

Jenkins Complete Notes

1. What is Jenkins?

- Jenkins is an automation server used for Continuous Integration and Continuous Deployment (CI/CD).
- It runs pipelines (a series of automated steps) to build, test, and deploy code.
- Pipelines are written using Jenkinsfile (Groovy-based syntax).

2. Key Jenkins Concepts

- Job: A task Jenkins performs (freestyle or pipeline).
- Pipeline: A script or declarative set of steps to automate tasks.
- Node: Any machine Jenkins can use to run jobs.
- Agent: Execution environment for pipeline steps (Jenkins node, Docker container, etc).

3. Jenkins with Docker

Option 1: Agent with Docker Image

```
pipeline {  
    agent {  
        docker { image 'python:3.10' }  
    }  
    stages {  
        stage('Test') {  
            steps {  
                sh 'pip install -r requirements.txt'  
                sh 'pytest'  
            }  
        }  
    }  
}
```

```
}  
  
}  
  
}
```

Option 2: Agent with Dockerfile

```
pipeline {  
  agent {  
    dockerfile {  
      filename 'Dockerfile'  
      dir 'app'  
    }  
  }  
  stages {  
    stage('Run Tests') {  
      steps {  
        sh 'pytest'  
      }  
    }  
  }  
}
```

4. Running Jenkins as Docker Container

```
docker run -d -p 8080:8080 -p 50000:50000 \  
-v jenkins_home:/var/jenkins_home \  
-v /var/run/docker.sock:/var/run/docker.sock \  
jenkins/jenkins:lts
```

5. Jenkins with Volumes & Archiving

```
post {  
    always {  
        archiveArtifacts artifacts: '**/report.html', allowEmptyArchive: true  
    }  
}
```

6. Mail Notification in Post Actions

```
post {  
    failure {  
        mail to: 'you@example.com',  
            subject: "Build Failed",  
            body: "Build #${env.BUILD_NUMBER} failed. Check Jenkins."  
    }  
}
```

7. Declarative Pipeline Basics

- pipeline: Entry point
- agent: Defines where to run
- stages: Group of sequential steps
- steps: Actual shell or script commands
- environment: Define env variables
- post: Cleanup or notifications (e.g., always, failure)

8. Jenkins Agent Types Recap

Method	Use Case	Notes
-----	-----	-----

`docker { image '...' }` | Quick builds | Pulls and runs a container

`dockerfile { ... }` | Custom builds | Builds from your Dockerfile

`label 'linux'` | Slave node | Runs on a configured agent node

`none + stage.agent` | Per-stage | Define agent at stage level