$ ansible -m ping dev : it will ping all the host under dev group in inventory file.

$ ansible --list-hosts all : to list all the hosts

$ ansible --list-hosts dev : to list hosts under dev group

$ ansible-playbook main.yaml : to run the playbook called main.yaml

$ ansible-playbook -C main.yaml : it will run a dry check on main.yaml playbook

$ ansible-playbook --syntax-check main.yaml : to check the syntax of main.yaml

$ ansible-playbook --step main.yaml : it will ask for the permission before running each task

**Ansible facts**

$ ansible <host-name> -m setup |less : it will give you all the ansible facts variable for particular host (managed host)

$ ansible <host-name> -m setup -a “filter=\*ipv4” : gives you info about all the ipv4

$ ansible <host-name> -m setup -a “filter=hostname” : gives you hostname of the particular host

$ ansible <host-name> -m setup -a “filter=ansible\_fqdn” : gives you DNS name of host

**Ansible-galaxy**

$ ansible-galaxy init /etc/ansible/roles/apache --offline : it will create a role for apache in described directory , but when you mentioned --offline it will just create folder structure and not pull actual role from ansible-galaxy repo.

/etc/ansible/hosts : inventory file location

/etc/ansible/ansible.cfg : config file location

/usr/lib/pythonx.x/site-packages/ansible/modules directory : module location

**to create Key-less authentication b/w Control and client nodes.**

$ ssh-keygen -t rsa : to generate the ssh key, so we can copy and paste it in all the client nodes to enable key-less auth. Key will be generates inside hidden folder called .ssh

$ scp -p id\_rsa.pub client1:/home/viral/.ssh/authorized\_keys : on control node go inside hidden directory .ssh and then run this cmd to copy id\_rsa.pub from control node to all the client server (here client1) , to make key-less authentication b/w two servers.

**Shell scripting :**

/opt/aptm/ctm\_stop.sh > /tmp/out2 ; : here we are running .sh file and putting its output in /tmp/out2

cat /tmp/out2|grep -iw ‘failed’ > /dev/null ; : here we are looking for ‘failed’ keyword and putting it in to null so we

won’t see it

if [ $? == 0 ] ; then echo ‘task failed’ ; : here $? refers to the previous query, if it is true then it will display 0 else

anything but 0

else echo ‘task passed’ ; fi : here we are echoing ‘task passed’ if $? != 0