

9. SonarQube

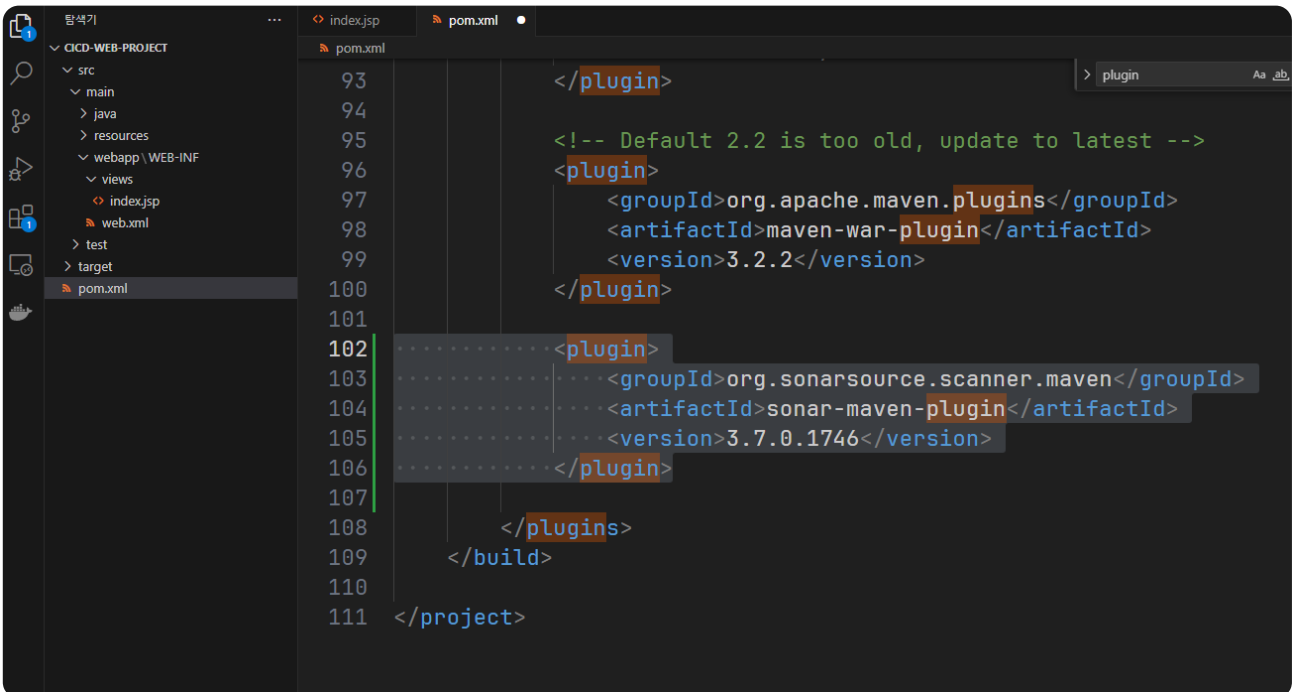
// sonarqube와 jenkins 연동

1. docker로 sonarqube start

- `docker pull sonarqube`
- `docker run --rm -p 9000:9000 --name sonarqube sonarqube`
- localhost:9000 으로 접속테스트

2. Maven Project에 Plugin 설정 추가

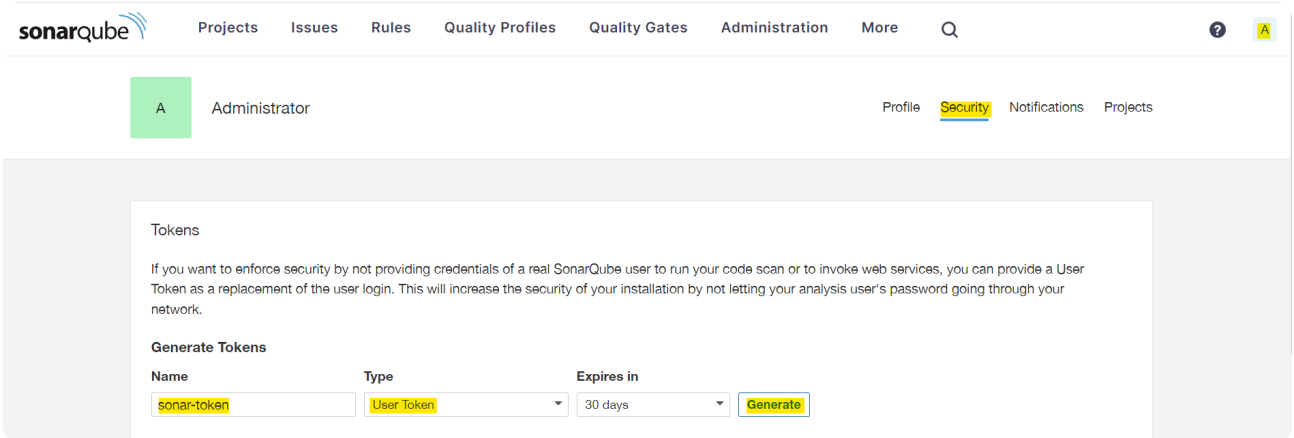
- pom.xml 파일에 추가
 - <https://docs.sonarsource.com/sonarqube/latest/analyzing-source-code/scanners/sonarscanner-for-maven/>



- 플러그인 최신 버전으로 수정 : 3.7.0.1746 -> 3.9.1.2184
 - <https://mvnrepository.com/artifact/org.sonarsource.scanner.maven/sonar-maven-plugin>

3. sonarqube

- localhost:9000 접속
 - 기본 계정 admin / admin
- My Account -> Security -> 토큰 생성
squ_fb18913e15faf6d1544af60a670278044e8d8ad0



- Maven에서 SonarQube 빌드 (local)
 - mvn sonar:sonar -Dsonar.host.url=http://IP_address:9000 -Dsonar.login=sonar-token
- Windows Powershell 등에서 명령어 오류가 발생하면 다음과 같이 인용부호(작은따옴표)를 붙이고 실행
 - mvn sonar:sonar '-Dsonar.host.url=http://IPaddress:9000'-Dsonar.login=[the-sonar-token]_'

```
mvn sonar:sonar -Dsonar.host.url=http://192.168.56.1:9000 -  
Dsonar.login=squ_fb18913e15faf6d1544af60a670278044e8d8ad0
```

4. jenkins + sonarqube

- jenkins 플러그인 설치
 - SonarQube Scanner
- Credentials 추가
 - 이전에 생성했던 sonarqube token 추가

New credentials

Kind

Secret text

Scope ?

Global (Jenkins, nodes, items, all child items, etc)

Secret

.....

ID ?

sonarqube-token

- jenkins 관리 -> system -> sonarqube servers
 - docker network inspect bridge 명령으로 powershell에서 ip 확인

SonarQube servers

If checked, job administrators will be able to inject a SonarQube server configuration as enviro



Environment variables

SonarQube installations

List of SonarQube installations

Name

SonarQube-server

Server URL

Default is http://localhost:9000

172.17.0.3:9000

Server authentication token

SonarQube authentication token. Mandatory when anonymous access is disabled.

sonarqube-token

Add ▾

- jenkins pipeline 설정

```
pipeline {
  agent any
  tools {
    maven 'Maven3.9.4'
  }
  stages {
    stage('github clone') {
      steps {
        git 'https://github.com/aheeej/cicd-web-project.git'
      }
    }
  }
}
```

```
}
stage('build'){
    steps {
        sh '''
            echo build start
            mvn clean compile package -DskipTests=true
            '''
    }
}
stage('SonarQube analysis'){
    steps {
        withSonarQubeEnv('SonarQube-server') {
            sh 'mvn sonar:sonar'
        }
    }
}
// stage('ssh publisher') {
//     steps {
//         sshPublisher(publishers: [sshPublisherDesc(configName: 'dock
//     }
// }
}
}
```