Step - 1:

Need to install CentOS on 3 three VM. One is for the master node, one is for the Web Server (frontend & backend) and another is for Database. Remember we need to disable the selinux.

Step - 2:

Need to update packages of all VM by the below command:

 # yum update -y

Step - 3:

Now we will install Jenkins on the master VM.

1. Jenkins is a java program so firstly we need to install java on the master VM.

   # yum install java-11-openjdk.x86_64 -y

2. To install the latest stable version of Jenkins , we have to add the official Jenkins repository to the system. Execute the below commands to add the key and repo.

   # yum install wget -y

   # sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo

   # sudo rpm --import https://pkg.jenkins.io/redhat/jenkins.io-2023.key

3. Now install, start and enable jenkins by below commands

   # yum install jenkins -y
   # systemctl start jenkins
   # systemctl enable jenkins

   Note: If you want to hit your jenkins URL first you need to stop firewalld on centos or need to enable the port on firewalld. Otherwise you can not access jenkins dashboard on browser.

Step - 3:

Now we will install Ansible on the master VM.

1. Firstly we need to install the epel-release package so that we can install ansible from that package. For this we need to run the below command:
   # sudo yum install epel-release
2. Now we will install ansible by below command:
   # yum install ansible

Step - 4:

We will add ansible clients which is our Web Server VM, Database Server VM ip address in ansible configuration file.

 # sudo vim /etc/ansible/hosts

[webservers]

192.168.20.15

[dbservers]

192.168.20.14

Step - 5:

We need to copy the ssh key of the master server in ansible client machines. So that the Ansible server to its nodes communication will be passwordless.

 # ssh-keygen

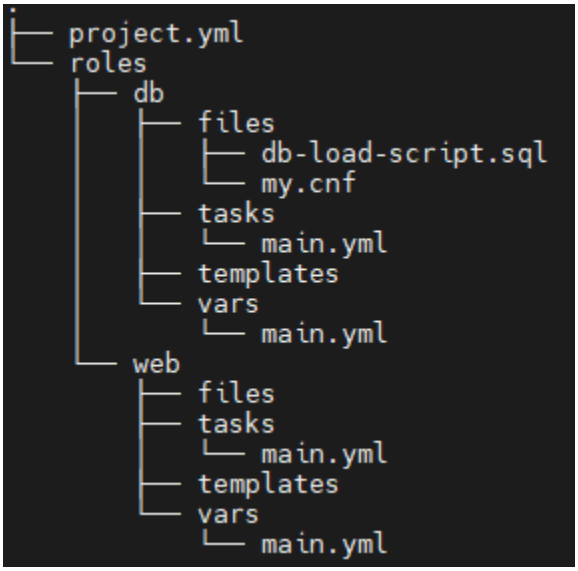
 # ssh-copy-id root@192.168.20.15 [Web Server IP]

 # ssh-copy-id root@192.168.20.14 [Database Server IP]

Step - 6:

Now we will create an ansible-playbook to install, start mandatory services in node.

Firstly we need to create files (httpd_project.yml) and a directory (roles) in the same directory. Under the "roles" directory we created another directory which is called "web" and "db". Under this "web" and "db" directory we created four directory (vars, tasks, templates, files). We created "main.yml" files in the vars and tasks directory. In the files of "db" directory we created two files "db-load-script.sql" and "my.cnf".



 # Now in the “project.yml” file we have added the below lines:

```
---
  - name: DB Service
    hosts: dbservers
    roles:
      - db

  - name: Web Service
    hosts: webservers
    roles:
      - web
```

Note: We have provided the host name and roles information in this files. So it will go to “web” and “db” roles and execute the tasks

 # We have mentioned the below lines in “roles/web/tasks/main.yml”.

```
- name: Installation Services
  yum:
    name:
      - libselinux-python
      - libsemanage-python
      - httpd
      - git
      - php
      - php-mysql
    state: installed
  tags: install
```

Note: Installing mandatory services

```
- name: Start firewalld
  service: name=firewalld state=started enabled=yes
  tags: start firewalld
```

Note: start firewall

```
- name: Insert firewalld rule for httpd
  firewalld: port={{ httpd_port }}/tcp permanent=true state=enabled immediate=yes
  tags: enable httpd port
```

Note: enable httpd port in firewall

```
- name: insert firewalld rule for mysql
  firewalld: port={{ mysql_port }}/tcp permanent=true state=enabled immediate=yes
  tags: enable mysql port
```
Note: enable mysql port in firewall

```
- name: Set index.php as the default page
  replace:
    path: /etc/httpd/conf/httpd.conf
    regexp: 'DirectoryIndex index.html'
    replace: '#DirectoryIndex index.html \nDirectoryIndex index.php'
  tags: rename html file
```

Note: Rename html file to php in configuration file

```
- name: http service state
  service: name=httpd state=started enabled=yes
  tags: httpd start
```

Note: Starting httpd service

```
- name: Copy the code from repository
  git: repo={{ repository }} dest=/var/www/html/ force=yes
  tags: clone
```

Note: Clone repository

```
- name: replace ip in index.php file
  command: sed -i 's/172.20.1.101/192.168.20.14/g' /var/www/html/index.php
  tags: replace IP
```

Note: Replace the IP

 # We added the below lines in “roles/web/vars/main.yml” file

```
---
httpd_port: 80
mysql_port: 3306
repository: https://github.com/Debaice06/Ansible_Projects_Ecommerce.git
```

Note: We have mentioned the variable in this file

 # We have mentioned the below lines in "roles/db/tasks/main.yml".

```
- name: Installation Services
  yum:
    name:
      - libselinux-python
      - libsemanage-python
      - mariadb-server
      - MySQL-python
      - php-mysql
    state: installed
  tags: install

- name: Start firewalld
  service: name=firewalld state=started enabled=yes
  tags: start firewalld

- name: Insert firewalld rule for mysql
  firewalld: port={{ mysql_port }}/tcp permanent=true state=enabled immediate=yes
  tags: enable mysql port

- name: Restart firewalld
  service: name=firewalld state=reloaded enabled=yes
  tags: restarted firewalld

- name: Copy Mysql configuration file
  copy: src=files/my.cnf dest=/etc/my.cnf
  tags: mysql conf copy

- name: Start MariaDB Service
  service: name=mariadb state=started enabled=yes
  tags: start mariadb

- name: Create Application Database
  mysql_db: name={{ dbname }} state=present
  tags: create database
```

```
- name: Create Application DB User
  mysql_user: name={{ dbuser }} password={{ dbpassword }} priv=*.*:ALL
host='192.168.20.14' state=present
  tags: create user

- name: Move db-load-script to db host
  copy:
    src: files/db-load-script.sql
    dest: /tmp/db-load-script.sql
  tags: copy sql

- name: Load Inventory Data
  shell: mysql -f < /tmp/db-load-script.sql
  tags: run sql
```

 # We have mentioned the below lines in “roles/db/vars/main.yml”.

```
---
mysql_port: 3306
dbname: ecomdb
dbuser: ecomuser
dbpassword: ecompassword
```

 # We have mentioned the below lines in “roles/db/files/db-load-script.sql”

```
GRANT ALL PRIVILEGES ON *.* TO 'ecomuser'@'192.168.20.15' IDENTIFIED BY
'ecompassword' WITH GRANT OPTION;
FLUSH PRIVILEGES;
USE ecomdb;
CREATE TABLE products (id mediumint(8) unsigned NOT NULL auto_increment,Name
varchar(255) default NULL,Price varchar(255) default NULL, ImageUrl varchar(255) default
NULL,PRIMARY KEY (id)) AUTO_INCREMENT=1;
INSERT INTO products (Name,Price,ImageUrl) VALUES ("Laptop","100","c-1.png"),
("Drone","200","c-2.png"),("VR","300","c-3.png"),("Tablet","50","c-5.png"),("Watch","90","c6.
png"),("Phone Covers","20","c-7.png"),("Phone","80","c-8.png"),("Laptop","150","c-4.png");
```

 # We have mentioned the below lines in “roles/db/files/db-load-script.sql”

```
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
symbolic-links=0
[mysqld_safe]
log-error=/var/log/mariadb/mariadb.log
pid-file=/var/run/mariadb/mariadb.pid
!includedir /etc/my.cnf.d
```

Step - 7:

Now run ansible-playbook by below command

 # ansible-playbook project.yml