**Read Jenkins Declarative Script from Github using Jenkins pipeline**

**Note:**  This Jenkins script contain maven build , scm code checkout and sonarqube server Itegration and tomcat deployment stages

**Note**: We can see list of credentials that we already created in Jenkins

Dashboard 🡪 Manage Jenkins 🡪 Credentials under Security [scroll down you will see]

**Step1**: make sure tomcat should not have the root user, so create one new user using

$ sudo useradd –R tomcat

**Step2**: if you wanna change the from root user to newly created user then use below cmd.

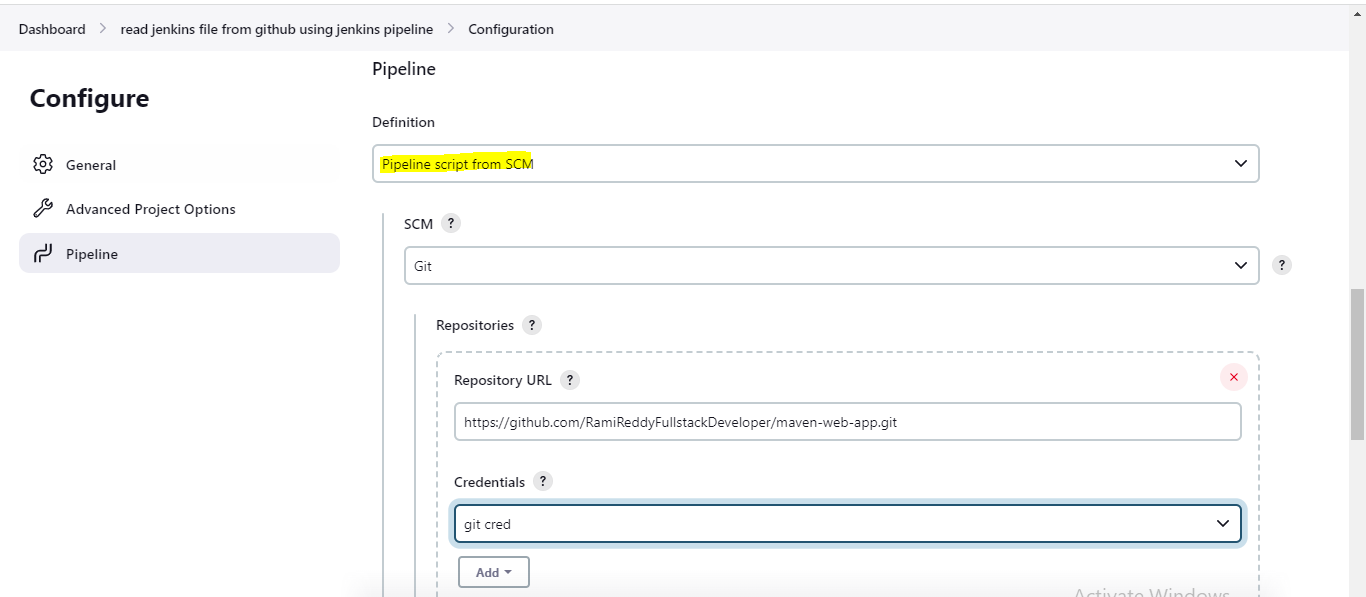
$ sudo chown –R tomcat:tomcat

**Step1:**  Login into Jenkins using username and password (default username and password admin)

**Step2**: Jenkins Dashboard ---> Click New Item ---> Name : read jenkins script path from github using jenkins pipeline and select pipeline and click **OK.**

**Step3**: provide description:

**Step4**: go to pipeline definition and **select Pipeline Script from SCM** and provide Repository URL and add git Credentials.



**Step5**: look at **Branches to Build** section continuation of above screenshot and specify **which branch** we need to tell Jenkins to pick and also specify the **script path** (script file should be placed root of the project.

**Note**: Refer <https://github.com/RamiReddyFullstackDeveloper/maven-web-app> for real time jenkins script path.

**Script**:

currentBuild.displayName = "Tomcat-Demo"+currentBuild.number

pipeline {

agent any

tools {

maven 'MVN\_HOME' //this is we need to configure in jeninns Global Tool Configuration

}

stages {

stage('SCM-Checkout') {

steps {

git branch: 'main', credentialsId: 'git\_cred', url: 'https://github.com/RamiReddyFullstackDeveloper/maven-web-app.git'

}

}

stage('mvn-package') {

steps {

sh 'mvn clean compile package'

}

}

stage('Tomcat-deployment') {

steps {

sshagent([**'tomcat-cred-ssh'**]) {

sh """

scp -o StrictHostKeyChecking=no target/tomcat-demo.war ec2-user@3.110.94.5:/opt/tomcat/webapps/

ssh ec2-user@3.110.94.5 /opt/tomcat/bin/shutdown.sh

ssh ec2-user@3.110.94.5 /opt/tomcat/bin/startup.sh

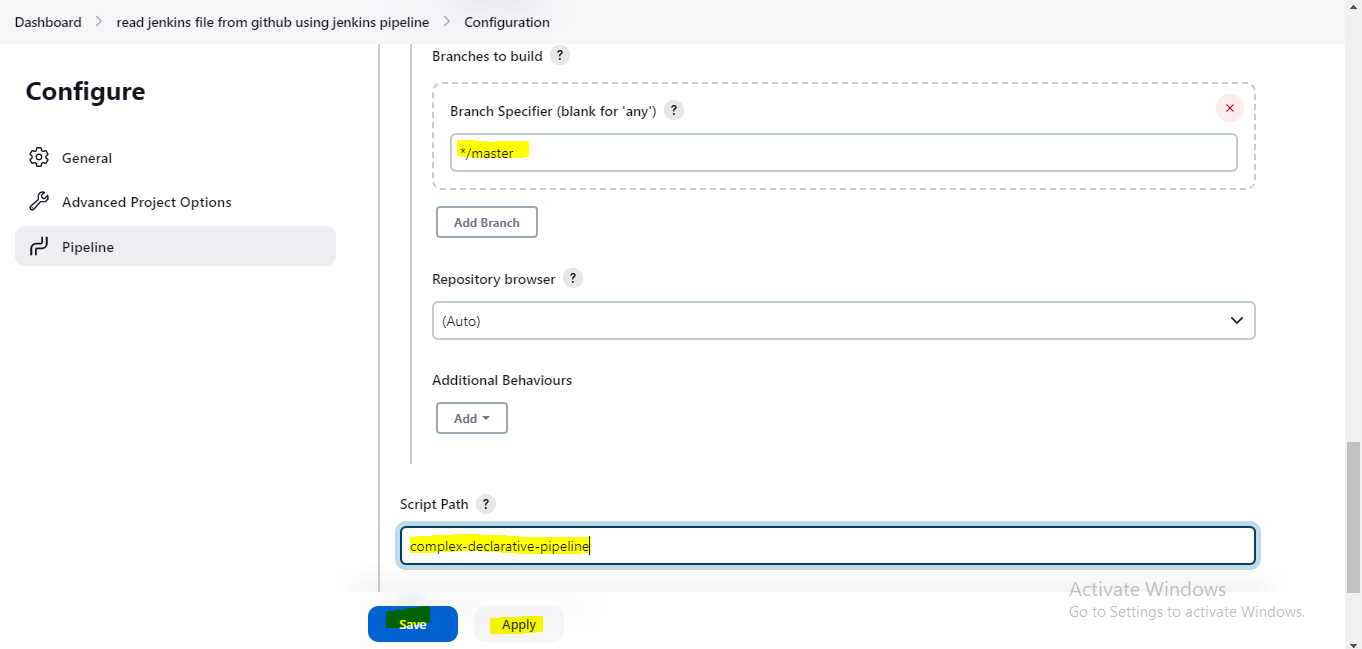
"""

}

}

}

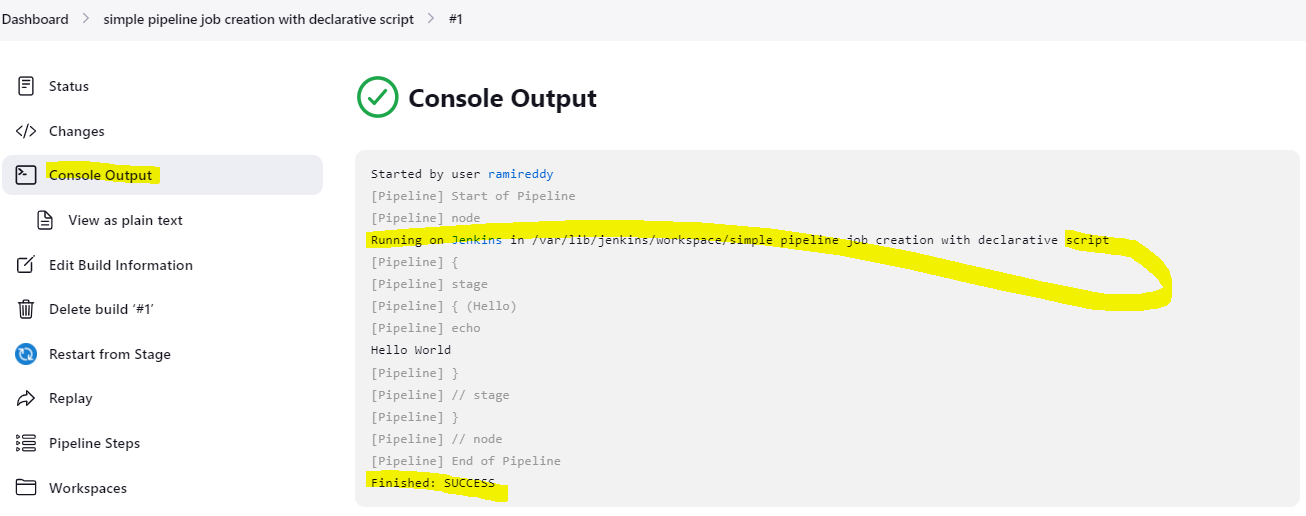
}

} 

**Step6**: Click Apply and Save

**Step7:** Now click Build Now.

**Step8**: Click on build number and click console output for viewing the logs.

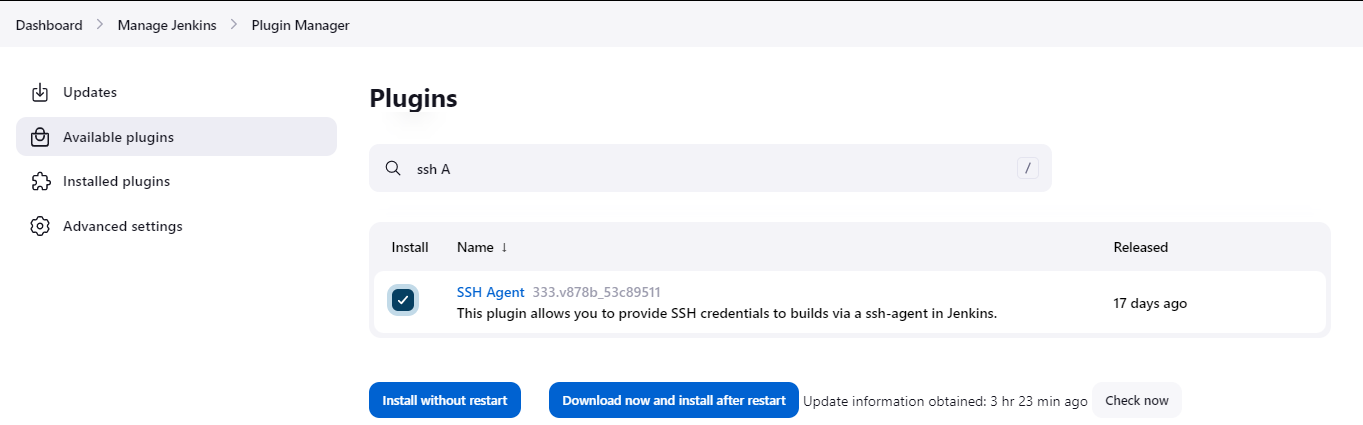


**Step8**: For editing the job click configure.

**Tomcat deployment stage configuration:**

**Step1:** Go to Jenkins Dashboard ---> Manage Jenkins ---> Available Plugins -> search for **SSH Agent**

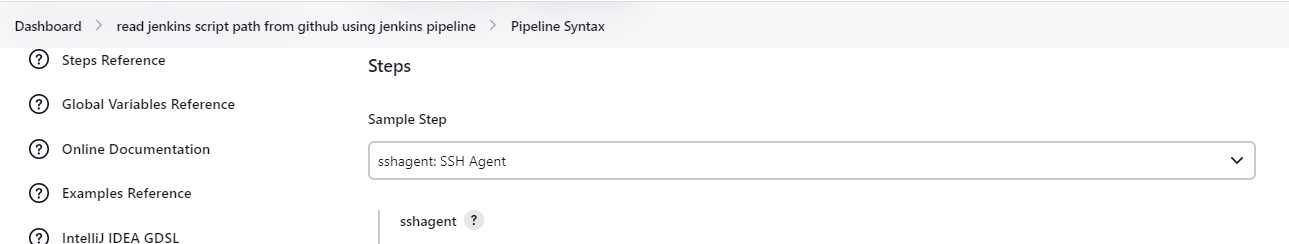
And click on Download now and install after restart.



**Step2**: After install and click restart button.

**Step3**: Now go to pipeline section all the way down and click on the **pipeline syntax.**

**Step4**: now look for **ssh Agent** from **Sample Step** drop down.



**Step5**: Click Add Button and select **Jenkins** for adding tomcat credentials.

Select **kind** as SSH Username with private key

**ID** give any name and **description** also same as ID

**Step6:** enter user name as ec2-user

**Step7:** select radio button enter directly asprivate key.

**Step8**: And click Add as key

**Step9**: if you using windows machine you might have .ppk file in your local for connecting putty. now generate the .pem key using ppk file using putty gen.

**Step10**: now open the .pem file and copy the private key and paste into private key section.





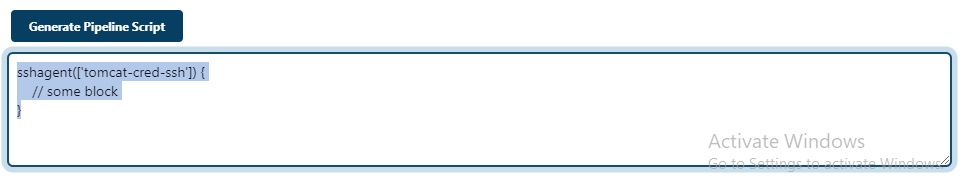
**Step11:** Now click Add

**Step12**: now click generate pipeline script.

sshagent([**'tomcat-cred-ssh'**]) {

// some block

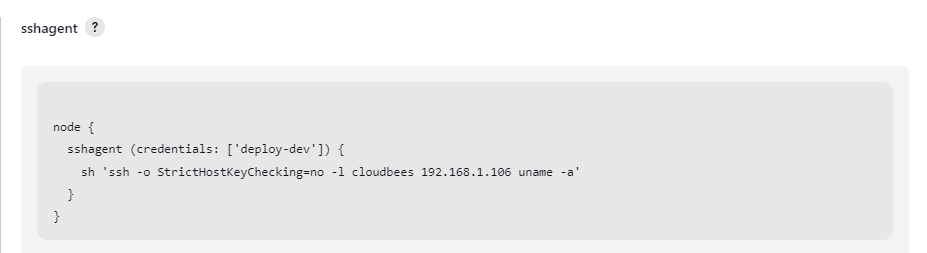
}



**Step13**: this is the below over all from step4 to step12



**Step14**: Now click above highlighted question mark.



**Step15**: From step number 12 and 14

**'tomcat-cred-ssh'** is the ID at the time of we create credentials for tomcat refer **step** 5

stage('Tomcat-deployment') {

steps {

sshagent(['tomcat-cred-ssh']) {

sh """

scp -o StrictHostKeyChecking=no **target/tomcat-demo.war** [ec2-user@3.110.94.5:/opt/tomcat/webapps/](mailto:ec2-user@3.110.94.5:/opt/tomcat/webapps/)

ssh ec2-user@3.110.94.5 /opt/tomcat/bin/shutdown.sh #tomcat server ip and it’s installation path of shutdown

ssh ec2-user@3.110.94.5 /opt/tomcat/bin/startup.sh #tomcat server ip and it’s installation path of startup

"""

}}}

**Note**:  **target/tomcat-demo.war** forthis location pls refer **mvn package** stage and provide location here

stage('mvn-package') {

steps {

sh 'mvn clean compile package'

}

}

**Step16**: Make sure above **ssh user name (step15)** and at the time of **credentials creation user name** (step5) is must be same as **tomcat server installation user name** not tomcat server user name even not mandatory but it’s recommended.