Jenkins Interview questions

===========================

1. what is Continuous Integration?

Continuous integration is a process

get scm 🡪 test 🡪 static code analysis🡪 build 🡪

nexus 🡪 build 🡪 failure send notification (CI)

1. what type of jobs have you configured in jenkins?

* Pipeline
* Multi-configure
* Freestyle Job
* Maven Using Plugin

1. what are the types of jobs are available in jenkins?

* Freestyle project.
* Maven project.
* Pipeline.
* Multibranch pipeline.
* Multi-configuration project.
* Github organization.

1. what is pipeline ?

Pipeline is group of plugins which are used to implement CI/CD goals. By using Jenkins file we can implement the small to complex structure of CI/CD process.

1. what is difference b/w freestyle and pipeline?

* Freestyle Job is used when everything is given according to plugin. Complex structure is hard to implement.
* Pipeline job is used when structure of CI/CD complex such as read data from PoM.xml.We Insert Data via executing job.

1. what is declarative pipeline?

A declarative pipeline is a new way to write pipeline.

It’s take less no.of. executor to write a code.

1. what is difference between declarative pipeline and script based pipeline?

* Executor
* Release
* Stage

1. write the pipeline syntax?

*Script Based:*

Schedules the steps contained within the block to run by adding an item to the Jenkins queue. As soon as an executor is free on a node, the steps will run.

node {

stage('Build') {

//

}

stage('Test') {

//

}

stage('Deploy') {

//

}

}

*Declarative Based:*

pipeline {

agent {

label 'linux-machine'

}

stages {

stage('build step') {

Steps{

Sh ‘echo Job run’

}

}

}

1. what is master/slave architecture ? what is the use of master/slave ?

* Master-slave architectures are used to help stabilize a system
* Master Jenkins can’t execute lot’s of job simultaneously. So, we need slave to transfer buildload.

1. How many ways we can connect the slaves?

* Launch agent by conn. It to the controller
* Launch agent via conn. Of command on the controller
* Launch agent via SSH

1. how to run each stage in different slaves

stage{

agent {

}

}

1. How many ways we can provide security for your jenkins server ?

Configure Global Security 🡪 LDAP database / Matrix based security

1. what is sonarqube ? have you configure ? How you configure ?

It’s code quality maintenance tool./

### Custom QualityGate

1. what type of artifactory repository tool have you used?

**nexus/jfrog/s3 oss – pro**

1. what is the use of artifactory tools?

Artifactory:**Artifactory is a repository manager**. · The task of a repository manager is to store artifacts, the result of a build, and make them available to other projects.

Artifacts: Artifact in maven is **a resource generated by a maven project**. Each maven project can have exactly one artifact like a jar, war, ear , etc.

e.g: Nexus/Jfrog

1. How you declare a variables in pipeline?

Variable=value

1. what is DSL language? domain spcific language?

 a programming language or specification language dedicated to a particular problem domain, [Groovy]

1. what is upstream/downstream projects? what is the use of it? which scenario you configure?

There Is a dependency of each other.

1. If a have 10 repositories in github how many jobs you can configure ?

We can achive this task using Pipeline. But we can do that due to packages and deploymets. + Dependent Repository

1. How to take backup my jenkins? thin backup plugin

Whole Jenkin folder Backup.

Var/lib/Jenkins 🡪 copy the folder and keep it in the another path

1. what is jenkins Home directory?

.jenkins

1. how to deleted old builds automatically?

Description 🡪 Discard old builds

1. how to configure multiple environment deployment?

Multicofiguration job

1. how to maintain aws credentials in Jenkins

AWS Plugin & config

ss

1. what are the plugins have you used in your project?
2. Seed jobs in Jenkins

Shared Librabry: You can have the seed job **automatically create build jobs for feature branches**. You can have your release jobs build docker containers and push.

CI/CD & Jenkins Basics –

1. What is Continuous Integration (CI) and Continuous Deployment/Delivery (CD) ?

1.CI: Jenkins detects a push to GitHub and runs unit tests.

2.CD: Jenkins builds a Docker image, pushes it to AWS ECR, and waits for manual approval.

3.Continuous Deployment: If approved, Jenkins deploys the container to AWS ECS automatically.

1. What types of jobs have you configured in Jenkins ?

Freestyle & pipeline

1. What is a Jenkins pipeline, freestyle job and a pipeline job ?

A Jenkins pipeline is a series of steps that automate the build, test, and deployment process in a CI/CD workflow. It is defined using code (Jenkinsfile), making it scalable, maintainable, and version-controlled.

A black table with white text

AI-generated content may be incorrect.

Jenkins Pipelines -

1. What is Continuous Integration (CI) and Continuous Deployment/Delivery (CD) ?

A screenshot of a black and white screen

AI-generated content may be incorrect.

1. How do you declare variables in a Jenkins pipeline ?

environment {

APP\_NAME = "MyApp"

BUILD\_VERSION = "1.0.0" }

1. How do you implement parallel execution in a Jenkins pipeline ?

Parallel execution in Jenkins allows different stages or tasks to run simultaneously, reducing build time and improving efficiency.

A black screen with white text

AI-generated content may be incorrect.

Jenkins Architecture & Agents -

1. What is the master/slave (controller/agent) architecture in Jenkins ? What is its purpose ?

Jenkins follows a Master-Slave (Controller-Agent) architecture to distribute workload across multiple machines, improving performance and scalability.

A black and white text on a black background

AI-generated content may be incorrect.

1. How many ways can you connect slave nodes in Jenkins ?

For on-premise Jenkins: Use SSH (Linux) or JNLP (Windows) agents.

For containerized environments: Use Docker or Kubernetes agents.

For cloud-based scaling: Use AWS EC2, Azure, or GCP agents.

A black screen with white text

AI-generated content may be incorrect.

1. How do you configure different stages to run on different slave nodes ?

agent { label 'node1' }

A screenshot of a computer

AI-generated content may be incorrect.

A computer screen shot of white text

AI-generated content may be incorrect.

Security & Access Control

1. How do you secure Jenkins (authentication, authorization, and secret management) ?

A screenshot of a computer

AI-generated content may be incorrect.

1. What is Matrix Authorization Strategy, and how does it help in Jenkins security ?

Manual Backup Approach –

* /var/lib/jenkins/

Automated Backup Approaches –

* ThinBackup

Advanced Jenkins Pipelines & Automation –

1. What is Matrix Authorization Strategy, and how does it help in Jenkins security ?

Flexible & Scalable: Works for small & enterprise Jenkins setups

Enhanced Security: Restricts unauthorized job execution & modifications

Supports Enterprise Authentication: Works with LDAP, SAML, GitHub OAuth

Fine-Grained Control: Prevents accidental deletions & security risks

1. How do you implement error handling and retries in a Jenkins pipeline ?

Wrap risky steps in try-catch for controlled error handling

Use catchError when a failure shouldn’t stop the entire pipeline

Always log errors using echo "Error: ${e.message}"

Use finally block for cleanup tasks

Optimization & Troubleshooting -

1. How do you optimize Jenkins performance and reduce build times ?

Optimizing Jenkins Performance & Reducing Build Times

Use Distributed Builds

- Configure Jenkins agents to distribute workloads.

- Assign jobs to specific nodes using \*\*labels\*\*.

Optimize Build Execution

- Use Parallel Stages in pipelines to speed up builds.

- Limit build history by enabling \*\*"Discard Old Builds".

Reduce Job Load & Cleanup

- Use "Throttle Builds" Plugin" to limit concurrent builds.

- Enable \*\*clean workspace\*\* before/after builds (`cleanWs()`).

Improve Jenkins Server Performance

- Allocate \*\*more CPU & RAM\*\* to Jenkins master.

- Use an \*\*external database (MySQL/PostgreSQL)\*\* instead of the default one.

- Tune \*\*Java Garbage Collection\*\* (`-XX:+UseG1GC`).

Optimize Plugins & Caching

- Remove \*\*unused plugins\*\* to reduce overhead.

- Use \*\*build caching\*\* (Maven, Docker layer caching) to avoid redundant tasks.

1. A Jenkins job is failing, but the logs do not show any errors. How do you debug it ?

tail -f /var/log/jenkins/jenkins.log

tail -f /var/log/jenkins/jenkins\_slave.log

1. How do you configure automatic workspace and old build cleanup in Jenkins ?

options { cleanWs() }

1. What strategies can you use to optimize large-scale Jenkins setups with many jobs ?

Set up Jenkins agents (slaves) to distribute workloads.

Use labels to assign jobs to specific nodes based on resources.

Dynamically provision agents using AWS EC2, Kubernetes, or Docker.

1. How would you troubleshoot a slow Jenkins server with high build queue times ?

Delete unused jobs, old builds, and workspaces (cleanWs()).

Use the "Workspace Cleanup Plugin" to automate cleanup.

Update Jenkins & Plugins to the latest stable versions.

High Availability & Scalability –

1. How do you implement Jenkins High Availability (HA) ? Explain different HA architectures.

Use backup & restore (ThinBackup, S3 snapshots).

Offload builds to Jenkins agents to reduce master load.

Monitor Jenkins with Prometheus, Grafana, CloudWatch.

Use DNS-based failover (Route 53, Cloudflare) for master recovery.

Jenkins Backup, Disaster Recovery & Maintenance -

1. How do you take a backup of Jenkins using the ThinBackup plugin ?

Install ThinBackup from Plugin Manager.

Configure backup path & schedule.

Backup manually or automatically.

Restore via Manage Jenkins → ThinBackup → Restore.

1. A Jenkins job is failing, but the logs do not show any errors. How do you debug it ?

Go to Job → Configure → Build Triggers

Enable Discard Old Builds

Set:

Max # of builds to keep (e.g., 20)

Max # of days to keep builds (e.g., 10)

1. How do you manage multiple environment deployments in Jenkins ?

parameters {

choice(name: 'ENV', choices: ['dev', 'staging', 'prod'], description: 'Select Environment')

}

1. How do you use Jenkins in a microservices-based architecture ?

Use Declarative Pipelines – Cleaner & easier to maintain.

Use Shared Libraries – Reuse common functions across pipelines.

Parameterize Pipelines – Avoid hardcoding; support multiple environments.

Implement Error Handling & Retries – Use retry() and try-catch.

Optimize Performance – Use parallel execution & cleanWs().

Secure Credentials – Store secrets in Jenkins Credentials Plugin.

Enable Monitoring & Notifications – Set build retention, Slack/email alerts.

1. How do you handle concurrent builds and avoid conflicts in shared resources ?

Use "Throttle Concurrent Builds" Plugin – Limit builds per job/category.

Use "Lockable Resources" Plugin – Lock shared resources (DB, servers).

Disable Concurrent Builds – Use disableConcurrentBuilds() in pipelines.

Use Unique Workspaces – Avoid conflicts with ws('@unique').