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

Test methods should be discoverable

Analyze your code

☢ Code Smell   ⬆ Major ⃝   🏷 tests unused phpunit

The PHPUnit test runner does execute public methods defined within test classes that have a name starting with *"test"* or the *@test* annotation. Methods that do not convey to this will not get executed.

This rule raises an issue on methods marked as test methods (by name or annotation) but do not have a public visibility. An issue is also raised on public methods that are not marked as tests, do contain assertions, and are not called from within another discoverable test method within the class. No issues are raised in abstract classes.

Noncompliant Code Example

```
class MyTest extends \PHPUnit\Framework\TestCase {
    private function testA() { // Noncompliant
        $this->assertTrue(getValue());
    }

    public function b() { // Noncompliant
        $this->assertTrue(getValue());
    }
}
```

Compliant Solution

```
class MyTest extends \PHPUnit\Framework\TestCase {
    public function testA() { // Compliant
        $this->assertTrue(getValue());
    }

    public function testB() {
        $this->b();
    }

    public function b() { // Compliant
        $this->assertTrue(getValue());
    }
}
```

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