

Here is a compiled Q&A for Docker interview preparation:

Docker Interview Questions and Answers

1. Core Components of Docker Architecture

- **Docker Daemon** (`dockerd`): Runs in the background, builds and manages containers.
- **Docker Client (CLI)**: Interacts with the daemon via Docker commands.
- **Docker Registry**: Stores images (Docker Hub, private registries).
- **Docker Objects**: Images, Containers, Networks, Volumes.

2. Docker Image vs Container

- **Image**: Read-only blueprint.
- **Container**: Running instance with a writable layer.

3. CMD vs ENTRYPOINT

- **CMD**: Default command, can be overridden.
- **ENTRYPOINT**: Fixed command, CMD can provide default args.

4. COPY vs ADD

- **COPY**: Copy files/directories from host.
- **ADD**: COPY + remote URLs + auto-extract tar archives.

5. Docker Networking

- **Bridge**: Default, container-to-container communication on same host.
- **Host**: Shares host network.
- **Overlay**: Multi-host communication (Swarm/K8s).
- **None**: No network.
- Use user-defined bridge for DNS-based container name resolution.

6. Volumes, Bind Mounts, tmpfs

- **Volumes**: Docker-managed, persistent.
- **Bind Mounts**: Host-managed, flexible, less portable.
- **tmpfs**: In-memory, fast, non-persistent.

7. Multi-Stage Build

- Multiple `FROM` statements.
- Only copy necessary artifacts to final stage.
- Reduces final image size, improves security.

8. Troubleshooting Containers

- `docker logs <container>`
- `docker ps -a` for exit codes

- `docker exec -it <container> /bin/sh` for interactive debugging
- Check entrypoint, dependencies, resource limits, restart policies.

9. Docker Compose vs Swarm

- **Compose:** Single-host, multi-container orchestration.
- **Swarm:** Multi-host, production-grade orchestration, scaling, HA.
- Compose: `docker-compose up`, Swarm: `docker stack deploy`.

10. Docker Run vs Exec vs Attach

- **docker run:** Start new container.
- **docker exec:** Run command inside running container.
- **docker attach:** Attach terminal to running container process.

11. Reducing Docker Image Size

- Use smaller base image (`alpine`).
- Multi-stage builds.
- Clean cache/temp files.
- Combine RUN commands.
- Install only necessary packages.

12. Container Restart (CrashLoop) Troubleshooting

- Check logs.
- Inspect exit code.
- Run interactively.
- Check resource constraints.
- Verify entrypoint/CMD.
- Check dependencies.

13. Docker Networking Between Containers

- Default bridge: no container name resolution.
- User-defined bridge: containers can communicate using names.
- Overlay network: multi-host networking.

This document can be exported as a PDF for your **quick revision** before interviews.