**Basic Commands**

1. **docker version**

• Displays the Docker version installed on the system.

**2. docker-v**

• Short form of docker version. It shows the Docker version.

**3. docker info**

• Provides detailed information about the Docker installation

**4. docker --help**

Displays general help information

Example: To get information about specific commands:

• docker images --help: Details about managing images

• docker run --help: Details about running containers,

**5.docker login**

Logs into a Docker registry, such as Docker Hub. Used for push or pull docker images from Docker Hub.

###############################################################################

Images Commands

**6. docker images**

* Lists all the Docker images present on the machine

**7. docker pull**

• Pulls an image from a Docker registry. You can find Docker images here:

<https://hub.docker.com/search?q=&type=image>

Example: docker pull ubuntu

**8. docker rmi**

Removes Docker images

• docker images -q: Lists image IDs.

* docker rmi <image ID>: Deletes the specified image.
* After deletion, confirm with docker images.

Container Commands

**9. docker ps & docker run**

• docker ps: Lists running containers

* docker run <image name >: Creates a container from a specified image if local image is not available then it will pull from Docker Hub automatically

• Example: docker run ubuntu.

• Example: docker run -it ubuntu //For interaction

**10. docker start**

* Starts a stopped container.

• Example: docker start <container id>

**11. docker stop**

• Stops a running container.

 • Example: docker stop <container id>,

**12. docker rm**

* Removes a container.

Example: docker rm <container id or name>

System commands

12. docker stats

• Provides resource usage statistics for running containers, such as CPU, memory, etc.

13. docker system df

• Displays disk usage related to Docker.

14. docker system prune

• Cleans up unused data, such as stopped containers

• docker system prune -f. Forcefully removes all stopped containers

These commands form the basic toolkit for managing Docker containers and images, as well as maintaining the Docker environment.

##########################################################3

Pull Docker Images

1. Pull Selenium-hub image using command

* docker pull selenium/hub

1. Pull FireFox image using command

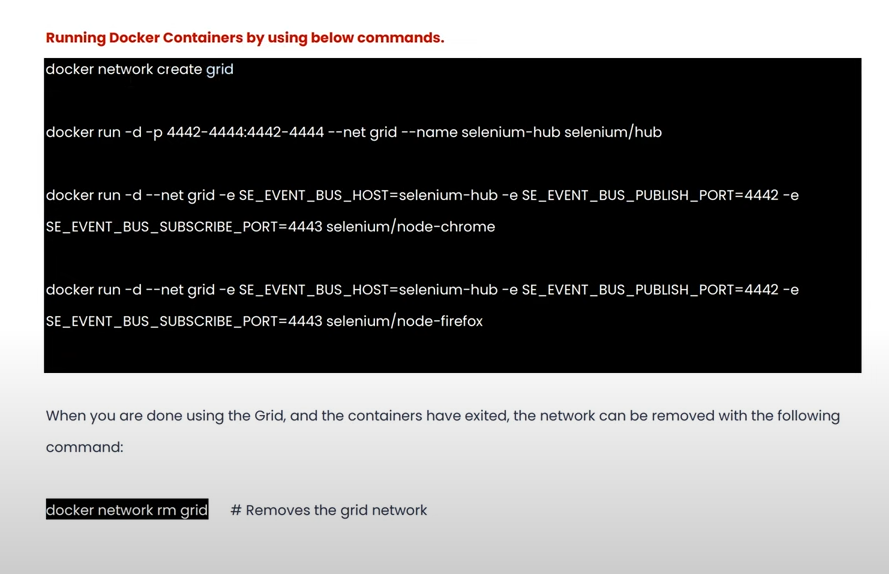
* docker pull selenium/node-firefox

1. Pull Chrome image using below command

* docker pull selenium/node-chrome

Verify Images

* docker images



In the above screenshot, we need to run all the cmds for every os and browser, and browsers and OS keep on increasing, to avoid lots of cmds for every browser, we have yaml file configured.

1. Import the file from Git(pavanSDET), and place under project
2. To run the yaml file -🡪 go to the path of yaml file placed in your system, from there run in cmd
3. Docker-compose up --🡪 to run yaml file
4. execute the tc FROM **grid-docker.xml**
5. To check if hub and node in running state-🡪 http://localhost:4444/grid/console
6. To increase number of nodes : docker-compose scale chrome=3(it is useful, when we want to run in chrome for 2 tc)
7. To stop the grid and cleanup the created containers, run: docker-compose down