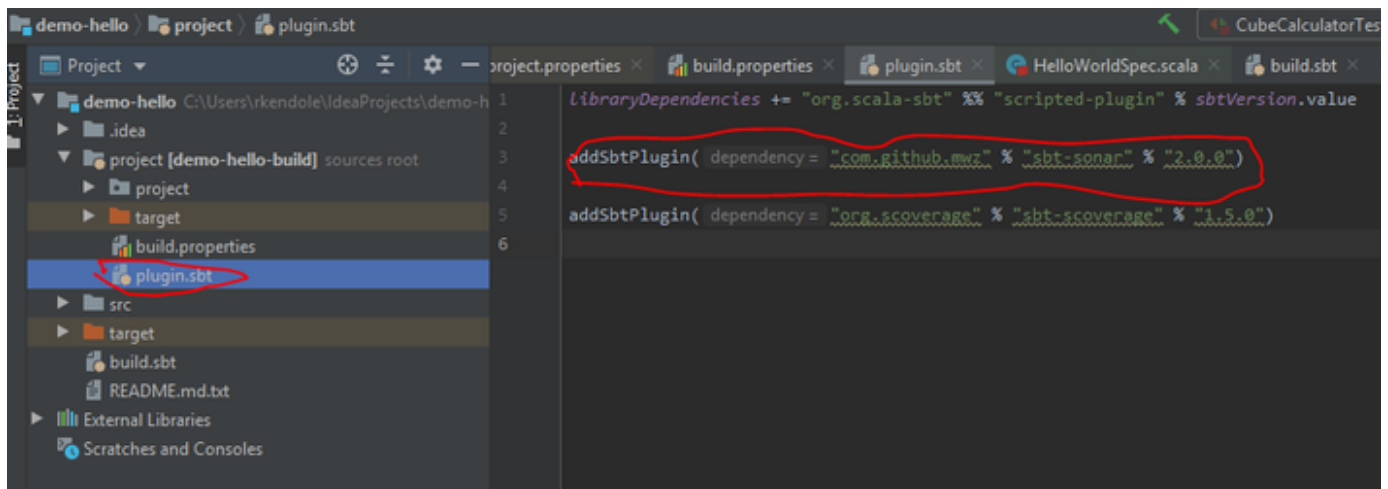# How To - jenkins -Sonar integration for Scala

sbt-sonar:

An sbt plugin which provides an easy way to integrate Scala projects with SonarQuabe.

To install this plugin in your project, add the following to your `./project/plugins.sbt` file:

```
addSbtPlugin("com.github.mwz" % "sbt-sonar" % "2.0.0")
```
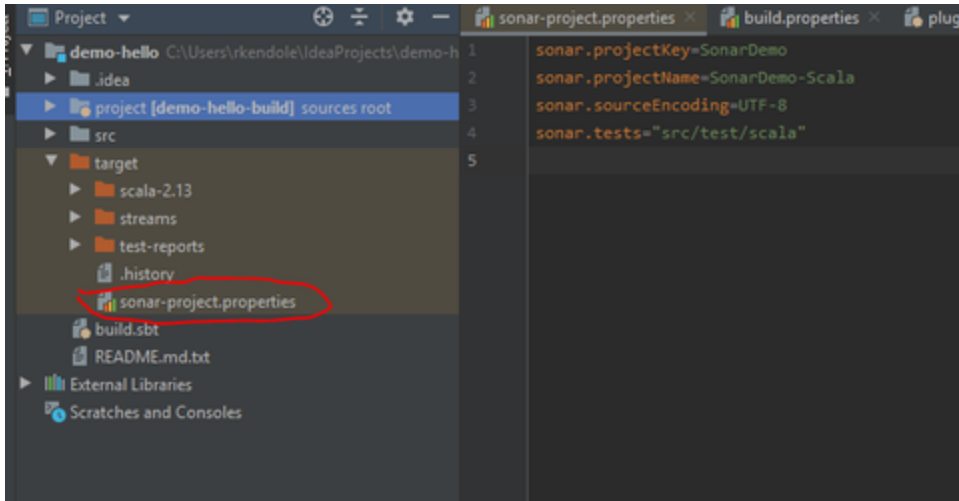

You can define your project properties either in the external config file `sonar-project.properties`, which should be located in the root directory of your project.

sonar-project.properties will look like:

```
#sonar-project.properties

sonar.projectName
sonar.projectKey
sonar.sourceEncoding
sonar.tests
```
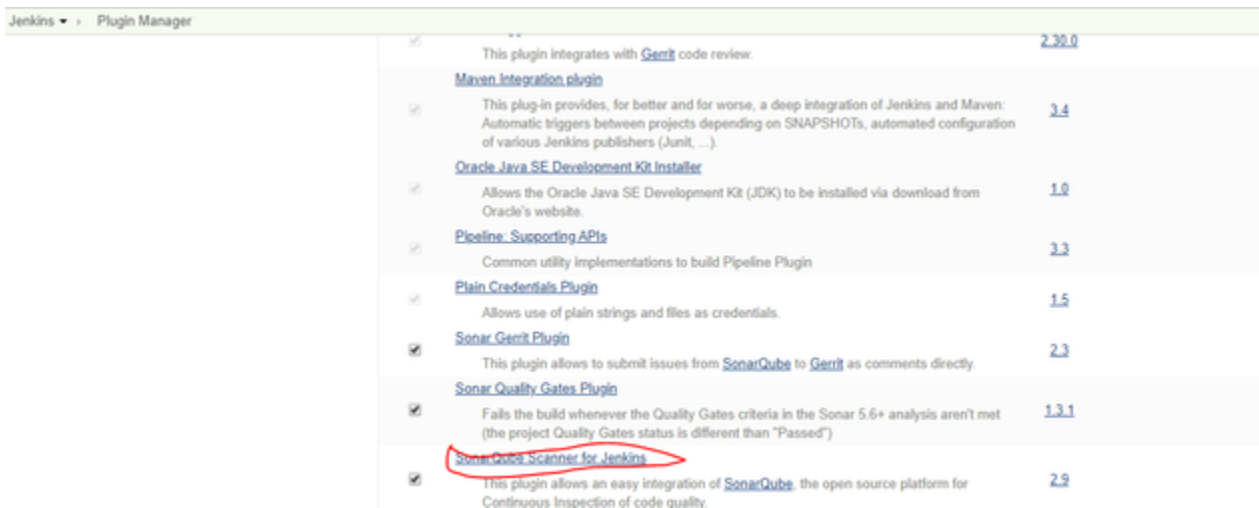
Now run the below commands to publish code analysis and code coverage reports in your sonarQube server.

```
sbt   coverage test
sbt   coverageReport
sbt   sonarScan
```

Jenkins integration to sonar

1.Install *SonarQube plugin* to Jenkins

Manage Jenkins >> Manage Plugins >> Available Tab and select "SonarQube Scanner for Jenkins" plugin and install.

And install Sbt for scala program

☑	**Mailer** This plugin allows you to configure email notifications for build results	1.27	Uninstall
☑	**Matrix Authorization Strategy** Offers matrix-based security authorization strategies (global and per-project).	1.1	Uninstall
☑	**Matrix Project Plugin** Multi-configuration (matrix) project type.	1.14	Uninstall
☑	**Oracle Java SE Development Kit Installer** Allows the Oracle Java SE Development Kit (JDK) to be installed via download from Oracle's website.	1.0	Uninstall
☑	**OWASP Markup Formatter** Uses the OWASP Java HTML Sanitizer to allow safe-seeming HTML markup to be entered in project descriptions and the like.	1.1	Uninstall
☑	**PAM Authentication** Adds Unix Pluggable Authentication Module (PAM) support to Jenkins	1.1	Uninstall
☑	**sbt plugin** This plugin allows running SBT empowered scala projects in Hudson.	1.5	**Uninstall**
☑	**WMI Windows Agents** Allows you to connect to Windows machines and start slave agents on them.	1.0	Uninstall

Go to Manage Jenkins >>cofiguring the system, Search SonarQube servers section,

Jenkins ▾ › configuration

Delet...

Add

☐ Tool Locations

SonarQube servers

Environment variables   ☐ Enable injection of SonarQube server configuration as build environment variables

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

SonarQube installations

Name   Sonar

Server URL   http://inovartis.devops.altimetrik.io 9000

Default is http://localhost:9000

Server authentication token   ← Add ▾

- none - ▾

SonarQube authentication token. Mandatory when anonymous access is disabled.

Advanced...

Delete SonarQub...

Add SonarQube

List of SonarQube installations

*Configure **sonarquabe scanner in jenkins***

manageJenkins > Global Tool Configuration

Jenkins ▾ ›   ENABLE AUTO REFRESH

Improper user account validation

Bitbucket Approve Plugin 1.0.3:
Bitbucket Approve Plugin stores credentials in plain text

Configure System
Configure global settings and paths.

Configure Global Security
Secure Jenkins; define who is allowed to access/use the system.

Configure Credentials
Configure the credential providers and types.

Global Tool Configuration
Configure tools, their locations and automatic installers.

Reload Configuration from Disk
Discard all the loaded data in memory and reload everything from file system. Useful when you modified config files directly on disk.

Manage Plugins
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
⚠ There are updates available

ovartis.devops.altimetrik.io:8084

System Information

SonarQube Scanner

SonarQube Scanner installations

Add SonarQube Scanner

SonarQube Scanner

Name sonar

☑ Install automatically

Install from Maven Central

Version SonarQube Scanner 4.1.0.1829 ▼

Sbt

Sbt installations

Add Sbt

Sbt
name sbt

sbt launch arguments

Save   Apply
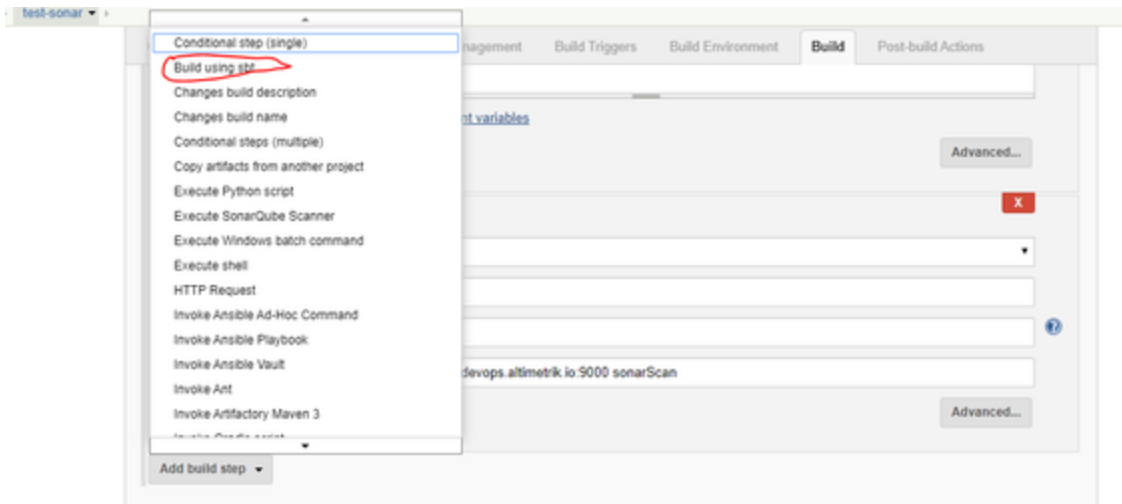
Save it.

Run sonar with sonar.properties

Create a Job- *New Item*


Click on Add build step build using sbt

And the add sbtlauncher name & actions

Run the build

```
[info] CPD calculation finished
[info] Analysis report generated in 96ms, dir size=25 KB
[info] Analysis reports compressed in 14ms, zip size=9 KB
[info] Analysis report uploaded in 324ms
[info] ANALYSIS SUCCESSFUL, you can browse http://novartis.devops.altimetrik.io:9000/dashboard/index/demo-hello
[info] Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
[info] More about the report processing at http://novartis.devops.altimetrik.io:9000/api/ce/task?id=AW1mB_fK1zrs5x8rhE3_
[info] Task total time: 5.867 s
[success] Total time: 7 s, completed Sep 25, 2019 1:27:41 AM
Build step 'Build using sbt' changed build result to SUCCESS
Finished: SUCCESS
```

Go to Sonar dashboard
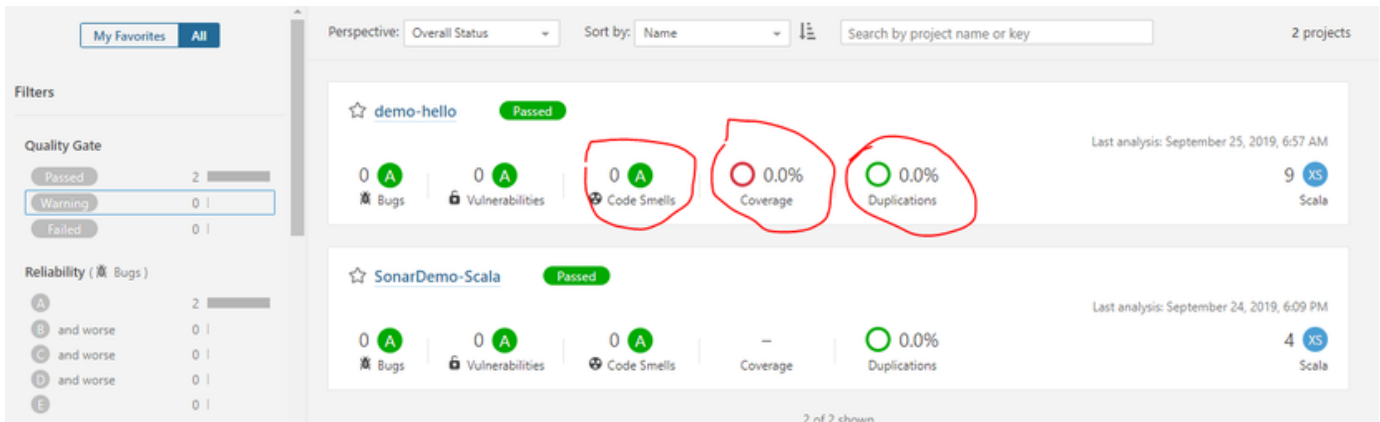
Run sonar with build.sbt

Now add the `sbt-sonar` plugin dependency to your scala project .

```
addSbtPlugin("com.github.mwz" % "sbt-sonar" % "2.0.0")
```

This sbt plugin can be used to run sonar-scanner launcher to analyze a Scala project with SonarQube.

Add dependencies libraries

Source	Description
..	
📁 project/target/config-classes	
📁 target	
📄 build.properties	sonar
📄 Dependencies.scala	dependency added
📄 plugin.sbt	plugin.sbt edited online with Bitbucket

```
import sbt._

object Dependencies {
  lazy val scalaTest = "org.scalatest" %% "scalatest" % "3.0.5"
}
```

Configuration

Now, configure the sonar-properties in your project. This can be done in 2 ways

1.Use sonar-project.properties file This file has to be placed in your root directory

2.Configure Sonar-properties in build.sbt file

By default, the plugin expects the properties to be defined in the sonarProperties setting key in sbt.

```
import sbtsonar.SonarPlugin.autoImport.sonarProperties

import Dependencies._

ThisBuild / scalaVersion    := "2.12.8"
ThisBuild / version         := "0.1.0-SNAPSHOT"


lazy val root = (project in file("."))
  .settings(
    name := "demo-hello",
    libraryDependencies += scalaTest % Test
  )

  sonarProperties ++= Map(
  "sonar.host.url" -> "http://novartis.devops.altimetrik.io:9000/",
  "sonar.projectName" -> "demo-hello",
  "sonar.projectKey" -> "demo-hello",
  "sonar.sourceEncoding" -> "UTF-8",
  "sonar.java.source" -> "1.8",
  "sonar.sources" -> "src/main/scala",
  "sonar.tests" -> "src/test/scala",
)
```
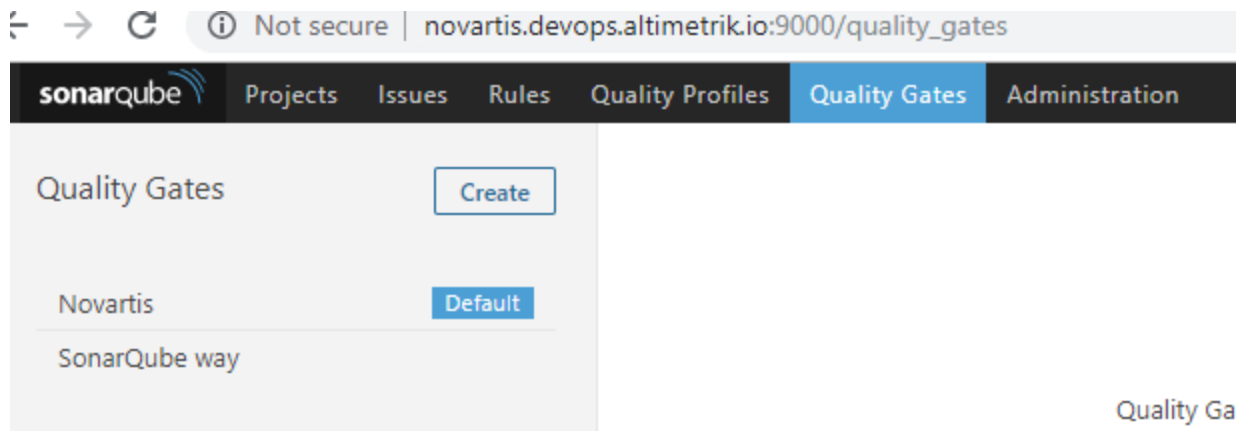
Run the build and go to sonar dashboard.

Create quality gates

1. Go to sonar dashboard >Quality gates

Click on create option give the name

Create Quality Gate

Name *

[   ]

Create   Cancel

Add conditions and save it.

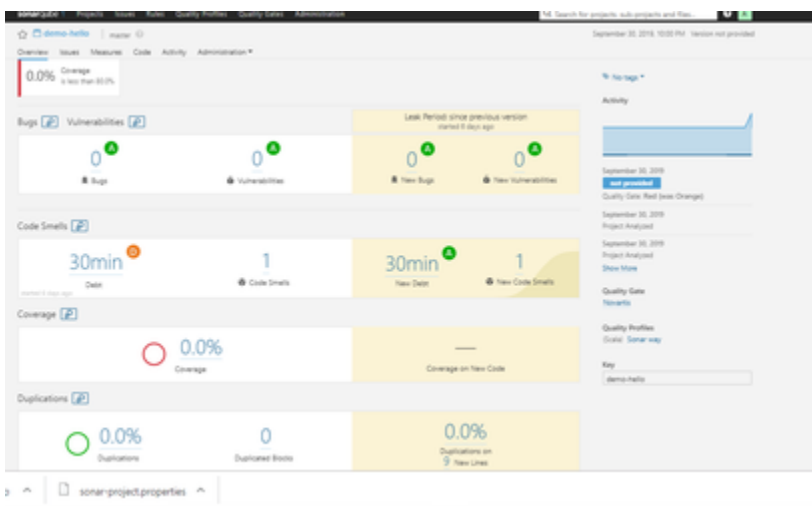| Quality Gates	Create	Novartis						Rename	Copy	Unset as Default	Delete

Novartis   Default

SonarQube way

Conditions

Only project measures are checked against thresholds. Sub-projects, directories and files are ignored. More

Metric	Over Leak Period	Operator		Warning	Error		
Conditions to Cover on New Code	Always	equals	▾		5	Update	Delete
Coverage	☐	is less than	▾		80	Update	Delete
Skipped Unit Tests	☐	is not	▾		0	Update	Delete

Add Condition   ▾

Projects

NOTE: Unit test cases count not publishing on sonar dashboard. There is bug in sonar scala plugin. Once bug fix is done will implement this .