

Here's a list of the top 200 most commonly used Docker commands, along with a brief of each. This list includes commands for managing images, containers, networks, volumes, and more:



1. Basic Docker Commands

1. `docker --version` - Display the installed Docker version.
2. `docker info` - Display system-wide information about Docker.
3. `docker help` - Show help for Docker commands.
4. `docker pull <image>` - Download an image from Docker Hub.
5. `docker push <image>` - Upload an image to Docker Hub.
6. `docker images` - List all Docker images on the local machine.
7. `docker rmi <image>` - Remove a Docker image.
8. `docker search <term>` - Search for images on Docker Hub.

2. Container Management

9. `docker run <image>` - Create and start a container from an image.

10. `docker ps` - List running containers.
11. `docker ps -a` - List all containers, including stopped ones.
12. `docker exec -it <container> <command>` - Execute a command in a running container.
13. `docker stop <container>` - Stop a running container.
14. `docker start <container>` - Start a stopped container.
15. `docker restart <container>` - Restart a container.
16. `docker rm <container>` - Remove a stopped container.
17. `docker logs <container>` - Fetch logs from a container.
18. `docker attach <container>` - Attach to a running container's console.
19. `docker commit <container> <new-image>` - Create a new image from a container's changes.
20. `docker rename <old-container> <new-container>` - Rename a container.

3. Image Management

21. `docker build -t <image-name>:<tag> <path>` - Build an image from a Dockerfile.
22. `docker tag <image> <new-image>` - Tag an image for easier reference.
23. `docker history <image>` - Show the history of an image.
24. `docker inspect <image>` - Display detailed information about an image.
25. `docker save -o <file>.tar <image>` - Save an image to a tar file.
26. `docker load -i <file>.tar` - Load an image from a tar file.

4. Docker Networking

- 27. `docker network ls` - List all Docker networks.
- 28. `docker network create <network-name>` - Create a new Docker network.
- 29. `docker network rm <network-name>` - Remove a Docker network.
- 30. `docker network inspect <network-name>` - Display detailed information about a network.
- 31. `docker network connect <network> <container>` - Connect a container to a network.
- 32. `docker network disconnect <network> <container>` - Disconnect a container from a network.

5. Docker Volumes

- 33. `docker volume ls` - List all Docker volumes.
- 34. `docker volume create <volume-name>` - Create a new Docker volume.
- 35. `docker volume rm <volume-name>` - Remove a Docker volume.
- 36. `docker volume inspect <volume-name>` - Display detailed information about a volume.
- 37. `docker run -v <volume-name>:<container-path> <image>` - Mount a volume in a container.

6. Docker Compose

- 38. `docker-compose up` - Start services defined in a `docker-compose.yml` file.
- 39. `docker-compose down` - Stop and remove services defined in a `docker-compose.yml` file.

- 40. `docker-compose build` - Build or rebuild services defined in a `docker-compose.yml` file.
- 41. `docker-compose logs` - View logs from services defined in a `docker-compose.yml` file.
- 42. `docker-compose exec <service> <command>` - Execute a command in a running service.
- 43. `docker-compose ps` - List containers for services defined in a `docker-compose.yml` file.

7. Docker Swarm

- 44. `docker swarm init` - Initialize a Docker Swarm.
- 45. `docker swarm join` - Join a node to an existing Swarm.
- 46. `docker swarm leave` - Remove a node from a Swarm.
- 47. `docker service create <service>` - Create a new service in a Swarm.
- 48. `docker service ls` - List services in a Swarm.
- 49. `docker service scale <service>=<replicas>` - Scale a service in a Swarm.

8. Docker Registry

- 50. `docker login` - Log in to a Docker registry.
- 51. `docker logout` - Log out from a Docker registry.
- 52. `docker registry` - Interact with a Docker registry.

9. Docker System

- 53. `docker system df` - Show disk usage of Docker objects.
- 54. `docker system prune` - Remove unused data to free up space.

- 55. `docker system events` - Get real-time events from the Docker daemon.

10. Container Statistics

- 56. `docker stats` - Display a live stream of container(s) resource usage statistics.
- 57. `docker top <container>` - Display the running processes of a container.

11. Docker Configuration

- 58. `docker config ls` - List all Docker configs.
- 59. `docker config create <config-name>` - Create a new Docker config.
- 60. `docker config rm <config-name>` - Remove a Docker config.

12. Docker Secrets

- 61. `docker secret ls` - List all Docker secrets.
- 62. `docker secret create <secret-name> <file>` - Create a new Docker secret.
- 63. `docker secret rm <secret-name>` - Remove a Docker secret.

13. Advanced Docker Commands

- 64. `docker update <container>` - Update configuration of a running container.
- 65. `docker cp <container>:<path> <local-path>` - Copy files/folders from a container to the local filesystem.
- 66. `docker cp <local-path> <container>:<path>` - Copy files/folders from the local filesystem to a container.

- 67. `docker inspect <container>` - Display detailed information about a container.
- 68. `docker wait <container>` - Block until a container stops, then print its exit code.

14. Docker Health Checks

- 69. `docker run --health-cmd "<command>"` - Specify a command to check the health of a container.
- 70. `docker run --health-interval <duration>` - Specify how often to perform health checks.
- 71. `docker run --health-timeout <duration>` - Specify a maximum time to allow one check to run.

15. Environment Variables

- 72. `docker run -e <key>=<value>` - Set an environment variable in a container.
- 73. `docker run --env-file <file>` - Set environment variables from a file.

16. Container Restart Policies

- 74. `docker run --restart <policy>` - Specify a restart policy for a container.

17. Docker Build Options

- 75. `docker build --no-cache` - Build an image without using cache.
- 76. `docker build --build-arg <key>=<value>` - Set build-time variables.

18. Networking Options

- 77. `docker run --network <network>` - Specify the network a container should join.

78. `docker run --ip <ip-address>` - Specify an IP address for the container.

19. Volumes and Bind Mounts

79. `docker run -v <host-path>:<container-path>` - Bind mount a host path to a container path.
80. `docker run --mount type=volume,source=<volume>,target=<path>` - Specify a volume mount.

20. Container Execution

81. `docker run -d <image>` - Run a container in detached mode.
82. `docker run -it <image>` - Run a container in interactive mode with a terminal.

21. Managing Docker Daemon

83. `docker daemon` - Start the Docker daemon.
84. `docker run --privileged` - Run a container with extended privileges.

22. Container Cleanup

85. `docker rm -f <container>` - Forcefully remove a running container.
86. `docker rmi -f <image>` - Forcefully remove an image.

23. Docker Plugin Management

87. `docker plugin ls` - List installed plugins.
88. `docker plugin install <plugin>` - Install a new plugin.
89. `docker plugin remove <plugin>` - Remove a plugin.

24. Inspecting and Logging

- 90. `docker inspect <object>` - Get detailed information about a Docker object.
- 91. `docker logs -f <container>` - Follow logs from a container.

25. Docker Contexts

- 92. `docker context ls` - List Docker contexts.
- 93. `docker context create <context>` - Create a new Docker context.
- 94. `docker context use <context>` - Set the current Docker context.

26. Resource Management

- 95. `docker update --cpu-shares <value> <container>` - Update the CPU shares of a container.
- 96. `docker update --memory <value> <container>` - Update the memory limit of a container.

27. Managing Docker Desktop

- 97. `docker desktop` - Open Docker Desktop application.

28. Checking for Updates

- 98. `docker version --format '{{.Client.Version}}'` - Get the version of the Docker client.
- 99. `docker version --format '{{.Server.Version}}'` - Get the version of the Docker server.

29. Docker Machine Commands

- 100. `docker-machine create <machine-name>` - Create a new Docker Machine.
- 101. `docker-machine ls` - List all Docker Machines.

- 102. `docker-machine env <machine-name>` - Display environment variables for the Docker Machine.
- 103. `docker-machine stop <machine-name>` - Stop a Docker Machine.
- 104. `docker-machine start <machine-name>` - Start a Docker Machine.

30. Setting Up Dockerfile

- 105. `FROM <image>` - Specify the base image.
- 106. `RUN <command>` - Execute a command in the image during build.
- 107. `CMD ["executable","param1","param2"]` - Specify the default command to run in a container.
- 108. `ENTRYPOINT ["executable", "param1", "param2"]` - Set the entry point for a container.
- 109. `WORKDIR /path` - Set the working directory for instructions.
- 110. `COPY <source> <destination>` - Copy files or directories from host to image.
- 111. `ADD <source> <destination>` - Copy files or directories from host to image with URL support and automatic unpacking of compressed files.
- 112. `ENV <key> <value>` - Set environment variables in the image.
- 113. `EXPOSE <port>` - Inform Docker that the container listens on the specified network ports at runtime.
- 114. `VOLUME ["/data"]` - Create a mount point with the specified path and mark it as holding externally mounted volumes from native host or other containers.

31. Docker Compose File Configuration

- 115. version: '3' - Specify the Compose file format version.
- 116. services: - Define the services in the Compose file.
- 117. image: - Specify the image for a service.
- 118. build: - Specify the build context for a service.
- 119. ports: - Map container ports to host ports.
- 120. volumes: - Mount volumes into the service container.
- 121. environment: - Set environment variables for a service.
- 122. depends_on: - Specify dependencies between services.

32. Health Check Configuration

- 123. healthcheck: - Configure health checks for services in Docker Compose.

33. Advanced Networking Options

- 124. driver: - Specify the network driver to use.
- 125. ipam: - Configure IP address management options.

34. Scaling Services in Docker Compose

- 126. docker-compose up --scale <service>=<replicas> - Scale a service to a specified number of replicas.

35. Deploying Applications

- 127. docker deploy <stack> - Deploy a stack to a Swarm.
- 128. docker stack ls - List deployed stacks in a Swarm.

36. Managing Swarm Nodes

- 129. docker node ls - List nodes in the Swarm.
- 130. docker node update <node> - Update a node's configuration.

- 131. `docker node promote <node>` - Promote a node to a manager in a Swarm.
- 132. `docker node demote <node>` - Demote a manager to a worker in a Swarm.

37. Container Orchestration

- 133. `docker stack deploy -c <file> <stack>` - Deploy a stack using a Compose file.

38. Troubleshooting

- 134. `docker exec -it <container> /bin/bash` - Start a shell in a running container.
- 135. `docker inspect --format='{{json .State}}' <container>` - Inspect the state of a container in JSON format.
- 136. `docker logs --tail <number> <container>` - Fetch the last few lines of logs from a container.

39. Managing Docker Services

- 137. `docker service inspect <service>` - Inspect a service.
- 138. `docker service update <service>` - Update a service.
- 139. `docker service remove <service>` - Remove a service.

40. Docker Secrets Management

- 140. `docker secret inspect <secret>` - Inspect a Docker secret.

41. Docker Configurations Management

- 141. `docker config create <name> <file>` - Create a configuration from a file.

42. Advanced Volume Management

- 142. `docker volume prune` - Remove all unused volumes.

43. Resource Limiting Options

- 143. `--cpus` - Limit the number of CPUs available to a container.
- 144. `--memory-swappiness` - Set the memory swappiness of a container.

44. Docker Context Switching

- 145. `docker context create <context-name> --docker <host>` - Create a new Docker context with a specified host.
- 146. `docker context inspect <context-name>` - Inspect a Docker context.

45. Cleaning Up Docker Environment

- 147. `docker system prune -a` - Remove unused data, including images, containers, and networks.

46. Interactive Mode Commands

- 148. `docker run -it <image> bash` - Start a container in interactive mode with a bash shell.

47. Docker File System Commands

- 149. `docker run --rm -v $(pwd):/app -w /app <image> <command>` - Run a command in a mounted volume.

48. Docker Build Arguments

- 150. `ARG <name>[=<default>]` - Define a variable that users can pass at build-time to the Dockerfile with the `docker build` command.

49. Docker Compose Override Files

- 151. `docker-compose -f <file> -f <override-file> up` - Use multiple Compose files to configure services.

50. Container Startup Options

- 152. `--entrypoint` - Override the default entry point of the image.

51. Docker Build Context

153. `docker build --file <dockerfile> <context>` - Specify the Dockerfile and build context.

52. Docker Info Command Options

154. `docker info --format` - Format the output of the info command.

53. Using Docker Hub

155. `docker hub` - Access Docker Hub from the command line.

54. Container Environment Variables

156. `docker run -e KEY=value` - Set an environment variable when starting a container.

55. Container Exit Code

157. `docker inspect <container> --format='{{.State.ExitCode}}'` - Get the exit code of a container.

56. Handling Stopped Containers

158. `docker ps -f "status=exited"` - List stopped containers.

57. Container Networking Options

159. `--network-alias` - Add an alias to a container in a specified network.

58. Multi-Stage Builds

160. `FROM <base-image> AS <stage-name>` - Define a stage in a multi-stage Dockerfile.

59. Docker System Commands

161. `docker system df -v` - Show detailed disk usage of Docker objects.

60. Networking Command Options

162. `docker network connect --alias <alias> <network>`
`<container>` - Connect a container to a network with an alias.

61. Docker Compose Variables

163. `COMPOSE_PROJECT_NAME` - Specify a project name for Docker Compose.

62. Working with Docker Volumes

164. `docker volume prune --force` - Force remove unused volumes without confirmation.

63. Container Capabilities

165. `--cap-add` - Add Linux capabilities to a container.
166. `--cap-drop` - Drop Linux capabilities from a container.

64. Docker Security Options

167. `--security-opt` - Set security options for a container.

65. Custom Docker Networks

168. `docker network create --driver <driver> <network-name>` - Create a custom network with a specified driver.

66. Docker Inspect Command Options

169. `docker inspect --format='{{.NetworkSettings.IPAddress}}'`
`<container>` - Get the IP address of a container.

67. Docker Events

170. `docker events` - Get a real-time stream of events from the Docker server.

68. Service Health Checks

171. `docker service update --health-cmd <command>`
`<service>` - Update the health check command for a service.

69. Image Layers

172. `docker history --no-trunc <image>` - Show the history of an image without truncating the output.

70. Docker Build Caching

173. `docker build --no-cache=true` - Disable build cache.

71. Container Restart Policies

174. `docker run --restart always` - Always restart a container unless it is explicitly stopped.

72. Managing Docker Swarm Services

175. `docker service logs <service>` - View logs for a specific service in a Swarm.

73. Service Scaling

176. `docker service scale <service>=<replicas>` - Scale a service to a specified number of replicas.

74. Container Resource Allocation

177. `docker run --memory-swap <memory>` - Set the total memory limit (memory + swap).

75. Docker Volume Mount Options

178. `--mount type=bind,source=<host-path>,target=<container-path>` - Specify a bind mount.

76. Docker System Cleanup

179. `docker system prune --volumes` - Remove unused data, including volumes.

77. File Sharing in Docker

180. `-v <host-path>:<container-path>` - Create a bind mount to share files between host and container.

78. Running Containers in Detached Mode

181. `docker run -d <image>` - Run a container in detached mode.

79. Container Health Status

182. `docker inspect <container> --format='{{.State.Health.Status}}'` - Get the health status of a container.

80. Using Docker Logs

183. `docker logs -f <container>` - Follow logs of a running container.

81. Viewing Docker Disk Usage

184. `docker system df` - Show disk usage for Docker objects.

82. Creating a Docker Network

185. `docker network create <network-name>` - Create a new Docker network.

83. Inspecting Docker Networks

186. `docker network inspect <network-name>` - Inspect a Docker network.

84. Removing Docker Networks

187. `docker network rm <network-name>` - Remove a Docker network.

85. Container Resource Limits

188. `--cpu-shares` - Set CPU shares for a container.

86. Docker Compose with Multiple Files

189. `docker-compose -f <file1> -f <file2> up` - Use multiple Docker Compose files.

87. Managing Docker Contexts

190. `docker context use <context-name>` - Switch to a different Docker context.

88. Docker Run Command Options

191. `--rm` - Automatically remove the container when it exits.

89. Using Docker with Systemd

192. `sudo systemctl start docker` - Start the Docker service on systems using systemd.

90. Checking Docker Version

193. `docker --version` - Show the Docker version.

91. Docker Command Line Help

194. `docker <command> --help` - Display help for a specific Docker command.

92. Running a Container with a Specific User

195. `docker run -u <username>` - Run a container with a specific user.

93. Building Images with a Dockerfile

196. `docker build -t <image-name> .` - Build an image from a Dockerfile in the current directory.

94. Running Containers in Background

197. `docker run -d <image>` - Start a container in detached mode.

95. Docker Networking Commands

198. `docker network create -d <driver> <network-name>` - Create a new network with a specific driver.

96. Displaying Docker System Information

199. `docker version` - Show version information for Docker client and server.

97. Running a Shell in a Container

200. `docker exec -it <container> sh` - Start a shell in a running container using sh.