

## ***Wiederholung: SELECT Statement***

Syntax (vereinfacht):

```
SELECT * | [DISTINCT] spalte_1, ..., spalte_n
FROM tabelle_1, ..., tabelle_m
[WHERE bedingung];
```

bedingung besteht aus Spaltennamen, Ausdrücken, Konstanten und Vergleichsoperatoren

## ***Wiederholung: Änderungsoperationen in SQL***

DML Syntax (vereinfacht):

**insert** Fügt einer Tabelle neue Zeilen hinzu:

```
INSERT INTO tabelle [(spalte_1 , ... , spalte_n)]
VALUES                (wert_1 , ... , wert_n)
```

**update** Ändert bestehende Zeilen:

```
UPDATE tabelle
SET     spalte_1=wert_1 [, spalte_2=wert_2 ...]
[WHERE bedingung]
```

**delete** Löscht Zeilen:

```
DELETE FROM tabelle
[WHERE bedingung]
```

bedingung besteht aus Spaltennamen, Ausdrücken, Konstanten und Vergleichsoperatoren

## Wiederholung: Join Operationen (Verbund)

- Joins werden in der FROM-Klausel spezifiziert
- Kreuzverbund (Kreuzprodukt)  $R \times S$   
... FROM R CROSS JOIN S (äquivalent zu ... FROM R, S)
- Innerer Verbund
  - R [INNER] JOIN S ON bedingung  
Zeilenpaar aus  $R$  und  $S$  passt zusammen, falls *bedingung* logisch wahr ist
  - R [INNER] JOIN S USING ( $a_1, \dots, a_n$ )  
abkürzende Schreibweise für gleiche Attribute: USING ( $a, b$ ) entspricht ON ( $R.a = S.a$  AND  $R.b = S.b$ ), zusätzlich werden doppelte Spalten unterdrückt
  - R NATURAL [INNER] JOIN S  
abkürzende Schreibweise: steht für USING mit genau den Spaltennamen, die in beiden Tabellen vorhanden sind
- Äußerer Verbund
  - R LEFT | RIGHT | FULL [OUTER] JOIN S ON bedingung
  - R LEFT | RIGHT | FULL [OUTER] JOIN S USING ( $a_1, \dots, a_n$ )
  - R NATURAL LEFT | RIGHT | FULL [OUTER] JOIN S
  - Linker äußerer Verbund  $R \bowtie S$ : Ergebnisrelation enthält in jedem Fall ein Tupel für jede Zeile in der linken Relation  $R$  (existiert kein passendes in  $S$ , so wird das Join-Resultat mit NULL-Werten aufgefüllt).
  - Rechter äußerer Verbund  $R \bowtie S$ : Ergebnisrelation enthält in jedem Fall ein Tupel für jede Zeile in der rechten Relation  $S$
  - Voller äußerer Verbund: Ergebnisrelation enthält in jedem Fall ein Tupel für jede Zeile in der linken und für jede Zeile in der rechten Relation