

# Bank-Datenbank

## SQL-Übungen

### Lösungen

#### Aufgabe 1 – Grundoperationen

- a) Finde alle Kredite mit einem Betrag > 22000 \$.  
`SELECT * FROM kredit WHERE Betrag > 22000 ORDER BY Betrag`
- b) Ermittle die Namen aller Kunden.  
`SELECT KuName FROM kunde ORDER BY KuName`
- c) Finde die Kreditnummern aller Kredite mit einem Betrag > 22000 \$.  
`SELECT Knummer, Betrag FROM kredit WHERE Betrag > 22000 ORDER BY Betrag`
- d) Ermittle die Namen aller Kunden, die in Detroit wohnen.  
`SELECT KuName FROM kunde WHERE KuStadt = "Detroit" ORDER BY KuName`
- e) Bestimme den Namen des Angestellten mit der Telefonnummer 0815/4711.  
`SELECT AName FROM angestellter WHERE TelefonNr = "0815/4711"`
- f) Gib die Namen der Angestellten an, die eine Telefonnummer haben, die mit 0815 beginnt.  
`SELECT AName FROM angestellter WHERE TelefonNr LIKE "0815%" ORDER BY AName`
- g) Finde die Sozialversicherungsnummer aller Kunden, die über ein Konto oder einen Kredit verfügen.  
`SELECT KuSozVersNr FROM inhaber  
UNION SELECT KuSozVersNr FROM schuldner  
ORDER BY KuSozVersNr`
- h) Gib die Sozialversicherungsnummer aller Kunden an, die ein Konto und einen Kredit besitzen.  
`SELECT DISTINCT KuSozVersNr FROM inhaber  
WHERE KuSozVersNr IN (SELECT KuSozVersNr FROM schuldner)  
ORDER BY KuSozVersNr`
- i) Gib die Sozialversicherungsnummer aller Kunden an, die ein Konto, aber keinen Kredit haben.  
`SELECT KuSozVersNr FROM inhaber EXCEPT (SELECT KuSozVersNr FROM schuldner)`
- j) Finde die Namen aller Kontoinhaber.  
`SELECT DISTINCT KuName FROM kunde, inhaber  
WHERE kunde.KuSozVersNr = inhaber.KuSozVersNr ORDER BY KuName`
- k) Bestimme die Kredite, die in der Zweigstelle Elmwood verwaltet werden.  
`SELECT Knummer, Betrag FROM kredit WHERE ZwName = "Elmwood" ORDER BY Knummer`

- l) Ermittle die Nummern der Kredite, die in der Zweigstelle Elmwood verwaltet werden und einen Betrag über 10000\$ haben.

```
SELECT KNummer, Betrag FROM kredit
WHERE ZwName="Elmwood" and Betrag > 10000 ORDER BY KNummer
```

- m) (schwer!) Bestimme die Sozialversicherungsnummer aller Kunden, die über ein Girokonto, aber nicht über ein Sparkonto verfügen.

```
SELECT DISTINCT KuSozVersNr
FROM girokonto INNER JOIN inhaber USING(KontoNr)
WHERE KuSozVersNr NOT IN
      (SELECT DISTINCT KuSozVersNr
       FROM sparkonto INNER JOIN inhaber USING(KontoNr))
```

- n) Finde die Namen aller Kunden, die ein Sparkonto besitzen.

```
SELECT DISTINCT KuName FROM kunde, inhaber, sparkonto
WHERE kunde.KuSozVersNr = inhaber.KuSozVersNr
      AND inhaber.KontoNr = sparkonto.KontoNr
ORDER BY KuName
```

- o) Ermittle die Sozialversicherungsnummer nebst Namen aller Kunden, die einen Kredit bei der Zweigstelle River Site haben.

```
SELECT kunde.KuSozVersNr, kunde.KuName
FROM kunde JOIN schuldner ON kunde.KuSozVersNr = schuldner.KuSozVersNr
      JOIN kredit ON schuldner.KNummer = kredit.KNummer
WHERE kredit.ZwName = "River Site" ORDER BY kunde.KuSozVersNr
```

- p) Finde heraus, welche Kunden den gleichen Namen wie ihr Berater haben.

```
SELECT kunde.KuName AS Kunde, angestellter.AName AS Berater
FROM kunde JOIN ansprechpartner USING(KuSozVersNr)
      JOIN angestellter USING(ASozVersNr)
WHERE kunde.KuName = angestellter.AName ORDER BY kunde.KuName
```

- q) Bestimme den höchsten Girokontostand und die zugehörige Kontonummer.

```
SELECT Kontostand, KontoNr
FROM girokonto
WHERE Kontostand = (SELECT MAX(Kontostand) FROM girokonto)
```

- r) Gib die Namen aller Kunden an, die in der gleichen Stadt und Straße wohnen wie Taylor, Tim.

```
SELECT KuName
FROM kunde
WHERE KuStadt = (SELECT KuStadt FROM kunde WHERE KuName = "Taylor, Tim")
      AND KuStrasse = (SELECT KuStrasse FROM kunde WHERE KuName = "Taylor, Tim")
ORDER BY KuName
```

## Aufgabe 2 – erweiterte Operationen

- a) Ermittle, wie viel Geld (zusammen) auf allen Girokonten der Bank vorhanden ist.
- ```
SELECT SUM(Kontostand) AS GiroGesamtbetrag FROM girokonto
```
- b) Finde den höchsten Kreditbetrag, der bei der Bank vergeben wurde, und den zugehörigen Kunden.
- ```
SELECT MAX(Betrag) AS Höchstkreditbetrag, KuName
FROM kredit JOIN schuldner ON kredit.Knummer = schuldner.Knummer
      JOIN kunde ON schuldner. KuSozVersNr = kunde. KuSozVersNr
WHERE Betrag = (SELECT MAX(Betrag) FROM Kredit)
```
- c) Gib für jeden Kunden den durchschnittlichen Zinssatz seiner Sparkonten (hier eindeutig) an.
- ```
SELECT inhaber.KuSozVersNr, AVG(sparkonto.Zinssatz) AS DurchschnittsZinssatz
FROM inhaber, sparkonto
WHERE inhaber.KontoNr = sparkonto.KontoNr
GROUP BY inhaber.KuSozVersNr
```
- d) Bestimme für jeden Kunden die Anzahl seiner Girokonten.
- ```
SELECT inhaber.KuSozVersNr, COUNT(girokonto.KontoNr) AS AnzahlGirokonten
FROM inhaber, girokonto
WHERE inhaber.KontoNr = girokonto.KontoNr
GROUP BY inhaber.KuSozVersNr
```
- e) Ermittle, über wie viel Geld jeder Kunde auf seinen Girokonten verfügen kann.
- ```
SELECT inhaber.KuSozVersNr, SUM(girokonto.Kontostand) AS VerfügbaresGeld
FROM inhaber, girokonto
WHERE inhaber.KontoNr = girokonto.KontoNr
GROUP BY inhaber.KuSozVersNr
```
- f) Finde für jede Stadt die Anzahl ihrer Zweigstellen und die Gesamteinlagen aller Zweigstellen der Stadt.
- ```
SELECT ZwStadt, COUNT(ZwName) AS AnzahlZwStellen, SUM(Einlagen) AS Gesamteinlagen
FROM zweigstelle
GROUP BY ZwStadt
```
- g) Ermittle die Anzahl der Kunden, die über Sparkonten mit gleichem Zinssatz und Kontostand verfügen.
- ```
SELECT Kontostand, Zinssatz, COUNT(inhaber.KuSozVersNr) AS AnzahlKunden
FROM sparkonto, inhaber
WHERE sparkonto.KontoNr = inhaber.KontoNr
GROUP BY Zinssatz, Kontostand
```

### Aufgabe 3 – abgeleitete Operationen

- a) Bestimme die Sozialversicherungsnummer aller Kunden, die einen Kredit und ein Konto besitzen.  
SELECT DISTINCT inhaber.KuSozVersNr FROM inhaber, schuldner  
WHERE inhaber.KuSozVersNr = schuldner.KuSozVersNr ORDER BY inhaber.KuSozVersNr
- b) Finde die Sozialversicherungsnummern aller Kreditnehmer nebst Kreditnummer und Betrag.  
SELECT schuldner.KuSozVersNr, kredit.KNummer, kredit.Betrag  
FROM schuldner JOIN kredit ON schuldner.KNummer = kredit.KNummer  
ORDER BY schuldner. KuSozVersNr
- c) Finde die Namen und Anschriften aller Kreditnehmer der Bank sowie den Betrag des Kredits.  
SELECT kunde.KuName, kunde.KuStrasse, kunde.KuStrasse, kredit.Betrag  
FROM kunde JOIN schuldner ON kunde.KuSozVersNr = schuldner.KuSozVersNr  
JOIN kredit ON schuldner.KNummer = kredit.KNummer  
ORDER BY kunde.KuName
- d) Ermittle die Zweigstellennamen aller Kunden, die einen Kredit haben und in Detroit wohnen.  
SELECT kredit.ZwName FROM kunde, schuldner, kredit  
WHERE kunde.KuStadt = "Detroit" AND kunde.KuSozVersNr = schuldner.KuSozVersNr  
AND schuldner.KNummer = kredit.KNummer  
ORDER BY kredit.ZwName
- e) (schwer!) Bestimme die Namen aller Kunden, die sowohl ein Spar- wie auch ein Girokonto haben.  
SELECT kunde.KuName FROM kunde  
WHERE kunde.KuSozVersNr IN  
(SELECT inhaber.KuSozVersNr FROM inhaber JOIN girokonto USING(KontoNr))  
AND kunde.KuSozVersNr IN  
(SELECT inhaber.KuSozVersNr FROM inhaber JOIN sparkonto USING(KontoNr))  
ORDER BY kunde.KuName
- f) Ermittle die Namen der Angestellten, die Kunden aus Houston betreuen, sowie die Anzahl der jeweils betreuten Kunden.  
SELECT angestellter.AName, count(angestellter.AName) AS AnzahlKunden  
FROM angestellter JOIN ansprechpartner USING(ASozVersNr)  
JOIN kunde USING(KuSozVersNr)  
WHERE kunde.KuStadt="Houston"  
GROUP BY angestellter.AName
- g) (schwer!) Gib die Namen aller Kunden an, die einen Kredit bei jeder Zweigstelle in Cleveland haben.  
SELECT kunde.KuName  
FROM kunde join schuldner on kunde.KuSozVersNr = schuldner.KuSozVersNr  
JOIN kredit ON schuldner.KNummer = kredit.KNummer  
JOIN zweigstelle ON kredit.ZwName = zweigstelle.ZwName  
WHERE zweigstelle.ZwStadt = "Cleveland"  
GROUP BY kunde.KuName  
HAVING COUNT(kunde.KuName)  
= (SELECT COUNT(ZwName) FROM zweigstelle WHERE ZwStadt = "Cleveland")