

PL/SQL static code analysis: Individual "WHERE" clause conditions should not be unconditionally true or false

1 minute

WHERE clause conditions that reinforce or contradict the definitions of their columns are useless; they are always either unconditionally true or unconditionally false. For instance, there's no point in including `AND column IS NOT NULL` if the column is defined as non-null.

Noteworthy

This rule raises issues only when a **Data Dictionary** is provided during the analysis. See <https://docs.sonarqube.org/display/PLUG/Data+Dictionary>

Noncompliant Code Example

```
CREATE TABLE product
(id INT,
 name VARCHAR(6) NOT NULL,
 mfg_name VARCHAR(6),
 mfg_id INT
 ...
```

```
SELECT name, price
```

FROM product

WHERE name is not null -- Noncompliant; always true. This column is NOT NULL

AND mfg_name = 'Too long name' -- Noncompliant; always false. This column can contain only 6 characters