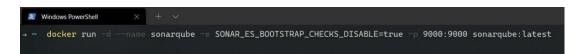**Name: Abhishek Chaurasiya**
Class: D15B / 9
ADVDEVOPS PRACTICAL 8

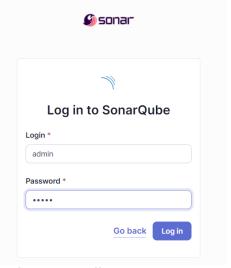**AIM: Integrating Jenkins with SonarQube.**

Open up Jenkins Dashboard on localhost, port 8080
Run SonarQube in a Docker container using this command -
$ docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true
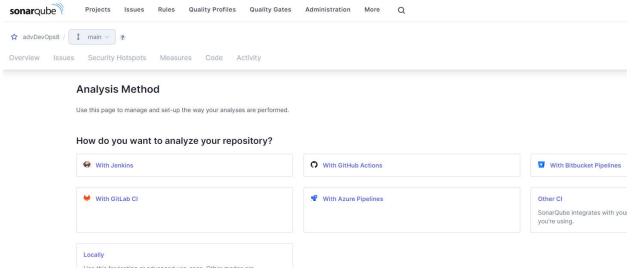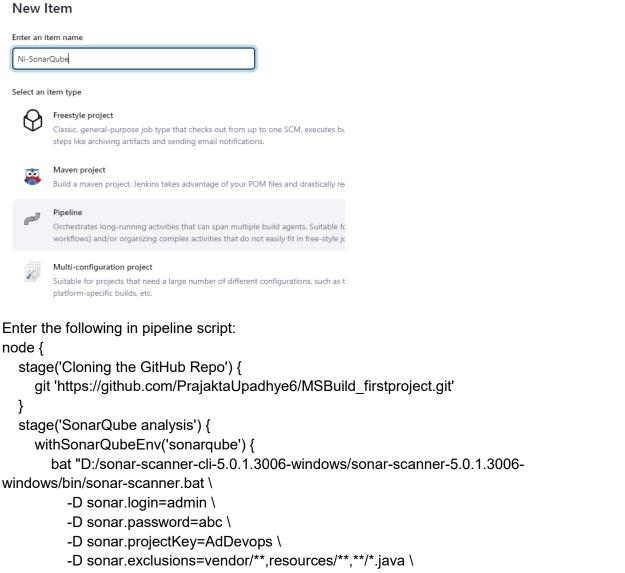**-p 9000:9000 sonarqube:latest**

sonarqube

Create project manually
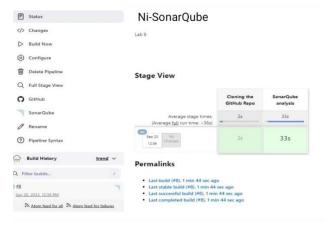
In Jenkins create a pipeline here named "SonarQube"

Enter the following in pipeline script:

```
node {
    stage('Cloning the GitHub Repo') {
        git 'https://github.com/PrajaktaUpadhye6/MSBuild_firstproject.git'
    }
    stage('SonarQube analysis') {
        withSonarQubeEnv('sonarqube') {
            bat "D:/sonar-scanner-cli-5.0.1.3006-windows/sonar-scanner-5.0.1.3006-
windows/bin/sonar-scanner.bat \
                -D sonar.login=admin \
                -D sonar.password=abc \
                -D sonar.projectKey=AdDevops \
                -D sonar.exclusions=vendor/**,resources/**,**/*.java \
```

```
                -D sonar.host.url=http://127.0.0.1:9000/"
        }
    }
}
```

Pipeline

Definition

Pipeline script

Script ?

```
 1 ▾ node {
 2 ▾     stage('Cloning the GitHub Repo') {
 3             git 'https://github.com/PrajaktaUpadhye6/MSBuild_firstproject.git'
 4         }
 5 ▾     stage('SonarQube analysis') {
 6 ▾         withSonarQubeEnv('sonarqube') {
 7                 bat "D:/sonar-scanner-cli-5.0.1.3006-windows/sonar-scanner-5.0.1.3006-windows/bin/sonar-scanner.bat \
 8                     -D sonar.login=admin \
 9                     -D sonar.password=Admin \
10                     -D sonar.projectKey=AdDevops \
11                     -D sonar.exclusions=vendor/**,resources/**,**/*.java \
12                     -D sonar.host.url=http://127.0.0.1:9000/"
13             }
14         }
15 }
16
```

✓ Use Groovy Sandbox ?

It is a java sample project which has a lot of repetitions and issues that will be detected by SonarQube.
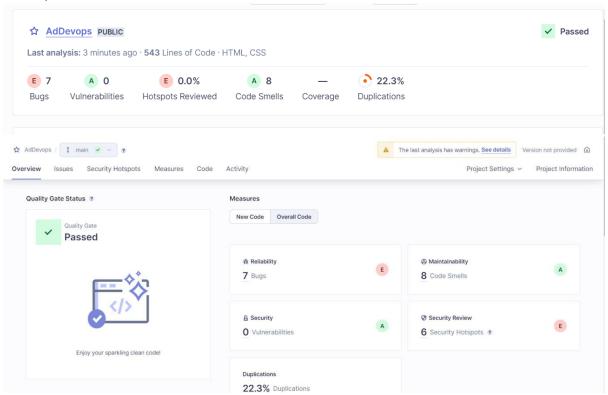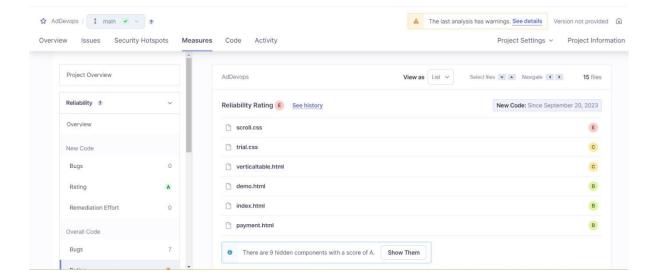
Build and run:

Console output:

```
17:49:39.413 INFO  CPD Executor CPD calculation finished (done) | time=159837ms
17:49:39.428 INFO  SCM revision ID 'ba799ba7e1b576f04a4612322b0412c5e6e1e5e4'
17:52:16.075 INFO  Analysis report generated in 4781ms, dir size=127.2 MB
17:52:33.696 INFO  Analysis report compressed in 17599ms, zip size=29.6 MB
17:52:34.987 INFO  Analysis report uploaded in 1290ms
17:52:34.989 INFO  ANALYSIS SUCCESSFUL, you can find the results at: http://127.0.0.1:9000/dashboard?id=sonarqube
17:52:34.989 INFO  Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
17:52:34.989 INFO  More about the report processing at http://127.0.0.1:9000/api/ce/task?id=8c04859e-050f-41e5-b320-ee32d44d4259
17:52:43.609 INFO  Analysis total time: 16:49.875 s
17:52:43.612 INFO  SonarScanner Engine completed successfully
17:52:44.416 INFO  EXECUTION SUCCESS
17:52:44.419 INFO  Total time: 17:04.839s
[Pipeline] }
[Pipeline] // withSonarQubeEnv
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Declarative: Post Actions)
[Pipeline] echo
Pipeline completed successfully!
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```
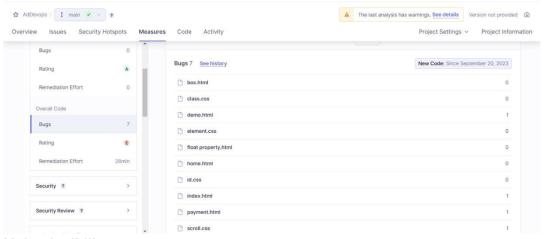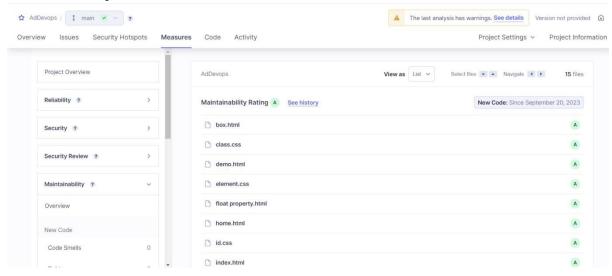
sonarqube:

Reliability:

Bugs:

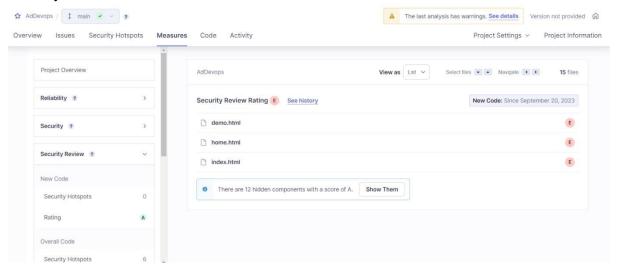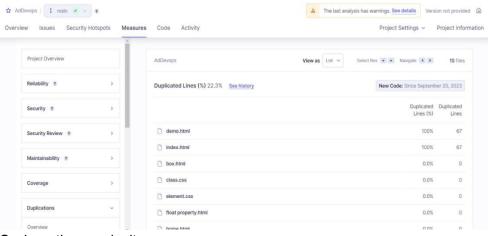## Maintainaibility:

Security:
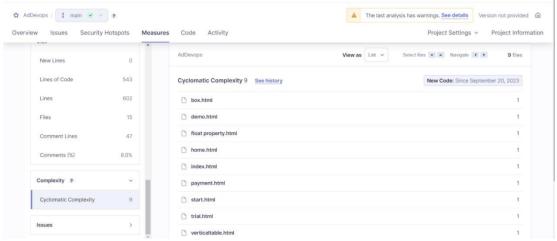

Duplications:

Cyclomatic complexity:

Conclusion: Thus, we have successfully integrated Jenkins with SonarQube.