

Kap. 1: Vertiefung von SQL

- Filterung doppelter Zeilen per **DISTINCT** (SELECT DISTINCT year FROM movie)
- Vereinigung von Select-Ergebnissen mit **UNION** (...) UNION (...)
 - o Filter doppelte Ergebnisse (Unterdrückung der Filterung mit **UNION ALL**)
- **Gruppierung**: Datensätze mit gemeinsamen Eigenschaften zusammenfassen.

```
SELECT year, avg(rating), count(*) FROM MR
WHERE votes > 200000
GROUP BY year
HAVING ...
```

Sortierung und Unterabfragen

```
SELECT ROW_NUMBER() OVER
      (ORDER BY Rating DESC) AS rn, title
FROM mr
WHERE type = 'C'
ORDER BY 1
```

Lösung: Join auf Unterabfrage

```
(SELECT year, max(rating) rating
FROM mr
WHERE type = 'C' GROUP BY year)
```

```
SELECT * FROM mr JOIN X ON
(mr.rating = X.rating and mr.year = X.year)
```

Analytische Funktionen

Single Row Funktionen

Funktion OVER (PARTITION BY Year ORDER BY Rating DESC)

Funktion ist Rank(), Dense_Rank(), Row_Number()

Vollständige Aggregationen

Funktion OVER (PARTITION BY Year)

Funktion ist Sum(X), Count(*), Avg(X)

Kumulierende Aggregationen

Funktion OVER (PARTITION BY Year ORDER BY Rating DESC)

Funktion ist Sum(X), Count(*), Avg(X)

- ROW_NUMBER: Fortlaufende Nummerierung innerhalb einer Partition Beispiel: 1, 2, 3, 4
- RANK: Beispiel: 1, 2, 2, 4
- DENSE_RANK: Wie rank, nur dass keine Lücke entsteht Beispiel: 1, 2, 2, 3
- SUM ohne Sortierung: Summierung über die ganze Partition
- SUM mit Sortierung: Summierung bis zur aktuellen Zeile inkl. aller Zeilen mit gleichem Sortierschlüssel
- AVG, COUNT: Analog zu SUM; Unterscheidung zwischen "mit Sortierung" und "ohne Sortierung"

```
SELECT id, wert1, wert2,
      ROW_NUMBER() OVER (ORDER BY WERT1) as a,
      RANK() OVER (ORDER BY WERT1) as b,
      DENSE_RANK() OVER (ORDER BY WERT1) as c,
      ROW_NUMBER() OVER (PARTITION BY WERT1 ORDER BY WERT2) as d,
      RANK() OVER (PARTITION BY WERT1 ORDER BY WERT2) as e,
      DENSE_RANK() OVER (PARTITION BY WERT1 ORDER BY WERT2) as f,
      SUM(WERT2) OVER (PARTITION BY WERT1) as g,
      SUM(WERT2) OVER (PARTITION BY WERT1 ORDER BY ID) as h
FROM a2;
```