















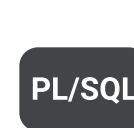


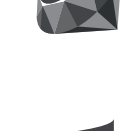




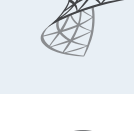









-  Secrets
-  ABAP
-  Apex
-  C
-  C++
-  CloudFormation
-  COBOL
-  C#
-  CSS
-  Flex
-  Go
-  HTML
-  Java
-  JavaScript
-  Kotlin
-  Kubernetes
-  Objective C
-  PHP
-  PL/I
-  PL/SQL
-  Python
-  RPG
-  Ruby
-  Scala
-  Swift
-  Terraform
-  Text
-  TypeScript
-  **T-SQL**
-  VB.NET
-  VB6
-  XML














# 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... 

"CASE" expressions should end with "ELSE" clauses
 Code Smell
"IF ... ELSEIF" constructs should end with "ELSE" clauses
 Code Smell
Control structures should use BEGIN...END blocks
 Code Smell
String literals should not be duplicated
 Code Smell
Expressions should not be too complex
 Code Smell
"WHERE" clauses should not contain redundant conditions
 Bug
Track lack of SQL Server session configuration
 Code Smell
Duplicate values should not be passed as arguments
 Code Smell
Track parsing failures
 Code Smell
Functions and stored procedure should not have too many lines of code
 Code Smell
Track uses of "NOSONAR" comments
 Code Smell

## "CASE" expressions should end with "ELSE" clauses

[Analyze your code](#)

 Code Smell  Critical   cwe

The requirement for a final `ELSE` clause is defensive programming. The clause should either take appropriate action, or contain a suitable comment as to why no action is taken.

### Noncompliant Code Example

```
SELECT
    CASE category
        WHEN 'A' THEN 21
        WHEN 'B' THEN 33
    END shipping_cost
FROM product
```

### Compliant Solution

```
SELECT
    CASE category
        WHEN 'A' THEN 21
        WHEN 'B' THEN 33
        ELSE 42
    END shipping_cost
FROM product
```

### See

- [MITRE, CWE-478](#) - Missing Default Case in Switch Statement
- [CERT, MSC01-C](#) - Strive for logical completeness

Available In:

**sonarlint**  | **sonarcloud**  | **sonarqube**  Developer Edition