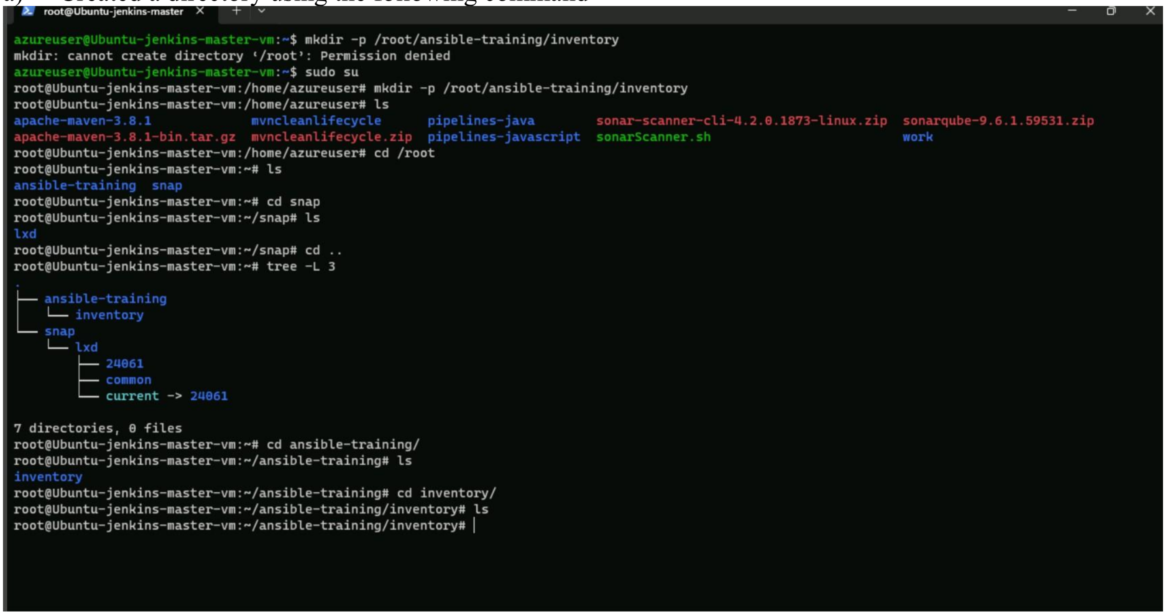
Sanghamitra Dasgupta
DevOps

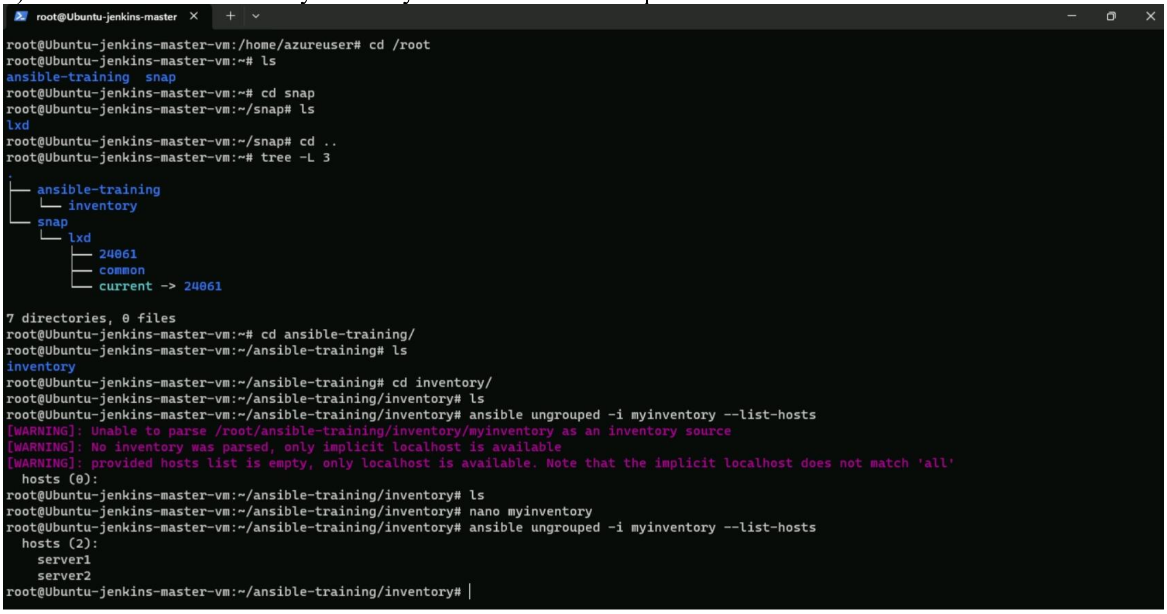
1. A

a) Created a directory using the following command



```
azureuser@Ubuntu-jenkins-master-vm:~$ mkdir -p /root/ansible-training/inventory
mkdir: cannot create directory ‘/root’: Permission denied
azureuser@Ubuntu-jenkins-master-vm:~$ sudo su
root@Ubuntu-jenkins-master-vm:/home/azureuser# mkdir -p /root/ansible-training/inventory
root@Ubuntu-jenkins-master-vm:/home/azureuser# ls
apache-maven-3.8.1             mvncleanlifecycle      pipelines-java        sonar-scanner-cli-4.2.0.1873-linux.zip  sonarqube-9.6.1.59531.zip
apache-maven-3.8.1-bin.tar.gz  mvncleanlifecycle.zip  pipelines-javascript  sonarScanner.sh                         work
root@Ubuntu-jenkins-master-vm:/home/azureuser# cd /root
root@Ubuntu-jenkins-master-vm:~# ls
ansible-training  snap
root@Ubuntu-jenkins-master-vm:~# cd snap
root@Ubuntu-jenkins-master-vm:~/snap# ls
lxd
root@Ubuntu-jenkins-master-vm:~/snap# cd ..
root@Ubuntu-jenkins-master-vm:~# tree -L 3
.
├── ansible-training
│   └── inventory
└── snap
    └── lxd
        ├── 24061
        ├── common
        └── current -> 24061

7 directories, 0 files
root@Ubuntu-jenkins-master-vm:~# cd ansible-training/
root@Ubuntu-jenkins-master-vm:~/ansible-training# ls
inventory
root@Ubuntu-jenkins-master-vm:~/ansible-training# cd inventory/
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# ls
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# |
```

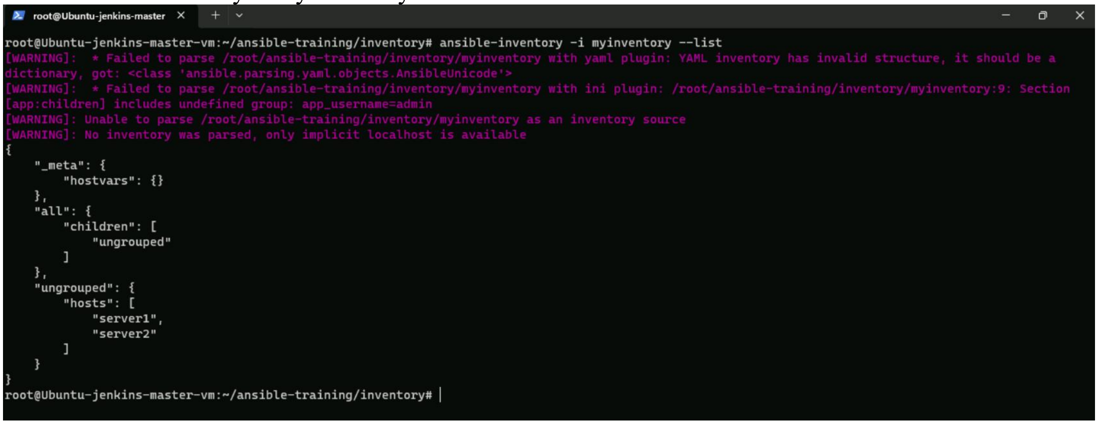
b) Created a file called myinventory and added the subsequent details



```
root@Ubuntu-jenkins-master-vm:/home/azureuser# cd /root
root@Ubuntu-jenkins-master-vm:~# ls
ansible-training  snap
root@Ubuntu-jenkins-master-vm:~# cd snap
root@Ubuntu-jenkins-master-vm:~/snap# ls
lxd
root@Ubuntu-jenkins-master-vm:~/snap# cd ..
root@Ubuntu-jenkins-master-vm:~# tree -L 3
.
├── ansible-training
│   └── inventory
└── snap
    └── lxd
        ├── 24061
        ├── common
        └── current -> 24061

7 directories, 0 files
root@Ubuntu-jenkins-master-vm:~# cd ansible-training/
root@Ubuntu-jenkins-master-vm:~/ansible-training# ls
inventory
root@Ubuntu-jenkins-master-vm:~/ansible-training# cd inventory/
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# ls
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# ansible ungrouped -i myinventory --list-hosts
[WARNING]: Unable to parse /root/ansible-training/inventory/myinventory as an inventory source
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'
  hosts (0):
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# ls
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# nano myinventory
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# ansible ungrouped -i myinventory --list-hosts
  hosts (2):
    server1
    server2
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# |
```

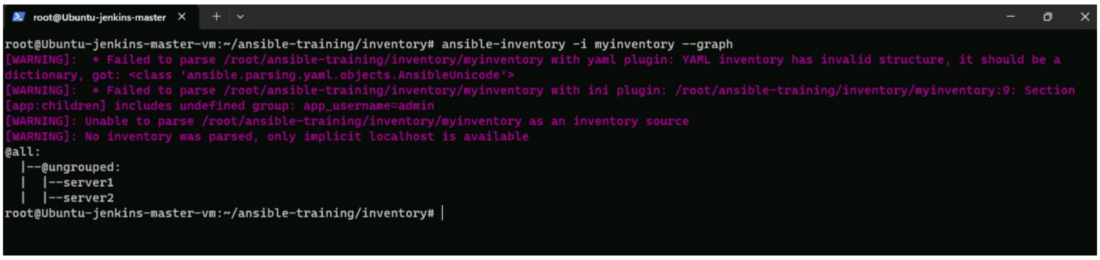
c) Once the above tasks were done the following commands needed to be executed

i. ansible-inventory -i myinventory --list



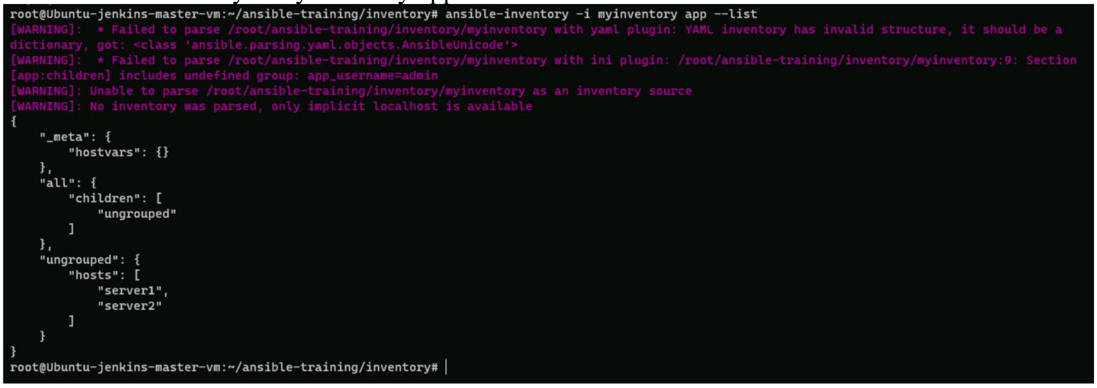
```
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# ansible-inventory -i myinventory --list
[WARNING]:  * Failed to parse /root/ansible-training/inventory/myinventory with yaml plugin: YAML inventory has invalid structure, it should be a
dictionary, got: <class 'ansible.parsing.yaml.objects.AnsibleUnicode'>
[WARNING]:  * Failed to parse /root/ansible-training/inventory/myinventory with ini plugin: /root/ansible-training/inventory/myinventory:9: Section
[app:children] includes undefined group: app_username=admin
[WARNING]: Unable to parse /root/ansible-training/inventory/myinventory as an inventory source
[WARNING]: No inventory was parsed, only implicit localhost is available
{
    "_meta": {
        "hostvars": {}
    },
    "all": {
        "children": [
            "ungrouped"
        ]
    },
    "ungrouped": {
        "hosts": [
            "server1",
            "server2"
        ]
    }
}
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# |
```

ii. ansible-inventory -i myinventory --graph



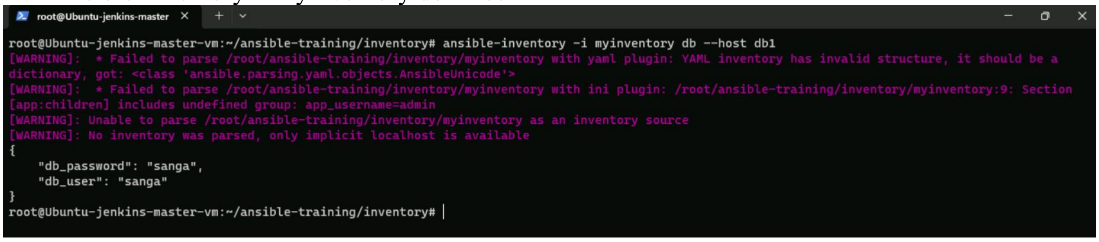
```
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# ansible-inventory -i myinventory --graph
[WARNING]:  * Failed to parse /root/ansible-training/inventory/myinventory with yaml plugin: YAML inventory has invalid structure, it should be a
dictionary, got: <class 'ansible.parsing.yaml.objects.AnsibleUnicode'>
[WARNING]:  * Failed to parse /root/ansible-training/inventory/myinventory with ini plugin: /root/ansible-training/inventory/myinventory:9: Section
[app:children] includes undefined group: app_username=admin
[WARNING]: Unable to parse /root/ansible-training/inventory/myinventory as an inventory source
[WARNING]: No inventory was parsed, only implicit localhost is available
@all:
  |--@ungrouped:
  |  |--server1
  |  |--server2
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# |
```

iii. ansible-inventory -i myinventory app --list



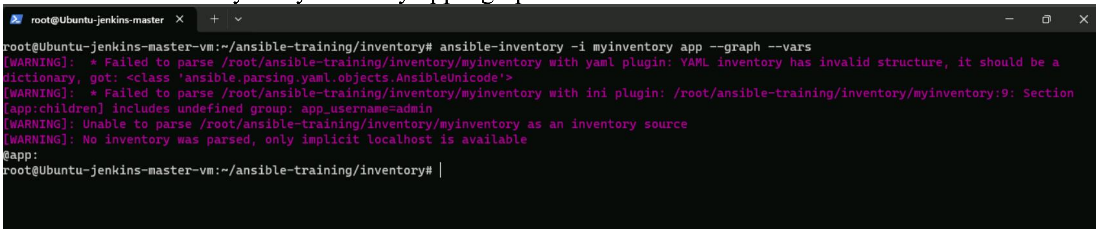
```
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# ansible-inventory -i myinventory app --list
[WARNING]:  * Failed to parse /root/ansible-training/inventory/myinventory with yaml plugin: YAML inventory has invalid structure, it should be a
dictionary, got: <class 'ansible.parsing.yaml.objects.AnsibleUnicode'>
[WARNING]:  * Failed to parse /root/ansible-training/inventory/myinventory with ini plugin: /root/ansible-training/inventory/myinventory:9: Section
[app:children] includes undefined group: app_username=admin
[WARNING]: Unable to parse /root/ansible-training/inventory/myinventory as an inventory source
[WARNING]: No inventory was parsed, only implicit localhost is available
{
    "_meta": {
        "hostvars": {}
    },
    "all": {
        "children": [
            "ungrouped"
        ]
    },
    "ungrouped": {
        "hosts": [
            "server1",
            "server2"
        ]
    }
}
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# |
```

iv. ansible-inventory -i myinventory db --host



```
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# ansible-inventory -i myinventory db --host db1
[WARNING]:  * Failed to parse /root/ansible-training/inventory/myinventory with yaml plugin: YAML inventory has invalid structure, it should be a
dictionary, got: <class 'ansible.parsing.yaml.objects.AnsibleUnicode'>
[WARNING]:  * Failed to parse /root/ansible-training/inventory/myinventory with ini plugin: /root/ansible-training/inventory/myinventory:9: Section
[app:children] includes undefined group: app_username=admin
[WARNING]: Unable to parse /root/ansible-training/inventory/myinventory as an inventory source
[WARNING]: No inventory was parsed, only implicit localhost is available
{
    "db_password": "sanga",
    "db_user": "sanga"
}
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# |
```

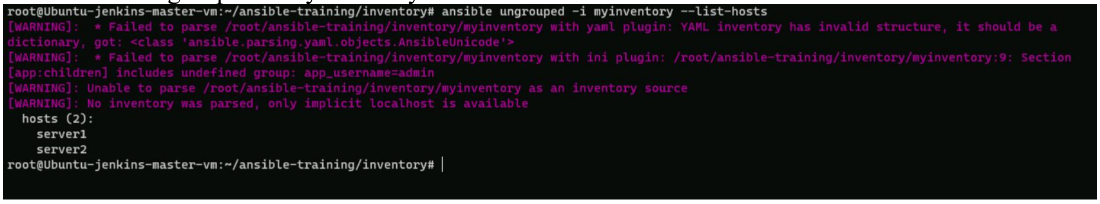
v. ansible-inventory -i myinventory app --graph --vars



```
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# ansible-inventory -i myinventory app --graph --vars
[WARNING]:  * Failed to parse /root/ansible-training/inventory/myinventory with yaml plugin: YAML inventory has invalid structure, it should be a
dictionary, got: <class 'ansible.parsing.yaml.objects.AnsibleUnicode'>
[WARNING]:  * Failed to parse /root/ansible-training/inventory/myinventory with ini plugin: /root/ansible-training/inventory/myinventory:9: Section
[app:children] includes undefined group: app_username=admin
[WARNING]: Unable to parse /root/ansible-training/inventory/myinventory as an inventory source
[WARNING]: No inventory was parsed, only implicit localhost is available
@app:
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# |
```

vi. ansible all-i myinventory --list-hosts

vii. ansible ungrouped -i myinventory --list-hosts

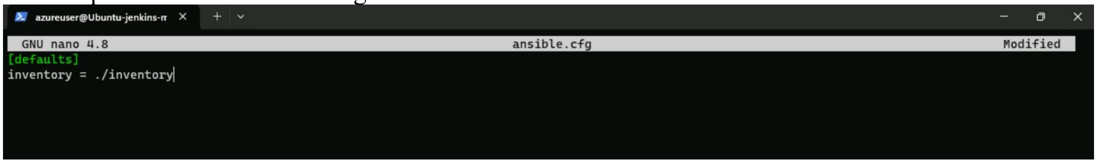


```
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# ansible ungrouped -i myinventory --list-hosts
[WARNING]:  * Failed to parse /root/ansible-training/inventory/myinventory with yaml plugin: YAML inventory has invalid structure, it should be a
dictionary, got: <class 'ansible.parsing.yaml.objects.AnsibleUnicode'>
[WARNING]:  * Failed to parse /root/ansible-training/inventory/myinventory with ini plugin: /root/ansible-training/inventory/myinventory:9: Section
[app:children] includes undefined group: app_username=admin
[WARNING]: Unable to parse /root/ansible-training/inventory/myinventory as an inventory source
[WARNING]: No inventory was parsed, only implicit localhost is available
  hosts (2):
    server1
    server2
root@Ubuntu-jenkins-master-vm:~/ansible-training/inventory# |
```

2. Customising Ansible Environment

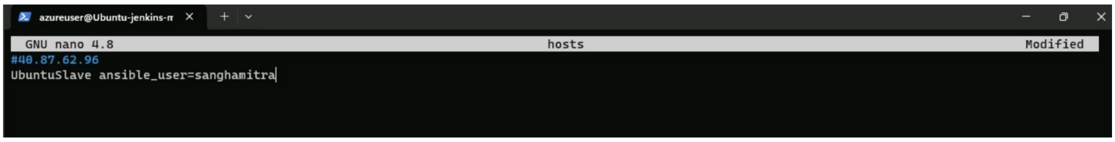
a) First we create an adhoc directory

b) Next up we create an ansible.cfg file in it and add a defaults section to it



```
  GNU nano 4.8                                ansible.cfg                                Modified
[defaults]
inventory = ./inventory
```

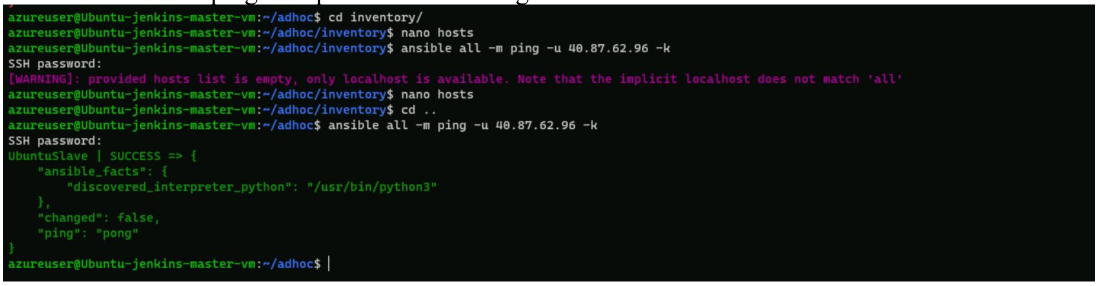
c) Now we create a directory called *inventory* and add the ip address of our manage node



```
  GNU nano 4.8                                hosts                                      Modified
#40.87.62.96
UbuntuSlave ansible_user=sanghamitra
```

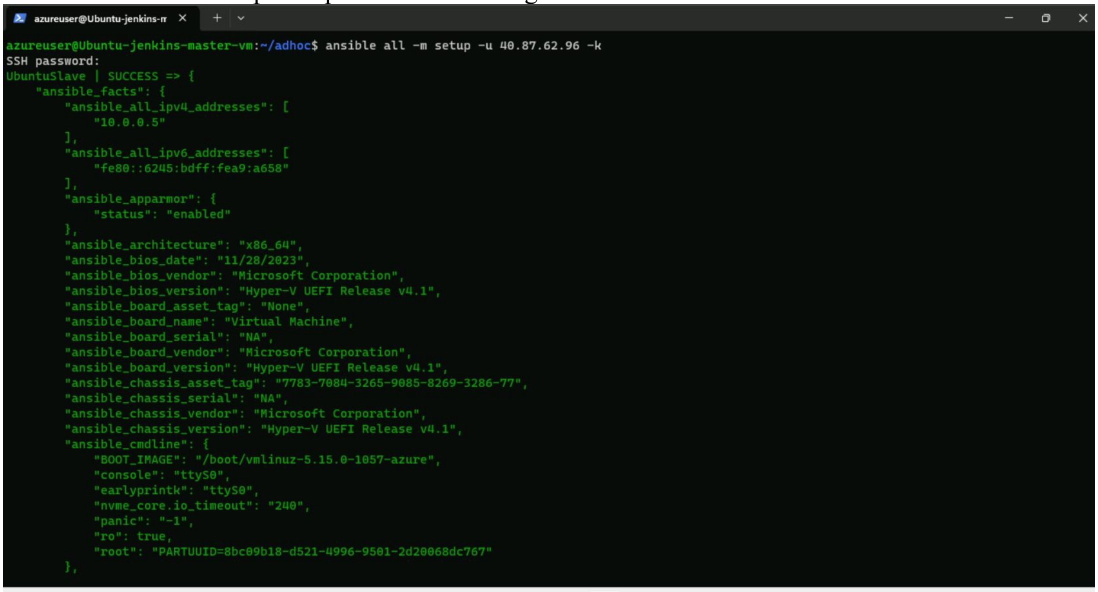
d) Now we need to run the following commands:

i. ansible all -m ping -u <ip-address-of-manage-node> -k



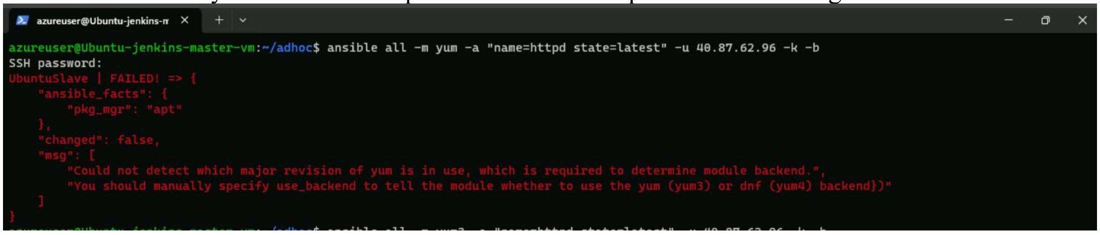
```
azureuser@Ubuntu-jenkins-master-vm:~/adhoc$ cd inventory/
azureuser@Ubuntu-jenkins-master-vm:~/adhoc/inventory$ nano hosts
azureuser@Ubuntu-jenkins-master-vm:~/adhoc/inventory$ ansible all -m ping -u 40.87.62.96 -k
SSH password:
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'
azureuser@Ubuntu-jenkins-master-vm:~/adhoc/inventory$ nano hosts
azureuser@Ubuntu-jenkins-master-vm:~/adhoc/inventory$ cd ..
azureuser@Ubuntu-jenkins-master-vm:~/adhoc$ ansible all -m ping -u 40.87.62.96 -k
SSH password:
UbuntuSlave | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
azureuser@Ubuntu-jenkins-master-vm:~/adhoc$ |
```

ii. ansible all -m setup -u <ip-address-of-manage-node> -k



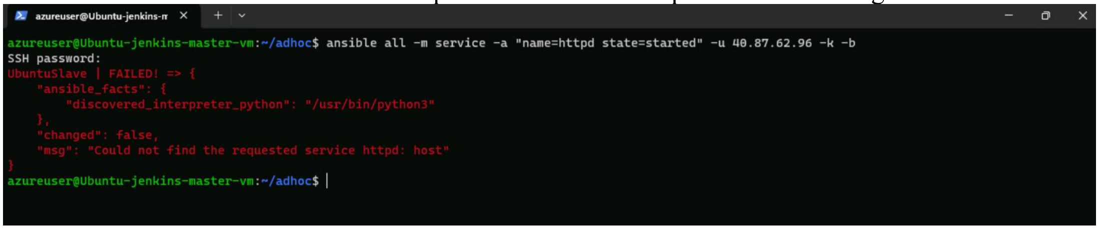
```
azureuser@Ubuntu-jenkins-master-vm:~/adhoc$ ansible all -m setup -u 40.87.62.96 -k
SSH password:
UbuntuSlave | SUCCESS => {
    "ansible_facts": {
        "ansible_all_ipv4_addresses": [
            "10.0.0.5"
        ],
        "ansible_all_ipv6_addresses": [
            "fe80::6245:bdff:fea9:a658"
        ],
        "ansible_apparmor": {
            "status": "enabled"
        },
        "ansible_architecture": "x86_64",
        "ansible_bios_date": "11/28/2023",
        "ansible_bios_vendor": "Microsoft Corporation",
        "ansible_bios_version": "Hyper-V UEFI Release v4.1",
        "ansible_board_asset_tag": "None",
        "ansible_board_name": "Virtual Machine",
        "ansible_board_serial": "NA",
        "ansible_board_vendor": "Microsoft Corporation",
        "ansible_board_version": "Hyper-V UEFI Release v4.1",
        "ansible_chassis_asset_tag": "7783-7084-3265-9085-8269-3286-77",
        "ansible_chassis_serial": "NA",
        "ansible_chassis_vendor": "Microsoft Corporation",
        "ansible_chassis_version": "Hyper-V UEFI Release v4.1",
        "ansible_cmdline": {
            "BOOT_IMAGE": "/boot/vmlinuz-5.15.0-1057-azure",
            "console": "ttyS0",
            "earlyprintk": "ttyS0",
            "nvme_core.io_timeout": "240",
            "panic": "-1",
            "ro": true,
            "root": "PARTUUID=8bc09b18-d521-4996-9501-2d20068dc767"
        },
```

iii. ansible all -m yum -a “name=httpd state=latest” -u <ip-address-of-manage-node> -k -b



```
azureuser@Ubuntu-jenkins-master-vm:~/adhoc$ ansible all -m yum -a "name=httpd state=latest" -u 40.87.62.96 -k -b
SSH password:
UbuntuSlave | FAILED! => {
    "ansible_facts": {
        "pkg_mgr": "apt"
    },
    "changed": false,
    "msg": [
        "Could not detect which major revision of yum is in use, which is required to determine module backend.",
        "You should manually specify use_backend to tell the module whether to use the yum (yum3) or dnf (yum4) backend})"
    ]
}
```

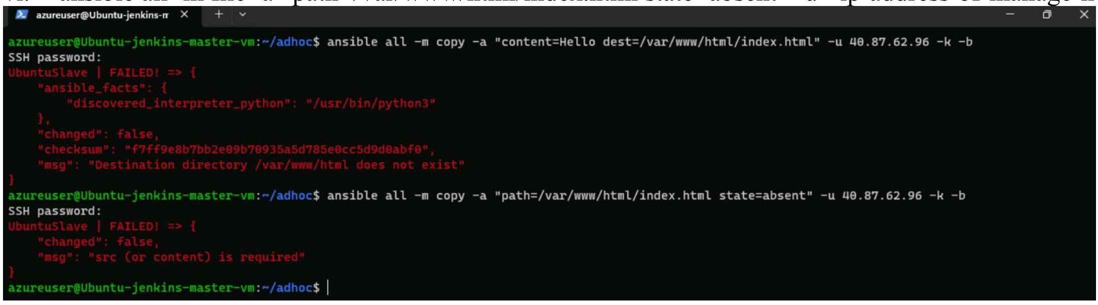
iv. ansible all -m service -a “name=httpd state=started” -u <ip-address-of-manage-node> -k -b



```
azureuser@Ubuntu-jenkins-master-vm:~/adhoc$ ansible all -m service -a "name=httpd state=started" -u 40.87.62.96 -k -b
SSH password:
UbuntuSlave | FAILED! => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "msg": "Could not find the requested service httpd: host"
}
azureuser@Ubuntu-jenkins-master-vm:~/adhoc$ |
```

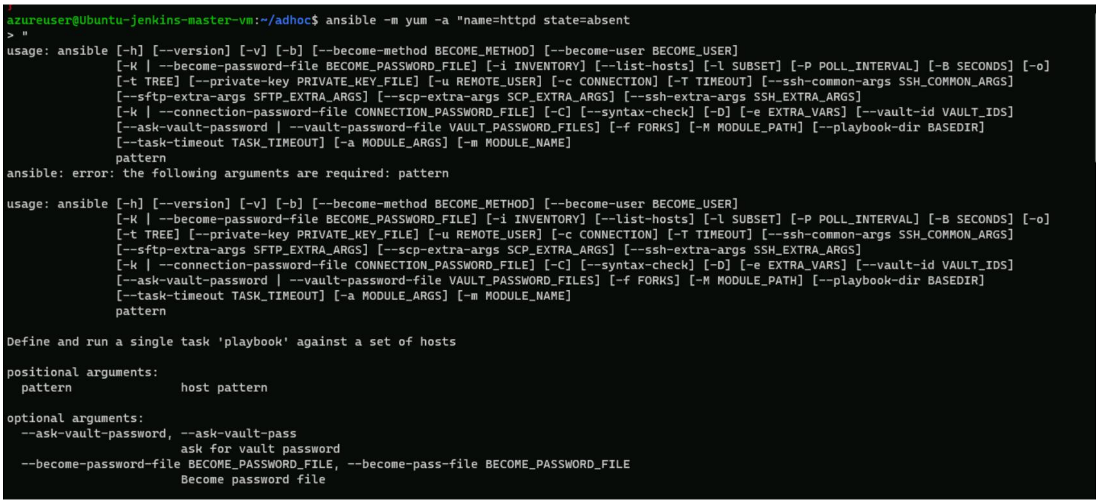
v. ansible all -m copy -a “content=Hello dest=/var/www/html/index.html” -u <ip-address-of-manage-node> -k -b

vi. ansible all -m file -a “path=/var/www/html/index.html state=absent” -u <ip-address-of-manage-node> -k -b



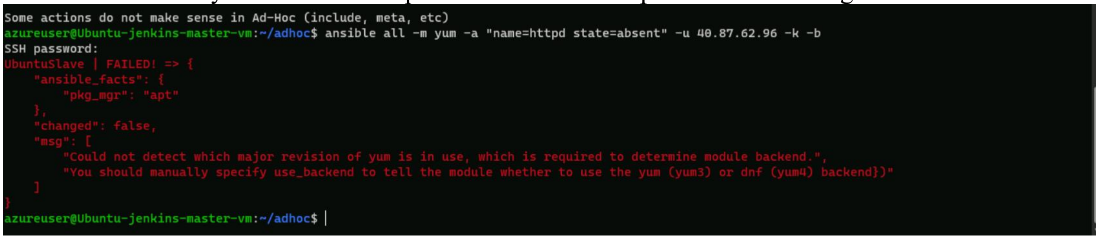
```
azureuser@Ubuntu-jenkins-master-vm:~/adhoc$ ansible all -m copy -a "content=Hello dest=/var/www/html/index.html" -u 40.87.62.96 -k -b
SSH password:
UbuntuSlave | FAILED! => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "checksum": "f7ff9e8b7bb2e09b70935a5d785e0cc5d9d0abf0",
    "msg": "Destination directory /var/www/html does not exist"
}
azureuser@Ubuntu-jenkins-master-vm:~/adhoc$ ansible all -m copy -a "path=/var/www/html/index.html state=absent" -u 40.87.62.96 -k -b
SSH password:
UbuntuSlave | FAILED! => {
    "changed": false,
    "msg": "src (or content) is required"
}
azureuser@Ubuntu-jenkins-master-vm:~/adhoc$ |
```

vii. ansible -m yum -a “name=httpd state=absent”



```
azureuser@Ubuntu-jenkins-master-vm:~/adhoc$ ansible -m yum -a "name=httpd state=absent
> "
usage: ansible [-h] [--version] [-v] [-b] [--become-method BECOME_METHOD] [--become-user BECOME_USER]
               [-K | --become-password-file BECOME_PASSWORD_FILE] [-i INVENTORY] [--list-hosts] [-l SUBSET] [-P POLL_INTERVAL] [-B SECONDS] [-o]
               [-t TREE] [--private-key PRIVATE_KEY_FILE] [-u REMOTE_USER] [-c CONNECTION] [-T TIMEOUT] [--ssh-common-args SSH_COMMON_ARGS]
               [--sftp-extra-args SFTP_EXTRA_ARGS] [--scp-extra-args SCP_EXTRA_ARGS] [--ssh-extra-args SSH_EXTRA_ARGS]
               [-k | --connection-password-file CONNECTION_PASSWORD_FILE] [-C] [--syntax-check] [-D] [-e EXTRA_VARS] [--vault-id VAULT_IDS]
               [--ask-vault-password | --vault-password-file VAULT_PASSWORD_FILES] [-f FORKS] [-M MODULE_PATH] [--playbook-dir BASEDIR]
               [--task-timeout TASK_TIMEOUT] [-a MODULE_ARGS] [-m MODULE_NAME]
               pattern
ansible: error: the following arguments are required: pattern

usage: ansible [-h] [--version] [-v] [-b] [--become-method BECOME_METHOD] [--become-user BECOME_USER]
               [-K | --become-password-file BECOME_PASSWORD_FILE] [-i INVENTORY] [--list-hosts] [-l SUBSET] [-P POLL_INTERVAL] [-B SECONDS] [-o]
               [-t TREE] [--private-key PRIVATE_KEY_FILE] [-u REMOTE_USER] [-c CONNECTION] [-T TIMEOUT] [--ssh-common-args SSH_COMMON_ARGS]
               [--sftp-extra-args SFTP_EXTRA_ARGS] [--scp-extra-args SCP_EXTRA_ARGS] [--ssh-extra-args SSH_EXTRA_ARGS]
               [-k | --connection-password-file CONNECTION_PASSWORD_FILE] [-C] [--syntax-check] [-D] [-e EXTRA_VARS] [--vault-id VAULT_IDS]
               [--ask-vault-password | --vault-password-file VAULT_PASSWORD_FILES] [-f FORKS] [-M MODULE_PATH] [--playbook-dir BASEDIR]
               [--task-timeout TASK_TIMEOUT] [-a MODULE_ARGS] [-m MODULE_NAME]
               pattern

Define and run a single task 'playbook' against a set of hosts

positional arguments:
  pattern               host pattern

optional arguments:
  --ask-vault-password, --ask-vault-pass
                        ask for vault password
  --become-password-file BECOME_PASSWORD_FILE, --become-pass-file BECOME_PASSWORD_FILE
                        Become password file
```

viii. ansible all -m yum -a “name=httpd state=absent” -u <ip-address-of-manage-node> -k -b



```
Some actions do not make sense in Ad-Hoc (include, meta, etc)
azureuser@Ubuntu-jenkins-master-vm:~/adhoc$ ansible all -m yum -a "name=httpd state=absent" -u 40.87.62.96 -k -b
SSH password:
UbuntuSlave | FAILED! => {
    "ansible_facts": {
        "pkg_mgr": "apt"
    },
    "changed": false,
    "msg": [
        "Could not detect which major revision of yum is in use, which is required to determine module backend.",
        "You should manually specify use_backend to tell the module whether to use the yum (yum3) or dnf (yum4) backend})"
    ]
}
azureuser@Ubuntu-jenkins-master-vm:~/adhoc$ |
```