sonar RULES Products

- 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

TypeScript static code analysis

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

All rules 279 | Vulnerability 27 | Bug 51 | Security Hotspot 43 | Code Smell 158 | Quick Fix 50

Tags | Search by name...

Code Smell

"===" and "!==" should be used instead of "==" and "!="
Code Smell

Functions should not have too many lines of code
Code Smell

Track comments matching a regular expression
Code Smell

Statements should be on separate lines
Code Smell

Magic numbers should not be used
Code Smell

Collapsible "if" statements should be merged
Code Smell

Standard outputs should not be used directly to log anything
Code Smell

Files should not have too many lines of code
Code Smell

Lines should not be too long
Code Smell

Debugger statements should not be used
Vulnerability

Regular expressions using Unicode character classes or property escapes should enable the unicode flag
Bug

The base should be provided to

Boolean expressions should not be gratuitous

Analyze your code

Code Smell | Major | cwe suspicious redundant

If a boolean expression doesn't change the evaluation of the condition, then it is entirely unnecessary, and can be removed. If it is gratuitous because it does not match the programmer's intent, then it's a bug and the expression should be fixed.

Noncompliant Code Example

```
if (a) {
  if (a) { // Noncompliant
    doSomething();
  }
}
```

Compliant Solution

```
if (a) {
  if (b) {
    doSomething();
  }
}

// or
if (a) {
  doSomething();
}
```

See

- MITRE, CWE-571 - Expression is Always True
- MITRE, CWE-570 - Expression is Always False

Available In:
sonarlint | sonarcloud | sonarqube

"parseInt"

Bug

Function declarations should not be made within blocks

Bug

Writing cookies is security-sensitive

Security Hotspot

"continue" should not be used

Code Smell

Primitive return types should be used

Code Smell