**Docker**

* Docker is a set of platform as a service product that use OS-level virtualization to deliver software in packages called containers.
* It is designed in order to make the process Create, Deploy and Run application using “containers” easily.

**Why we use Docker:**

* Sometimes, the application will work properly in developers machine but when developer try to move to production environment it will not work due to dependencies problem, software configuration problem etc.
* To overcome this problem we use Docker.

**Difference between Docker and VM:**

|  |  |
| --- | --- |
| **Docker** | **VM** |
| For different layers of application different containers are used. | For different layers of application different virtual machines are used. |
| All containers shares the single host OS. | Each VM will have its own guest OS. |
| Use of single OS will make the process faster. | Use of multiple OS will result in slowing the process. |
| No waste of money and resources. | Waste of money and resources. |

**Commands:**

**docker images**

- List all images.

**docker image ls**

- List all images.

**docker ps**

- List all running containers.

**docker container ls -a**

- List all stopped and running containers.

**docker ps -q**

- List all running containers ID.

**docker ps -l -q**

- Displays latest running container ID.

**docker ps -a**

- List all containers.

**docker container rm container\_name/id**

- To remove a container.

**docker container rmi image\_name/id**

- To remove a image.

**docker container stop container\_name/id**

- To stop a container.

.

**docker pull “image name”**

- To pull images from docker hub repository.

**docker container create container\_name**

- To create a container.

**docker start container\_name**

- To start a container

**docker commit container\_name “Image\_name”**

- Allows user to take a running container and save its current state as an image.

**docker run -d -it –name “container name” image\_name**

- To run a image and create a container in detach mode.

**docker build -t ‘image name” .**

- To build an image.

**docker exec -it container\_name/id bash**

- To open a running container bash.

**docker container inspect “container\_name”**

- To display information about container.

**GIT**

**git config -global user.name “name”**

- Command to set name to use with commits.

**git config -global user.email “email address”**

- Command to set email address to use with commits.

**git init**

- Creates a empty local repository.

**git clone “url”**

- To clone a remote repository.

**git add “file name”**

- To stage a file.

**git add .**

- To stage multiple files.

**git commit -m “message”**

- To commit staged files.

**git status**

- Displays files that has to be committed.

**git push “variable name” master**

- To push files to master branch of remote repository.

**git pull**

- It pull files from remote repository to local repository.

**git branch “branch name”**

- Creates a new branch

**git checkout “branch name”**

- To move to another branch.

**git checkout -b “branch name”**

- To create new branch and move to it.

**git merge “branch name”**

- merge

**Ansible**

**sudo apt install ansible**

- To install ansible

**sudo apt install software-properties-common**

**sudo add-apt-repository –yes –update ppa:ansible/ansible**

**ssh-keygen**

- To generate key

**ssh-copy-id** [**username@ipaddress**](mailto:username@ipaddress)

- To connect to node VM

**Playbooks:**

**Install apache and update packages**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

---

- name: install httpd packages and update sample webpage and start service

hosts: all

become: yes # root user going to excute

tasks:

- name: Update all packages to their latest version

apt:

name: "\*"

state: latest

- name: Install apache2 httpd (state=present is optional)

apt:

name: apache2

state: present

- name: Copy index test page

copy:

src: "index.html"

dest: "/var/www/index.html"

force: yes

- name: Restart Apache

service:

name: apache2

state: restarted

**Git Install:**

**\*\*\*\*\*\*\*\*\***

---

- name: Install git on all nodes

hosts: all

become: yes

tasks:

- name: Install Git

apt:

name: git

state: present

update\_cache: yes

**To Create Multiple Users in a Node VM using Loop:**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

---

- name: learn loop inside playbook

hosts: all

become: yes

tasks:

- name: Add several users on nodes

ansible.builtin.user:

name: "{{ item }}"

state: present

loop:

- testuser1

- testuser2

- nighp

- foo

**Declaring and Initializing a Variable:**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

---

- name: learn varibale declation and usage inside playbook

hosts: all

become: yes

vars:

package: apache2

user\_allowed: true

tasks:

- name: We are uninstalling {{package}}

apt:

name: { { package } }

state: absent

- name: should not present {{package}}

service:

name: apache2

state: stopped