**Jenkins**

* Jenkins is a open source project written in java that run on windows, linux, macOS
* Consist of plugins
* It was developed by sun microsystem they named as HUDSON latter they renamed as Jenkins
* Whenever developers write a code, we integrate all the code of all developers at any point of time and we build, test and deliver/deploy to client. This is called CI/CD
* Automation tools is used in Jenkins to automate the projects without any interference of humans.
* Jenkins port number is 8080
* Jenkins overcome the draw back of both agile & waterfall methodologies
* Jenkins is a heart of DevOps
* We will integrate all the DevOps tools with Jenkins & deploy the applications on to the web server (Tomcat)
* We will integration git, maven, dockers, kubernetes, ansible, taraform for deployment
* In Jenkins in order to CI/CD we need to create jobs/projects
* Running the jobs/ projects is known as “Build”
* Build or development both are same
* Jenkins have master /slaves architecture

**Workflow**

* We need to install git, maven, java
* Once developer put the code in github Jenkins pull that code and send to maven for build
* Once build is done, Jenkins pull that code and send to selenium for testing
* Once testing is done, Jenkins pull that code and sends to the artifactory as per requirement
* We can also deploy with Jenkins

**Master and Slaves**

* Master will be one and slaves will be many
* Master will assign work to the slaves/nodes
* Master and slaves will be connected by using SSH key
* To assign work to the slaves/nodes we need to create labels
* It reduces the work load

**Commands to connect Jenkins(master) to Jenkins(slaves/nodes)**

* We need to create one instance

**Jenkins(slaves/nodes)**

* create Jenkins user---> useradd Jenkins
* create a password-🡪 password Jenkins
* we need to give the privileges to slaves-🡪 visudo
* connect to the ssh-🡪vim/etc/ssh/sshd\_config
* restart the file-🡪 systemctl restart sshd
* check the status-🡪 systemctl status sshd

**CI (Continuous Integrations)**

* Developers frequently merge code changes into a shared repository, where automated builds and tests run

**CD (Continuous Deployments)**

* Code changes that pass tests are automatically deployed to the products

**Need for CI/CD**

* Faster software delivery
* Early software delivery
* Early bug detection & fixes
* Reduced Manual effort
* Improved software quality
* Better collaboration

**Types of jobs**

1. **Freestyle project**

* Most basic job type in Jenkins, useful for simple build automation

EX: run a script to compile and packages a java project using maven

1. **Pipeline project**

* Used to defined CI/CD workflows as code using a jenkinsfile
* Supports multiple stages

EX: A Jenkins pipeline for automated testing & deployment of web app

**Important configuration files in Jenkins**

* Jenkins home directory:/var/lib/Jenkins
* Install plugins: /var/lib/Jenkins/plugins
* Created jobs list: /var/lib/Jenkins/workspace
* Node info: /var/lib/Jenkins/nodes
* Jenkins log info: /var/lib/Jenkins/log
* List of jobs: /var/lib/Jenkins/jobs
* Created user list: /var/lib/Jenkins/users