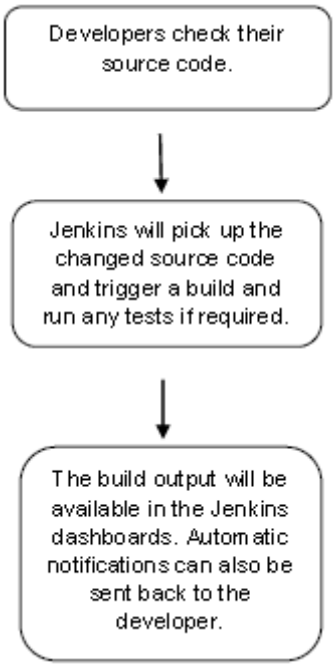
TestNG With Jenkins

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.

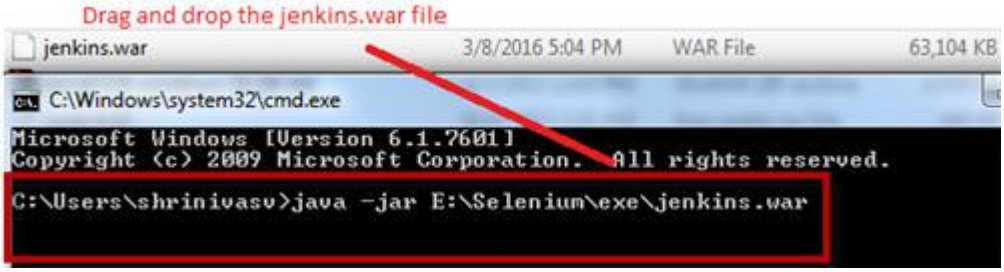
Step #1:

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:

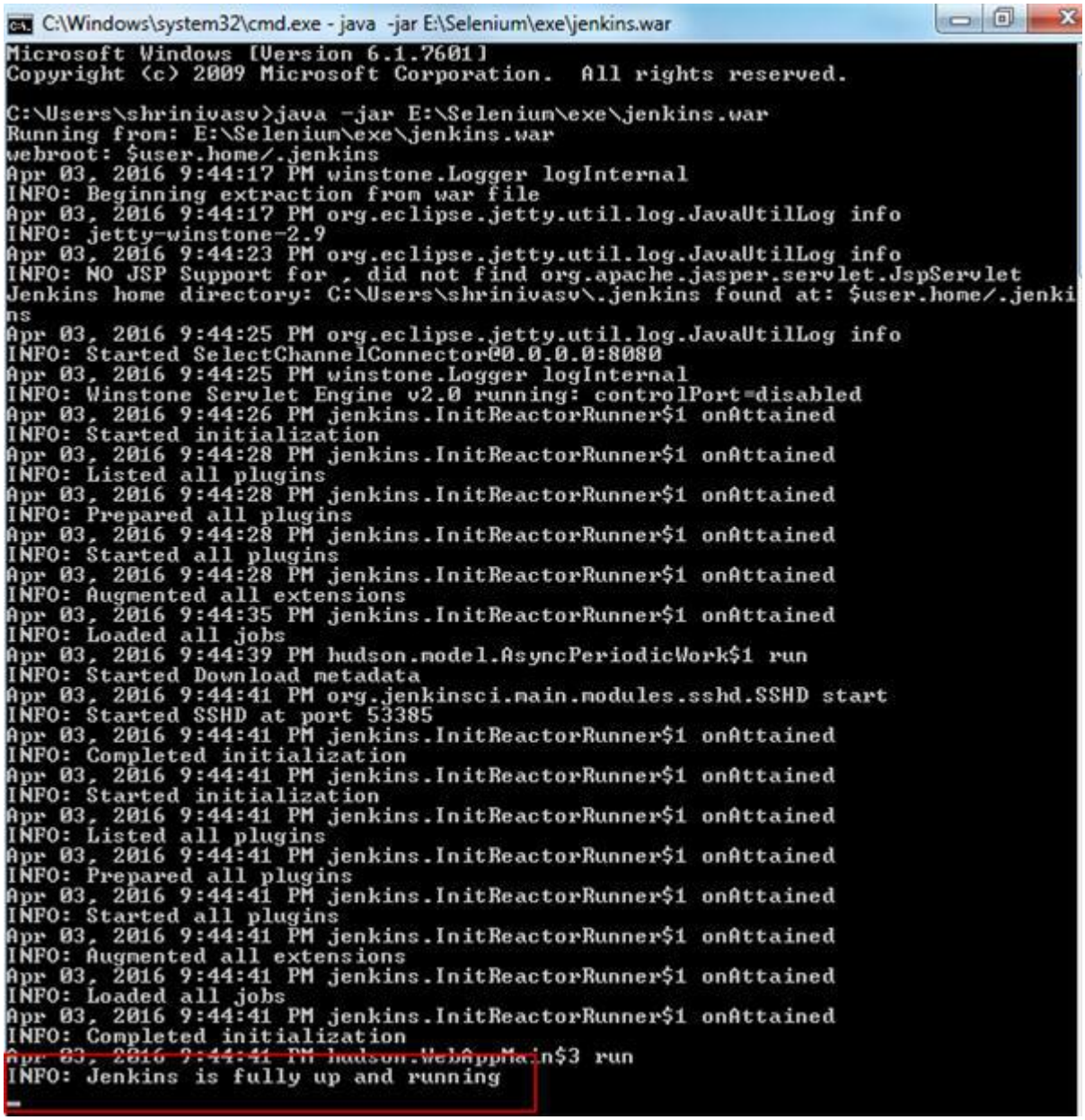
1) Open the command prompt and type ***java –jar*** and enter path of .war file

(Note: Click on any image for enlarged view)

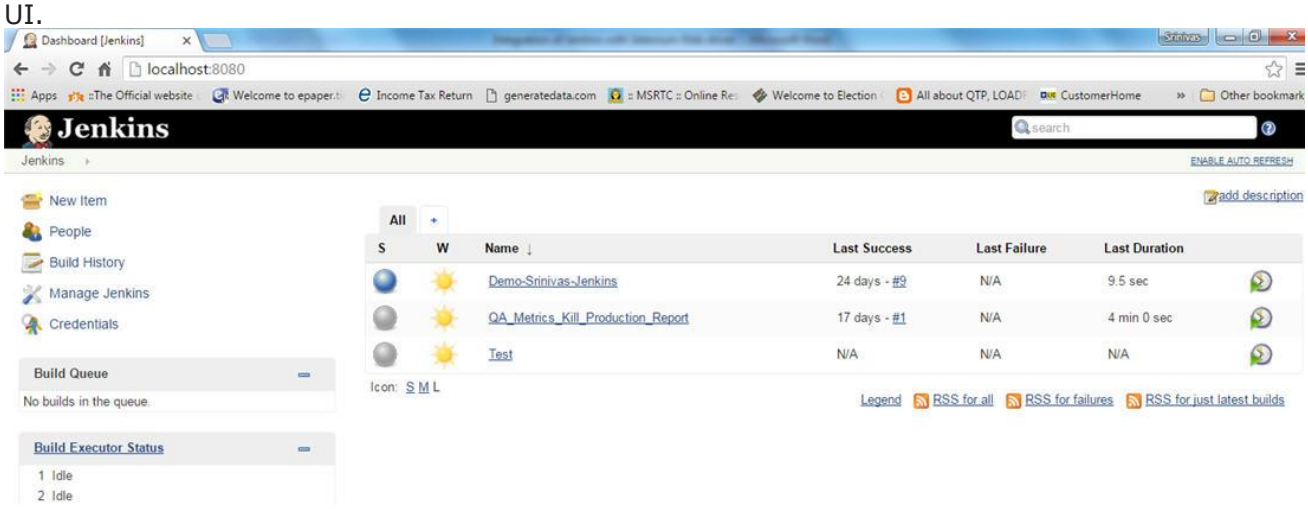


2) Press enter and check if your Jenkins.war file started to run and check the status information on the command prompt console.

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

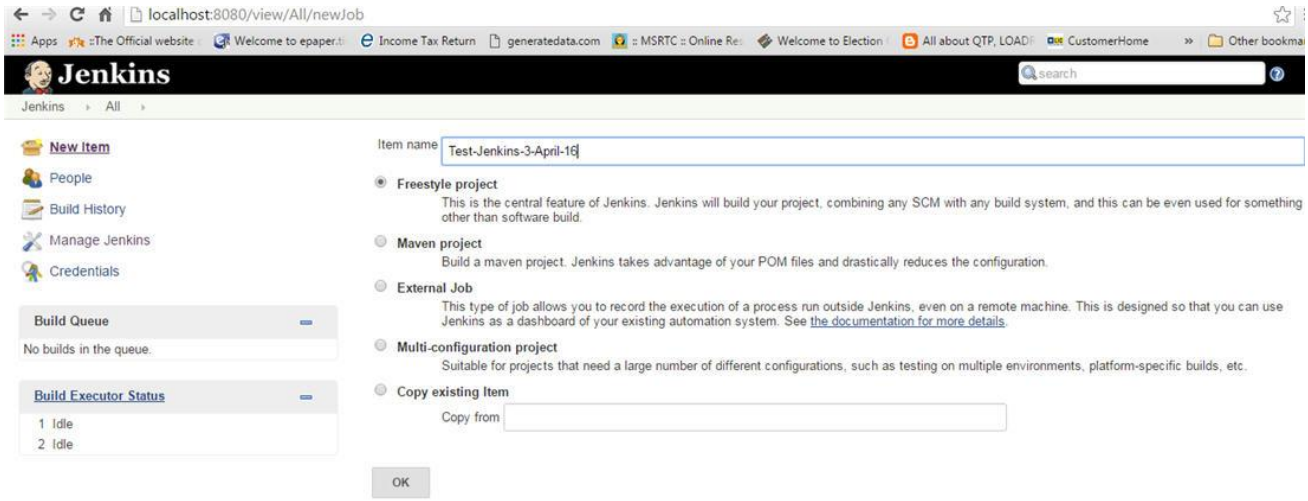


3) Now check whether your Jenkins is ready to use; by default, it uses port 8080. Type "***http://localhost:8080***" in the browser and press enter. It will show you Jenkins UI.

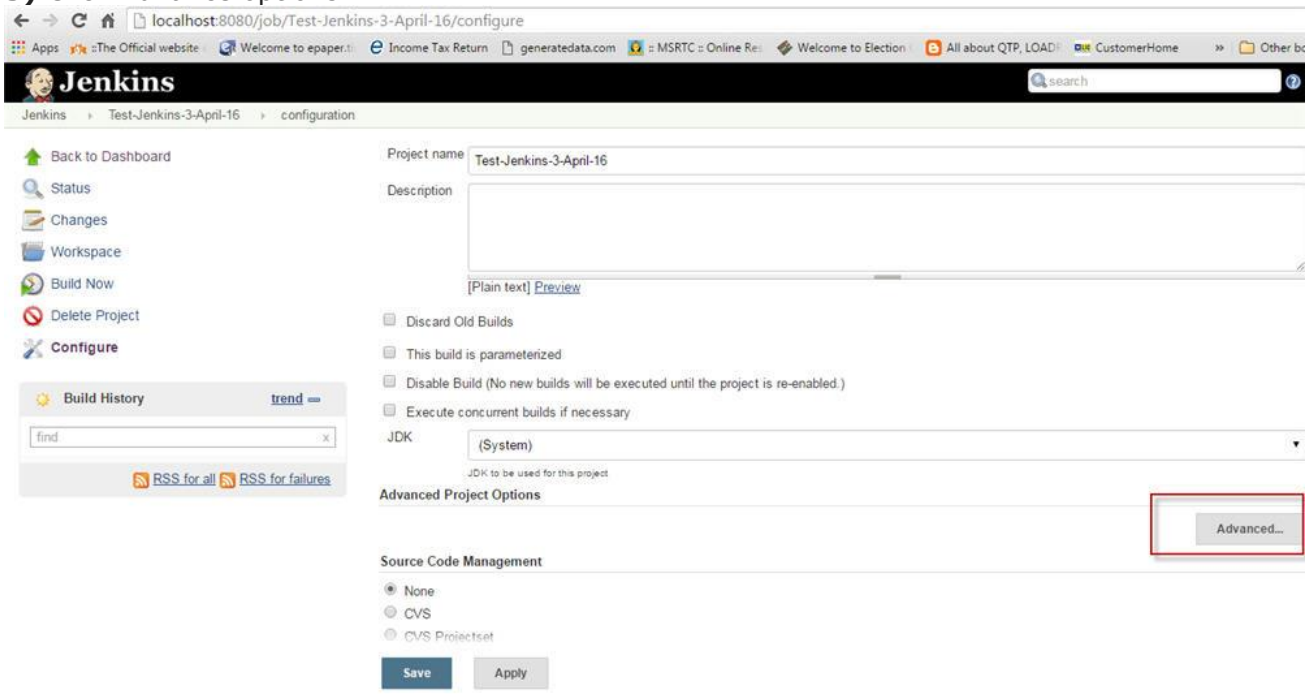


It will load the Jenkins dashboard empty by default and user has to set up the id and password .I created some Jenkins job in the above screenshot as an example and hence, it did not load empty.

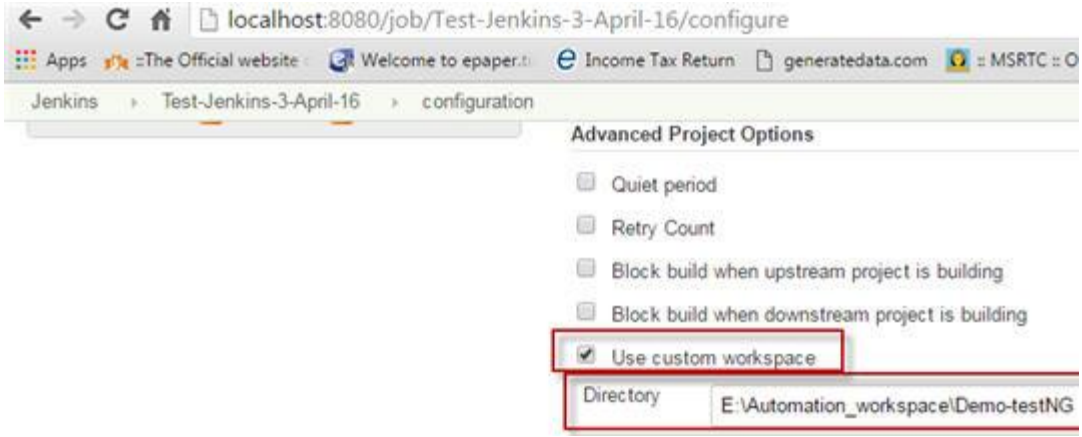
For adding the batch file follow the below steps:
1) Go to the Jenkins dashboard, create a new job in Jenkins
2) Click on new item and enter the item name and check the freestyle project radio button



3) Click Advance options



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

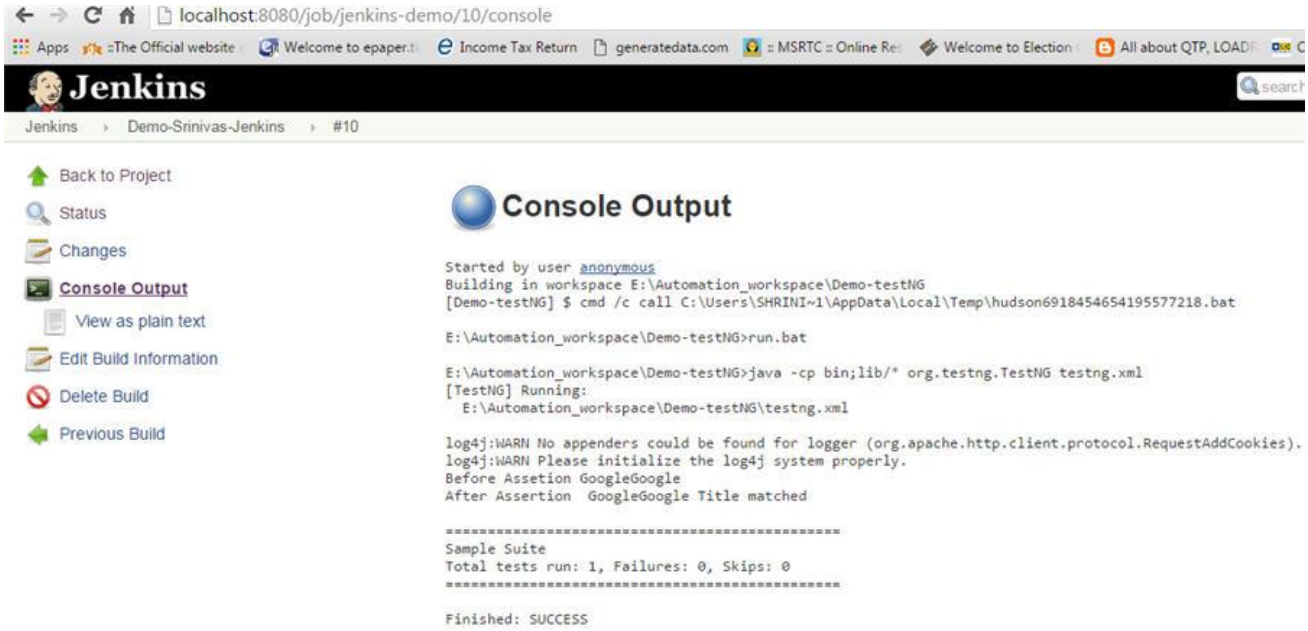


5) Then go to Build and Select option from drop down box, execute your build through Windows batch command

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:

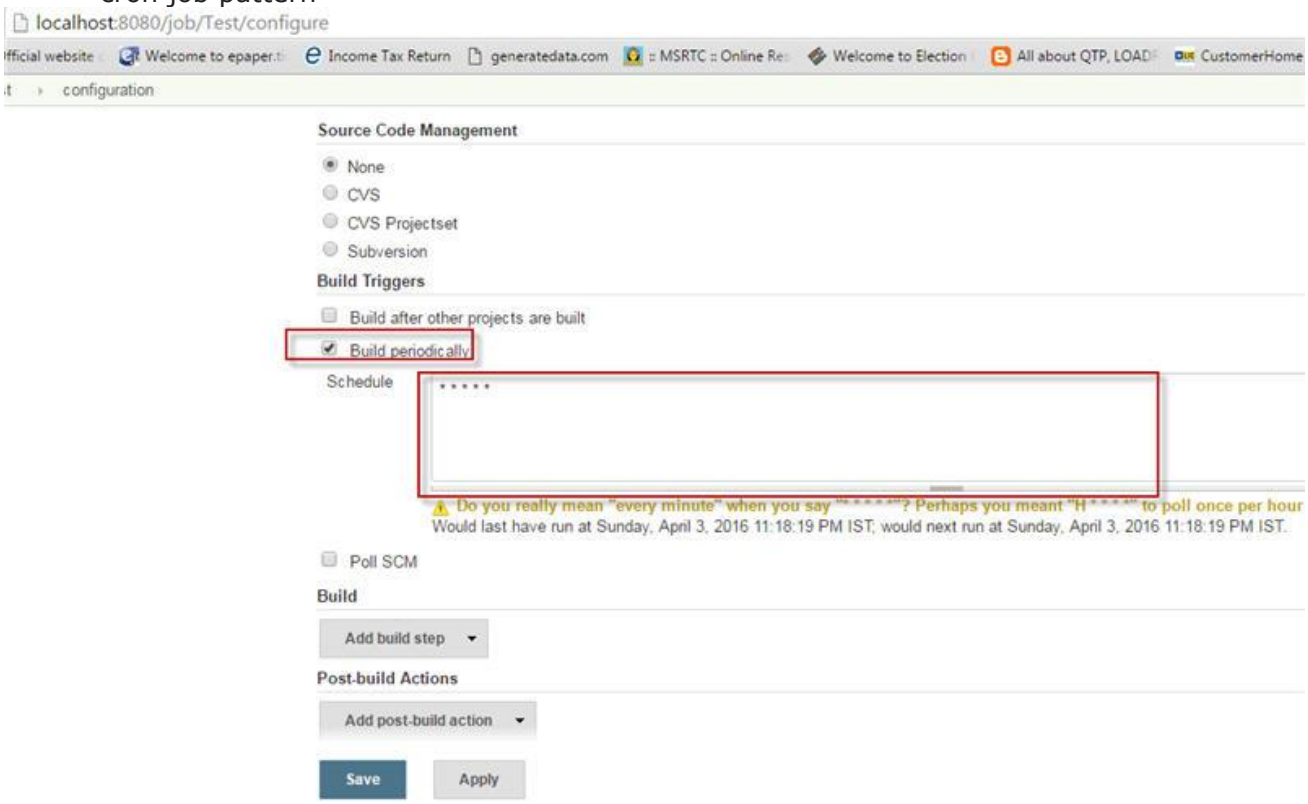
- How to start Jenkins
- How to configure Jenkins with Selenium
- Creating your batch file and executing it through Jenkins.

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 you 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



- 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

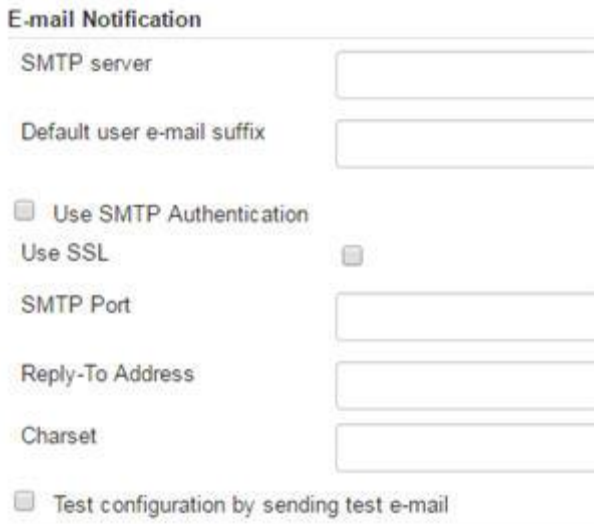
There is no manual intervention. After scheduling the script, it will run on the scheduled time.

How to add email notifications

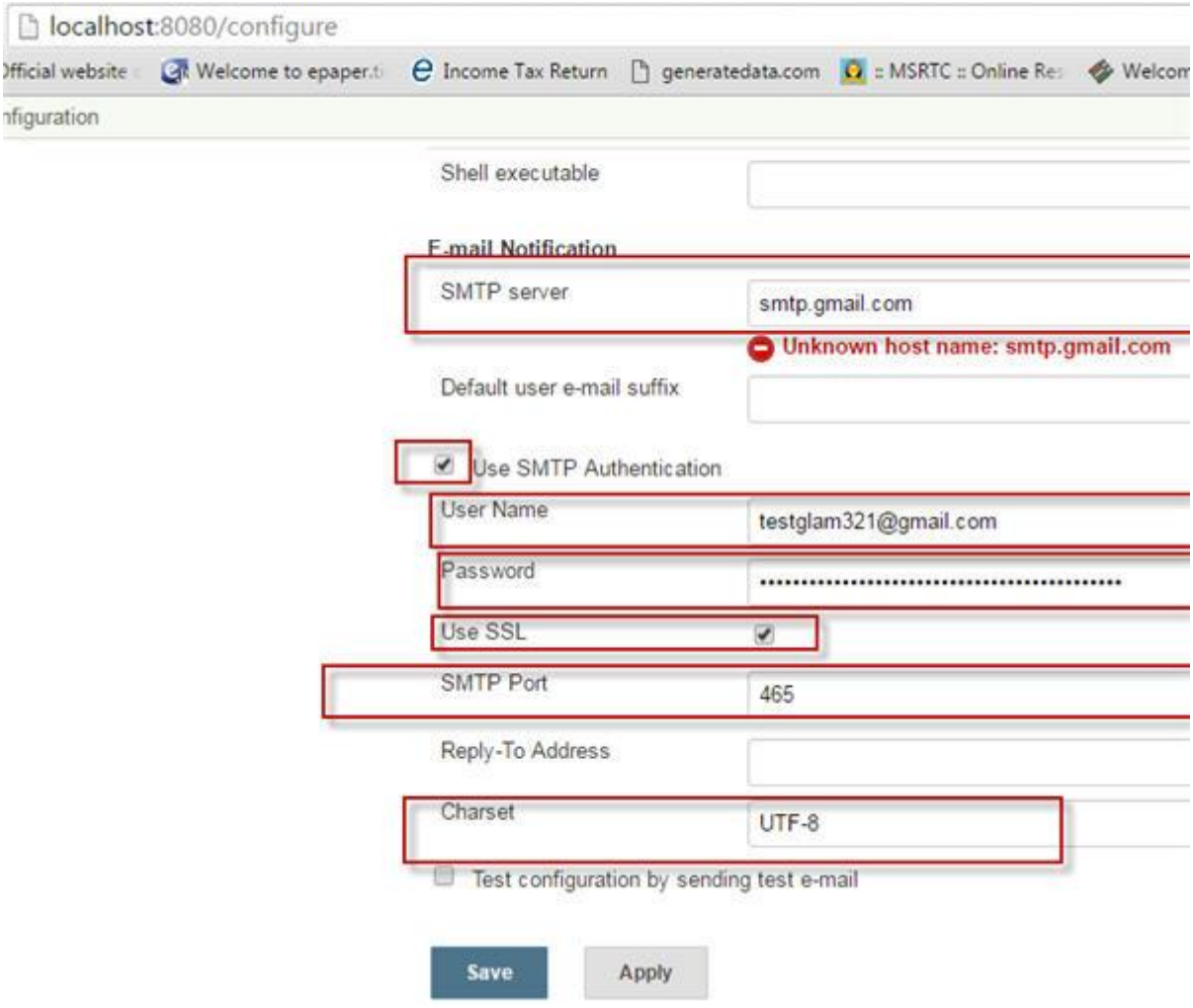
Next, we will cover how to add email notifications.

Refer the below steps:

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



- 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



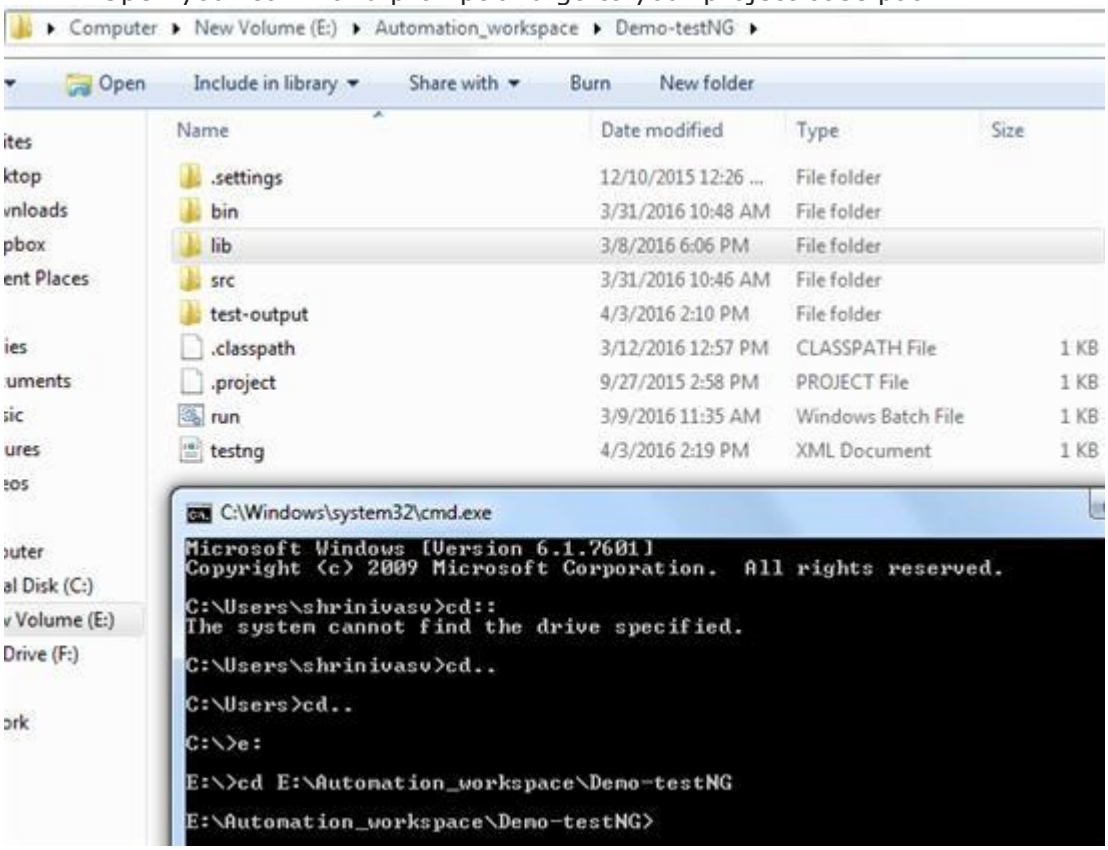
- Check your email configuration settings by clicking on Test configuration button.
- So, whenever the build passes or fails you will get the email notification.

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.

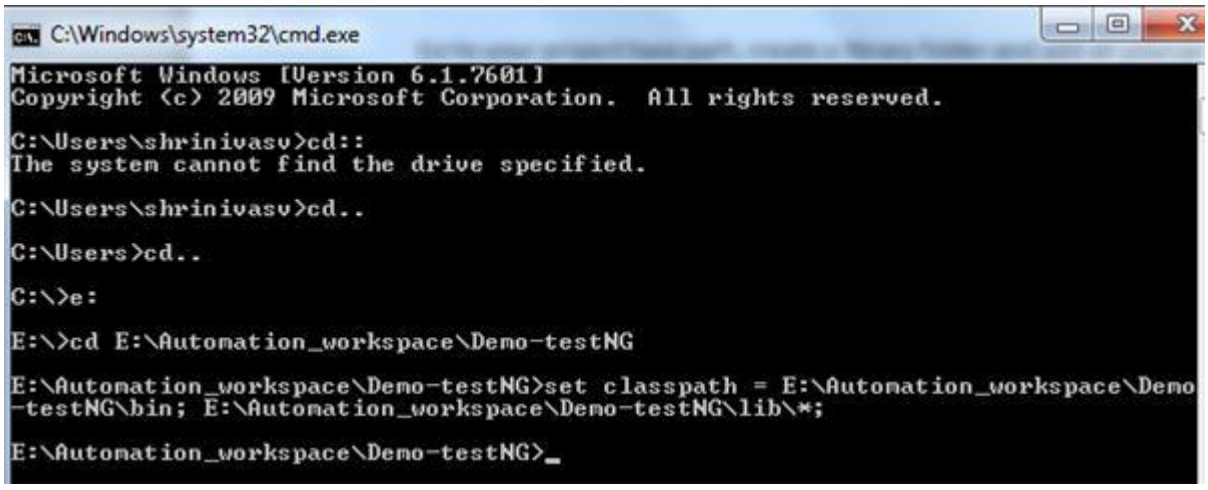
Follow below steps:

- 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***

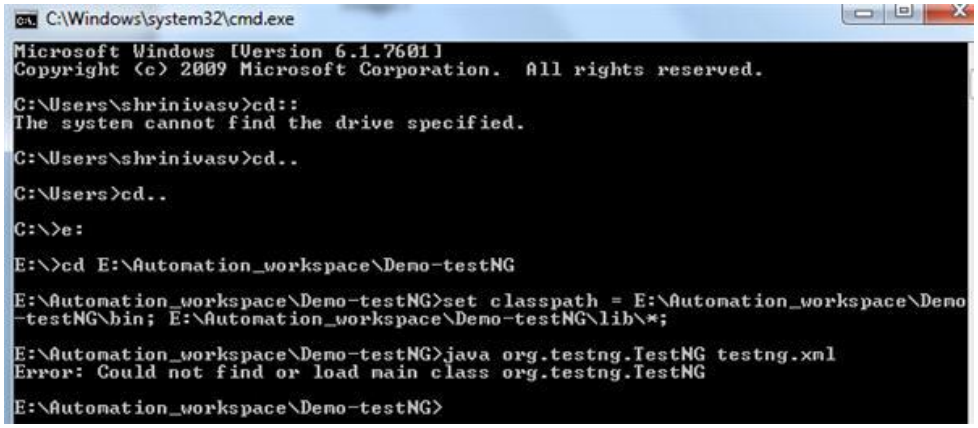
```
E:\Automation_workspace\Demo-testNG>java org.testng.TestNG testng.xml
[TestNG] Running:
  E:\Automation_workspace\Demo-testNG\testng.xml

log4j:WARN No appenders could be found for logger (org.apache.http.client.protoc
ol.RequestAddCookies).
log4j:WARN Please initialize the log4j system properly.
Before Assetion 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"



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.