**Build and Configure CI/CD Pipeline with Selenium WebDriver**

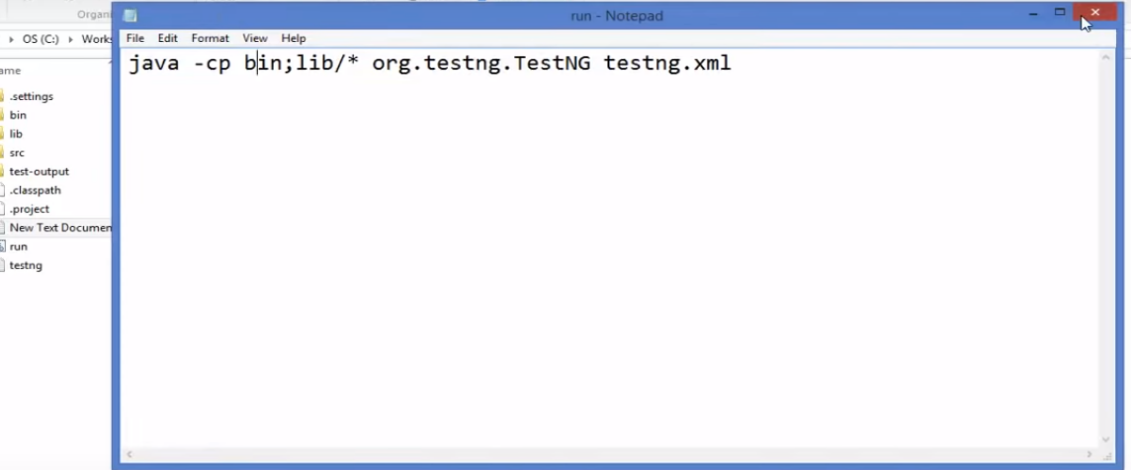
**Step 1:** Foking the git repository

* Fork the following repository

<https://github.com/canindit75/JenkinsDemo>

**Step 2:** Create a Jenkins pipeline job

* Jenkins.war file is already present in your practice lab in cd /usr/share/jenkins directory.
* Go to jenkins.war location Start the Jenkins by using command on command prompt:**java -jar jenkins.war.**
* Open browser and type **localhost:8080.**
* Enter the password.
* Create a job.
* Pass a name.
* Select **Pipeline.**
* Click on Ok.
* Create a text file name it **run.sh** in your lab and keep the below given code in it.



* Give executable permission to **run.sh** using the commands below:

**chmod 755 run.sh**

**chmod 777 run.sh**

* Push **run.sh in your repository** under master branch.

**git push <reponame> master**

**git status**

* Go to Jenkins pipeline job.
* Write a groovy script in the pipeline.

node {

def mvnHome

stage('Preparation') { // for display purposes

// Get some code from a GitHub repository

git 'https://github.com/jglick/simple-maven-project-with-tests.git'

// Get the Maven tool.

// \*\* NOTE: This 'M3' Maven tool must be configured

// \*\* in the global configuration.

mvnHome = tool 'maven3'

}

stage('Build') {

// Run the maven build

withEnv(["MVN\_HOME=$mvnHome"]) {

**if** (isUnix()) {

sh '"$MVN\_HOME/bin/mvn" -Dmaven.test.failure.ignore clean package'

} **else** {

sh ‘"%MVN\_HOME%\bin\mvn" -Dmaven.test.failure.ignore clean package’

}

}

}

stage('Results') {

junit '\*\*/target/surefire-reports/TEST-\*.xml'

archiveArtifacts 'target/\*.jar'

}

}

* Click on Apply and Save.
* Click on Build now.