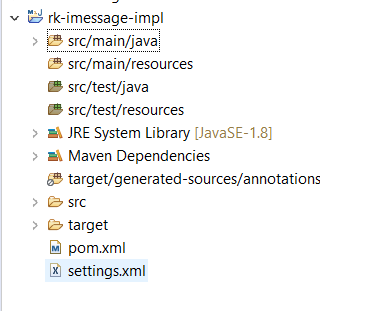
**Process – 1: Manually pulling jar file (artefacts) from JFrog artifactory using Maven commands**

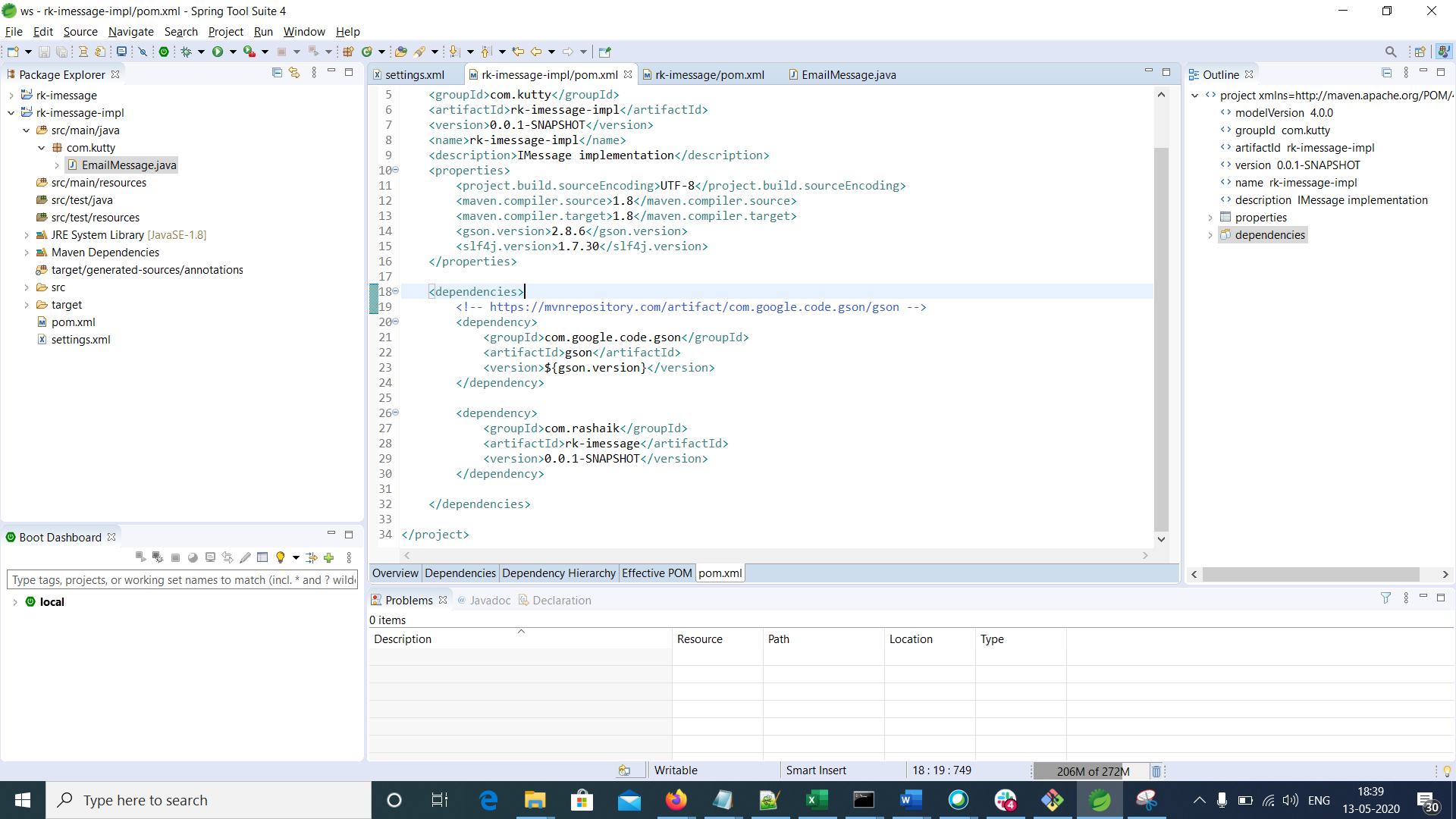
* Code Base: Required files – Project Structure



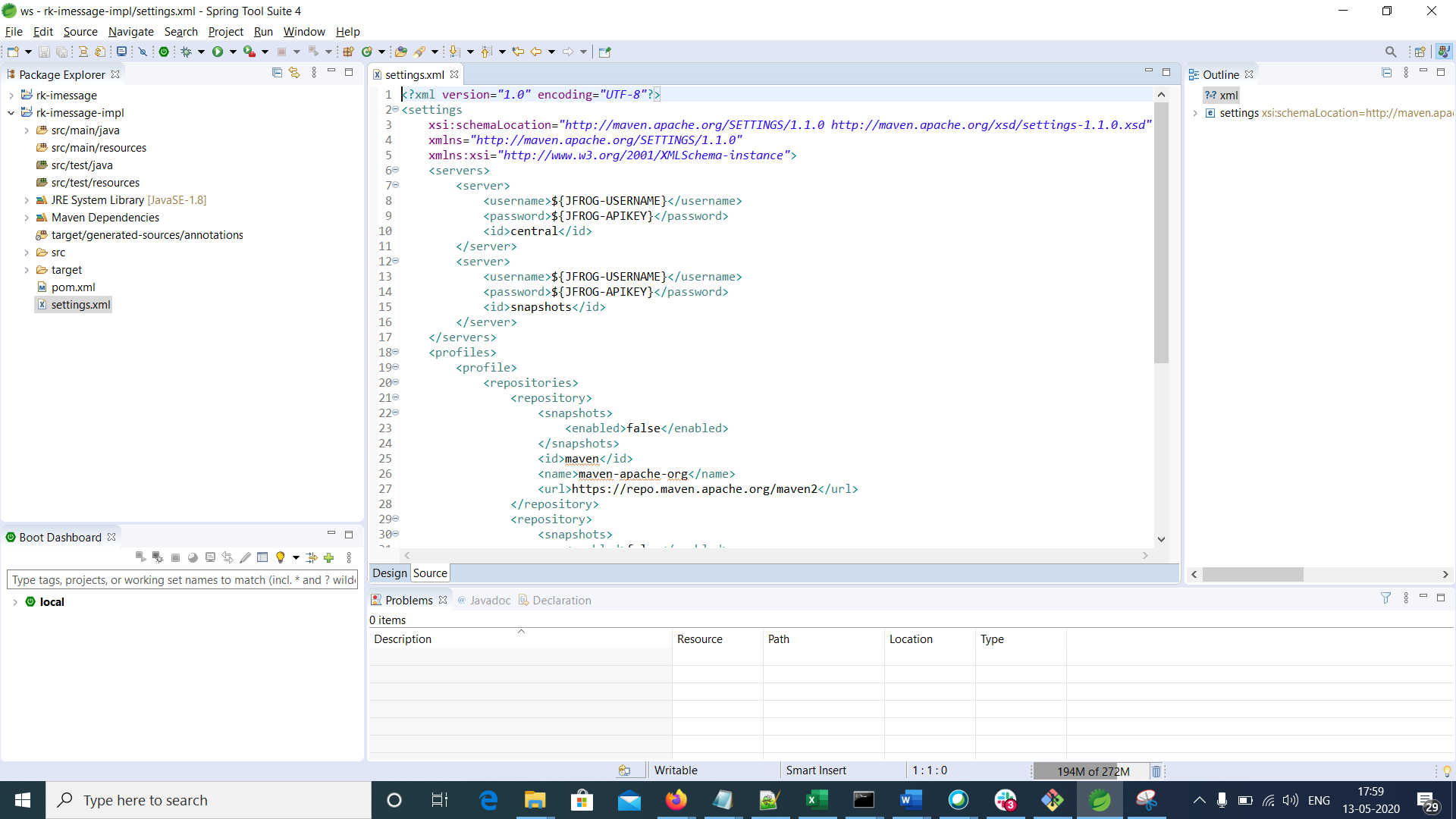
* Execute commands to pull artefact from JFrog

mvn clean compile --settings settings.xml -Dusername={USER\_NAME} -Dpassword={API\_KEY}

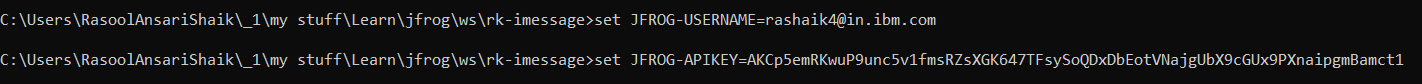
* Pom.xml – NO distributionManagement



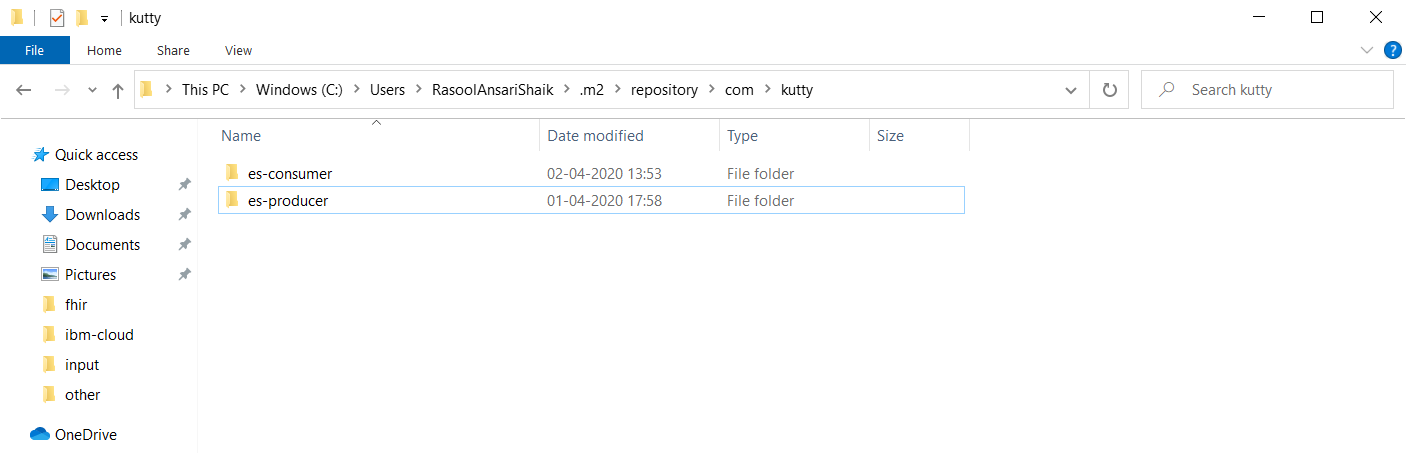
Settings.xml



Set JFROG-USERNAME and JFOG-APIKEY values as the environment variables. OR you can directly use them.

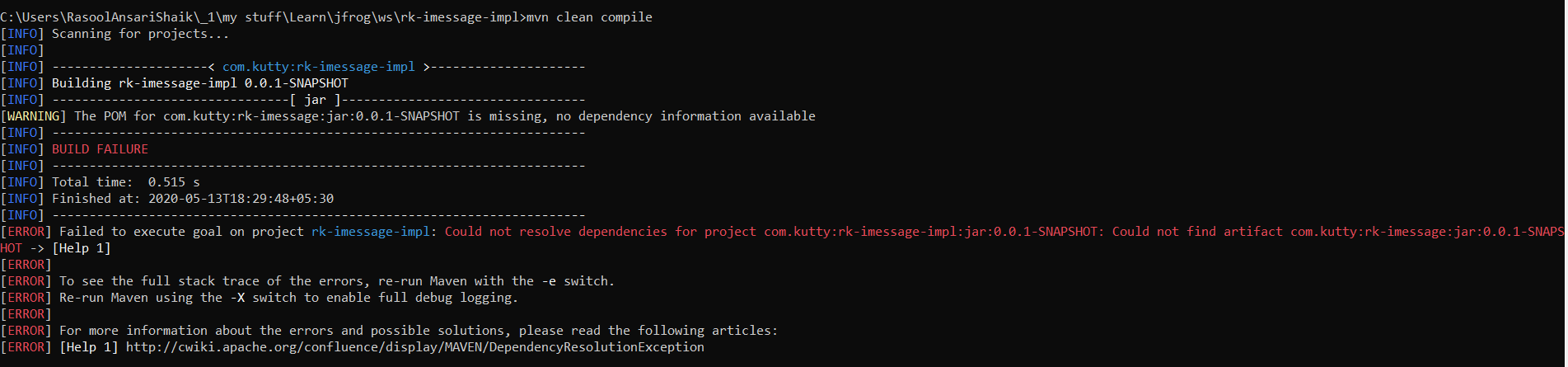


* JFrog url - <https://na.artifactory.swg-devops.com/artifactory>
* Artefactory - wh-phytel-unicorn-team-maven-local
* I have removed if the rk-imessage artefact in local.m2 repo for the testing purpose.



* Next, I run the below command without settings.xml. That is we are not referring rk-imessage from JFrog. Expected output should be having errors.

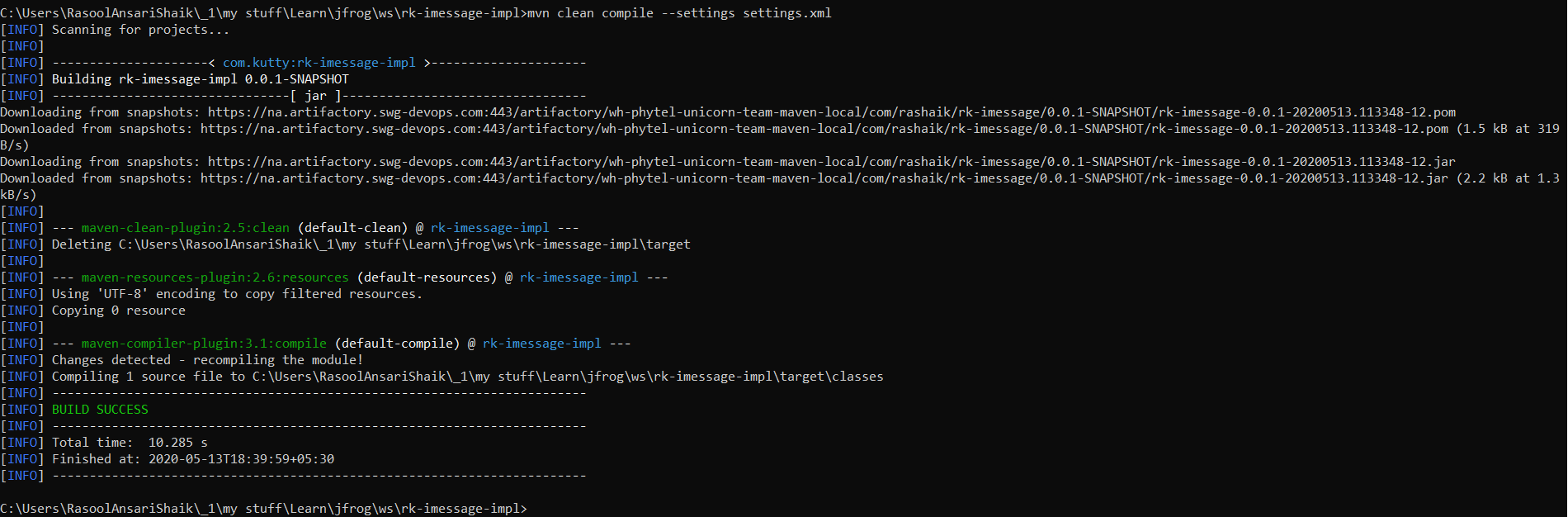
mvn clean compile



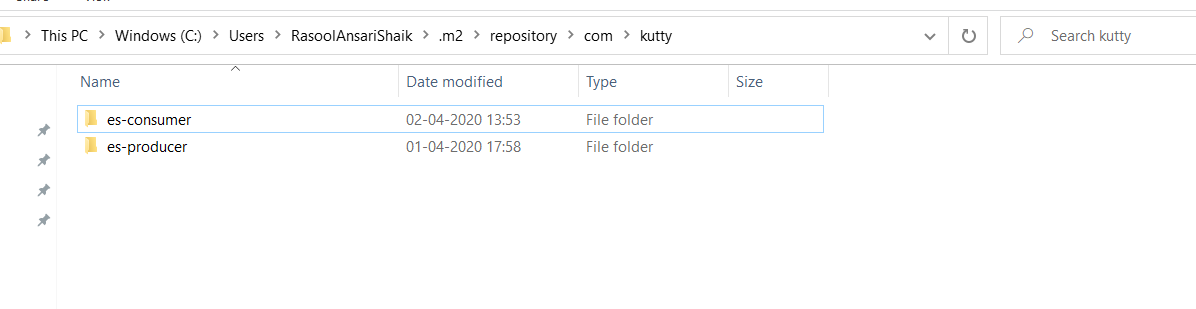
It is expected as there is no artefact available for rk-imessage.

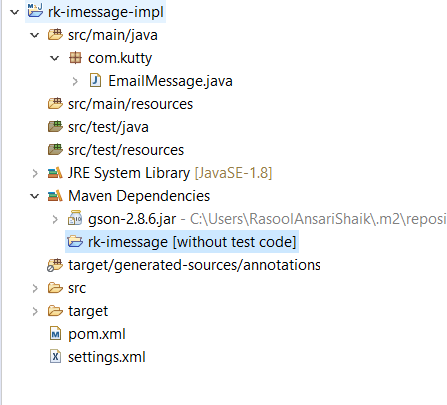
* Now execute the same command with settings.xml

mvn clean compile --settings settings.xml



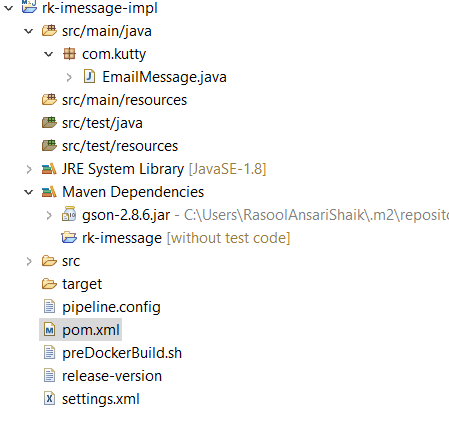
It executed successfully. The command don’t pull the artefact to local .m2 repo. It would be available to the project workspace only.



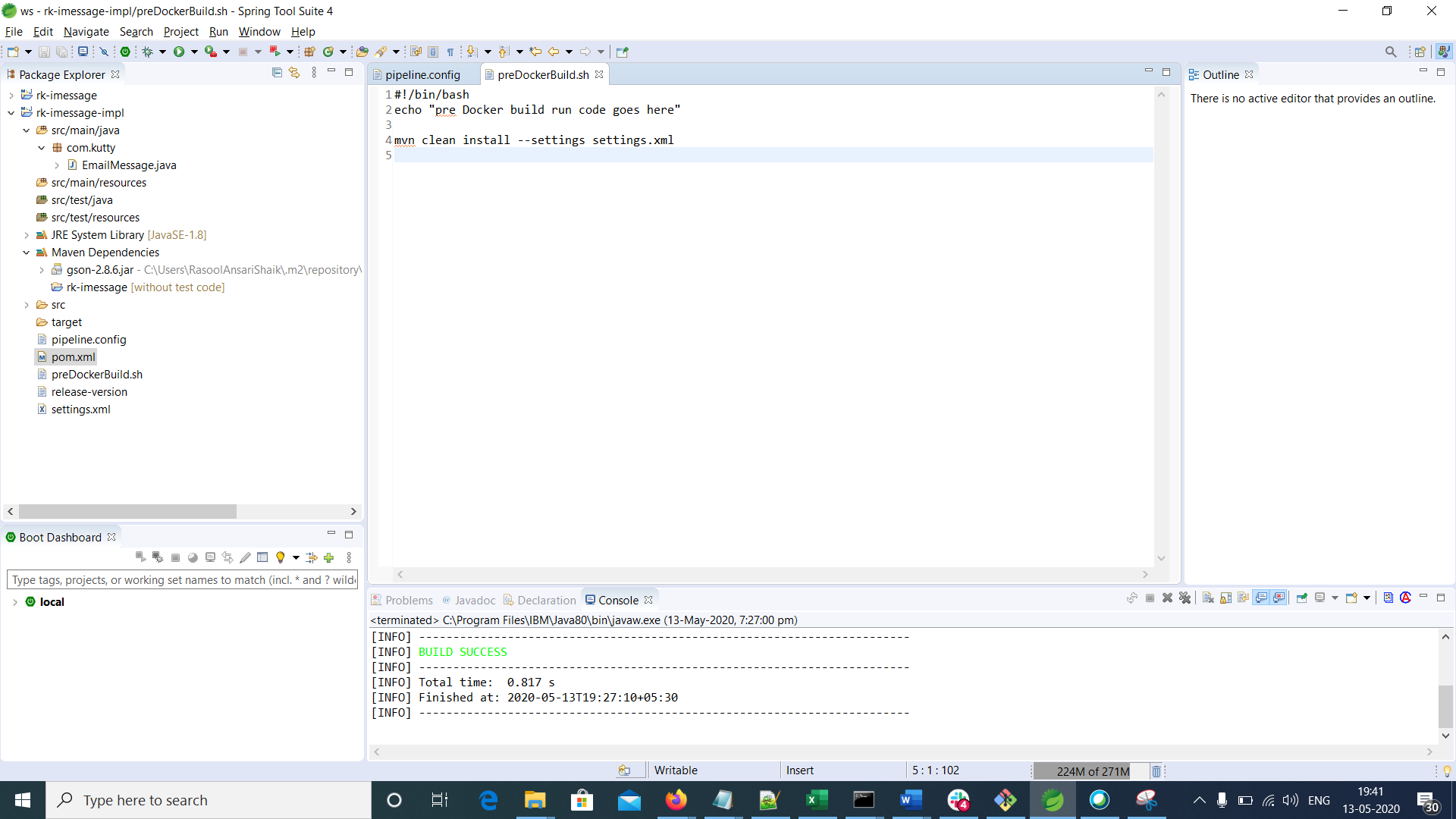


**Process – 2: Automatically pulling jar file from JFrog artifactory using CICD pipeline**

* Code Base: Required files – Project Structure



* In the CI pipeline’s Pre-Docker-Build stage, preDockerBuild.sh file is execute. So put the required code in this shell file.



Using above command in the preDockerBuild.sh file, the dependency artefacts will get pulled from JFrog artefactory to cluster.