

## 4 코드품질을 위한 Sonarqube 활용

### 03 Sonarqube 코드품질 설정 및 Docker 빌드 스캔 수행

## 실습 진행사항 목록

03

Sonarqube  
코드품질 설정 및  
Docker 빌드 스캔  
수행

### Sonarqube 코드품질 설정 및 Docker 빌드 스캔 수행

- Gradle에서 SonarQube 및 Jacoco 플러그인 설정
- Gradle 및 Docker 빌드, SonarQube 정적분석 스캔 수행
- SonarQube에서 정적코드 품질분석 결과확인

### 사전 준비사항

- SonarQube 서버용 AWS EC2 VM 1개 (이전 실습 때 사용)
- AWS ECR Repository 1개 (Docker 이미지 저장용)

# Jacoco 소개

03

Sonarqube  
코드품질 설정 및  
Docker 빌드 스캔  
수행

## Java Code Coverage

자바 소스파일의 코드 커버리지 제공  
테스트케이스에 의한 테스트수 측정

## Instruction Coverage

코드 실행량 측정 커버리지

## Branch Coverage

if나 switch 문의 분기 확인  
실행된 것과 실행되지 않은 부분 측정

## Cyclomatic complexity

function 테스트시 최소 경로 정보  
모든 코드를 커버하기 위한 테스트 수

## SonarQube 및 Jacoco 실습 명령어

03

Sonarqube  
코드품질 설정 및  
Docker 빌드 스캔  
수행

- 실습 예제코드 빌드 명령어 :
  - `./gradlew clean build --info`
- Jacoco 코드 커버리지 측정 및 리포트 작성 명령어 :
  - `./gradlew jacocoTestCoverageVerification --info`
  - `./gradlew jacocoTestReport --info`
- SonarQube 코드 품질 스캔 결과 연동 명령어 :
  - `./gradlew sonarqube --info`
- Docker 이미지 빌드 및 Push 명령어 :
  - `./gradlew jib --console=plain`