

Datenbanken 1 Praktikum 9

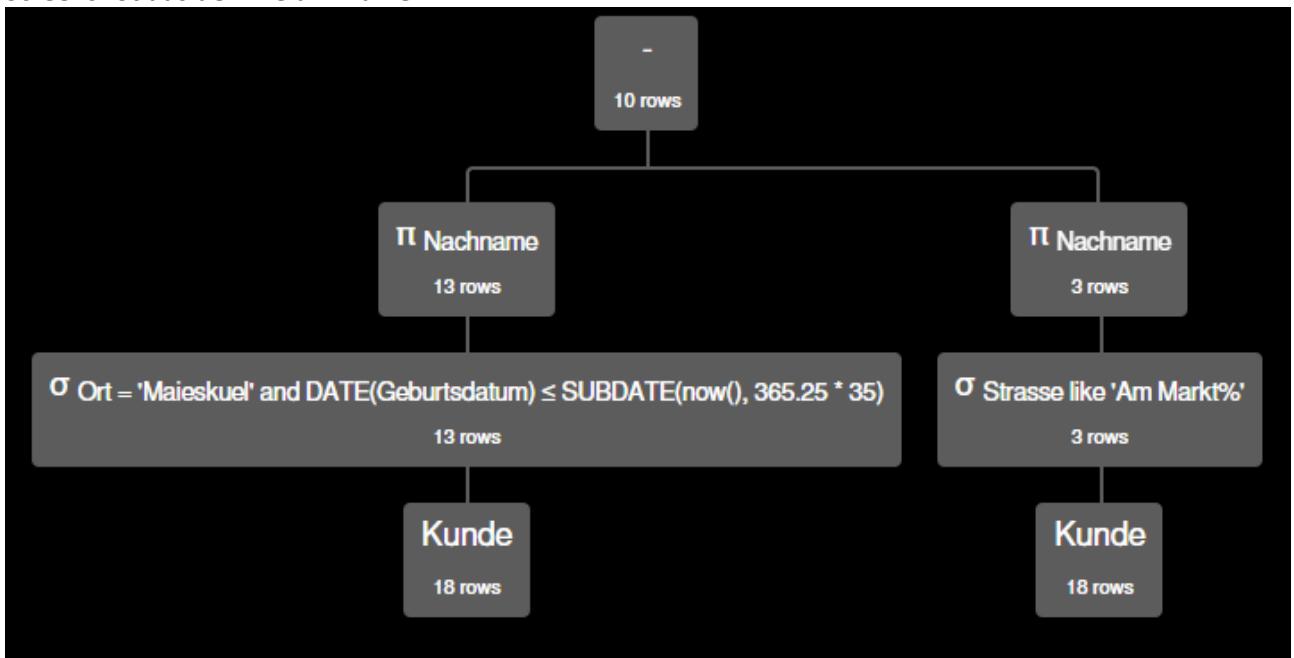
Aufgabe 1 – Abfragen in SQL

- a) Zeigen Sie die Nachnamen aller Kunden an, die in Maieskuel, jedoch nicht in der Straße „Am Markt“ wohnen und die mindestens 35 Jahre alt sind. Geben Sie die Abfrage in der relationalen Algebra an. Nutzen Sie eine Mengenoperation.

RelaX-Trainer

```
π Nachname (σ Ort = 'Maieskuel' ∧
date(Geburtsdatum) ≤ subDate(now(), 365.25
* 35) Kunde)
-
π Nachname (σ Strasse LIKE 'Am Markt%'
Kunde)
```

Screenshot aus dem RelaX-Trainer:



- b) Wie lautet die SQL-Abfrage zu Teilaufgabe a? Nutzen Sie auch hierbei eine Mengenoperation.

```
SELECT Nachname FROM Kunde WHERE FLOOR ((SYSDATE - Geburtsdatum)/365) >35
```

```
MINUS
```

```
SELECT Nachname FROM Kunde WHERE UPPER(Strasse) LIKE 'AM MARKT%'
```

```
ORDER BY Nachname DESC;
```

	NACHNAME
1	Wille
2	S. S. Assin
3	Ratte
4	Mueller
5	Kratzbaum
6	Döhmer
7	Duck
8	Drach

- c) Listen Sie mittels einer Abfrage die Bezeichnungen aller Artikelkategorien der milchhaltigen Getränke auf. Beachten Sie dabei, dass jede Artikelkategorie auch Unterkategorien beinhalten kann. Nutzen Sie zur Lösung Mengenoperationen.

Datenbanken 1 Praktikum 9

Mengenoperationen und Subquery (Hier muss vorher klar sein, wie tief die Artikelkategorien verschachtelt sind):

```
SELECT Bezeichnung
FROM Artikelkategorie
WHERE bezeichnung like 'Milchhaltige'
UNION
SELECT Bezeichnung
FROM Artikelkategorie
WHERE Oberkategorie IN (
SELECT KategorieKrz
FROM Artikelkategorie
WHERE bezeichnung like 'Milchhaltige')
UNION
SELECT Bezeichnung
FROM Artikelkategorie
WHERE oberkategorie IN (
SELECT KategorieKrz
FROM Artikelkategorie a
WHERE oberkategorie IN (
SELECT KategorieKrz
FROM Artikelkategorie
WHERE bezeichnung like 'Milchhaltige')
);
```

d) Lösen Sie die Teilaufgabe c) mit Hilfe von Verbundoperationen.

Mit Verbundoperationen (Hier muss vorher klar sein, wie tief die Artikelkategorien verschachtelt sind):

```
SELECT ak1.Bezeichnung, ak2.Bezeichnung, ak3.Bezeichnung
FROM Artikelkategorie ak1
LEFT OUTER JOIN Artikelkategorie ak2
ON ak1.KategorieKrz = ak2.Oberkategorie
LEFT OUTER JOIN Artikelkategorie ak3
ON ak2.KategorieKrz = ak3.Oberkategorie
WHERE ak1.Bezeichnung like 'Milchhaltige';
```

BEZEICHNUNG	BEZEICHNUNG_1	BEZEICHNUNG_2
1 Milchhaltige Milchsorte		Kaffeegetraenk
2 Milchhaltige Milchsorte		H-Milch
3 Milchhaltige Milchsorte		Kakao

e) Lösen Sie die Teilaufgabe c) mit Hilfe der rekursiven WITH-Klausel. Was ist der Vorteil dieser Methode gegenüber den Lösungen in Teilaufgaben c) und d)?

Mit der rekursiven WITH-Klausel:

```
WITH ak(KategorieKrz, Oberkategorie, Bezeichnung) AS (
-- Start-Kategorie
SELECT KategorieKrz,
Oberkategorie,
Bezeichnung
FROM Artikelkategorie
WHERE Bezeichnung like 'Milchhaltige'
UNION ALL
-- Unterkategorie
SELECT child.KategorieKrz, child.Oberkategorie, child.Bezeichnung
FROM Artikelkategorie child, ak
WHERE child.Oberkategorie = ak.KategorieKrz)
```

Datenbanken 1 Praktikum 9

```
)  
SEARCH DEPTH FIRST BY KategorieKrz SET order1  
SELECT Bezeichnung  
FROM ak  
ORDER BY order1;
```

◆ BEZEICHNUNG
1 Milchhaltige
2 Milchsorte
3 Kaffeegetraenk
4 H-Milch
5 Kakao

Die Suchtiefe ist in den Lösungen c) und d) auf zwei Hierarchiestufen begrenzt. Durch den rekursiven Durchlauf der Hierarchie können auch tiefer verschachtelte Hierarchien durchsucht werden.

- f) Lösen Sie die Teilaufgabe c) mit Hilfe von CONNECT BY PRIOR.

Mit CONNECT BY PRIOR (Oracle-spezifisch)

```
SELECT Bezeichnung FROM Artikelkategorie  
START WITH Bezeichnung like 'Milchhaltige'  
CONNECT BY PRIOR KategorieKrz = Oberkategorie;
```

◆ BEZEICHNUNG
1 Milchhaltige
2 Milchsorte
3 Kaffeegetraenk
4 H-Milch
5 Kakao