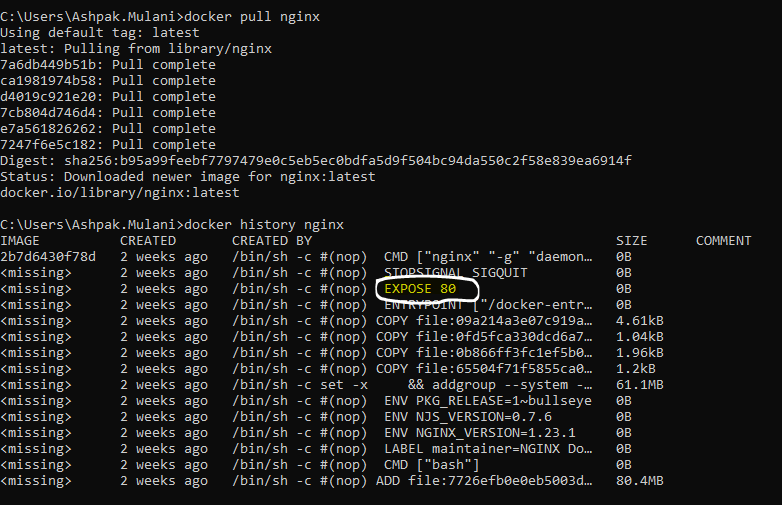
Image Basics

Images are managed in layered file system. Each change done to base image is stored in separate layer as delta only.

To get image locally on machine use docker pull <<Image name>> Ex. docker pull nginx

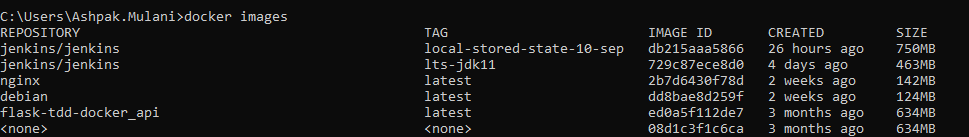
If we run docker history imagename then we can see all layers in the images shown below. Output shows line by line command used to create that extra layer. Ex. Exposing port 80 command created new layer and that layer size is almost 0B since it has only tiny setting info in it.



# Image Tags

If we list all docker images from machine then we also see tags in returned results.

Usually, tag is set manually by developers and it is set to version of image or ‘latest’



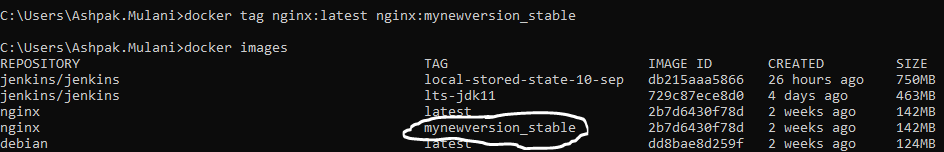
If we set a new tag then new image is created because one image can have one tag.

**Syntax :** docker tag source\_image[:TAG] target\_image[:TAG]

Ex.

Docker tag nginx:latest nginx:myLatest\_stable

Then we get one more entry of image with new tag. Tags can be used to search images locally and in repo.



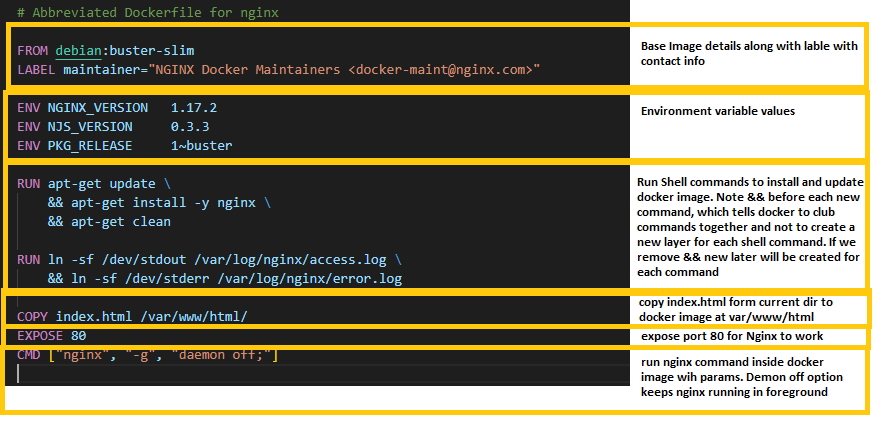
# Build Docker Image

DockerFile is used to create docker image using one base image.

Sample docker file



Docker file contains different sections which updates base image by running commands and changing settings



To build docker image we need to run docker build command

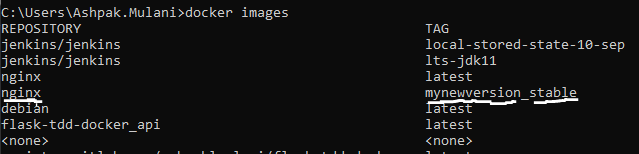
Docker build -t myenginx .

**-t** option is to specify tag for image. In this case its ‘myenginx’

**.** at the end signifies that use DockerFile from current directory

# Delete Image

List the images first Docker images



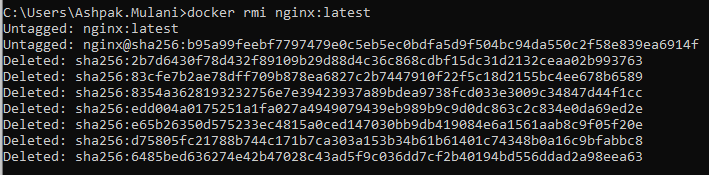
Now lets delete nginx image highlighted using docker rmi <<image:tag>>

Ex. docker rmi nginx:mynewversion\_stable

This will only remove tag because docker is smart and to reduce disk space it doesn’t clone original image so save space. If we go and delete original image then it will on backed clone original image before deleting and make separate copy for tagged image.



Deleting normal image should look like this/ It removes all tags and the delete all layers from disk.



sdc