Contents

[Docker Basics 4](#_Toc195273006)

[1) What is Docker? 4](#_Toc195273007)

[2) What is a hypervisor? 4](#_Toc195273008)

[3) What is virtualization? 4](#_Toc195273009)

[4) What is containerization? 4](#_Toc195273010)

[5) Difference between virtualization and containerization? 4](#_Toc195273011)

[6) What language is used in a Dockerfile? 4](#_Toc195273012)

[7) What are Docker images? 4](#_Toc195273013)

[8) What is Docker Hub? 4](#_Toc195273014)

[9) What is a Dockerfile? 5](#_Toc195273015)

[10) What is Docker architecture? 5](#_Toc195273016)

[11) What is Docker Compose? 5](#_Toc195273017)

[12) What is Docker Swarm? 5](#_Toc195273018)

[13) What are namespaces in Docker? 5](#_Toc195273019)

[Container Lifecycle 5](#_Toc195273020)

[14) Lifecycle of a container? 5](#_Toc195273021)

[15) What is Docker Machine? 5](#_Toc195273022)

[Advanced Questions 5](#_Toc195273023)

[1) What does docker system prune do? 5](#_Toc195273024)

[2) Is data lost when a container exits? 5](#_Toc195273025)

[3) Where is Docker used? 5](#_Toc195273026)

[4) How is Docker different from other containerization tools? 5](#_Toc195273027)

[5) Can I use JSON instead of YAML for Docker Compose? 5](#_Toc195273028)

[6) How scalable are Docker containers? 5](#_Toc195273029)

[7) Can you remove a paused container? 6](#_Toc195273030)

[8) Can a container restart automatically? 6](#_Toc195273031)

[9) Will cloud replace Docker? 6](#_Toc195273032)

[10) How many containers can run per host? 6](#_Toc195273033)

[11) Are stateful apps suitable for Docker? 6](#_Toc195273034)

[12) Does Docker Compose wait for dependencies? 6](#_Toc195273035)

[13) How to monitor Docker in production? 6](#_Toc195273036)

[14) Is Docker Compose suitable for production? 6](#_Toc195273037)

[Commands for Docker 6](#_Toc195273038)

[1) How do you check the version of Docker installed? 6](#_Toc195273039)

[2) How do you list all running containers? 6](#_Toc195273040)

[3) How do you list all containers (including stopped)? 6](#_Toc195273041)

[4) How do you stop a running container? 6](#_Toc195273042)

[5) How do you start a stopped container? 6](#_Toc195273043)

[6) How do you remove a container? 6](#_Toc195273044)

[7) How do you force remove a running container? 6](#_Toc195273045)

[8) How do you list all Docker images? 6](#_Toc195273046)

[9) How do you remove a Docker image? 6](#_Toc195273047)

[10) How do you build an image from a Dockerfile? 6](#_Toc195273048)

[11) How do you run a container from an image? 6](#_Toc195273049)

[12) How do you run a container in detached mode? 6](#_Toc195273050)

[13) How do you run a container and map ports? 6](#_Toc195273051)

[14) How do you run a container with a specific name? 7](#_Toc195273052)

[15) How do you execute a command inside a running container? 7](#_Toc195273053)

[16) How do you copy a file from host to container? 7](#_Toc195273054)

[17) How do you copy a file from container to host? 7](#_Toc195273055)

[18) How do you check logs of a container? 7](#_Toc195273056)

[19) How do you see the resource usage of containers? 7](#_Toc195273057)

[20) How do you list all Docker volumes? 7](#_Toc195273058)

[21) How do you create a Docker volume? 7](#_Toc195273059)

[22) How do you inspect a container’s details? 7](#_Toc195273060)

[23) How do you remove all stopped containers and unused images/volumes? 7](#_Toc195273061)

[24) How do you tag an image before pushing to Docker Hub? 7](#_Toc195273062)

[25) How do you push an image to Docker Hub? 7](#_Toc195273063)

# Docker Basics

1. What is Docker?  
   A platform to develop, ship, and run applications in lightweight, isolated containers using shared OS resources.
2. What is a hypervisor?  
   A software layer that creates and runs virtual machines (VMs), allowing multiple OS instances to share a single physical host.

Types:

Type 1 (Bare-metal): Runs directly on hardware (e.g., VMware ESXi).

Type 2 (Hosted): Runs on a host OS (e.g., VirtualBox).

1. What is virtualization?  
   Creating virtual versions of physical resources (servers, OS, storage) to run multiple isolated environments on one hardware system.

Types:

Hardware Virtualization: Uses hypervisors for full OS instances (e.g., VMware).

OS-Level Virtualization: Uses containers (e.g., Docker) to share the host OS.

1. What is containerization?  
   Bundling an application with its dependencies and configurations into an isolated, portable container.

### Difference between virtualization and containerization?

| Feature | Virtualization (VMs) | Containerization (Docker) |
| --- | --- | --- |
| Isolation | Full OS-level | Process-level (shares host OS) |
| Performance | Slower (OS overhead) | Faster (no OS boot) |
| Resource Usage | High (dedicated resources per VM) | Low (shared resources) |
| Use Case | Running different OSes | Isolated apps on the same OS |

1. What language is used in a Dockerfile?  
   A scripting language with its own syntax (not a programming language).
2. What are Docker images?  
   Read-only templates containing application code, dependencies, and settings to create containers.
3. What is Docker Hub?  
   A registry to store and share Docker images (like a warehouse for containers).
4. What is a Dockerfile?  
   A text file with instructions to automate Docker image builds (e.g., FROM, COPY, RUN).

### What is Docker architecture?

Daemon (dockerd) or Server: Manages containers/images.

Client (docker CLI): User interface.

Images/Containers: Templates/running instances.

Registry: Stores images (e.g., Docker Hub).

1. What is Docker Compose?  
   A tool to define and run multi-container apps using a docker-compose.yml file.
2. What is Docker Swarm?  
   Docker’s built-in orchestration tool for managing container clusters.
3. What are namespaces in Docker?  
   Linux kernel features isolating resources (processes, networks) for containers.

# Container Lifecycle

1. Lifecycle of a container?  
   Created → Running → (Paused ↔ Running) → Stopped → Deleted.
2. What is Docker Machine?  
   A tool to install Docker Engine on virtual hosts.

# Advanced Questions

1. What does docker system prune do?  
   Cleans up unused Docker objects (containers, networks, images) to free disk space.
2. Is data lost when a container exits?  
   No, data persists unless the container is deleted.
3. Where is Docker used?  
   CI/CD, cloud deployments, microservices, debugging, and multi-tenancy.
4. How is Docker different from other containerization tools?  
   Docker is cloud-agnostic, lightweight, and has a vast ecosystem (e.g., Docker Hub).
5. Can I use JSON instead of YAML for Docker Compose?  
   Yes, but YAML is the standard.
6. How scalable are Docker containers?  
   Highly scalable (used by Google, Heroku, etc.) with proper orchestration (e.g., Swarm/Kubernetes). States : Created, Running, Paused, Exited, Dead.
7. Can you remove a paused container?  
   No, it must be stopped first.
8. Can a container restart automatically?  
   Only if configured with --restart policies (e.g., --restart unless-stopped).
9. Will cloud replace Docker?  
   No, Docker and cloud services complement each other.
10. How many containers can run per host?  
    Unlimited, limited only by host resources.
11. Are stateful apps suitable for Docker?  
    Yes, with persistent volumes (docker volume create).
12. Does Docker Compose wait for dependencies?  
    Yes, it follows dependency order defined in docker-compose.yml.
13. How to monitor Docker in production?  
    Use docker stats, docker events, or tools like Prometheus.
14. Is Docker Compose suitable for production?  
    Yes, but orchestration tools (e.g., Kubernetes) are better for large-scale deployments.

# Commands for Docker

1. How do you check the version of Docker installed?  
   docker --version
2. How do you list all running containers?  
   docker ps
3. How do you list all containers (including stopped)?  
   docker ps -a
4. How do you stop a running container?  
   docker stop <container\_id>
5. How do you start a stopped container?  
   docker start <container\_id>
6. How do you remove a container?  
   docker rm <container\_id>
7. How do you force remove a running container?  
   docker rm -f <container\_id>
8. How do you list all Docker images?  
   docker images
9. How do you remove a Docker image?  
   docker rmi <image\_id>
10. How do you build an image from a Dockerfile?  
    docker build -t myimage:tag .
11. How do you run a container from an image?  
    docker run <image\_name>
12. How do you run a container in detached mode?  
    docker run -d <image\_name>
13. How do you run a container and map ports?  
    docker run -p 8080:80 <image\_name>
14. How do you run a container with a specific name?  
    docker run --name mycontainer <image\_name>
15. How do you execute a command inside a running container?  
    docker exec -it <container\_id> bash
16. How do you copy a file from host to container?  
    docker cp hostfile.txt <container\_id>:/path/
17. How do you copy a file from container to host?  
    docker cp <container\_id>:/path/file.txt ./
18. How do you check logs of a container?  
    docker logs <container\_id>
19. How do you see the resource usage of containers?  
    docker stats
20. How do you list all Docker volumes?  
    docker volume ls
21. How do you create a Docker volume?  
    docker volume create myvolume
22. How do you inspect a container’s details?  
    docker inspect <container\_id>
23. How do you remove all stopped containers and unused images/volumes?  
    docker system prune
24. How do you tag an image before pushing to Docker Hub?  
    docker tag myimage username/myimage:tag
25. How do you push an image to Docker Hub?  
    docker push username/myimage:tag