




 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

Security Hotspot
Statically serving hidden files is security-sensitive
Security Hotspot
Using intrusive permissions is security-sensitive
Security Hotspot
Disabling auto-escaping in template engines is security-sensitive
Security Hotspot
Using shell interpreter when executing OS commands is security-sensitive
Security Hotspot
Setting loose POSIX file permissions is security-sensitive
Security Hotspot
Formatting SQL queries is security-sensitive
Security Hotspot
Comma operator should not be used
Code Smell
Regular expressions should not contain empty groups
Code Smell
Regular expressions should not contain multiple spaces
Code Smell
Chai assertions should have only one reason to succeed
Code Smell
Single-character alternations in regular expressions should be replaced with character classes
Code Smell

A conditionally executed single line should be denoted by indentation

Analyze your code

Code Smell

Critical

confusing suspicious

In the absence of enclosing curly braces, the line immediately after a conditional is the one that is conditionally executed. By both convention and good practice, such lines are indented. In the absence of both curly braces and indentation the intent of the original programmer is entirely unclear and perhaps not actually what is executed. Additionally, such code is highly likely to be confusing to maintainers.

Noncompliant Code Example

```
if (condition) // Noncompliant
doTheThing();

doTheOtherThing();
somethingElseEntirely();

foo();
```

Compliant Solution




```
if (condition)
doTheThing();

doTheOtherThing();
somethingElseEntirely();

foo();
```

Available In:

sonarlint | sonarcloud | sonarqube

<p>Reluctant quantifiers in regular expressions should be followed by an expression that can't match the empty string</p> <p> Code Smell</p>
<p>Tests should check which exception is thrown</p> <p> Code Smell</p>
<p>Character classes in regular expressions should not contain the same character twice</p> <p> Code Smell</p>
<p>Names of regular expressions named groups should be used</p>