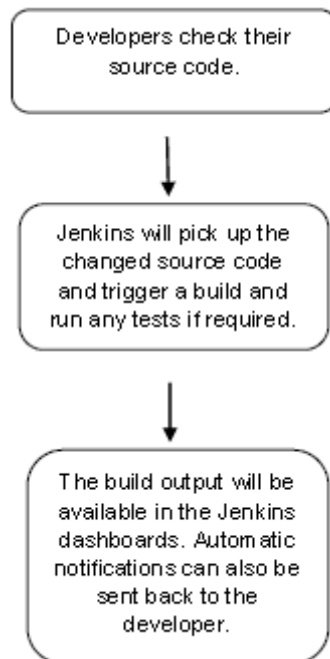


TestNG With Jenkins

Jenkin

Jenkins is a software that allows **continuous integration**. Jenkins will be installed on a server where the central build will take place. The following flowchart demonstrates a very simple workflow of how Jenkins works.



Along with Jenkins, sometimes, one might also see the association of **Hudson**. Hudson is a very popular open-source Java-based continuous integration tool developed by Sun Microsystems which was later acquired by Oracle. After the acquisition of Sun by Oracle, a fork was created from the Hudson source code, which brought about the introduction of Jenkins.

What is Continuous Integration?

Continuous Integration is a development practice that requires developers to integrate code into a shared repository at regular intervals. This concept was meant to remove the problem of finding later occurrence of issues in the build lifecycle. Continuous integration requires the developers to have frequent builds. The common practice is that whenever a code commit occurs, a build should be triggered.

Advantages of using Jenkins are:

- It is a cross-platform and can be used on Windows, Linux, Mac OS, and Solaris environments
- It is a free and open source tool
- Widely used and well documented
- Integration with wide variety of tool and technologies

Apart from Jenkins, we have many more tools in the market such as:

- Anthill
- Bamboo
- Cruise Control
- Team City and many more.

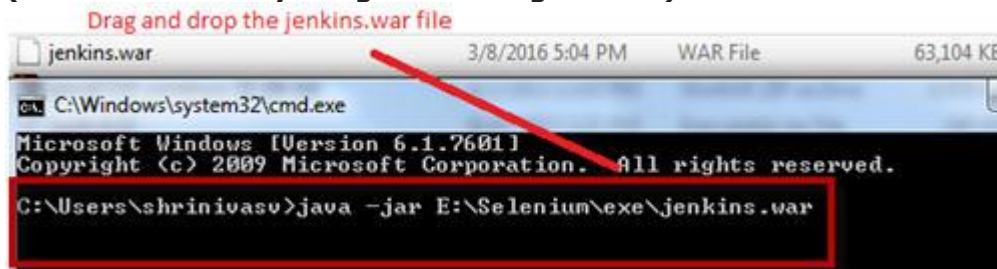
Step #1:

Download Jenkins from the official website of Jenkins – **Jenkins**. Download the latest .war file. Jenkins can be started via the command line or can run in a web application server.

Refer to the below steps for the execution through the command line:

1) Open the command prompt and type **java -jar** and enter path of .war file

(Note: Click on any image for enlarged view)



2) Press enter and check if your Jenkins.war file started to run and check the status information on the command prompt console.

It should show – **Jenkins is fully up and running**

```
C:\Windows\system32\cmd.exe - java -jar E:\Selenium\exe\jenkins.war
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\shrinivasu>java -jar E:\Selenium\exe\jenkins.war
Running from: E:\Selenium\exe\jenkins.war
webroot: $user.home/.jenkins
Apr 03, 2016 9:44:17 PM winstone.Logger logInternal
INFO: Beginning extraction from war file
Apr 03, 2016 9:44:17 PM org.eclipse.jetty.util.log.JavaUtilLog info
INFO: jetty-winstone-2.9
Apr 03, 2016 9:44:23 PM org.eclipse.jetty.util.log.JavaUtilLog info
INFO: NO JSP Support for , did not find org.apache.jasper.servlet.JspServlet
Jenkins home directory: C:\Users\shrinivasu\.jenkins found at: $user.home/.jenki
ns
Apr 03, 2016 9:44:25 PM org.eclipse.jetty.util.log.JavaUtilLog info
INFO: Started SelectChannelConnector@0.0.0.0:8080
Apr 03, 2016 9:44:25 PM winstone.Logger logInternal
INFO: Winstone Servlet Engine v2.0 running: controlPort=disabled
Apr 03, 2016 9:44:26 PM jenkins.InitReactorRunner$1 onAttained
INFO: Started initialization
Apr 03, 2016 9:44:28 PM jenkins.InitReactorRunner$1 onAttained
INFO: Listed all plugins
Apr 03, 2016 9:44:28 PM jenkins.InitReactorRunner$1 onAttained
INFO: Prepared all plugins
Apr 03, 2016 9:44:28 PM jenkins.InitReactorRunner$1 onAttained
INFO: Started all plugins
Apr 03, 2016 9:44:28 PM jenkins.InitReactorRunner$1 onAttained
INFO: Augmented all extensions
Apr 03, 2016 9:44:35 PM jenkins.InitReactorRunner$1 onAttained
INFO: Loaded all jobs
Apr 03, 2016 9:44:39 PM hudson.model.AsyncPeriodicWork$1 run
INFO: Started Download metadata
Apr 03, 2016 9:44:41 PM org.jenkinsci.main.modules.sshd.SSHD start
INFO: Started SSHD at port 53385
Apr 03, 2016 9:44:41 PM jenkins.InitReactorRunner$1 onAttained
INFO: Completed initialization
Apr 03, 2016 9:44:41 PM jenkins.InitReactorRunner$1 onAttained
INFO: Started initialization
Apr 03, 2016 9:44:41 PM jenkins.InitReactorRunner$1 onAttained
INFO: Listed all plugins
Apr 03, 2016 9:44:41 PM jenkins.InitReactorRunner$1 onAttained
INFO: Prepared all plugins
Apr 03, 2016 9:44:41 PM jenkins.InitReactorRunner$1 onAttained
INFO: Started all plugins
Apr 03, 2016 9:44:41 PM jenkins.InitReactorRunner$1 onAttained
INFO: Augmented all extensions
Apr 03, 2016 9:44:41 PM jenkins.InitReactorRunner$1 onAttained
INFO: Loaded all jobs
Apr 03, 2016 9:44:41 PM jenkins.InitReactorRunner$1 onAttained
INFO: Completed initialization
Apr 03, 2016 9:44:41 PM hudson.WebAppMain$3 run
INFO: Jenkins is fully up and running
```

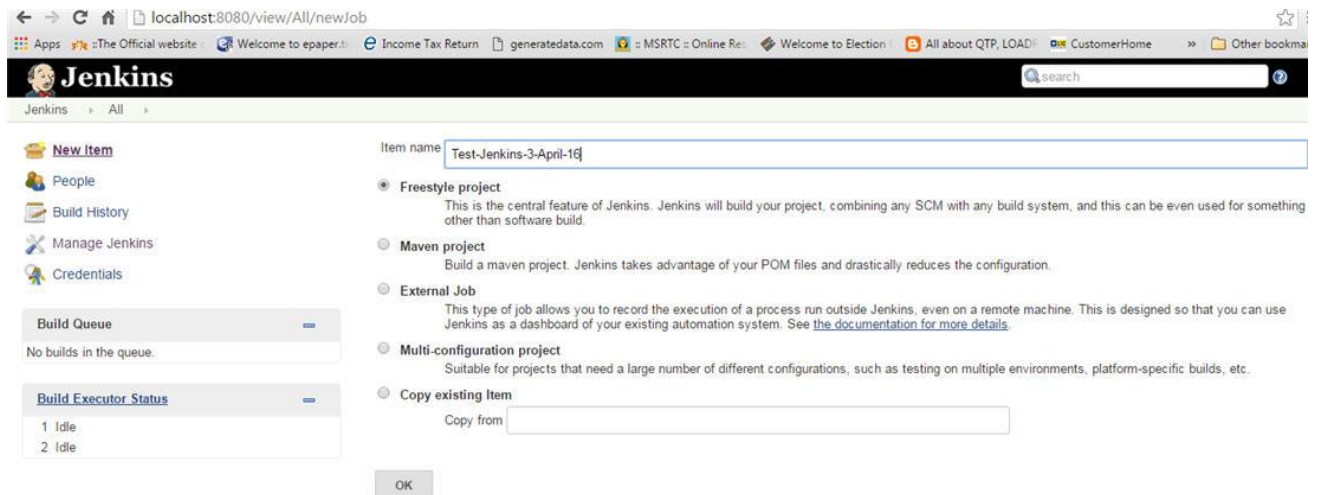
3) Now check whether your Jenkins is ready to use; by default, it uses port 8080. Type "<http://localhost:8080>" in the browser and press enter. It will show you Jenkins UI.

S	W	Name	Last Success	Last Failure	Last Duration
		Demo-Srinivas-Jenkins	24 days - #9	N/A	9.5 sec
		QA_Metrics_Kill_Production_Report	17 days - #1	N/A	4 min 0 sec
		Test	N/A	N/A	N/A

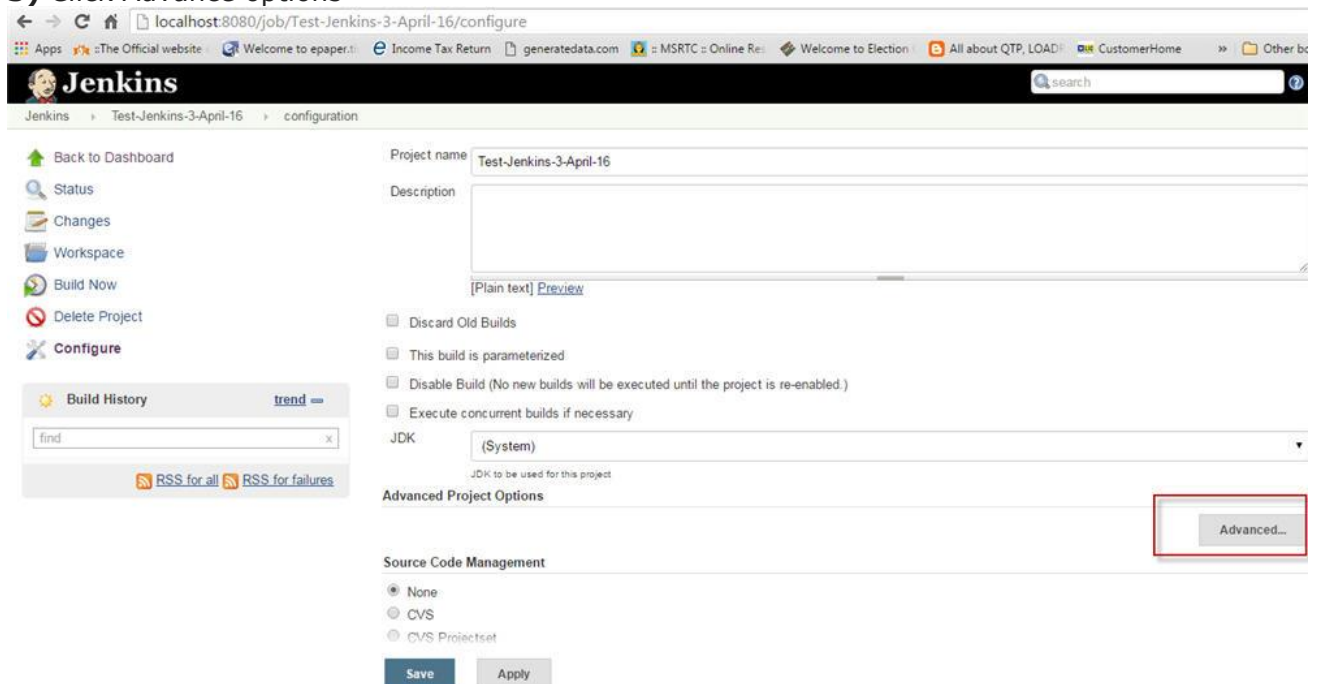
It will load the Jenkins dashboard empty by default and user has to set up the id and password .I created some Jenkins job in the above screenshot as an example and hence, it did not load empty.

For adding the batch file follow the below steps:

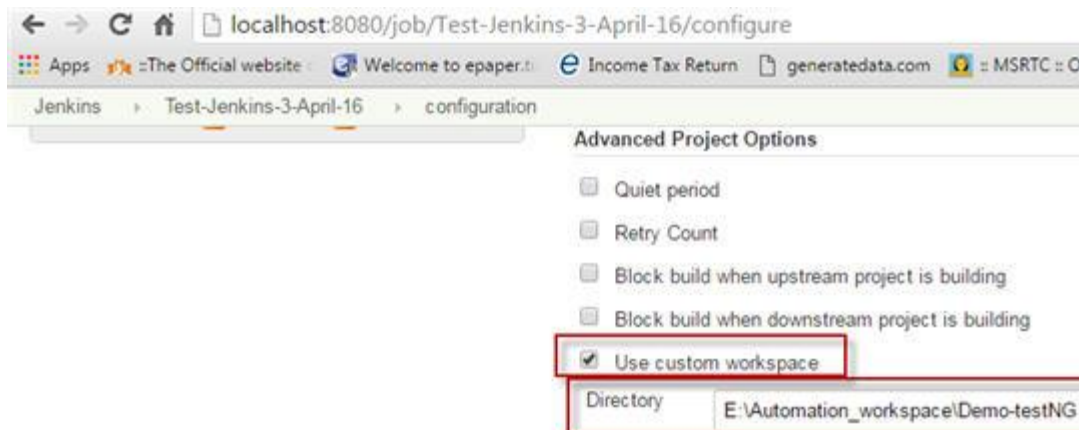
- 1) Go to the Jenkins dashboard, create a new job in Jenkins
- 2) Click on new item and enter the item name and check the freestyle project radio button



3) Click Advance options



- 4) Click on use custom workspace and give your Selenium script project workspace path: "E:\Automation_workspace\Demo-testNG"



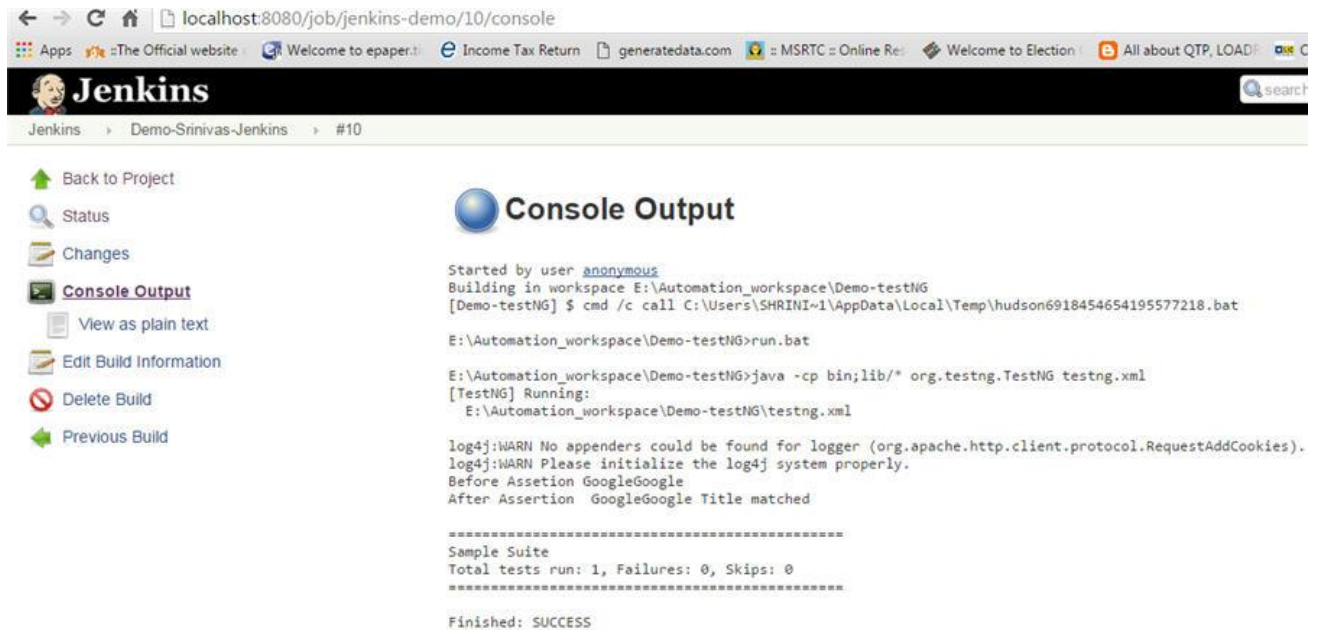
5) Then go to Build and Select option from drop down box, execute your build through Windows batch command

6) And give your batch file name here – “run.bat”



7) Click on apply and save

8) Click on build now and see the build result on console output



So far we have learned:

- How to start Jenkins
- How to configure Jenkins with Selenium
- Creating your batch file and executing it through Jenkins.

As you all are aware Jenkins is a very powerful tool which is mainly used for running nightly builds. Hence, we shall now learn how to schedule your build and send email notifications to the concerned team.

Scheduling Jenkins job

For scheduling your batch file, perform the below steps:

- Go to dashboard and click on the Jenkins job
- Click on configure and then on advanced option
- Then go to **Build triggers** and select **Build periodically** option and enter your cron job pattern

The screenshot shows the Jenkins job configuration page for a job named 'Test'. The browser address bar shows 'localhost:8080/job/Test/configure'. The page has several sections: 'Source Code Management' with radio buttons for 'None', 'CVS', 'CVS Projectset', and 'Subversion'; 'Build Triggers' with checkboxes for 'Build after other projects are built' and 'Build periodically' (which is checked and highlighted with a red box); a 'Schedule' text field containing '*****' (also highlighted with a red box); a warning message below the schedule field: 'Do you really mean "every minute" when you say "*****"? Perhaps you meant "H * * * * *" to poll once per hour. Would last have run at Sunday, April 3, 2016 11:18:19 PM IST, would next run at Sunday, April 3, 2016 11:18:19 PM IST.'; 'Poll SCM' checkbox; 'Build' section with an 'Add build step' button; 'Post-build Actions' section with an 'Add post-build action' button; and 'Save' and 'Apply' buttons at the bottom.

- To understand cron job pattern follow this [wiki link](#)
I entered * * * * * which means it will run my job every minute

- Click on apply and save
There is no manual intervention. After scheduling the script, it will run on the scheduled time.

How to add email notifications

Next, we will cover how to add email notifications.

Refer the below steps:

- Go to the section 'Manage Jenkins'
- Click on configure system
- Select Email notification

E-mail Notification

SMTP server	<input type="text"/>
Default user e-mail suffix	<input type="text"/>
<input type="checkbox"/> Use SMTP Authentication	
Use SSL	<input type="checkbox"/>
SMTP Port	<input type="text"/>
Reply-To Address	<input type="text"/>
Charset	<input type="text"/>
<input type="checkbox"/> Test configuration by sending test e-mail	

- Give your SMTP server address. I am using Gmail, as I can't mention my official server address. To know your official server address, contact your network support team
- I entered SMTP server name = **smtp.gmail.com**
- Click on the advance link and check Use SMTP Authentication check box
- Provide username, password and SMTP port number; it is 465 for Gmail. Check charset and make sure it is = UTF-8

localhost:8080/configure

Official website: Welcome to epaper.ti Income Tax Return generatedata.com MSRTC :: Online Re: Welcom

Configuration

Shell executable

E-mail Notification

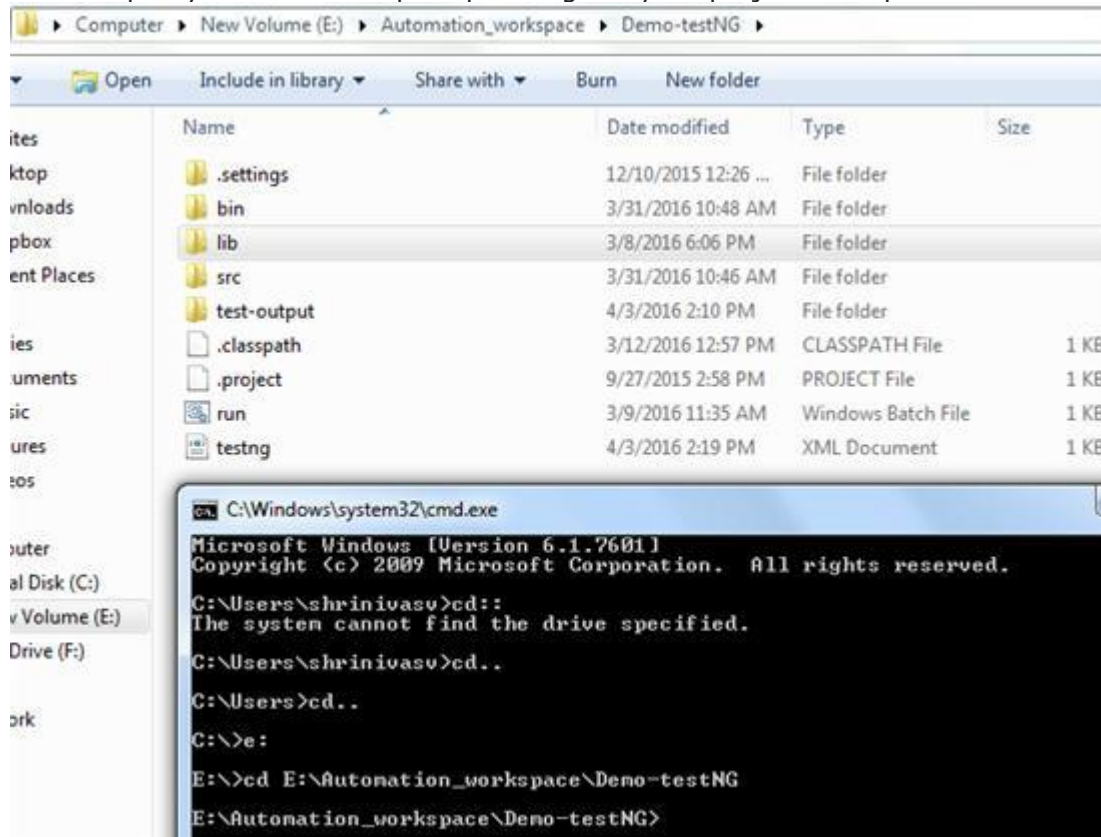
SMTP server	smtp.gmail.com
Unknown host name: smtp.gmail.com	
Default user e-mail suffix	<input type="text"/>
<input checked="" type="checkbox"/> Use SMTP Authentication	
User Name	testglam321@gmail.com
Password
Use SSL	<input checked="" type="checkbox"/>
SMTP Port	465
Reply-To Address	<input type="text"/>
Charset	UTF-8
<input type="checkbox"/> Test configuration by sending test e-mail	

- Check your email configuration settings by clicking on Test configuration button.
- So, whenever the build passes or fails you will get the email notification.

Running Selenium script through command line

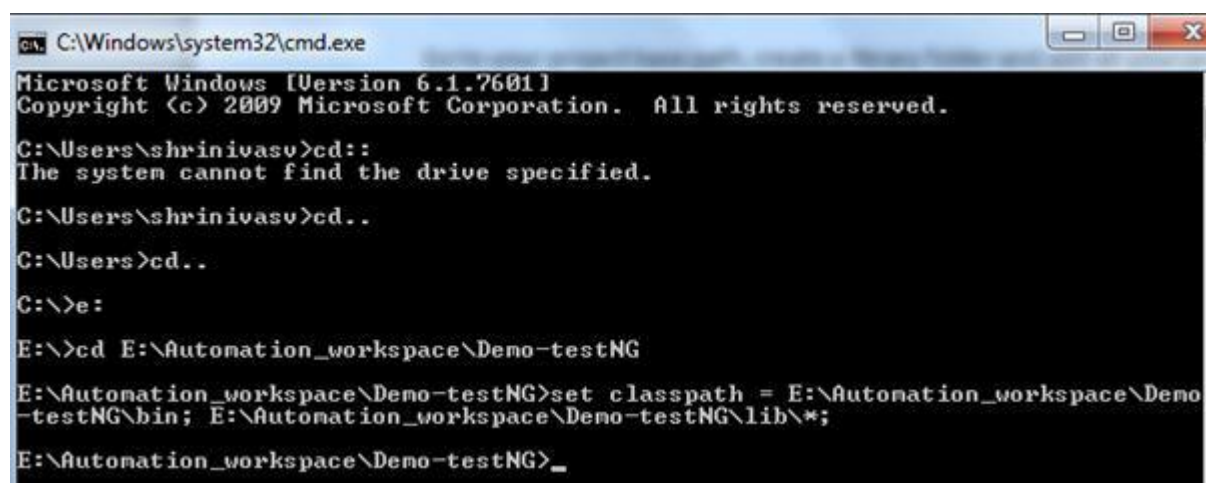
We will now see **how we can run Selenium script through command prompt**. This part has nothing to do with Jenkins. I am sharing this to give extra insights on Selenium. **Follow below steps:**

- Open your command prompt and go to your project base path



- Set class path for your script file; which means we are specifying that our binary and library files are stored in this location

```
E:\Automation_workspace\Demo-testNG > set classpath =  
E:\Automation_workspace\Demo-testNG\bin;E:\Automation_workspace\Demo-  
testNG\lib\*;
```



- Execute your testng.xml file by typing the command – **java org.testng.TestNG testng.xml**


```

E:\Automation_workspace\Demo-testNG>java org.testng.TestNG testng.xml
[TestNG] Running:
  E:\Automation_workspace\Demo-testNG\testng.xml

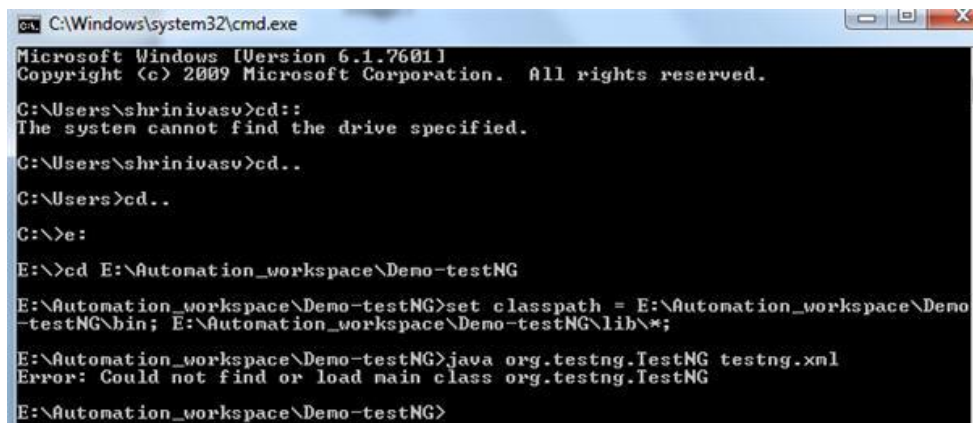
log4j:WARN No appenders could be found for logger <org.apache.http.client.protocol.RequestAddCookies>.
log4j:WARN Please initialize the log4j system properly.
Before Assertion GoogleGoogle
After Assertion GoogleGoogle Title matched

=====
Sample Suite
Total tests run: 1, Failures: 0, Skips: 0
=====

```

- When you press enter your script will start executing and you can see the test result in the UI

Sometimes while executing your script you may face error which says, "Could not find or load main class org.testng.TestNG"



```

C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\shrinivasu>cd::
The system cannot find the drive specified.

C:\Users\shrinivasu>cd..
C:\Users>cd..
C:\>e:
E:\>cd E:\Automation_workspace\Demo-testNG
E:\Automation_workspace\Demo-testNG>set classpath = E:\Automation_workspace\Demo-
-testNG\bin; E:\Automation_workspace\Demo-testNG\lib\*;
E:\Automation_workspace\Demo-testNG>java org.testng.TestNG testng.xml
Error: Could not find or load main class org.testng.TestNG
E:\Automation_workspace\Demo-testNG>

```

Then you need to close your command prompt and again set your class path as mentioned above and repeat the same steps. Your error will get resolved and the script will run.

Conclusion

Integration of Jenkins with selenium provides you to run your script each time there is any change in software code and deploy the code in a new environment. With Jenkins, you can save execution history and test reports.

In short, Jenkins is very useful when you have test cases ready and you want them to run using a single click. We can create or schedule a build to run the test cases using a batch file.