

//You are given a project to demonstrate how to build (execute) a Jenkins job.

- How to build Jenkins job with Cucumber

Development Environment:

- JRE: OpenJDK Runtime Environment 11.0.2
- Maven path setup
- Eclipse IDE for Enterprise Java Developers v2019-03 (4.11.0)
- Jenkin war file

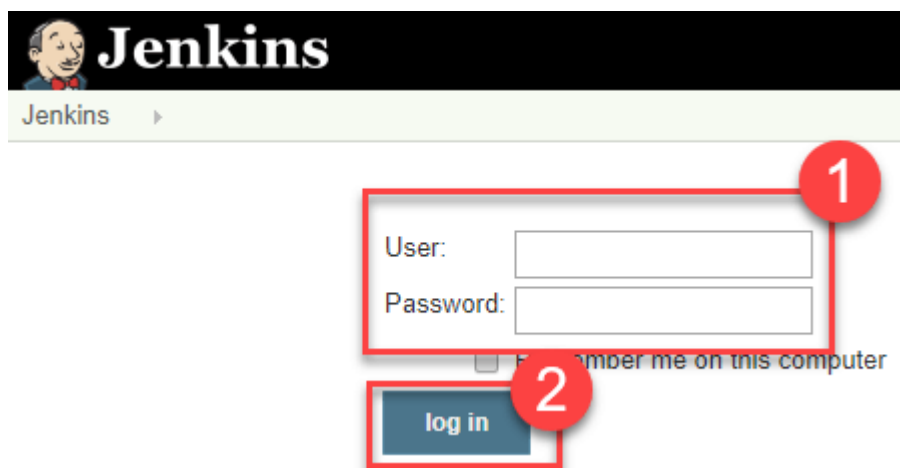
This guide has two subsections, namely:

5.15.1 Executing through Jenkins

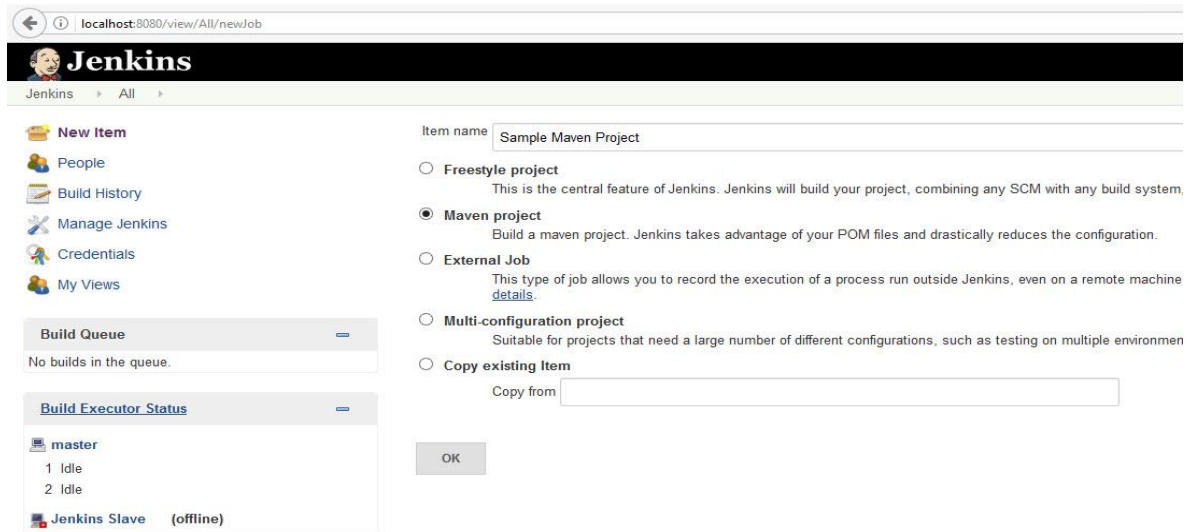
5.15.2 Pushing the code to your GitHub repositories

Step 5.15.1: Executing through Jenkins

- Login into Jenkins



- Create a new job as Maven Project



- After then click on the created projected from the Jenkins Dashboard and click on “Configure”.
- In Job Configuration, go to the “Build” section and provide the path of pom.xml which has to be executed.
- And click on “Save” and click on “Build Now” to start the execution on Jenkins.



- After Executing the build, you can review the result as below:

```
Results :

Tests run: 8, Failures: 0, Errors: 0, Skipped: 0

[JENKINS] Recording test results
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 13.848 s
[INFO] Finished at: 2017-01-28T22:26:26+05:30
[INFO] Final Memory: 15M/36M
[INFO] -----
[JENKINS] Archiving F:\Selenium\mavenparameterize\pom.xml to com.easy\mavenparameterize\0.0.1-SNAPSHOT\mavenparameterize-0.0.1-SNAPSHOT.pom
channel stopped
Finished: SUCCESS
```

Step 5.15.2: Pushing the code to GitHub repositories

Open your command prompt and navigate to the folder where you have created your files

```
cd <folder  
path>
```

Initialize your repository using the following command:

```
git init
```

Add all the files to your git repository using the following command:

```
git add .
```

Commit the changes using the following command:

```
git commit .
```

-m "Changes have been committed."

Push the files to the folder you initially created using the following command:

```
git push -u
```

```
origin master
```