57.19Jun.AnsibleRoles
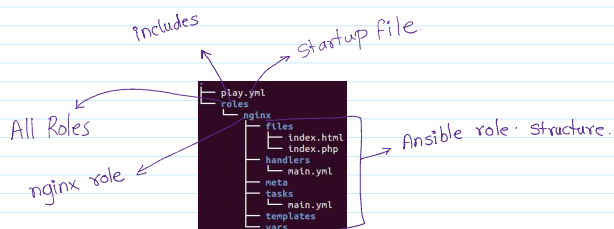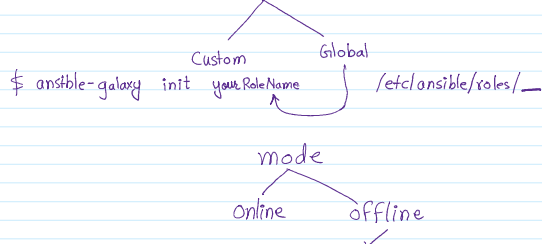
A. Ansible Role & Galaxy
B. Jenkins Integrations with:
   a. Github
   b. Java
   c. Maven
   d. Docker
   e. Ansible
C. Minikube (AWS/GCloud Shell/PWK)
D. K8S CRUD App Deployments

Ansible Role & Galaxy

ansible.playbook.

Playbook → Ansible Control Server — SSH → MS1

— SSH → MS2

— MS3

Inventory

Playbook

task's

t1 → U1

t2 ← U2

t3 → U3
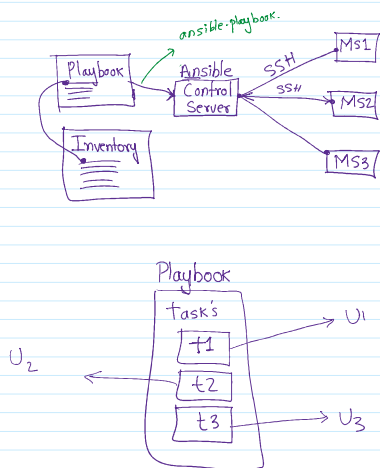
Ansible Role

- best enterprise workLoad
- Reduce rewritting.
- Encapsulate.
- breaking a playbook.
- Structure of Ansible role.
- Ansible-galaxy CLI for manage role's.
- folder /roles

Custom   Global

$ ansible-galaxy init yourRoleName   /etc/ansible/roles/__

mode

Online   offline

---

includes

Startup file.

```
├── play.yml
└── roles
    └── nginx
        ├── files
        │   ├── index.html
        │   └── index.php
        ├── handlers
        │   └── main.yml
        ├── meta
        ├── tasks
        │   └── main.yml
        ├── templates
        └── vars
```

All Roles

nginx role

Ansible role: structure.

- Role allow us to create very minimal playbook.

files

- files to be tranferred to managed Server(s)
  → – index.html – index.php – app.jar – app.war.

- files to be tranferred to managed server(s)
  - index.html - index.php - app.jar - app.war.
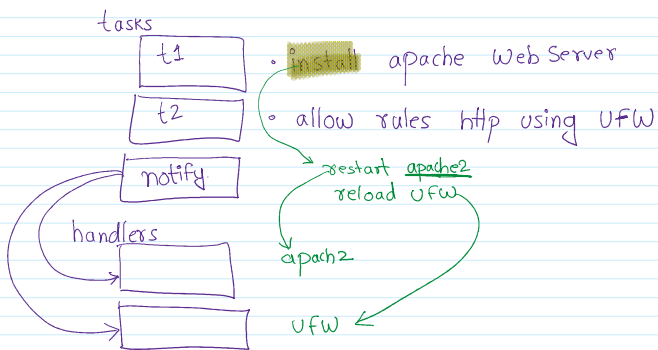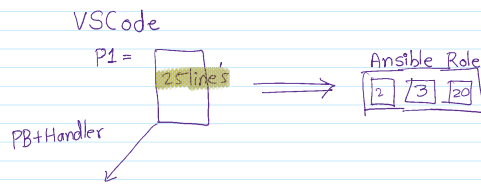

CS → MS1
CS → MS2
CS → MS3
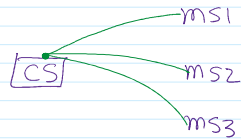
handlers
- main.yml   default file.
- All handlers that were in our Playbook
  can now be added into this directory.
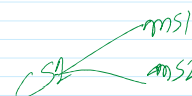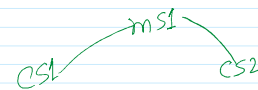
tasks
- Playbook tasks will be added.
- main.yml default.

defaults
- all default variables

VSCode

P1 = [25 lines] → Ansible Role.
                    [2] [3] [20]

PB+Handler ↓

tasks
[t1] - install   apache Web Server
[t2] - allow rules http using UFW

[notify] → restart apache2
            reload ufw

handlers
[ ]   apach2
[ ]   →   UFW

become: yes
tasks:
  - name: Install Apache2 on Ubuntu server   ← Task 1
    apt:
      name: apache2
      state: present
      update_cache: yes

  - name: Allow HTTP traffic on UFW firewall   ← Task 2
    ufw:
      rule: allow
      port: http
      proto: tcp

    notify:
      - Restart apache2
      - Reload ufw firewall

handlers:
  - name: Restart apache2   ← Handler 1
    service:
      name: apache2
      state: restarted

  - name: Reload ufw firewall   ← Handler 2
    ufw:
      state: enabled

CS1 → MS1   CS2

CS1 → MS1
       MS2

become : yes == sudo

sudo apt install nginx

become_User : John ← Linux User.

sudo — john
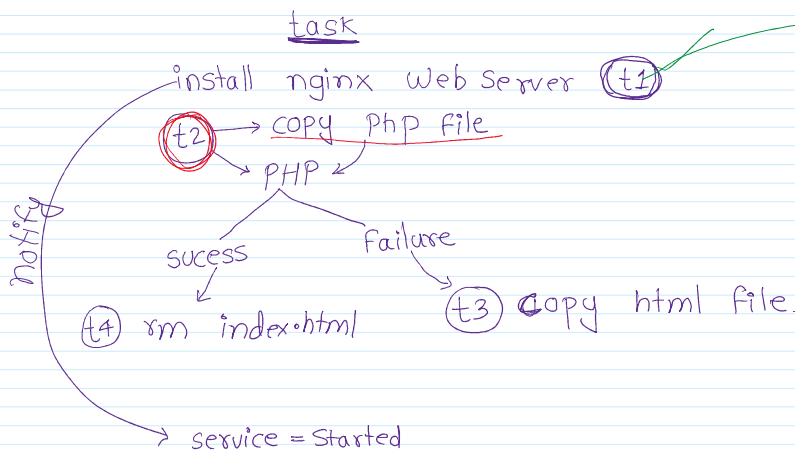
become : yes or become-user : root

- hosts: ws
  tasks:
  - name: Installs Nginx
    apt: pkg=nginx state=present update_cache=true
    notify:
      - Start Nginx

  - name: Upload default index.php for host
    copy: src=files/index.php dest=/usr/share/nginx/html/ mode=0644
    register: php
    ignore_errors: True

  - name: Remove index.html for host
    command: rm /usr/share/nginx/html/index.html
    when: php|success

  - name: Upload default index.html for host
    copy: src=files/index.html dest=/usr/share/nginx/html/ mode=0644
    when: php|failed

  handlers:
  - name: Start Nginx
    service: name=nginx state=started

→ nginx

module = Copy

param's = 3

① src

② dest

③ mode=777
= 000777

task

install nginx web server (t1)

(t2) → copy php file

→ PHP

sucess   Failure

(t4) rm index.html   (t3) copy html file.

notify

service = Started

- hosts: ws
  tasks:
  - name: Installs Nginx
    apt: pkg=nginx state=present update_cache=true
    notify:
      - Start Nginx

  - name: Upload default index.php for host
    copy: src=static_files/index.php dest=/usr/share/nginx/html/ mode=0644
    register: php
    ignore_errors: True

  - name: Remove index.html for host
    command: rm /usr/share/nginx/html/index.html
    when: php is success

  - name: Upload default index.html for host
    copy: src=static_files/index.html dest=/usr/share/nginx/html/ mode=0644
    when: php|failed

  handlers:
  - name: Start Nginx
    service: name=nginx state=started

t1
t2
t4
t3

WS

ms1
ms2

CS

ms3
ms4
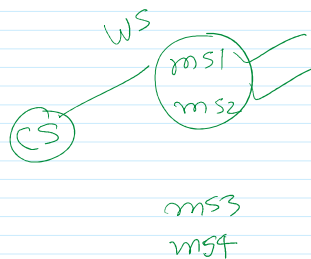
Ansible Role

✓ 1] Install Ansible (ansible-galaxy)
✓ 2] Test ansible ping.
  3] Ansible role initialize.
  4] Check Structure ansible role
  5] Playbook ⟶ role.
  6] Test/Apply.

```bash
#!/bin/bash
sudo apt update
sudo apt install python3-pip -y
sudo mkdir /etc/ansible
sudo bash -c 'cat<<EOF > /etc/ansible/hosts
[ws]
127.0.0.1
EOF'
sudo apt install ansible -y
ssh-keygen -b 3072 -t rsa -f ~/.ssh/sshkey -q -N ""
cat ~/.ssh/sshkey.pub | cat >> ~/.ssh/authorized_keys
touch ~/.ssh/config
sudo bash -c 'cat<<EOF > ~/.ssh/config
Host *
    User root
    IdentityFile /root/.ssh/sshkey
EOF'
sudo ansible --version > /tmp/ansibleVersion.txt


$ cat  /tmp/ansibleVersion.txt

$ ansible --version
$ ansible-playbook --version
$ ansible-galaxy --version

$ apt install tree
$ ansible-galaxy init /etc/ansible/roles/nginx --offline


root@ip-172-31-4-128:/etc/ansible/roles# cat play.yml
---
- hosts: ws
  roles:
    - role: nginx



root@ip-172-31-4-128:/etc/ansible/roles/nginx# cat files/index.php
<!DOCTYPE html>
<html>
   <head>
      <title>Example</title>
   </head>
   <body>
      <p>This is an example of a simple HTML page with one paragraph.</p>
   </body>
</html>

root@ip-172-31-4-128:/etc/ansible/roles/nginx# cp files/index.php files/index.html



root@ip-172-31-4-128:/etc/ansible/roles/nginx# cat tasks/main.yml
---
# tasks file for /etc/ansible/roles/nginx
- name: Install nginx
  apt: pkg=nginx state=present update_cache=true
  notify:
    - Start Nginx

- name: Upload default index.php for host
  copy: src=files/index.php dest=/usr/share/nginx/html/ mode=0644
  register: php
  ignore_errors: True

- name: Remove index.html for host
  command: rm /usr/share/nginx/html/index.html
  when: php is success

- name: Upload default index.html for host
  copy: src=files/index.html dest=/usr/share/nginx/html/ mode=0644
  when: php is failed


root@ip-172-31-4-128:/etc/ansible/roles/nginx# cat handlers/main.yml
- name: Start Nginx
  service: name=nginx state=started
# handlers file for /etc/ansible/roles/nginx
root@ip-172-31-4-128:/etc/ansible/roles/nginx# cd ..
root@ip-172-31-4-128:/etc/ansible/roles# ansible-playbook play.yml

root@ip-172-31-4-128:/etc/ansible/roles# cd /usr/share/nginx/html/
root@ip-172-31-4-128:/usr/share/nginx/html# ls
index.php
root@ip-172-31-4-128:/usr/share/nginx/html# cat index.php

root@ip-172-31-4-128:/usr/share/nginx/html# curl http://127.0.0.1
```