
































-  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



PHP static code analysis

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

All rules 268

 Vulnerability 40












 Bug 51

 Security Hotspot 33

 Code Smell 144

Tags ▾

Search by name... 🔍

be used to end lines	
More than one property should not be declared per statement	
The "var" keyword should not be used	
"<?php" and "<?=" tags should be used	
File names should comply with a naming convention	
Comments should not be located at the end of lines of code	
Local variable and function parameter names should comply with a naming convention	
Field names should comply with a naming convention	
Lines should not end with trailing whitespaces	
Files should contain an empty newline at the end	
Modifiers should be declared in the correct order	
An open curly brace should be located at the beginning of a line	

Boolean literals should not be redundant

Analyze your code

 Code Smell

 Minor ?

 clumsy

Redundant Boolean literals should be removed from expressions to improve readability.

Noncompliant Code Example

```
if ($booleanVariable == true) { /* ... */ }
if ($booleanVariable != true) { /* ... */ }
if ($booleanVariable || false) { /* ... */ }
doSomething(!false);
```

Compliant Solution

```
if ($booleanVariable) { /* ... */ }
if (!$booleanVariable) { /* ... */ }
if ($booleanVariable) { /* ... */ }
doSomething(true);
```

Exceptions

The use of literal booleans in comparisons which use identity operators (=== and !==) are ignored.

Available In:

 sonarlint

 sonarcloud

 sonarqube

 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

Method and function names should comply with a naming convention

 Code Smell

Creating cookies with broadly defined "domain" flags is security-sensitive

 Security Hotspot