

Docker Management Command's

Command	Description	Example
Container Commands		
docker create	Create a new container from an image.	docker create ubuntu:latest
docker start	Start with one or more stop containers.	docker start my-container
docker stop	Stop running one or more containers.	docker stop my-container
docker restart	Restart one or more running containers.	docker restart my-container
docker pause	Pause the process inside a running container.	docker pause my-container
docker unpause	Unpause the process inside a paused container.	docker unpause my-container
docker rename	Rename the container.	docker rename my-container new-container-name
docker rm	Remove one or more containers.	docker rm my-container
docker ps	List the running containers on the host.	docker ps
docker stats	Display real-time resource usage statistics for one or more containers.	docker stats my-container
docker top	Display the processes running inside a container.	docker top my-container
docker logs	Display the logs generated by the container.	docker logs my-container
Image Commands		
docker pull	This command is used to pull an image from a Docker registry.	docker pull ubuntu:latest
docker build	This command is used to build a new Docker image from a Dockerfile.	docker build -t my-image.

Command	Description	Example
docker push	This command is used to push a Docker image to a Docker registry.	docker push my-image
docker tag	This command is used to tag a Docker image with a new name or version.	docker tag my-image my-image:latest
docker rmi	This command is used to remove one or more Docker images.	docker rmi my-image
docker images	This command is used to list the Docker images available on the host.	docker images
docker history	This command is used to display the history of a Docker image.	docker history my-image
docker save	This command is used to save a Docker image to a tar archive.	docker save my-image -o my-image.tar
docker load	This command is used to load a Docker image from a tar archive.	docker load -i my-image.tar
Network Commands		
docker network create	Create a new Docker network	docker network create my-network
docker network connect	Connect a container to a Docker network	docker network connect my-network my-container
docker network disconnect	Disconnect the container from the Docker network	docker network disconnect my-network my-container
docker network ls	List the Docker networks on the host	docker network ls
docker network inspect	Display detailed information about the Docker network	docker network inspect my-network
docker network rm	Remove Docker network	docker network rm my-network
Volume Commands		
docker volume create	This command is used to create a new Docker volume.	docker volume create my-volume

Command	Description	Example
<code>docker volume ls</code>	This command is used to list the Docker volumes on the host.	<code>docker volume ls</code>
<code>docker volume inspect</code>	This command is used to display detailed information about a Docker volume.	<code>docker volume inspect my-volume</code>
<code>docker volume rm</code>	This command is used to remove a Docker volume.	<code>docker volume rm my-volume</code>
<code>docker run -v</code>	This option is used to create and mount a volume to a container at runtime.	<code>docker run -v my-volume:/app/data my-image</code>
<code>docker inspect -f '{{.Mounts}}'</code>	This command is used to display the mount information of a container, including the volume(s) it's using.	<code>docker inspect -f '{{.Mounts}}' my-container</code>
System Commands		
<code>docker version</code>	This command displays the Docker version information that is currently installed on the system.	<code>docker version</code>
<code>docker info</code>	This command displays Docker system-wide information, including the number of running containers, images, and storage driver information.	<code>docker info</code>
<code>docker events</code>	This command shows the real-time events from the Docker daemon, such as container creation, deletion, or network creation.	<code>docker events</code>
<code>docker system df</code>	This command shows the disk usage of the Docker system, including space used by images, containers, and volumes.	<code>docker system df</code>
<code>docker system prune</code>	This command is used to free up disk space by removing all unused resources, including containers, images,	<code>docker system prune</code>

Command	Description	Example
	volumes, and networks.	
docker login	This command is used to log in to a Docker registry. You need to authenticate it before pushing or pulling images.	docker login
docker logout	This command is used to log out from the Docker registry.	docker logout
Docker Compose Commands		
docker-compose up	Create and start all containers defined in the Compose file.	docker-compose up
docker-compose down	Stop and remove all containers defined in the Compose file.	docker-compose down
docker-compose build	Build or rebuild the services defined in the Compose file.	docker-compose build
docker-compose start	Start all containers defined in the Compose file.	docker-compose start
docker-compose stop	Stop all containers defined in the Compose file.	docker-compose stop
docker-compose logs	Displays the logs of all containers defined in the Compose file.	docker-compose logs
Docker Swarm Commands		
docker swarm init	Initialize a new Docker Swarm cluster on the current Docker host.	docker swarm init --advertise-addr <manager-node-ip-address>
docker swarm join	Join a Docker Swarm cluster as a worker node or manager node.	docker swarm join --token <join-token> <manager-node-ip-address>:<port>
docker swarm leave	Leave the Docker Swarm cluster by stopping and removing the node from the cluster.	docker swarm leave --force
docker stack deploy	Deploy the Docker stack to the Docker Swarm cluster.	docker stack deploy --compose-file <docker-compose-file> <stack-name>
docker service rm	Remove the service from the Docker Swarm cluster.	docker service rm <service-name>

Command	Description	Example
docker service create	Creates a new service in the Docker Swarm cluster.	docker service create --name <service-name> --replicas <number-of-replicas> <image-name>
docker service ls	List all services in the Docker Swarm cluster.	docker service ls
Registry Commands		
docker login	Log in to the Docker registry server.	docker login myregistry.com
docker logout	Log out from the Docker registry server.	docker logout myregistry.com
docker search	Searches for an image on Docker Hub or other registries.	docker search nginx
docker pull	Pull an image from the registry to your local machine.	docker pull nginx
docker push	Push an image from your local machine to register.	docker push myregistry.com/myimage:tag
docker tag	Tags an image with a new name and/or tag.	docker tag myimage myregistry.com/myimage:tag
Debugging Commands		
docker ps	This command lists all the running containers with their details such as container ID, image used, command, status, etc.	docker ps
docker logs	This command shows the logs of a container.	docker logs container_name
docker exec	This command is used to execute a command inside a running container.	docker exec container_name ls -l /
docker inspect	This command is used to get detailed information about a container or an image.	docker inspect container_name
docker port	This command is used to list the port mappings of a container.	docker port container_name
docker top	This command is used to see the processes running inside a container.	docker top container_name

Command	Description	Example
Dockerfile Commands		
FROM	set the base image to build the Dockerfile.	FROM ubuntu:latest
RUN	run the command during the image building process.	RUN apt-get update && apt-get install -y curl
COPY	copy files and directories from the host into the Docker image.	COPY app /app
CMD	specifies the default command to run the container starts.	CMD ["node", "app.js"]
WORKDIR	set the working directory for any RUN, CMD, ENTRYPOINT, COPY, or ADD commands that follow it.	WORKDIR /app
ENV	sets of environmental variables that can be used during the image building process or when the container is running.	ENV NODE_ENV production
EXPOSE	document the ports that the container listens to at runtime.	EXPOSE 8080
VOLUME	create a mount point for the volume in the container.	VOLUME /data
USER	set the user or UID to where the container should run as.	USER node
Multi-Stage Build Commands		
FROM	Specify the base image to start the build process.	FROM alpine:latest
WORKDIR	Set a working directory for any RUN, CMD, ENTRYPOINT, COPY, and ADD commands that follow it.	WORKDIR /app
COPY	Copying files or directories from the build context into the container. Suitable for multiple uses.	COPY app.py /app/

Command	Description	Example
RUN	Executes a command inside the container during build time.	RUN pip install -r requirements.txt
ARG	Declares a variable that can be passed on to the Docker build command using --build-arg.	ARG version
CMD or ENTRYPOINT	During container startup, both are Dockerfile instructions that specify what command to run.	["python", "app.py"]
ENV	Sets environment variables for the container.	ENV FLASK_APP=app.py
LABEL	Label describes an image or container as a key-value pair.	LABEL <'key'>=<'value'>
Health Check Commands		
HEALTHCHECK	Using this command, you can check the health of a container.	HEALTHCHECK --interval=5m --timeout=3s CMD curl -f http://localhost/ exit 1
docker inspect --format='{{json .State.Health}}' <container>	Using this command, you can inspect a container's health status and get detailed information about its current state.	docker inspect --format='{{json .State.Health}}' my-container
docker container ls -filter health=unhealthy	This command is used to list all containers that have a failed health check.	docker container ls --filter health=unhealthy
Config Commands		
docker config create	Creates a new config with the specified name and content.	docker config create myconfig myconfig.txt
docker config inspect	Displays detailed information about the config.	docker config inspect myconfig
docker config ls	List all configs.	docker config ls
docker config rm	Remove one or more configs.	docker config rm myconfig
docker config update	Updates the config with new content.	docker config update myconfig myconfig-updated.txt

Command	Description	Example
docker service create	Create a new service with one or more configs.	docker service create --name myservice --config source=myconfig,target=/app/config.txt myimage
docker service update	Update a service with one or more configs.	docker service update --config-rm myoldconfig --config-add mynewconfig myservice
Buildx Commands		
docker buildx ls	List all the available builders.	docker buildx ls
docker buildx create	Creates a new builder instance.	docker buildx create --name mybuilder
docker buildx use	Sets the current builder context.	docker buildx use mybuilder
docker buildx inspect	Displays detailed information about the current builder instance.	docker buildx inspect --bootstrap
docker buildx build	Builds an image using the current builder context.	docker buildx build --platform linux/amd64,linux/arm64 -t myimage:latest