Docker Tasks

1. Docker Installation

Install Docker on your local system (Windows, macOS, or Linux).

```
Installed:
  containerd.io-1.7.28-1.el9.x86_64          docker-buildx-plugin-0.29.1
-1.el9.x86_64             docker-ce-3:28.5.1-1.el9.x86_64
  docker-ce-cli-1:28.5.1-1.el9.x86_64        docker-ce-rootless-extras-2
8.5.1-1.el9.x86_64        docker-compose-plugin-2.40.3-1.el9.x86_64

Complete!
```

Verify installation by running:

docker --version

docker info

```
[root@localhost ~]# docker --version
Docker version 28.5.1, build e180ab8
[root@localhost ~]# docker info
Client: Docker Engine - Community
 Version:    28.5.1
 Context:    default
 Debug Mode: false
 Plugins:
  buildx: Docker Buildx (Docker Inc.)
    Version:  v0.29.1
    Path:     /usr/libexec/docker/cli-plugins/docker-buildx
  compose: Docker Compose (Docker Inc.)
    Version:  v2.40.3
    Path:     /usr/libexec/docker/cli-plugins/docker-compose
```

2. Working with Docker Images

Search for the official Ubuntu image on Docker Hub.

```
[root@localhost ~]# docker search ubuntu
NAME                      DESCRIPTION                                     STARS     OFFICIAL
ubuntu                    Ubuntu is a Debian-based Linux operating sys…   17718     [OK]
ubuntu/squid              Squid is a caching proxy for the Web. Long-t…   119
ubuntu/nginx              Nginx, a high-performance reverse proxy & we…   133
ubuntu/cortex             Cortex provides storage for Prometheus. Long…   4
```

Pull it to your local system using docker pull.

```
[root@localhost ~]# docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
4b3ffd8ccb52: Pull complete
Digest: sha256:66460d557b25769b102175144d538d88219c077c678a49af4afca6fbf
c1b5252
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
```

List all available images using:

docker images

```
[root@localhost ~]# docker images
REPOSITORY   TAG       IMAGE ID       CREATED        SIZE
ubuntu       latest    97bed23a3497   4 weeks ago    78.1MB
```

3. Creating a Docker Container

Run a container using the Ubuntu image.

Execute a command inside the container:

echo "Hello, Docker!"

```
[root@localhost ~]# docker run ubuntu:latest echo "Hello, Docker!"
Hello, Docker!
[root@localhost ~]# docker container ls -a
CONTAINER ID   IMAGE           COMMAND                  CREATED         STATUS                     PORTS     NAMES
2007882e0906   ubuntu:latest   "echo 'Hello, Docker…"   6 seconds ago   Exited (0) 5 seconds ago             laughing_torvalds
```

4. Inspect Container (Created State)

Verify that the Ubuntu container is in the "created" state.

```
[root@localhost ~]# docker container ls -a
CONTAINER ID   IMAGE           COMMAND                  CREATED         STATUS                     PORTS     NAMES
eb2edefc5f71   ubuntu          "/bin/bash"              2 minutes ago   Created                              dreamy_curie
2007882e0906   ubuntu:latest   "echo 'Hello, Docker…"   8 minutes ago   Exited (0) 8 minutes ago             laughing_torvalds
```

```
[root@localhost ~]# docker inspect eb2edefc5f71
[
    {
        "Id": "eb2edefc5f710664bce1aff7c5c25ce6b5b1258ad5a739045d59c104ab3d975b",
        "Created": "2025-11-05T09:20:55.7753402372",
        "Path": "/bin/bash",
        "Args": [],
        "State": {
            "Status": "created",
```

5. Run Container (Running State)

Execute the Ubuntu container and confirm it’s in the "running" state.

```
[root@localhost ~]# docker run -d --name httpd -p 8080:80 httpd
0228ba33b7bfd095d9573dc6cb89d4a259a862aa8c8829cc228c92670d21fc3d
[root@localhost ~]# docker ps
CONTAINER ID   IMAGE     COMMAND              CREATED         STATUS         PORTS                                     NAMES
0228ba33b7bf   httpd     "httpd-foreground"   4 seconds ago   Up 3 seconds   0.0.0.0:8080->80/tcp, [::]:8080->80/tcp   httpd
```

6. Launch and Remove HTTP Container

Launch an HTTP container.

```
[root@localhost ~]# docker run -d --name httpd -p 8080:80 httpd
0228ba33b7bfd095d9573dc6cb89d4a259a862aa8c8829cc228c92670d21fc3d
```

Verify its state.

```
[root@localhost ~]# docker ps
CONTAINER ID   IMAGE     COMMAND              CREATED         STATUS         PORTS                                     NAMES
0228ba33b7bf   httpd     "httpd-foreground"   4 seconds ago   Up 3 seconds   0.0.0.0:8080->80/tcp, [::]:8080->80/tcp   httpd
```

Remove both the container and its image from your system.

```
[root@localhost ~]# docker stop 0228ba33b7bf
0228ba33b7bf
[root@localhost ~]# docker rm 0228ba33b7bf
0228ba33b7bf
[root@localhost ~]# docker rmi httpd:latest
Untagged: httpd:latest
Untagged: httpd@sha256:ecfd5ca1bfe1fc5e44a5836c5188bde7f397b50c7a5bb603a017543e29948a01
Deleted: sha256:6a4fe18d08d26a61b32d6aa5d17b2417c4f4de63e8460abf4cd824b911b9de88
Deleted: sha256:c78f79d223643af30c382eefcd1efc285a994a2895a4fad225330a6c3e1d1797
Deleted: sha256:655ea42223e5a87df151e8f2a0199419176fdf645c5d1504467f12df46f66a09
Deleted: sha256:c9dccfd4d12e5bac98e0c53df0fb6964e2cc26020cbd4dcf57419961db1dcf6b
Deleted: sha256:ed6e7e3a72de02ccd775dcd7b09dbc0b8dc41bc1e683027f3b7ab8764977be78
Deleted: sha256:006a1ccbeba5f22631ee410cb46c1c5293e3804f2e352dd32b99c3df49a26aef
Deleted: sha256:36d06fe0cbc654e5f67d58c960ed33e53127e4a3288d8ce6f6a60a9c311794d4
```

7. List Short Container IDs

Execute a command that displays only the short container IDs.

```
[root@localhost ~]# docker ps -aq
dc89541cc522
2007882e0906
```

8. Clone & Build Custom Image

Clone the repository: lancachenet/ubuntu-nginx

```
[root@localhost ~]# git clone https://github.com/lancachenet/ubuntu-nginx.git
Cloning into 'ubuntu-nginx'...
```

Build a Docker image using the provided Dockerfile.

Add relevant tags while building the image.

```
[root@localhost ubuntu-nginx]# docker build -t ubuntu-nginx .
[+] Building 110.8s (9/9) FINISHED
```

```
[root@localhost ubuntu-nginx]# docker images
REPOSITORY     TAG       IMAGE ID       CREATED          SIZE
ubuntu-nginx   latest    f80dbdb51c79   27 seconds ago   195MB
```

9. Run & Test the Webpage

Run a Docker container using the newly built image.

```
[root@localhost ~]# docker images
REPOSITORY              TAG       IMAGE ID       CREATED          SIZE
cloudethix_webserver    latest    0bc1c9f69bef   52 seconds ago   152MB
ubuntu-nginx            latest    f80dbdb51c79   6 minutes ago    195MB
ubuntu                  latest    97bed23a3497   4 weeks ago      78.1MB
[root@localhost ~]# docker container run -it --rm -d -p 8080:80 --name my_demo_webserver cloudethix_webserver
6e9ddfa56024286f67856529e7c63cf5e521ac4a38e8d4f3e27f9e504eb378d6
```

Use the curl command to verify that the webpage is accessible.

```
[root@localhost ~]# curl 10.10.56.231:8080
 <!doctype html>
 <html lang="en">
 <head>
 <meta charset="utf-8">
 <title>Cloud-Ethix, Docker Nginx</title>
 </head>
 <body>
 <h2>Hello from Nginx container, Cloud-Ethix</h2>
 </body>
 </html>
```

10. Container Detach & Reattach

Launch an Ubuntu container.

```
[root@localhost ~]# docker run -it ubuntu
root@6e9f491173cf:/# [root@localhost ~]# docker ps -a
CONTAINER ID   IMAGE                COMMAND                  CREATED              STATUS
          NAMES
6e9f491173cf   ubuntu               "/bin/bash"              13 seconds ago       Up 12 seconds
```

Demonstrate the escape sequence (detach).

While inside the container, press the escape sequence: Ctrl +p Ctrl +q

Reattach the container after detaching.

Check and note down the container's IP address.

```
[root@localhost ~]# docker attach 6e9f491173cf
root@6e9f491173cf:/# hostname -I
172.17.0.3
```

12. Inspect Host Configuration

Use a Docker command to inspect the hostname and the /etc/hosts file of the httpd container.

```
[root@localhost ~]# docker exec 2d2dcb383085 hostname
2d2dcb383085
[root@localhost ~]# docker exec 2d2dcb383085 cat /etc/hosts
127.0.0.1       localhost
::1     localhost ip6-localhost ip6-loopback
fe00::  ip6-localnet
ff00::  ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
172.17.0.3      2d2dcb383085
```

13. Monitor Container Stats

```
[root@localhost ~]# docker run -d --name ubuntu-cont ubuntu-nginx:latest
29f03e44217077291c712c0403d89b42df9b59f56fb5984d114dcf5c30062904
[root@localhost ~]# docker ps
CONTAINER ID   IMAGE                COMMAND                  CREATED          STATUS          PORTS
29f03e442170   ubuntu-nginx:latest  "/bin/bash -e /init/…"   8 seconds ago    Up 7 seconds    80/tcp
```

Inspect a container created with the lancachenet/ubuntu-nginx image.

```
[root@localhost ~]# docker inspect 29f03e442170
[
    {
        "Id": "29f03e44217077291c712c0403d89b42df9b59f56fb5984d114dcf5c30062904",
        "Created": "2025-11-05T10:25:03.178583124Z",
        "Path": "/bin/bash",
        "Args": [
            "-e",
            "/init/entrypoint",
            "/init/supervisord"
        ],
        "State": {
            "Status": "running",
```

Monitor its resource usage and stats using the top command.

```
CONTAINER ID   NAME          CPU %     MEM USAGE / LIMIT     MEM %     NET I/O     BLOCK I/O         PIDS
29f03e442170   ubuntu-cont   0.08%     29.27MiB / 764.7MiB   3.83%     3kB / 126B   29.5MB / 32.8kB   5
```

```
top - 10:28:15 up  1:45,  0 user,  load average: 0.37, 0.18, 0.12
Tasks:   6 total,   1 running,   5 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.0 us,  0.2 sy,  0.0 ni, 98.1 id,  0.0 wa,  1.7 hi,  0.0 si,  0.0 st
MiB Mem :    764.7 total,     61.3 free,    511.6 used,    322.2 buff/cache
MiB Swap:   2048.0 total,   1186.3 free,    861.7 used.    253.1 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
      1 root      20   0   37148  27960  11240 S   0.0   3.6   0:00.40 supervisord
     20 root      20   0    7340   3600   3352 S   0.0   0.5   0:00.00 startnginx.sh
     21 root      20   0   49984  10436   8972 S   0.0   1.3   0:00.02 nginx
     22 www-data  20   0   50404   3552   1772 S   0.0   0.5   0:00.00 nginx
     23 www-data  20   0   50404   3496   1716 S   0.0   0.4   0:00.00 nginx
     24 root      20   0   11836   5564   3404 R   0.0   0.7   0:00.04 top
```

14. Nginx Containers Log Review

Start a new Nginx container.

```
[root@localhost ~]# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS     NAMES
9f94602eff78   nginx     "/docker-entrypoint.…"   18 seconds ago   Up 17 seconds   80/tcp    nginx-new
```

Review and compare the logs for both Nginx containers.

```
[root@localhost ~]# docker logs ubuntu-cont > old.logs
[root@localhost ~]# docker logs nginx-new > new.logs
2025/11/05 10:32:12 [notice] 1#1: using the "epoll" event method
2025/11/05 10:32:12 [notice] 1#1: nginx/1.29.3
2025/11/05 10:32:12 [notice] 1#1: built by gcc 14.2.0 (Debian 14.2.0-19)
2025/11/05 10:32:12 [notice] 1#1: OS: Linux 5.14.0-617.el9.x86_64
2025/11/05 10:32:12 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1073741816:1073741816
2025/11/05 10:32:12 [notice] 1#1: start worker processes
2025/11/05 10:32:12 [notice] 1#1: start worker process 29
2025/11/05 10:32:12 [notice] 1#1: start worker process 30
```

```
[root@localhost ~]# diff old.logs new.logs
1,14c1,9
< Executing hook /hooks/entrypoint-pre.d/00_asciilogo.sh
<    _                 _____          _                       _
<   | |               / ____|        | |                     | |
<   | |     __ _ _ __ | |     __ _  ___| |__   ___   _ __   ___| |_
<   | |    / _` | '_ \| |    / _` |/ __| '_ \ / _ \ | '_ \ / _ \ __|
<   | |___| (_| | | | | |___| (_| | (__| | | |  __/_| | | |  __/ |_
<   |______\__,_|_| |_|\_____\__,_|\___|_| |_|\___(_)_| |_|\___|\__|
<
<
< Executing hook /hooks/supervisord-pre.d/20_test_files_setup
< Checking if /var/www/ is empty - Directory not empty.. don't touch content
< Executing hook /hooks/supervisord-pre.d/21_cleanup_log_files
< Cleaning up log files older than 7 days
< Starting Supervisord
---
> /docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
> /docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
> /docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
> 10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
> 10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
> /docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
> /docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
> /docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
> /docker-entrypoint.sh: Configuration complete; ready for start up
```

15. System Events & Filters

Examine Docker system events filtered by:

Specific date, and

```
[root@localhost ~]# docker events --since "2025-11-05T13:00:00" --until "2025-11-05T15:59:59"
2025-11-05T13:00:25.193961735+03:00 container create d79acac98f8c9526845fb7113b9eff369ba13730fc66cf7a3b5fe15382246034 (image=ubuntu, name=eager_yalo
w, org.opencontainers.image.ref.name=ubuntu, org.opencontainers.image.version=24.04)
2025-11-05T13:00:25.203081338+03:00 container attach d79acac98f8c9526845fb7113b9eff369ba13730fc66cf7a3b5fe15382246034 (image=ubuntu, name=eager_yalo
w, org.opencontainers.image.ref.name=ubuntu, org.opencontainers.image.version=24.04)
2025-11-05T13:00:25.660262928+03:00 network connect f3c9ba855a3be22c73cfdcbe5d0423abe000ecd012df120c10db810ff7edefff (container=d79acac98f8c9526845f
b7113b9eff369ba13730fc66cf7a3b5fe15382246034, name=bridge, type=bridge)
```

The last 30 minutes.

```
[root@localhost ~]# docker events --since "30m"
2025-11-05T13:17:42.710783768+03:00 container create 14ce7c5ac5ad4c4fd495495b93dd869da52b41a9f3f2ab6b88b7535afa7a936a (image=ubuntu-nginx:latest, ma
intainer=LanCache.Net Team <team@lancache.net>, name=gracious_mayer, org.opencontainers.image.ref.name=ubuntu, org.opencontainers.image.version=24.0
4)
2025-11-05T13:17:42.714695117+03:00 container attach 14ce7c5ac5ad4c4fd495495b93dd869da52b41a9f3f2ab6b88b7535afa7a936a (image=ubuntu-nginx:latest, ma
intainer=LanCache.Net Team <team@lancache.net>, name=gracious_mayer, org.opencontainers.image.ref.name=ubuntu, org.opencontainers.image.version=24.0
4)
```

Apply two filters simultaneously using name and event.

16. Custom Hostname & Logs

Create an Ubuntu container with: A meaningful name, and Hostname set to cloudethix.com.

```
[root@localhost ~]# docker run -it --name ubuntu-cloud --hostname cloudethix.com ubuntu
root@cloudethix:/#
```

Stop or kill the container.

Check its exit code and review its logs for details.

```
[root@localhost ~]# docker stop ubuntu-cloud
ubuntu-cloud
[root@localhost ~]# docker inspect ubuntu-cloud --format='{{.State.ExitCode}}'
137
```

17. Container Management Operations

Create and start an Ubuntu container with a random name, then rename it to Ubuntu01.

```
[root@localhost ~]# docker ps -a
CONTAINER ID   IMAGE     COMMAND       CREATED          STATUS                      PORTS     NAMES
853d10123232   ubuntu    "/bin/bash"   28 seconds ago   Exited (0) 27 seconds ago             musing_bhabha
[root@localhost ~]# docker rename musing_bhabha ubuntu01
[root@localhost ~]# docker ps -a
CONTAINER ID   IMAGE     COMMAND       CREATED              STATUS                          PORTS     NAMES
853d10123232   ubuntu    "/bin/bash"   About a minute ago   Exited (0) About a minute ago             ubuntu01
```

Launch another container named Ubuntu02.

```
[root@localhost ~]# docker run -dit --name ubuntu02 ubuntu
8dc3f5c0ac0ddf79cf6ed79cfdd5ef95975e8e15516e89c92680e1e388b58cda
[root@localhost ~]# docker ps -a
CONTAINER ID   IMAGE     COMMAND       CREATED         STATUS                     PORTS     NAMES
8dc3f5c0ac0d   ubuntu    "/bin/bash"   5 seconds ago   Up 5 seconds                         ubuntu02
```

Check the hostname for both containers.

```
[root@localhost ~]# docker exec ubuntu01 hostname
0e266f87cf4f
[root@localhost ~]# docker exec ubuntu02 hostname
8dc3f5c0ac0d
```

Perform the following:

Pause and unpause Ubuntu02.

```
[root@localhost ~]# docker pause ubuntu02
ubuntu02
[root@localhost ~]# docker unpause ubuntu02
ubuntu02
```

Stop, start, and restart Ubuntu01.

```
[root@localhost ~]# docker stop ubuntu01
ubuntu01
[root@localhost ~]# docker start ubuntu01
ubuntu01
[root@localhost ~]# docker restart ubuntu01
ubuntu01
```

Inspect stats and system events for both.

```
CONTAINER ID   NAME       CPU %     MEM USAGE / LIMIT   MEM %     NET I/O         BLOCK I/O   PIDS
0e266f87cf4f   ubuntu01   0.00%     840KiB / 764.7MiB   0.11%     2.76kB / 126B   0B / 0B     1
8dc3f5c0ac0d   ubuntu02   0.00%     868KiB / 764.7MiB   0.11%     3.96kB / 126B   0B / 0B     1
```

```
[root@localhost ~]# docker events --filter "container=ubuntu01" --filter "event=start" --filter "event=stop" --since "10m"
2025-11-05T14:27:13.041953293+03:00 container start 853d10123232a4bef4530c46a2a1565dfc046c08376ee680eb81f81127d6f2c6 (image=ubuntu, name=ubuntu01, o
rg.opencontainers.image.ref.name=ubuntu, org.opencontainers.image.version=24.04)
2025-11-05T14:27:58.811331749+03:00 container start 0e266f87cf4f63fb773f3ea41466c422501000e150571bff64ba95cd8581f15a (image=ubuntu, name=ubuntu01, o
rg.opencontainers.image.ref.name=ubuntu, org.opencontainers.image.version=24.04)
2025-11-05T14:30:35.594017544+03:00 container stop 0e266f87cf4f63fb773f3ea41466c422501000e150571bff64ba95cd8581f15a (image=ubuntu, name=ubuntu01, or
g.opencontainers.image.ref.name=ubuntu, org.opencontainers.image.version=24.04)
2025-11-05T14:30:42.247023319+03:00 container start 0e266f87cf4f63fb773f3ea41466c422501000e150571bff64ba95cd8581f15a (image=ubuntu, name=ubuntu01, o
rg.opencontainers.image.ref.name=ubuntu, org.opencontainers.image.version=24.04)
2025-11-05T14:31:02.233267811+03:00 container stop 0e266f87cf4f63fb773f3ea41466c422501000e150571bff64ba95cd8581f15a (image=ubuntu, name=ubuntu01, or
g.opencontainers.image.ref.name=ubuntu, org.opencontainers.image.version=24.04)
2025-11-05T14:31:02.710127755+03:00 container start 0e266f87cf4f63fb773f3ea41466c422501000e150571bff64ba95cd8581f15a (image=ubuntu, name=ubuntu01, o
rg.opencontainers.image.ref.name=ubuntu, org.opencontainers.image.version=24.04)
```

Finally, kill and delete the containers, ensuring all created directories are removed.

```
[root@localhost ~]# docker kill ubuntu01 ubuntu02
ubuntu01
ubuntu02
[root@localhost ~]# docker rm ubuntu01 ubuntu02
ubuntu01
ubuntu02
```

18. DockerHub Repository Setup

Create a DockerHub account.

Set up a repository with a meaningful name.