

Lesson 02 Demo 08

Inspecting, Removing, and Pruning Images

Objective: To demonstrate the process of creating, inspecting, removing, and pruning Docker images

Tools required: Docker and Ubuntu

Prerequisites: None

Steps to be followed:

1. Inspect, remove, and prune docker images

Step 1: Inspect, remove, and prune docker images

1.1 Use the following command to pull the nginx image from Docker Hub:

sudo docker pull nginx:latest

```
bharathgsimplil@ip-172-31-42-255:~$ sudo docker pull nginx:latest
latest: Pulling from library/nginx
8a1e25ce7c4f: Pull complete
e78b137be355: Pull complete
39fc875bd2b2: Pull complete
035788421403: Pull complete
87c3fb37cbf2: Pull complete
c5cdd1ce752d: Pull complete
33952c599532: Pull complete
Digest: sha256:6db391d1c0cfb30588ba0bf72ea999404f2764febf0f1f196acd5867ac7efa7e
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
bharathgsimplil@ip-172-31-42-255:~$ █
```

1.2 Use the following command to inspect a Docker image:

sudo docker image inspect nginx:latest

```
bharathgsimplil@ip-172-31-42-255:~$ sudo docker image inspect nginx:latest
[
  {
    "Id": "sha256:92b11f67642b62bbb98e7e49169c346b30e20cd3c1c034d31087e46924b9312e",
    "RepoTags": [
      "nginx:latest"
    ],
    "RepoDigests": [
      "nginx@sha256:6db391d1c0cfb30588ba0bf72ea999404f2764feb0f1f196acd5867ac7efa7e"
    ],
    "Parent": "",
    "Comment": "buildkit.dockerfile.v0",
    "Created": "2024-02-14T18:24:57Z",
    "Container": "",
    "ContainerConfig": {
      "Hostname": "",
      "Domainname": "",
      "User": "",
      "AttachStdin": false,
      "AttachStdout": false,
      "AttachStderr": false,
      "Tty": false,
      "OpenStdin": false,
      "StdinOnce": false,
      "Env": null,
      "Cmd": null,

```

1.3 Use the following command to list all the running images:

sudo docker images

```
bharathgsimplil@ip-172-31-42-255:~$ sudo docker images
REPOSITORY      TAG         IMAGE ID      CREATED        SIZE
nginx           latest     92b11f67642b  4 weeks ago   187MB
registry        2          a8781fe3b7a2  6 weeks ago   25.4MB
bharathgsimplil@ip-172-31-42-255:~$
```

1.4 Use the following command to remove an image:

sudo docker image rm IMAGE_ID

```
bharathgsimplil@ip-172-31-42-255:~$ sudo docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
nginx          latest    92b11f67642b   4 weeks ago    187MB
registry       2         a8781fe3b7a2   6 weeks ago    25.4MB
bharathgsimplil@ip-172-31-42-255:~$ sudo docker image rm 92b11f67642b
Untagged: nginx:latest
Untagged: nginx@sha256:6db391d1c0cfb30588ba0bf72ea999404f2764feb0f1f196acd5867ac7efa7e
Deleted: sha256:92b11f67642b62bbb98e7e49169c346b30e20cd3c1c034d31087e46924b9312e
Deleted: sha256:d9e826dbb4b3c5770fe92638baa8c6614f210d782a5d021a123fe9fa1f92c23d
Deleted: sha256:2a75285e888884bed4d630896c86ecd71739c6e82669e21ad7a050f33c9ac48d
Deleted: sha256:32bfe3f040358ab8f9872a63d4ddefdc68f35d991ca10a812cbac5912ae9f97b
Deleted: sha256:1330486eb62ea7e96f384961b77b0fc85f5d4422e761114ef3a72e7cb89751a4
Deleted: sha256:a375372209a0f2b2c697a52cce46bc41b495bf86184ae83dd5146e20c22078eb
Deleted: sha256:450787ca55caa59d0288de9cf36fc6b77d1b208a77eb837ec3d25b385f99cafb
Deleted: sha256:a483da8ab3e941547542718cacd3258c6c705a63e94183c837c9bc44eb608999
bharathgsimplil@ip-172-31-42-255:~$ █
```

Note: Replace IMAGE_ID with Id of the image you wish to delete

1.5 Use the following command to remove all untagged images:

sudo docker image prune --all

```
bharathgsimplil@ip-172-31-42-255:~$ sudo docker image prune --all
WARNING! This will remove all images without at least one container associated to them.
Are you sure you want to continue? [y/N] y
Total reclaimed space: 0B
bharathgsimplil@ip-172-31-42-255:~$ █
```

By following these steps, you have successfully demonstrated how to create, inspect, remove, and prune Docker images for efficient Docker image management.