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

JavaScript static code analysis

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

| All rules (285) | 🔒 Vulnerability (29) | 🐛 Bug (62) | Security Hotspot (43) | Code Smell (151) | Quick Fix (41) |

Tags ⌄   Search by name...

Object literal shorthand syntax should be used
⚙ Code Smell

Strings and non-strings should not be added
⚙ Code Smell

Object literal syntax should be used
⚙ Code Smell

"undefined" should not be assigned
⚙ Code Smell

Trailing commas should not be used
⚙ Code Smell

Array constructors should not be used
⚙ Code Smell

Quotes for string literals should be used consistently
⚙ Code Smell

Statements should end with semicolons
⚙ Code Smell

Comments should not be located at the end of lines of code
⚙ Code Smell

Loops should not contain more than a single "break" or "continue" statement
⚙ Code Smell

Variable, property and parameter names should comply with a naming convention
⚙ Code Smell

Lines should not end with trailing whitespaces
⚙ Code Smell

Redundant pairs of parentheses should be removed

Analyze your code

⚙ Code Smell   🔴 Major ?   Quick Fix ?   🏷 confusing

The use of parentheses, even those not required to enforce a desired order of operations, can clarify the intent behind a piece of code. But redundant pairs of parentheses could be misleading, and should be removed.

Noncompliant Code Example

```
let x = (y / 2 + 1);    //Compliant even if those parenthesis

if (a && ((x+y > 0))) {  // Noncompliant
  //...
}

return ((x + 1));  // Noncompliant
```

Compliant Solution

```
let x = (y / 2 + 1);

if (a && (x+y > 0)) {
  //...
}

return (x + 1);
```

Available In:

sonarlint | sonarcloud | sonarqube

Files should contain an empty newline at the end

⊕ Code Smell

An open curly brace should be located at the end of a line

⊕ Code Smell

Tabulation characters should not be used

⊕ Code Smell

Function and method names should comply with a naming convention

⊕ Code Smell