

Datenbanken 1 - Praktikum 3

Hinweis:

Für die Praktikumsaufgaben wird der [Relational Algebra Trainer](#) für die Datenbank verwendet.

Beachten Sie, dass der Relational Algebra Trainer case-sensitive ist. Zwischen Groß- und Kleinschreibung wird also unterschieden.

Aufgabe 1 - Abfragen in der relationalen Algebra

- a) Zeigen Sie die Vornamen aller weiblichen Kunden.

π vorname (σ anrede = 'Frau' (kunde))

- b) Zu welcher Artikelkategorie gehört der Artikel mit der Bezeichnung 'Multisaft'?
Tipp: Verbundbedingung formulieren!

π Artikelkategorie.Bezeichnung σ Artikel.Artikelname = 'Multisaft' (Artikel \bowtie Artikel.KategorieKrz = Artikelkategorie.KategorieKrz (Artikelkategorie))

- c) Zeigen Sie die Anzahl aller Kunden aus Maieskuel an.

γ ort; Count(Kundennummer) \rightarrow Anzahl (σ ort = 'Maieskuel' Kunde)

- d) Zeigen Sie die Nachnamen aller Kunden aus Maieskuel an, die nicht in der Straße „Am Markt“ wohnen.
Nutzen Sie einen Mengenoperator.
Sortieren Sie absteigend nach dem Nachnamen.

τ Nachname DESC (π Nachname (σ ort = 'Maieskuel' Kunde) - π Nachname (σ strasse LIKE 'Am Markt%' (Kunde)))

- e) Listen Sie die Namen aller Lieferer, die das Getränk „Multisaft“ liefern können. Die Liste soll aufsteigend sortiert ausgegeben werden.
Benennen Sie die Ergebnisspalte in „Lieferer“ um.

ρ Lieferer \leftarrow I.Vorname τ Vorname ASC π I.Vorname σ a.Artikelname = 'Multisaft' (((ρ a Artikel) \bowtie a.Artikelnummer = Iha.Artikelnummer (ρ Iha Lieferer_has_Artikel)) \bowtie I.Lieferernummer = Iha.Lieferernummer ρ I (Lieferer))

- f) Listen Sie die Vor- und Nachnamen aller Kunden, die den Artikel „Pils“ im Warenkorb haben.

π Kunde.Vorname, Kunde.Nachname (σ Artikel.Artikelname LIKE 'Pils' (Artikel \bowtie Artikel.Artikelnummer = Warenkorb.Artikelnummer Warenkorb \bowtie Warenkorb.Kundennummer = Kunde.Kundennummer Kunde))

Aufgabe 2 – Von der relationalen Algebra in natürliche Sprache

Was bewirken die folgenden relationalen Ausdrücke?

Datenbanken 1 - Praktikum 3

- a) $\pi \text{ num } (\sigma \text{ Bestellnummer} = \text{null } (\gamma \text{ Bestellnummer; COUNT(*) } \rightarrow \text{ num } (\text{Kunde} \bowtie \text{Bestellung})))$

Alle Kunden, die keine Bestellung getätigt haben, werden aufgezählt.

- b) $\tau \text{ Anzahl desc } (\gamma \text{ Kunde.Kundennummer, Kunde.Nachname; COUNT(Bestellnummer) } \rightarrow \text{ Anzahl } (\text{Kunde} \bowtie \text{Bestellung}))$

Alle Kunden mit der Angabe von Bestellnummer, Nachname und der Anzahl der Bestellungen, die mindestens eine Bestellung gemacht haben. Die Ausgabe ist nach der Anzahl der Bestellungen absteigend sortiert.

- c) $\pi \text{ Nachname } (\sigma \text{ Ort} = \text{'Maieskuel'} \wedge \text{Anrede} = \text{'Herr'} (\text{Kunde}))$
- $\pi \text{ Nachname } (\sigma \text{ Strasse like 'Rotlichtgasse\%'} (\text{Kunde}))$

Die Nachnamen aller männlichen Kunden aus Maieskuel, die nicht in der Straße „Rotlichtgasse“ wohnen.

- d) $\tau \text{ AnzBestellung desc } (\gamma \text{ k.Ort; COUNT(b.Bestellnummer) } \rightarrow \text{ AnzBestellung } ((\rho \text{ k Kunde }) \bowtie \text{ k.Kundennummer} = \text{ b.Kundennummer } (\rho \text{ b Bestellung})))$

Die Anzahl der Bestellungen je Ort absteigend sortiert nach dem Ort.