




 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


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

Track lack of copyright and license headers

Analyze your code

 Code Smell

 Blocker






Each source file should start with a header stating file ownership and the license which must be used to distribute the application.

This rule must be fed with the header text that is expected at the beginning of every file.

Compliant Solution

```
/*
 * SonarQube, open source software quality management tool.
 * Copyright (C) 2008-2013 SonarSource
 * mailto:contact AT sonarsource DOT com
 *
 * SonarQube is free software; you can redistribute it and/o
 * modify it under the terms of the GNU Lesser General Publi
 * License as published by the Free Software Foundation; eit
 * version 3 of the License, or (at your option) any later v
 *
 * SonarQube is distributed in the hope that it will be usef
 * but WITHOUT ANY WARRANTY; without even the implied warran
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See
 * Lesser General Public License for more details.
 *
 * You should have received a copy of the GNU Lesser General
 * along with this program; if not, write to the Free Softwa
 * Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-
 */
```

Available In:

 |  | 

© 2008-2022 SonarSource S.A., Switzerland. All content is copyright protected. SONAR, SONARSOURCE, SONARLINT, SONARQUBE and SONARCLOUD are trademarks of SonarSource S.A. All other trademarks and copyrights are the property of their respective owners. All rights are expressly reserved.

[Privacy Policy](#)

https://rules.sonarsource.com/typescript/RSPEC-1451

1/2

<div>Files should contain an empty newline at the end</div> <div> Code Smell</div>
<div>An open curly brace should be located at the end of a line</div> <div> Code Smell</div>
<div>Tabulation characters should not be used</div> <div> Code Smell</div>
<div>Function and method names should comply with a naming convention</div> <div> Code Smell</div>