sonar RULES

Products ⌄

⊘	Secrets
SAP	ABAP
APEX	Apex
C	C
C++	C++
☁	CloudFormation
COBOL	COBOL
C#	C#
⬚	CSS
✕	Flex
‑GO	Go
⬚	HTML
☕	Java
JS	JavaScript
⬚	Kotlin
	Objective C
php	PHP
PL/I	PL/I
PL/SQL	PL/SQL
⬚	Python
RPG	RPG
⬚	Ruby
⬚	Scala
⬚	Swift
⬚	Terraform
⬚	Text
TS	**TypeScript**
⬚	T-SQL
VB	VB.NET
VB6	VB6
XML	XML

TypeScript static code analysis

TS 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

"continue" should not be used

Analyze your code

⊗ Code Smell   ⊘ Minor ⌄   🏷 bad-practice

`continue` is an unstructured control flow statement. It makes code less testable, less readable and less maintainable. Structured control flow statements such as `if` should be used instead.

Noncompliant Code Example

```
for (i = 0; i < 10; i++) {
  if (i == 5) {
    continue;  /* Noncompliant */
  }
  alert("i = " + i);
}
```

Compliant Solution

```
for (i = 0; i < 10; i++) {
  if (i != 5) {  /* Compliant */
    alert("i = " + i);
  }
}
```

Available In:

sonarlint ⊙ | **sonar**cloud ☁ | 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