

Day 10

Ansible tools

Server Machine -> to deploy the application

Web application we need server.

Tomcat, web logic, jboss, Websphere, IIS, nginx, Apache etc.

Web Server or Application Server.

ie java, python, express js, angular application , react js

Database Server

MySQL, Mongo Db, Oracle, Db2 etc.

Node or Machine : if installed all required software.

We need to create more than one machine under one cluster environment.

10 machine --> provide service for Java application

Java7 working and using tomcat 7.

Java8: we need to update java8 in all machine

Using ansible tool we connect all those machine and we can install all required software, database, server through ansible playbook.

10 host machine

4 -> java software

java_host

All four machine ip address .

3 -> python software

Python_host

All three machine ip address

3 -> node js software

Node_host

All three machine ip address

In local machine or VM machine

Create simple index.html page and create Dockerfile using nginx

then push this code in git hub account.

Vm machine

Then create Jenkin pipe line job which is responsible to pull the project

And re-build and push the image to docker hub account whenever we push the

New changes in git hub account.

We can view new changes in browser.

EC2 instance

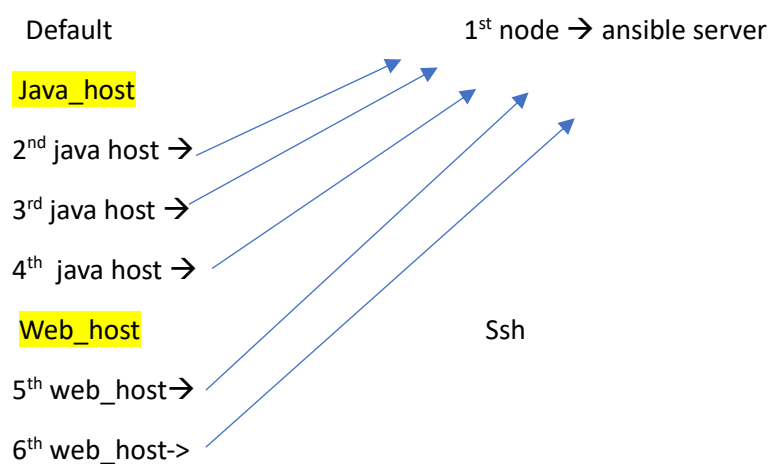
Then create Jenkin pipe line job which is responsible to pull the project

And re-build and push the image to docker hub account whenever we push the

New changes in git hub account.

We can view new changes in browser.

We will create totally 6 nodes



Create 6 instance

And in ansible server instance install software

sudo su

to move root user

or

sudo - i

please install these command in ansible server instance

sudo apt-get update

sudo apt-get install software-properties-common

sudo apt-add-repository ppa:ansible/ansible

sudo apt-get update

sudo apt-get install ansible

Ansible provide modules which contains set of function which help to execute more than one service in different host machine. Those host machine provided in inventory file.

Using ansible if we want to do some task we can achieve using 2 ways.

1. Ansible ad hoc : this option is useful if we want to do any single task then we use ansible ad hoc.
2. Ansible play book : if we want to do more than one task then we use ansible play book. All those instruction we will write in a .yaml file

Ansible command module

Using ad hoc commands.

ansible all -m shell -a "date"

copy module

ansible all -m copy -a "src=info dest=/home/ansible"

now we copy and paste file in particular group host name

ansible java_host -m copy -a "src=java_file dest=/home/ansible"

file create module

```
ansible web_host -m file -a "path=/home/ansible/index.html state=touch"
```

We will create demo.java program in java host machines using ansible playbook.

Ansible command module

Using ad hoc commands.

```
ansible all -m shell -a "date"
```

copy module

```
ansible all -m copy -a "src=info dest=/home/ansible"
```

now we copy and paste file in particular group host name

```
ansible java_host -m copy -a "src=java_file dest=/home/ansible"
```

file create module

```
ansible web_host -m file -a "path=/home/ansible/index.html state=touch"
```

```
vi java_file.yml
```

```
---
```

```
- hosts: java_host
```

become: true

tasks:

- name: creating the file

file: path=/home/ansible/demo.java state=touch

verify file details using command as

cat java_file.yml

ansible-playbook java_file.yml

installing java using ad hocs

ansible java_host -m apt -a "name=openjdk-8-jdk state=present" --become

installing nginx server

vi nginx_server_start.yml

- hosts: web_host

become: true

tasks:

- name: install nginx server

apt: name=nginx state=latest

- name: start nginx servver

service:

name: nginx

state: started

```
vi nginx_stop_uninstall.yml
```

```
- hosts: web_host
```

```
  become: true
```

```
  tasks:
```

```
    - name: stop nginx
```

```
      service:
```

```
        name: nginx
```

```
        state: stopped
```

```
    - name: uninstall nginx
```

```
      apt: name=nginx state=absent
```