Docker Commands in Linux & Ubuntu for DevOps

Docker is an open-source platform that automates the deployment of applications inside lightweight, portable containers. For DevOps engineers, Docker is a key tool used for CI/CD, scaling, and efficient application management. Below is a categorized list of commonly used Docker commands in Linux/Ubuntu environments with explanations.

1. Basic Commands

- docker --version : Show Docker version.
- docker info : Display system-wide information about Docker installation.
- docker help : Show available Docker commands.

2. Image Management

- docker pull : Download an image from Docker Hub.
- docker images : List all downloaded images.
- docker rmi : Remove an image.
- docker build -t . : Build an image from Dockerfile in the current directory.
- docker tag : Assign a tag to an image.

3. Container Management

- docker ps : List running containers.
- docker ps -a : List all containers (including stopped ones).
- docker run -it : Run container interactively with terminal access.
- docker run -d : Run container in detached (background) mode.
- docker run -p 8080:80 : Map host port 8080 to container port 80.
- docker start : Start a stopped container.
- docker stop : Stop a container.
- docker restart : Restart a container.
- docker rm : Remove a container.

4. Working Inside Containers

- docker exec -it /bin/bash : Access container shell interactively.
- docker logs : View logs of a container.
- docker top : Show processes running inside container.
- docker inspect : Display detailed configuration and state of a container.

5. Volumes & Storage

- docker volume ls : List all volumes.
- docker volume create : Create a new volume.
- docker run -v :/path/in/container : Attach volume to container.
- docker volume rm : Remove a volume.

6. Networks

- docker network ls : List all networks.
- docker network create : Create a new network.
- docker network connect : Connect a container to a network.
- docker network disconnect : Disconnect container from a network.

7. Docker Compose Commands

- docker-compose up : Start all services defined in docker-compose.yml.
- docker-compose down : Stop and remove containers, networks, and volumes.
- docker-compose ps : List containers managed by docker-compose.
- docker-compose logs : View logs of services.

8. Cleanup Commands

- docker system prune : Remove unused data (containers, images, networks).
- docker container prune : Remove all stopped containers.
- docker image prune : Remove unused images.
- docker volume prune : Remove unused volumes.

9. Best Practices for DevOps Engineers

- Use descriptive container and image names for better management.
- Avoid running containers as root for security reasons.
- Clean up unused images and containers regularly to save disk space.
- Use Docker Compose for multi-container applications.
- Integrate Docker with CI/CD pipelines for automation.