Assignment

NOTE: Do not forget to see manual page using "--help" option in command when searching for options/commands for a particular task.

1. Install Docker, either on your native OS or on a VM. Make sure it runs. Type "docker -v" to check if it's installed.

If you can't install or configure Docker, you can use the online docker setup to do the assignment.

Step1 Goto:- https://www.katacoda.com/courses/kubernetes/playground

Step2 Click on "continue" button on the left panel

Step3 Click on "launch.sh" button on the left panel

Step4 From the right panel use the top console to execute below command:-

docker -v

Try below commands for help

docker --help ---> This command shows all available options and commands to work with images and containers

docker image --help ---> This command shows all the avaialble options and commands to work with docker images

docker container --help ---> This command shows all the avaialble options and commands to work with docker containers

NOTE:- DO NOT TRY TO USE INTERNET TO SOLVE ASSIGNMENT,

BETTER USE THE ABOVE --help OPTION TO SEE THE MANUAL OF ANY PARTICULAR COMMAND AND FIGURE OUT THE SOLUTIONS ON YOUR OWN.

2. Find a image from dockerhub of your choice(recommeded: nginx), don't use browser, pull the official image from dockerhub --> **docker pull nginx:latest**

3. List all the available images in your machine/vm, make sure you see recently pulled image in the list. --> **docker images -a**

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

**docker images -q nginx**

5. Create a container of your image --> **docker conatiner create nginx:latest**

6. List all the running containers --> **docker container ls**

7. List all the running and stopped containers --> **docker container ls -a**

8. Find out the "Full" containerId of the container and write it below. **-> docker container ls -a -q** ContainerId: 3983898ed2db

9. Find out how many image layers are used to build this image. -->**docker history 4392e5dad77d** [4392e5dad77d - imageid]

10. Get the Apache Tomcat 7 server image from the docker hub. -> **docker pull tomcat:7.0**

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

**creating docker container:** **docker container create tomcat:7.0**,

**run the container: docker container start sweet\_davinci**

**[sweet\_davinci - container name]**

12. Find out what is the IP Address of the Apache Tomcat Container that it is running on ->

**docker inspect -f '{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}' 0026d9bdb6a5 [containerId] IPAddress: 172.17.0.2**

13. Which Port it is using? --> 8080/TCP

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

15. What is the disk size of Apache Tomcat image? --> 529MB

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 inspect 232f786fb1d**

"JAVA\_HOME=/usr/local/openjdk-8",

"JAVA\_VERSION=8u252",

"CATALINA\_HOME=/usr/local/tomcat",

17. Find out which port is exposed for tomcat? -->"Ports": {

"8080/tcp": [

{

"HostIp": "0.0.0.0",

"HostPort": "8888"

}

]

},

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

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

**image 1: docker pull ubuntu:xenial**

**image 2: docker pull ubuntu:xenial-20200326**

20. Run the container of ubuntu in attached mode. **--> docker run -i -t ubuntu:xenial /bin/bash**

21. Run the container of another ubuntu in detached mode. -->**docker run -d -i -t ubuntu:xenial-20200326 /bin/bash**

22. Check how many ubuntu containers are running and stopped --> **docker ps -a**

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

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

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

26. Login as root user in ubuntu container --> **docker run -i -t ubuntu:xenial /bin/bash**

27. Create a file with any name in root directory

1. apt install nano

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

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.

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

32. Login as a root user

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.

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.

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.

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

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.

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

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