
































-  Secrets
-  ABAP
-  Apex
-  C
-  C++
-  CloudFormation
-  COBOL
-  C#
-  CSS
-  Flex
-  Go
-  HTML
-  Java
-  JavaScript
-  Kotlin
-  Objective C
-  PHP
-  PL/I
-  PL/SQL
-  Python
-  RPG
-  **Ruby**
-  Scala
-  Swift
-  Terraform
-  Text
-  TypeScript
-  T-SQL
-  VB.NET
-  VB6
-  XML



# Ruby static code analysis

Unique rules to find Bugs, Security Hotspots, and Code Smells in your RUBY code


All rules 42

Bug 7

Security Hotspot 2

Code Smell 33

Tags ▾

Search by name... 

Class names should comply with a naming convention	Code Smell
Method names should comply with a naming convention	Code Smell
Track uses of "TODO" tags	Code Smell
Track lack of copyright and license headers	Code Smell
Octal values should not be used	Code Smell
"case" statements should not be nested	Code Smell
Control flow statements "if", "for", "while", "until", "case" and "begin...rescue" should not be nested too deeply	Code Smell
"if ... else if" constructs should end with "else" clauses	Code Smell
Expressions should not be too complex	Code Smell
Ruby parser failure	Code Smell
Functions should not have too many lines of code	Code Smell
Statements should be on separate lines	

Functions should not have too many lines of code

Analyze your code

Code Smell

Major ?

brain-overload

A function that grows too large tends to aggregate too many responsibilities. Such functions inevitably become harder to understand and therefore harder to maintain.


Above a specific threshold, it is strongly advised to refactor into smaller functions which focus on well-defined tasks.

Those smaller functions will not only be easier to understand, but also probably easier to test.


Available In:

sonarlint | sonarcloud | sonarqube


© 2008-2022 SonarSource S.A., Switzerland. All content is copyright protected. SONAR, SONARSOURCE, SONARLINT, SONARQUBE and SONARCLOUD are trademarks of SonarSource S.A. All other trademarks and copyrights are the property of their respective owners. All rights are expressly reserved. [Privacy Policy](#)

 Code Smell


"case when" clauses should not have too many lines of code

 Code Smell


Files should not have too many lines of code

 Code Smell

Lines should not be too long

 Code Smell

Tabulation characters should not be used

 Code Smell