

4.5 Anfragen mit Mengenoperatoren

UNION, INTERSECT und EXCEPT.

- ▶ Die beteiligten Tabellen müssen zueinander *kompatible* Spaltentypen haben.
- ▶ Die Resultatspalte bekommt dann jeweils den allgemeineren Typ.

Welche Länder sind Teil von Europa und Asien?

```
SELECT LCode FROM Lage
  WHERE Kontinent = 'Europe'
INTERSECT
SELECT LCode FROM Lage
  WHERE Kontinent = 'Asia'
```

Welche Landcodes treten in der Relation Land oder der Relation Lage auf?

```
SELECT LCode, 'Stadt' AS Kategorie FROM Stadt  
UNION  
SELECT LCode, 'Lage' AS Kategorie FROM Lage
```

Welche Landcodes treten in der Relation Land und nicht in der Relation Lage auf?

```
SELECT LCode FROM Land  
EXCEPT  
SELECT LCode FROM Lage
```

Hinweis: Oracle kennt nicht EXCEPT, aber dafür MINUS!

Duplikate

- ▶ Duplikate werden berücksichtigt, sofern die Varianten UNION ALL, INTERSECT ALL, EXCEPT ALL verwendet werden. Anderenfalls wird standardmäßig DISTINCT angenommen.
- ▶ Im Falle einer Verwendung von ALL verhalten sich die Operatoren wie folgt. Hat der erste Operand n Duplikate einer Zeile und der zweite Operand m , wobei $0 \leq n, m$, dann hat das Ergebnis bei UNION $n + m$, bei INTERSECT $\min(n, m)$, und bei EXCEPT $\max(n - m, 0)$ Duplikate dieses Tupels.

4.6 Geschachtelte Anfragen

Eine Anfrage heißt *geschachtelt*, wenn sie in der SELECT-, FROM-, oder WHERE-, bzw. HAVING-Klausel selbst wieder eine SQL-Anfrage enthält.

Zum Testen des Ergebnisses einer Teilanfrage (engl. subquery) existieren die Operatoren: IN, ANY, ALL, UNIQUE, EXISTS und NOT.

Welche Länder befinden sich gemeinsam mit Russland auf einem Kontinent?

```
SELECT DISTINCT LCode FROM Lage
  WHERE Kontinent IN
    (SELECT Kontinent FROM Lage WHERE LCode = 'RU')
```

Welche Länder haben eine größere Fläche als mindestens ein anderes Land?

```
SELECT LName FROM Land
WHERE Fläche > ANY
  (SELECT Fläche FROM Land)
```

Welche Länder haben eine größere Fläche als mindestens ein anderes europäisches Land?

```
SELECT LName FROM Land
WHERE Fläche > ANY
  (SELECT Fläche FROM Land, Lage
   WHERE Land.LCode = Lage.LCode AND
         Lage.Kontinent = 'Europe')
```

Welche Länder haben eine größere Fläche als alle anderen Länder?

```
SELECT LName FROM Land L1
WHERE Fläche > ALL
  (SELECT Fläche FROM Land L2
   WHERE L1.LCode <> L2.LCode)
```

Bei Verwendung von *Korrelationsvariablen*, (*- namen*) wird die Teilanfrage pro möglicher Wertekombination der Korrelationsvariablen ihrer übergeordneten Anfragen genau einmal ausgeführt.

Bei Verwendung von *Korrelationsvariablen* wird die Teilanfrage pro möglicher Wertekombination der Korrelationsvariablen ihrer übergeordneten Anfragen genau einmal ausgeführt.

Zu welchen Ländern ist genau ein Kontinent bekannt?

```
SELECT LName FROM Land L1
WHERE UNIQUE
      (SELECT L2.Kontinent FROM Lage L2
       WHERE L1.LCode = L2.LCode)
```

Zu welchen Ländern ist genau ein Kontinent bekannt?

```
SELECT LName FROM Land L1
WHERE 1 =
      (SELECT COUNT(*) FROM Lage L2
       WHERE L1.LCode = L2.LCode)
```

Division!

Kann für die Division der relationalen Algebra $R \div S$, wobei R über $X = \{X_1, \dots, X_n\}$, S über $Y = \{Y_1, \dots, Y_m\}$ und $Y \subseteq X, Z = X \setminus Y$, $Z = \{Z_1, \dots, Z_k\}$ eine systematische Transformation nach SQL angegeben werden?

$$R \div S = \pi[Z]R - \pi[Z](((\pi[Z]R) \times S) - R)$$

$R \div S$:

```
SELECT DISTINCT Z1, ..., Zk
FROM R
EXCEPT
SELECT Z1, ..., Zk FROM
  (SELECT *
   FROM (SELECT Z1, ..., Zk FROM R)
        CROSS JOIN (SELECT * FROM S)
   EXCEPT
    SELECT * FROM R)
```

Nachteil: SQL-Ausdruck enthält ein kartesisches Produkt!

Division á la Algebra

Welche Länder sind Mitglied in denselben Organisationen wie Österreich? ²

```

SELECT DISTINCT LCode FROM Mitglied
MINUS /* EXCEPT */
SELECT Lcode FROM
(
  SELECT LCode, Organisation FROM
    (SELECT LCode FROM Mitglied)
  CROSS JOIN
    (SELECT Organisation FROM Mitglied WHERE LCode = 'A')
  MINUS /* EXCEPT */
  SELECT LCode, Organisation FROM Mitglied
)

```

²Es interessiert hier nur der LCode - d.h. die Division in der Algebra lautet $\pi[LCode, Organisation]Mitglied \div \pi[Organisation](\sigma[LCode = 'A']Mitglied)$.

Division ohne kartesisches Produkt

Welche Länder sind Mitglied in denselben Organisationen wie Österreich?

```
SELECT DISTINCT LCode
  FROM Mitglied M
 WHERE NOT EXISTS
   ((SELECT Organisation FROM Mitglied
      WHERE LCode = 'A')
  EXCEPT
   (SELECT Organisation FROM Mitglied
      WHERE LCode = M.LCode))
```

Welche Länder sind Mitglied in denselben Organisationen wie Österreich?

```
SELECT DISTINCT LCode
  FROM Mitglied M
 WHERE NOT EXISTS
   ((SELECT Organisation FROM Mitglied
      WHERE LCode = 'A')
  EXCEPT
   (SELECT Organisation FROM Mitglied
      WHERE LCode = M.LCode))
```

```
(SELECT Organisation FROM Mitglied WHERE LCode = 'A')
```

```
{ EU }
```

```
(SELECT Organisation FROM Mitglied WHERE LCode = M.LCode)
```

<u>M.LCode</u>	<u>Organisationen</u>
A	{ EU }
D	{ EU, WEU }
ET	{ UN }
I	{ EU, NAM }
TR	{ UN, CERN }

Division alternativ A.

Welche Länder sind Mitglied in denselben Organisationen wie Österreich?

```
SELECT DISTINCT LCode
  FROM Mitglied M1
 WHERE NOT EXISTS (
   SELECT M2.Organisation FROM Mitglied M2
  WHERE M2.LCode = 'A' AND NOT EXISTS (
   SELECT Organisation FROM Mitglied M3
  WHERE M3.LCode = M1.LCode AND
        M3.Organisation = M2.Organisation))
```

$$A \setminus B = \{x \mid x \in A \wedge x \notin B\}$$

Welche Länder sind Mitglied in denselben Organisationen wie Österreich?

```
SELECT DISTINCT LCode
  FROM Mitglied M1
 WHERE NOT EXISTS (
   SELECT M2.Organisation FROM Mitglied M2
  WHERE M2.LCode = 'A' AND NOT EXISTS (
   SELECT Organisation FROM Mitglied M3
  WHERE M3.LCode = M1.LCode AND
        M3.Organisation = M2.Organisation))
```

M1	M2	M3	NOT EXISTS innen	NOT EXISTS außen
A	A	A	false	true
D	A	D	false	true
ET	A	ET	true	false
I	A	I	false	true
TR	A	TR	true	false

Division alternativ B.

Welche Länder sind Mitglied in denselben Organisationen wie Österreich?

```
SELECT DISTINCT LCode
  FROM Mitglied M1
 WHERE NOT EXISTS (
   SELECT M2.Organisation FROM Mitglied M2, Mitglied M3
  WHERE M3.LCode = 'A' AND M2.LCode = M1.LCode AND
        M3.Organisation NOT IN (
   SELECT M4.Organisation FROM Mitglied M4
  WHERE M4.LCode = M2.LCode)
 )
```

NOT IN ist problematisch bei Nullwerten (s. Kapitel 4.8(3)).

Division zum Letzten.

Welche Länder sind Mitglied in denselben Organisationen wie Österreich?

```
SELECT DISTINCT M1.LCode
FROM Mitglied M1, Mitglied M2
WHERE M2.LCode = 'A' AND
      M1.Organisation = M2.Organisation
GROUP BY M1.LCode
HAVING COUNT(M1.Organisation) = (
  SELECT COUNT(M3.Organisation)
  FROM Mitglied M3 WHERE M3.LCode = 'A' )
```

Wie verhalten sich die unterschiedlichen Varianten bei einer Division durch eine leere Tabelle?

Gleichheit!

Welche Länder sind Mitglied in GENAU denselben Organisationen wie Österreich?

```
SELECT DISTINCT LCode FROM Mitglied M WHERE
  NOT EXISTS
    ((SELECT Organisation FROM Mitglied WHERE LCode = 'A')
     EXCEPT
    (SELECT Organisation FROM Mitglied WHERE LCode = M.LCode))
AND NOT EXISTS
    ((SELECT Organisation FROM Mitglied WHERE LCode = M.LCode)
     EXCEPT
    (SELECT Organisation FROM Mitglied WHERE LCode = 'A'))
```

4.7 Fortgeschrittenes³

Literatur: Oracle Database SQL Language Reference 11g Release 1 (11.1)

ORA_HASH (expr [,max_bucket] [,seed_value])

- ▶ ORA_HASH is a function that computes a hash value for a given expression.
- ▶ The expr argument determines the data for which you want Oracle Database to compute a hash value.
- ▶ The optional max_bucket argument determines the maximum bucket value returned by the hash function. You can specify any value between 0 and 4294967295. The default is 4294967295.
- ▶ The optional seed_value argument enables Oracle to produce many different results for the same set of data.
- ▶ The function returns a NUMBER value.

Test auf Gleichheit von Tabellen - nur als notwendige Bedingung garantiert korrekt.

```
SELECT DISTINCT LCode FROM Mitglied M WHERE
  (SELECT SUM(ORA_HASH(LCode||Organisation||Art)) FROM Mitglied WHERE LCode = M.LCode) =
  (SELECT SUM(ORA_HASH(LCode||Organisation||Art)) FROM Mitglied WHERE LCode = 'A')
```

³Zur selbständigen, freiwilligen Vertiefung - nicht klausurrelevant!

WITH-Klausel

ROLLUP Operator

CUBE Operator

4.8 Struktur der Syntax

Orthogonalität der Syntax

- ▶ Ein Ausdruck, der eine Tabelle definiert, ist überall dort zulässig ist, wo eine Tabelle stehen darf.
- ▶ Ein skalarer Ausdruck ist überall dort zulässig ist, wo ein skalarer Wert stehen darf.
- ▶ Ein bedingter Ausdruck ist überall dort zulässig ist, wo ein Wahrheitswert stehen darf.

(1) tabellenwertiger Ausdruck

- ▶ Ein Anfrageausdruck definiert eine Tabelle.
- ▶ Jeder Tabellenbezeichner ist ein Anfrageausdruck.
- ▶ Jeder SFW-Ausdruck ist ein Anfrageausdruck.
- ▶ Ein Verbundausdruck ist ebenfalls ein Anfrageausdruck.
- ▶ Die üblichen Mengenoperatoren können verwendet werden, um Anfrageausdrücke zu bilden.
- ▶ Ein *Tabellen-Konstruktor* der Form
`VALUES ('a1', ..., 'an'), ('b1', ..., 'bn'), ...` ist ein Anfrageausdruck.

Liste die Namen, die für Städte und Länder verwendet werden.

```
SELECT Name
  FROM (SELECT SName AS Name
        FROM Stadt UNION
        SELECT LName AS Name
        FROM Land) T
```

Berechne die Anzahl der Menschen aller Länder, die in der größten Stadt ihres Landes leben.

```
SELECT SUM(Großstädter)
  FROM (SELECT LCode, MAX(Einwohner) AS Großstädter
        FROM Stadt
        GROUP BY LCode) T
```

(2) skalarer Anfrageausdruck

Anstelle eines Wertes, bzw Spaltenbezeichners, ist auch ein geklammerter Tabellenausdruck zulässig, sofern er *skalar* ist, d.h. genau einen Wert definiert.

Bestimme zu jeder Stadt den Mittelwert der Einwohnerzahl aller Städte, die weniger Einwohner haben als sie selbst.

```
SELECT SName, Einwohner,
       (SELECT AVG(Einwohner) FROM Stadt S2
        WHERE S2.Einwohner < S1.Einwohner)
       AS kleinerMittelwert
FROM Stadt S1
```

Bestimme diejenigen asiatischen Länder, deren Flächenanteil in Asien kleiner ist als der Anteil der Türkei in Asien.

```
SELECT DISTINCT LCode, Prozent FROM Lage
WHERE Kontinent = 'Asia' AND
      Prozent <
      (SELECT Prozent FROM Lage
       WHERE LCode = 'TR' AND Kontinent = 'Asia')
```

(3) bedingter Ausdruck

- ▶ Ein Ausdruck, dem ein Wahrheitswert zugeordnet werden kann, ist ein bedingter Ausdruck.
- ▶ Bedingte Ausdrücke sind Teil einer WHERE-, HAVING- und ON-Klausel.
- ▶ Wesentlich für die korrekte Verwendung bedingter Ausdrücke ist die Miteinbeziehung des Auftretens von Nullwerten.

Vergleiche unter Annahme Tabelle Land enthält Wunderland mit Nullwert für HStadt:

Welche Städte (SName) sind keine Hauptstadt von irgendeinem Land?

```
SELECT SName FROM Stadt S
  WHERE S.SName NOT IN (SELECT HStadt FROM Land)
```

Resultat leere Tabelle.

mit:

Welche Städte (SName) sind keine Hauptstadt von irgendeinem Land?

```
SELECT SName FROM Stadt S
  WHERE NOT EXISTS (
    SELECT HStadt FROM Land
    WHERE HStadt = S.SName )
```

Resultat Freiburg, Munich, Nuremberg, Karlsruhe.