

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



Java static code analysis

Unique rules to find Bugs, Vulnerabilities, Security Hotspots, and Code Smells in your JAVA code

- All rules 632
- Vulnerability 53
- Bug 154
- Security Hotspot 36
- Code Smell 389
- Quick Fix 42

Tags ▾

Search by name... 🔍

Overriding methods should do more than simply call the same method in the super class
Code Smell
Classes that override "clone" should be "Cloneable" and call "super.clone()"
Code Smell
Public constants and fields initialized at declaration should be "static final" rather than merely "final"
Code Smell
Local variable and method parameter names should comply with a naming convention
Code Smell
Exception classes should be immutable
Code Smell
Field names should comply with a naming convention
Code Smell
Primitive wrappers should not be instantiated only for "toString" or "compareTo" calls
Code Smell
Case insensitive string comparisons should be made without intermediate upper or lower casing
Code Smell
Collection.isEmpty() should be used to test for emptiness
Code Smell
String.valueOf() should not be appended to a String
Code Smell
Interface names should comply with a

Constructors of an "abstract" class should not be declared "public"

Analyze your code

Code Smell Major ? design

Abstract classes should not have public constructors. Constructors of abstract classes can only be called in constructors of their subclasses. So there is no point in making them public. The `protected` modifier should be enough.

Noncompliant Code Example





```
public abstract class AbstractClass1 {
    public AbstractClass1 () { // Noncompliant, has public m
        // do something here
    }
}
```

Compliant Solution

```
public abstract class AbstractClass2 {
    protected AbstractClass2 () {
        // do something here
    }
}
```

Available In:

sonarlint | sonarcloud | sonarqube

naming convention  Code Smell
"throws" declarations should not be superfluous  Code Smell
Unnecessary imports should be removed  Code Smell
Return of boolean expressions should not be wrapped into an "if-then-else" statement  Code Smell