

- 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... 🔍

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

Framework-provided functions should be used to test exceptions

Analyze your code

Code Smell

Minor ?

tests phpunit

PHPUnit provides helper functions and annotations to verify that a given block of code throws an exception and to assert different properties of that exception. The provided helper functions are:

- expectException()
- expectExceptionCode()
- expectExceptionMessage()
- expectExceptionMessageRegExp()

This check raises an issue when the throw of an exception is verified using a *try-catch* approach instead of relying on the provided helper functions.

Noncompliant Code Example

```
public function testA()  
{  
    try {  
        doSomething();  
        $this->fail("Assertion should have been thrown")  
    } catch (MyException $e) {  
        assertEquals("Exception message", $e->getMessage())  
    }  
}
```

Compliant Solution

```
public function testB()  
{  
    $this->expectException(MyException::class);  
    $this->expectExceptionMessage("Exception message");  
  
    doSomething();  
}
```

See

thePHP.cc: [Questioning PHPUnit Best Practice](#)

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