

## Aufgabe 1 – Abfragen in SQL

Finden Sie bitte für **alle** Aufgaben **zwei** strukturell unterschiedliche Lösungen. Beispielsweise nutzt dann eine Lösung mit Verbundoperationen (Joins) und die andere Lösung Unterabfragen (Subqueries).

Geben Sie die zugehörenden SQL-Abfragen an und führen Sie diese auf der Datenbank aus:

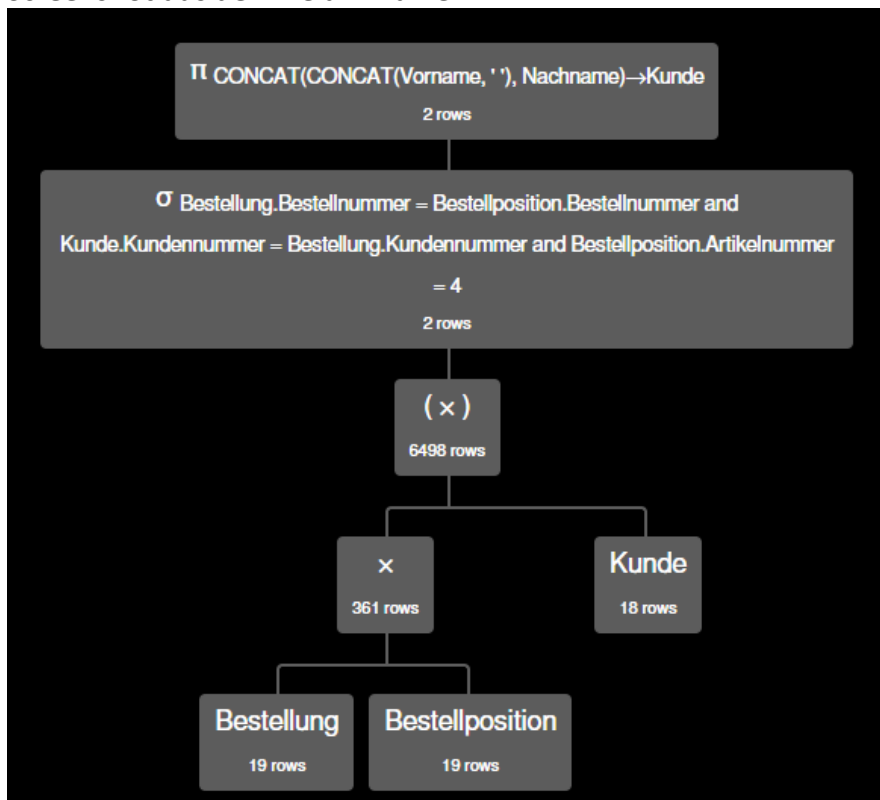
Kunde
A. S. S. Assin
Amber Love

- a) Listen Sie alle Kunden, die Artikel 4 bestellt haben mit den Angaben Vorname und Nachname. Die Ausgabe soll in folgender Form erfolgen:

## Variante 1 – Verbundoperation (Cross Join):

SQL	RelaX-Trainer
<pre>SELECT Vorname    ' '    Nachname as Kunde FROM Bestellung b, Bestellposition p, Kunde k WHERE b.Bestellnummer=p.Bestellnummer AND k.Kundennummer=b.Kundennummer AND p.Artikelnummer=4;</pre>	<pre><math>\pi</math> concat(concat(Vorname, ' '), Nachname)→Kunde <math>\sigma</math> Bestellung.Bestellnummer = Bestellposition.Bestellnummer and Kunde.Kundennummer = Bestellung.Kundennummer and Bestellposition.Artikelnummer = 4 (Bestellung <math>\times</math> Bestellposition <math>\times</math> Kunde)</pre>

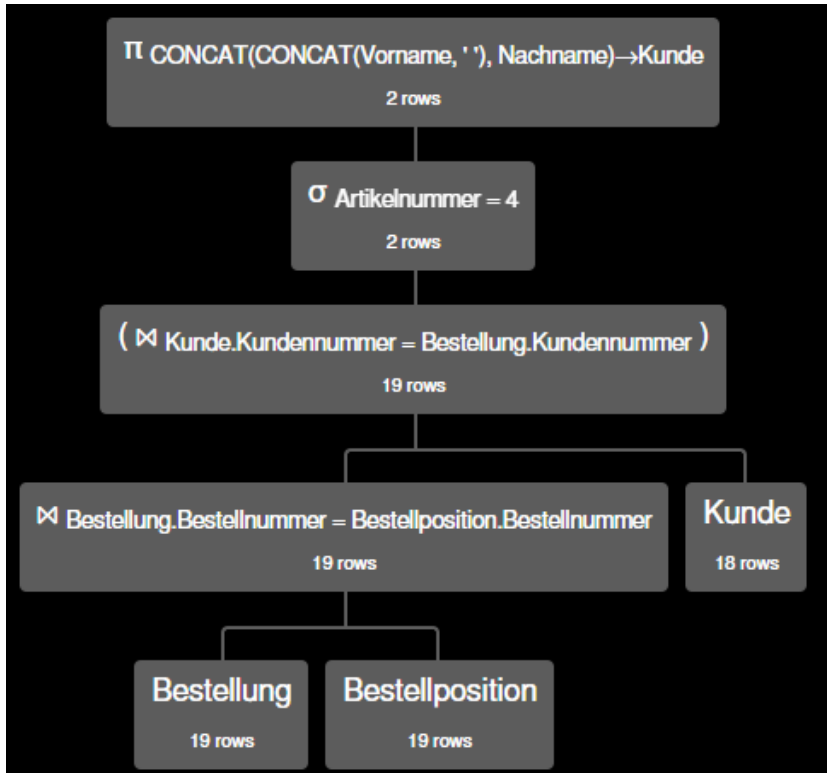
## Screenshot aus dem RelaX-Trainer:



## Variante 2 – Verbundoperation:

SQL	RelaX-Trainer
<pre>SELECT Vorname    ' '    Nachname As Kunde FROM Bestellung b JOIN Bestellposition bp ON b.Bestellnummer=bp.Bestellnummer JOIN Kunde k on k.Kundennummer=b.Kundennummer WHERE Artikelnummer=4;</pre>	$\pi \text{ concat}(\text{concat}(\text{Vorname}, ' '), \text{Nachname}) \rightarrow \text{Kunde} \sigma \text{ Artikelnummer} = 4$ $(\text{Bestellung} \bowtie \text{Bestellung.Bestellnummer} = \text{Bestellposition.Bestellnummer} \text{ Bestellposition} \bowtie \text{Kunde.Kundennummer} = \text{Bestellung.Kundennummer} \text{ Kunde})$

Screenshot aus dem RelaX-Trainer:



**Variante 3 – Unterabfrage:**

```

Select Vorname || ' ' || Nachname AS Kunde
FROM Kunde
WHERE Kundennummer IN
(SELECT DISTINCT Kundennummer FROM Bestellung NATURAL JOIN Bestellposition
WHERE Artikelnummer=4);

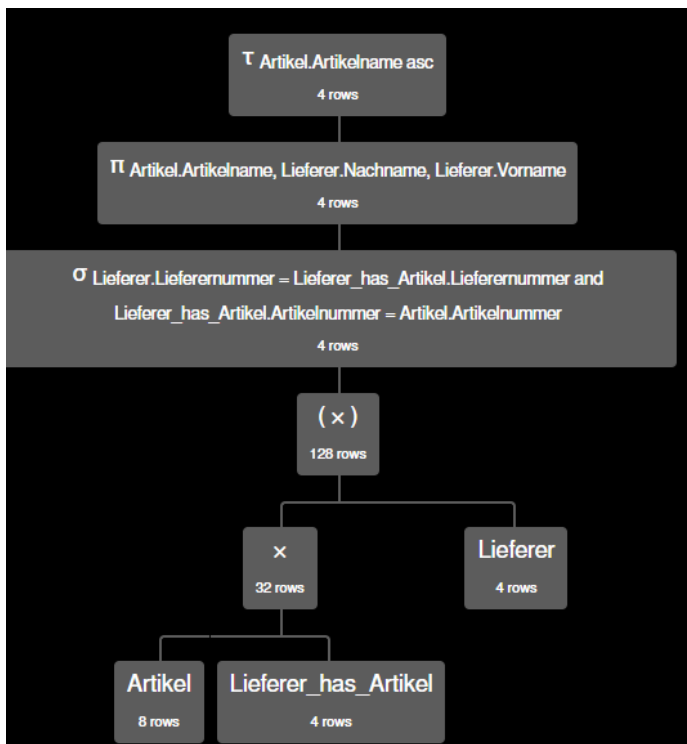
```

- b) Listen Sie alle Lieferanten, sortiert nach den Artikelbezeichnungen der Artikel die sie liefern können mit Angaben von Vorname, Nachname und Name des Artikels.

**Variante 1 – Verbundoperation (Cross Join):**

SQL	RelaX-Trainer
<pre>SELECT Vorname, Nachname, Artikelname FROM Lieferer l, Lieferer_has_artikel lha, Artikel a WHERE l.Lieferernummer=lha.Lieferernummer AND a.artikelnummer=lha.artikelnummer ORDER BY a.Artikelname;</pre>	$\tau \text{ Artikel.Artikelname asc } \pi \text{ Artikel.Artikelname, Lieferer.Nachname, Lieferer.Vorname } \sigma$ $\text{Lieferer.Lieferernummer} = \text{Lieferer\_has\_Artikel.Lieferernummer and}$ $\text{Lieferer\_has\_Artikel.Artikelnummer} = \text{Artikel.Artikelnummer (Artikel x Lieferer\_has\_Artikel x Lieferer)}$

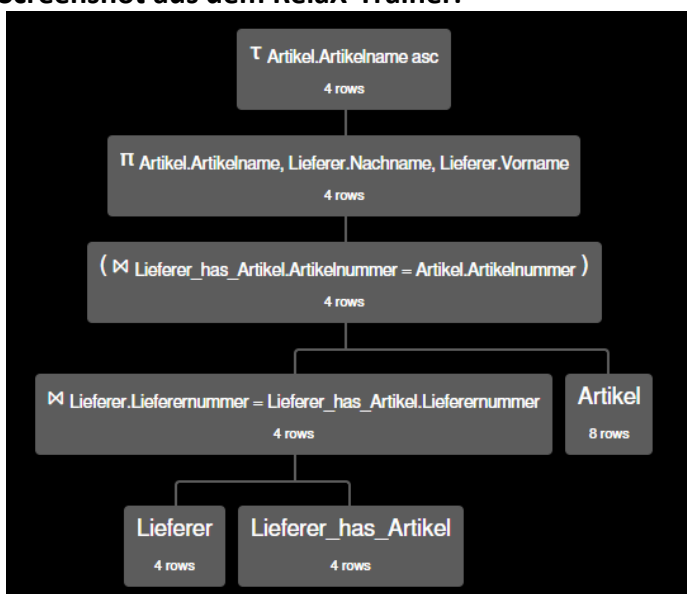
Screenshot aus dem RelaX-Trainer:



## Variante 2 – Verbundoperation:

SQL	RelaX-Trainer
<pre>select vorname, nachname, artikelname from artikel natural join lieferer_has_artikel natural join lieferer order by artikelname;</pre>	<pre><math>\tau</math> Artikel.Artikelname asc <math>\pi</math> Artikel.Artikelname, Lieferer.Nachname, Lieferer.Vorname (Lieferer <math>\bowtie</math> Lieferer.Lieferernummer = Lieferer_has_Artikel.Lieferernummer Lieferer_has_Artikel <math>\bowtie</math> Lieferer_has_Artikel.Artikelnummer = Artikel.Artikelnummer Artikel)</pre>

## Screenshot aus dem RelaX-Trainer:



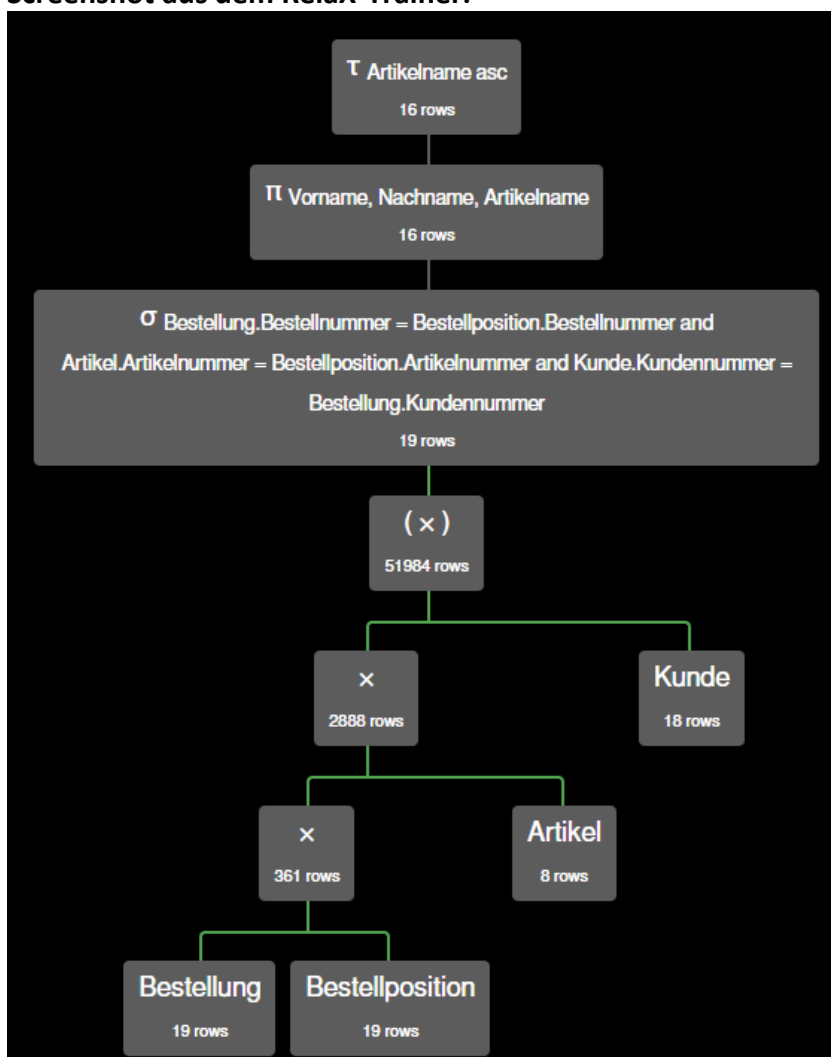
**Variante 3 – Unterabfrage:**

```
SELECT Vorname, Nachname, Artikelname
FROM Lieferer Natural JOIN
(SELECT Artikelname, lieferernummer
FROM lieferer_has_artikel lha Natural JOIN Artikel a) la
ORDER BY artikelname;
```

- c) Listen Sie alle Kunden, die Bestellungen haben. Listen Sie neben dem Vornamen und dem Nachnamen auch die Artikelbezeichnung der bestellten Artikel sortiert nach der Artikelbezeichnung.

**Variante 1 – Verbundoperation (Cross Join):**

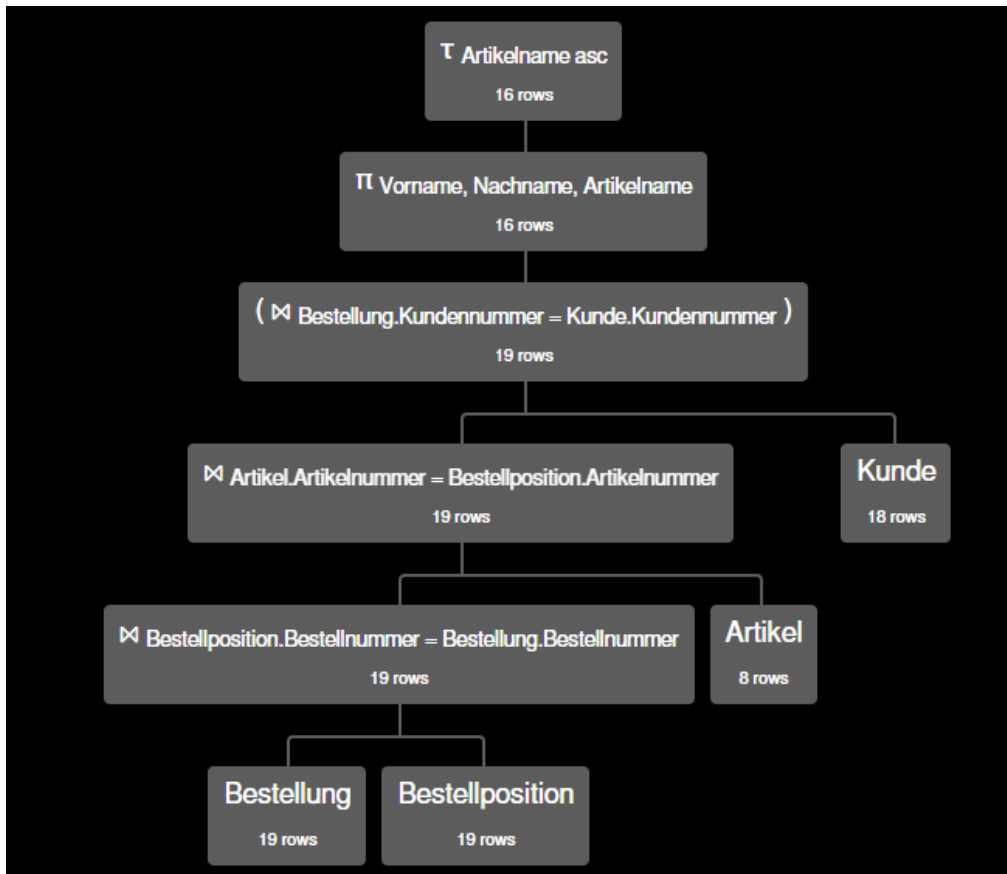
SQL	RelaX-Trainer
<pre>SELECT Vorname, Nachname, Artikelname FROM Bestellung b, Bestellposition p, Artikel a, Kunde k WHERE b.Bestellnummer=p.Bestellnummer AND a.Artikelnummer=p.Artikelnummer AND k.Kundennummer=b.Kundennummer ORDER BY Artikelname;</pre>	$\tau$ Artikelname asc $\pi$ Vorname, Nachname, Artikelname $\sigma$ Bestellung.Bestellnummer = Bestellposition.Bestellnummer and Artikel.Artikelnummer = Bestellposition.Artikelnummer and Kunde.Kundennummer = Bestellung.Kundennummer (Bestellung $\times$ Bestellposition $\times$ Artikel $\times$ Kunde)

**Screenshot aus dem RelaX-Trainer:**

## Variante 2 – Verbundoperation:

SQL	RelaX-Trainer
SELECT Vorname, Nachname, Artikelname FROM Bestellung b JOIN Bestellposition bp ON bp.Bestellnummer=b.Bestellnummer JOIN Artikel a on a.Artikelnummer=bp.Artikelnummer JOIN Kunde k on b.Kundennummer=k.Kundennummer ORDER BY Artikelname;	$\tau$ Artikelname asc $\pi$ Vorname, Nachname, Artikelname (Bestellung $\bowtie$ Bestellposition.Bestellnummer = Bestellung.Bestellnummer Bestellposition $\bowtie$ Artikel.Artikelnummer = Bestellposition.Artikelnummer Artikel $\bowtie$ Bestellung.Kundennummer = Kunde.Kundennummer Kunde)

## Screenshot aus dem RelaX-Trainer:



## Variante 3 – Unterabfrage:

```

SELECT (SELECT Vorname FROM Kunde WHERE Kundennummer = b.Kundennummer),
(SELECT Nachname FROM Kunde WHERE Kundennummer = b.Kundennummer) ,
(SELECT Artikelname FROM Artikel WHERE Artikelnummer=bp.Artikelnummer) Artikelname
FROM Bestellposition bp JOIN Bestellung b USING (Bestellnummer)
ORDER BY Artikelname;

```