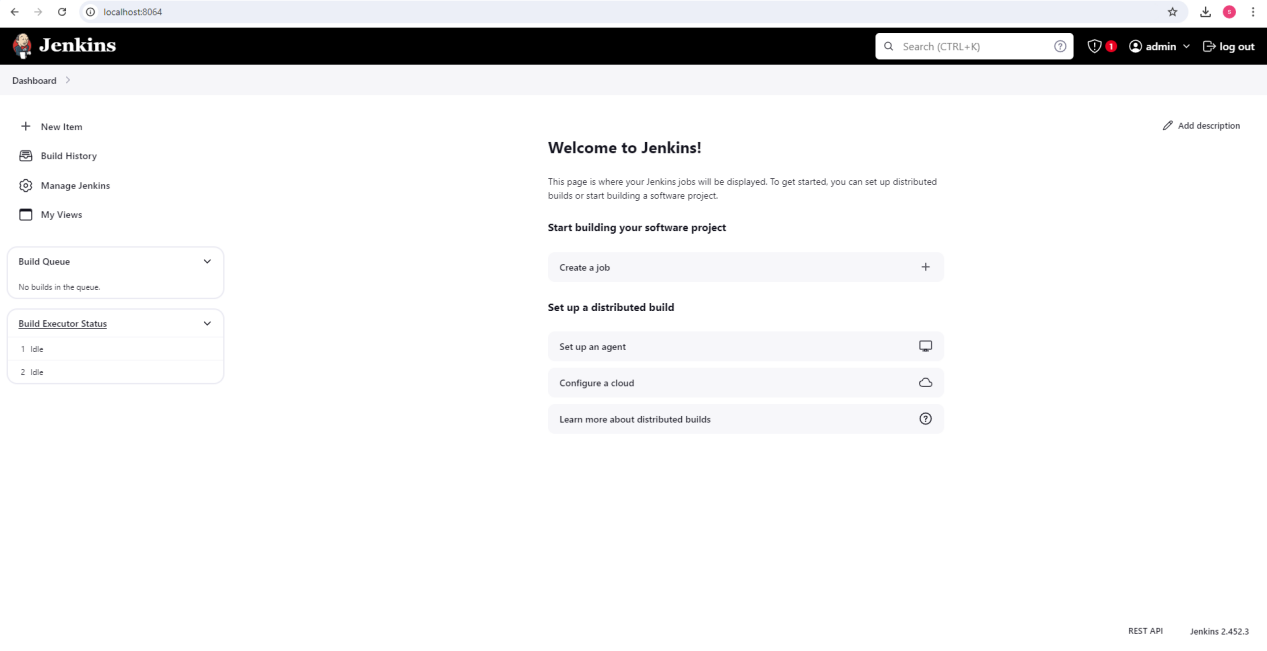
**PLUGINS IN JENKINS**

**STEP 1: Go to jenkins URL using <http://localhost:8064/>**

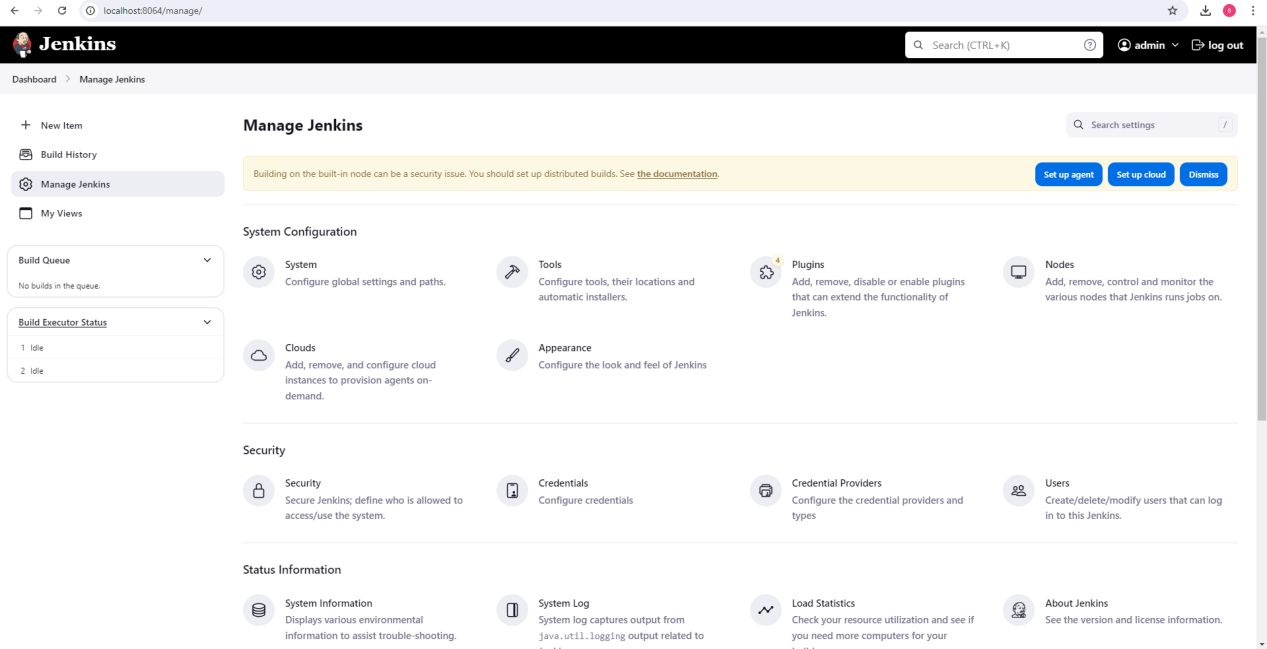
**STEP 2:**  Give username as admin and password from program files/jenkins.err

**STEP 3:** after login-> dashboard

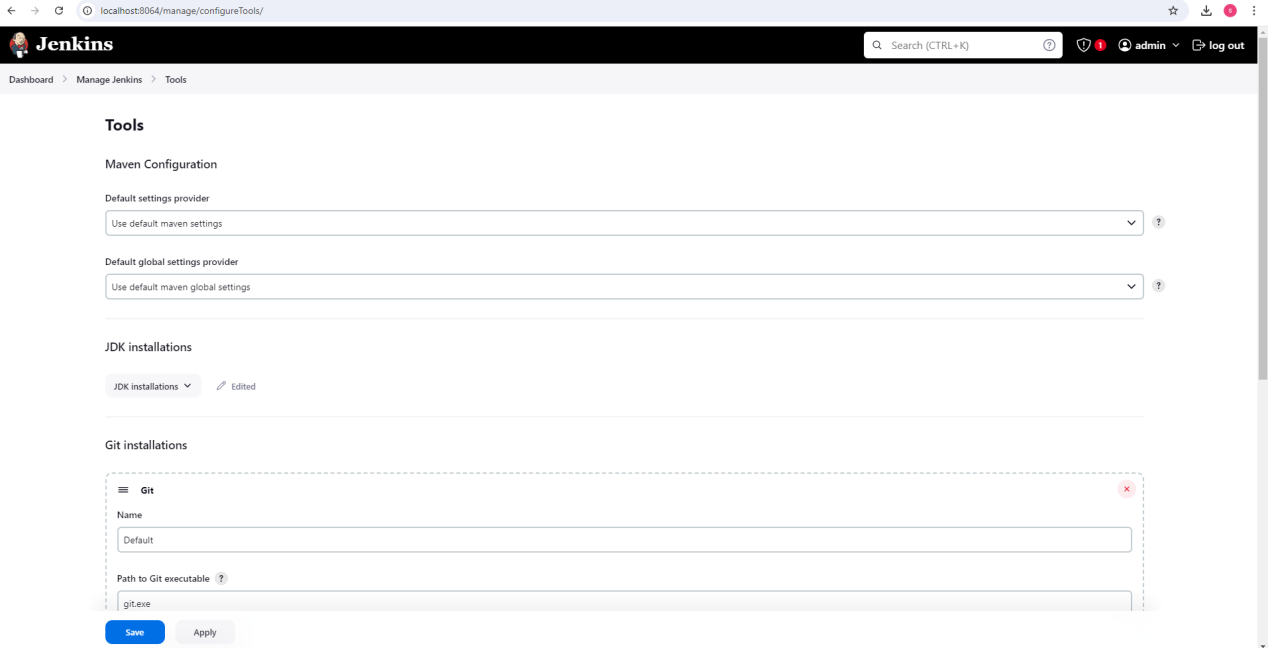
****

**Followed by other steps**

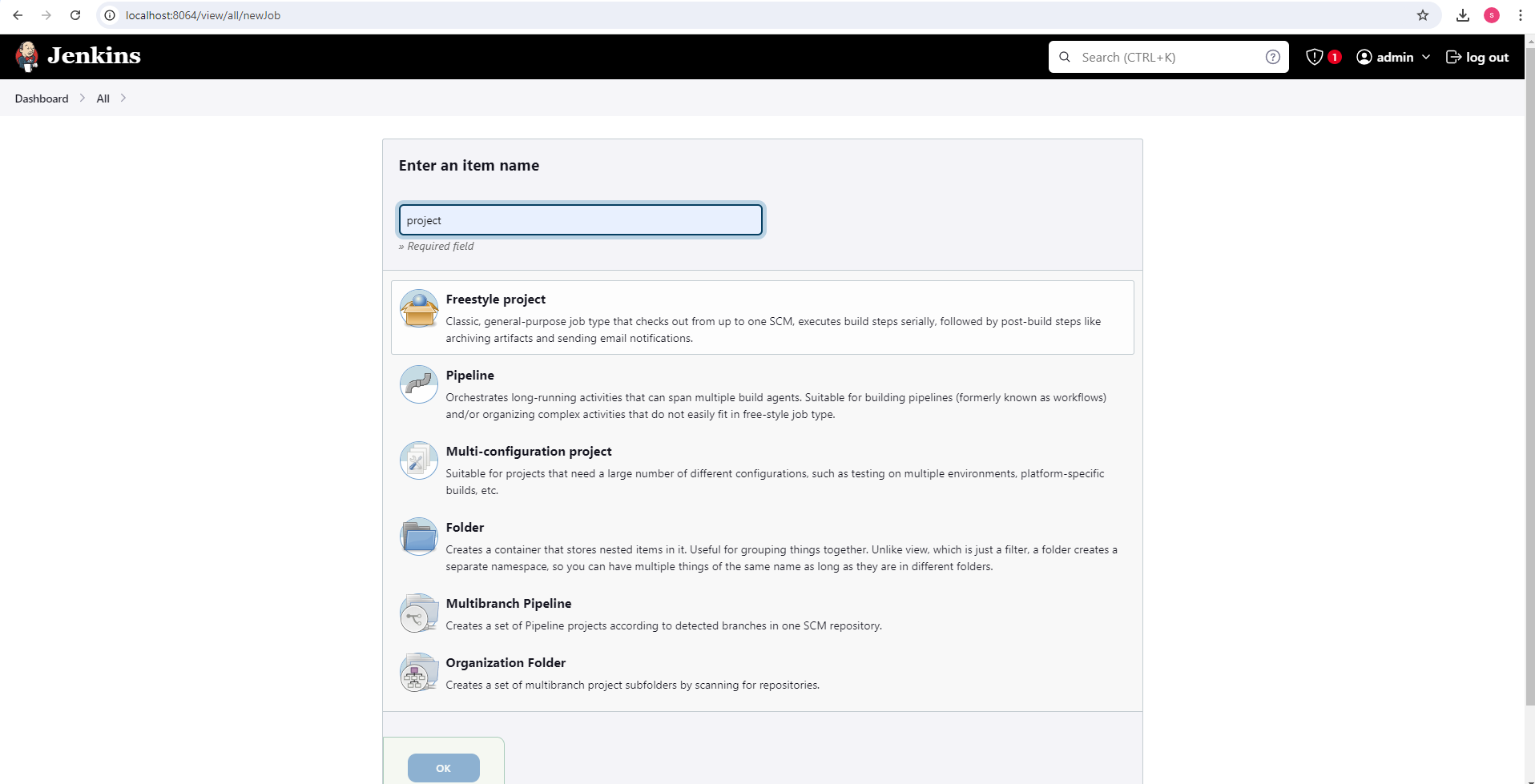
**->Dashboard-> manage jenkins(Manage Plugins from the Jenkins dashboard)**

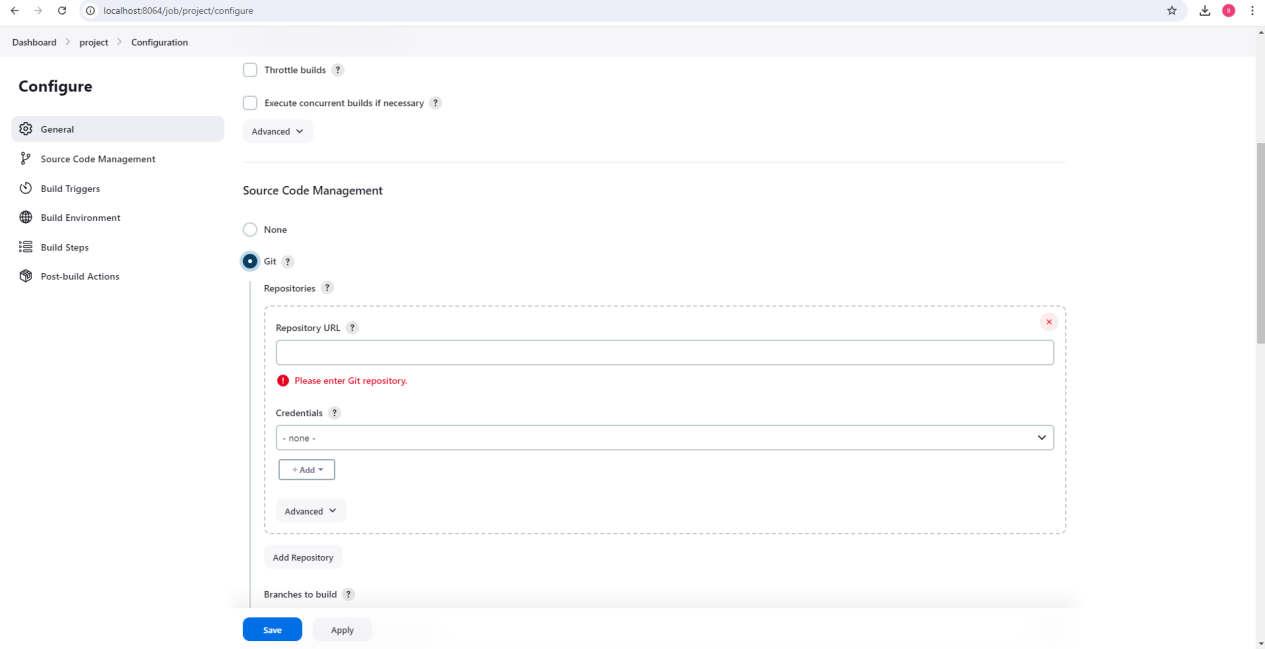
****

**Manage jenkins->tools**

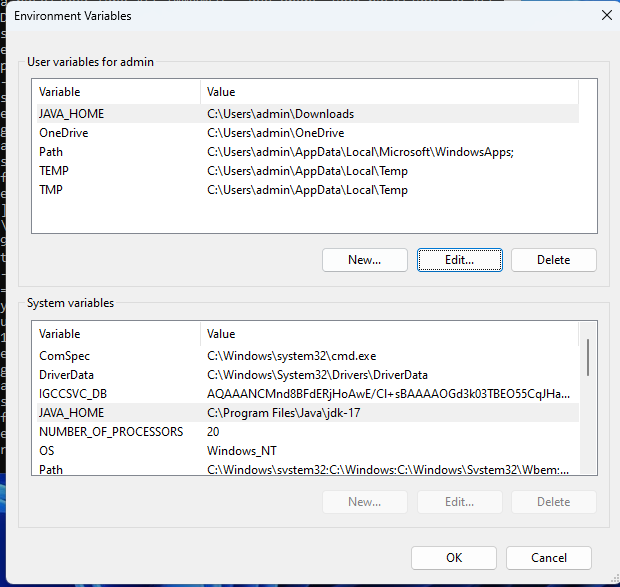
****

**Tools-> new time-> give file name ->freestyle project->ok**

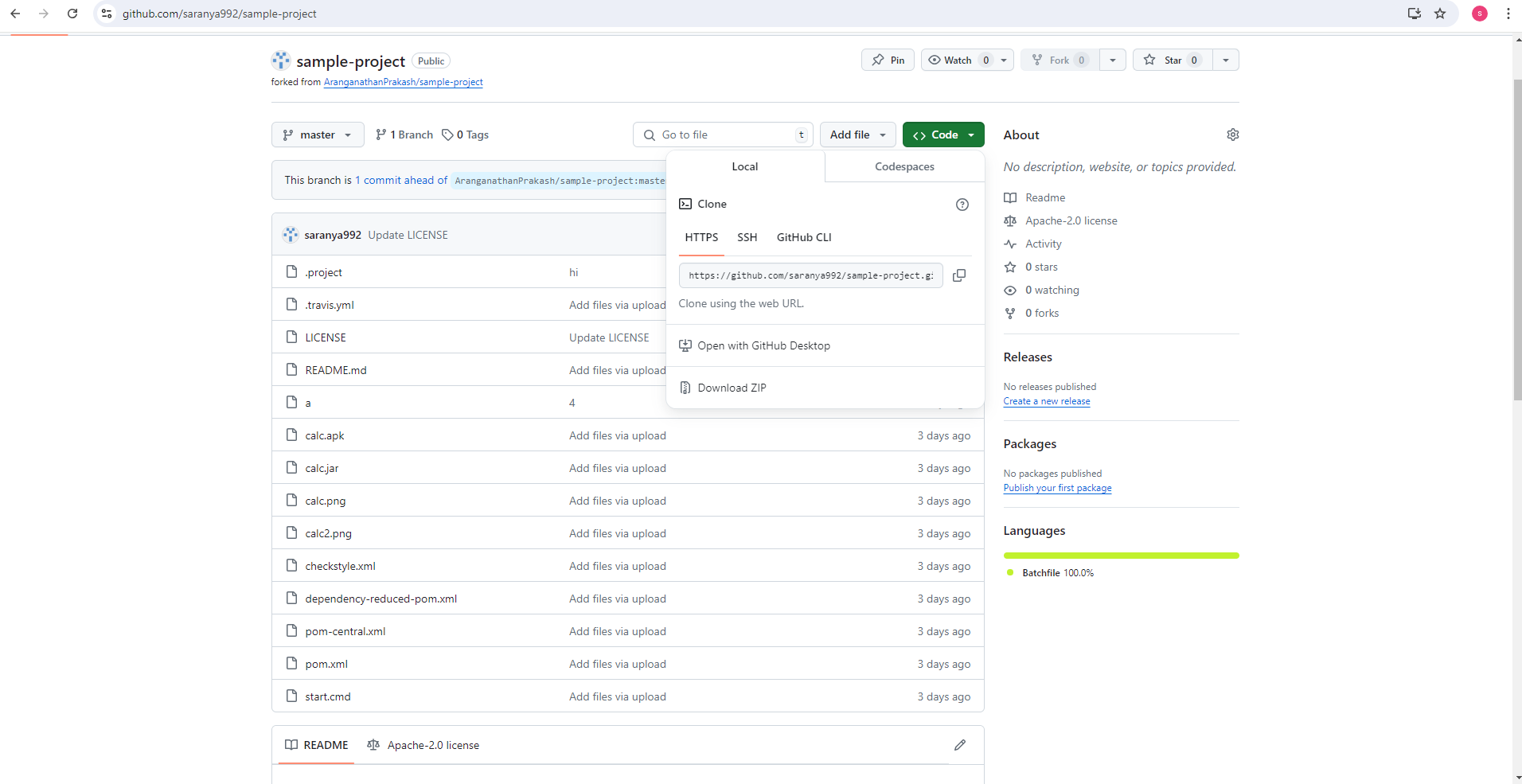
****

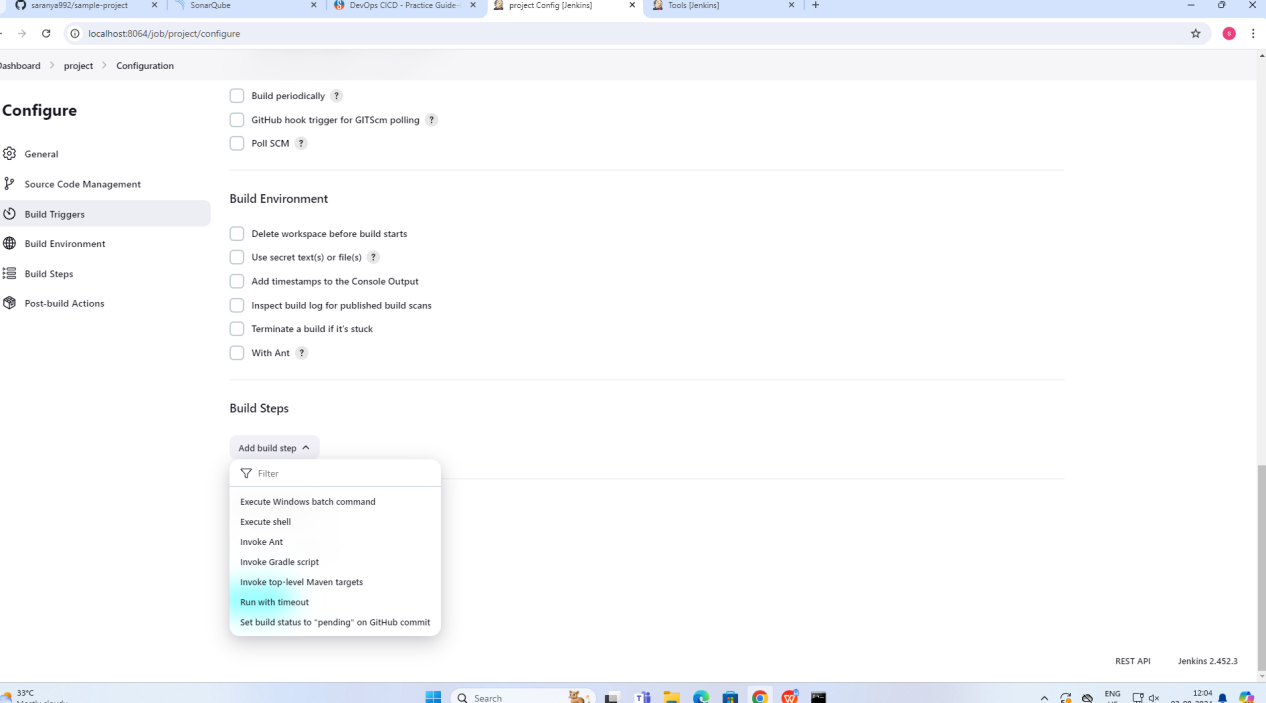
****

**Copy url from environment variable (JAVA\_HOME) upto jdk (use this link to install jdk in jenkins)**

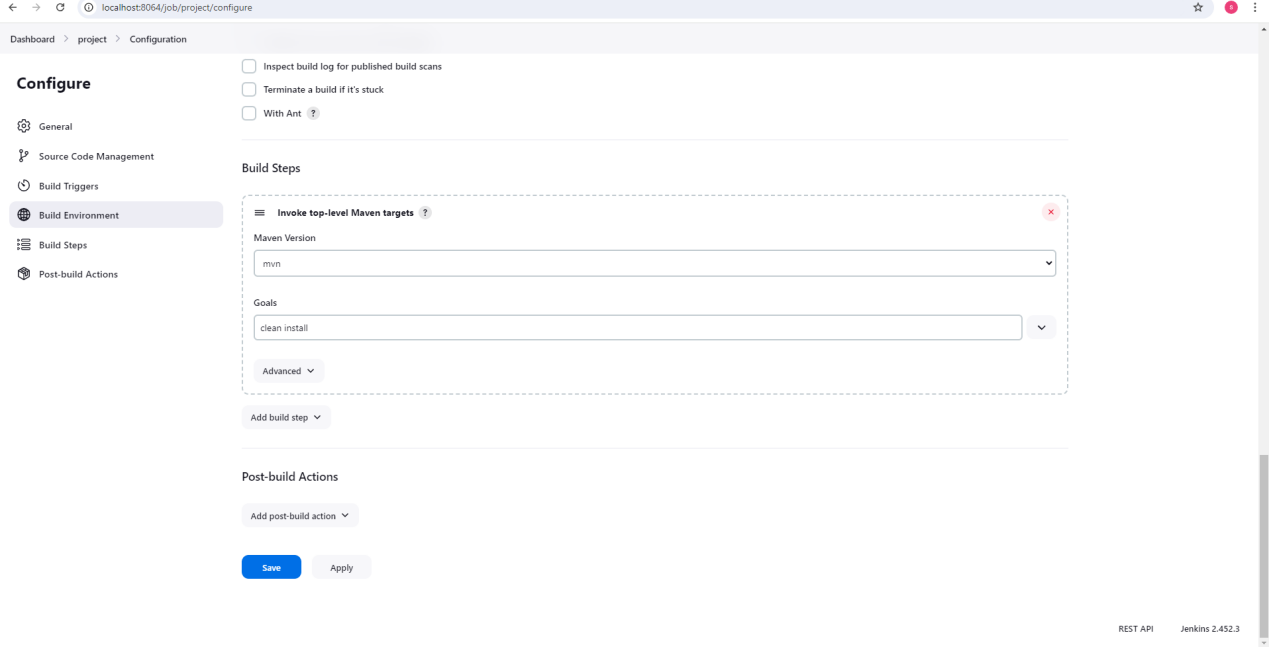
****

**Copy the github repository URl from git hub and paste and save**

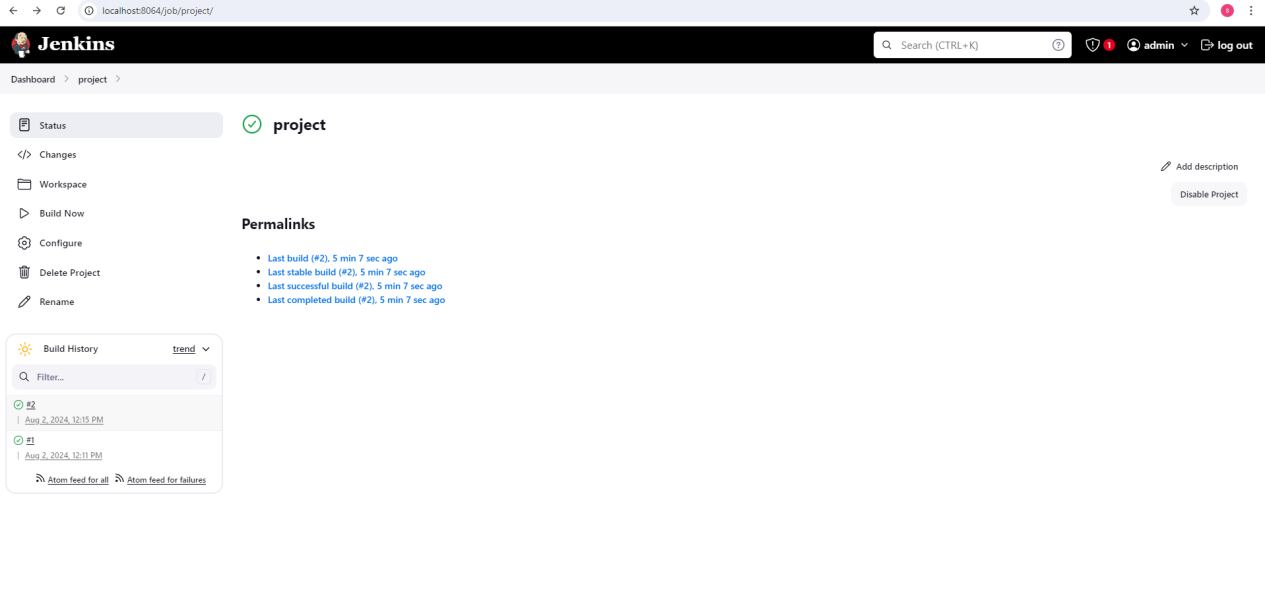


**Again go back to dashboard and select and open project we created ->Select configure option -> Source Code Management->git?->copy and paste sample project repository and save(refer previous screenshot)-> select build setup->select invoke top level maven**

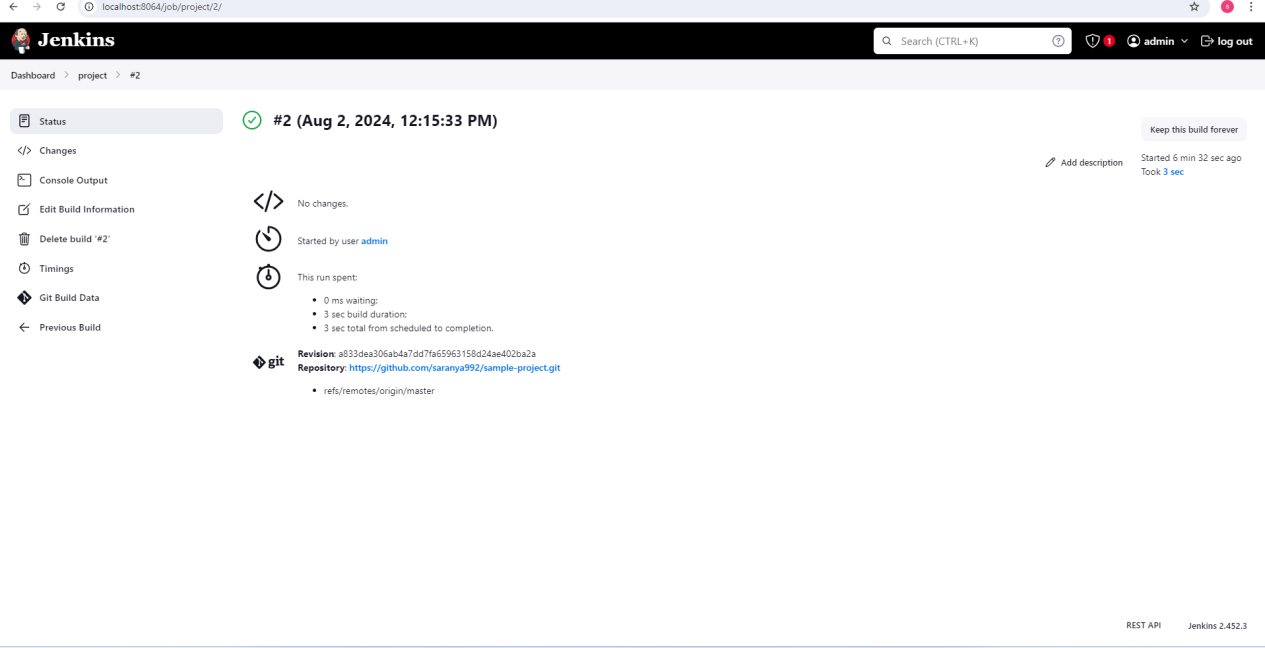
**->give any name to mavenlike Maven\_HOME->goals->type clean install->**



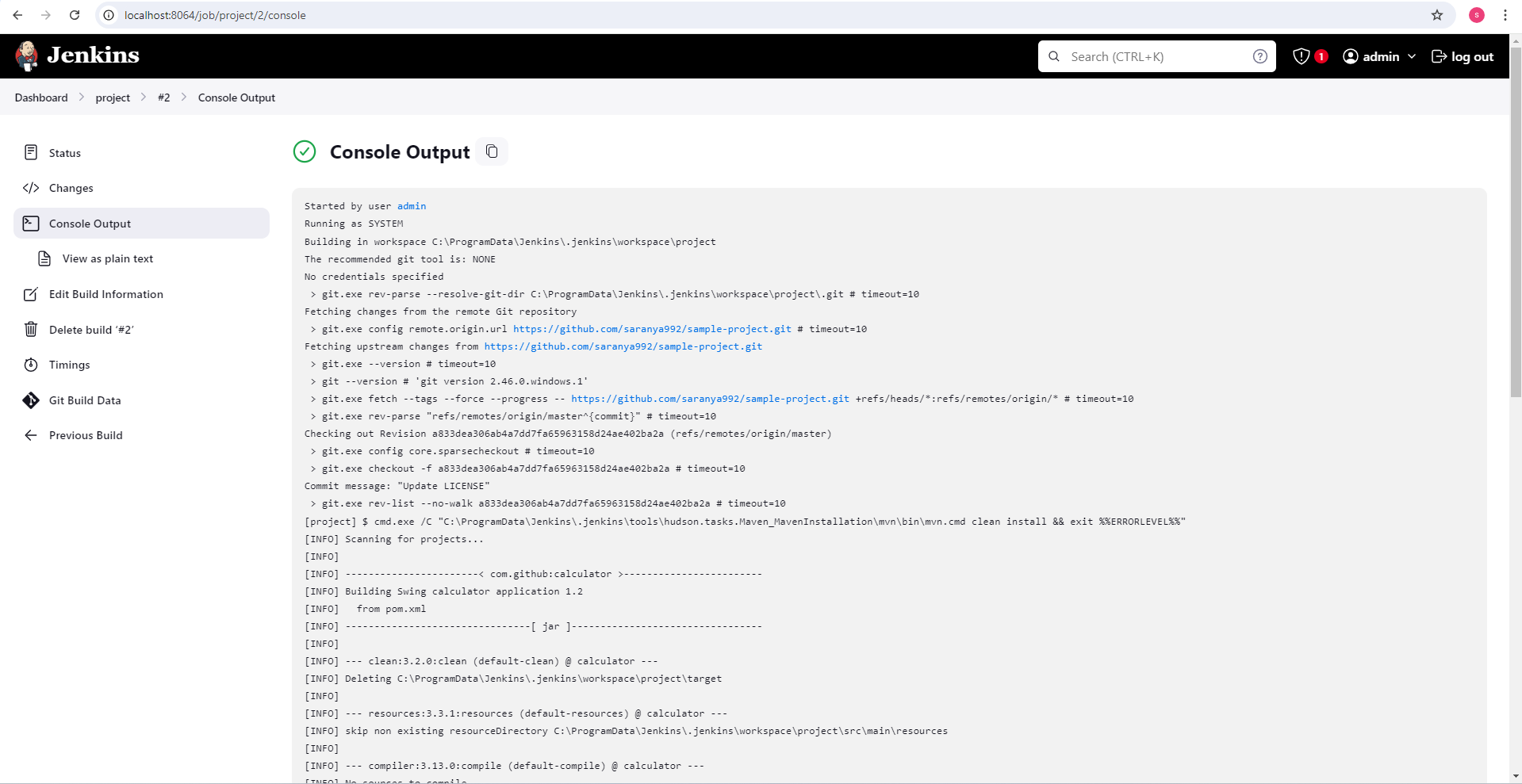
**Select execution given below build history**

****

**It will take you to the another page , it show like this**

****

**Click console output ->maven installed**

****

**Started by user admin**

**Running as SYSTEM**

**Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\project**

**The recommended git tool is: NONE**

**No credentials specified**

**> git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\.jenkins\workspace\project\.git # timeout=10**

**Fetching changes from the remote Git repository**

**> git.exe config remote.origin.url https://github.com/saranya992/sample-project.git # timeout=10**

**Fetching upstream changes from https://github.com/saranya992/sample-project.git**

**> git.exe --version # timeout=10**

**> git --version # 'git version 2.46.0.windows.1'**

**> git.exe fetch --tags --force --progress -- https://github.com/saranya992/sample-project.git +refs/heads/\*:refs/remotes/origin/\* # timeout=10**

**> git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10**

**Checking out Revision a833dea306ab4a7dd7fa65963158d24ae402ba2a (refs/remotes/origin/master)**

**> git.exe config core.sparsecheckout # timeout=10**

**> git.exe checkout -f a833dea306ab4a7dd7fa65963158d24ae402ba2a # timeout=10**

**Commit message: "Update LICENSE"**

**> git.exe rev-list --no-walk a833dea306ab4a7dd7fa65963158d24ae402ba2a # timeout=10**

**[project] $ cmd.exe /C "C:\ProgramData\Jenkins\.jenkins\tools\hudson.tasks.Maven\_MavenInstallation\mvn\bin\mvn.cmd clean install && exit %%ERRORLEVEL%%"**

**[INFO] Scanning for projects...**

**[INFO]**

**[INFO] -----------------------< com.github:calculator >------------------------**

**[INFO] Building Swing calculator application 1.2**

**[INFO] from pom.xml**

**[INFO] --------------------------------[ jar ]---------------------------------**

**[INFO]**

**[INFO] --- clean:3.2.0:clean (default-clean) @ calculator ---**

**[INFO] Deleting C:\ProgramData\Jenkins\.jenkins\workspace\project\target**

**[INFO]**

**[INFO] --- resources:3.3.1:resources (default-resources) @ calculator ---**

**[INFO] skip non existing resourceDirectory C:\ProgramData\Jenkins\.jenkins\workspace\project\src\main\resources**

**[INFO]**

**[INFO] --- compiler:3.13.0:compile (default-compile) @ calculator ---**

**[INFO] No sources to compile**

**[INFO]**

**[INFO] --- resources:3.3.1:testResources (default-testResources) @ calculator ---**

**[INFO] skip non existing resourceDirectory C:\ProgramData\Jenkins\.jenkins\workspace\project\src\test\resources**

**[INFO]**

**[INFO] --- compiler:3.13.0:testCompile (default-testCompile) @ calculator ---**

**[INFO] No sources to compile**

**[INFO]**

**[INFO] --- surefire:3.2.5:test (default-test) @ calculator ---**

**[INFO] No tests to run.**

**[INFO]**

**[INFO] --- jar:3.4.2:jar (default-jar) @ calculator ---**

**[WARNING] JAR will be empty - no content was marked for inclusion!**

**[INFO] Building jar: C:\ProgramData\Jenkins\.jenkins\workspace\project\target\calculator.jar**

**[INFO]**

**[INFO] --- shade:3.6.0:shade (default) @ calculator ---**

**[INFO] Including org.swinglabs:swing-layout:jar:1.0.3 in the shaded jar.**

**[INFO] Dependency-reduced POM written at: C:\ProgramData\Jenkins\.jenkins\workspace\project\dependency-reduced-pom.xml**

**[WARNING] calculator.jar, swing-layout-1.0.3.jar define 1 overlapping resource:**

**[WARNING] - META-INF/MANIFEST.MF**

**[WARNING] maven-shade-plugin has detected that some files are**

**[WARNING] present in two or more JARs. When this happens, only one**

**[WARNING] single version of the file is copied to the uber jar.**

**[WARNING] Usually this is not harmful and you can skip these warnings,**

**[WARNING] otherwise try to manually exclude artifacts based on**

**[WARNING] mvn dependency:tree -Ddetail=true and the above output.**

**[WARNING] See https://maven.apache.org/plugins/maven-shade-plugin/**

**[INFO] Replacing original artifact with shaded artifact.**

**[INFO] Replacing C:\ProgramData\Jenkins\.jenkins\workspace\project\target\calculator.jar with C:\ProgramData\Jenkins\.jenkins\workspace\project\target\calculator-1.2-shaded.jar**

**[INFO]**

**[INFO] --- install:3.1.2:install (default-install) @ calculator ---**

**[INFO] Installing C:\ProgramData\Jenkins\.jenkins\workspace\project\dependency-reduced-pom.xml to C:\Windows\system32\config\systemprofile\.m2\repository\com\github\calculator\1.2\calculator-1.2.pom**

**[INFO] Installing C:\ProgramData\Jenkins\.jenkins\workspace\project\target\calculator.jar to C:\Windows\system32\config\systemprofile\.m2\repository\com\github\calculator\1.2\calculator-1.2.jar**

**[INFO] ------------------------------------------------------------------------**

**[INFO] BUILD SUCCESS**

**[INFO] ------------------------------------------------------------------------**

**[INFO] Total time: 0.997 s**

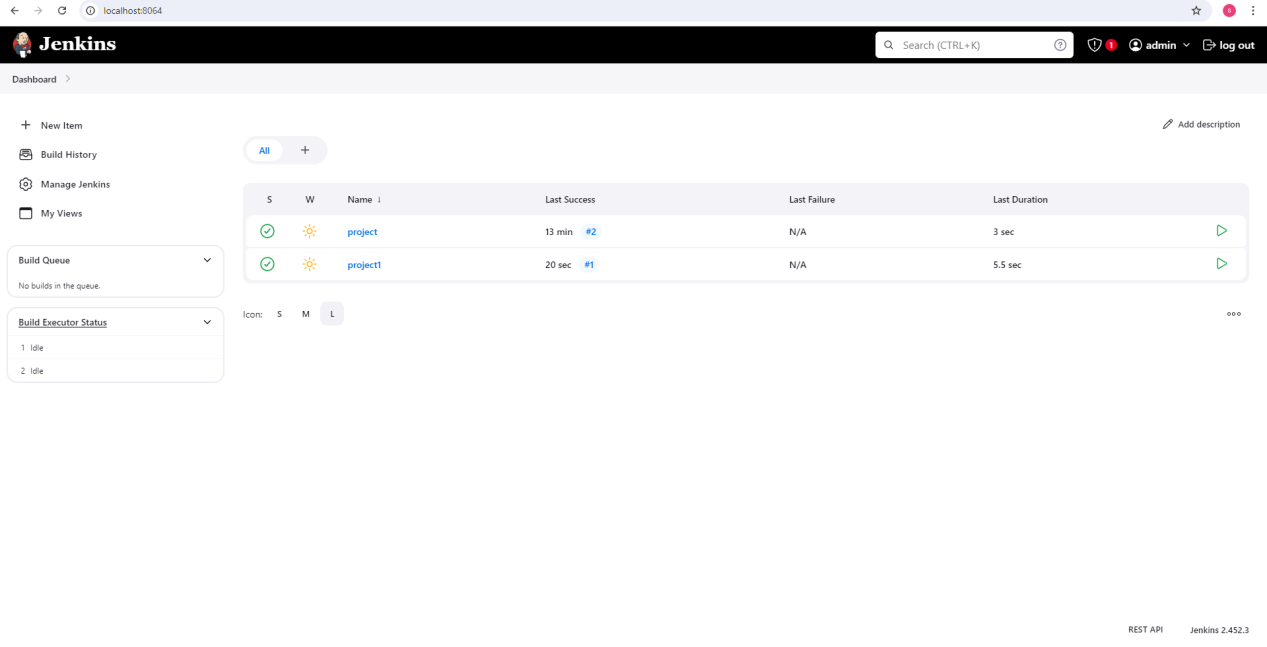
**[INFO] Finished at: 2024-08-02T12:15:36+05:30**

**[INFO] ------------------------------------------------------------------------**

**Finished: SUCCESS**

**Finally it shows success**

**ATLAST IT SHOWS THE DASHBOARD LIKE :**

****

Jenkins Dashboard -> Manage Jenkins -> Manage Plugins -> Available Tab -> Search "copy artifact" -> Check the box -> Install

