
































-  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

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

## "catch" clauses should do more than rethrow

Analyze your code

Code Smell Minor error-handling unused finding clumsy

A catch clause that only rethrows the caught exception has the same effect as omitting the catch altogether and letting it bubble up automatically, but with more code and the additional detriment of leaving maintainers scratching their heads.

Such clauses should either be eliminated or populated with the appropriate logic.

### Noncompliant Code Example

```
$s = "";
try {
    $s = readMyFile($fileName);
} catch (Exception $e) {
    throw $e; // Noncompliant
}
```

### Compliant Solution

```
$s = "";
try {
    $s = readMyFile($fileName);
} catch (Exception $e) {
    error_log($e->getMessage());
    throw $e; // Compliant
}
```

or

```
$s = "";
try {
    $s = readMyFile($fileName);
} catch (Exception $e) {
    error_log($e->getMessage());
    throw new MyException("an exception occurred", 2, $e)
}
```

or

```
$s = readMyFile($fileName);
```

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