



Datenbankmanagementsysteme / SS 2015

Dreiwertige Logik in SQL: TRUE, FALSE, UNKOWN

SQL realisiert eine dreiwertige Logik, d.h. außer den Wahrheitswerten TRUE und FALSE gibt es noch einen dritten Wahrheitswert UNKOWN. Dies hat auf ALLE Bedingungsauadrücke signifikante Auswirkungen!

Zu beachten ist dabei insbesondere, dass Integritätsbedingungen (Constraints) nur dann verletzt werden, wenn sie FALSE sind. D.h. z.B. dass der Constraint UNIQUE durchaus mehrere NULL-Attributwerte zulässt, ebenso wie CHECK(VALUE IN ('SE', 'WiInf', 'Magister', 'Lehramt')) nicht ausschließt, dass der Attributwert NULL sein darf. Selektionsbedingungen (WHERE-/HAVING/ON-Klauseln) dagegen sind nur dann erfüllt, wenn sie TRUE sind.

Wahrheitstabellen der dreiwertigen Logik in SQL:

| AND | T | U | F |
|-----|---|---|---|
| T | T | U | F |
| U | U | U | F |
| F | F | F | F |

| OR | T | U | F |
|----|---|---|---|
| T | T | T | T |
| U | T | U | U |
| F | T | U | F |

| NOT | |
|-----|---|
| T | F |
| U | U |
| F | T |

Beispiele:

| | | |
|--------------------|----------------------|------------|
| 3 > 4 AND 4 > NULL | <=> FALSE AND UNKOWN | <=> FALSE |
| 3 > 4 OR 4 > NULL | <=> FALSE OR UNKOWN | <=> UNKOWN |
| 3 < 4 OR 4 < NULL | <=> TRUE OR UNKOWN | <=> TRUE |
| NOT (3 = NULL) | <=> NOT (UNKOWN) | <=> UNKOWN |

Test auf NULL-Wert:

NULL-Werte sollte grundsätzlich mit IS NULL oder IS NOT NULL überprüft werden, denn das Ergebnis des folgenden Vergleichs ist UNKOWN.

NULL = NULL <=> UNKOWN

Test auf TRUE, FALSE und UNKOWN:

Für einen Bedingungsauadruck p (z.B. für p := A < B) gilt:

| p = | TRUE | FALSE | UNKOWN |
|-----------------|------|-------|--------|
| p IS TRUE | T | F | F |
| p IS NOT TRUE | F | T | T |
| p IS FALSE | F | T | F |
| p IS NOT FALSE | T | F | T |
| p IS UNKOWN | F | F | T |
| p IS NOT UNKOWN | T | T | F |