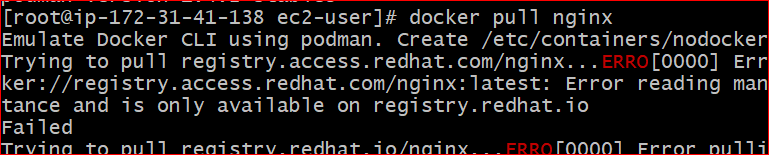
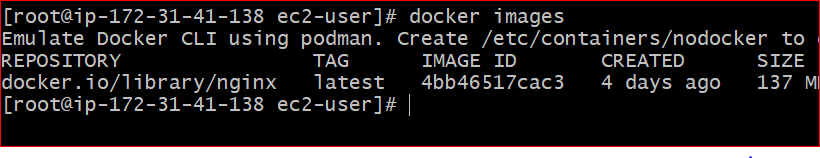
Find a image from dockerhub of your choice(recommeded: nginx), don't use browser, pull the official image from dockerhub

docker pull nginx



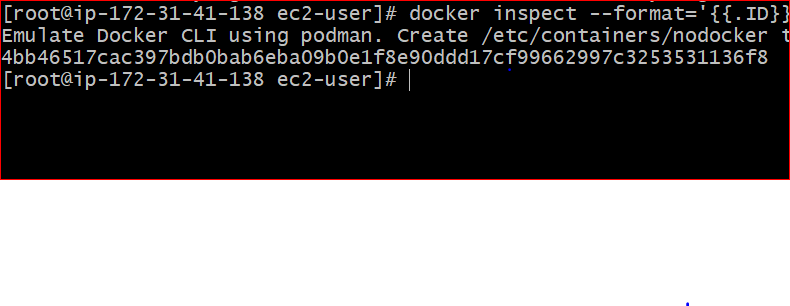
List all the available images in your machine/vm, make sure you see recently pulled image in the list

Docker images



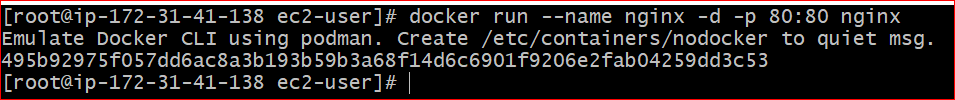
Find out the "Full" ImageId of the image that you pulled and write it below.

docker inspect --format='{{.ID}}' 4bb46517cac3



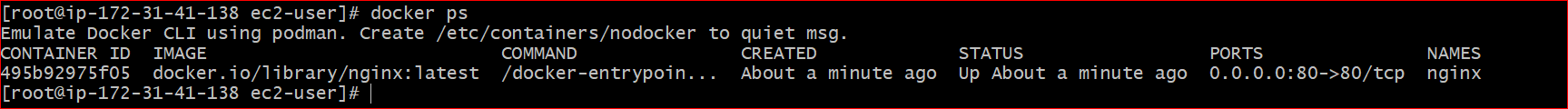
Create a container of your image

docker run --name nginx -d -p 80:80 nginx



List all the running containers

Docker ps

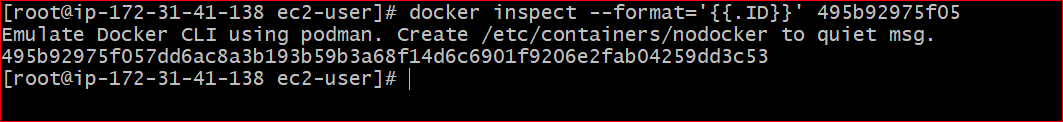


List all the running and stopped containers

Docker ps –a

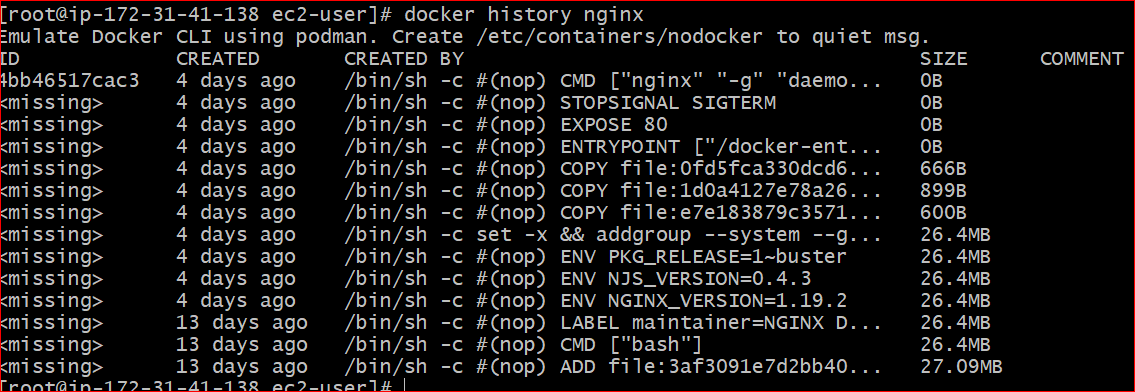
Find out the "Full" containerId of the container and write it below.

docker inspect --format='{{.ID}}' 495b92975f05



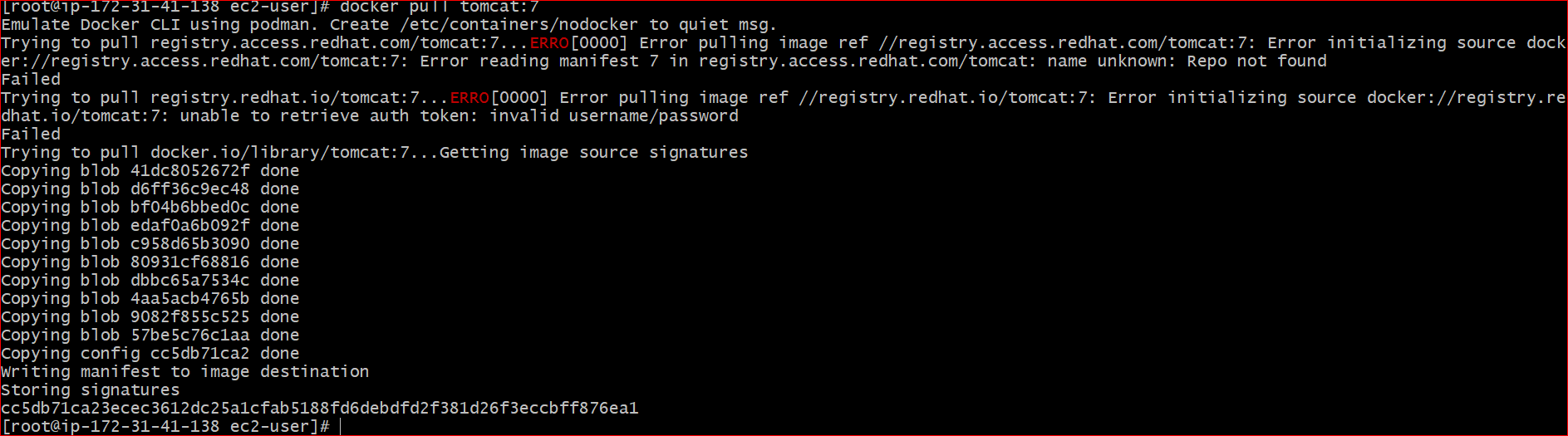
Find out how many image layers are used to build this image.

Docker history nginx



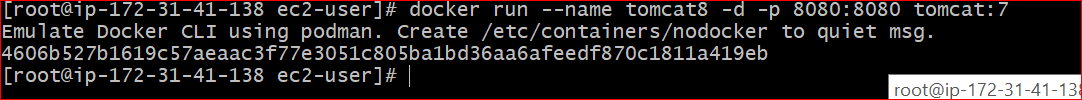
Get the Apache Tomcat 7 server image from the docker hub.

Docker pull tomcat:7



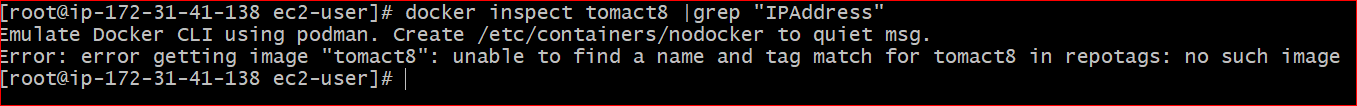
Run the Apache Tomcat 7, I mean create a container of Apache Tomcat.

docker run --name tomcat8 -d -p 8080:8080 tomcat:7



Find out what is the IP Address of the Apache Tomcat Container that it is running on

docker inspect tomact8 |grep "IPAddress"



Which Port it is using?

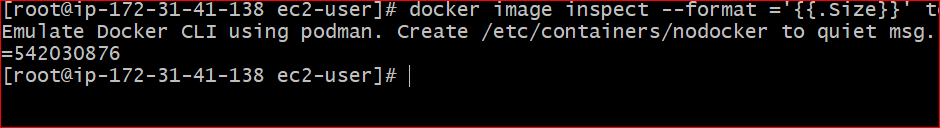
docker inspect 4606b527b161

14. Try to access the Tomcat's home page from your machine/vm.

https://ipaddress:8080

15. What is the disk size of Apache Tomcat image?

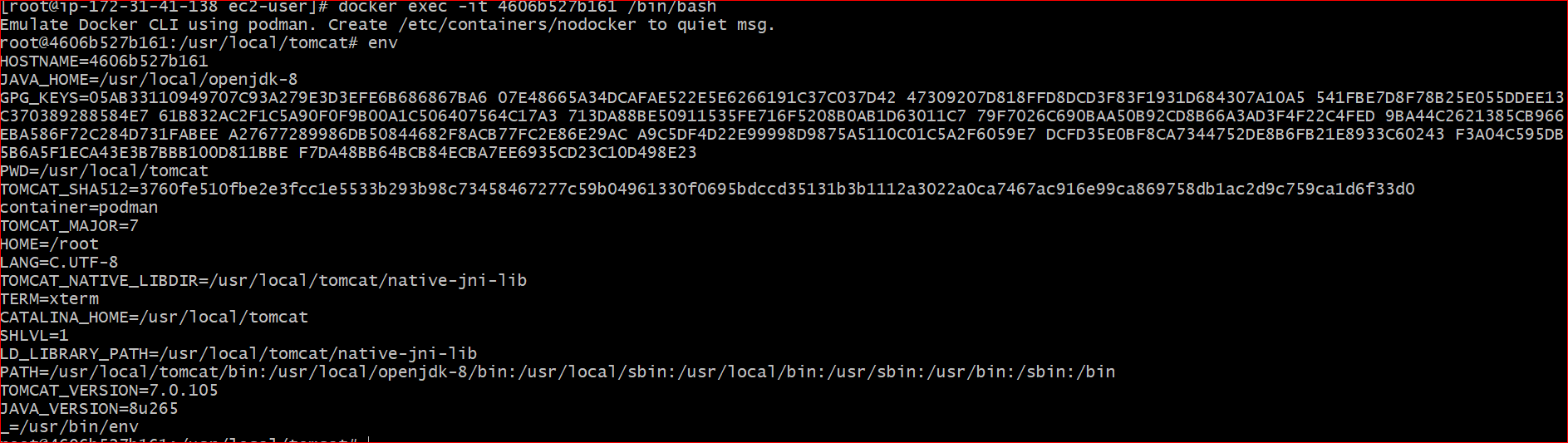
docker image inspect --format ='{{.Size}}' tomcat:7



16. Find out list of all environment variables that is configured for tomcat image, can you see JAVA\_HOME and CATALINA\_HOME? What did you notice about it?

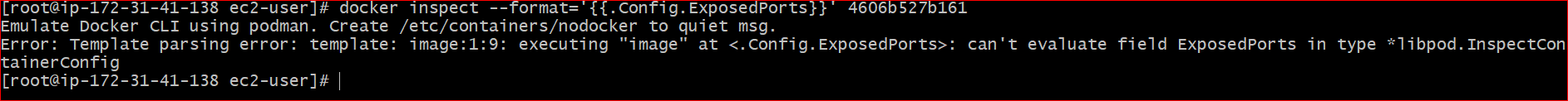
docker exec -it 4606b527b161 /bin/bash

env



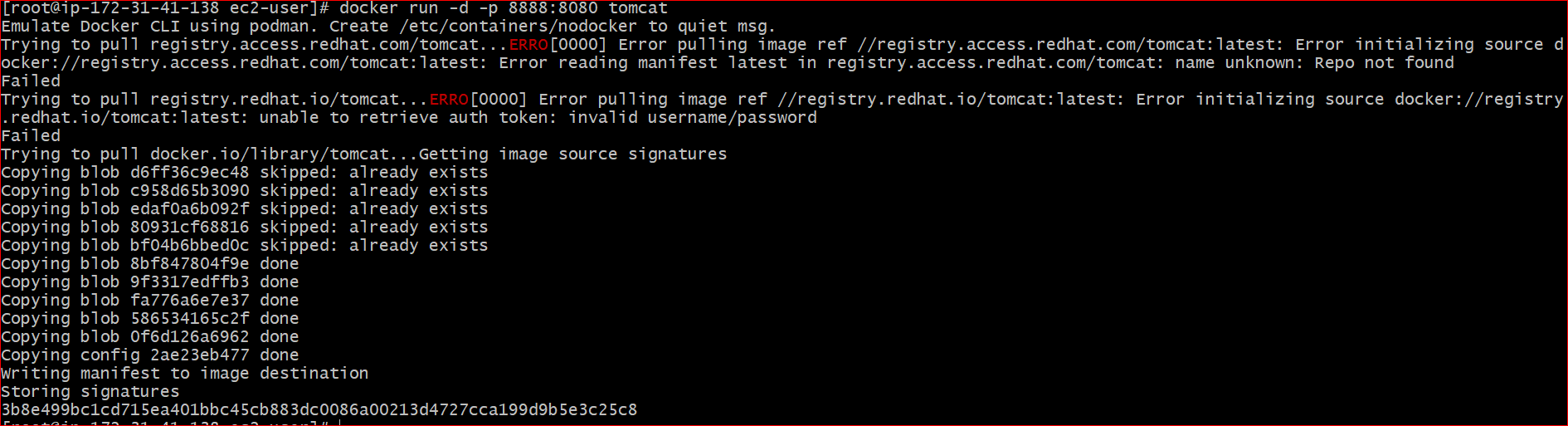
17. Find out which port is exposed for tomcat?

docker inspect --format='{{.Config.ExposedPorts}}' 4606b527b161



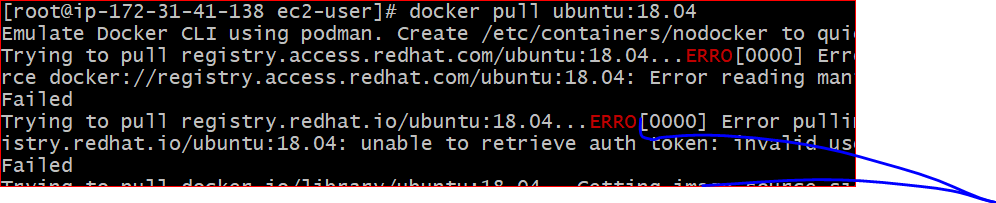
18. Run multiple conntainers of tomcat on different port and access it's home page.

docker run -d -p 8888:8080 tomcat



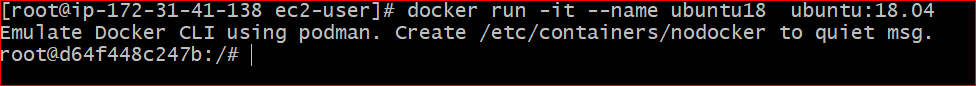
19. Pull ubuntu os from dockerhub, try to pull 2 images of ubuntu, Except the latest one.

docker pull ubuntu:18.04



20. Run the container of ubuntu in attached mode.

docker run -it --name ubuntu18 ubuntu:18.04

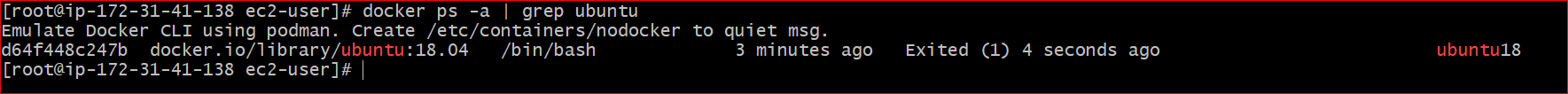


21. Run the container of another ubuntu in detached mode.

docker run -d --name ubuntu16 ubuntu:16.04

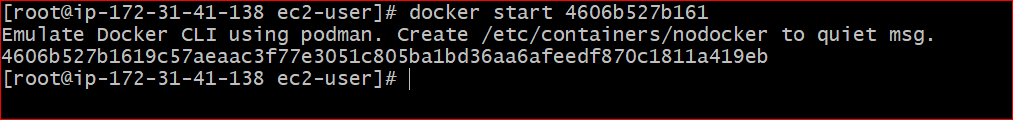
22. Check how many ubuntu containers are running and stopped

docker ps -a | grep ubuntu



23. Is the tomcat container running? If no, start one.

docker start 4606b527b161

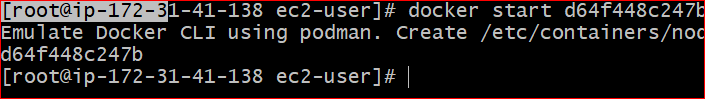


24. Check the logs, generated by tomcat container(don't forget to make request to tomcat's home page to see the log).

docker logs -f 4606b527b161

25. Check if ubuntu conatiner is running? If no, start one in attached mode to the terminal.

Docker start d64f448c247b



26. Login as root user in ubuntu container

docker exec --user root -it d64f448c247b /bin/bash

27. Create a file with any name in root directory

touch root.txt

28. Install software of your choice in ubuntu container using "apt-get install"

Apt-get update

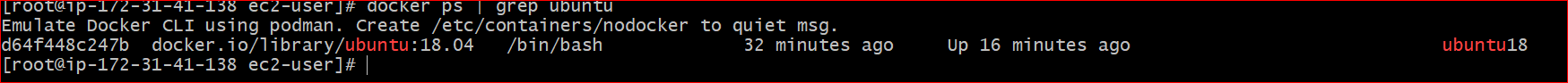
Apt-get install vim

29. Now exit the ubuntu shell, are you back to your host machine, if not, come back to the host machine.

exit

30. Check if the ubuntu container is running.

docker ps | grep Ubuntu

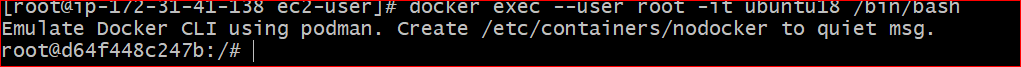


31. Create a new ubuntu container out of the same image as that previous container in attached mode.

docker run -it --name ubuntu18 -p 8081:8080 ubuntu:18.04

32. Login as a root user

docker exec --user root -it ubuntu18 /bin/bash



33. Check if you can see the file created in previous container, you will not see the file as well as software that you installed in the previous container. Now kill this Container.

docker rm <container id>

34. Do you have the previous ubuntu container where you created the file and installed the software? If no reapeat step 25 to 29.

35. Create an Image out of the existing container.

docker commit d64f448c247b ubunutimage

36. Now Create a Container out of this image and login into it to see if you can see the file and software installed by you in the previous container.

docker run -it --name newubuntu ubunutimage

37. Do you have running tomcat container? If yes, Stop it and kill all tomcat container.

docker stop <id>

docker kill id

38. Create an index.html file with following code in it:-

<h1>This is Tomcat Container</h1>

Now, Start a tomcat container in such a way that on hitting its URL for home page it should show the above html page.

docker run -d -p 8080:8080 -v /root/index.html:/usr/local/tomcat/webapps/ROOT/index.jsp --name tomcat tomcat:latest

39. type below command:-

docker images --help

Now, try to run command that proves the concept of following three options:-

1. -a

2. -f

3. -q

docker images -a =>gives all images

docker images -q =>gives image id's

docker images -f=>gives filter options

docker images --filter "dangling=true" =>gives all the images which are not used by the containers

docker images --filter "before=tomcat:8"=>gives all the images which we used before tomcat:8

write atleast 1 command using each option above and prove their concepts as described in the --help.

40. type below command:-

docker ps --help

Now, try to run command that proves the concept of following six options:-

1. -a

2. -f

3. -q

4. -n

5. -l

6. -s

1. docker ps -a=>gives all containers which are stopped or exicted started

2. docker ps -f status=exicted=>gives all the continaers were exicted

3. docker ps -q=>gives all containerid's

4. docker ps -n 2=>gives last 2 contianers

5. docker ps -l=>gives latest continaer were running

6. docker ps -s=>gives the size of the continers used