Lab 1

Lab 1: Understanding Basic Docker Operations

Objective

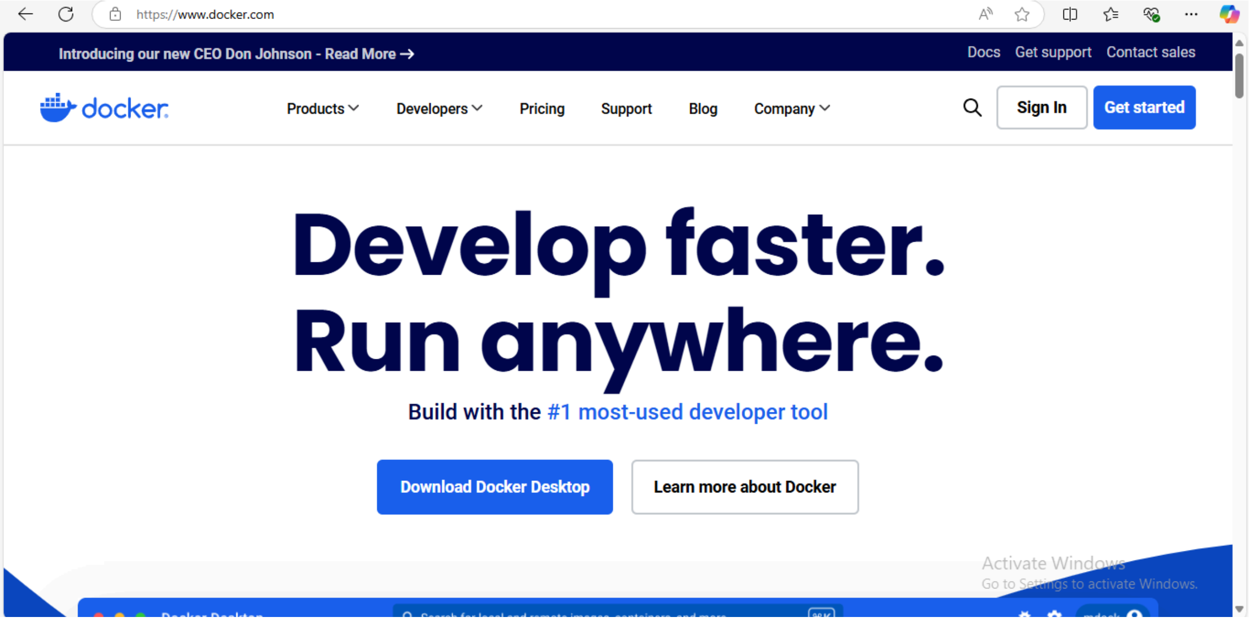
This lab is designed to provide hands-on experience with Docker, focusing on basic operations such as pulling, tagging, and pushing images, as well as running containers, managing volumes, and configuring networks. This will help learners understand the fundamental Docker commands and concepts without involving Docker file creation or docker compose.

Exercise 1: Sign in with your Docker account

To pull & push images to Docker Hub, you will need to sign in with a Docker account.

1. Open the Docker Dashboard. (Hoping you have already installed Docker in your machine)
2. Select **Sign in** at the top-right corner.
3. If needed, create an account and then complete the sign-in flow.

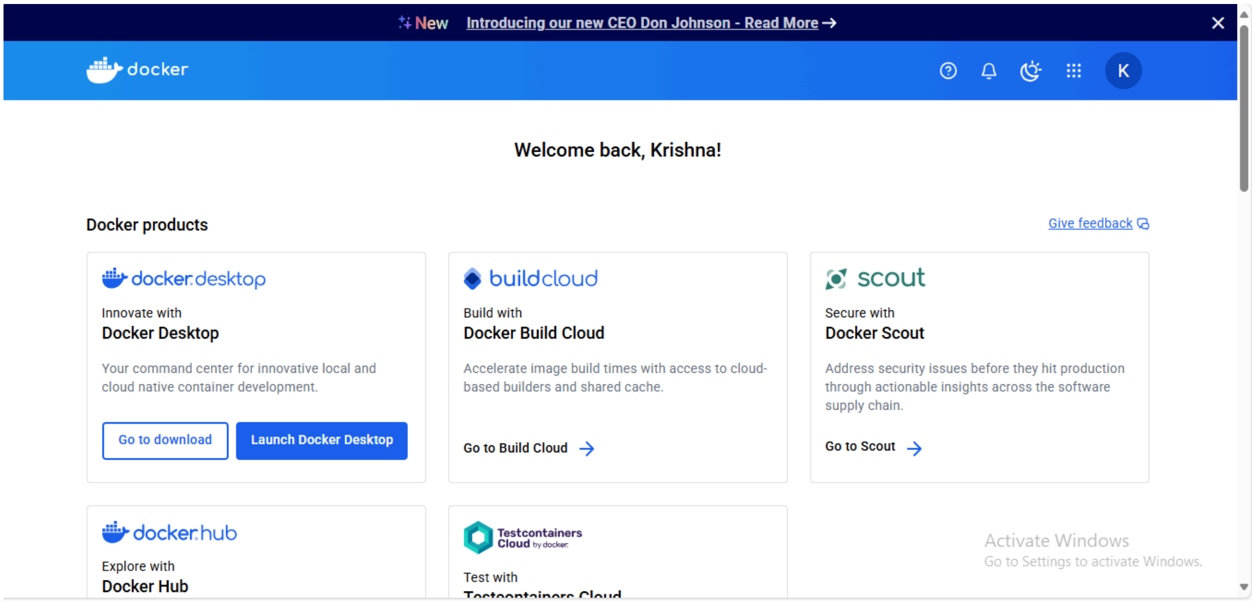
Once you're done, you should see the **Sign in** button turn into a profile picture.



Introducing our new CEO Don Johnson - Read More →

Develop faster.
Run anywhere.

Build with the #1 most-used developer tool



New Introducing our new CEO Don Johnson - Read More →

Welcome back, Krishna!

Docker products

Give feedback

docker desktop
Innovate with
Docker Desktop
Your command center for innovative local and cloud native container development.
Go to download | Launch Docker Desktop

build cloud
Build with
Docker Build Cloud
Accelerate image build times with access to cloud-based builders and shared cache.
Go to Build Cloud →

scout
Secure with
Docker Scout
Address security issues before they hit production through actionable insights across the software supply chain.
Go to Scout →

docker hub
Explore with
Docker Hub

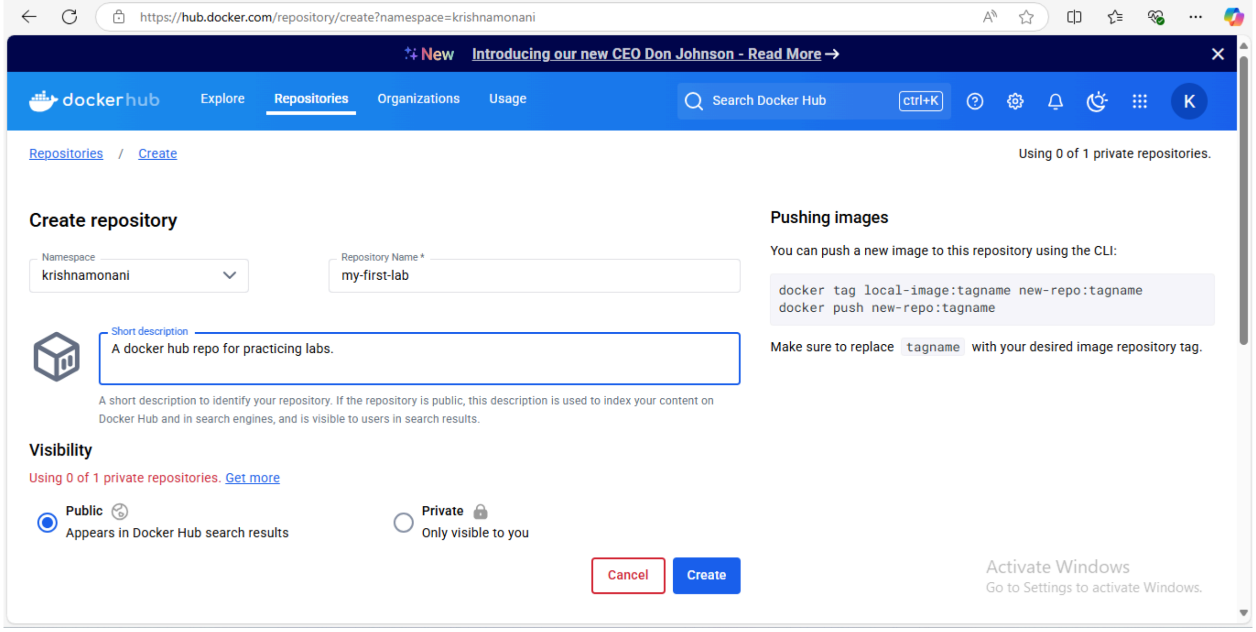
Testcontainers Cloud by docker
Test with
Testcontainers Cloud

Activate Windows
Go to Settings to activate Windows.

Exercise 2: Create an image repository

Now that you have an account, you can create an image repository. Just as a Git repository holds source code, an image repository stores container images.

1. Go to Docker Hub (Links to an external site.)
2. Select **Create repository**.
3. On the **Create repository** page, enter the following information:
    - **Repository name** - *my-first-lab*
    - **Short description** - feel free to enter a description if you'd like
    - **Visibility** - select **Public** to allow others to pull images from this repository
4. Select **Create** to create the public repository.



https://hub.docker.com/repository/create?namespace=krishnamonani

New Introducing our new CEO Don Johnson - Read More →

docker hub | Explore | Repositories | Organizations | Usage | Search Docker Hub ctrl+K

Repositories / Create

Using 0 of 1 private repositories.

Create repository

Namespace: krishnamonani

Repository Name *: my-first-lab

Short description: A docker hub repo for practicing labs.

A short description to identify your repository. If the repository is public, this description is used to index your content on Docker Hub and in search engines, and is visible to users in search results.

Visibility

Using 0 of 1 private repositories. Get more

● Public
Appears in Docker Hub search results

○ Private
Only visible to you

Cancel | Create

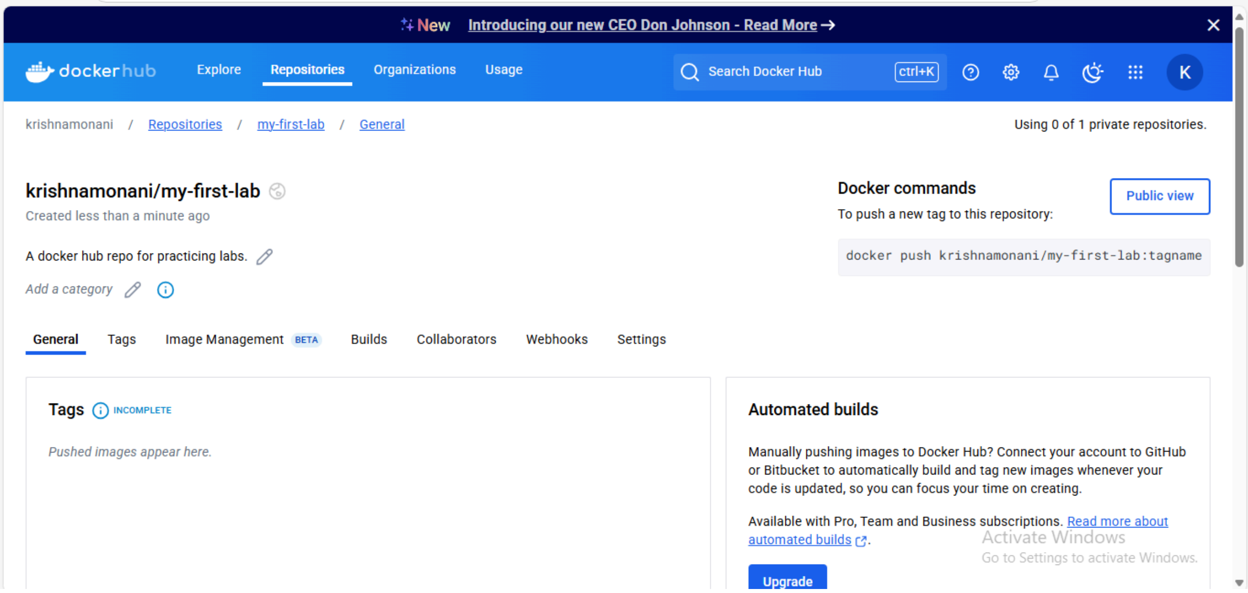
Pushing images

You can push a new image to this repository using the CLI:

```
docker tag local-image:tagname new-repo:tagname
docker push new-repo:tagname
```

Make sure to replace `tagname` with your desired image repository tag.

Activate Windows
Go to Settings to activate Windows.



krishnamonani / Repositories / my-first-lab / General

Using 0 of 1 private repositories.

krishnamonani/my-first-lab

Created less than a minute ago

A docker hub repo for practicing labs.

Add a category

Docker commands

To push a new tag to this repository:

```
docker push krishnamonani/my-first-lab:tagname
```

Tags INCOMPLETE

Pushed images appear here.

Automated builds

Manually pushing images to Docker Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.

Available with Pro, Team and Business subscriptions. Read more about automated builds.

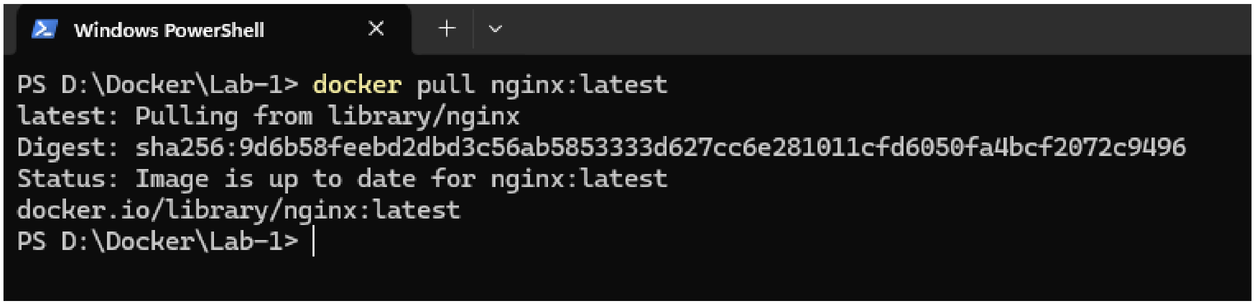
krishnamonani/my-first-lab general

Exercise 3: Pulling an Image from Docker Hub

To begin working with Docker images, you'll need to pull an existing image from Docker Hub.

- Open your terminal or command prompt.
- Pull the official `nginx` image (a lightweight, high-performance web server) by running:

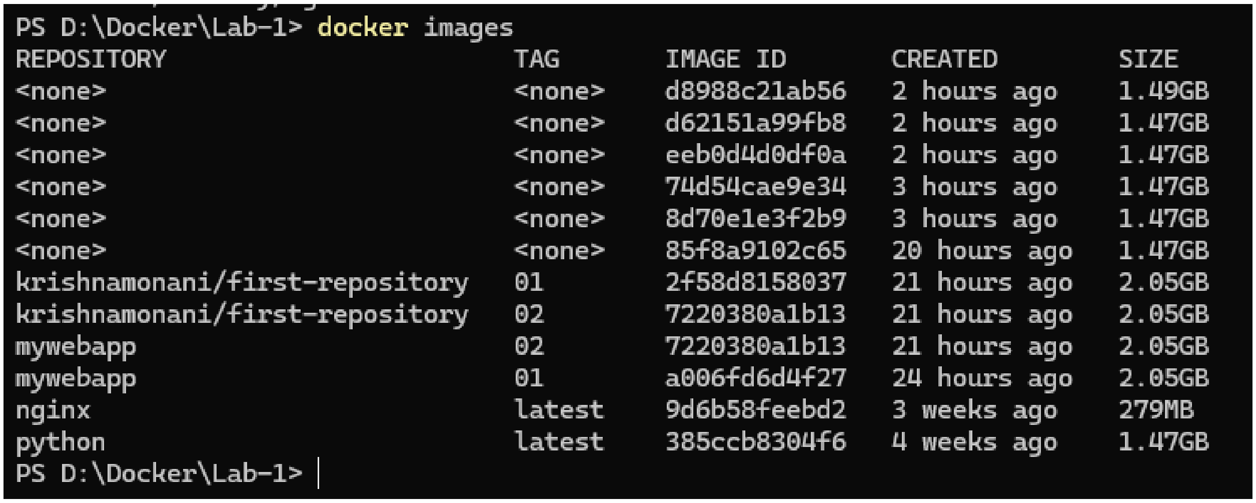
docker pull nginx:latest



```
PS D:\Docker\Lab-1> docker pull nginx:latest
latest: Pulling from library/nginx
Digest: sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496
Status: Image is up to date for nginx:latest
docker.io/library/nginx:latest
PS D:\Docker\Lab-1> 
```

- Verify that the image has been downloaded by listing the images:

docker images

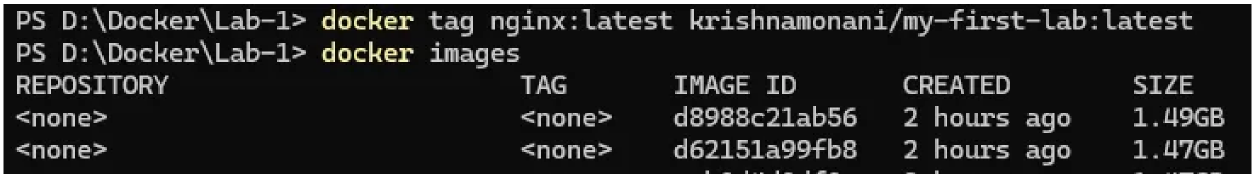


```
PS D:\Docker\Lab-1> docker images
REPOSITORY                        TAG       IMAGE ID       CREATED        SIZE
<none>                            <none>    d8988c21ab56   2 hours ago    1.49GB
<none>                            <none>    d62151a99fb8   2 hours ago    1.47GB
<none>                            <none>    eeb0d4d0df0a   2 hours ago    1.47GB
<none>                            <none>    74d54cae9e34   3 hours ago    1.47GB
<none>                            <none>    8d70e1e3f2b9   3 hours ago    1.47GB
<none>                            <none>    85f8a9102c65   20 hours ago   1.47GB
krishnamonani/first-repository    01        2f58d8158037   21 hours ago   2.05GB
krishnamonani/first-repository    02        7220380a1b13   21 hours ago   2.05GB
mywebapp                          02        7220380a1b13   21 hours ago   2.05GB
mywebapp                          01        a006fd6d4f27   24 hours ago   2.05GB
nginx                             latest    9d6b58feebd2   3 weeks ago    279MB
python                            latest    385ccb8304f6   4 weeks ago    1.47GB
PS D:\Docker\Lab-1>
```

Exercise 4: Tagging the Docker Image

- Tag the pulled image for your repository

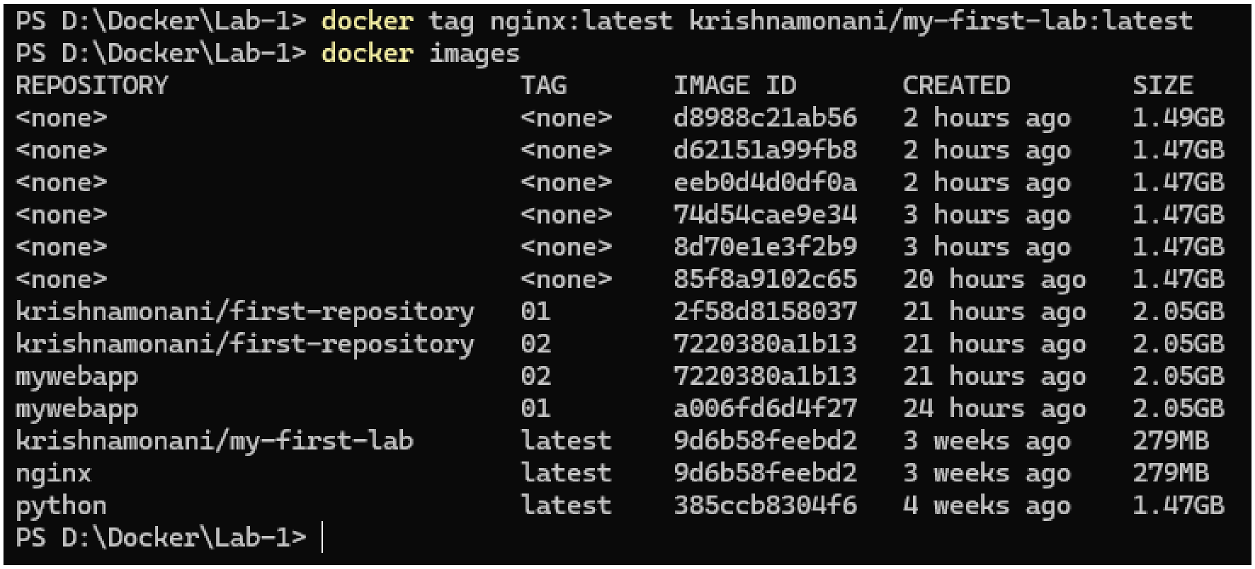
docker tag nginx:latest <your-username>/my-first-lab:latest



```
PS D:\Docker\Lab-1> docker tag nginx:latest krishnamonani/my-first-lab:latest
PS D:\Docker\Lab-1> docker images
REPOSITORY                        TAG       IMAGE ID       CREATED        SIZE
<none>                            <none>    d8988c21ab56   2 hours ago    1.49GB
<none>                            <none>    d62151a99fb8   2 hours ago    1.47GB
```

- Verify the new tag

docker images

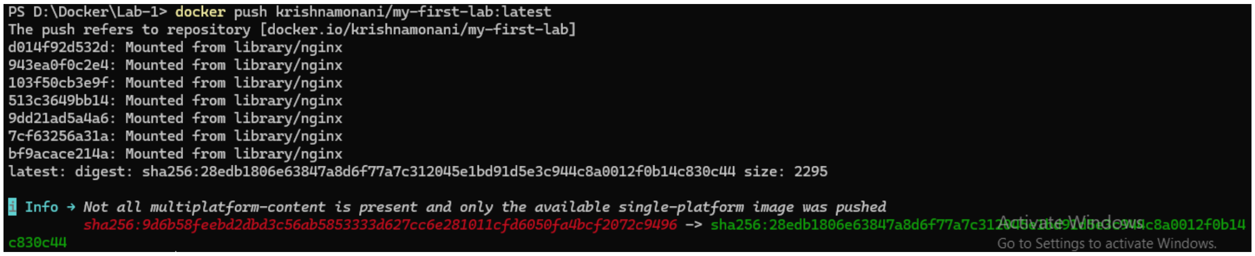


```
PS D:\Docker\Lab-1> docker tag nginx:latest krishnamonani/my-first-lab:latest
PS D:\Docker\Lab-1> docker images
REPOSITORY                        TAG       IMAGE ID       CREATED        SIZE
<none>                            <none>    d8988c21ab56   2 hours ago    1.49GB
<none>                            <none>    d62151a99fb8   2 hours ago    1.47GB
<none>                            <none>    eeb0d4d0df0a   2 hours ago    1.47GB
<none>                            <none>    74d54cae9e34   3 hours ago    1.47GB
<none>                            <none>    8d70e1e3f2b9   3 hours ago    1.47GB
<none>                            <none>    85f8a9102c65   20 hours ago   1.47GB
krishnamonani/first-repository    01        2f58d8158037   21 hours ago   2.05GB
krishnamonani/first-repository    02        7220380a1b13   21 hours ago   2.05GB
mywebapp                          02        7220380a1b13   21 hours ago   2.05GB
mywebapp                          01        a006fd6d4f27   24 hours ago   2.05GB
krishnamonani/my-first-lab        latest    9d6b58feebd2   3 weeks ago    279MB
nginx                             latest    9d6b58feebd2   3 weeks ago    279MB
python                            latest    385ccb8304f6   4 weeks ago    1.47GB
PS D:\Docker\Lab-1>
```

Exercise 5: Pushing the Image

- *Push the tagged image*

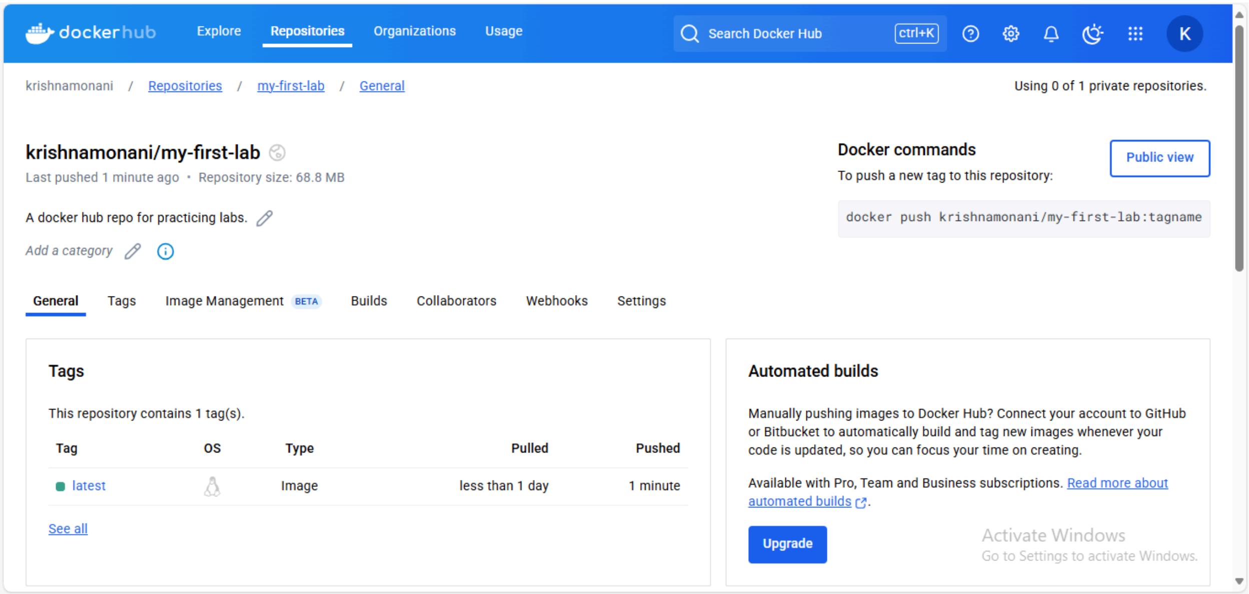
docker push <your-username>/my-first-lab:latest



```
PS D:\Docker\Lab-1> docker push krishnamonani/my-first-lab:latest
The push refers to repository [docker.io/krishnamonani/my-first-lab]
d014f92d532d: Mounted from library/nginx
943ea0f0c2e4: Mounted from library/nginx
103f50cb3e9f: Mounted from library/nginx
513c3649bb14: Mounted from library/nginx
9dd21ad5a4a6: Mounted from library/nginx
7cf63256a31a: Mounted from library/nginx
bf9acace214a: Mounted from library/nginx
latest: digest: sha256:28edb1806e63847a8d6f77a7c312045e1bd91d5e3c944c8a0012f0b14c830c44 size: 2295

i Info → Not all multiplatform-content is present and only the available single-platform image was pushed
         sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496 -> sha256:28edb1806e63847a8d6f77a7c312045e1bd91d5e3c944c8a0012f0b14
c830c44
```

- *Confirm the image is in your Docker Hub repository by visiting the website.*



krishnamonani / Repositories / my-first-lab / General

Using 0 of 1 private repositories.

krishnamonani/my-first-lab

Last pushed 1 minute ago • Repository size: 68.8 MB

A docker hub repo for practicing labs.

Add a category

Docker commands

To push a new tag to this repository:

```
docker push krishnamonani/my-first-lab:tagname
```

Tags

This repository contains 1 tag(s).

Tag	OS	Type	Pulled	Pushed
● latest		Image	less than 1 day	1 minute

See all

Automated builds

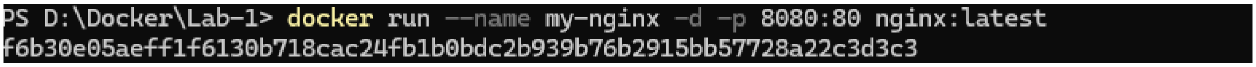
Manually pushing images to Docker Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.

Available with Pro, Team and Business subscriptions. Read more about automated builds.

Exercise 6: Running Containers

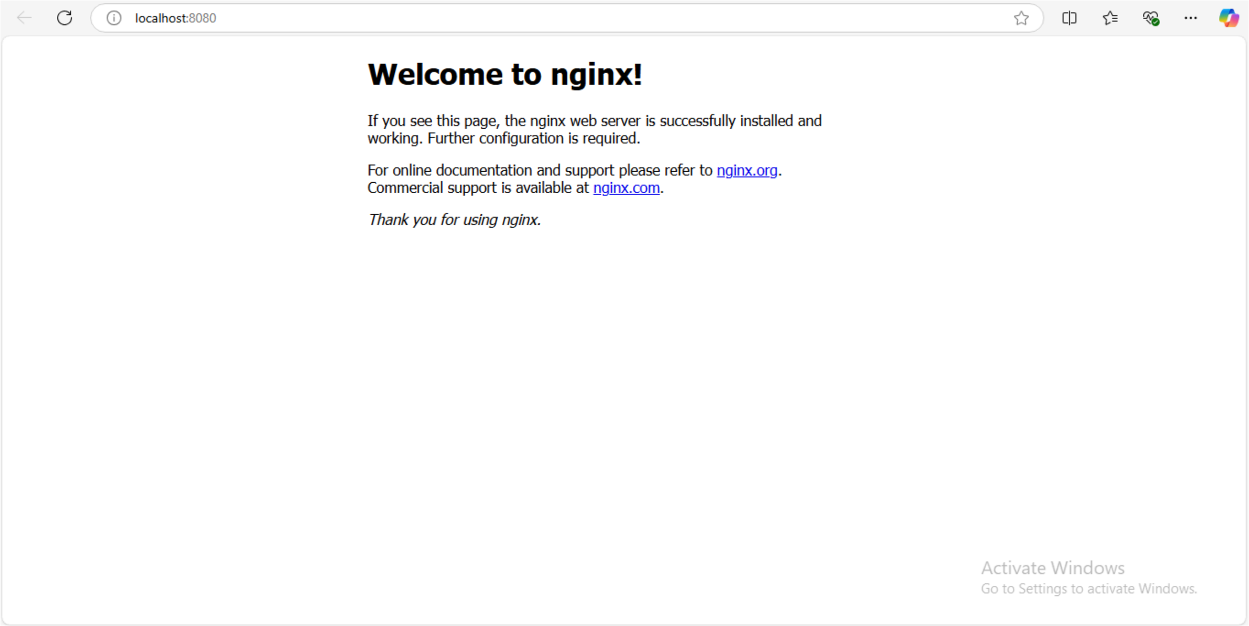
Start a container from the image you have pulled to understand the basics of container deployment.

docker run --name my-nginx -d -p 8080:80 nginx:latest



```
PS D:\Docker\Lab-1> docker run --name my-nginx -d -p 8080:80 nginx:latest
f6b30e05aeff1f6130b718cac24fb1b0bdc2b939b76b2915bb57728a22c3d3c3
```

Open http://localhost:8080 (Links to an external site.) on your browser to check if it worked.



Welcome to nginx!

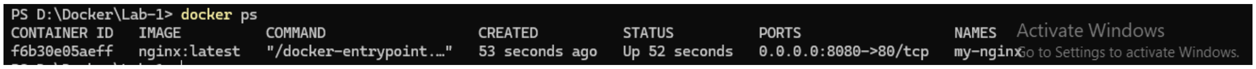
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

List all the running containers

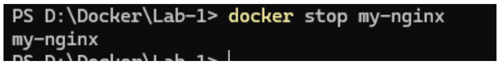
docker ps



```
PS D:\Docker\Lab-1> docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS          PORTS                  NAMES
f6b30e05aeff   nginx:latest   "/docker-entrypoint.…"   53 seconds ago   Up 52 seconds   0.0.0.0:8080->80/tcp   my-nginx
```

Stop the container

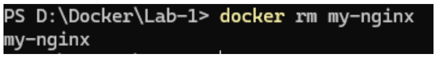
docker stop my-nginx



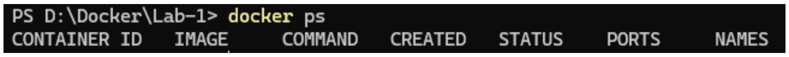
```
PS D:\Docker\Lab-1> docker stop my-nginx
my-nginx
PS D:\Docker\Lab-1>
```

Remove the container

docker rm my-nginx



```
PS D:\Docker\Lab-1> docker rm my-nginx
my-nginx
```



```
PS D:\Docker\Lab-1> docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
```

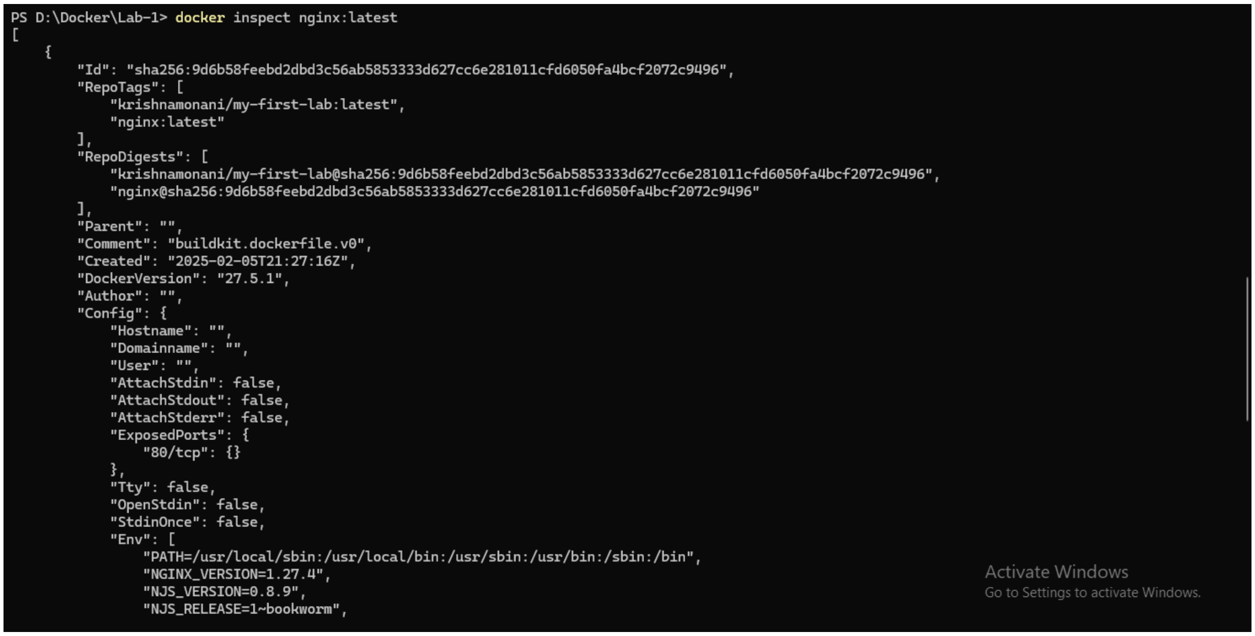
Explanation of flags:

- run: Runs a command in a new container.
- -name my-nginx: Assigns a name to the container for easier reference.
- d: Runs the container in detached mode.
- p 8080:80: Maps port 80 of the container to port 8080 on the host.
- ps: Lists running containers.
- stop: Stops a running container.
- rm: Removes one or more containers

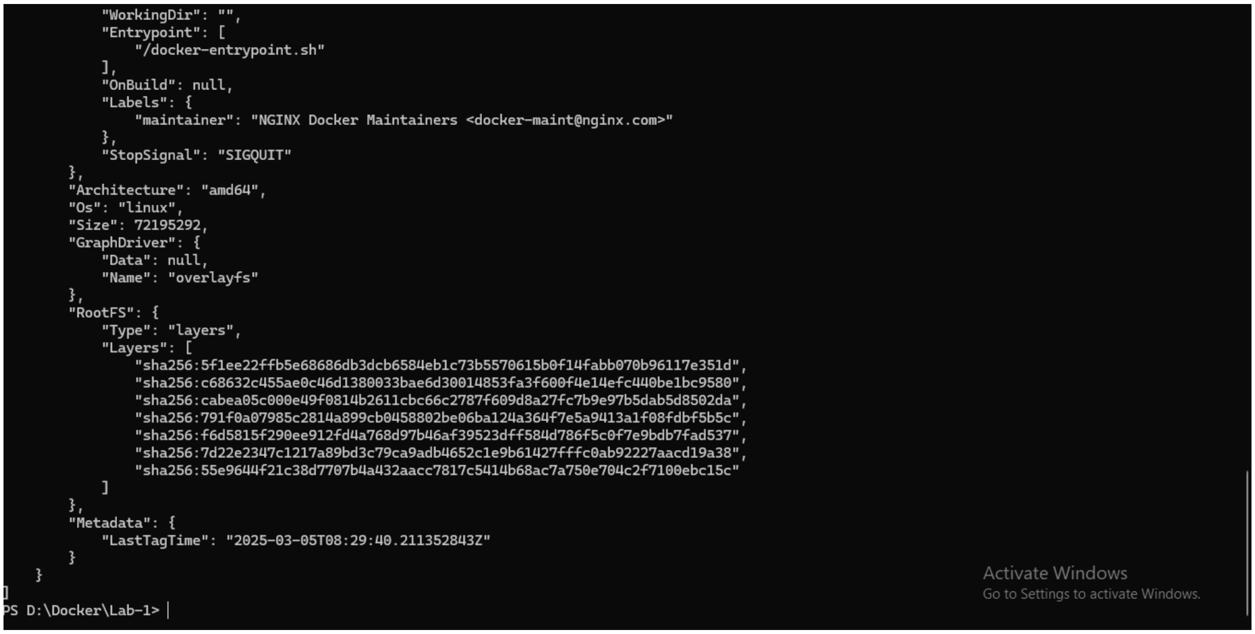
Exercise 7: Inspecting Images

Inspect the details of a Docker image.

docker inspect nginx:latest



```
PS D:\Docker\Lab-1> docker inspect nginx:latest
[
    {
        "Id": "sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496",
        "RepoTags": [
            "krishnamonani/my-first-lab:latest",
            "nginx:latest"
        ],
        "RepoDigests": [
            "krishnamonani/my-first-lab@sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496",
            "nginx@sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496"
        ],
        "Parent": "",
        "Comment": "buildkit.dockerfile.v0",
        "Created": "2025-02-05T21:27:16Z",
        "DockerVersion": "27.5.1",
        "Author": "",
        "Config": {
            "Hostname": "",
            "Domainname": "",
            "User": "",
            "AttachStdin": false,
            "AttachStdout": false,
            "AttachStderr": false,
            "ExposedPorts": {
                "80/tcp": {}
            },
            "Tty": false,
            "OpenStdin": false,
            "StdinOnce": false,
            "Env": [
                "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",
                "NGINX_VERSION=1.27.4",
                "NJS_VERSION=0.8.9",
                "NJS_RELEASE=1~bookworm",
```



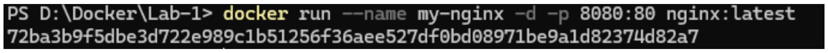
```
            "WorkingDir": "",
            "Entrypoint": [
                "/docker-entrypoint.sh"
            ],
            "OnBuild": null,
            "Labels": {
                "maintainer": "NGINX Docker Maintainers <docker-maint@nginx.com>"
            },
            "StopSignal": "SIGQUIT"
        },
        "Architecture": "amd64",
        "Os": "linux",
        "Size": 72195292,
        "GraphDriver": {
            "Data": null,
            "Name": "overlayfs"
        },
        "RootFS": {
            "Type": "layers",
            "Layers": [
                "sha256:5f1ee22ffb5e68686db3dcb6584eb1c73b5570615b0f14fabb070b96117e351d",
                "sha256:c68632c455ae0c46d1380033bae6d30014853fa3f600f4e14efc440be1bc9580",
                "sha256:cabea05c000e49f0814b2611cbc66c2787f609d8a27fc7b9e97b5dab5d8502da",
                "sha256:791f0a07985c2814a899cb0458802be06ba124a364f7e5a9413a1f08fdbf5b5c",
                "sha256:f6d5815f290ee912fd4a768d97b46af39523dff584d786f5c0f7e9bdb7fad537",
                "sha256:7d22e2347c1217a89bd3c79ca9adb4652c1e9b61427fffc0ab92227aacd19a38",
                "sha256:55e9644f21c38d7707b4a432aacc7817c5414b68ac7a750e704c2f7100ebc15c"
            ]
        },
        "Metadata": {
            "LastTagTime": "2025-03-05T08:29:40.211352843Z"
        }
    }
]
PS D:\Docker\Lab-1> |
```

```
[
{
"Id": "sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496",
"RepoTags": [
"krishnamonani/my-first-lab:latest",
"nginx:latest"
],
"RepoDigests": [
"krishnamonani/my-first-lab@sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496",
"nginx@sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496"
],
"Parent": "",
"Comment": "buildkit.dockerfile.v0",
"Created": "2025-02-05T21:27:16Z",
"DockerVersion": "27.5.1",
"Author": "",
"Config": {
"Hostname": "",
"Domainname": "",
"User": "",
"AttachStdin": false,
"AttachStdout": false,
"AttachStderr": false,
"ExposedPorts": {
"80/tcp": {}
},
"Tty": false,
"OpenStdin": false,
"StdinOnce": false,
"Env": [
"PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",
"NGINX_VERSION=1.27.4",
"NJS_VERSION=0.8.9",
"NJS_RELEASE=1~bookworm",
"PKG_RELEASE=1~bookworm",
"DYNPKG_RELEASE=1~bookworm"
],
"Cmd": [
"nginx",
"-g",
"daemon off;"
],
"Image": "",
"Volumes": null,
"WorkingDir": "",
"Entrypoint": [
"/docker-entrypoint.sh"
],
"OnBuild": null,
"Labels": {
"maintainer": "NGINX Docker Maintainers docker-maint@nginx.com"
},
"StopSignal": "SIGQUIT"
},
"Architecture": "amd64",
"Os": "linux",
"Size": 72195292,
"GraphDriver": {
"Data": null,
"Name": "overlayfs"
},
"RootFS": {
"Type": "layers",
"Layers": [
"sha256:5f1ee22ffb5e68686db3dcb6584eb1c73b5570615b0f14fabb070b96117e351d",
"sha256:c68632c455ae0c46d1380033bae6d30014853fa3f600f4e14efc440be1bc9580",
"sha256:cabea05c000e49f0814b2611cbc66c2787f609d8a27fc7b9e97b5dab5d8502da",
"sha256:791f0a07985c2814a899cb0458802be06ba124a364f7e5a9413a1f08fdbf5b5c",
"sha256:f6d5815f290ee912fd4a768d97b46af39523dff584d786f5c0f7e9bdb7fad537",
"sha256:7d22e2347c1217a89bd3c79ca9adb4652c1e9b61427fffc0ab92227aacd19a38",
"sha256:55e9644f21c38d7707b4a432aacc7817c5414b68ac7a750e704c2f7100ebc15c"
]
},
"Metadata": {
"LastTagTime": "2025-03-05T08:29:40.211352843Z"
}
}
]
```

Exercise 8: Viewing Container Logs

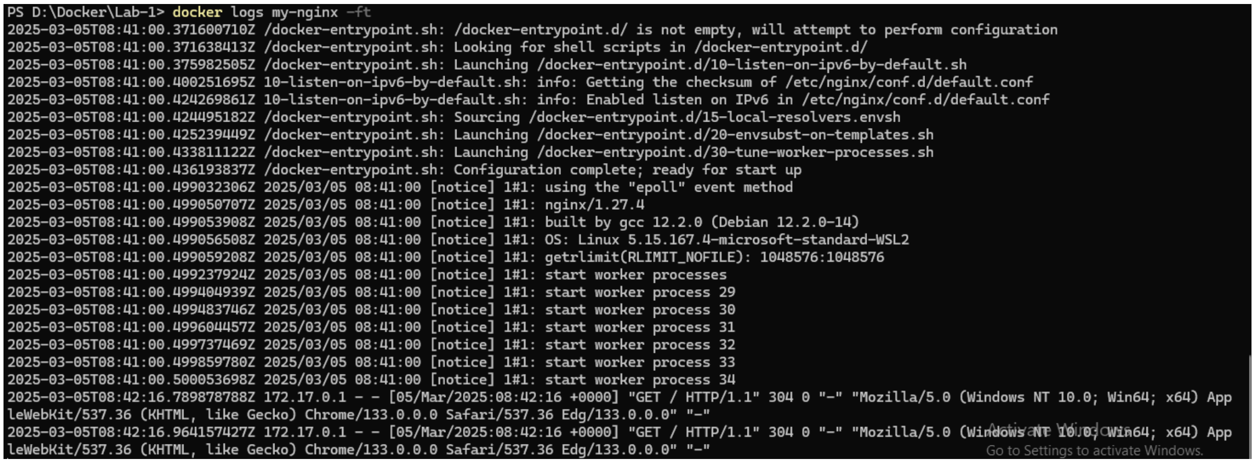
Check the runtime logs of a container, useful for debugging.

docker run --name my-nginx -d -p 8080:80 nginx:latest



```
PS D:\Docker\Lab-1> docker run --name my-nginx -d -p 8080:80 nginx:latest
72ba3b9f5dbe3d722e989c1b51256f36aee527df0bd08971be9a1d82374d82a7
```

docker logs my-nginx -ft



```
PS D:\Docker\Lab-1> docker logs my-nginx -ft
2025-03-05T08:41:00.371600710Z /docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
2025-03-05T08:41:00.371638413Z /docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
2025-03-05T08:41:00.375982505Z /docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
2025-03-05T08:41:00.400251695Z 10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
2025-03-05T08:41:00.424269861Z 10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
2025-03-05T08:41:00.424495182Z /docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
2025-03-05T08:41:00.425239449Z /docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
2025-03-05T08:41:00.433811122Z /docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
2025-03-05T08:41:00.436193837Z /docker-entrypoint.sh: Configuration complete; ready for start up
2025-03-05T08:41:00.499032306Z 2025/03/05 08:41:00 [notice] 1#1: using the "epoll" event method
2025-03-05T08:41:00.499050707Z 2025/03/05 08:41:00 [notice] 1#1: nginx/1.27.4
2025-03-05T08:41:00.499053908Z 2025/03/05 08:41:00 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
2025-03-05T08:41:00.499056508Z 2025/03/05 08:41:00 [notice] 1#1: OS: Linux 5.15.167.4-microsoft-standard-WSL2
2025-03-05T08:41:00.499059208Z 2025/03/05 08:41:00 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2025-03-05T08:41:00.499237924Z 2025/03/05 08:41:00 [notice] 1#1: start worker processes
2025-03-05T08:41:00.499404939Z 2025/03/05 08:41:00 [notice] 1#1: start worker process 29
2025-03-05T08:41:00.499483746Z 2025/03/05 08:41:00 [notice] 1#1: start worker process 30
2025-03-05T08:41:00.499604457Z 2025/03/05 08:41:00 [notice] 1#1: start worker process 31
2025-03-05T08:41:00.499737469Z 2025/03/05 08:41:00 [notice] 1#1: start worker process 32
2025-03-05T08:41:00.499859780Z 2025/03/05 08:41:00 [notice] 1#1: start worker process 33
2025-03-05T08:41:00.500053698Z 2025/03/05 08:41:00 [notice] 1#1: start worker process 34
2025-03-05T08:42:16.789878788Z 172.17.0.1 - - [05/Mar/2025:08:42:16 +0000] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) App
leWebKit/537.36 (KHTML, like Gecko) Chrome/133.0.0.0 Safari/537.36 Edg/133.0.0.0" "-"
2025-03-05T08:42:16.964157427Z 172.17.0.1 - - [05/Mar/2025:08:42:16 +0000] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) App
leWebKit/537.36 (KHTML, like Gecko) Chrome/133.0.0.0 Safari/537.36 Edg/133.0.0.0" "-"
```

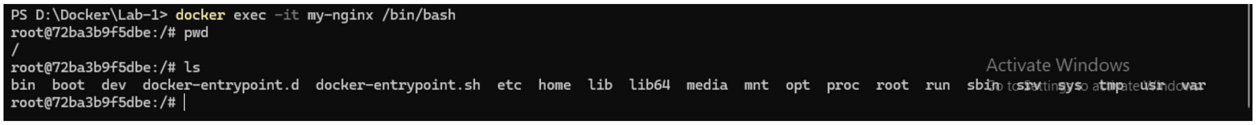
Explanation of flags:

- f means follow
- t means also print the timestamp

Exercise 9: Executing Commands in Containers

Execute commands inside a running container to interact with it.

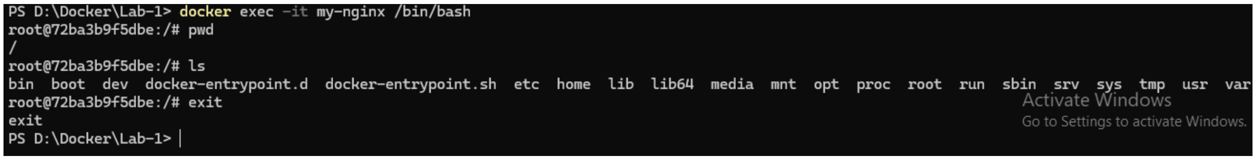
docker exec -it my-nginx /bin/bash



```
PS D:\Docker\Lab-1> docker exec -it my-nginx /bin/bash
root@72ba3b9f5dbe:/# pwd
/
root@72ba3b9f5dbe:/# ls
bin  boot  dev  docker-entrypoint.d  docker-entrypoint.sh  etc  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@72ba3b9f5dbe:/# |
```

The above command will take you inside the container, to come out run:

exit



```
PS D:\Docker\Lab-1> docker exec -it my-nginx /bin/bash
root@72ba3b9f5dbe:/# pwd
/
root@72ba3b9f5dbe:/# ls
bin  boot  dev  docker-entrypoint.d  docker-entrypoint.sh  etc  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@72ba3b9f5dbe:/# exit
exit
PS D:\Docker\Lab-1> |
```

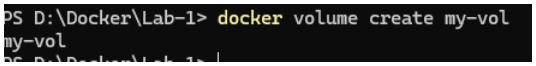
Explanation of flags:

- exec: Runs a command in a running container.
- it: Interactive terminal.

Exercise 10: Docker Volumes

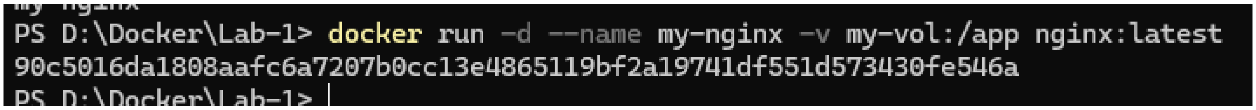
Manage data within Docker containers using volumes.

docker volume create my-vol

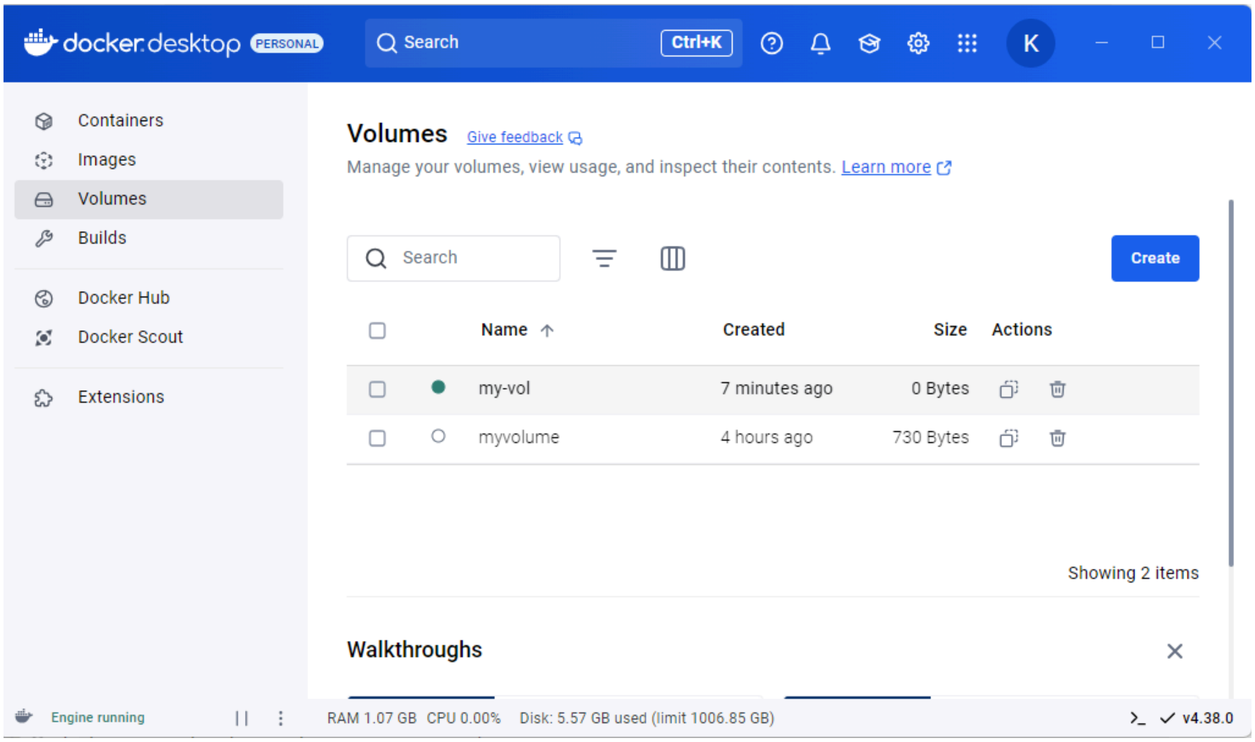


```
PS D:\Docker\Lab-1> docker volume create my-vol
my-vol
PS D:\Docker\Lab-1> |
```

docker run -d --name my-nginx -v my-vol:/app nginx:latest



```
my-nginx
PS D:\Docker\Lab-1> docker run -d --name my-nginx -v my-vol:/app nginx:latest
90c5016da1808aafc6a7207b0cc13e4865119bf2a19741df551d573430fe546a
PS D:\Docker\Lab-1> |
```



Volumes Give feedback

Manage your volumes, view usage, and inspect their contents. Learn more

□		Name ↑	Created	Size	Actions
□	●	my-vol	7 minutes ago	0 Bytes	
□	○	myvolume	4 hours ago	730 Bytes	

Showing 2 items

Explanation of flags:

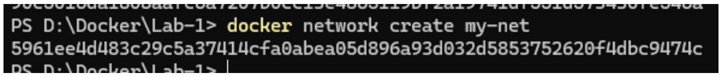
- volume create: Creates a new volume.
- v my-vol:/app: Mounts the volume my-vol at the /app directory inside the container.

Exercise 11: Docker Networks

Set up Docker networks to allow communication between two containers.

Create a network

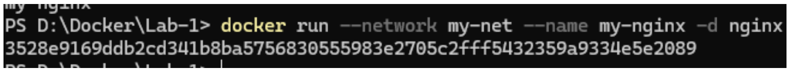
docker network create my-net



```
90c5016da1808aafc6a7207b0cc13e4865119bf2a19741df551d573430fe546a
PS D:\Docker\Lab-1> docker network create my-net
5961ee4d483c29c5a37414cfa0abea05d896a93d032d5853752620f4dbc9474c
PS D:\Docker\Lab-1> |
```

Create container1 using this network

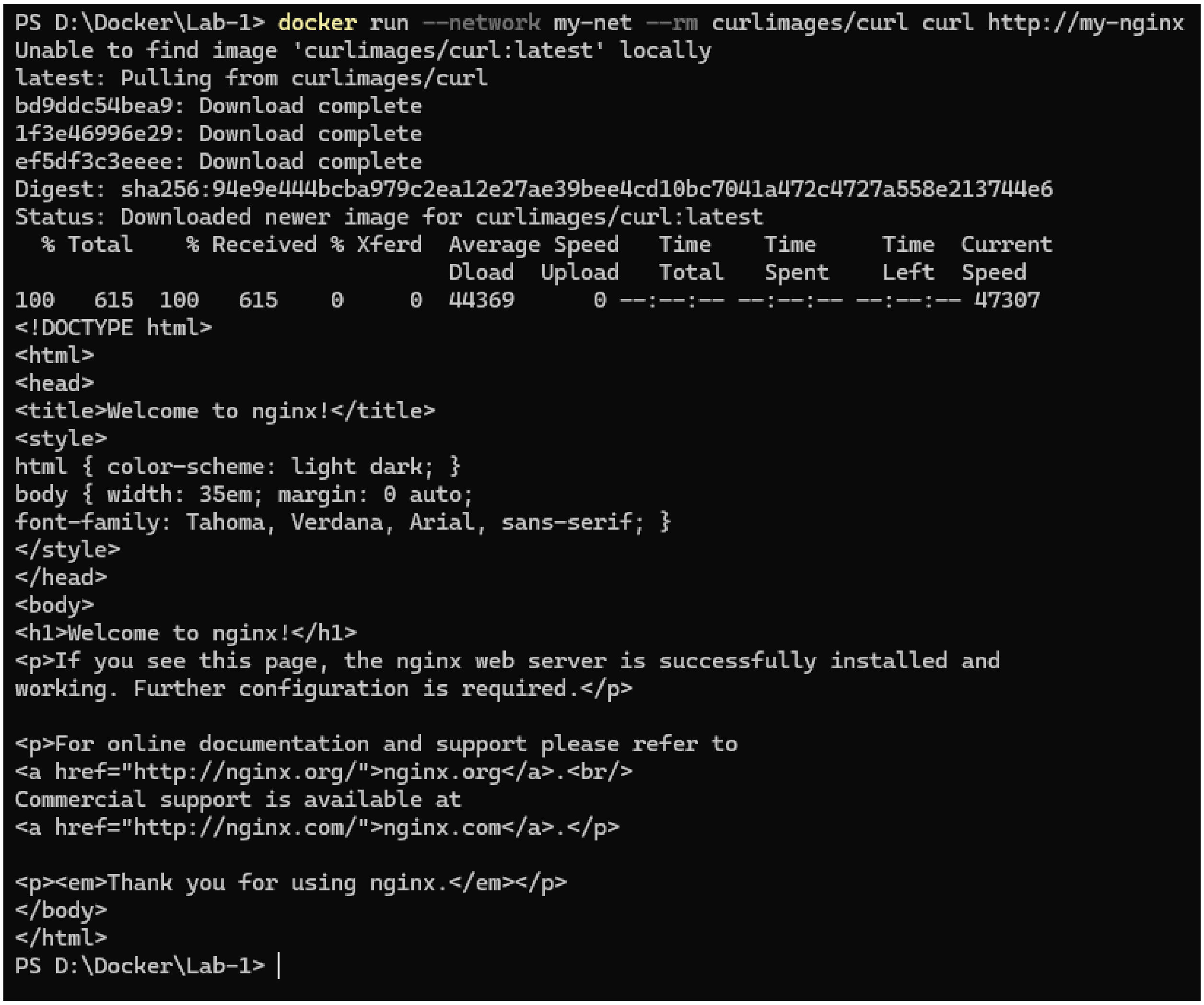
docker run --network my-net --name my-nginx -d nginx



```
my-nginx
PS D:\Docker\Lab-1> docker run --network my-net --name my-nginx -d nginx
3528e9169ddb2cd341b8ba5756830555983e2705c2fff5432359a9334e5e2089
PS D:\Docker\Lab-1> |
```

Create container2 using the same network and try to access container1

docker run --network my-net --rm curlimages/curl curl [http://my-nginx (Links to an external site.)]()



```
PS D:\Docker\Lab-1> docker run --network my-net --rm curlimages/curl curl http://my-nginx
Unable to find image 'curlimages/curl:latest' locally
latest: Pulling from curlimages/curl
bd9ddc54bea9: Download complete
1f3e46996e29: Download complete
ef5df3c3eeee: Download complete
Digest: sha256:94e9e444bcba979c2ea12e27ae39bee4cd10bc7041a472c4727a558e213744e6
Status: Downloaded newer image for curlimages/curl:latest
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   615  100   615    0     0  44369      0 --:--:-- --:--:-- --:--:-- 47307
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
PS D:\Docker\Lab-1> |
```

Explanation of flags:

- network create: Creates a new network.
- -network my-net: Connects a container to the my-net network.
- -rm: Automatically removes the container when it exits.

Exercise 12: Cleanup

Remove used images, containers, volumes, and networks to clean up your system.

docker rmi nginx:latest <your-username>/my-first-lab:latest

docker rm my-nginx

docker volume rm my-vol

docker network rm my-net

Now think how many commands you need to execute to cleanup thousands of containers in a production environment, thus simply run:

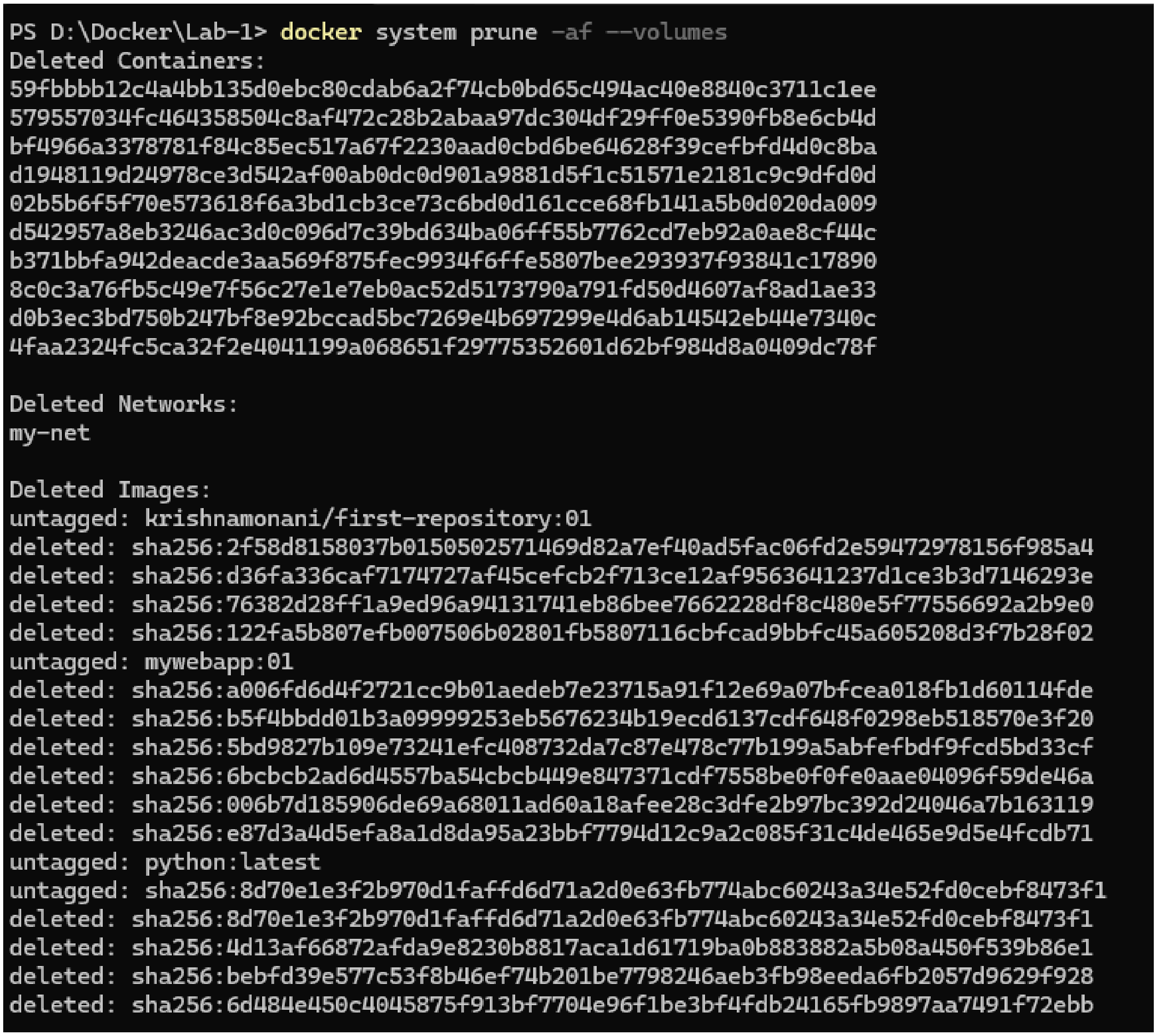
To cleanup everything like containers, images, network, build cache without volumes (Running containers will stay untouched)

docker system prune -af

OR

To cleanup everything like containers, images, network, build cache & volumes (Running containers will stay untouched)

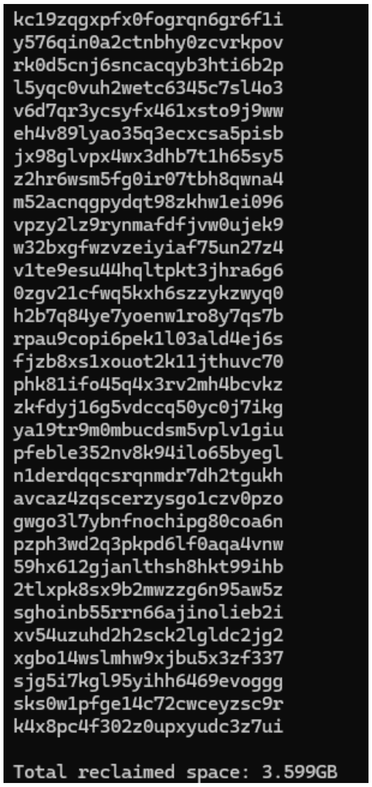
docker system prune -af --volumes



```
PS D:\Docker\Lab-1> docker system prune -af --volumes
Deleted Containers:
59fbbbb12c4a4bb135d0ebc80cdab6a2f74cb0bd65c494ac40e8840c3711c1ee
579557034fc464358504c8af472c28b2abaa97dc304df29ff0e5390fb8e6cb4d
bf4966a3378781f84c85ec517a67f2230aad0cbd6be64628f39cefbfd4d0c8ba
d1948119d24978ce3d542af00ab0dc0d901a9881d5f1c51571e2181c9c9dfd0d
02b5b6f5f70e573618f6a3bd1cb3ce73c6bd0d161cce68fb141a5b0d020da009
d542957a8eb3246ac3d0c096d7c39bd634ba06ff55b7762cd7eb92a0ae8cf44c
b371bbfa942deacde3aa569f875fec9934f6ffe5807bee293937f93841c17890
8c0c3a76fb5c49e7f56c27e1e7eb0ac52d5173790a791fd50d4607af8ad1ae33
d0b3ec3bd750b247bf8e92bccad5bc7269e4b697299e4d6ab14542eb44e7340c
4faa2324fc5ca32f2e4041199a068651f29775352601d62bf984d8a0409dc78f

Deleted Networks:
my-net

Deleted Images:
untagged: krishnamonani/first-repository:01
deleted: sha256:2f58d8158037b0150502571469d82a7ef40ad5fac06fd2e59472978156f985a4
deleted: sha256:d36fa336caf7174727af45cefcb2f713ce12af9563641237d1ce3b3d7146293e
deleted: sha256:76382d28ff1a9ed96a94131741eb86bee7662228df8c480e5f77556692a2b9e0
deleted: sha256:122fa5b807efb007506b02801fb5807116cbfcad9bbfc45a605208d3f7b28f02
untagged: mywebapp:01
deleted: sha256:a006fd6d4f2721cc9b01aedeb7e23715a91f12e69a07bfcea018fb1d60114fde
deleted: sha256:b5f4bbdd01b3a09999253eb5676234b19ecd6137cdf648f0298eb518570e3f20
deleted: sha256:5bd9827b109e73241efc408732da7c87e478c77b199a5abfefbdf9fcd5bd33cf
deleted: sha256:6bcbcb2ad6d4557ba54cbcb449e847371cdf7558be0f0fe0aae04096f59de46a
deleted: sha256:006b7d185906de69a68011ad60a18afee28c3dfe2b97bc392d24046a7b163119
deleted: sha256:e87d3a4d5efa8a1d8da95a23bbf7794d12c9a2c085f31c4de465e9d5e4fcdb71
untagged: python:latest
untagged: sha256:8d70e1e3f2b970d1faffd6d71a2d0e63fb774abc60243a34e52fd0cebf8473f1
deleted: sha256:8d70e1e3f2b970d1faffd6d71a2d0e63fb774abc60243a34e52fd0cebf8473f1
deleted: sha256:4d13af66872afda9e8230b8817aca1d61719ba0b883882a5b08a450f539b86e1
deleted: sha256:bebfd39e577c53f8b46ef74b201be7798246aeb3fb98eeda6fb2057d9629f928
deleted: sha256:6d484e450c4045875f913bf7704e96f1be3bf4fdb24165fb9897aa7491f72ebb
```



```
kc19zqgxpfx0fogrqn6gr6f1i
y576qin0a2ctnbhy0zcvrkpov
rk0d5cnj6sncacqyb3hti6b2p
l5yqc0vuh2wetc6345c7sl4o3
v6d7qr3ycsyfx461xsto9j9ww
eh4v89lyao35q3ecxcsa5pisb
jx98glvpx4wx3dhb7t1h65sy5
z2hr6wsm5fg0ir07tbh8qwna4
m52acnqgpydqt98zkhw1ei096
vpzy2lz9rynmafdfjvw0ujek9
w32bxgfwzvzeiyiaf75un27z4
v1te9esu44hqltpkt3jhra6g6
0zgv21cfwq5kxh6szzykzwyq0
h2b7q84ye7yoenw1ro8y7qs7b
rpau9copi6pek1l03ald4ej6s
fjzb8xs1xouot2k11jthuvc70
phk81ifo45q4x3rv2mh4bcvkz
zkfdyj16g5vdccq50yc0j7ikg
ya19tr9m0mbucdsm5vplv1giu
pfeble352nv8k94ilo65byegl
n1derdqqcsrqnmdr7dh2tgukh
avcaz4zqscerzysgo1czv0pzo
gwgo3l7ybnfnochipg80coa6n
pzph3wd2q3pkpd6lf0aqa4vnw
59hx612gjanlthsh8hkt99ihb
2tlxpk8sx9b2mwzzg6n95aw5z
sghoinb55rrn66ajinolieb2i
xv54uzuhd2h2sck2lgldc2jg2
xgbo14wslmhw9xjbu5x3zf337
sjg5i7kgl95yihh6469evoggg
sks0w1pfge14c72cwceyzsc9r
k4x8pc4f302z0upxyudc3z7ui

Total reclaimed space: 3.599GB
```