Docker & Kubernetes Mock Interview Questions and Answers (50)

Docker (25 Questions)

- Q: What is Docker?

A: Docker is a containerization platform that packages applications and their dependencies into lightweight, portable containers, ensuring consistent execution across environments.

- Q: What are Docker images and containers?

A: A Docker image is a blueprint (read-only template) used to create containers. A container is a running instance of an image.

- Q: Difference between Virtual Machine and Docker container?

A: VMs run a full OS with hypervisors; containers share the host OS kernel, making them faster and lightweight.

- Q: What is Docker Hub?

A: Docker Hub is a cloud-based registry for sharing and managing Docker images.

- Q: What is a Dockerfile?

A: A Dockerfile is a script containing instructions to build a Docker image automatically.

- Q: Difference between COPY and ADD in Dockerfile?

A: Both copy files, but ADD can also extract archives and fetch files from URLs, while COPY only copies local files.

- Q: What is a Docker volume?

A: Volumes are used to persist data generated by containers outside the container lifecycle.

- Q: What are Docker namespaces?

A: Namespaces provide isolation for processes, networking, and file systems in containers.

- Q: What is Docker Compose?

A: Docker Compose is a tool to define and run multi-container applications using a YAML file.

- Q: What is the difference between CMD and ENTRYPOINT?

A: CMD provides default arguments for a container, while ENTRYPOINT defines the main executable.

- Q: What is Docker Swarm?

A: Docker Swarm is Docker's native clustering tool for managing multiple nodes and scaling services.

- Q: How do you check running containers?

A: By using: docker ps

- Q: Difference between docker stop and docker kill?

A: docker stop gracefully stops a container (sends SIGTERM), while docker kill forcefully stops it (SIGKILL).

- Q: What is multi-stage build in Docker?

A: It allows using multiple FROM statements in a Dockerfile to optimize image size.

- Q: What are Docker networks?

A: They allow communication between containers. Types include bridge, host, and overlay.

- Q: Can a container run multiple processes?

A: Ideally one process per container, but multiple processes can run using process managers like Supervisor.

- Q: How do you remove all stopped containers?

A: docker container prune

- Q: What is the difference between Docker and Podman?

A: Docker requires a daemon, Podman is daemonless and rootless.

- Q: What is the purpose of docker exec?

A: It executes a command inside a running container.

- Q: What is the difference between docker run and docker start?

A: docker run creates and starts a new container, docker start restarts an existing stopped container.

- Q: What is a base image?

A: A minimal image used as the starting point for building custom images (e.g., Ubuntu, Alpine).

- Q: How do you secure Docker containers?

A: Use minimal base images, run as non-root, enable resource limits, and scan images for vulnerabilities.

- Q: What is Docker overlay network?

A: A multi-host network that allows containers to communicate across different hosts.

- Q: How do you reduce Docker image size?

A: Use smaller base images (Alpine), multi-stage builds, and clean up temporary files in Dockerfile.

- Q: How do you monitor Docker containers?

A: Using commands like docker stats, logging drivers, or monitoring tools like Prometheus + Grafana.

Kubernetes (25 Questions)

- Q: What is Kubernetes?

A: Kubernetes is an open-source container orchestration platform for automating deployment, scaling, and management of containerized applications.

- Q: What is a Pod in Kubernetes?

A: A Pod is the smallest deployable unit in Kubernetes that contains one or more containers.

- Q: What is the difference between ReplicaSet and Deployment?

A: ReplicaSet ensures a defined number of pod replicas, while Deployment provides rolling updates and rollback features.

- Q: What are Kubernetes Services?

A: Services expose Pods to the network. Types: ClusterIP, NodePort, LoadBalancer, and ExternalName.

- Q: What is the role of kube-apiserver?

A: It is the entry point for all Kubernetes REST requests and communication between cluster components.

- Q: What is ETCD in Kubernetes?

A: ETCD is a distributed key-value store that holds cluster state and configuration data.

- Q: What is the role of kube-scheduler?

A: It assigns pods to nodes based on resource availability and constraints.

- Q: What is a Namespace in Kubernetes?

A: Namespaces are logical partitions for organizing cluster resources.

- Q: What are ConfigMaps and Secrets?

A: ConfigMaps store non-sensitive configuration data, while Secrets store sensitive data (passwords, tokens).

- Q: Difference between StatefulSet and Deployment?

A: Deployment manages stateless apps, StatefulSet manages stateful apps requiring stable network identity and persistent storage.

- Q: What is a DaemonSet?

A: Ensures that a copy of a Pod runs on all (or selected) nodes in the cluster.

- Q: What is a Job and CronJob in Kubernetes?

A: Job runs tasks to completion, CronJob schedules tasks periodically.

- Q: What is the difference between NodePort and LoadBalancer service?

A: NodePort exposes service on a node's port, LoadBalancer uses external cloud load balancers.

- Q: What is kube-proxy?

A: A network proxy that manages service networking and load balancing across pods.

- Q: What is a Persistent Volume (PV) and Persistent Volume Claim (PVC)?

A: PV is cluster-level storage, PVC is a request for storage by a pod.

- Q: How do you perform rolling updates in Kubernetes?

A: Using kubectl rollout commands on Deployments.

- Q: What is the difference between Horizontal Pod Autoscaler and Vertical Pod Autoscaler?

A: HPA scales pods horizontally (more replicas), VPA scales resource limits vertically (CPU/memory).

- Q: What is a sidecar container?

A: An additional container in a Pod that supports the main container (e.g., logging, proxy).

- Q: What is taint and toleration in Kubernetes?

A: Taints prevent pods from scheduling on nodes unless they tolerate it.

- Q: What is a Kubernetes Ingress?

A: Ingress manages external HTTP/HTTPS access to services using rules.

- Q: What is the role of kubelet?

A: An agent running on each node to ensure containers are running as expected.

- Q: What is Helm in Kubernetes?

A: Helm is a package manager for Kubernetes applications using Helm charts.

- Q: How do you troubleshoot a failing Pod?

A: Use commands like: kubectl describe pod <pod-name>, kubectl logs <pod-name>

- Q: How does Kubernetes handle self-healing?

A: Kubernetes automatically restarts failed pods, reschedules them, and replaces unresponsive nodes.

- Q: What security best practices should you follow in Kubernetes?

A: Use RBAC, network policies, secrets management, restrict root access, and enable pod security policies.