**Docker Questions Intermediate to Advance:**

**Intermediate Docker Interview Questions**

1. What are the key differences between Docker containers and virtual machines?

2. How do Docker images differ from containers?

3. Explain the use of docker-compose and its advantages.

4. What is the purpose of the Dockerfile? Walk through a basic example.

5. How does Docker networking work by default?

6. How do you create a custom Docker network and why?

7. What’s the role of a Docker volume?

8. What are the types of Docker volumes and when do you use them?

9. How can you persist data in Docker containers?

10. What is the purpose of the EXPOSE instruction in a Dockerfile?

11. How do you inspect the logs of a Docker container?

12. How do you access a running container’s shell?

13. What’s the difference between CMD and ENTRYPOINT in a Dockerfile?

14. How do you restart a container automatically on failure?

15. How do you limit CPU and memory usage for a container?

16. What is a bind mount and how does it differ from a volume?

17. What happens when a container is deleted? Is the data lost?

18. What are Docker tags and how are they used in image management?

19. What is the difference between docker stop and docker kill?

20. How do you clean up unused images and containers?

21. How can you check the container’s resource usage?

22. What are the lifecycle states of a Docker container?

23. What’s the difference between COPY and ADD in Dockerfiles?

24. What are Docker labels used for?

25. What is the .dockerignore file and why is it useful?

26. How do you handle environment variables in Docker?

27. What is a multi-stage build in Docker and why is it useful?

28. How can you see which container is using which port?

29. How do you connect multiple containers using Docker Compose?

30. What is the purpose of a Docker registry?

31. How do you pull a specific version of a Docker image?

32. How do you share images within an organization securely?

33. What’s the purpose of health checks in a Dockerfile?

34. How do you build a Docker image from a Dockerfile?

35. How do you push Docker images to a remote registry?

36. What’s the difference between docker ps and docker ps -a?

37. What is an orphaned volume? How do you handle it?

38. How do you debug a Docker build that’s failing?

39. How do you list all Docker networks?

40. What are the differences between bridge, host, and none networks?

41. How do you make a container communicate with a host service?

42. How do you copy files from a container to the host?

43. How do you remove all stopped containers?

44. What are container image layers?

45. What is a dangling image? How do you remove it?

46. How do you inspect the details of an image?

47. What happens when two containers try to use the same port?

48. What is the use of --link in Docker networking?

49. What’s the difference between docker exec and docker run?

50. How do you update a Docker container with a new image version?

**Advanced Docker Interview Questions**

1. How does Docker ensure container isolation on a Linux system?

2. What are cgroups and namespaces, and how do they work in Docker?

3. How do you reduce the size of a Docker image significantly?

4. What security concerns should you consider when building Docker images?

5. How do you scan Docker images for vulnerabilities?

6. What is Docker Content Trust (DCT), and how does it enhance security?

7. How does Docker handle image layering under the hood?

8. What is a Union File System in Docker, and why is it important?

9. How do you implement image immutability in CI/CD pipelines?

10. How do you handle Docker secrets in production?

11. What is the difference between secrets and environment variables?

12. How can you pass secrets securely to a container without hardcoding?

13. Explain how to use docker save and docker load.

14. How do you ensure containers are ephemeral and stateless?

15. What are the best practices to follow while writing Dockerfiles?

16. What are the common anti-patterns in Dockerfile usage?

17. How do you layer caching in a Dockerfile for optimal performance?

18. Explain the build cache mechanism in Docker and how it works.

19. What is the difference between Docker BuildKit and traditional build?

20. How do you enable Docker BuildKit?

21. What is the impact of using latest tag in production deployments?

22. How do you version Docker images properly in production environments?

23. What tools can you use for Docker image analysis and auditing?

24. How do you monitor Docker containers in a production environment?

25. How do you set up logging for Docker containers in a centralized way?

26. What’s the difference between logging drivers: json-file, syslog, fluentd?

27. How do you troubleshoot a container that runs fine locally but not in prod?

28. How can you mount a configuration file dynamically inside a container?

29. What are init containers and how are they used?

30. What is the difference between ENTRYPOINT as exec vs shell form?

31. Why is it not recommended to run containers as root?

32. How can you configure non-root users inside containers?

33. How do you handle large file transfers between containers and host?

34. How do you analyze Docker container performance bottlenecks?

35. How can you isolate container network traffic for compliance?

36. How can you limit container IO bandwidth?

37. What is user namespace remapping in Docker?

38. How do you run Docker containers in a fully isolated environment?

39. What is the best way to back up persistent data in Docker?

40. How do you enforce resource quotas across multiple containers?

41. How do you update secrets/configs without restarting containers?

42. How do you prevent container escape vulnerabilities?

43. How do you avoid race conditions during container startup?

44. What are zombie containers and how do you clean them up?

45. How do you run Docker containers on Kubernetes vs standalone Docker?

46. How do you simulate network failures in containers for testing?

47. What are the common issues with Docker and IPv6?

48. How do you enable IPv6 support for Docker containers?

49. How do you configure custom DNS for Docker containers?

50. How do you restrict container access to certain domains or IPs?

51. How do you automate container health and readiness checks?

52. How do you roll back a faulty image deployed to production?

53. What is Docker’s image signing mechanism?

54. How do you restrict container image sources to trusted registries?

55. What are container runtime interfaces (CRI)? How does Docker interact with them?

56. How does Docker manage shared layers across images?

57. What’s the purpose of squash option while building images?

58. What are tmpfs mounts and their benefits in Docker?

59. How can you isolate containers using AppArmor/SELinux profiles?

60. How do you perform fine-grained container access controls?

61. How do you implement rate limiting for containers?

62. How do you manage image lifecycles in an enterprise setup?

63. What’s the role of container image scanning in DevSecOps?

64. How do you detect and mitigate CVEs in base images?

65. How do you rotate container credentials without downtime?

66. What’s the best way to share a Docker volume across multiple services?

67. How do you secure Docker’s remote API?

68. What are the implications of Docker socket exposure?

69. How do you manage Docker Swarm secrets and configs?

70. How do you handle rollbacks in Docker Swarm vs Kubernetes?

71. What are the challenges with multi-arch builds and how do you solve them?

72. How can you build images for ARM and x86 architectures?

73. What is a layered filesystem in Docker and its pros/cons?

74. How does Docker handle file system changes during container runtime?

75. How can you trace file changes inside a running container?

76. How can you export container logs to external systems?

77. What is Docker rootless mode and how do you enable it?

78. How do you implement backup and restore for containerized databases?

79. What are best practices for ephemeral vs persistent volumes?

80. How do you avoid storage bloat in containerized environments?

81. What’s the impact of Docker garbage collection?

82. How do you set up a local Docker registry with authentication?

83. How do you mirror Docker Hub images for offline access?

84. What’s the role of overlay networks in Docker Swarm?

85. How do you perform zero-downtime deployments using Docker?

86. What is blue-green deployment using Docker?

87. How do you load test containers under real-world conditions?

88. What’s the difference between Docker pause and stop?

89. What are health checks and how do they help in orchestration?

90. How do you design a CI/CD pipeline with Docker at the core?

91. How do you promote Docker images across environments (dev → prod)?

92. What is the benefit of using .dockerignore in builds?

93. How do you run Docker-in-Docker (dind)? What are the risks?

94. How do you isolate container workloads on shared infrastructure?

95. How do you containerize an app that uses system-level dependencies?

96. What are the limitations of Docker for production-scale apps?

97. How does Docker interact with kernel-level features?

98. How can you optimize image rebuilds in large monorepos?

99. What’s the difference between daemonless and traditional Docker runtimes?

100. What tools complement Docker for production-grade deployments?