⚇ What is Jenkins?

- Jenkins is an open-source automation server used for Continuous Integration/Continuous Delivery (CI/CD). It helps automate the build, testing, and deployment process.

- **Example:** Jenkins can trigger automatic builds when new code is pushed to a Git repository.

⚇ How does Jenkins work?

- Jenkins automates parts of software development by using jobs to pull code from repositories, build, and deploy it.

- **Example:** A Jenkins job pulls code from GitHub, runs tests, and deploys the app.

⚇ What are the advantages of using Jenkins?

- Automates repetitive tasks, integrates with various tools, supports plugins, and is easy to configure for continuous integration.

- **Example:** Jenkins automates daily builds and deployments to production.

⚇ Explain the architecture of Jenkins.

- Jenkins follows a master-slave architecture where the master controls the build process, and slaves execute jobs on different environments.

- **Example:** A master Jenkins server delegates tasks to different agent nodes (Windows/Linux).

▣ What is Continuous Integration in Jenkins?

- Continuous Integration is a practice where developers integrate code into a shared repository frequently. Jenkins automates this process.

- **Example:** Jenkins builds the project and runs tests whenever code is pushed.

▣ What are Jenkins Pipelines?

- Pipelines are a series of steps to build, test, and deploy code in Jenkins. They are defined in a Jenkinsfile.

- **Example:** A pipeline can have stages like "Build," "Test," and "Deploy."

▣ How do you install Jenkins?

- Jenkins can be installed using a WAR file or native packages (Linux, Windows, Mac).

- **Example:** Install Jenkins on Ubuntu with sudo apt-get install jenkins.

▣ What are the system requirements for installing Jenkins?

- Jenkins requires at least 256MB RAM, 1GB disk space, and Java 8 or higher.

- **Example:** Java must be installed using sudo apt install openjdk-8-jdk.

▣ How do you configure Jenkins?

- Configuration involves setting up credentials, agents, security, and plugins.

- **Example:** Set Git credentials under "Manage Jenkins" > "Manage Credentials."

▢ What is a Jenkins job?

- A Jenkins job is a task or set of tasks configured to be run within Jenkins.

- **Example:** A "Freestyle Project" to build a Java application.

▢ How do you create a Jenkins job?

- In Jenkins dashboard, click "New Item," select the job type, and configure the tasks.

- **Example:** Create a Freestyle job to compile code using Maven.

▢ Explain the different types of Jenkins jobs.

- Freestyle, Pipeline, Multibranch Pipeline, and Maven jobs.

- **Example:** A Maven job builds a project using the Maven tool.

▢ How do you trigger a Jenkins job automatically?

- Jobs can be triggered by SCM changes or scheduled with cron expressions.

- **Example:** Poll Git every 5 minutes for changes using H/5 * * * *.

❓ How do you set up a Jenkins job to build periodically?

- Use cron syntax in the "Build Triggers" section.
- **Example:** Build every day at midnight using 0 0 * * *.

❓ What is a Jenkinsfile?

- A Jenkinsfile is a text file that contains the pipeline code.
- **Example:** A declarative Jenkinsfile with stages for build, test, and deploy.

❓ How do you create a Jenkinsfile?

- Create a Jenkinsfile in the root of your project repository.
- **Example:**

Copy code

```
pipeline {
  stages {
    stage('Build') {
      steps {
        echo 'Building...'
      }
    }
  }
}
```

What are the stages in a Jenkins Pipeline?

- Stages are defined steps in the pipeline like Build, Test, and Deploy.

- **Example:** Stages:

    - Build

    - Test

    - Deploy

What is a Jenkins agent (node)?

- An agent is a machine that runs tasks (jobs) as assigned by the master.

- **Example:** A Linux machine can be set as an agent to run Linux-specific builds.

How do you set up Jenkins agents (nodes)?

- Go to "Manage Jenkins" > "Manage Nodes," add a new node, and configure.

- **Example:** Set up a Linux node with SSH.

How does Jenkins handle distributed builds?

- Jenkins uses agents to distribute the build load across multiple machines.

- **Example:** Builds run on different OS environments using specific nodes.

What is a declarative pipeline in Jenkins?

- A more structured way to define pipelines using a simpler syntax.

- **Example:**

Copy code

```
pipeline {
  agent any
  stages {
    stage('Build') {
      steps {
        echo 'Building...'
      }
    }
  }
}
```

⬚ **What is a scripted pipeline in Jenkins?**

- A scripted pipeline uses   for more complex and flexible workflows.

- **Example:**

Copy code

```
node {
  stage('Build') {
    echo 'Building...'
```

```
  }
}
```

❓ How do you choose between declarative and scripted pipelines?

- Use declarative for simplicity and scripted for complex workflows.

- **Example:** Declarative is better for common CI tasks, scripted for advanced control.

❓ What are Jenkins plugins?

- Plugins add extra features or integration with other tools.

- **Example:** GitHub plugin to integrate with GitHub repositories.

❓ How do you install Jenkins plugins?

- Go to "Manage Jenkins" > "Manage Plugins," search, and install.

- **Example:** Install "Docker" plugin for Docker builds.

❓ What are some essential Jenkins plugins you use?

- Git, Pipeline, Blue Ocean, Docker, Email Extension.

- **Example:** Git plugin to pull code from repositories.

❓ How do you manage Jenkins plugins?

- Through "Manage Plugins" to install, update, or disable plugins.

- **Example:** Regularly update plugins to avoid vulnerabilities.

How do you secure Jenkins?

- Use role-based access, enable SSL, secure credentials.

- **Example:** Set up role-based access control using the "Role Strategy" plugin.

How do you create a backup in Jenkins?

- Back up the $JENKINS_HOME directory, which contains configurations and jobs.

- **Example:** Use tools like ThinBackup plugin for scheduled backups.

What is Jenkins Blue Ocean?

- Blue Ocean is a modern user interface for Jenkins Pipelines.

- **Example:** Visualize pipeline stages in a clean, intuitive format.

What is Jenkins Master-Slave architecture?

- The master assigns jobs to slaves (agents), distributing workloads.

- **Example:** Master handles orchestration, agents execute builds.

How do you configure Jenkins master-slave architecture?

- Set up nodes in "Manage Nodes" and configure SSH connections.

- **Example:** Create a Linux agent for Linux-specific builds.

⬚ How do you integrate Jenkins with Git?

- Use the Git plugin, configure the repository URL, and set credentials.

- **Example:** Pull code from a GitHub repo and build it automatically.

⬚ What is Jenkins Multibranch Pipeline?

- A pipeline that automatically builds branches in a repository.

- **Example:** Detect branches in GitHub and build each branch independently.

⬚ How do you set up a Multibranch Pipeline in Jenkins?

- Create a Multibranch Pipeline job and configure the repository.

- **Example:** Jenkins automatically builds all feature branches.

⬚ How do you set environment variables in Jenkins?

- Set environment variables in the pipeline script or job configuration.

- **Example:**

Copy code

```
environment {
  MY_VAR = 'value'
```

}

❓ How do you pass parameters to Jenkins jobs?

- Enable "This project is parameterized" and define parameters.

- **Example:** Add a string parameter BRANCH_NAME to choose a branch during the build.

❓ How do you view build history in Jenkins?

- View build history on the project's main page, showing previous builds and statuses.

- **Example:** Check failed builds and view logs to troubleshoot.

❓ How do you handle failed builds in Jenkins?

- Use post-build actions like sending notifications or rolling back changes.

- **Example:** Configure email notifications for build failures.

❓ How do you handle notifications in Jenkins?

- Use the Email Extension plugin to send notifications for build results.

- **Example:** Send an email on build failure with logs attached.

❓ What is Jenkins Pipeline as Code?

- Pipelines as Code defines build pipelines in a version-controlled Jenkinsfile.

- **Example:** Store Jenkinsfile in the repo to automate pipelines.

⬜ **What are shared libraries in Jenkins?**

- Shared libraries allow reuse of pipeline code across multiple jobs.

- **Example:** Define common functions in a shared library.

⬜ **How do you implement shared libraries in Jenkins?**

- Store reusable   code in a shared repository and load it in Jenkinsfile.

- **Example:**

Copy code

@Library('my-shared-library') _

⬜ **What are post actions in Jenkins pipelines?**

- Actions that run after the pipeline stages, like notifications or cleanups.

- **Example:**

Copy code

```
post {
 always {
  cleanWs()
 }
```

}

How do you monitor Jenkins logs?

- View logs under "Manage Jenkins" > "System Log" or via the command line.

- **Example:** Check Jenkins service logs with tail -f /var/log/jenkins/jenkins.log.

How do you integrate Jenkins with Docker?

- Use the Docker plugin to build, run, and manage Docker containers.

- **Example:** Build and deploy Docker containers in the Jenkins pipeline.

What is the role of Jenkins in DevOps?

- Jenkins automates CI/CD pipelines, enabling faster releases in a DevOps environment.

- **Example:** Jenkins triggers automatic deployments to production.

How do you scale Jenkins for large projects?

- Use master-slave architecture, distributed builds, and cloud integrations to scale.

- **Example:** Use AWS EC2 instances as Jenkins agents to scale builds.

How do you perform load balancing in Jenkins?

- Set up multiple agents and distribute jobs across them for balanced workloads.

- **Example:** Configure nodes for different job types to manage build load.

❓ How do you troubleshoot Jenkins build failures?

- Check the console output, logs, and error messages for build failures.

- **Example:** A failed build due to missing dependencies would show in logs.