

```

1 import ...
2
3 public class RunSonarProcessor {
4
5     @Override
6     public Project process(Project project) throws Exception {
7
8
9         String defaultBranchName = getDefaultBranch(gitDirectory);
10        Git git = Git.open(gitDirectory);
11        numberOfCommits = countNumberOfCommits(
12            git.getRepository(), defaultBranchName
13        );
14        frameSize = numberOfCommits / this.frameNumber // frameNumber = 5;
15        for(int i = 0 ; i < this.frameNumber ; i++)
16        {
17            try {
18                git.checkout().
19                    setCreateBranch(false).
20                    setName(defaultBranchName).
21                    setForce(true).call();
22
23                ObjectId Myhead = git.getRepository().
24                    resolve("HEAD~" +
25                        (frameSize * (this.frameNumber - i - 1)));
26                git.checkout().
27                    setForce(true).
28                    setName(Myhead.getName()).
29                    call();
30
31                this.initializeSonar();
32                executeSonarScanner(project, projectDirectory, i + 1);
33            } catch (Exception e)
34            {
35                logger.error("Fail to execute Sonar: "+e.getMessage());
36            }
37        }
38
39        return project;
40    }
41 }
42

```