



COMPLETE DOCKER BASIC COMMANDS - ENHANCED & ORGANIZED

▮ SYSTEM & INFO COMMANDS

- `docker --help` - Show all Docker commands and options^[1]
- `docker version` - Display Docker version info (client & server)^{[2] [1]}
- `docker info` - Show system-wide Docker information^[1]
- `docker system df` - Check Docker disk space usage^[3]
- `docker system prune` - Remove all unused containers, networks, images^[1]
- `docker system prune -a` - Remove everything + unused images^[3]

▮ IMAGE MANAGEMENT COMMANDS

Basic Image Commands

- `docker images` **or** `docker image ls` - List all images^{[4] [1]}
- `docker pull image-name` - Download image from registry^{[4] [1]}
- `docker pull image-name:tag` - Pull specific version^[1]
- `docker build -t name:tag .` - Build image from Dockerfile^{[4] [1]}
- `docker tag source target` - Create new tag for image^{[4] [1]}
- `docker push image-name` - Upload image to registry^{[1] [4]}

Image Cleanup Commands

- `docker rmi image-name` - Remove image^{[4] [1]}
- `docker rmi -f image-name` - Force remove image^[1]
- `docker image prune` - Remove unused images^{[4] [1]}
- `docker image prune -a` - Remove all unused images^[4]

Image Information Commands

- `docker inspect image-name` - Show detailed image info^[1] ^[4]
- `docker history image-name` - Show image layer history^[1] ^[4]
- `docker save image > file.tar` - Export image to tar file^[4]
- `docker load < file.tar` - Import image from tar file^[4]

CONTAINER LIFECYCLE COMMANDS

Create & Run Containers

- `docker create image-name` - Create container (don't start)^[5] ^[6]
- `docker run image-name` - Create + start container^[6] ^[1]
- `docker run -d image-name` - Run in background (detached)^[6] ^[1]
- `docker run -it image-name` - Run interactive with terminal^[6] ^[1]
- `docker run --name my-container image` - Give container a name^[6] ^[1]
- `docker run --rm image` - Auto-remove when container stops^[6] ^[1]

Start/Stop Containers

- `docker start container-name` - Start stopped container^[5] ^[6]
- `docker stop container-name` - Stop running container^[5] ^[6]
- `docker restart container-name` - Restart container^[5] ^[6]
- `docker pause container-name` - Pause container processes^[5] ^[6]
- `docker unpause container-name` - Resume paused container^[5] ^[6]
- `docker kill container-name` - Force kill container^[6]

Container Information Commands

- `docker ps` - Show running containers^[6] ^[1]
- `docker ps -a` - Show all containers (running + stopped)^[6] ^[1]
- `docker ps -q` - Show only container IDs^[1]
- `docker inspect container-name` - Detailed container info^[6] ^[1]
- `docker stats` - Live resource usage statistics^[5] ^[6]
- `docker top container-name` - Show running processes in container^[5] ^[6]
- `docker port container-name` - Show port mappings^[6]

▮ INTERACT WITH CONTAINERS

Execute Commands

- `docker exec -it container-name sh` - Open shell in container^[7] ^[1]
- `docker exec -it container-name bash` - Open bash shell^[7]
- `docker exec container-name command` - Run single command^[7] ^[1]
- `docker exec -d container-name command` - Run command in background^[7]
- `docker exec -u username container-name command` - Run as specific user^[7]

View Logs & Debug

- `docker logs container-name` - Show container logs^[1] ^[6]
- `docker logs -f container-name` - Follow live logs^[1]
- `docker logs --tail 50 container-name` - Show last 50 log lines^[1]
- `docker logs --since 1h container-name` - Show logs from last hour^[1]
- `docker attach container-name` - Attach to running container^[6]

▮ FILE OPERATIONS

- `docker cp file.txt container:/path` - Copy file TO container^[8] ^[6]
- `docker cp container:/path/file.txt .` - Copy file FROM container^[8] ^[6]
- `docker diff container-name` - Show file changes in container^[6]
- `docker export container > file.tar` - Export container filesystem^[6]

▮ CLEANUP COMMANDS

Remove Containers

- `docker rm container-name` - Remove stopped container^[1] ^[6]
- `docker rm -f container-name` - Force remove running container^[6] ^[1]
- `docker rm -v container-name` - Remove container + its volumes^[5]
- `docker container prune` - Remove all stopped containers^[1] ^[6]

System Cleanup

- `docker system prune` - Remove unused containers, networks, images^[1]
- `docker system prune -a` - Remove everything unused^[1]
- `docker system prune -f` - Force cleanup without confirmation^[1]

▮ NETWORKING COMMANDS

- `docker network ls` - List all networks^[1]
- `docker network create network-name` - Create custom network^[1]
- `docker network connect network container` - Connect container to network^[1]
- `docker network disconnect network container` - Disconnect from network^[1]
- `docker network inspect network-name` - Show network details^[1]
- `docker network rm network-name` - Remove network^[1]
- `docker network prune` - Remove unused networks^[1]

▮ VOLUME COMMANDS

- `docker volume ls` - List all volumes^{[8] [1]}
- `docker volume create volume-name` - Create named volume^{[8] [1]}
- `docker volume inspect volume-name` - Show volume details^{[8] [1]}
- `docker volume rm volume-name` - Remove volume^{[8] [1]}
- `docker volume prune` - Remove unused volumes^{[8] [1]}

▮ REGISTRY & AUTH COMMANDS

- `docker login` - Login to Docker Hub^{[8] [1]}
- `docker login registry-url` - Login to private registry^[1]
- `docker logout` - Logout from registry^{[8] [1]}
- `docker search term` - Search Docker Hub for images^[1]

⚙️ ADVANCED RUN OPTIONS

Port & Network

- `-p 8080:80` - Map host port 8080 to container port 80^[1]
- `-P` - Publish all exposed ports to random host ports^[1]
- `--network=network-name` - Connect to specific network^[1]
- `--network=host` - Use host network (no isolation)^[1]

Environment & Variables

- `-e VAR=value` - Set environment variable^[1]
- `--env-file=file.env` - Load environment from file^[1]
- `-w /app` - Set working directory^[1]

Memory & CPU Limits

- `--memory=512m` - Limit memory usage^[1]
- `--cpus=1.5` - Limit CPU usage^[1]
- `--restart=always` - Auto-restart policy^[1]

Volume Mounting

- `-v host-path:container-path` - Bind mount^[1]
- `-v volume-name:container-path` - Named volume mount^[1]
- `--mount type=bind,src=host-path,dst=container-path` - Advanced mount^[1]

Ready for your questions or let's move to Docker Compose! ☐

✱

1. <https://www.cherryservers.com/blog/docker-commands-cheat-sheet>
2. <https://spacelift.io/blog/docker-commands-cheat-sheet>
3. <https://pagertree.com/blog/docker-commands-cheat-sheet>
4. <https://docs.docker.com/reference/cli/docker/image/>
5. <https://www.linuxteck.com/docker-management-command-cheat-sheet/>
6. <https://docs.docker.com/reference/cli/docker/container/>
7. <https://www.digitalocean.com/community/tutorials/how-to-use-docker-exec-to-run-commands-in-a-docker-container>
8. <https://www.geeksforgeeks.org/devops/docker-cheat-sheet/>
9. <https://dev.to/waji97/docker-image-management-3558>
10. <https://www.geeksforgeeks.org/devops/docker-instruction-commands/>
11. <https://docs.docker.com/docker-hub/repos/manage/hub-images/manage/>
12. <https://labs.iximiuz.com/tutorials/docker-container-management-commands>
13. <https://www.theknowledgeacademy.com/blog/docker-images-command/>
14. https://docs.docker.com/get-started/docker_cheatsheet.pdf
15. <https://www.docker.com/products/cli/>
16. <https://buddy.works/tutorials/docker-commands-cheat-sheet>
17. <https://k21academy.com/docker-kubernetes/docker-container-lifecycle-management/>
18. https://www.reddit.com/r/docker/comments/17zldvu/the_ultimate_docker_cheat_sheet/
19. <https://blog.devops.dev/docker-101-the-complete-cheat-sheet-for-devs-ops-01ba7b13193a>
20. <https://dockerlabs.collabnix.com/docker/cheatsheet/>