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...

added

Code Smell

Primitive types should be omitted from initialized or defaulted declarations

Code Smell

Non-null assertions should not 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

Optional boolean parameters should have default value

Analyze your code

Code Smell Major

Having default value for optional boolean parameters makes the logic of function when missing that parameter more evident. When providing a default value is not possible, it is better to split the function into two with a clear responsibility separation.

Noncompliant Code Example

```
function countPositiveNumbers(arr: number[], countZero?: boo
  // ...
}

function toggleProperty(property: string, value?: boolean) {
  if (value !== undefined) {
    setProperty(property, value);
  } else {
    setProperty(property, calculateProperty());
  }
}
```

Compliant Solution

```
function countPositiveNumbers(arr: number[], countZero = fal
  // ...
}

function toggleProperty(property: string, value: boolean) {
  setProperty(property, value);
}

function togglePropertyToCalculatedValue(property: string) {
  setProperty(property, calculateProperty());
}
```

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