168.56.103]

TASK [update repository index (CentOS)] ****************************************
*
skipping: [192.168.56.102]
skipping: [192.168.56.103]
ok: [sumaya@192.168.56.110]

TASK [install update (Ubuntu)] *************************************************
*
skipping: [sumaya@192.168.56.110]
ok: [192.168.56.102]
ok: [192.168.56.103]

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

```
skipping: [192.168.56.103]
ok: [sumaya@192.168.56.110]

TASK [install update (Ubuntu)] *************************************************
*
skipping: [sumaya@192.168.56.110]
ok: [192.168.56.102]
ok: [192.168.56.103]

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

TASK [Gathering Facts] *********************************************************
*
ok: [192.168.56.102]
ok: [192.168.56.103]
ok: [sumaya@192.168.56.110]

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

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

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

TASK [Gathering Facts] *********************************************************
```

```
ok: [192.168.56.102]
ok: [sumaya@192.168.56.110]

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

TASK [Gathering Facts] ******************************************************
*
ok: [192.168.56.103]
ok: [sumaya@192.168.56.110]

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

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

PLAY RECAP ******************************************************************
*
192.168.56.102             : ok=5    changed=0    unreachable=0    failed=0
skipped=1    rescued=0    ignored=0
192.168.56.103             : ok=5    changed=0    unreachable=0    failed=0
skipped=1    rescued=0    ignored=0
sumaya@192.168.56.110      : ok=5    changed=0    unreachable=0    failed=0
skipped=1    rescued=0    ignored=0
```

```
kevin@Workstation:~/Act7$ tree
.
├── files
│   ├── ansible.cfg
│   ├── default_site.html
│   ├── inventory
│   ├── roles
│   │   ├── base
│   │   │   └── tasks
│   │   │       └── main.yml
│   │   ├── db_servers
│   │   │   └── tasks
│   │   │       └── main.yml
│   │   ├── file_servers
│   │   │   └── tasks
│   │   │       └── main.yml
│   │   ├── web_servers
│   │   │   └── tasks
│   │   │       └── main.yml
│   │   └── workstations
│   │       └── tasks
│   │           └── site.yml
│   └── site.yml
└── README.md
```

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.

```
kevin@Workstation:~/Act7$ ansible-playbook --ask-become-pass site.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************
*
ok: [192.168.56.102]
ok: [192.168.56.103]
ok: [sumaya@192.168.56.110]

TASK [update repository index (CentOS)] ****************************************
*
skipping: [192.168.56.102]
skipping: [192.168.56.103]
ok: [sumaya@192.168.56.110]

TASK [install update (Ubuntu)] *************************************************
*
skipping: [sumaya@192.168.56.110]
ok: [192.168.56.102]
ok: [192.168.56.103]

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

TASK [Gathering Facts] *********************************************************
*
```

```
ok: [192.168.56.102]
ok: [192.168.56.103]
ok: [sumaya@192.168.56.110]

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

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

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

TASK [Gathering Facts] *********************************************************
*
ok: [sumaya@192.168.56.110]
ok: [192.168.56.102]

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

TASK [Gathering Facts] *********************************************************
*
ok: [192.168.56.103]
ok: [sumaya@192.168.56.110]
```

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

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

PLAY RECAP *********************************************************************
*
```

```
192.168.56.102             : ok=5    changed=0    unreachable=0    failed=0
skipped=1    rescued=0    ignored=0
192.168.56.103             : ok=5    changed=0    unreachable=0    failed=0
skipped=1    rescued=0    ignored=0
sumaya@192.168.56.110      : ok=5    changed=0    unreachable=0    failed=0
skipped=1    rescued=0    ignored=0
```

[db_servers]

```
  GNU nano 2.9.3                          main.yml

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

  - 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]

```
 GNU nano 2.9.3                                main.yml

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

[web_servers]

```
 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: 0644

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

- name: install apache and php for CentOS servers
  tags: apache,centos,httpd
  dnf:
    name:
```

```
- 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
  when: ansible_distribution == "CentOS"
```

[workstations]

```
GNU nano 2.9.3                          site.yml

- 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
    owner: root
    group: root
```

Name	Last commit message	Last commit date
..		
base/tasks	HOA7	2 minutes ago
db_servers/tasks	HOA7	2 minutes ago
file_servers/tasks	HOA7	2 minutes ago
web_servers/tasks	HOA7	2 minutes ago
workstations/tasks	HOA7	2 minutes ago

https://github.com/KevinS4160/Act7/tree/main

Reflections:

Answer the following:

1. What is the importance of creating roles?
    - Roles are independent of each other, meaning that the execution of one role does not depend on the execution of another role. This makes it easier to modify and reuse roles, eliminating the need for rewriting plays and tasks in the playbook file.

2. What is the importance of managing files?
    - Managing files in Github playbook in Ubuntu is important for version control and collaboration purposes. Additionally, using roles in GitHub playbooks provides an efficient way to organize and reuse code, making it easier to manage complex automation tasks

Conclusion:

- After doing the activity I learned how to manage and arrange the playbooks directories to its designating files and also run different commands to allow to run ansible.cfg to different locations like base, web_servers, db_servers, file_servers and workstations.