**Setup the Java/JDK**

Download and setup the java (from Oracle website) on the build box and setup the java as the environment path variable and check the java setup by seeing the version



**Setup the Maven**

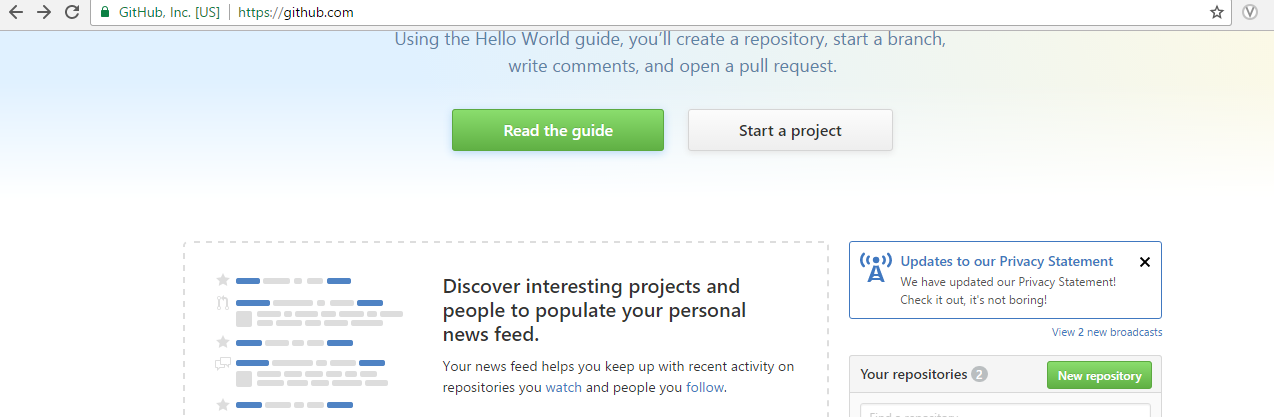
Download the maven package from <https://maven.apache.org/> and defined path in the environment variable for maven



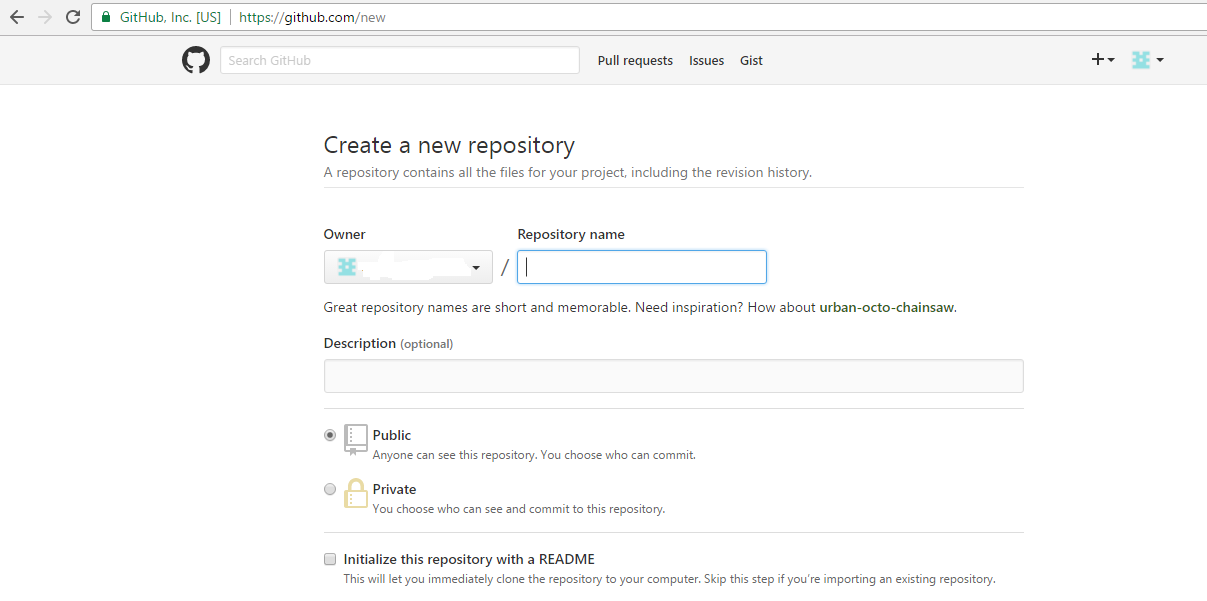
**GITHUB**

Register in GitHub in <https://github.com>

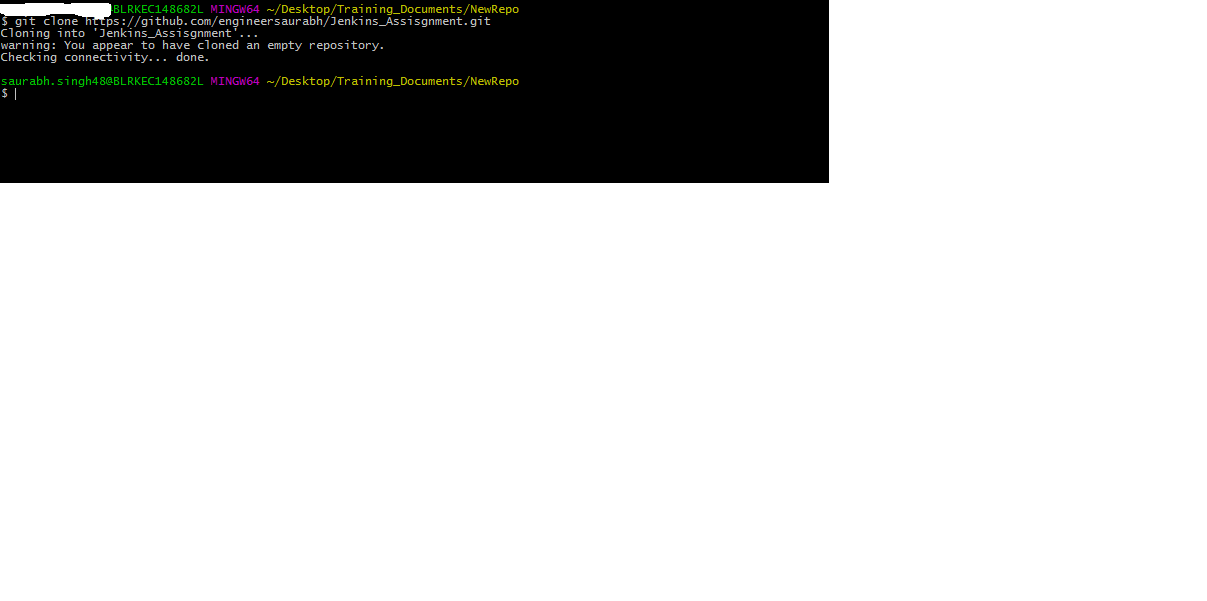
Click on New Repository



Create a repository where you want to create the application repository i.e. Jenkins\_Assisgnment



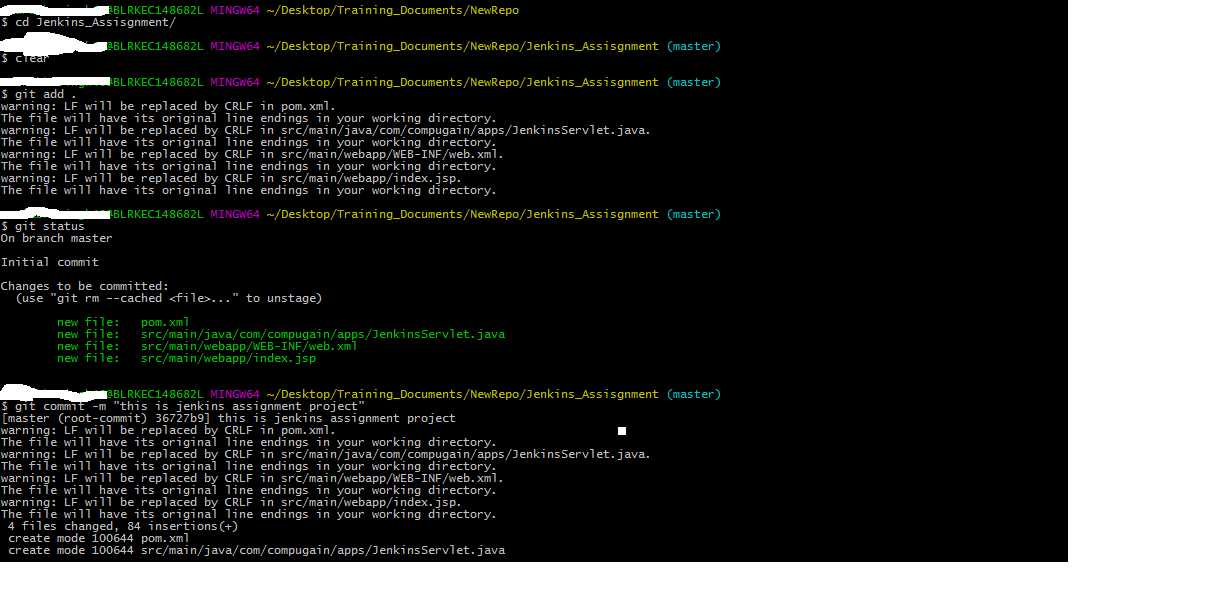
Open Git bash command after setup and clone the repository in the local drive



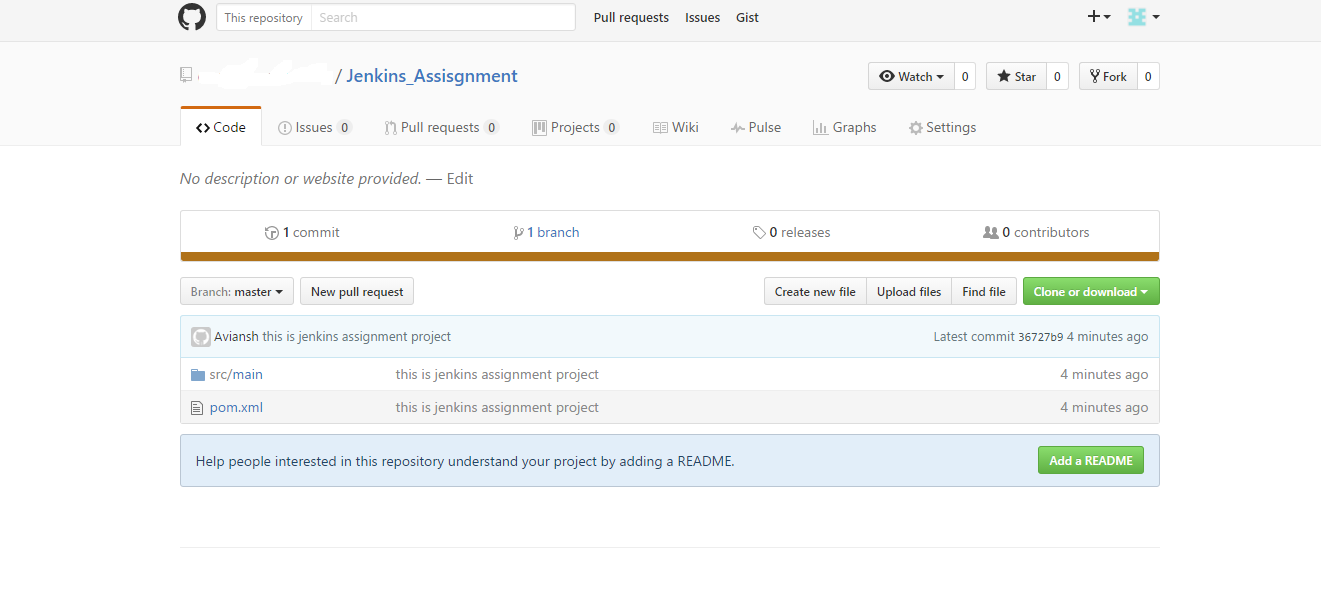
git add .

git commit –m “ this is my Jenkins project “

git push

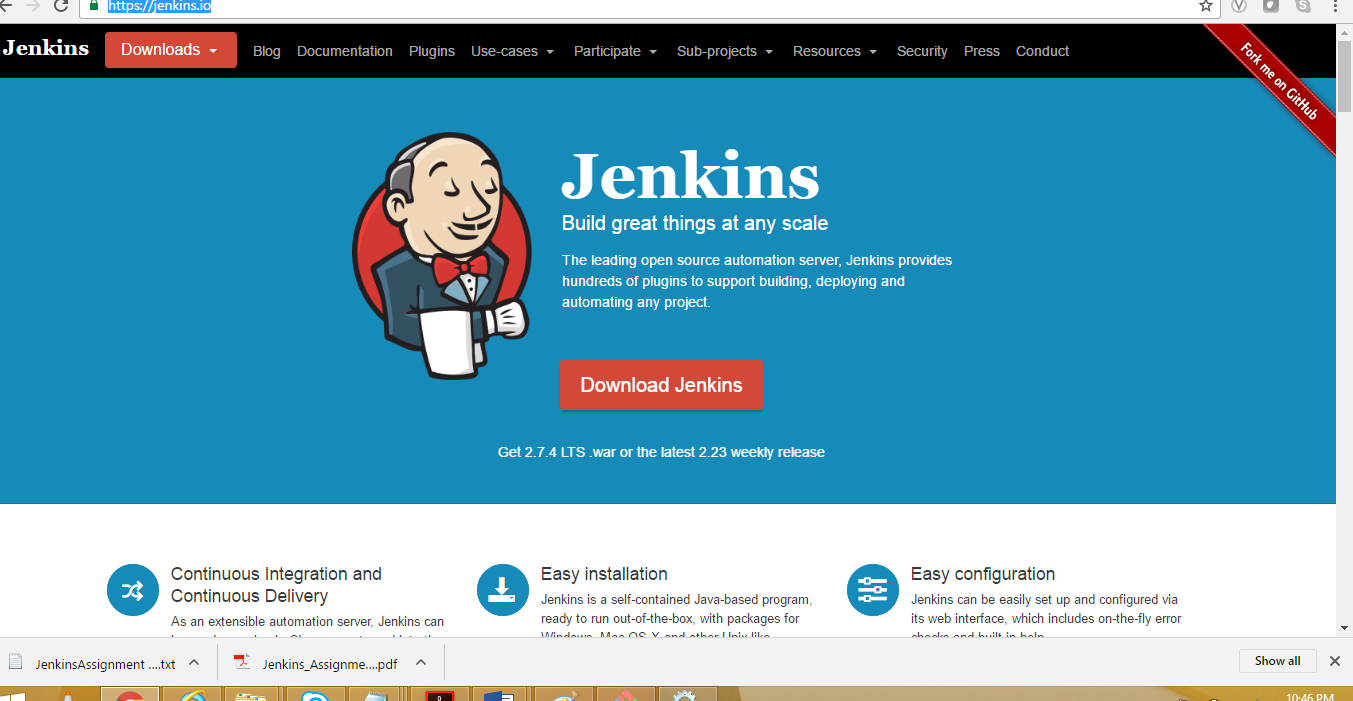


File check-in in the repository



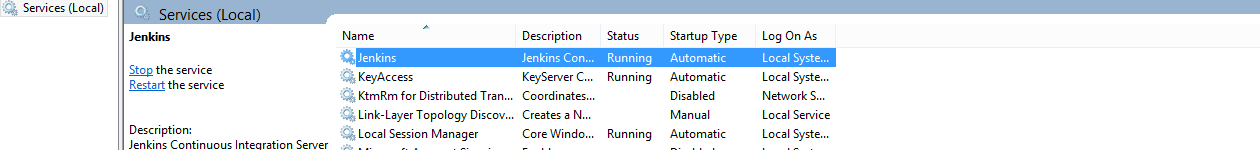
Setup the Jenkins

URL - <https://jenkins.io/>

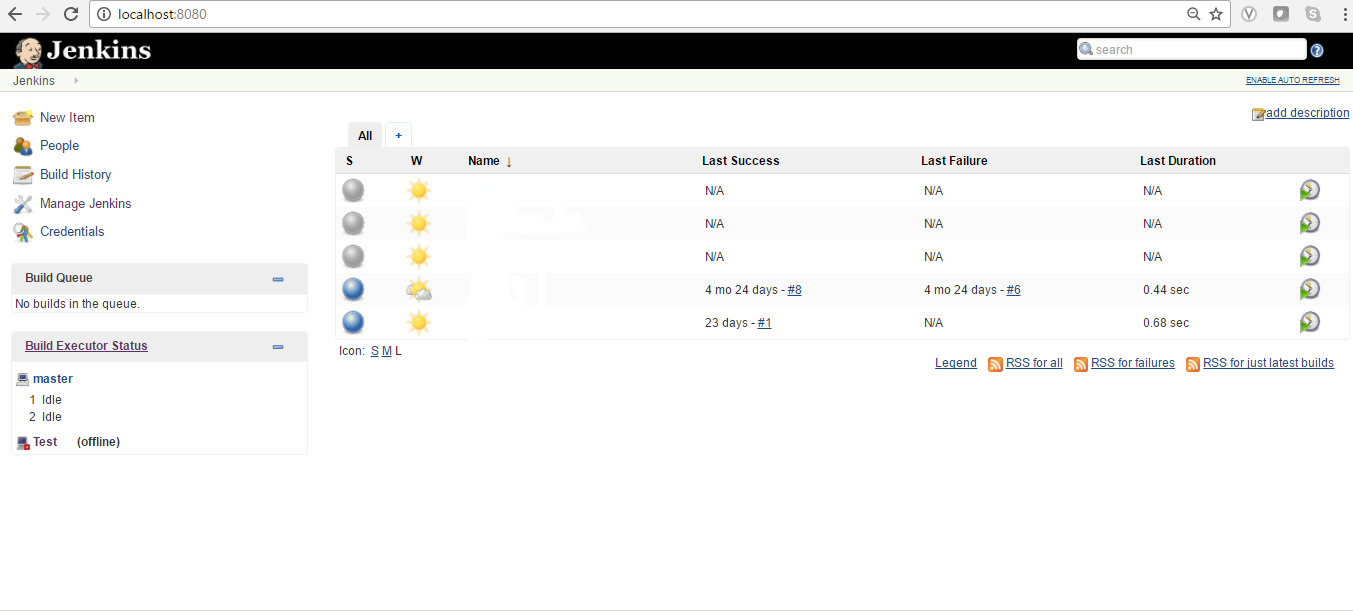


Downloaded the window versions and make the setup on the environment

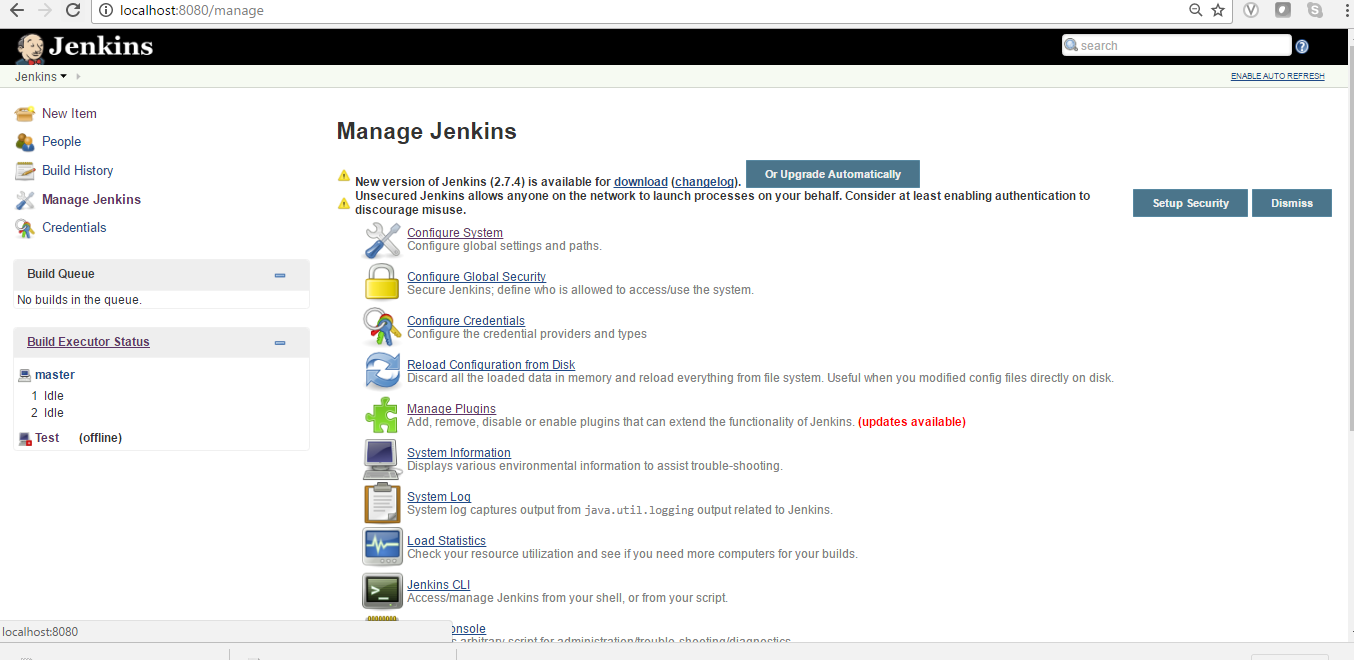
Setup the Jenkins on the window environment and Jenkins is running as services



Jenkins Dashboard



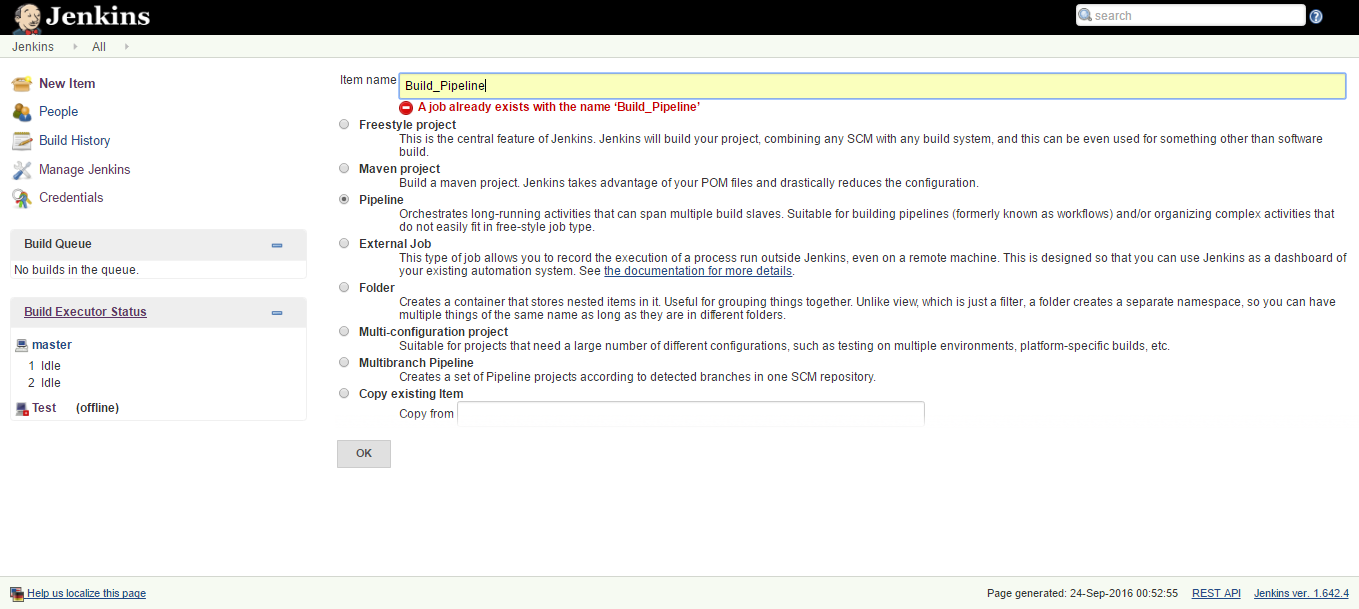
Click on Mange Jenkins for adding the plugin which is required like git build and other Jenkins build pipeline workflow plugin



Search for github plugin and add the plugin and restart the Jenkins

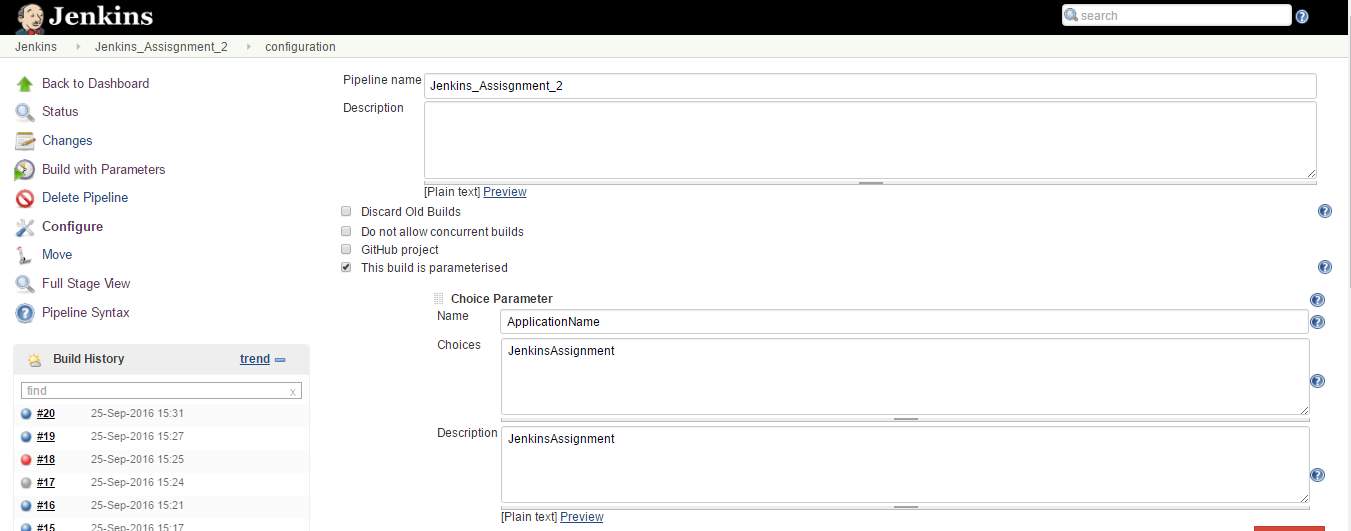
Search for the Build pipeline or download .hpi file and upload through manage

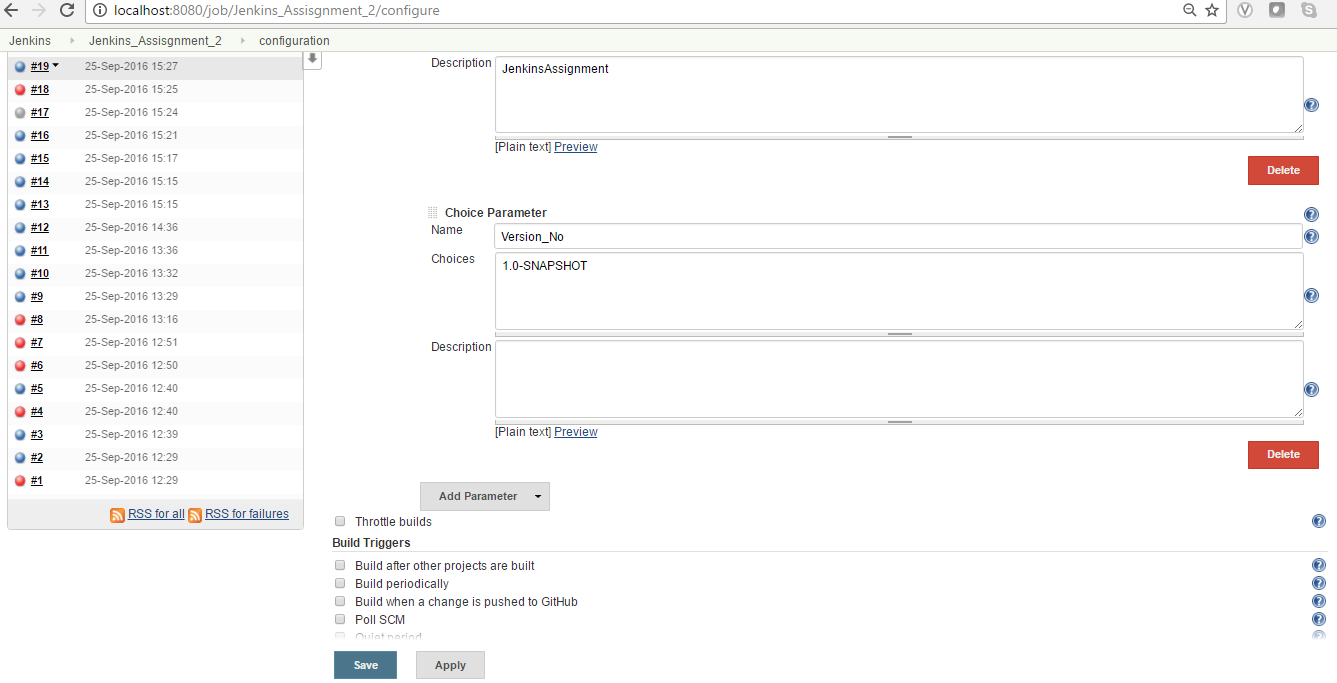
Configure the Build Pipeline job and select pipeline option for the creation of the job

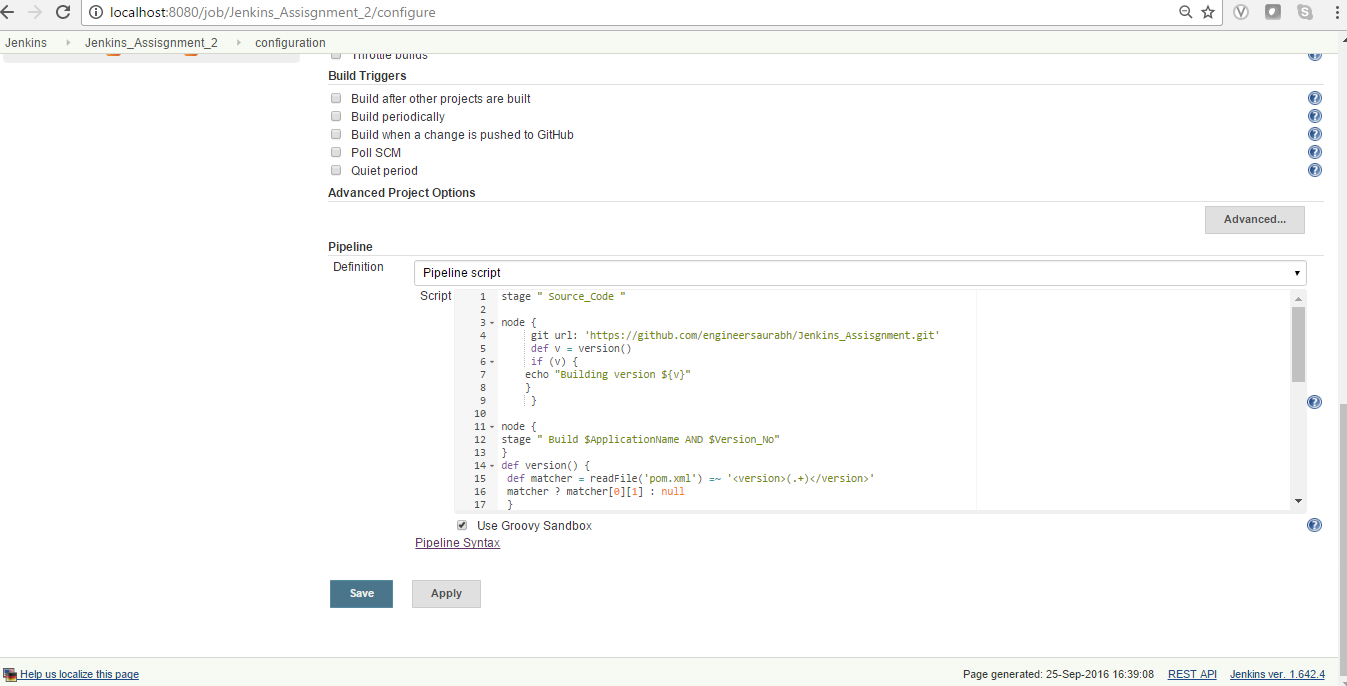


Configure Build Job

Define Parameters as in the screen shot







**Configure Groovy Scripting as below**

**Groovy Code without deployment on the EC2**

stage " Source\_Code "

node {

git url: 'https://github.com/karthiksys/Jenkins\_Assignment.git'

def v = version()

if (v) {

echo "Building version ${v}"

}

}

node {

stage " Build $ApplicationName AND $Version\_No"

}

def version() {

def matcher = readFile('pom.xml') =~ '<version>(.+)</version>'

matcher ? matcher[0][1] : null

}

node {

def mvnHome = tool 'M3'

bat "D:\\Software\\apache-maven-3.3.9-bin\\apache-maven-3.3.9\\bin\\mvn -B -Dmaven.test.failure.ignore verify clean package"

}

node {

stage "Awaiting for Approval for deploying $ApplicationName AND $Version\_No "

timeout(time: 5, unit: 'MINUTES') {

input message: 'Awaiting approval for deploying $ApplicationName and $Version\_No. Are you ready to deploy?'

}

}

node {

stage " Deploying $ApplicationName AND $Version\_No "

echo " NO real EC2 environment.. Just a blank stage"

}

node {

stage " Executing Test "

echo " NO real EC2 environment.. Just a blank stage"

}

**Screen Shot of the Build Pipeline**

