**Jenkins part- 2**

* **Meaning of plugins:**
* Plugins are software components that add external and additional features to our project.
* In order to install plugins, we need to utilize Jenkins dashboard.
* **Types of plugins:**
* Build plugins
* Source code management plugins
* Notification plugins
* User interface plugins
* Testing plugins
* Deployment plugins
* Monitoring plugins
* Git pligin
* Pipeline plugin
* Slack notification plugin
* Junit plugin
* Job Domain specific language plugin
* Parameterized trigger plugin
* **Ways to install plugins in Jenkins:**
* There are three ways:

1. In Jenkins dashboard-manage Jenkins-manage plugins-available-plugin name-install.
2. We need to install the plugin in our laptop-the download plugin needs to be upload into Jenkins dashboard (manage Jenkins)-advanced –upload plugin-choose download file-select downloaded file-file-open-upload.
3. We need to download the plugin in our local laptop-copy that plugin to Jenkins instance-set a path:/var/lib/Jenkins/plugins.

We will restart the Jenkins instance.

* **Jenkins pipeline:**
* It is a code.
* It has set of stages and stages contains steps will execute entire pipeline.
* In pipeline we have two types:

1. Scripted pipeline:

* Scripted pipeline is written in groovy language. Groovy is a scripting language used to define Jenkins pipeline. Allowing for more flexibility and control over the flow of pipeline execution.
* Scripted pipeline starts with node keyword.

1. Declarative pipeline:

* It is more structure and simple syntax for defining Jenkins pipeline.
* Declarative pipeline starts with pipeline keyword.
* Example:
* Pipeline {
* Agent any
* Stages {
* Stage {
* Steps {
* Echo "hello world”
* }
* }
* }
* }
* **Steps for Jenkins pipeline:**

1. Start Jenkins
2. Install “pipeline” plugins
3. Create a new item-pipeline-stage pipeline-ok
4. Create or get Jenkins file in pipeline section.
5. Pipeline script-select the script.

* **Build trigger:**
* Build triggers in Jenkins are mechanisms that automatically start a build based on creation events or conditions.
* **Types of build triggers:**
* **Webhook:**

Whenever developers commit the code, it will automatically trigger the job.

* **Poll SCM: (Source Control Management)**

When we change the code in a particular time, it will automatically trigger the job.

* **Build periodically:**

It will gets trigger the job, even if the changes are made or not.

* **Parameterized builds:**
* A parameterized build in Jenkins allows you to pass parameters to your build jobs.
* Steps to create parameterized builds:
* We need install plugins.
* Manage plugins-plugins-available-git parameters install
* Go to GitHub-check the branches
* Copy the code URL from the GitHub.
* Go to Jenkins-select new item-give name of the git parameter-freestyle-scm-git-repo link-build-invoke top level maven target-goal-mvn clean package-general-add parameters-apply and save.