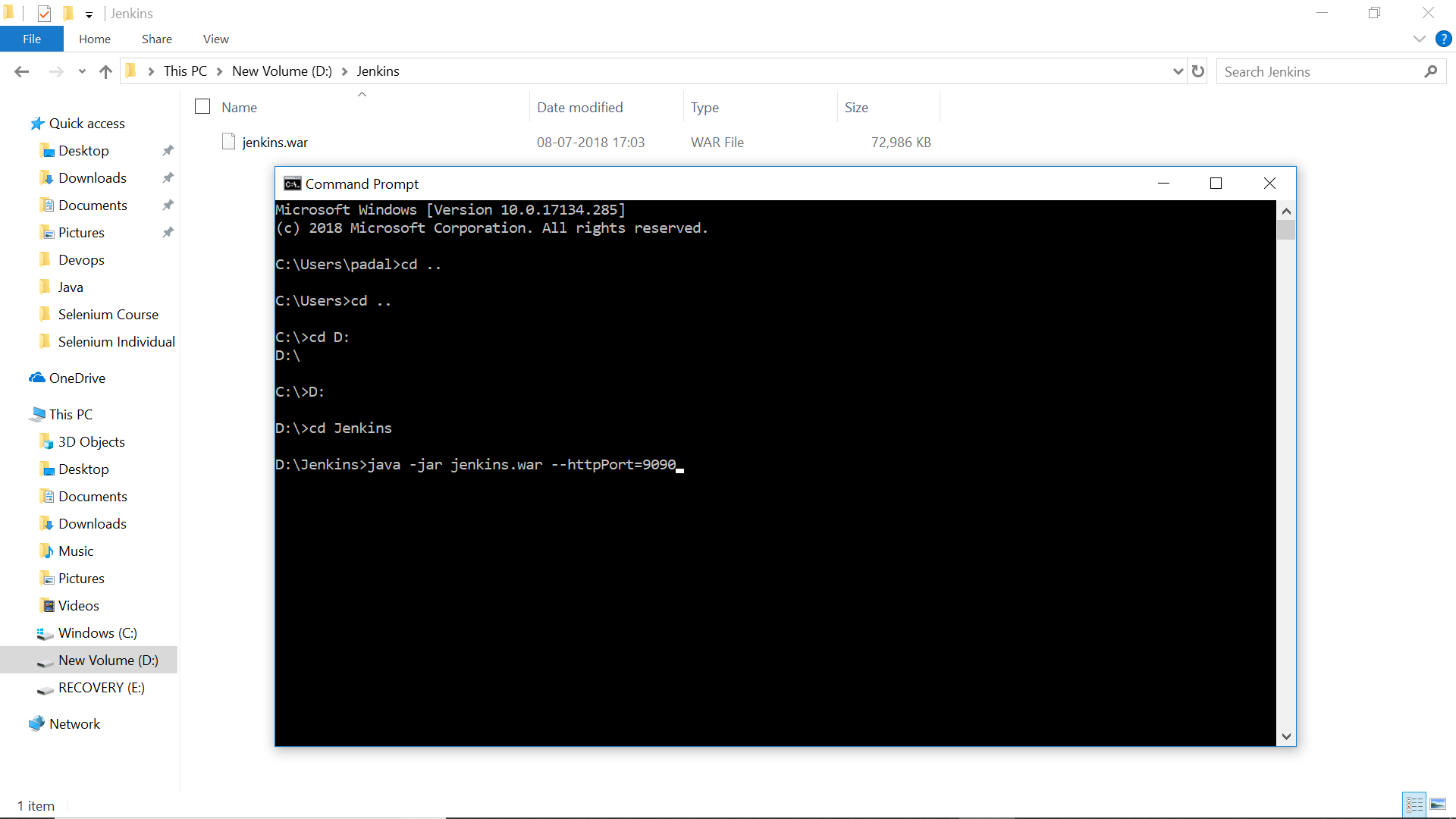
**Running Tests From Jenkins:**

In CMD, go to the folder where the Jenkins war file is located, and run the following command.

java -jar jenkins.war --httpPort=9090

I have downloaded the war file to Jenkins folder in D drive so I am running the command from there.

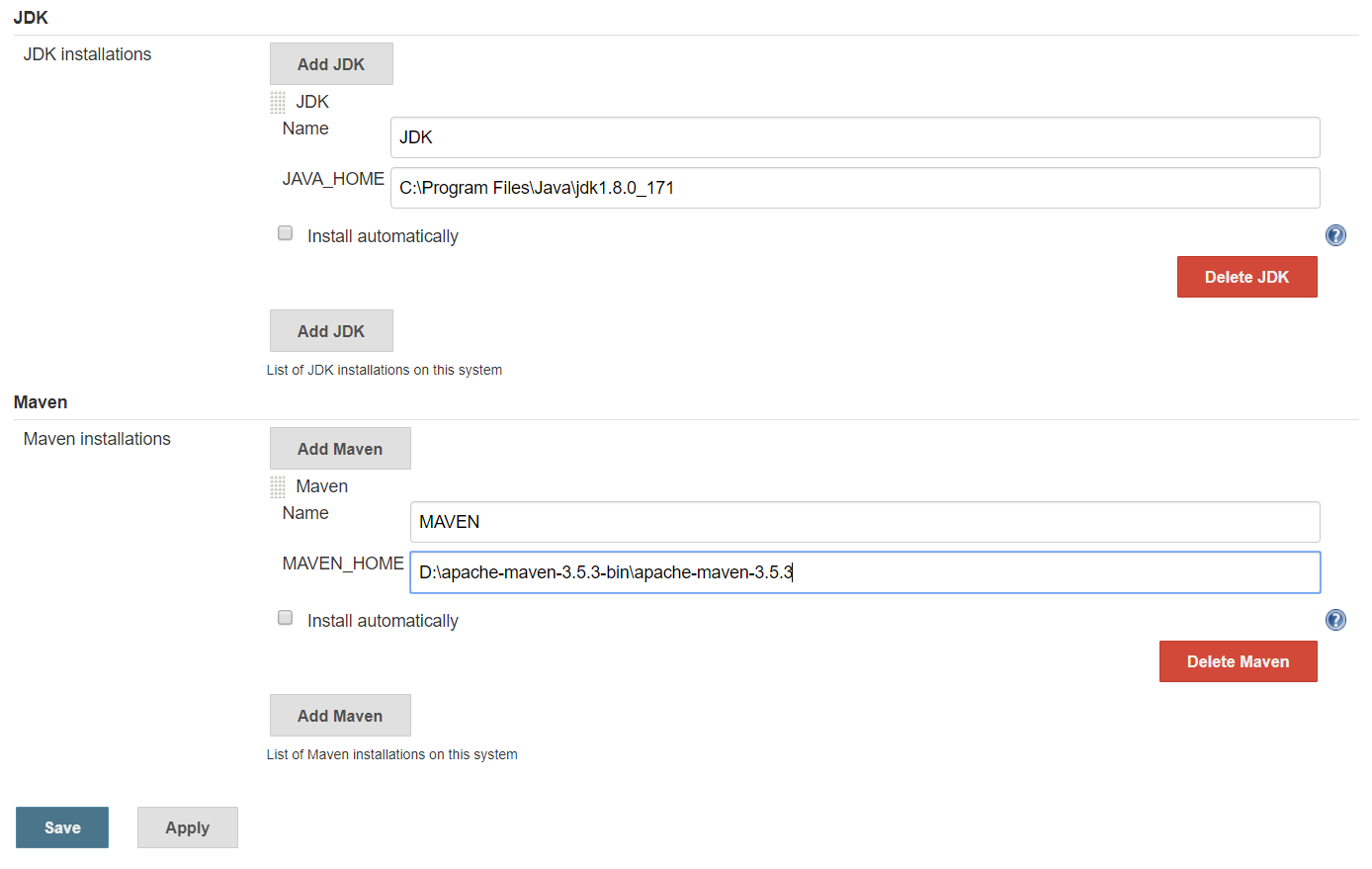


Open Jenkins page by typing the following in browser.

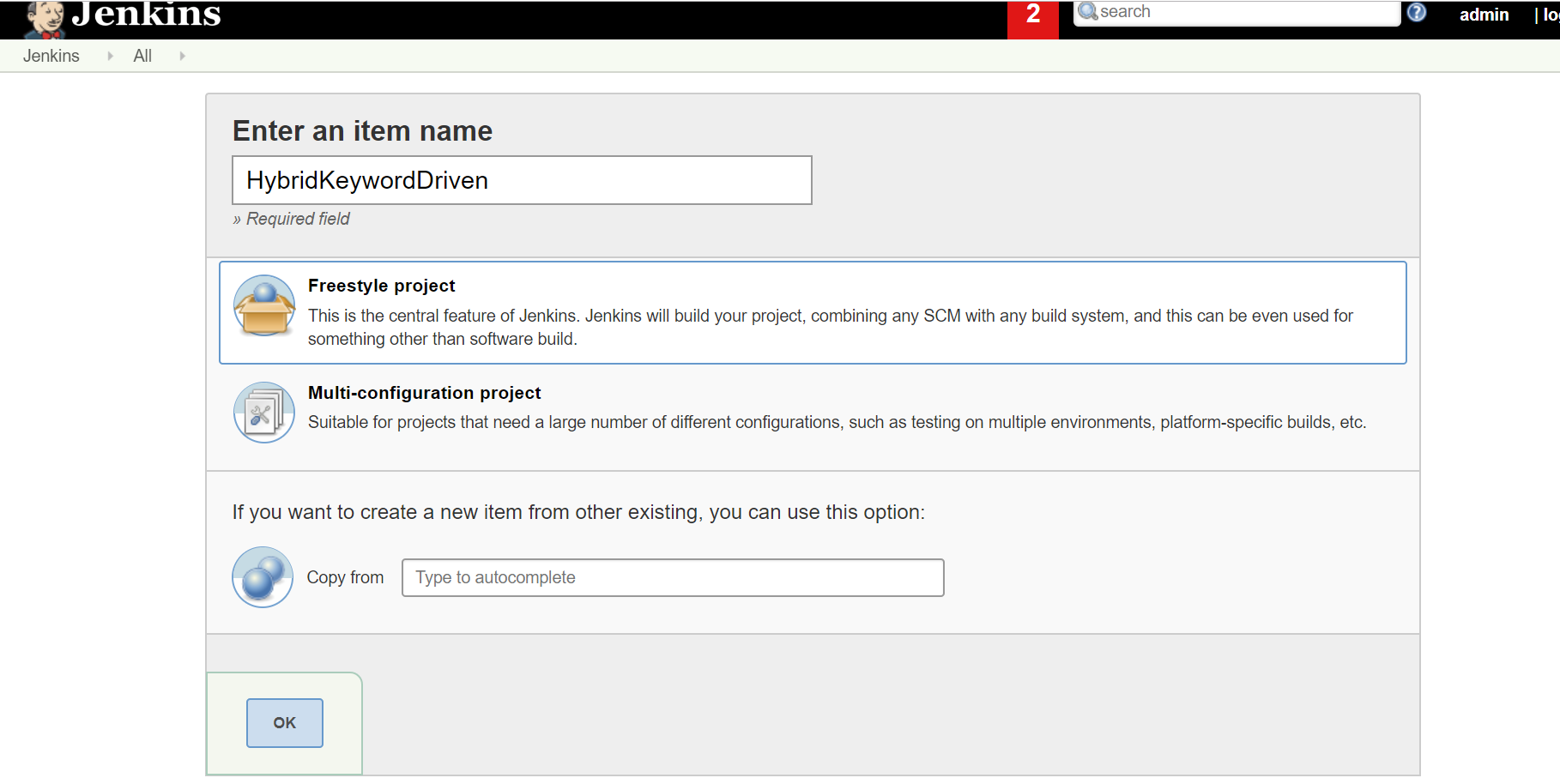
<http://localhost:9090>

Login to Jenkins with the username and password set earlier.

If you haven’t setup the JDK and maven paths already, go to Manage Jenkins and Global Tool Configuration and set up the paths for JDK and Maven.



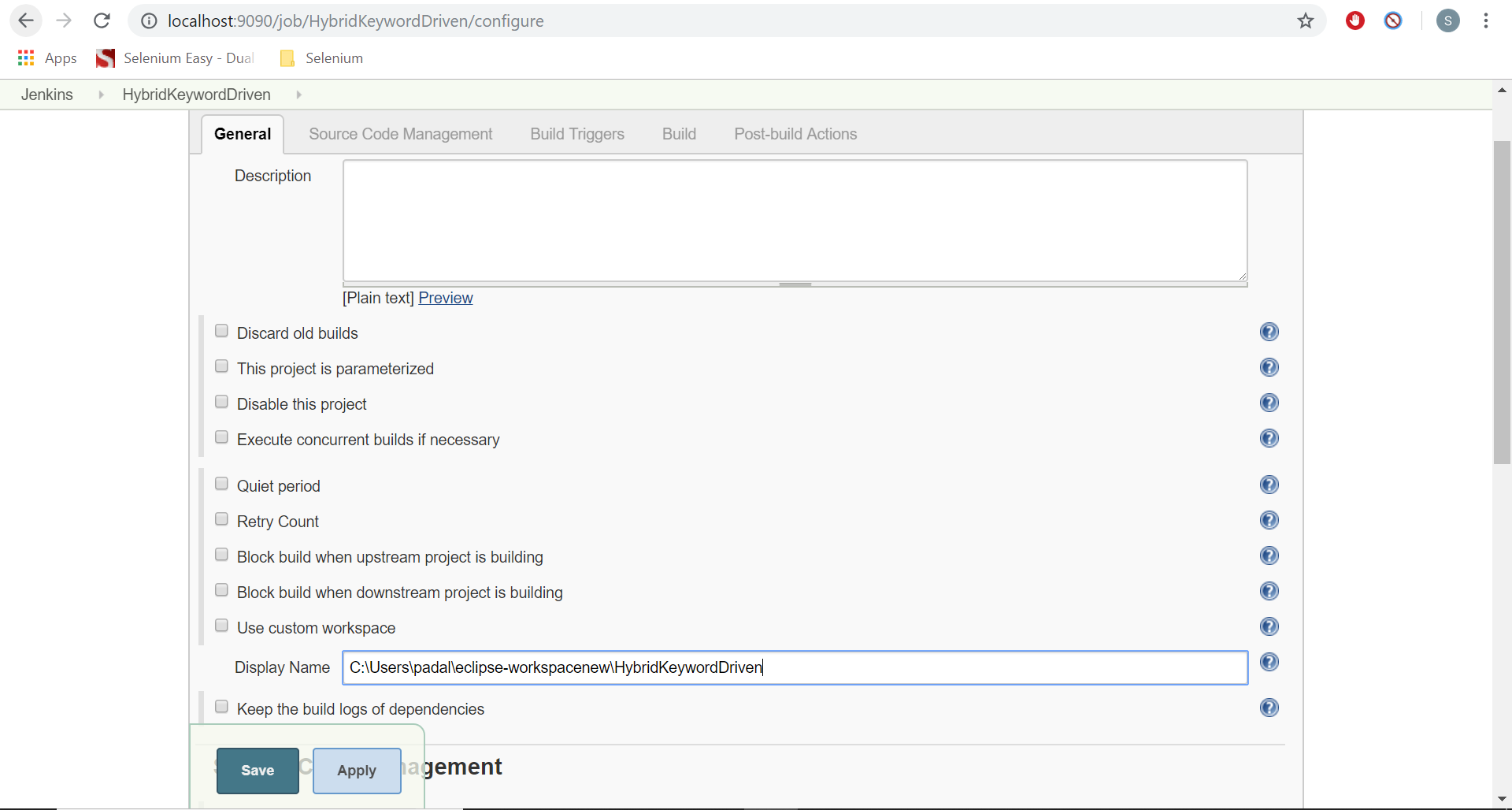
Now go to Jenkins home page and set up a new job.



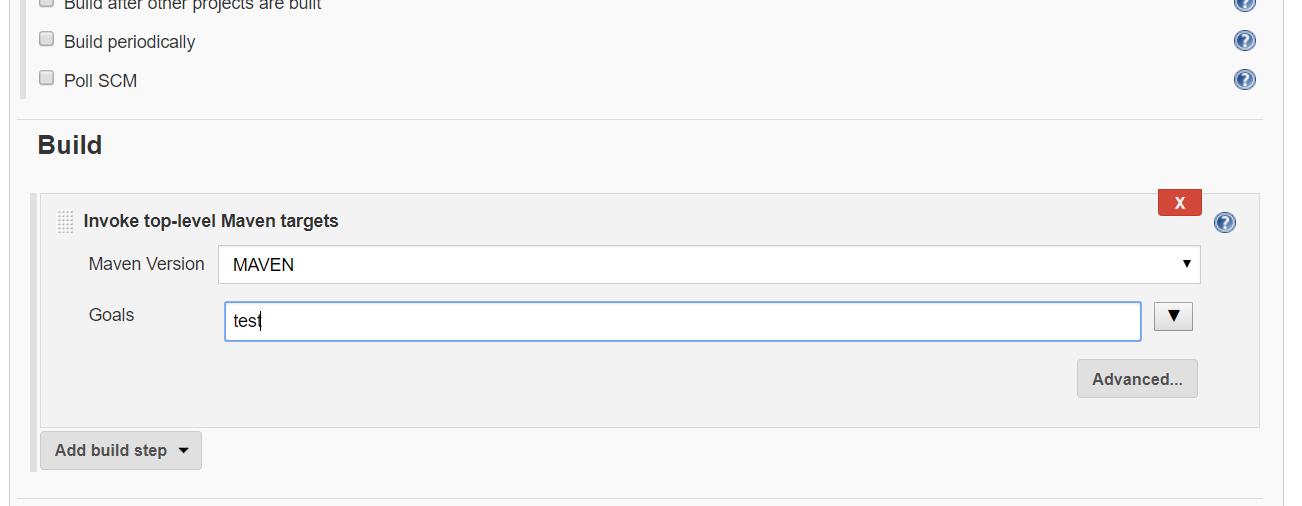
I am creating a project with name HybridKeywordDriven.

In order to set up the project which is in our local machine, under GENERAL tab click on Advanced button and select “Custom Workspace” and enter the url of the project. We need to give the url until where the POM.xml is located.

You can get the directory where the pom.xml is, by righ clicking on the pom.xml and click on properties in Eclipse.

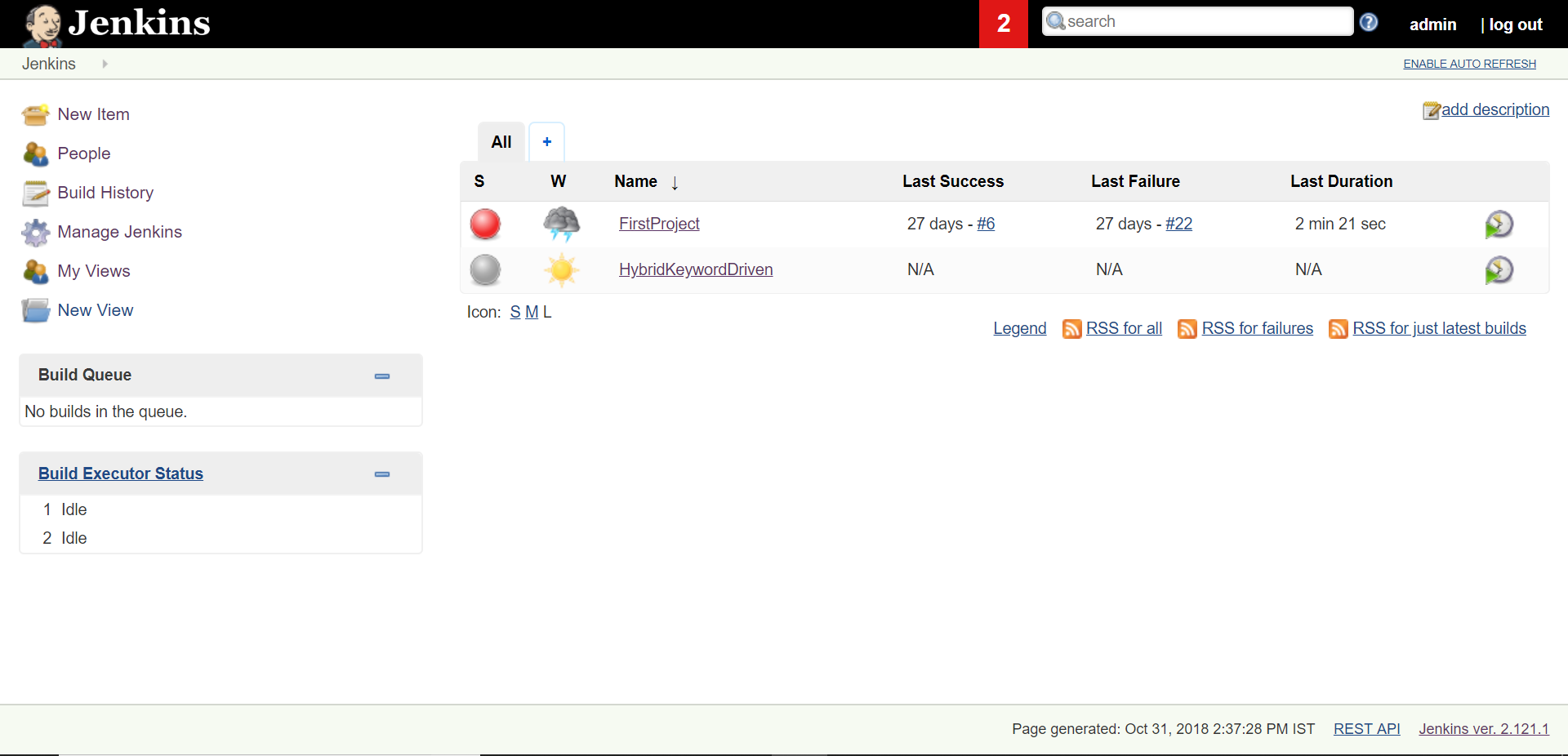


Under “Build” section click on “Add Build Step” and select “Invoke Top-Level Maven Targets”. You can select the maven version here and enter the goal as “test”. You don’t need to provide mvn command here.



Now save the job.

The job should show on the home page of Jenkins.



Now on the main system start the selenium standalone server.

java -jar selenium-server-standalone-3.9.0.jar -role hub

Start the selenium standalone servers on nodes.

java -jar selenium-server-standalone-3.9.0.jar -role webdriver -hub http://192.168.56.101:4444/grid/register -port 5567 -host 192.168.56.102

Change the registration url and standalone server versions and ip address fo the nodes as per your configuration.

Now start the Jenkins job. It should start running the tests parallelly.