

- 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
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

Functions and variables should not be defined outside of classes

Analyze your code

Code Smell

Blocker

design

Defining and using global variables and global functions, when the convention dictates OOP can be confusing and difficult to use properly for multiple reasons:

- You run the risk of name clashes.
- Global functions must be stateless, or they can cause difficult-to-track bugs.
- Global variables can be updated from anywhere and may no longer hold the value you expect.
- It is difficult to properly test classes that use global functions.

Instead of being declared globally, such variables and functions should be moved into a class, potentially marked `static`, so they can be used without a class instance.

This rule checks that only object-oriented programming is used and that no functions or procedures are declared outside of a class.

Noncompliant Code Example

```
<?php

$name = "Bob"; // Noncompliant

function doSomething($arg) { // Noncompliant
    //...
}

class MyClass {
    //...
}
```

Compliant Solution

```
<?php
class MyClass {

    public static $name = "Bob"; // Compliant

    public static function doSomething($arg) {
        //...
    }
    //...
}
```

Available In:

 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