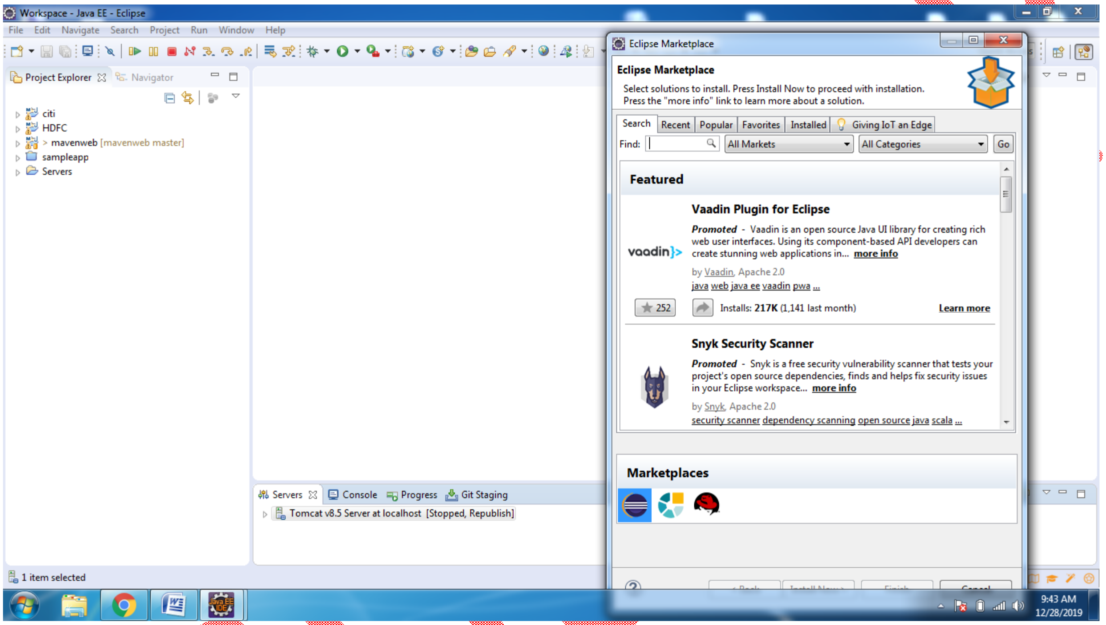
Sonar Qube

SonarLint:

SonarLint is an IDE extension that helps you detect and fix quality issues as you write code. Like a spell checker, **SonarLint** squiggles flaws so that they can be fixed before committing code.

SonarLint Installation in Eclipse IDE:

Go to eclipse market place



Eclipse Marketplace

Select solutions to install. Press Install Now to proceed with installation.
Press the "more info" link to learn more about a solution.

Find: All Markets All Categories

Featured

Vaadin Plugin for Eclipse

Promoted - Vaadin is an open source Java UI library for creating rich web user interfaces. Using its component-based API developers can create stunning web applications in... **more info**

by Vaadin, Apache 2.0
java web java ee vaadin pwa ...

252 Installs: **217K** (1,141 last month) **Learn more**

Snyk Security Scanner

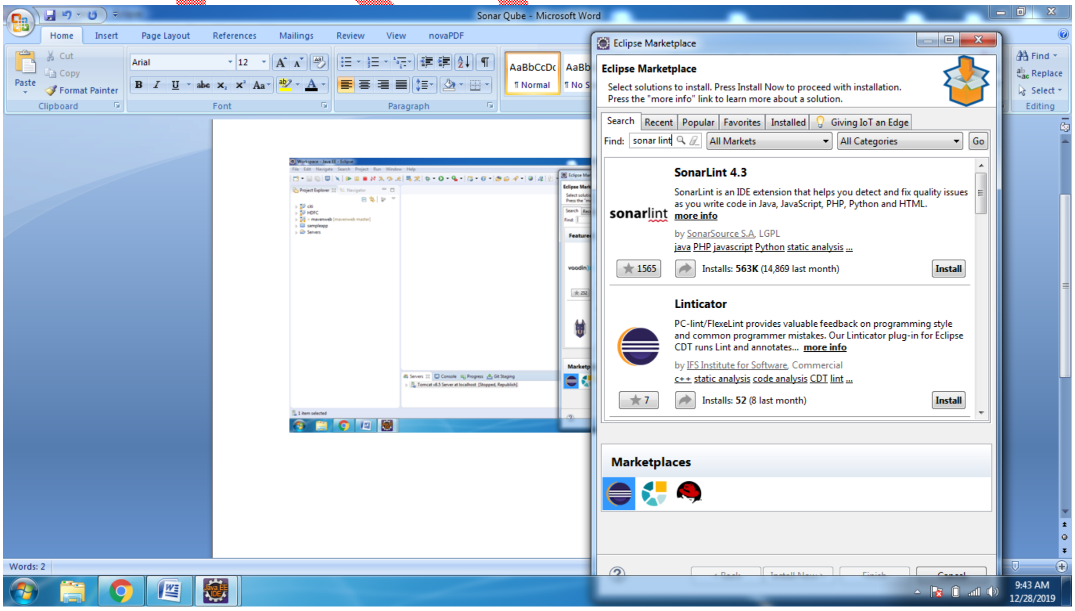
Promoted - Snyk is a free security vulnerability scanner that tests your project's open source dependencies, finds and helps fix security issues in your Eclipse workspace... **more info**

by Snyk, Apache 2.0
security scanner dependency scanning open source java scala ...

Marketplaces

Tomcat v8.5 Server at localhost [Stopped, Republish]

Search for sonar lint plugin and install



Eclipse Marketplace

Select solutions to install. Press Install Now to proceed with installation.
Press the "more info" link to learn more about a solution.

Find: sonar lint All Markets All Categories

SonarLint 4.3

SonarLint is an IDE extension that helps you detect and fix quality issues as you write code in Java, JavaScript, PHP, Python and HTML. **more info**

by SonarSource S.A, LGPL
java PHP javascript Python static analysis ...

1565 Installs: **563K** (14,869 last month) **Install**

Linticator

PC-lint/FlexeLint provides valuable feedback on programming style and common programmer mistakes. Our Linticator plug-in for Eclipse CDT runs Lint and annotates... **more info**

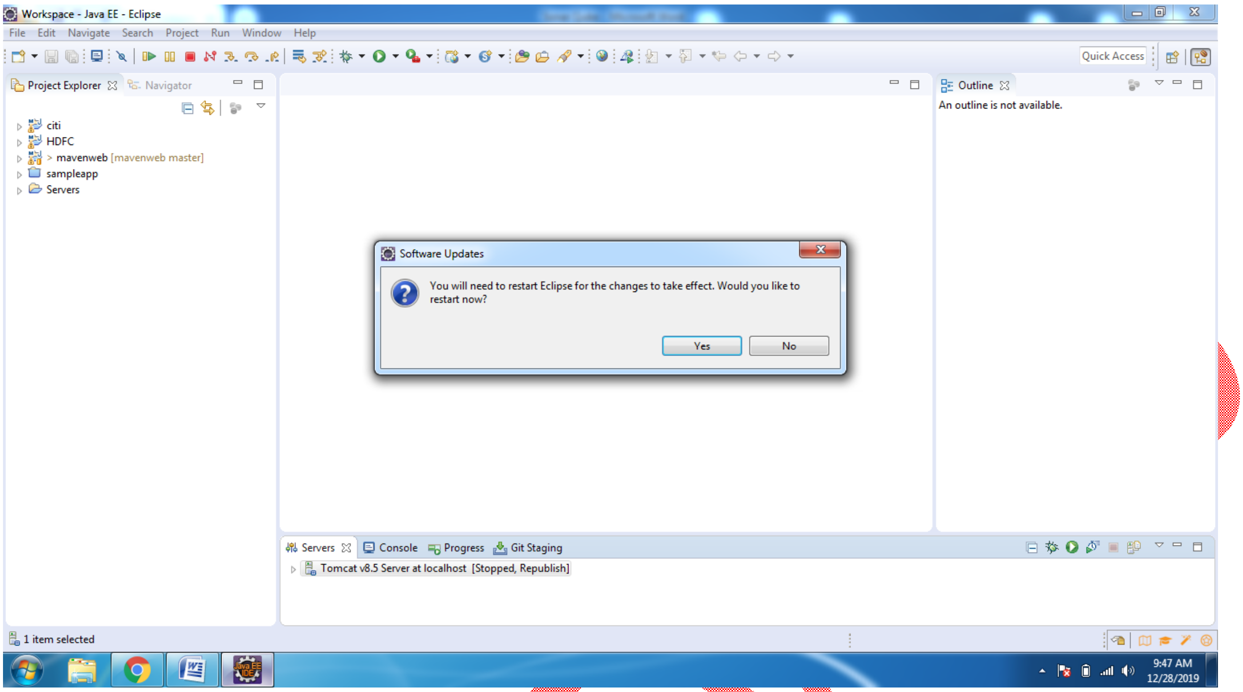
by IFS Institute for Software, Commercial
c++ static analysis code analysis CDT lint ...

7 Installs: **52** (8 last month) **Install**

Marketplaces

...... Prepared by PSDDEVOPS

Restart the eclipse



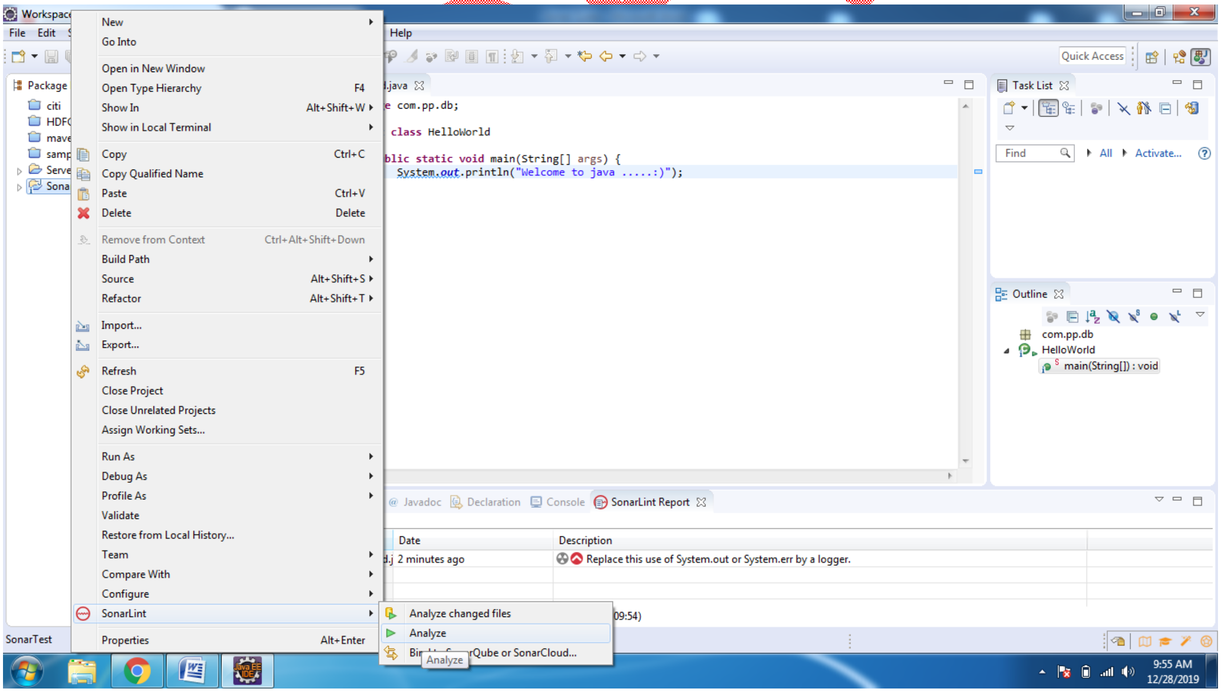
An outline is not available.

Software Updates

You will need to restart Eclipse for the changes to take effect. Would you like to restart now?

Tomcat v8.5 Server at localhost [Stopped, Republish]

Enable the Sonar lint



```
e com.pp.db;

 class HelloWorld

blic static void main(String[] args) {
 System.out.println("Welcome to java .....:)");
```

Date	Description
2 minutes ago	Replace this use of System.out or System.err by a logger.

SonarQube:

SonarQube is a open-source platform for continuous inspection of code quality developed by SonarSource.It's written in java language and it's supports various languages.

Features:

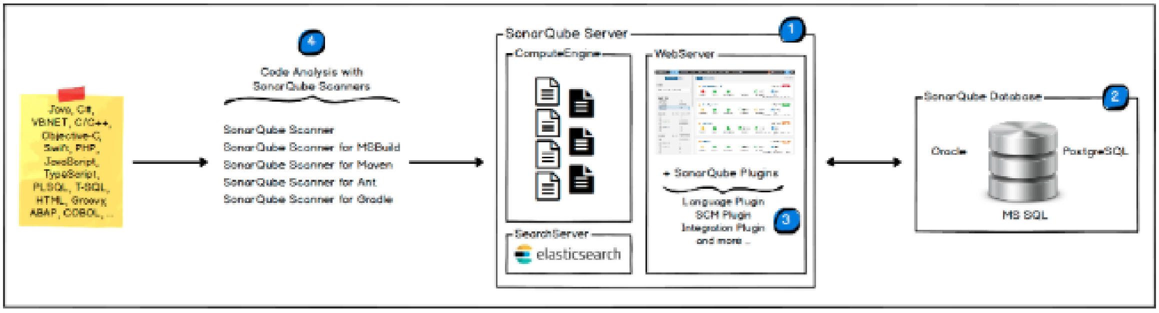
- Open source
- Initially it is developed only for Java projects.
- Today it is supporting over **20+** languages.
- I will generate the report if any issues in html format/PDF format.
- It provides the historical reports. (Compare with old and new reports)
- Integrated with Maven, Ant ,Jenkins and Gradle.
- It is a web based tool supports multiple languages (Java, C#, JS ...)
- It will support multi OS platform (Windows. MAC, Linux...).
- It will support multiple databases (MySQL, Oracle, Microsoft SQL Server, PostgreSQL ...),The default database is h2.
- It will identify the below category of issues.
    - Duplicated code
    - Coding standards
    - Unit tests
    - Complex code
    - Comments
    - Potential Bugs
    - Architecture & Design
- Commercial: ABAP, Cfamily(C, C++, and Objective-C), COBOL, PL/SQL, Visual Basic, Natural, VB.Net, RPG, Swift)
- Open source: Java, Java Script, C#, Web(HTML, JSP, JSF, ..) XML, Python, Groovy, PHP, Puppet, Lua, Groovy, FxCop, Flex, Erlang …

Pre-requisites

- The only prerequisite for running SonarQube is to have Java (Oracle JRE 8 onwards or OpenJDK 8 onwards) installed on your machine.
- MySQL DB 5.6 onwards – Optional
- Oracle 11g onwards – Optional

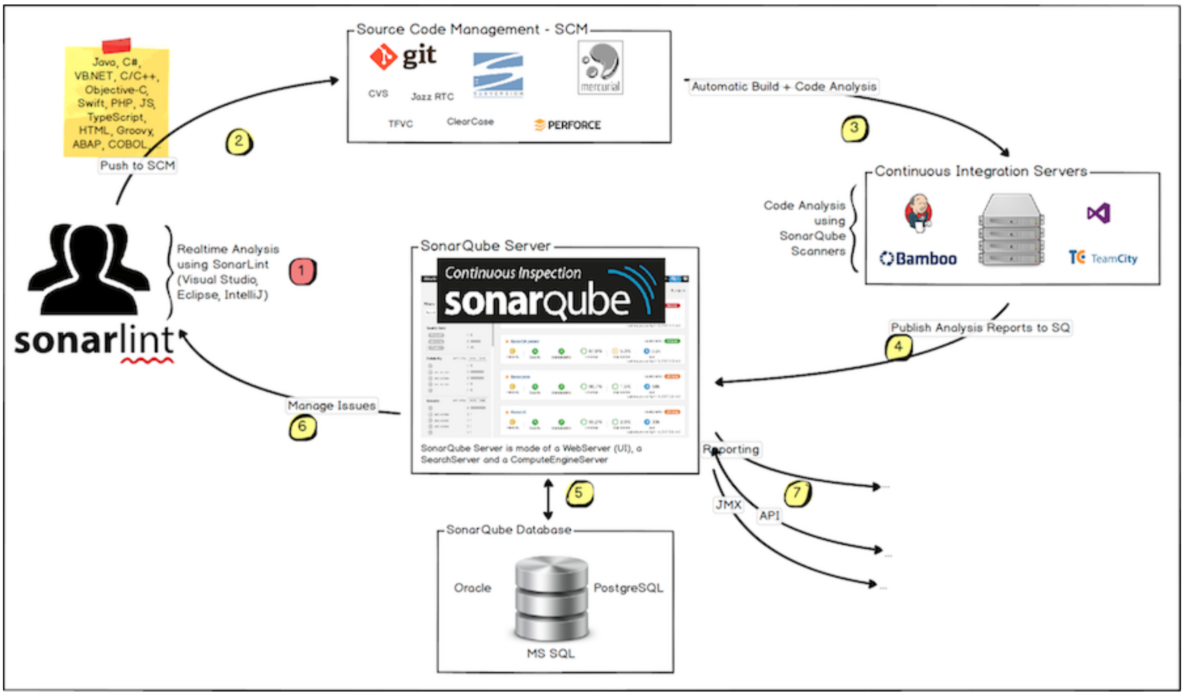
SonarQube Details	
Type	Source Continuous Code Quality
Vendor	SonarQube
Open Source	Yes
Version	8.x+
Operating system	Cross Platform
Software Download	URL http://www.sonarqube.org/downloads/
Is executable software	No, download as zip, extract and use it
Reference Websites	https://docs.sonarqube.org/

Architecture


Java, C#, VB.NET, C/C++, Objective-C, Swift, PHP, JavaScript, TypeScript, PLSQL, T-SQL, HTML, Groovy, ABAP, COBOL, ...
4
Code Analysis with SonarQube Scanners
SonarQube Scanner
SonarQube Scanner for MSBuild
SonarQube Scanner for Maven
SonarQube Scanner for Ant
SonarQube Scanner for Gradle
SonarQube Server
1
ComputeEngine
WebServer
+ SonarQube Plugins
Language Plugin
SCM Plugin
Integration Plugin
and more ...
3
SearchServer
elasticsearch
SonarQube Database
2
Oracle
PostgreSQL
MS SQL

1. One SonarQube Server starting 3 main processes:
   - Web Server for developers, managers to browse quality snapshots and configure the SonarQube instance
   - Search Server based on Elasticsearch to back searches from the UI
   - Compute Engine Server in charge of processing code analysis reports and saving them in the SonarQube Database
2. One SonarQube Database to store:
   - the configuration of the SonarQube instance (security, plugins settings, etc.)
   - the quality snapshots of projects, views, etc.
3. Multiple SonarQube Plugins installed on the server, possibly including language, SCM, integration, authentication, and governance plugins
4. One or more SonarScanners running on your Build / Continuous Integration Servers to analyze projects

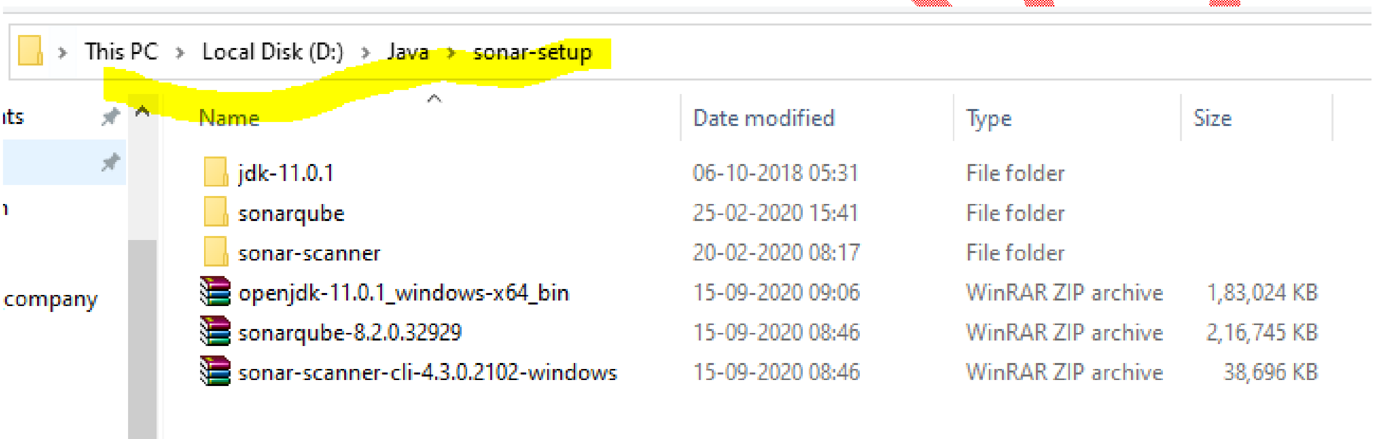
Integration


Java, C#, VBNET, C/C++, Objective-C, Swift, PHP, JS, TypeScript, HTML, Groovy, ABAP, COBOL
Push to SCM
Source Code Management - SCM
git
CVS
Jazz RTC
TFVC
ClearCase
PERFORCE
Automatic Build + Code Analysis
Continuous Integration Servers
Code Analysis using SonarQube Scanners
Bamboo
TeamCity
Realtime Analysis using SonarLint (Visual Studio, Eclipse, IntelliJ)
sonarlint
SonarQube Server
Continuous Inspection
sonarqube
SonarQube Server is made of a WebServer (UI), a SearchServer and a ComputeEngineServer
Publish Analysis Reports to SQ
Manage Issues
Reporting
JMX
API
SonarQube Database
Oracle
PostgreSQL
MS SQL
1
2
3
4
5
6
7

1. Developers code in their IDEs and use SonarLint to run local analysis.
2. Developers push their code into their favourite SCM : git, SVN, TFVC, ...
3. The Continuous Integration Server triggers an automatic build, and the execution of the SonarScanner/Runner required to run the SonarQube analysis.
4. The analysis report is sent to the SonarQube Server for processing.
5. SonarQube Server processes and stores the analysis report results in the SonarQube Database, and displays the results in the UI.
6. Developers review, comment, challenge their Issues to manage and reduce their Technical Debt through the SonarQube UI.
7. Managers receive Reports from the analysis. Ops use APIs to automate configuration and extract data from SonarQube. Ops use JMX to monitor SonarQube Server.

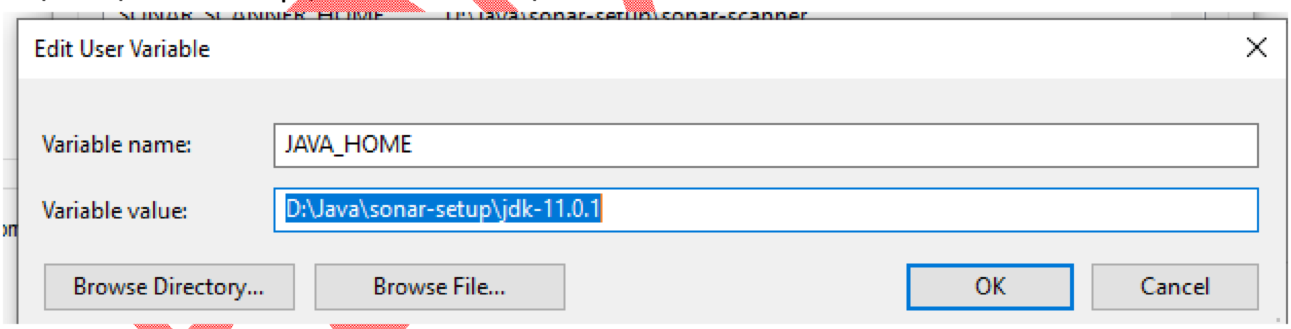
Installation on Windows:

1. Download and extract the SonarQube server from
   https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-8.2.0.32929.zip
2. Download and extract the Sonar scanner from
   https://binaries.sonarsource.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-4.3.0.2102-windows.zip
3. Download JDK 11+ version and extract and set path
   https://download.java.net/java/ga/jdk11/openjdk-11_windows-x64_bin.zip


This PC › Local Disk (D:) › Java › sonar-setup

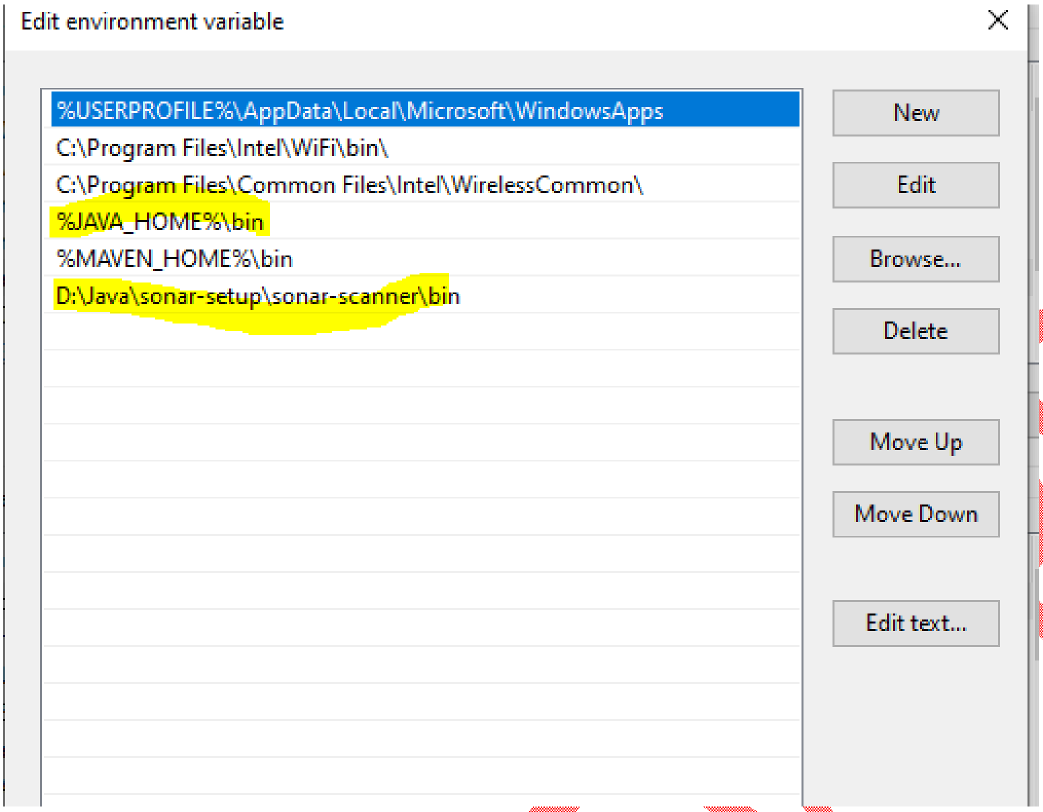
Name	Date modified	Type	Size
jdk-11.0.1	06-10-2018 05:31	File folder	
sonarqube	25-02-2020 15:41	File folder	
sonar-scanner	20-02-2020 08:17	File folder	
openjdk-11.0.1_windows-x64_bin	15-09-2020 09:06	WinRAR ZIP archive	1,83,024 KB
sonarqube-8.2.0.32929	15-09-2020 08:46	WinRAR ZIP archive	2,16,745 KB
sonar-scanner-cli-4.3.0.2102-windows	15-09-2020 08:46	WinRAR ZIP archive	38,696 KB

4. Setup the path for sonar scanner in environment variable
D:\Java\sonar-setup\sonar-scanner\bin


Edit User Variable

Variable name: JAVA_HOME

Variable value: D:\Java\sonar-setup\jdk-11.0.1



```
%USERPROFILE%\AppData\Local\Microsoft\WindowsApps
C:\Program Files\Intel\WiFi\bin\
C:\Program Files\Common Files\Intel\WirelessCommon\
%JAVA_HOME%\bin
%MAVEN_HOME%\bin
D:\Java\sonar-setup\sonar-scanner\bin
```

5. Update the sonar-scanner.properties file under D:\Java\sonar-setup

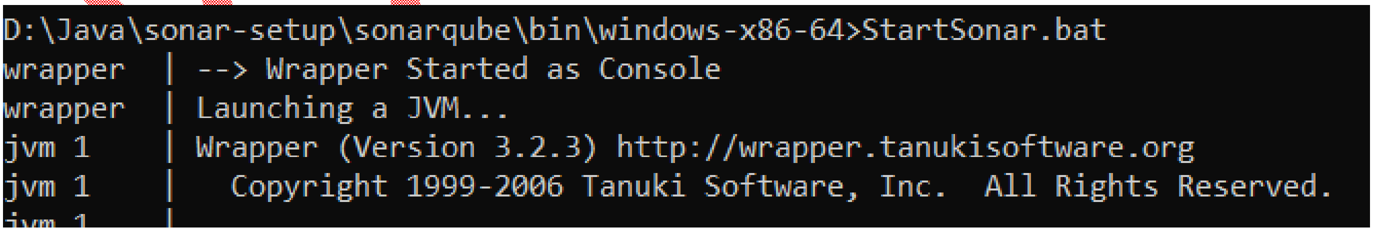
    sonar.projectKey=JavaProject

    sonar.projectName=JavaProject

    sonar.projectVersion=1.0

    sonar.sources=D:/Workspace/mavenweb/src/main/java/com/pp
6. Run the SonarQube server.

    Cmd> D:\Java\sonar-setup\sonarqube\bin\windows-x86-64>StartSonar.bat

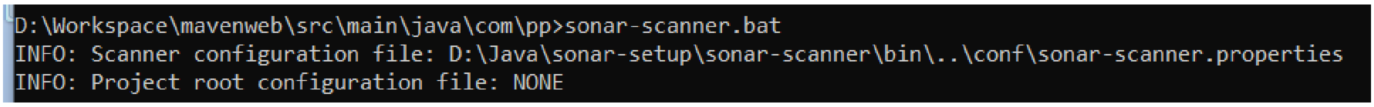


```
D:\Java\sonar-setup\sonarqube\bin\windows-x86-64>StartSonar.bat
wrapper  | --> Wrapper Started as Console
wrapper  | Launching a JVM...
jvm 1    | Wrapper (Version 3.2.3) http://wrapper.tanukisoftware.org
jvm 1    |   Copyright 1999-2006 Tanuki Software, Inc.  All Rights Reserved.
jvm 1    |
```

7. Run the sonarqube server http://localhost:9000
8. Run the Sonarqube scanner to analyze the reports.

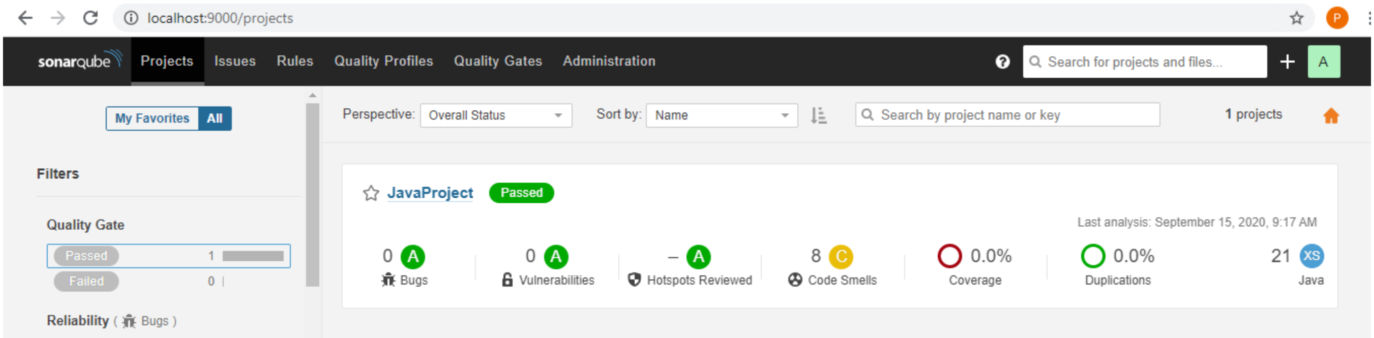
    cmd>D D:\Workspace\mavenweb\src\main\java\com\pp>sonar-scanner.bat

...... Prepared by PSDDEVOPS



```
D:\Workspace\mavenweb\src\main\java\com\pp>sonar-scanner.bat
INFO: Scanner configuration file: D:\Java\sonar-setup\sonar-scanner\bin\..\conf\sonar-scanner.properties
INFO: Project root configuration file: NONE
```

9. Login to the server admin\admin



Installation on Ubuntu:

1. Download SonarQube server from https://www.sonarqube.org/downloads/

   sudo apt-get update

   sudo add-apt-repository ppa:openjdk-r/ppa

   sudo apt-get install openjdk-8-jdk

   cd /opt

   sudo wget https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-6.2.zip

   sudo unzip sonarqube-6.2.zip

   sudo mv sonarqube-6.2/ sonarqube

   sudo chmod 777 -R sonarqube
2. Run the SonarQube server

   cd /opt/sonarqube/bin/linux-x86-64

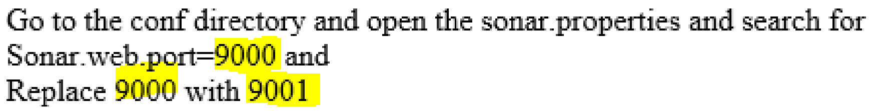
   sh sonar.sh start

3. Install the sonar scanner

   wget https://binaries.sonarsource.com/Distribution/sonar-scanner-cli/sonar-scanner-2.8.zip

Changing the port number

```
cd /opt/sonarqube/
```



Go to the conf directory and open the sonar.properties and search for
Sonar.web.port=9000 and
Replace 9000 with 9001

Sonar Integration with Maven:

1. Download and configure maven in your system.
2. Open settings.xml and update the few entries.

```
<pluginGroups>
    <pluginGroup>org.sonarsource.scanner.maven</pluginGroup>
  </pluginGroups>

<profile>
   <id>sonar</id>
   <activation>
   <activeByDefault>true</activeByDefault>
   </activation>
   <properties>
   <!-- Optional URL to server. Default value is http://localhost:9000 -->
   <sonar.host.url> http://localhost:9000 </sonar.host.url>
 <!--  <sonar.jdbc.url>your database URL</sonar.jdbc.url>
<sonar.jdbc.driver>Your database driver</sonar.jdbc.driver>
<sonar.jdbc.username>DB username</sonar.jdbc.username>
<sonar.jdbc.password>db password</sonar.jdbc.password>       -->
    </properties>
</profile>
```

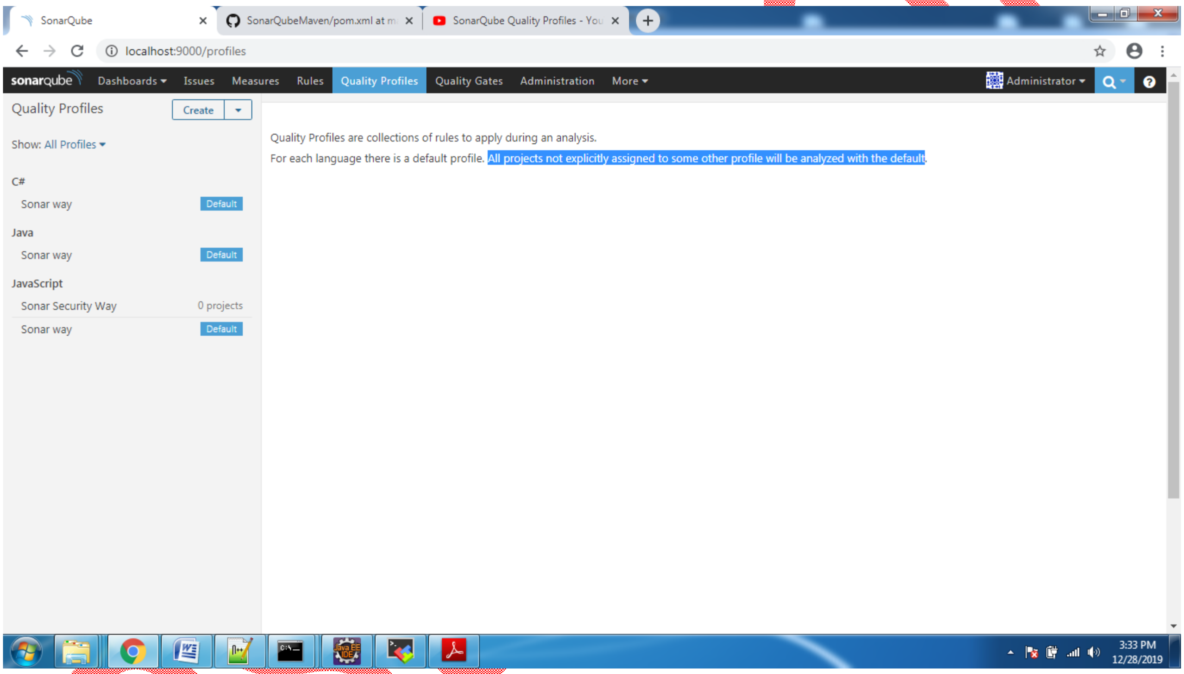
3. Create the maven based project and use.If we found any exception update the pom.xml file. mvn install sonar:sonar

```
<plugin>
      <groupId>org.sonarsource.scanner.maven</groupId>
      <artifactId>sonar-maven-plugin</artifactId>
      <version>3.3.0.603</version>
 </plugin>
```

Quality Profiles:

- Quality Profiles are set of rules which are available in the SonarQube.
- Here we can define set of rules which needs to be analyzed particular language.
- Each language contains default profile name as ***sonar way***.
- Sonar way contains the active rules.
- All projects not explicitly assigned to some other profile will be analyzed with the default.
- By default C#,java and java script will be available.

Click on Quality Profiles →



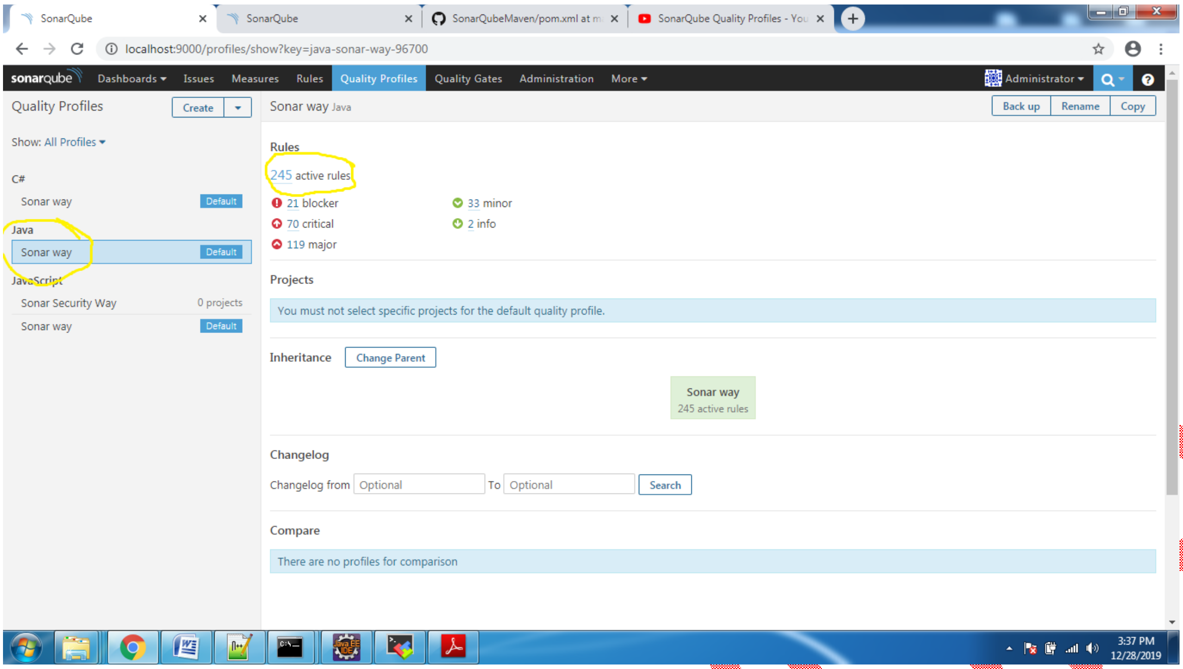
Quality Profiles

Quality Profiles are collections of rules to apply during an analysis.

For each language there is a default profile. All projects not explicitly assigned to some other profile will be analyzed with the default.

C#	
Sonar way	Default
Java	
Sonar way	Default
JavaScript	
Sonar Security Way	0 projects
Sonar way	Default

Click on Java Sonar way contains list of rules



localhost:9000/profiles/show?key=java-sonar-way-96700

Quality Profiles

Show: All Profiles

C#
- Sonar way — Default

Java
- Sonar way — Default

JavaScript
- Sonar Security Way — 0 projects
- Sonar way — Default

Sonar way Java

Rules

245 active rules

21 blocker	33 minor
70 critical	2 info
119 major	

Projects

You must not select specific projects for the default quality profile.

Inheritance

Sonar way
245 active rules

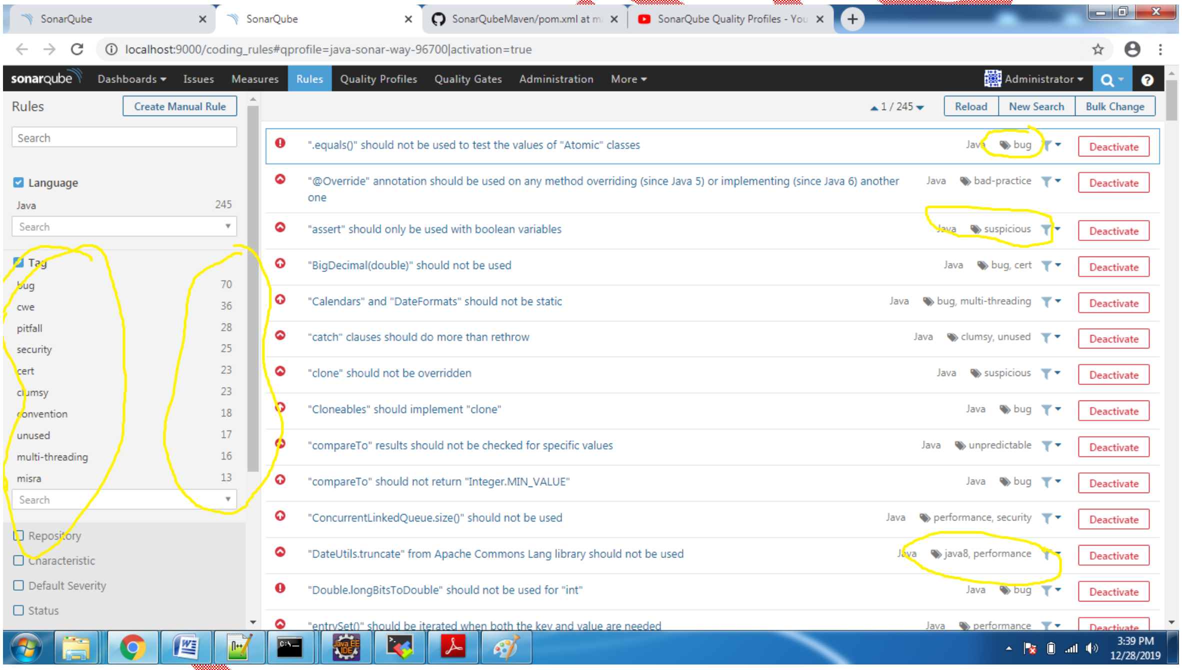
Changelog

Changelog from Optional To Optional

Compare

There are no profiles for comparison

Click on the rules it contains list of rules for default profile



localhost:9000/coding_rules#qprofile=java-sonar-way-96700|activation=true

Rules

☑ Language

Java	245

☑ Tag

bug	70
cwe	36
pitfall	28
security	25
cert	23
clumsy	23
convention	18
unused	17
multi-threading	16
misra	13

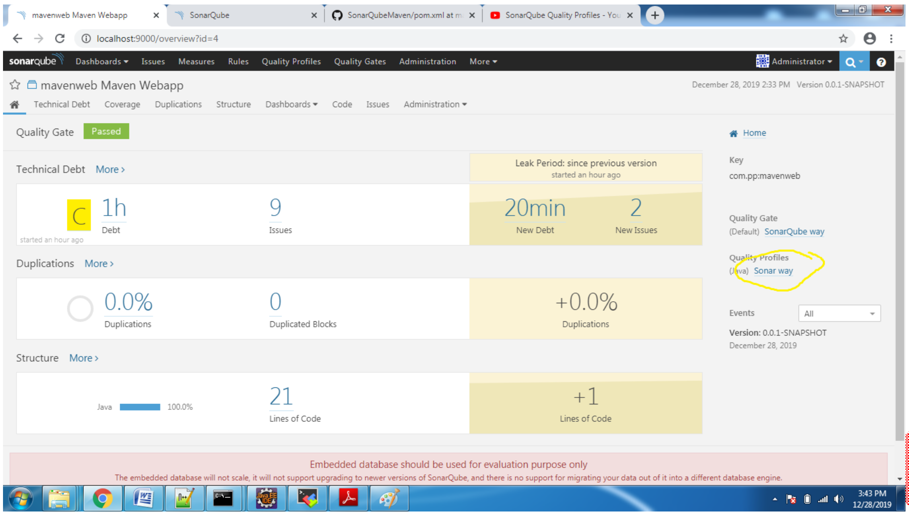
☐ Repository
☐ Characteristic
☐ Default Severity
☐ Status

1 / 245

Rule	Language	Tags
".equals()" should not be used to test the values of "Atomic" classes	Java	bug
"@Override" annotation should be used on any method overriding (since Java 5) or implementing (since Java 6) another one	Java	bad-practice
"assert" should only be used with boolean variables	Java	suspicious
"BigDecimal(double)" should not be used	Java	bug, cert
"Calendars" and "DateFormats" should not be static	Java	bug, multi-threading
"catch" clauses should do more than rethrow	Java	clumsy, unused
"clone" should not be overridden	Java	suspicious
"Cloneables" should implement "clone"	Java	bug
"compareTo" results should not be checked for specific values	Java	unpredictable
"compareTo" should not return "Integer.MIN_VALUE"	Java	bug
"ConcurrentLinkedQueue.size()" should not be used	Java	performance, security
"DateUtils.truncate" from Apache Commons Lang library should not be used	Java	java8, performance
"Double.longBitsToDouble" should not be used for "int"	Java	bug
"entrySet()" should be iterated when both the key and value are needed	Java	performance

In our project we are not specifying any profile default profile will applicable.

…… Prepared by PSDDEVOPS



mavenweb Maven Webapp

December 28, 2019 2:33 PM Version 0.0.1-SNAPSHOT

Quality Gate Passed

Technical Debt More ›

Leak Period: since previous version
started an hour ago

C 1h Debt — 9 Issues — 20min New Debt — 2 New Issues

started an hour ago

Duplications More ›

0.0% Duplications — 0 Duplicated Blocks — +0.0% Duplications

Structure More ›

Java 100.0% — 21 Lines of Code — +1 Lines of Code

Key
com.pp:mavenweb

Quality Gate
(Default) SonarQube way

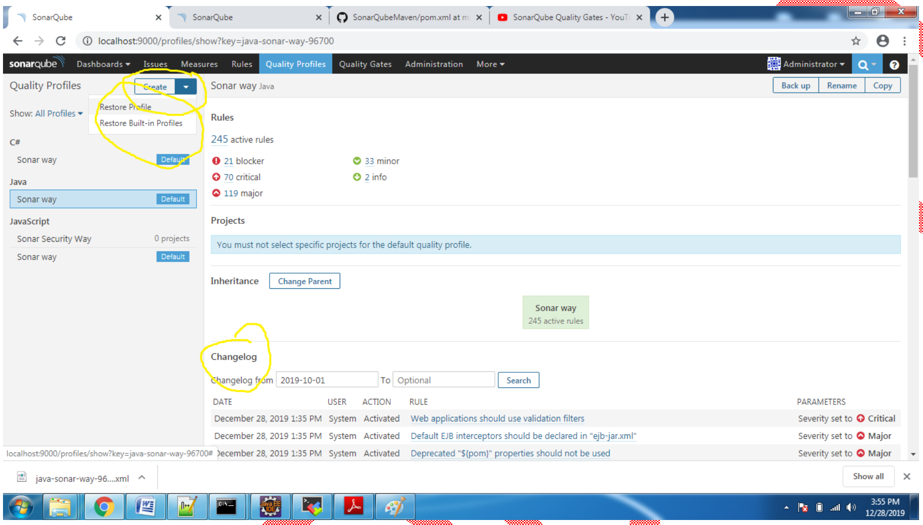
Quality Profiles
(Java) Sonar way

Events All

Version: 0.0.1-SNAPSHOT
December 28, 2019

Embedded database should be used for evaluation purpose only

The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.



Quality Profiles

C#
Sonar way Default

Java
Sonar way Default

JavaScript
Sonar Security Way 0 projects
Sonar way Default

Sonar way Java

Rules

245 active rules

21 blocker
70 critical
119 major
33 minor
2 info

Projects

You must not select specific projects for the default quality profile.

Inheritance

Sonar way
245 active rules

Changelog

Changelog from 2019-10-01 To Optional

DATE	USER	ACTION	RULE	PARAMETERS
December 28, 2019 1:35 PM	System	Activated	Web applications should use validation filters	Severity set to Critical
December 28, 2019 1:35 PM	System	Activated	Default EJB interceptors should be declared in "ejb-jar.xml"	Severity set to Major
December 28, 2019 1:35 PM	System	Activated	Deprecated "${pom}" properties should not be used	Severity set to Major

Changelog:

- SonarQube provides changelog for every Quality Profile. It is nice, because we can track changes.

Restore Profile:

- Restore the rules from other sonar instance.

Restore Built in profile:

- Restoring the updated rules.

Backup:

- Backup the profile

Copy

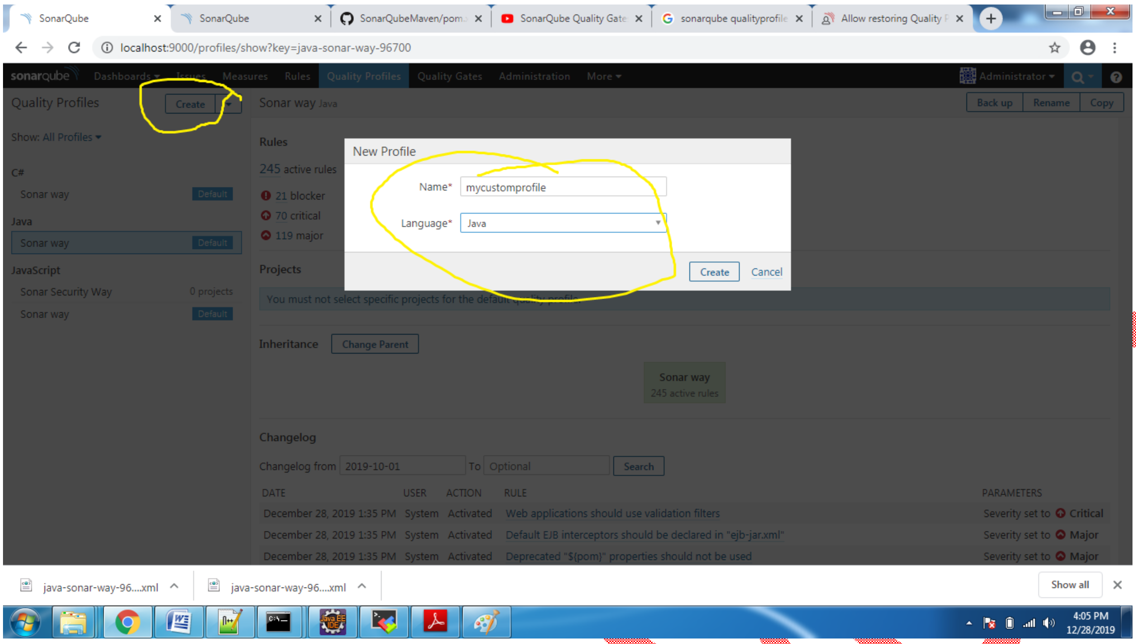
- Copy the rules from one profile to other.

Rename:

- Rename the profile

Creating the custom Quality profile

Click on create → Enter the name and language



localhost:9000/profiles/show?key=java-sonar-way-96700

Sonar way Java

Rules: 245 active rules — 21 blocker, 70 critical, 119 major

New Profile

Name*: mycustomprofile

Language*: Java

You must not select specific projects for the default quality profile.

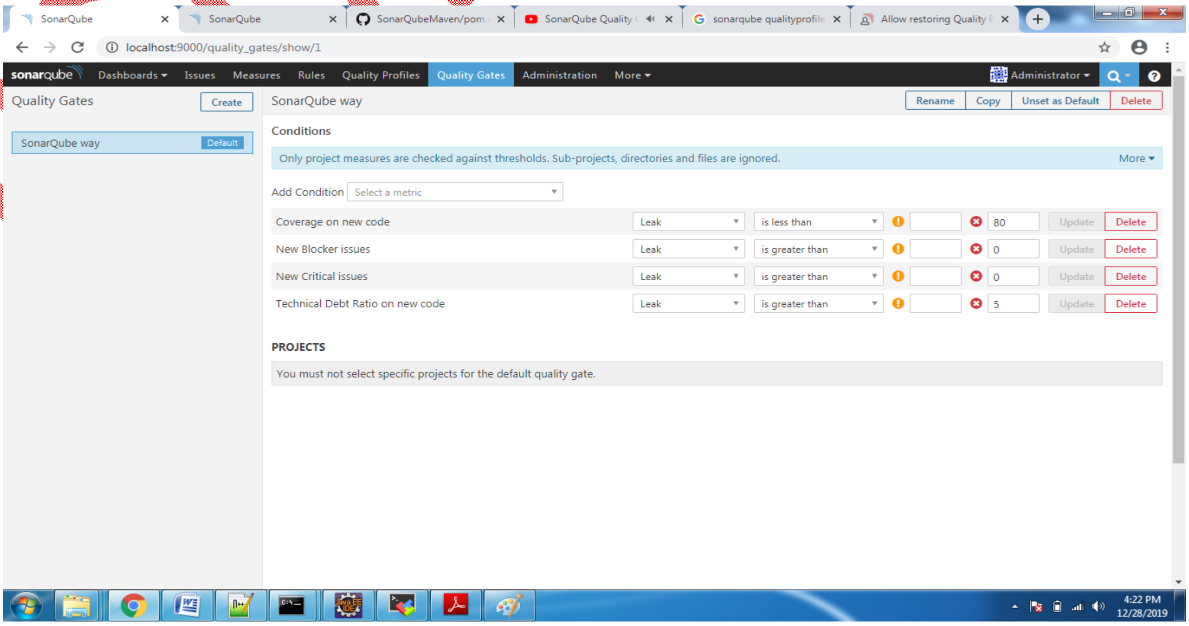
Inheritance: Sonar way, 245 active rules

Changelog from 2019-10-01 To Optional

DATE	USER	ACTION	RULE	PARAMETERS
December 28, 2019 1:35 PM	System	Activated	Web applications should use validation filters	Severity set to Critical
December 28, 2019 1:35 PM	System	Activated	Default EJB interceptors should be declared in "ejb-jar.xml"	Severity set to Major
December 28, 2019 1:35 PM	System	Activated	Deprecated "${pom}" properties should not be used	Severity set to Major

Quality Gates:

- Quality Gates are collections of simple Boolean thresholds set on project measures. A project must pass each of the thresholds in order to pass the Quality Gate as a whole.
- It is possible to set a default Quality Gate, which will be applied to all projects not explicitly assigned to some other gate.
- SonarQube way/Sonar way is the default Quality Gate.



localhost:9000/quality_gates/show/1

SonarQube way

Conditions

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

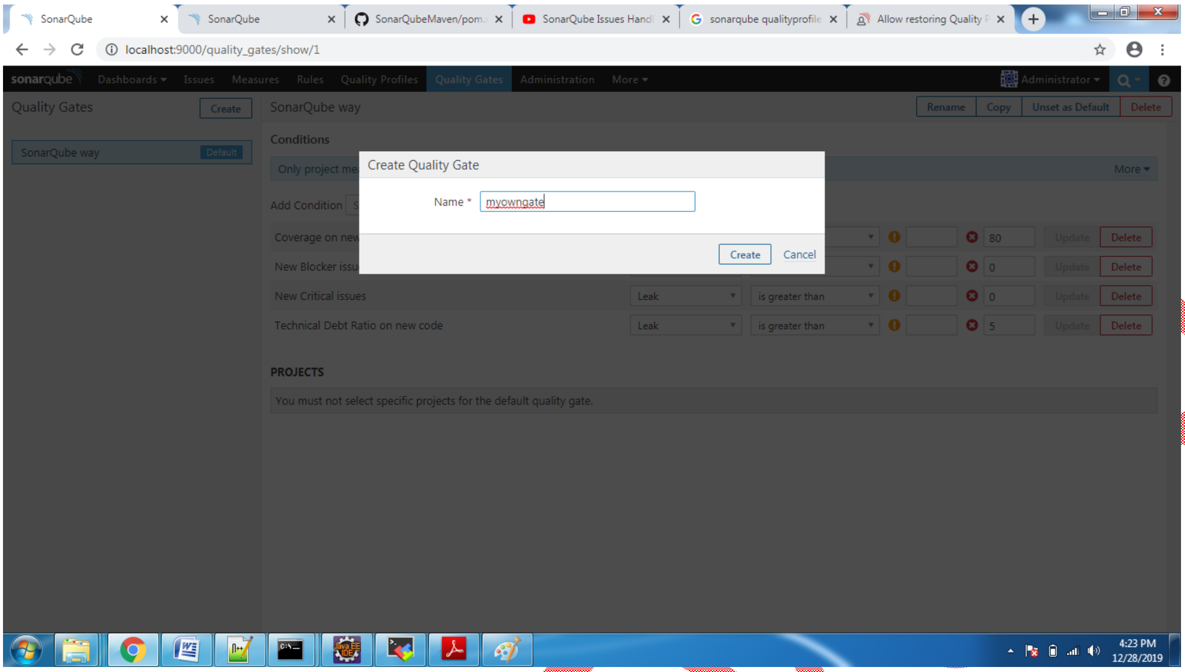
Add Condition: Select a metric

Metric	Period	Operator	Warning	Error
Coverage on new code	Leak	is less than		80
New Blocker issues	Leak	is greater than		0
New Critical issues	Leak	is greater than		0
Technical Debt Ratio on new code	Leak	is greater than		5

PROJECTS

You must not select specific projects for the default quality gate.

Creating Custom Gate:



localhost:9000/quality_gates/show/1

SonarQube way

Create Quality Gate

Name * myowngate

Coverage on new				80
New Blocker issu				0
New Critical issues	Leak	is greater than		0
Technical Debt Ratio on new code	Leak	is greater than		5

You must not select specific projects for the default quality gate.

4:23 PM
12/28/2019

Issue Types

There are three types of issues:

1. **Bug** – A coding error that will break your code and needs to be fixed immediately.
2. **Vulnerability** – A point in your code that's open to attack.
3. **Codesmell** – A maintainability issue that makes your code confusing and difficult to maintain.

Issue Severity

Each issue has one of five severities:

1. **BLOCKER**

   Bug with a high probability to impact the behavior of the application in production. memory leak, unclosed JDBC connection, ... The code MUST be immediately fixed.
2. **CRITICAL**

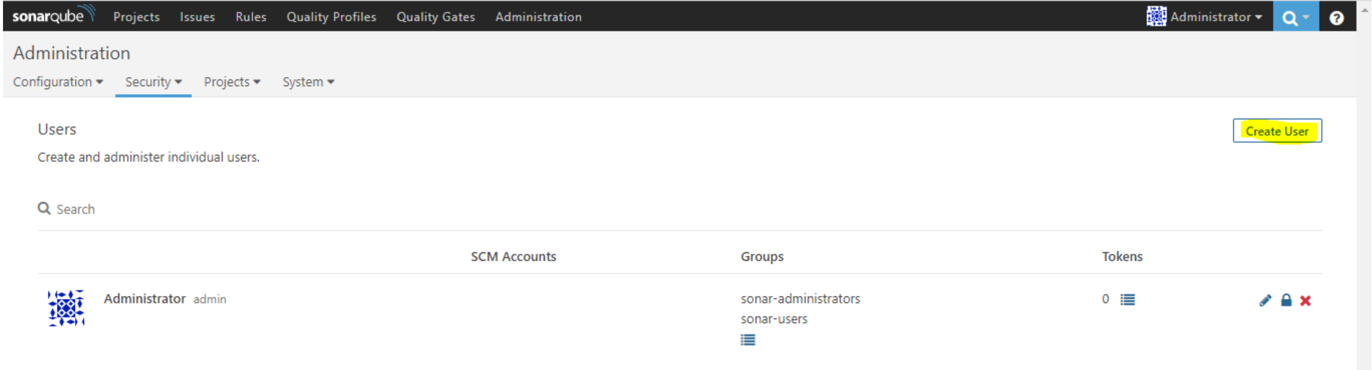
   Either a bug with a low probability to impact the behavior of the application in production or an issue which represents a security flaw: empty catch block, SQL injection, ... The code MUST be immediately reviewed.
3. **MAJOR**

   Quality flaw which can highly impact the developer productivity: uncovered piece of code, duplicated blocks, unused parameters, ...

4. **MINOR**
   Quality flaw which can slightly impact the developer productivity: lines should not be too long, "switch" statements should have at least 3 cases, ...
5. **INFO**
   Neither a bug nor a quality flaw, just a finding.

Creating the New Users

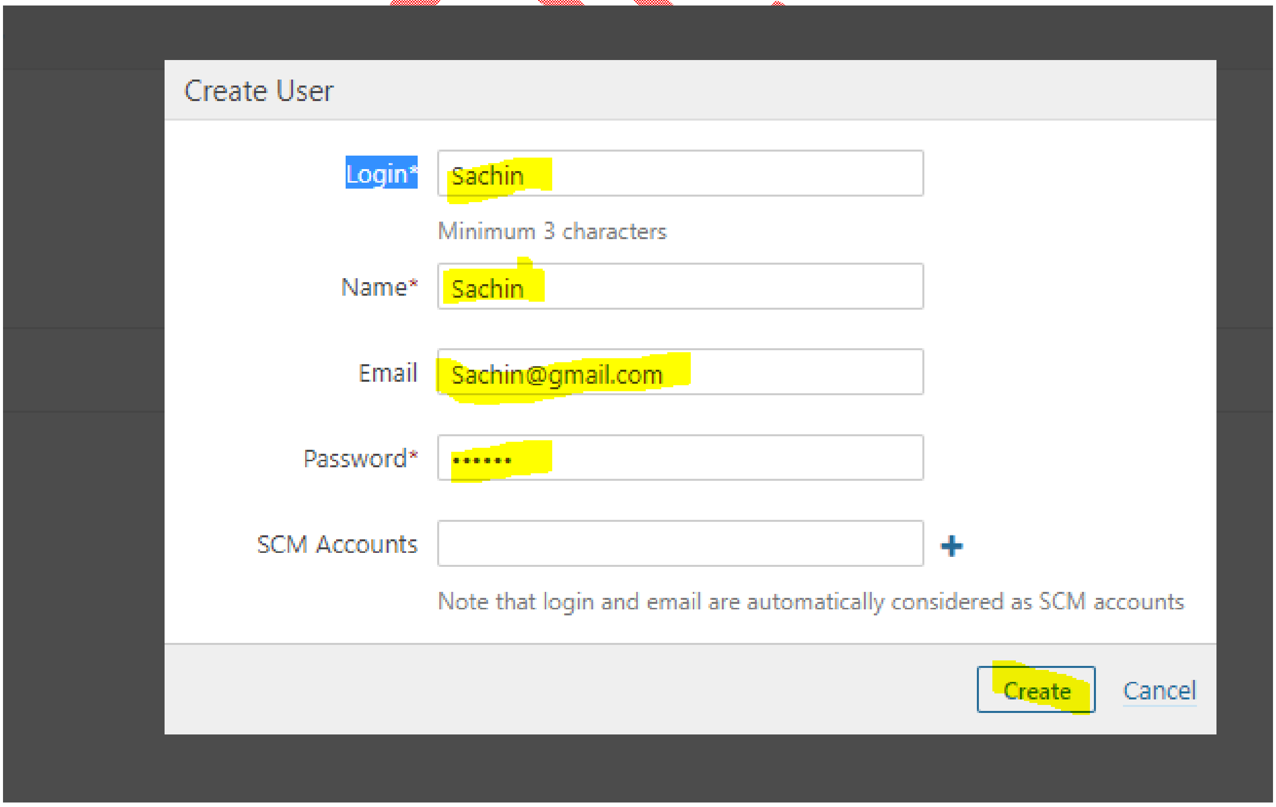
Click on Administrator → Security → Users → Create User



Users

Create and administer individual users.

	SCM Accounts	Groups	Tokens
Administrator admin		sonar-administrators sonar-users	0



Create User

Login* Sachin

Minimum 3 characters

Name* Sachin

Email Sachin@gmail.com

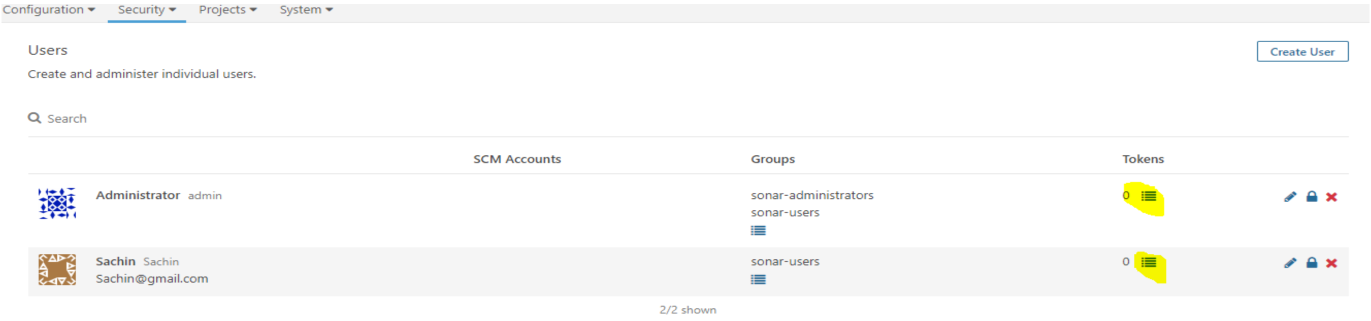
Password* ••••••

SCM Accounts

Note that login and email are automatically considered as SCM accounts

Generating Token:

Go to Users → Token



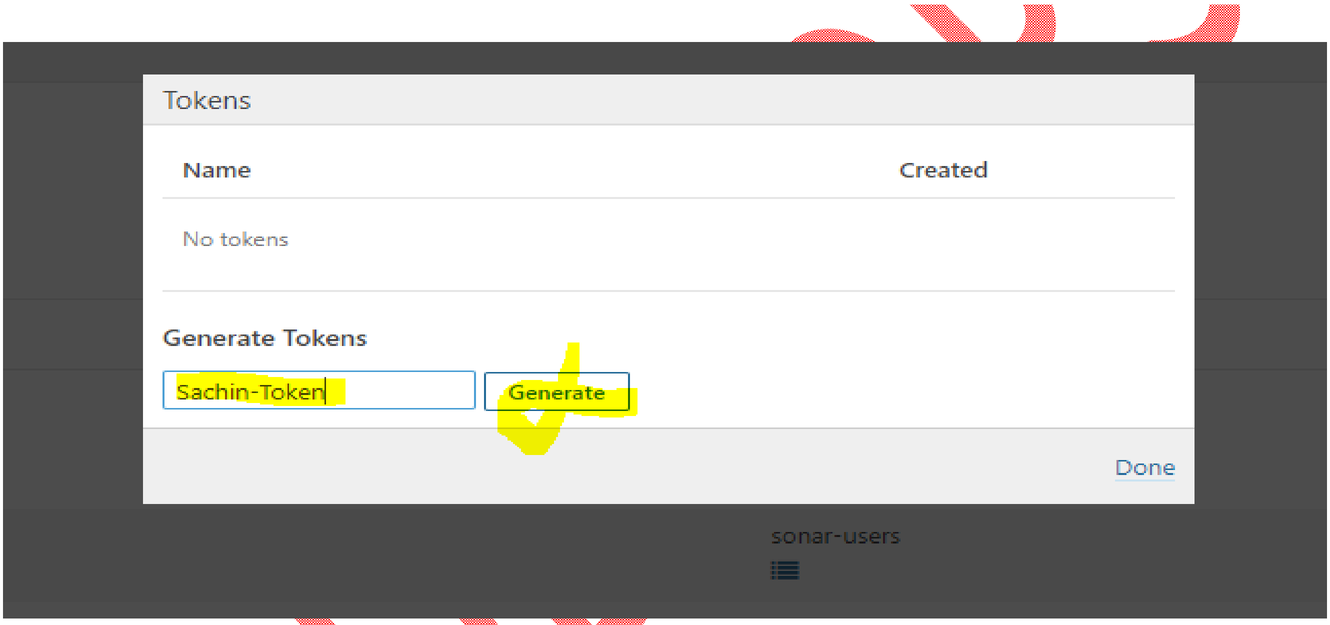
Users

Create and administer individual users.

Search

	SCM Accounts	Groups	Tokens
Administrator admin		sonar-administrators sonar-users	0
Sachin Sachin Sachin@gmail.com		sonar-users	0

2/2 shown



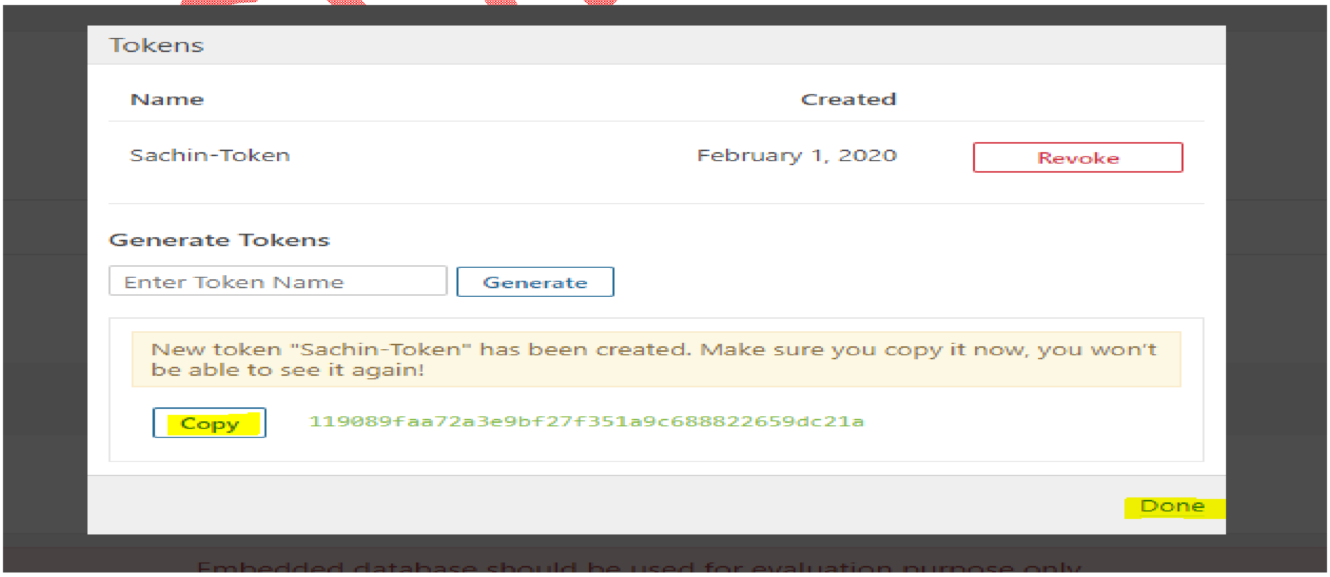
Tokens

Name	Created
No tokens	

Generate Tokens

Sachin-Token

sonar-users



Tokens

Name	Created	
Sachin-Token	February 1, 2020	Revoke

Generate Tokens

Enter Token Name

New token "Sachin-Token" has been created. Make sure you copy it now, you won't be able to see it again!

119089faa72a3e9bf27f351a9c688822659dc21a

Embedded database should be used for evaluation purpose only

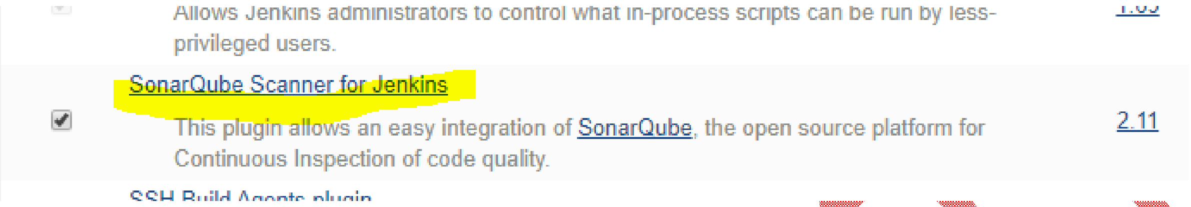
119089faa72a3e9bf27f351a9c688822659dc21a

…… Prepared by PSDDEVOPS

Integration with Jenkins

Install Sonarqube-scanner plugin

Go to Manage Jenkins → manage Plugins → Sonarqube-scanner



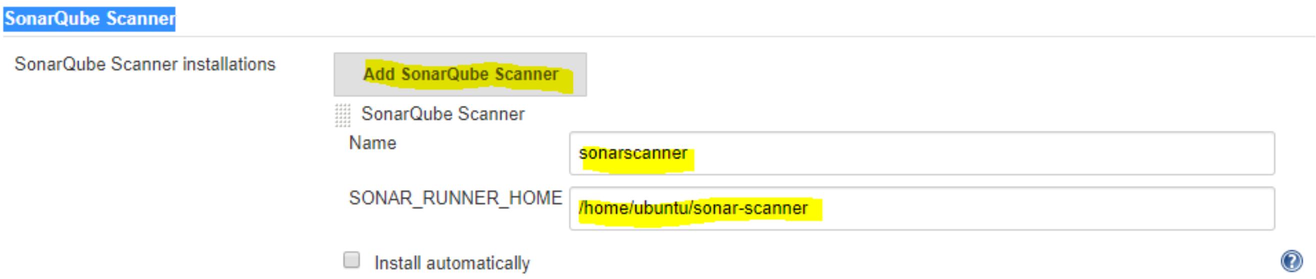
Allows Jenkins administrators to control what in-process scripts can be run by less-privileged users.

☑ SonarQube Scanner for Jenkins — 2.11

This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality.

SSH Build Agents plugin

Go to Global Tool Configuration → SonarQube Scanner → Add scanner details



SonarQube Scanner

SonarQube Scanner installations: Add SonarQube Scanner

SonarQube Scanner

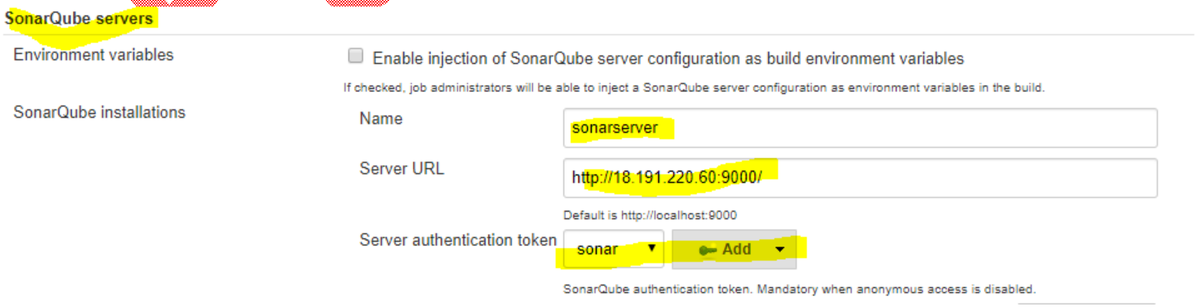
Name: sonarscanner

SONAR_RUNNER_HOME: /home/ubuntu/sonar-scanner

☐ Install automatically

Go to Configure System → Add SonarQube servers

Add the authentication token



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: sonarserver

Server URL: http://18.191.220.60:9000/

Default is http://localhost:9000

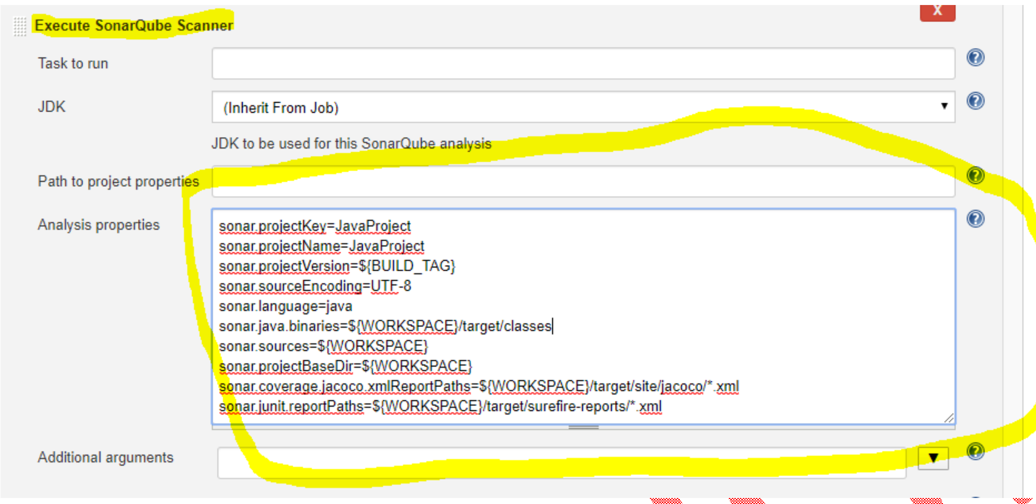
Server authentication token: sonar ▼ Add ▼

SonarQube authentication token. Mandatory when anonymous access is disabled.

…… Prepared by PSDDEVOPS

Go to sonar project configure below details.

Go to Build → Add build step → Select Execute Sonar Qube Scanner



Execute SonarQube Scanner

Task to run

JDK (Inherit From Job)

JDK to be used for this SonarQube analysis

Path to project properties

Analysis properties

```
sonar.projectKey=JavaProject
sonar.projectName=JavaProject
sonar.projectVersion=${BUILD_TAG}
sonar.sourceEncoding=UTF-8
sonar.language=java
sonar.java.binaries=${WORKSPACE}/target/classes
sonar.sources=${WORKSPACE}
sonar.projectBaseDir=${WORKSPACE}
sonar.coverage.jacoco.xmlReportPaths=${WORKSPACE}/target/site/jacoco/*.xml
sonar.junit.reportPaths=${WORKSPACE}/target/surefire-reports/*.xml
```

Additional arguments

Analysis properties:

sonar.projectKey=JavaProject

sonar.projectName=JavaProject

sonar.projectVersion=${BUILD_TAG}

sonar.sourceEncoding=UTF-8

sonar.language=java

sonar.java.binaries=${WORKSPACE}/target/classes

sonar.sources=${WORKSPACE}

sonar.projectBaseDir=${WORKSPACE}

sonar.coverage.jacoco.xmlReportPaths=${WORKSPACE}/target/site/jacoco/*.xml

sonar.junit.reportPaths=${WORKSPACE}/target/surefire-reports/*.xml

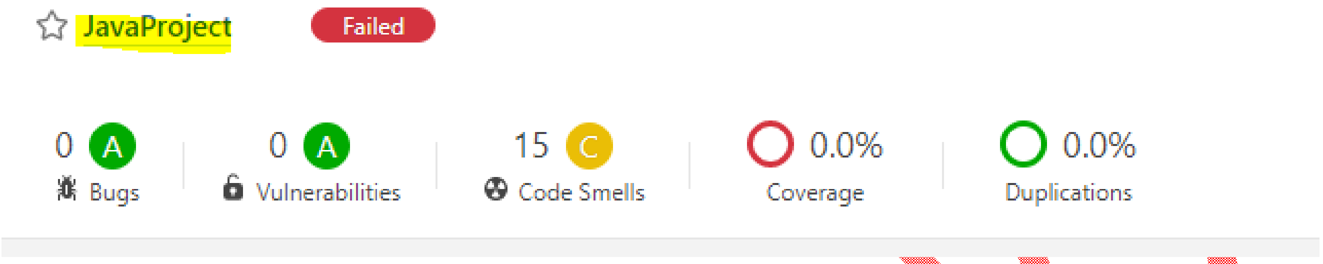
Exclusions

sonar.exclusions=**/test.java

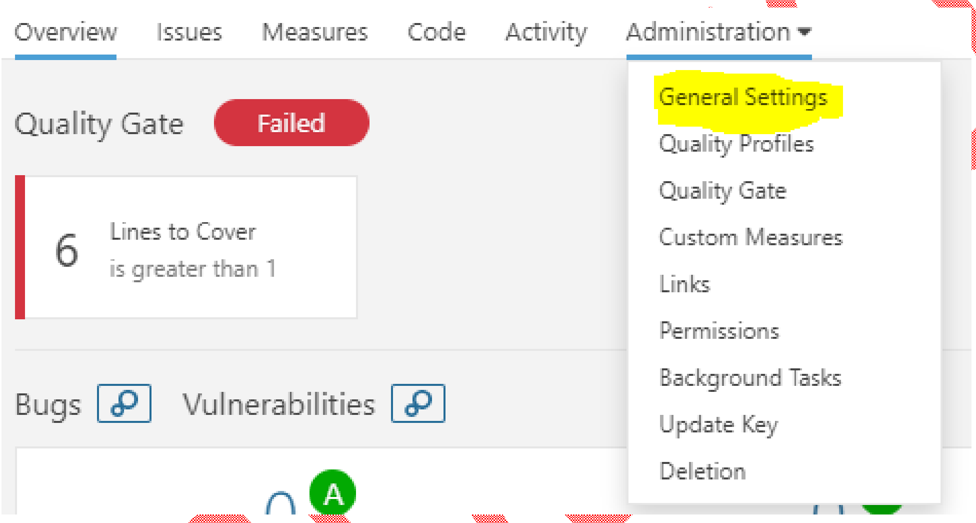
…… Prepared by PSDDEVOPS

Create the web-hook in Sonar Qube Server Side

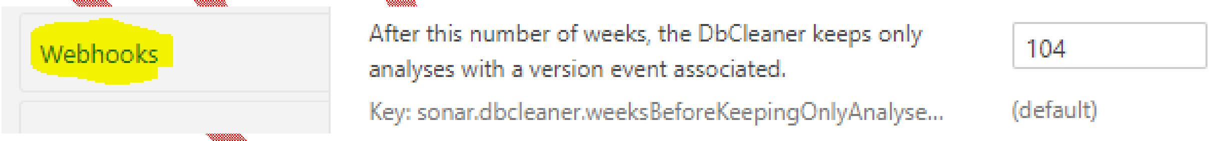
Go to home page → Select project



Goto Administration → General Settings



Click on webhooks



After this number of weeks, the DbCleaner keeps only analyses with a version event associated.

Key: sonar.dbcleaner.weeksBeforeKeepingOnlyAnalyse...

104

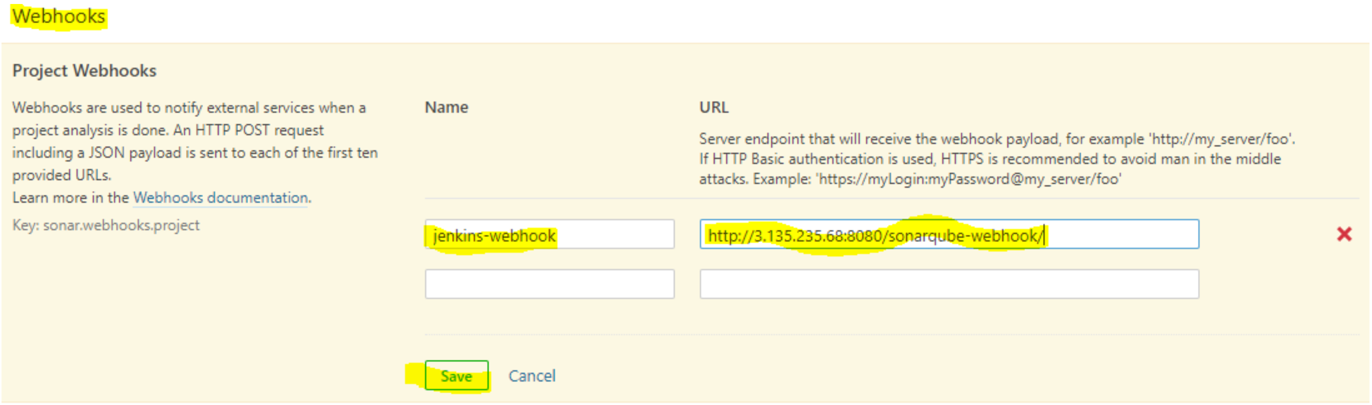
(default)

Enter the webhook Name and URL and click on Save.

Name: jenkins-webhook

URL: http://3.135.235.68:8080/sonarqube-webhook/

…… Prepared by PSDDEVOPS



Webhooks

Project Webhooks

Webhooks are used to notify external services when a project analysis is done. An HTTP POST request including a JSON payload is sent to each of the first ten provided URLs.
Learn more in the Webhooks documentation.

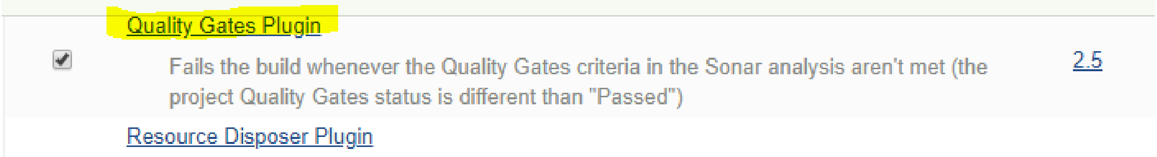
Key: sonar.webhooks.project

Name	URL
	Server endpoint that will receive the webhook payload, for example 'http://my_server/foo'. If HTTP Basic authentication is used, HTTPS is recommended to avoid man in the middle attacks. Example: 'https://myLogin:myPassword@my_server/foo'
jenkins-webhook	http://3.135.235.68:8080/sonarqube-webhook/

Configuring Sonar Gate plugin in Jenkins side

Install Quality Gates Plugin/Sonar Quality Gates Plugin from Jenkins side

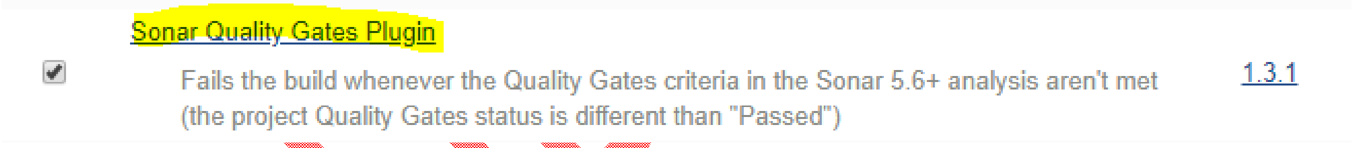
Go to Manage Jenkins → Manage plugins → Install



☑ Quality Gates Plugin 2.5

Fails the build whenever the Quality Gates criteria in the Sonar analysis aren't met (the project Quality Gates status is different than "Passed")

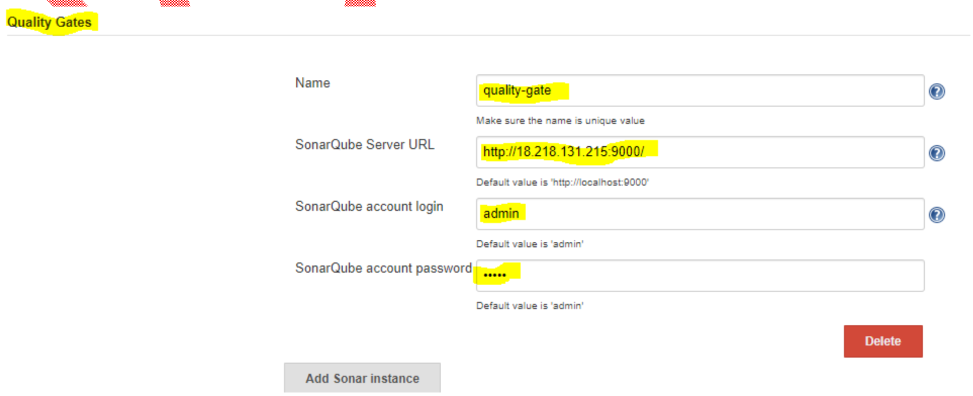
Resource Disposer Plugin



☑ Sonar Quality Gates Plugin 1.3.1

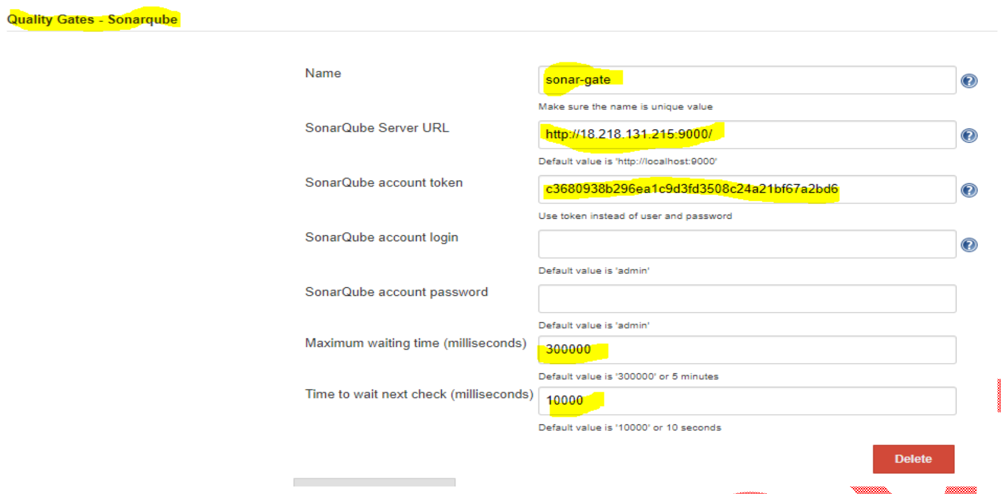
Fails the build whenever the Quality Gates criteria in the Sonar 5.6+ analysis aren't met (the project Quality Gates status is different than "Passed")

Go to configure system update below configure system.



Quality Gates

Name	quality-gate Make sure the name is unique value
SonarQube Server URL	http://18.218.131.215:9000/ Default value is 'http://localhost:9000'
SonarQube account login	admin Default value is 'admin'
SonarQube account password	..... Default value is 'admin'



Quality Gates - Sonarqube

Name: sonar-gate
Make sure the name is unique value

SonarQube Server URL: http://18.218.131.215:9000/
Default value is 'http://localhost:9000'

SonarQube account token: c3680938b296ea1c9d3fd3508c24a21bf67a2bd6
Use token instead of user and password

SonarQube account login:
Default value is 'admin'

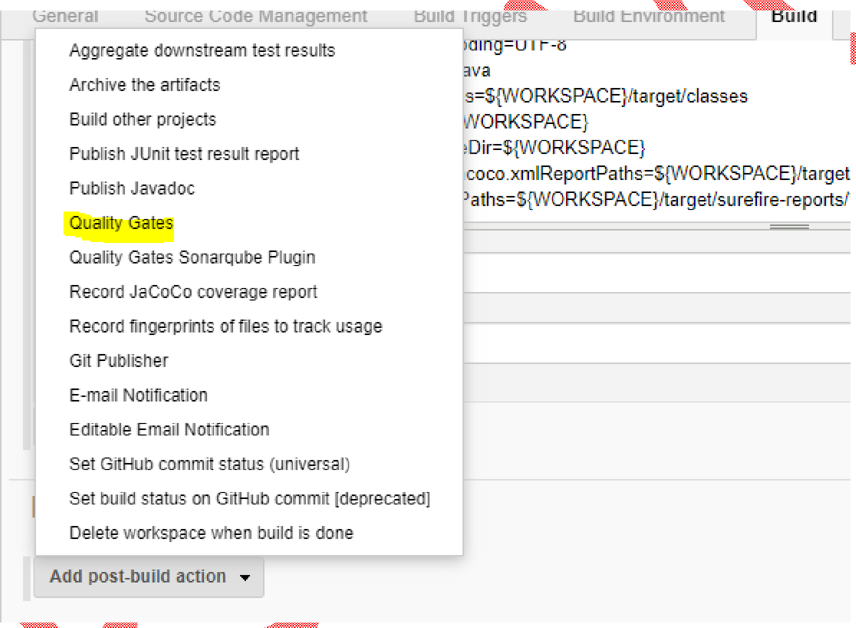
SonarQube account password:
Default value is 'admin'

Maximum waiting time (milliseconds): 300000
Default value is '300000' or 5 minutes

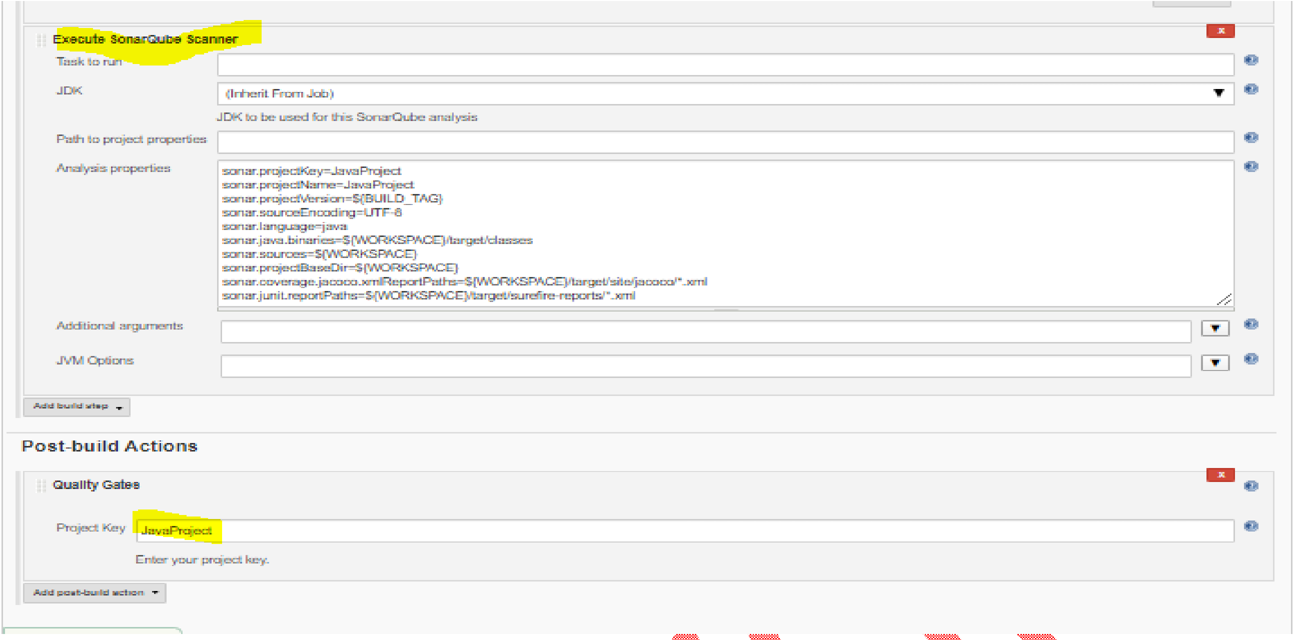
Time to wait next check (milliseconds): 10000
Default value is '10000' or 10 seconds

Go to our job configure below properties.

Go to →post build actions → Quality Gates → Enter the project key


General
Source Code Management
Build Triggers
Build Environment
Build
Aggregate downstream test results
Archive the artifacts
Build other projects
Publish JUnit test result report
Publish Javadoc
Quality Gates
Quality Gates Sonarqube Plugin
Record JaCoCo coverage report
Record fingerprints of files to track usage
Git Publisher
E-mail Notification
Editable Email Notification
Set GitHub commit status (universal)
Set build status on GitHub commit [deprecated]
Delete workspace when build is done
Add post-build action

```
oding=UTF-8
ava
s=${WORKSPACE}/target/classes
WORKSPACE}
Dir=${WORKSPACE}
coco.xmlReportPaths=${WORKSPACE}/target
Paths=${WORKSPACE}/target/surefire-reports/
```



Execute SonarQube Scanner
Task to run
JDK
(Inherit From Job)
JDK to be used for this SonarQube analysis
Path to project properties
Analysis properties

```
sonar.projectKey=JavaProject
sonar.projectName=JavaProject
sonar.projectVersion=${BUILD_TAG}
sonar.sourceEncoding=UTF-8
sonar.language=java
sonar.java.binaries=${WORKSPACE}/target/classes
sonar.sources=${WORKSPACE}
sonar.projectBaseDir=${WORKSPACE}
sonar.coverage.jacoco.xmlReportPaths=${WORKSPACE}/target/site/jacoco/*.xml
sonar.junit.reportPaths=${WORKSPACE}/target/surefire-reports/*.xml
```

Project Key: JavaProject

Save and Build the project

Interview Questions

1. What is SonarQube ?

   SonarQube is a open-source platform for continuous inspection of code quality developed by SonarSource.It's written in java language and it's supports various languages.

2. Why use SonarQube ?

   Sonar covers the 7 sections of code quality

- Architecture and Design
- Unit tests
- Duplicated code
- Potential bugs
- Complex code
- Coding standards
- Comments

3. What are the advantages of using SonarQube?

- Open source
- Initially it is developed only for Java projects.
- Today it is supporting over **20+** languages.
- I will generate the report if any issues in html format/PDF format.
- It provides the historical reports. (Compare with old and new reports)
- Integrated with Maven, Ant ,Jenkins and Gradle.
- It is a web based tool supports multiple languages (Java, C#, JS …)
- It will support multi OS platform (Windows. MAC, Linux…).

4. What are Quality Profiles in SonarQube?

- Quality Profiles are set of rules which are available in the SonarQube.
- Each language contains default profile name as ***sonar way***.
- Sonar way contains the active rules.
- All projects not explicitly assigned to some other profile will be analyzed with the default.
- By default C#,java and java script will be available.other profile will be analyzed with the default.

5. What are Quality Gates in SonarQube?

   - Quality Gates are collections of simple Boolean thresholds set on project measures. A project must pass each of the thresholds in order to pass the Quality Gate as a whole.

     (OR)

   - Quality Gate is the set of conditions the project must meet before it can be released into production.
   - It is possible to set a default Quality Gate, which will be applied to all projects not explicitly assigned to some other gate.
   - ***SonarQube way/Sonar way*** is the default Quality Gate.

6. What is role of database in SonarQube?
   Sonar uses a Derby or H2 as default database. When running Sonar, it says that these databases may only be used for evaluation. We can change this default database and use our custom DB.

7. What is the use of SonarQube Database ?
   SonarQube Database stores configuration of the SonarQube instance like security settings and they also store project quality snapshot.

8. Explain architecture of SonarQube?
   Refer the class notes

9. How to create reports in SonarQube?
   sonar.issuesreport.html.enable=true

10. What was the earlier name of SonarQube ?
    Sonar

11. At which port sonar server is available by default?
    9000

12. Does SonarQube only analysis java code ?
    No , SonarQube can analysis more than 20 languages

13. In which language SonarQube is written ?

Java

14. What are the main components of SonarQube Platform ?

- SonarQube plugin for languages
- SonarQube Scanner
- SonarQube Server
- SonarQube Database

15. What is the use of SonarQube Scanners ?

It analyzes projects source code on Continuous Integration Servers.

16. Mention basic steps for SonarQube processing?

- Developer develops code and sends its code into repository system like SCM, git
- An automatic build is fired in Continuous Integration Server (Jenkins) and execution of SonarQube Scanner happens for SonarQube analysis.
- Report is sent to SonarQube Server for processing.
- SonarQube Server processes the report and stores the analysis report results in the SonarQube Database (H2) and displays the results in the UI
- Developers review, comment, challenges their Issues to manage and reduce their Technical Debt through the SonarQube UI.

17. SonarQube Runner vs Scanner ?

"Runner" is the old name for "Scanner".

18. What is the SonarQube Plugins available which can be integrated with Eclipse, IntelliJ IDEA, Visual Studio, Visual Studio Code, and Atom?

SonarLint is an extension to your favorite IDE that provides on-the-fly feedback to developers on new bugs and quality issues injected into their code.

19. How to exclude the directory in SonarQube?

```
sonar.exclusions=src/java/test/**
sonar.exclusions=system/**, test/**, application/third_party/**, application/logs/**
```

20. Is SonarQube Replacement for Checkstyle, PMD, FindBugs ?

- Sonar will run CheckStyle, FindBugs and PMD, as well as a few other "plugins" such as Cobertura by default for Java projects. The main added value, however, is that it stores the history in a database.
- Sonar uses these 3 tools as plugins and aggregates the data from all three giving addition value by showing graphs and such from these tools. So they are complementary to sonar.

Thanks . . .

All the best . . .

...... Prepared by PSDDEVOPS