

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

Files that define symbols should not cause side-effects

Analyze your code effects

Bug

Major ?

psr1 user-experience

Files that define symbols such as classes and variables may be included into many files. Simply performing that inclusion should have no effect on those files other than declaring new symbols. For instance, a file containing a class definition should not also contain side-effects such as `print` statements that will be evaluated automatically on inclusion. Logic should be segregated into symbol-only files and side-effect-only files. The type of operation which is not allowed in a symbol-definition file includes but is not limited to:

- generating output
- modifying `ini` settings
- emitting errors or exceptions
- modifying global or static variables
- reading/writing files

Noncompliant Code Example

```
<?php

print "Include worked!";

class foo {
    // ...
}
```

Compliant Solution

```
<?php

class foo {

    public function log() {
        print "Include worked!";
    }

}
```

See

- PHP-FIG Basic Coding Standard PSR1, 2.3 - Side Effects

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

© 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](#)