

Scala static code analysis

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

1.	Hard-coded credentials are security-sensitive Security Hotspot
2.	Cognitive Complexity of functions should not be too high Code Smell
3.	String literals should not be duplicated Code Smell
4.	Methods should not be empty Code Smell
5.	All branches in a conditional structure should not have exactly the same implementation Bug
6.	Related "if"/"else if" statements and "case" in a "match" should not have the same condition Bug
7.	Identical expressions should not be used on both sides of a binary operator Bug
8.	All code should be reachable Bug
9.	Variables should not be self-assigned Bug
10.	Useless "if(true) {...}" and "if(false){...}" blocks should be removed Bug
11.	Methods should not have identical implementations Code Smell
12.	Two branches in a conditional structure should not have exactly the same implementation Code Smell
13.	"match" expressions should not have too many "case" clauses Code Smell
14.	Sections of code should not be commented out Code Smell
15.	Unused function parameters should be removed

	Code Smell
16.	Unused "private" methods should be removed Code Smell
17.	Track uses of "FIXME" tags Code Smell
18.	Nested blocks of code should not be left empty Code Smell
19.	Functions should not have too many parameters Code Smell
20.	Collapsible "if" statements should be merged Code Smell
21.	Using hardcoded IP addresses is security-sensitive Security Hotspot
22.	Multi-line comments should not be empty Code Smell
23.	Boolean checks should not be inverted Code Smell
24.	Unused local variables should be removed Code Smell
25.	Local variable and function parameter names should comply with a naming convention Code Smell
26.	Boolean literals should not be redundant Code Smell
27.	Class names should comply with a naming convention Code Smell
28.	Method names should comply with a naming convention Code Smell
29.	Track uses of "TODO" tags Code Smell
30.	Track lack of copyright and license headers Code Smell
31.	"match" statements should not be nested Code Smell
32.	

	Control flow statements "if", "for", "while", "match" and "try" should not be nested too deeply Code Smell
33.	"if ... else if" constructs should end with "else" clauses Code Smell
34.	Expressions should not be too complex Code Smell
35.	Scala parser failure Code Smell
36.	Methods should not have too many lines of code Code Smell
37.	Statements should be on separate lines Code Smell
38.	"match case" clauses should not have too many lines of code Code Smell
39.	Files should not have too many lines of code Code Smell
40.	Lines should not be too long Code Smell
41.	Tabulation characters should not be used Code Smell