**Joins**

Bereinigung Produkt-Tabelle.

Doppelte Datensätze selber entfernt und jetzt Vorbeuge-Maßnahme: **UNIQUE**

ALTER TABLE Produkt ADD CONSTRAINT

UK\_Produkt\_Name UNIQUE(Name)

**JOIN (verbinden) allgemein**

Zuerst normalisieren (= Aufteilen auf mehrere Tabellen) mit der Begründung der Vermeidung von Redundanz

Dann wieder zusammenfügen, um logische (zusammenhängende) Sicht auf die Daten zu bekommen.

**INNER JOIN**

Mit INNER JOIN werden ZUSAMMENGEHÖRNEDE daten zweier Tabellen zusammengefügt (ausgegeben).

SELECT \* FROM

Kunde k

INNER JOIN Bestellung b ON k.KundeId = b.KundeId

**LEFT JOIN**

Gibt ZUSAMMENGEHÖRNEDE daten zweier Tabellen PLUS Restmenge der linken Tabelle (erklären) aus

SELECT \* FROM

Bestellung b

LEFT JOIN BestellPosition bp ON b.BestellungId = bp.BestellungId

**RIGHT JOIN**

Gibt ZUSAMMENGEHÖRNEDE daten zweier Tabellen PLUS Restmenge der linken Tabelle (erklären) aus

SELECT \* FROM

BestellPosition bp

RIGHT JOIN Bestellung b ON b.BestellungId = bp.BestellungId

**FULL JOIN**

Damit wird die Gesamtmenge von beiden Tabellen ausgegeben.

SELECT \* FROM

Bestellung b

FULL JOIN BestellPosition bp ON b.BestellungId = bp.BestellungId

**CROSS JOIN**

Ein Cross-Join gibt alle möglichen Zeilenkombinationen aus zwei Tabellen zurück (wird auch als kartesisches Produkt bezeichnet). Auf alle Spalten aus einer Tabelle folgen alle Spalten aus der anderen Tabelle. Normalerweise ergibt sich daraus kein sinnvolles Ergebnisset.  
es kombiniert jede Zeile der ersten Tabelle mit jeder Zeile der zweiten Tabelle. Das führt dazu, dass die Anzahl der Zeilen im Ergebnis gleich dem Produkt der Zeilenanzahl der beiden Tabellen ist.

Ein CROSS JOIN wird in der Regel verwendet, wenn man alle möglichen Kombinationen der Zeilen beider Tabellen benötigt. Dies kann z. B. in bestimmten statistischen oder analytischen Szenarien nützlich sein, ist aber oft nicht das, was man möchte, da das Ergebnis sehr schnell sehr groß werden kann.

SELECT \* FROM

Bestellung b

CROSS JOIN BestellPosition bp

**Verschiedene JOIN-Arten kombinieren**

SELECT \* FROM

Kunde k

INNER JOIN Bestellung b ON k.KundeId = b.KundeId

***INNER Join BestellPosition bp ON bp.BestellungId = b.BestellungId***

**Z.B in DELETE verwenden**

Löschen von unnötigen Bestellungen ohne Löschposition

**Test / Evaluierung**

SELECT \*

FROM BestellPosition bp

RIGHT JOIN Bestellung b ON b.BestellungId = bp.BestellungId

WHERE bp.BestellungId IS NULL

**GO:**

DELETE Bestellung

FROM BestellPosition bp

LEFT JOIN Bestellung b ON b.BestellungId = bp.BestellungId

WHERE bp.BestellungId IS NULL