

- Secrets
- ABAP
- Apex
- C
- C++
- CloudFormation
- COBOL
- Code Smell
- 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...

make PHP 4-style calls to parent constructors	Code Smell
PHP 4 constructor declarations should not be used	Code Smell
Deprecated predefined variables should not be used	Code Smell
"switch" statements should not have too many "case" clauses	Code Smell
Classes should not have too many methods	Code Smell
Functions should not have too many lines of code	Code Smell
"for" loop stop conditions should be invariant	Code Smell
Sections of code should not be commented out	Code Smell
Unused function parameters should be removed	Code Smell
Unused "private" methods should be removed	Code Smell
Functions should not contain too many return statements	Code Smell

## Exceptions should not be created without being thrown

Analyze your code

Bug Major error-handling

Creating a new Exception without actually throwing it is useless and is probably due to a mistake.

### Noncompliant Code Example

```
if ($x < 0)
    new Exception('$x must be nonnegative');
```

### Compliant Solution


```
if ($x < 0)
    throw new Exception('$x must be nonnegative');
```

Available In: sonarlint sonarcloud sonarqube

Track uses of "FIXME" tags

 Code Smell

Generic exceptions `ErrorException`, `RuntimeException` and `Exception` should not be thrown

 Code Smell

Local variables should not have the same name as class fields

 Code Smell

Redundant pairs of parentheses should be removed

 Code Smell

Inheritance tree of classes should not be too deep