**Using SonarLint with SonarQube in IntelliJ IDE**

***What is SonarLint?*** SonarLint is a free IDE extension that lets you fix coding issues before they exist!

It helps us detect and fix quality issues as we write code. Like when we were kids teachers used to help us improve our hand writing.

**Advantages**- It gives instant feedback as we write code(bad code). If we use SonarQube in the build process, using SonarLint is a proactive step.

It will help you in following ways:

* Identify the new bugs and the quality issues in Java, typescript and javascript code. So, it’s like a good self-review of code changes.
* Code improvements and suggestions while writing code rather than running sonarQube afterwards.
* We can also run it for every class where we make any change.
* It can teach you what’s wrong with the code, show you best practices, and give you examples of fixes.

**Requirements-**SonarLint v4.7 onwards needs Java 11+

So, either we can have Java 11+ set as Java path system variable or we can install **‘Choose Runtime’** Plugin and select specific runtime for IntelliJ IDE.

**Instructions-**

→ **Download SonarLint plugin in Intellij:**

In Intellij — Go to File >> Plugins >> Type ‘SonarLint’ >> Install and Restart IDE.

→ **We can use the SonarLint either with**

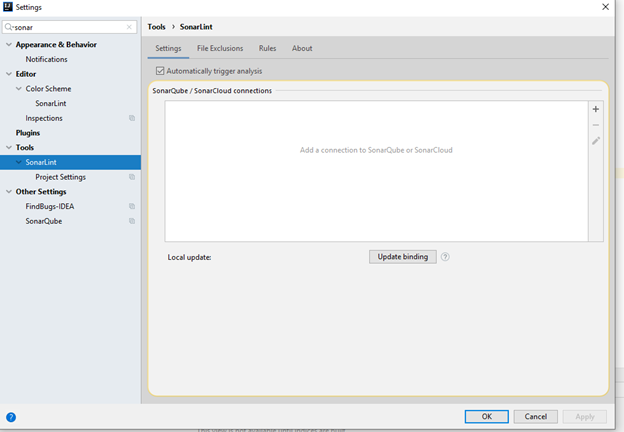
**(1)** predefined rules set and select/remove from that list, or

**(2)** we can choose the SonarQube rules.

→ **To integrate SonarQube(server) and SonarLint in our IDE** and run SonarQube code inspection rules per class to give results quickly.

1. Add the sonarQube connection binding.

File >> Setting >> Tools >> SonarLint >> Click on + sign on right side



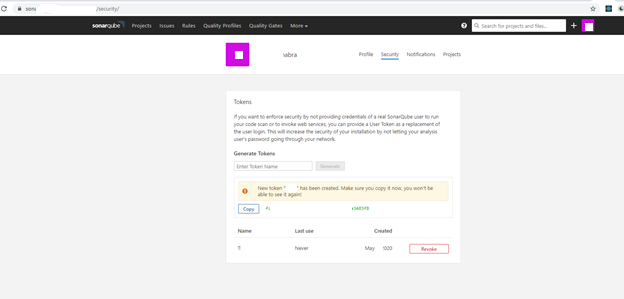
2) A new popup opens- Select SonarQube and write the sonarQubeURL.

一張含有 文字 的圖片

自動產生的描述

Enable/Disable proxy- As per your need.

3) Now, we need to create and add sonarqube authentication token. Go to your Sonarqube account and generate token and save it in notepad since it is one-time token.



4) Add this token in connection in SonarLint in Intellij IDE-

一張含有 文字 的圖片

自動產生的描述

5) Click on Finish and connection to SonarQube will be created.

Our SonarQube server is now integrated with SonarLint.

一張含有 文字 的圖片

自動產生的描述

6) Now we need to add the project settings, since our SonarQube server has many projects so we need to choose our specific project.

Go to SonarLint Project setting and select ‘Bind project to SonarQube’ option and add the project name.

一張含有 文字 的圖片

自動產生的描述

7) Final step in setup- Go to SonarLint settings and ‘Update Binding’ and see the logs in SonarLint logs.

一張含有 文字 的圖片

自動產生的描述

Sample SonarLint logs

**Note:** Whenever the SonarQube quality profile is modified- make sure to refresh all project bindings (in SonarLint) after you modified the Quality Profile. That way you ensure that SonarLint is in sync with latest state on SonarQube side.

**How to use SonarLint in Intellij IDE-**

There are several options to useSonarLint-

1) Analyze code per class file during coding.

2) Generate report- Once all the changes are done in all class files, we can analyze all the class files together and see results.

**Steps for Analyze code per class file:**

Go to class file and right click the on that file.

Then go to Analyze -> SonarLint -> Analyze File with SonarLint

一張含有 文字 的圖片

自動產生的描述

Then you can see the feedback on the bottom on the IDE with the description (Bad code you have written).

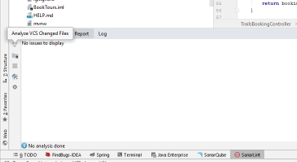
Example:

一張含有 文字 的圖片

自動產生的描述

We can now fix these code smells and try to write “Clean Code”.

**Steps for generate report for all the changes on local:**



It will start analyzing all the files changed on local-

一張含有 文字 的圖片

自動產生的描述

Once analysis is competed, it will show all the code issues (code smells):

一張含有 文字 的圖片

自動產生的描述

**Note**: SonarLint v4.7 onwards needs Java 11+ version.

So, either we can have Java 11+ set as Java path system variable or we can install **‘Choose Runtime’** Plugin and select specific runtime for IntelliJ IDE.

**References:** <https://www.sonarlint.org/>