Secrets
ABAP
Apex
C
C++
CloudFormation
COBOL
C#
CSS
Flex
Go
HTML
Java
JavaScript
Kotlin
Kubernetes
Objective C
PHP
PL/I
PL/SQL
Python
RPG
Ruby
Scala
Swift
Terraform
Text
TypeScript
T-SQL
VB.NET
VB6
XML

T-SQL static code analysis

Unique rules to find Bugs, Vulnerabilities, Security Hotspots, and Code Smells in your T-SQL code

| All rules 80 | 🔒 Vulnerability ① | 🐛 Bug 16 | 🛡 Security Hotspot ④ | ◉ Code Smell 59 |

Tags ⌄   🔍 Search by name...

Unary prefix operators should not be repeated
🐛 Bug

"NULL" should not be compared directly
🐛 Bug

Related "IF"/"ELSE IF" statements and "WHEN" clauses in a "CASE" should not have the same condition
🐛 Bug

Identical expressions should not be used on both sides of a binary operator
🐛 Bug

All code should be reachable
🐛 Bug

Loops with at most one iteration should be refactored
🐛 Bug

Variables should not be self-assigned
🐛 Bug

"GOTO" statements should not be used
◉ Code Smell

Deprecated system tables and views should not be used
◉ Code Smell

"ANSI_NULLS", "ANSI_PADDING" and "CONCAT_NULL_YIELDS_NULL" should not be configured
◉ Code Smell

"@@IDENTITY" should not be used
◉ Code Smell

"COALESCE", "IIF" and "CASE" input expressions

Unary prefix operators should not be repeated

Analyze your code

🐛 Bug   🔺 Major ⍰

Calling the +, –, ~ or NOT prefix operator twice does nothing: the second invocation undoes the first. Such mistakes are typically caused by accidentally double-tapping the key in question without noticing.

Either this is a bug, if the operator was actually meant to be called once, or misleading if done on purpose.

Noncompliant Code Example

```
DECLARE @v1 INTEGER = 1
DECLARE @v2 INTEGER = - - -@v1 -- Noncompliant; -@v1 should be used instead
DECLARE @v3 INTEGER = ~~~@v1 -- Noncompliant; ~@v1 should be used instead
DECLARE @v4 INTEGER = ++@v1 -- Noncompliant; operators are useless here

IF NOT NOT @v1 <> @v2 -- Noncompliant
    BEGIN
        PRINT @msg
    END
```

Compliant Solution

```
DECLARE @v1 INTEGER = 1
DECLARE @v2 INTEGER = -@v1
DECLARE @v3 INTEGER = ~@v1
DECLARE @v4 INTEGER = @v1

IF @v1 <> @v2
    BEGIN
        PRINT @msg
    END
```

Available In:

sonarlint | sonarcloud | sonarqube Developer Edition