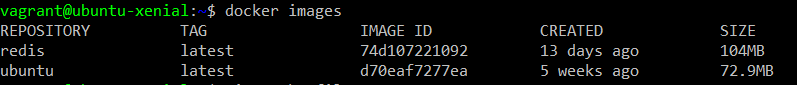
* **Creating a Docker image using Dockerfile and pushing it to Docker hub.**

We are going to create a Dockerfile in which we will specify the layers that will be used to build an image and then finally this image will be pushed to the Docker Hub.

The steps to be followed are:

**1.** Let us check which images are currently available in our local machine.

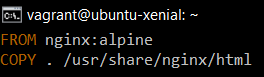
Command: docker images



**2.** Create a Docker file in the directory. Name this file “Dockerfile”.

Command: vim Dockerfile

Insert the following lines in the file.



The first line specifies that nginx:alpine image will be used as the base image for our image.

The second line specifies that the contents of current directory will be copied in /usr/share/nginx/html directory.

**3.** Create a file named index.html with a simple hello world code as follows.

Command: vim index.html

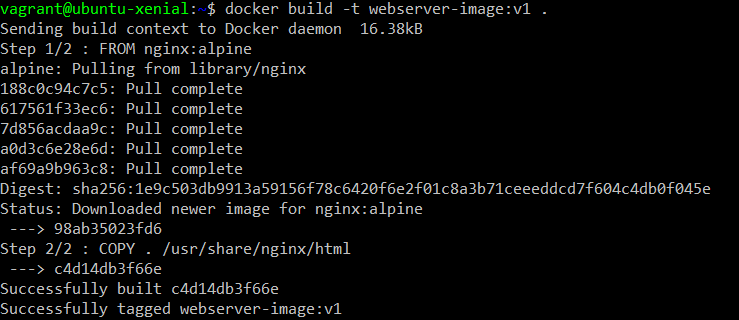


**4.** Now we are ready to build our Dockerfile to get the desired image in our local machine.

Command: docker build –t <image-name> .

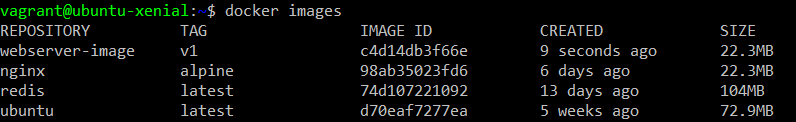
Here ‘.’ Specifies that the image will be built in current directory.

The image will be tagged with the specified name.



**5.** Let us verify that specified image is available now.

Command: docker images



The above output clearly shows the existence of newly built image.

**6.** Let us run a container with this image to verify that the image is correctly formed.

Command: docker run –d –p 80:80 <image-name>



In this step, the image is used to start a container and port 80 of the container is bind to port 80 of the local machine.

**7.** Let us verify that intended working of the image is being fulfilled. For this, we can use the following command.

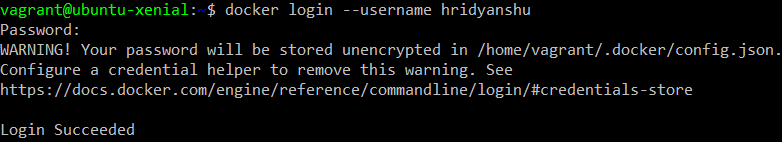
Command: curl 0.0.0.0:80



The above output specifies that the site is reachable.

**8.** Now let us push this image to the Docker Hub. To do so first let us login using our Docker Hub credentials.

Command: docker login –username <docker-hub-username>

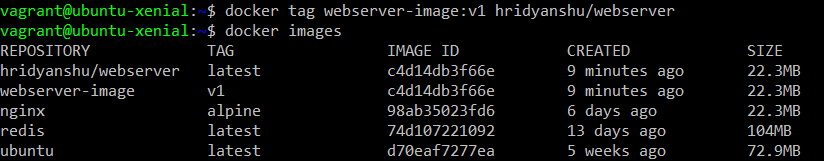


**9.** After successful login, let us tag our image with our username.

Command: docker tag <image-name> <image-tag>

We can verify the newly added tag.

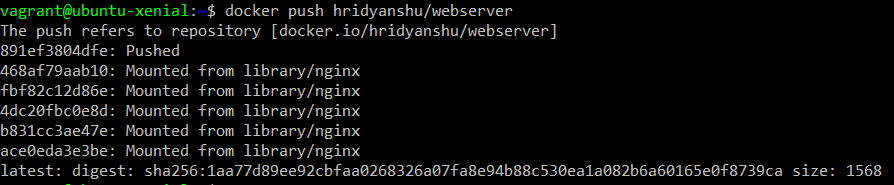
Command: docker images



The above output specifies that the webserver-image is tagged with a new tag hridyanshu/webserver.

**10.** Push the image to you Docker Hub account.

Command: docker push <image>



**11.** You can verify that the image is pushed to your Docker Hub account.

