

- 1. Docker installation:** Install Docker on your local system (Windows, macOS, or Linux). Verify the installation by running the `docker --version` and `docker info` commands.
- 2. Working with Docker images:** Search for the official Ubuntu image on Docker Hub, pull it to your local system using `docker pull`, and list all available images on your system using `docker images`.
- 3. Creating a Docker container:** Run a Docker container using the Ubuntu image, and execute a command inside the container (e.g., `echo "Hello, Docker!"`).
- 4. Inspect the Ubuntu Docker container** to ensure it is in the "created" state only.
- 5. Execute the Ubuntu Docker container** and confirm that the container's state is "running."
- 6. Launch an HTTP container**, verify its state, and then remove the container and image from your local system.
- 7. Execute a command** to display a filtered list of containers, showing only their short container IDs.

Q1. Docker installation: Install Docker on your local system (Windows, macOS, or Linux). Verify the installation by running the docker --version and docker info commands.

```
root@FARDIN:~# docker --version
Docker version 24.0.7, build afdd53b
root@FARDIN:~#
```

```
root@FARDIN:/etc/selinux# docker

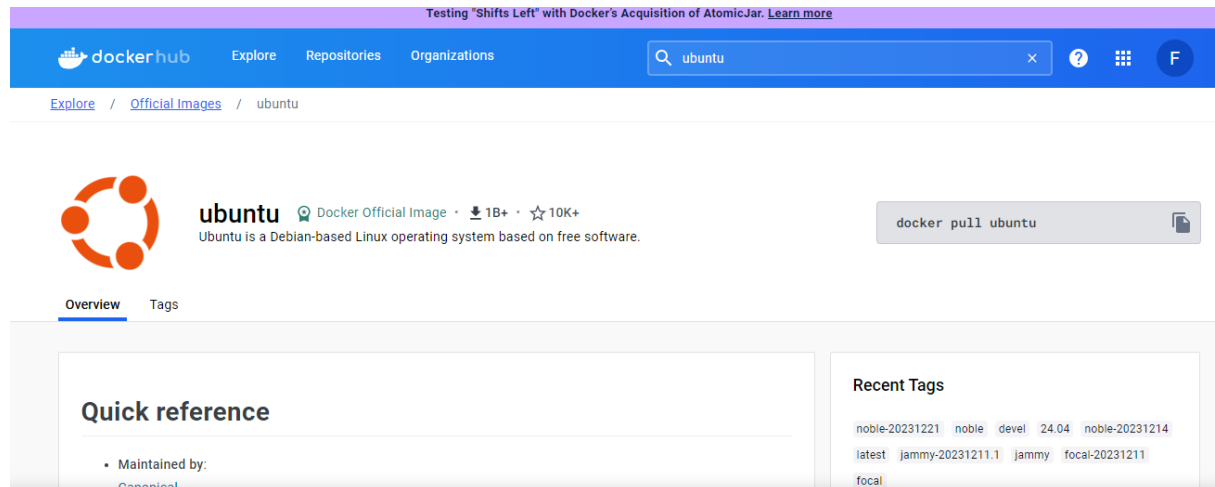
Usage:  docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Common Commands:
run      Create and run a new container from an image
exec     Execute a command in a running container
ps       List containers
build    Build an image from a Dockerfile
pull     Download an image from a registry
push     Upload an image to a registry
images   List images
login    Log in to a registry
logout   Log out from a registry
search   Search Docker Hub for images
version  Show the Docker version information
info     Display system-wide information

Management Commands:
builder  Manage builds
buildx*  Docker Buildx (Docker Inc., v0.11.2)
compose* Docker Compose (Docker Inc., v2.21.0)
container Manage containers
context  Manage contexts
image    Manage images
manifest Manage Docker image manifests and manifest lists
network  Manage networks
plugin   Manage plugins
system   Manage Docker
```

Q2. Working with Docker images: Search for the official Ubuntu image on Docker Hub, pull it to your local system using docker pull, and list all available images on your system using docker images.



Docker pull image

```
root@FARDIN:~# docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
a48641193673: Pull complete
Digest: sha256:6042500cf4b44023ea1894effe7890666b0c5c7871ed83a97c36c76ae560bb9b
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
```

Listing images

```
root@FARDIN:~# docker image ls
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
ubuntu        latest    174c8c134b2a   3 weeks ago   77.9MB
nginx         latest    d453dd892d93   2 months ago   187MB
httpd         latest    6fd77d7e5eb7   2 months ago   167MB
```

Q3. Creating a Docker container: Run a Docker container using the Ubuntu image, and execute a command inside the container (e.g., echo "Hello, Docker!").

Creating docker container

```
root@FARDIN:~# docker container create ubuntu
5fd3d77448feddcc84a109028a4dc62fbc2b04c08faeec9e949c59133ff6ae87
```

Starting docker container

```
root@FARDIN:~# docker container ls -a
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS
5fd3d77448fe   ubuntu    "/bin/bash"             44 seconds ago Created
atic_yonath    90162c3f8917  nginx                  "/docker-entrypoint...." 4 days ago    Exited (255) 2
st_almeida
root@FARDIN:~# docker container start 5fd3d77448fe
5fd3d77448fe
```

Running docker container

```
root@FARDIN:~# docker container run -itd ubuntu
0d90bde90c208c60c05f2eb1b1dea99fd9b19d09ce045f5e9d3d3fe3f82be7
root@FARDIN:~# docker container ls
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS          NAMES
0d90bde90c     ubuntu    "/bin/bash"             16 seconds ago Up 16 seconds          focused_shannon
```

```
Execute a command in a running container
root@FARDIN:~# docker container exec -it focused_shannon /bin/bash
root@0d90bde90c:/# echo "Hello, Docker!"
Hello, Docker!
root@0d90bde90c:/#
```

Q4. Inspect the Ubuntu Docker container to ensure it is in the "created" state only.

```
root@FARDIN:~# docker container create ubuntu
77585899c8a85fd2183cb74acc290c71d5d747f4b5baddc9060e08bfa3510cf2
root@FARDIN:~# docker container ls
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
root@FARDIN:~# docker container ls -a
CONTAINER ID   IMAGE     COMMAND   CREATED          STATUS      PORTS     NAMES
77585899c8a8   ubuntu    "/bin/bash"   14 seconds ago   Created             adoring_goldstine
```

```
root@FARDIN:~# docker container inspect adoring_goldstine
[
  {
    "Id": "77585899c8a85fd2183cb74acc290c71d5d747f4b5baddc9060e08bfa3510cf2",
    "Created": "2024-01-09T08:46:25.422680969Z",
    "Path": "/bin/bash",
    "Args": [],
    "State": {
      "Status": "created",
      "Running": false,
      "Paused": false,
      "Restarting": false,
      "OOMKilled": false,
      "Dead": false,
      "Pid": 0,
      "ExitCode": 0,
      "Error": "",
      "StartedAt": "0001-01-01T00:00:00Z",
      "FinishedAt": "0001-01-01T00:00:00Z"
    }
  },
]
```

Q5. Execute the Ubuntu Docker container and confirm that the container's state is "running."

```
root@FARDIN:~# docker container exec -it focused_shannon /bin/bash
root@0d90bdedc00c:/# echo "Hello, Docker!"
Hello, Docker!
root@0d90bdedc00c:/#
exit
root@FARDIN:~# docker container ls
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
0d90bdedc00c   ubuntu    "/bin/bash"   7 minutes ago   Up 7 minutes             focused_shannon
root@FARDIN:~# █
```

Q6. Launch an HTTP container, verify its state, and then remove the container and image from your local system.

```

root@FARDIN:~# docker container run -itd httpd
Unable to find image 'httpd:latest' locally
latest: Pulling from library/httpd
af107e978371: Already exists
eba4da411ea0: Already exists
4f4fb700ef54: Already exists
ed4d6291a6c2: Already exists
b42c390e1de9: Already exists
eafe388a0bb8: Already exists
Digest: sha256:f0a93744d8006e6f7ee5086c9ddccdcfa33d1091f15269a00547b4c382459c1f
Status: Downloaded newer image for httpd:latest
1d0d5314318aa1ff118847fd94501de64d7357d9c38a16402f9143a85d177a1c

```

```

root@FARDIN:~# docker image ls

```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
ubuntu	latest	174c8c134b2a	3 weeks ago	77.9MB
nginx	latest	d453dd892d93	2 months ago	187MB
httpd	latest	6fd77d7e5eb7	2 months ago	167MB

```

root@FARDIN:~# docker container ls

```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
1d0d5314318a	httpd	"httpd-foreground"	38 seconds ago	Up 37 seconds	80/tcp	tender_shannon
d034b8a7cb2a	ubuntu	"/bin/bash"	8 minutes ago	Up 8 minutes		relaxed_maxwell

```

root@FARDIN:~# docker container inspect tender_shannon
[
  {
    "Id": "1d0d5314318aa1ff118847fd94501de64d7357d9c38a16402f9143a85d177a1c",
    "Created": "2024-01-09T08:58:19.078688955Z",
    "Path": "httpd-foreground",
    "Args": [],
    "State": {
      "Status": "running",
      "Running": true,
      "Paused": false,
      "Restarting": false,
      "OOMKilled": false,
      "Dead": false,
      "Pid": 8784,
      "ExitCode": 0,
      "Error": "",
      "StartedAt": "2024-01-09T08:58:19.552242841Z",
      "FinishedAt": "0001-01-01T00:00:00Z"
    }
  },
  "tender_shannon"
]

```

```

root@FARDIN:~# docker container ls

```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
1d0d5314318a	httpd	"httpd-foreground"	4 minutes ago	Up 4 minutes	80/tcp	tender_shannon
d034b8a7cb2a	ubuntu	"/bin/bash"	12 minutes ago	Up 12 minutes		relaxed_maxwell

```

root@FARDIN:~# docker container stop tender_shannon
tender_shannon
root@FARDIN:~# docker container rm tender_shannon
tender_shannon
root@FARDIN:~# docker container ls

```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
d034b8a7cb2a	ubuntu	"/bin/bash"	13 minutes ago	Up 13 minutes		relaxed_maxwell

```

root@FARDIN:~#

```

```

root@FARDIN:~# docker image ls
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
ubuntu        latest    174c8c134b2a   3 weeks ago   77.9MB
nginx         latest    d453dd892d93   2 months ago   187MB
httpd         latest    6fd77d7e5eb7   2 months ago   167MB
root@FARDIN:~# docker image rm httpd
Error response from daemon: conflict: unable to remove repository reference "httpd" (must first delete or update image tagged httpd)
root@FARDIN:~# docker image rm -f httpd
Untagged: httpd:latest
Untagged: httpd@sha256:f0a93744d8006e6f7ee5086c9ddccdcfa33d1091f15269a00547b4c382459c1f
Deleted: sha256:6fd77d7e5eb732dacab601d4556c04a6c312928fb8989fe3b0a47d82db772441
root@FARDIN:~# docker image ls
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
ubuntu        latest    174c8c134b2a   3 weeks ago   77.9MB
nginx         latest    d453dd892d93   2 months ago   187MB

```

Q7. Execute a command to display a filtered list of containers, showing only their short container IDs.

```

root@FARDIN:~# docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS              PORTS
a15450cae8fb   6fd77d7e5eb7   "httpd-foreground"      17 minutes ago Exited (0) 17 minutes ago
jovial_solomon ubuntu         "/bin/bash"             23 minutes ago Up 23 minutes
relaxed_maxwell ubuntu         "/bin/bash"             27 minutes ago Created
adoring_goldstine ubuntu         "/bin/bash"             58 minutes ago Exited (137) 38 minutes ago
focused_shannon ubuntu         "/bin/bash"             About an hour ago Exited (0) About an hour ago
ecstatic_yonath ubuntu         "/bin/bash"             4 days ago Exited (255) 3 days ago
90162c3f8917   nginx         "/docker-entrypoint..." 4 days ago Exited (255) 3 days ago
tcp, :::85->80/tcp modest_almeida

```