**📘 Jenkins Introduction & Overview**

**🧰 What is Jenkins?**

**Jenkins** is an open-source automation server written in Java. It helps automate the **building, testing, and deployment** of software, making it a core part of **CI/CD pipelines**.

**🚀 Key Features**

* **Free and Open Source**
* **Extensible** with 1800+ plugins
* **Supports many tools** (Git, Maven, Docker, Kubernetes, etc.)
* **Web-based interface** for configuration and monitoring
* **Cross-platform**: Runs on Windows, Linux, Mac
* **Scriptable** with Groovy (via Pipeline DSL)

**🛠️ Typical Jenkins Use Cases**

* Automate builds and tests
* Deploy code to staging/production
* Run scheduled jobs (e.g., backups, sync tasks)
* Integrate with version control systems like GitHub/GitLab
* Monitor application health with custom scripts

**📦 Jenkins Architecture**

lua

CopyEdit

+-------------------+ +-------------------+

| Developer PC | | Browser |

+-------------------+ +-------------------+

\ /

\ /

Jenkins Master (Web UI + Controller)

|

+--------+--------+

| | |

Build Agent Build Agent Build Agent

* **Master/Controller**: Manages jobs, UI, and job scheduling.
* **Agent/Node**: Executes jobs on behalf of the master.

**🧪 CI/CD with Jenkins Pipeline**

A **pipeline** defines the series of steps Jenkins takes to build, test, and deploy your application.

**Sample Jenkinsfile (Declarative Pipeline)**

groovy

CopyEdit

pipeline {

agent any

stages {

stage('Checkout') {

steps {

git 'https://github.com/your-repo.git'

}

}

stage('Build') {

steps {

sh './build.sh'

}

}

stage('Test') {

steps {

sh './run-tests.sh'

}

}

stage('Deploy') {

steps {

sh './deploy.sh'

}

}

}

}

**⚙️ Jenkins Setup in 5 Steps**

1. **Install Jenkins**
   * Linux: sudo apt install jenkins
   * Docker: docker run -p 8080:8080 jenkins/jenkins:lts
2. **Unlock Jenkins**
   * Access via http://localhost:8080
   * Use initial admin password from /var/lib/jenkins/secrets/initialAdminPassword
3. **Install Recommended Plugins**
4. **Create Admin User**
5. **Start Creating Jobs**

**🔌 Commonly Used Plugins**

| **Plugin Name** | **Purpose** |
| --- | --- |
| Git | Integrate with GitHub/GitLab |
| Docker | Build/deploy Docker containers |
| Pipeline | Define CI/CD pipelines in code |
| Slack Notification | Send job results to Slack |
| Blue Ocean | Enhanced UI for pipelines |

**🧑‍💻 Jenkins Job Types**

| **Job Type** | **Description** |
| --- | --- |
| Freestyle | GUI-based simple job setup |
| Pipeline | Code-based, flexible, scriptable pipelines |
| Multibranch | Automatically builds branches |
| Folder | Organize jobs in folder hierarchy |

**🛡️ Security Best Practices**

* Always use HTTPS
* Create non-admin users with limited permissions
* Integrate with LDAP/SAML for authentication
* Use secrets management for credentials

**📊 Monitoring Jenkins**

* **Built-in Dashboard**: Job status, queue, build history
* **Metrics Plugins**: Prometheus, JMX
* **Logs**: /var/log/jenkins/jenkins.log

**✅ Best Practices**

* Keep Jenkins and plugins up to date
* Use Jenkinsfile to version pipeline as code
* Run agents in containers or separate nodes
* Use labels to organize agent capabilities
* Backup Jenkins home regularly