* **Introduction of Jenkins**:
* Jenkins is a free and open-source automation tool. It means which is used to automate the projects without using any interference of humans.
* Jenkins software developed by using java language.
* Jenkins is used to automate application build and deployment process.
* Build and deployment both are different.
* **Build**: taking the code, compiling the code, packaging the code, performing the code review, storing the artifacts into repository. All tasks come under build process.
* **Deploymen**t: deploy the application into tomcat server.
* Jenkins is also called CI/CD software. (continuous integration and continuous deployment)
* Jenkins tool is considered Centre of devops tool.
* **Why it is Centre and heart of devops:**
* Code is available in GitHub.
* Jenkins will interact with maven, sonar qube, nexus server and Apache tomcat.
* All the activities can be done by using Jenkins.
* Development team push the code into GitHub.
* Maven to perform project build process
* Sonar qube to perform code review of the project.
* Nexus to perform build artifacts into nexus server.
* Apache tomcat to perform application deploy into server. So that’s why we called it as Centre and heart devops.
* Jenkins workflow:
* So, it is called as integrated tool.
* It automates the entire procedure from code commit to application deployment.
* Jenkins running on port number 8080.
* It is the heart of devops.
* Jenkins is a plugins-based tool.
* It was developed by Hudson company.
* By using Jenkins, we can collaborate development and operations team.
* Jenkins CI/CD is one of the pioneers in automating the software development lifecycle.
* It provides high security.
* In Jenkins in order to CI/CD, we need to create jobs/projects first.
* We can create users in Jenkins.
* We can give file permissions in Jenkins.
* It is used to implement CI/CD workflows called pipelines
* **Two types of jobs in Jenkins:**
* **Free style job**: free style jobs are the simplest type of job in Jenkins. They providing a GUI for configuration tasks without requiring any scripting.
* **Pipeline jobs:** These are using a DSL domain specific language in Jenkins.
* **Pipeline**:
* pipelines are crucial for automating the software delivery process. There are two types of pipelines. pipeline means line by line execution.

1. **Scripted pipeline**: scripted pipelines are written in groovy, allowing for more flexibility and control over the pipeline’s execution.

* Example of scripted pipeline:
* Scripted pipeline starts with node.
* Node {
* Stage {
* Echo(“print hello world”)
* }
* }

1. **Declarative pipeline**: declarative pipeline provides a more structured and simple syntax for defining Jenkins pipelines.

* Example of declarative pipeline:
* Pipeline {
* Agent any
* Stages {
* Stage
* Steps {
* Echo(“python is a programming language”)
* }
* }
* }
* Declarative pipeline starts with pipeline.
* **Features of Jenkins**:
* Free and open-source automation tool.
* Installation is easy.
* Large number of plugins.
* Pipeline as code
* Real time monitoring
* **What is continuous integration (CI):**
* It is a process where you integrate a set of tools or set of processes that you follow before delivering the application to the client.
* **Continuous integration key features:**
* Automatic builds/tests: correct the defects early.
* Rapid feedback: immediate feedback about the quality of application.
* **What is continuous deployment:**
* It is a process where you deploy/deliver your application to your end user.
* Keeping application war/jar/ear file into server is called deployment.
* Continuous deployment diagram
* **Continuous deployment key features:**
* No human intervention.
* Automatic deployment: the code passes all the test it is automatically deploy to the production environment.
* **What is continuous integration and continuous deployment (CI/CD):**
* It is an approach to software development that combines the practices of CI/CD to make developing applications faster, safer and more efficient.
* CI/CD means whenever got code changed in git repository immediately, we want to do build and deploy that code into server.
* **Advantages of Jenkins:**
* User friendly, free and open source.
* Free of cost.
* It will take very less time for code deployment.
* By using Jenkins, we can collaborate development and operations team.
* Platform independence.
* Remove code errors at an early stage.
* High support plugins.
* **Uses and need of Jenkins in devops:**
* Jenkins plays a crucial role in the devops lifecycle.
* Jenkins in devops helps test any identifies issues simultaneously
* **Continuous integration**: Continuous integration is a software development practice where developers frequently integrate their code changes into shared repository.
* **Automated builds**: it automates the compilation and packaging of applications, reducing manual effort and minimizing errors.
* **Supports for multiple languages and tools**: Jenkins supports wide range of programming languages and can integrate with various tools.
* **Plugins**: Jenkins has a rich ecosystem of plugins that extend its functionality.
* **Monitoring and reporting:** it provides real time feedback on the status of builds and deployments, along with detailed reports on test results, code quality, and performance metrics.
* **Pipeline as code:** Jenkins has a structured pipeline.
* **Faster development cycle**: automating repetitive tasks, Jenkins accelerates the development process, allowing teams to deliver features and fixes more rapidly.
* **Improved collaboration**: it improves the collaboration between development and operation teams by providing a shared platform for building, testing and deploying applications.
* **Scalability:** based upon the needs it increases and decreases the servers. And managing large projects efficiently.
* **Cost-effective**: free and open source
* **Rapid Feedback:** Jenkins provides immediate feedback to developers about their code changes.
* **Parameterized jobs:**
* In Jenkins parameterized build allows you to create flexible and customizable build jobs by defining parameters that can be passed when triggering a build.
* Key features:
* Dynamic input
* Types of parameters: string parameters, Boolean parameters, file parameter.
* Reusability
* Flexibility
* Efficiency
* **Sequential jobs** (upstream and downstream): sequential jobs executed one after the other. Each job must complete before the next one starts.
* **Parallel jobs**: parallel jobs are executed simultaneously. Multiple jobs can run at the same time. These jobs are speed up overall build process.
* **Environments:**
* We will create different environments for one project.
* **Use of environments:** to ensure software bug free and to deliver high quality product to the client on time.
* Dev environment
* QA environment
* UAT environment
* Prod environment
* These environments created by devops engineer.
* Our application will be tested to the different environments before deliver to the client.
* **Jenkins Master/slave architecture:**
* We will have multiple jobs to deploy one application.
* If we run multiple jobs on single server/machine burden will be increase.
* To reduce burden on Jenkins server we can use master/slave architecture.
* **Master node:** master node is responsible distributed the load to slave nodes.
* **Slave node:** slave node will work based on the master node instructions.
* One master node and many slave nodes.
* Jenkins server will be installed in master machine.
* Job will be created on master machine.
* Job execution will happen on slave machine.
* Jenkins important configuration files:
* Jenkins home directory: /var/lib/jenkins
* Installed plugins: /var/lib/jenkins/plugins.
* Created jobs: /var/lib/jenkins/workspace.
* Nodes info: /var/lib/jenkins/nodes.
* Jenkins log info: /var/lib/jenkins/log
* List of jobs: /var/lib/jenkins/jobs.
* Created users list: /var/lib/jenkins/users.
* **Jenkins installation steps:**
* First, we need to take one ec2 instance with name as jenkins.
* Install prerequisites (java, git, and maven)
* After completion of installing these prerequisites, we need install jenkins.
* Install Jenkins by using some commands (Sudo wget commands)
* After completion of jenkins installation we need to active the server by using below command.
* Start the service of Jenkins. (Sudo systemctl start Jenkins)
* To see the status of Jenkins (Sudo systemctl status Jenkins)
* Copy the public Ip of the Jenkins instance and paste it in chrome with :8080
* Jenkins dashboard display.