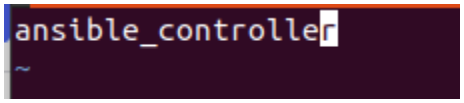
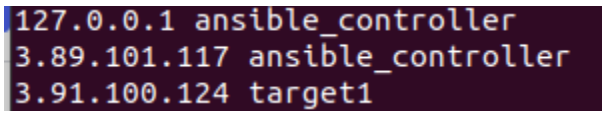
ANSIBLE- https://docs.ansible.com/

 # Change the name of you host to recognise on which server we are working
Sudo vim /etc/hostname
Remove everything & add ***ansible_controller***



sudo reboot

*# Now edit hosts name file to add target servers ip and name on **ansible_controller** & **on targets** as well.*
Sudo vim /etc/hosts



 # Now to enable ssh access from ansible controller to target machines, we need to generate ssh keys for controller server and then to add the same generated key to authorized_keys of target hosts.
Ansible_controller server
ssh-keygen

***#** copy id_rsa.pub to target server authorized_keys and confirm ssh access from ansible controller*
ssh ubuntu@<target_ip_address>

***Ansible Installation on ansible controller server**:-*
https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html#installing-ansible-on-ubuntu
sudo apt update -y
sudo apt install software-properties-common -y
sudo add-apt-repository --yes --update ppa:ansible/ansible
sudo apt install ansible -y
ansible --version

Ansible ssh and password connection method:-
https://docs.ansible.com/ansible/latest/user_guide/connection_details.html
Ansible Adding Hosts in inventory file:-
https://www.bogotobogo.com/DevOps/Ansible/Ansible-SSH-Connection-Setup-Run-Command.php

create inventory list in ansible_controller
mkdir project
cd project

EXAMPLE 1:-

sudo vim inventory2.txt #add below in inventory2.txt file

```
[prod]
3.91.100.124    ansible_connection=ssh    ansible_ssh_private_key_file=/home/ubuntu/.ssh/controller
 #IP of target      #ansible connection method                  #which key to be used to ssh
[dev]
4.91.100.1      ansible_connection=ssh    ansible_ssh_private_key_file=/home/ubuntu/.ssh/controller
4.91.101.2      ansible_connection=ssh    ansible_ssh_private_key_file=/home/ubuntu/.ssh/controller
#IP of target    #ansible connection method                  #which key to be used to ssh
```

ansible -i inventory2.txt -m ping prod # this command will ping the server under **prod** in the inventory2 file.

```
ubuntu@ansible_controller:~/.ssh/project$ ansible -i inventory2.txt -m ping prod
3.91.100.124 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```

ansible -i inventory2.txt -m ping dev # this command will ping the server under **dev** in the inventory2 file.

ansible -i inventory2.txt -m ping all # this command will ping **all** the servers under **prod & dev** inventory2.

EXAMPLE 2:-

sudo vim inventory.txt #add below in inventory.txt file

host1 ansible_host=3.91.100.124 ansible_connection=ssh ansible_ssh_private_key_file=/home/ubuntu/.ssh/controller

ansible -i inventory.txt -m ping host1 # this command will ping the server under **prod** in the inventory file.

```
ubuntu@ansible_controller:~/.ssh/project$ cat inventory.txt
host1 ansible_host=3.91.100.124  ansible_connection=ssh   ansible_ssh_private_key_file=/home/
ubuntu/.ssh/controller
ubuntu@ansible_controller:~/.ssh/project$ ansible -i inventory.txt -m ping host1
host1 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```

EXAMPLE 3:-

sudo vim inventory3.txt #add below in inventory3.txt file

3.91.100.124 ansible_connection=ssh ansible_ssh_private_key_file=/home/ubuntu/.ssh/controller

```
ubuntu@ansible_controller:~/.ssh/project$ cat inventory3.txt
3.91.100.124    ansible_connection=ssh     ansible_ssh_private_key_file=/home/ubuntu/.ssh/controller
ubuntu@ansible_controller:~/.ssh/project$ ansible -i inventory3.txt -m ping 3.91.100.124
3.91.100.124 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```

EXAMPLE 4:- IF ssh is setup with username and password
sudo vim inventory4.txt #add below in inventory4.txt file
host1 ansible_host=3.91.100.124 ansible_ssh_pass=<password> ansible_user=<username>

By default, Ansible uses username of ansible controller in our case its **ubuntu**
# Like if we ssh to a new instance it ask for to add key fingerprint, to disable it we need to below changes:-
sudo vim /etc/ansible/ansible.conf

```
# uncomment this to disable SSH key host checking
#host_key_checking = False
```

It will be under the [Default] Section
Just uncomment it and save the file.

ANSIBLE ROLES
https://galaxy.ansible.com/search?deprecated=false&tags=database&keywords=mysql&order_by=-relevance&page=1

```
- name: Install and Configure MySQL
  hosts: db-server1……db-server100
  roles:
      - mysql
```

MySQL-Role

```
tasks:
    - name: Install Pre-Requisites
      yum: name=pre-req-packages state=present

    - name: Install MySQL Packages
      yum: name=mysql state=present

    - name: Start MySQL Service
      service: name=mysql state=started

    - name: Configure Database
      mysql_db: name=db1 state=present
```

Roles path can be checked in ansible.conf

cat /etc/ansible/ansible.conf

Path:-

/etc/ansible/roles/