* Local🡪Github🡪Jenkins🡪Through Jenkins Plugin Maven🡪Stored in artifactory server As well as it will go to sonarqube(Code test) 🡪Testing server(Selenium will be there it will test everything / If everything ok then it will send to master)🡪Github master🡪From github to Ansible🡪

1. **Declarative pipeline Script Example:**

1st need to install the required plugins (git/Maven, etc..) (For .net msbuild plugin)

And set the environment variable path.(Global Tool Configuration)

Pipeline { agent any

stages stage('Compile') steps{ echo "Compiling the script" } } stage ('Build'){ steps{

echo "Building the script } } stage ('Test'){ steps{ echo "Testing the script" }}stage ('Deploy') steps echo "Deploying the script"

}} stage ('Release') steps { echo "Releasing the script" }}}}

Src (it’s not mandatory, but we can use for the source for vars) / vars (It’s mandatory and we will use all the logical function here only) / resources (it’s not mandatory , we can save our application here like json file )

1. **Configuration Of Maven With Selenium**:

In Jenkins🡪 New item🡪Select **Maven Project** 🡪**General** (Give What details you want to feed) 🡪 **Maven Info Plugin Configuration** (Options are GITHUB Project, Etc..) 🡪 **Source Code Mgmt** (None Or GIT or Subversion) 🡪 **Build Triggers** (Github Hook, Poll SCM, Etc..) 🡪 **Build Environment** (With Ant, etc..) 🡪 **Pre Steps** (Here also some options) 🡪 **Build (Choose Maven version So here we need to click tool config and install (Type Maven name and choose install automatically apply and save)** / Then need choose **Root POM (Copy the POM location and paste it here / Goals & Options (Install Clean etc..) / 🡪POST Steps** (Some rules) **🡪 Build Settings** (E-mail Notification) 🡪 **Post-Build Action** (Some Options is there)

We need to download **TESTNG Results Plugin** and Mention like this (**\*\*/name(Workplace).xml)**

1. **Maven Installtion:**

In Jenkins 🡪 Manage Jenkins 🡪 Manage Plugins 🡪 Search maven and install all related maven plugins.

1. **Jenkin Installation & Restart**

**Jenkins Installation**

1. Download Jenkins 2. Save to any location 3. goto command prompt (Windows) or Terminal (mac)

4. Go to folder where jenkins.war is located 5. java -jar jenkins.war

(After install we will get jenkins administrator password (save it some where)

6. goto browser http://localhost:8080 (Type administrator password to continue)

7. Install required plugins like suggested or manually install

8. Create user or skip / Get started with Jenkins

**Jenkins Restart**

1. Go to command prompt --> press ctrl+c key to come out and again type java -jar jenkins.war

2. Another method from browser http://localhost:8080/restart

1. **Selenium Installation**
2. Selenium Installation 🡪 Need to install JDK 🡪Need to install Eclipse 🡪Then Install JRE🡪Last install selenium
3. **Ansible Jenkins Integration**

* We can run ANSIBLE Playbook from Jenkins pipeline using a plug-in.
* From Apache tomcat we will execute the Ansible Playbook
* I need to install Ansible on my Jenkins (Sudo yum install Ansible)
* If Ansible not install, Then enable epel and try (sudo yum-config-manager –enable epel)

**In Jenkins Side:**

* Install Ansible Plugins in Jenkins (**Ansible** & Ansible Tower)
* We need to configure Jenkins installation path under **global tool configuration**
* Click **Add Ansible** (Provide **name** & **Path to Ansible executables directoy**) / To get Ansible path from linux type **which Ansible** (it will display the path, until **\bin** copy the path)
* Save
* Create a New item (provide **Name** / Select **Pipeline** Project / **Ok**
* Go to **pipeline** tab / Write the **Own SCRIPT** (Declarative pipeline) / Click **Pipeline Syntax** / **Sample step** select **git** / paste Ansible file git **repository url** / Select **Branch** (Master or main or any)
* Again go to **pipeline syntax** / **Sample step** select **Ansible playbook: Invoke an Ansible playbook** / Provide **Ansible tool** name / **playbook file path in workspace** (Ex: **apache.yml** in git, same file name need to provide here) / **inventory filepath in workspace (**Ex: **dev.inv** in git, same name need to provide here) / Then we add **SSH credential details** / Check box tick in **Disable the host SSH key check** / Click **Generate pipeline script** (Copy that script and paste it in **OWN Script**) / Save it / **Build now**