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...

Column names should be used in an "ORDER BY" clause
Code Smell

Queries should not join too many tables
Code Smell

Function and procedure names should comply with a naming convention
Code Smell

Columns to be read with a "SELECT" statement should be clearly defined
Code Smell

"CASE" expressions should not have too many "WHEN" clauses
Code Smell

Sections of code should not be commented out
Code Smell

Unused procedure and function parameters should be removed
Code Smell

Track uses of "FIXME" tags
Code Smell

Redundant pairs of parentheses should be removed
Code Smell

Functions and procedures should not have too many parameters
Code Smell

Collapsible "if" statements should be merged
Code Smell

Column names should be used in an "ORDER BY" clause

Analyze your code

Code Smell | Major | sql

Even though the `ORDER BY` clause supports using column numbers, doing so makes the code difficult to read and maintain. Therefore the use of column names is preferred.

Noncompliant Code Example

```
SELECT FIRST_NAME, LAST_NAME, REGION
FROM PERSONS
ORDER BY 2, 1
```

Compliant Solution

```
SELECT FIRST_NAME, LAST_NAME, REGION
FROM PERSONS
ORDER BY LAST_NAME, FIRST_NAME
```

Available In:

sonarlint | sonarcloud | sonarqube Developer Edition

© 2008-2022 SonarSource S.A., Switzerland. All content is copyright protected. SONAR, SONARSOURCE, SONARLINT, SONARQUBE and SONARCLOUD are trademarks of SonarSource S.A. All other trademarks and copyrights are the property of their respective owners. All rights are expressly reserved.
Privacy Policy