Ansible Roles

-> Roles are a level of abstraction for Ansible configuration in a modular and reusable format

-> As you add more and more functionality to your playbooks, they can become difficult to maintain

-> Roles allow you to break down a complex playbook into separate, smaller chunks that can be coordinated by a central entry point.

Sample playbook with Role

```
---
- hosts: all
  become: true
  roles:
    - apache
```

1. Ansible roles are consists of many playbooks, which is similar to modules in puppet and cook books in chef.
    We term the same in ansible as roles.

2. Roles are a way to group multiple tasks together into one container to do the automation in very effective manner with clean directory structures.

3. Roles are set of tasks and additional files for a certain role which allow you to break up the configurations.

4. It can be easily reuse the codes by anyone if the role is suitable to someone.

5. It can be easily modify and will reduce the syntax errors.

How do we create Ansible Roles?

-> To create an Ansible role, use "ansible-galaxy" command which has the templates to create it.

```
$ sudo su ansible

$ cd /home/ansible

$ mkdir roles

$ cd roles

$ ansible-galaxy init apache
```

> where, ansible-glaxy is the command to create the roles using the templates.

> init is to initiliaze the role.

> apache is the name of the role

List out the directory created under apache

```
$ sudo yum install tree
$ tree apache
```

We have got the clean directory structure with the ansible-galaxy command. Each directory must contain a main.yml file, which contains the relevant content.

+++++++++++++++++++++
Directory Structure:
+++++++++++++++++++++

tasks â€“ contains the main list of tasks to be executed by the role.

handlers â€“ contains handlers, which may be used by this role or even anywhere outside this role.

defaults â€“ default variables for the role.

vars â€“ other variables for the role. Vars has the higher priority than defaults.

files â€“ contains files required to transfer or deployed to the target machines via this role.

templates â€“ contains templates which can be deployed via this role.

meta â€“ defines some data / information about this role (author, dependency, versions, examples, etc,.)

+++
Lets take an example to create a role for Apache Web server.
+++

Below is a sample playbook to deploy Apache web server. Lets convert this playbook code into Ansible role.

```
- hosts: all
  become: true
  tasks:
   - name: Install Httpd
     yum:
      name: httpd
      state: present
   - name: Copy index.html
     template:
        src: index.html
        dest: /var/www/html/index.html
   - name: Start Http Server
     service:
       name: httpd
       state: started
```

First, move on to the Ansible roles directory and start editing the yml files.

```
$ cd roles/apache
```

1. Tasks
+++++++++++++++
Edit main.yml available in the tasks folder to define the tasks to be executed.

```
$ vi tasks/main.yml
```

```
---
# tasks file for roles/apache
- name: install httpd
  yum:
   name: httpd
   state: present
- name: Copy index.html
  copy:
    src=index.html
    dest=/var/www/html/
  notify:
  - restart apache
```

2. Files
++++++++++++
-> Copy required files into files directory or create index.html file with content

```
$ vi files/index.html

<write content here>
```

3. Handlers
+++++++++++++
Edit handlers main.yml to restart the server when there is a change. Because we have already defined it in the tasks with notify option. Use the same name â€œrestart apacheâ€� within the main.yml file as below.

```
$ vi handlers/main.yml

- name: restart apache
  service:
   name: httpd
   state: restarted
```

We have got all the required files for Apache roles. Lets apply this role into the ansible playbook â€œrunsetup.ymlâ€� as below to deploy it on the client nodes.

```
$ vi /home/ansible/runsetup.yml

---
- hosts: all
  become: true
  roles:
   - apache
...
```

-> Execute playbook which contains apache role

```
        $ ansible-playbook runsetup.yml
```

If you have created multiple roles, you can use the below format to add them in the playbook

```
---
- hosts: all
  become: true
  roles:
   - apache
   - jenkins
   - java
   - maven
   - sonar
```

===========================

What We Covered in Ansible?

===========================

1) What is Configuration Management
2) What is Ansible
3) Advantages of Ansible
4) Push Based Vs Pull Based Mechanism
5) Ansible Installation

6) Ansible Architecture
7) Host Inventory File
8) Host Groups in Inventory
9) Ansible Ad Hoc Commands
10) YAML
11) Working with YAML in VS Code IDE
12) Playbook Introduction
13) Playbook Commands
14) Writing Playbooks
15) Variables

    - Runtime vars
    - Playbook vars
    - Group vars
    - Host Vars

16) Handlers
17) Tags
18) Roles
19) Ansible Vault
20) Playbook for Multiple OS Family Based Hosts