**🔹 Jenkins — Top 20 Q&A**

**1. What is Jenkins?**

👉 Jenkins is an **open-source CI/CD automation server** used to build, test, and deploy software pipelines.

**2. Difference between CI and CD?**

👉 CI = Continuous Integration (merge + build + test).  
👉 CD = Continuous Deployment/Delivery (automatic release to production).

**3. Jenkins pipeline types?**

* **Declarative** (structured, recommended).
* **Scripted** (flexible, Groovy-based).

**4. Sample declarative pipeline?**

pipeline {

agent any

stages {

stage('Build') { steps { sh 'mvn clean install' } }

stage('Test') { steps { sh 'mvn test' } }

stage('Deploy') { steps { sh './deploy.sh' } }

}

}

**5. How do you secure credentials in Jenkins?**

👉 Use **Credentials Plugin** → inject as environment variables → never hardcode.

**6. How to schedule builds in Jenkins?**

👉 Use **CRON syntax** in job configuration (H/15 \* \* \* \* → every 15 mins).

**7. What is a Jenkins agent?**

👉 A worker node that runs jobs. Master schedules → Agents execute.

**8. How do you scale Jenkins?**

* Master-agent architecture.
* Use containers (K8s plugin).
* Use pipeline libraries.

**9. Difference between freestyle job and pipeline job?**

👉 Freestyle = simple GUI config.  
👉 Pipeline = script-based, scalable.

**10. What is Jenkinsfile?**

👉 File that stores pipeline-as-code inside repo.

**11. How do you handle parallel execution?**

👉 In Jenkins pipeline:

parallel {

stage('Unit Tests') { steps { sh 'mvn test' } }

stage('Lint') { steps { sh 'eslint .' } }

}

**12. How do you trigger jobs automatically?**

* SCM Webhooks (GitHub/GitLab).
* Poll SCM.
* Upstream/downstream jobs.

**13. How do you store artifacts in Jenkins?**

👉 Use **archiveArtifacts** or publish to Nexus/Artifactory.

**14. How do you handle rollback in Jenkins?**

👉 Store previous artifacts → redeploy previous version → pipeline rollback stage.

**15. How to integrate Jenkins with Docker?**

👉 Jenkins builds Docker image → pushes to registry → deploys container.

**16. How do you integrate Jenkins with Kubernetes?**

👉 Use **Kubernetes plugin** → launch ephemeral build agents as pods.

**17. How do you integrate Jenkins with Terraform?**

👉 Jenkins pipeline runs terraform plan → terraform apply.

**18. How do you secure Jenkins server?**

* HTTPS, RBAC, restrict anonymous access.
* Secrets in Vault.
* Keep plugins updated.

**19. How to monitor Jenkins?**

👉 Use Prometheus plugin, Grafana dashboards, logs.

**20. Real-world: Pipeline failed, what do you do?**

👉 Check logs → identify failing stage → rerun failed stage → fix issue → re-trigger build.

**🔹 Git — Top 20 Q&A**

**1. What is Git?**

👉 Git is a **distributed version control system** used to track code changes.

**2. Difference between Git and GitHub?**

👉 Git = VCS tool. GitHub = hosting service for Git repos.

**3. How do you resolve merge conflicts?**

👉 Manually edit files → git add → git commit.

**4. What is the difference between merge and rebase?**

* **Merge** → Creates a merge commit.
* **Rebase** → Rewrites history (cleaner, linear).

**5. What is Git stash?**

👉 Saves uncommitted changes temporarily.  
git stash save "msg" → git stash pop.

**6. What is Git cherry-pick?**

👉 Apply a specific commit from one branch to another.  
git cherry-pick <commit-id>.

**7. What is the difference between git pull and git fetch?**

* **pull** = fetch + merge.
* **fetch** = download changes, don’t merge.

**8. How do you undo last commit?**

👉 git reset --soft HEAD~1 (keep changes).  
👉 git reset --hard HEAD~1 (discard changes).

**9. Git branching strategies?**

* GitFlow.
* Trunk-based development.
* Feature branching.

**10. What is .gitignore used for?**

👉 Excludes files/folders (logs, build artifacts).

**11. What is Git detached HEAD?**

👉 State where HEAD points to a commit, not a branch.

**12. How do you squash commits?**

👉 git rebase -i HEAD~3 → squash.

**13. Difference between git clone and git fork?**

👉 Clone = copy repo locally. Fork = copy repo to your account.

**14. What is Git submodule?**

👉 Repo inside another repo.  
git submodule add <url>.

**15. How do you sign commits?**

👉 git commit -S -m "msg" → uses GPG keys.

**16. What is Git hook?**

👉 Script triggered by Git events (pre-commit, post-merge).

**17. How do you revert a commit?**

👉 git revert <commit-id> (creates new commit undoing changes).

**18. How do you view commit history?**

👉 git log --oneline --graph --decorate.

**19. How do you tag a commit?**

👉 git tag v1.0 <commit-id> → git push origin v1.0.

**20. Real-world: You accidentally pushed secrets?**

👉 Rotate secrets immediately → remove from repo (git filter-branch or BFG) → force push.

**🔹 Terraform — Top 20 Q&A**

**1. What is Terraform?**

👉 Terraform is an **IaC tool** that provisions infra using declarative configs.

**2. Difference between Terraform and Ansible?**

👉 Terraform → infra provisioning.  
👉 Ansible → config management.

**3. What is Terraform state file?**

👉 .tfstate keeps track of real infra.

**4. How do you secure Terraform state?**

👉 Store in **remote backend** (S3 + DynamoDB lock, Consul, GCS).

**5. Terraform plan vs apply?**

👉 plan → preview changes.  
👉 apply → execute changes.

**6. What is Terraform provider?**

👉 Plugin to interact with cloud resources (AWS, Azure, GCP).

**7. How do you reuse Terraform code?**

👉 Modules (local or registry).

**8. What is Terraform workspace?**

👉 Used for multiple environments (dev, staging, prod).

**9. How do you pass variables in Terraform?**

👉 terraform.tfvars or -var "key=value".

**10. Difference between terraform import and terraform taint?**

👉 import → bring existing resource under Terraform control.  
👉 taint → mark for recreation in next apply.

**11. How do you destroy infra?**

👉 terraform destroy.

**12. How do you handle secrets in Terraform?**

👉 Use Vault or SSM Parameter Store → never hardcode.

**13. What is Terraform drift?**

👉 When real infra != state file → fix by running terraform plan.

**14. What is terraform init?**

👉 Initializes working directory, downloads providers & modules.

**15. How do you manage multiple teams using Terraform?**

👉 Remote backend + state locking + workspaces.

**16. What are data sources in Terraform?**

👉 Read-only information from providers (e.g., aws\_ami).

**17. What are output variables?**

👉 Share values across modules (e.g., DB endpoint).

**18. What is depends\_on in Terraform?**

👉 Explicitly define resource dependency.

**19. How to test Terraform code?**

👉 Use terraform validate + terraform plan + terratest.

**20. Real-world: Your infra is drifting?**

👉 Run terraform refresh → compare → fix manually or via code.

**🔹 Kubernetes — Top 20 Q&A**

**1. What is Kubernetes?**

👉 Container orchestration platform for deployment, scaling, and management.

**2. Difference between Docker Swarm and Kubernetes?**

👉 Swarm = simple, less features.  
👉 K8s = advanced, industry standard.

**3. Core components of K8s?**

* API Server.
* etcd (state).
* Scheduler.
* Controller manager.
* Kubelet.
* Kube-proxy.

**4. What is a Pod?**

👉 Smallest deployable unit, wrapper around one or more containers.

**5. What is a Deployment?**

👉 Manages replica sets and rolling updates.

**6. What is a Service in K8s?**

👉 Exposes pods to external/internal traffic (ClusterIP, NodePort, LoadBalancer).

**7. How do you debug a pod?**

👉 kubectl describe pod <pod>  
👉 kubectl logs <pod>  
👉 kubectl exec -it <pod> -- sh

**8. What is ConfigMap vs Secret?**

👉 ConfigMap = non-sensitive configs.  
👉 Secret = base64-encoded sensitive data.

**9. Difference between StatefulSet and Deployment?**

👉 StatefulSet = stable network identity, persistent storage.  
👉 Deployment = stateless apps.

**10. What is a DaemonSet?**

👉 Ensures a pod runs on every node (e.g., logging agent).

**11. What is Ingress?**

👉 API object to expose HTTP/HTTPS routes to services.

**12. How do you do rolling updates in K8s?**

👉 kubectl rollout restart deployment <name>.

**13. How do you rollback in K8s?**

👉 kubectl rollout undo deployment <name>.

**14. What is a kubeconfig file?**

👉 Stores cluster access configs (API server, tokens, certs).

**15. How do you scale pods in K8s?**

👉 kubectl scale deployment myapp --replicas=5.

**16. What is HPA (Horizontal Pod Autoscaler)?**

👉 Auto-scales pods based on CPU/memory/custom metrics.

**17. What is a PV and PVC?**

👉 PV = Persistent Volume (storage resource).  
👉 PVC = Persistent Volume Claim (request for storage).

**18. How do you monitor K8s?**

👉 Prometheus + Grafana + kubectl metrics.

**19. How does K8s achieve self-healing?**

👉 Restarts crashed pods, reschedules failed ones.

**20. Real-world: Pod stuck in Pending?**

👉 Check kubectl describe pod → node resources → PVC binding.

**🔹 Prometheus & Grafana — Top 20 Q&A**

**1. What is Prometheus?**

👉 Open-source monitoring & alerting system.

**2. What is Grafana?**

👉 Visualization & analytics tool that integrates with Prometheus.

**3. How does Prometheus collect metrics?**

👉 Pull-based model → scrapes HTTP endpoints (/metrics).

**4. What is a Prometheus exporter?**

👉 Component that exposes metrics (Node exporter, cAdvisor).

**5. What is a PromQL query?**

👉 Prometheus Query Language for metrics (e.g., rate(http\_requests\_total[5m])).

**6. What is an Alertmanager?**

👉 Handles alerts from Prometheus, routes to email/Slack/PagerDuty.

**7. How do you set alerts in Prometheus?**

👉 Define alerting rules in YAML.  
Example:

groups:

- name: example

rules:

- alert: HighCPUUsage

expr: node\_cpu\_seconds\_total > 90

for: 5m

**8. What is the default storage in Prometheus?**

👉 Time-series DB (TSDB) stored locally.

**9. How do you scale Prometheus?**

👉 Use federation, sharding, Thanos, Cortex.

**10. What is Grafana dashboard?**

👉 Visual representation of metrics → panels (graphs, tables).

**11. How do you secure Grafana?**

👉 Enable auth, RBAC, integrate with LDAP/SSO.

**12. How do you persist Prometheus data?**

👉 Mount persistent volumes.

**13. What’s the difference between push vs pull in monitoring?**

👉 Prometheus = pull.  
👉 Use Pushgateway for short-lived jobs.

**14. What are recording rules in Prometheus?**

👉 Precompute queries for efficiency.

**15. How to integrate Prometheus with Kubernetes?**

👉 Use kube-state-metrics + service discovery.

**16. How to send Grafana alerts?**

👉 Configure alert channels (Slack, Teams, Email).

**17. What’s the difference between Prometheus and Grafana?**

👉 Prometheus = metrics collection & alerting.  
👉 Grafana = visualization.

**18. How do you optimize Grafana dashboards?**

👉 Use templating, variables, efficient queries.

**19. What is the retention period in Prometheus?**

👉 Default = 15 days, configurable with --storage.tsdb.retention.time.

**20. Real-world: Prometheus isn’t scraping targets?**

👉 Check targets page → service discovery → firewall/network → exporter running.

**🔹 Ansible — Top 20 Interview Q&A**

**1. What is Ansible?**

👉 Ansible is an **open-source configuration management, provisioning, and automation tool** that uses **SSH** and **YAML playbooks**.

**2. How is Ansible different from Terraform?**

* **Terraform** → Infra provisioning (create servers, networks, storage).
* **Ansible** → Config management (install packages, configure apps, deploy code).

**3. What language does Ansible use for playbooks?**

👉 **YAML (Yet Another Markup Language).**

**4. What is an Ansible inventory file?**

👉 File listing target hosts where playbooks run.  
Example:

[web]

192.168.1.10

[db]

192.168.1.20

**5. What is the difference between static and dynamic inventory?**

* **Static** → Defined in .ini/.yaml file.
* **Dynamic** → Generated via cloud plugins (AWS, GCP, Azure).

**6. What is an Ansible module?**

👉 A unit of work (package, copy, service, user).  
Example:

- name: Install nginx

apt:

name: nginx

state: present

**7. What are Ansible roles?**

👉 A way to **organize playbooks** into reusable components (tasks, handlers, templates, vars).

**8. How do you make playbooks idempotent?**

👉 Ansible ensures running the same playbook multiple times gives the same result (e.g., state: present).

**9. How do you run an ad-hoc Ansible command?**

👉 Example:

ansible web -m ping

ansible all -m shell -a "uptime"

**10. What is the difference between ansible and ansible-playbook?**

* **ansible** → Ad-hoc command execution.
* **ansible-playbook** → Runs structured YAML playbooks.

**11. What are Ansible facts?**

👉 Auto-discovered system variables about hosts (IP, OS, CPU).  
👉 Use: ansible all -m setup.

**12. What is Ansible Galaxy?**

👉 A repository of **pre-built roles** and playbooks shared by community.  
Command: ansible-galaxy install <role>.

**13. How do you manage secrets in Ansible?**

👉 Use **Ansible Vault**:

ansible-vault encrypt secrets.yml

ansible-playbook site.yml --ask-vault-pass

**14. How do you test Ansible playbooks?**

* ansible-playbook --check (dry run).
* Molecule framework for testing.

**15. What is a handler in Ansible?**

👉 A task triggered only if notified.  
Example:

- name: Restart nginx

service:

name: nginx

state: restarted

listen: "restart nginx"

**16. How do you run only specific tags in Ansible?**

👉 Tag tasks:

- name: Install nginx

apt: { name: nginx, state: present }

tags: install

Run with: ansible-playbook site.yml --tags install.

**17. What are conditionals in Ansible?**

👉 Example:

- name: Install nginx

apt: name=nginx state=present

when: ansible\_os\_family == "Debian"

**18. How do you use loops in Ansible?**

- name: Install packages

apt:

name: "{{ item }}"

state: present

loop:

- nginx

- git

- curl

**19. How do you improve performance in Ansible?**

* Use **--forks** for parallelism.
* Use **async** tasks.
* Limit hosts (-l host1).
* Use **caching** for facts.

**20. Real-world: Your playbook fails halfway. What do you do?**

👉 Use --start-at-task to resume from failed step → debug with -vvv → fix issue → re-run.