

# Versicherung - Die Datenbank

Eine Versicherung benötigt eine Datenbank.

## Beschreibung der Datenbank

Mitarbeiter sind eindeutig identifiziert durch ihre Mitarbeiternummer (MNR). Es wird auch noch der Vorname (VNAME), der Nachname (NNAME), die Sozialversicherungsnummer (SVNR) und das Geschlecht (GESCHLECHT) gespeichert. Optional kann für gekündigte Mitarbeiter ein Kündigungsdatum (DATUM) gespeichert werden. Es gibt zwei unterschiedliche Typen von Mitarbeitern: Vertreter und Kundenbetreuer. Bei Vertretern ist bekannt, welchen Dienstwagen (DIENSTWAGEN) sie benutzen. Bei Kundenbetreuern sind die Telefonnummer (TELEFON) und E-Mail-Adresse (EMAIL) bekannt. Alle Mitarbeiter besitzen einen anderen Mitarbeiter als Vorgesetzten, ausgenommen jene, die sich in der obersten Hierarchieebene befinden. Die Produkte der Versicherung sind eindeutig identifiziert durch die Produkt-ID (PID). Zusätzlich werden der Titel (TITEL), der Preis (PREIS) und die maximal gedeckte Schadenshöhe (MAXSCHADEN) eines jeden Produkts in der Datenbank abgelegt. Ein Produkt gehört jeweils zu einer Kategorie, wobei einer Kategorie beliebig viele Produkte zugeordnet werden können. Bei einer Kategorie sind folgende Eigenschaften bekannt: Eine global einzigartige Kategorie-ID (KATID) sowie der Titel der Kategorie (TITEL). Ein Vertreter kann nun beliebig viele Produkte vertreiben und erhält dazu jeweils die Vorgabe (VORGABE), welche Menge er von einem beliebigen Produkt verkaufen sollte. Kunden sind eindeutig identifiziert durch ihre Kundennummer (KNR). Es werden bei Kunden weiters der Vorname (VNAME), der Nachname (NNAME) und das Geburtsdatum (GEBDAT) hinterlegt. Kunden können nun im Zuge von Aufträgen Produkte von Vertretern erwerben. Ein Auftrag ist identifiziert durch den jeweiligen Kunden und das gewählte Produkt, sowie das Kaufdatum (DATUM). Zusätzlich wird noch gespeichert, in welcher Höhe an Prozenten der Kunde einen Rabatt (RABATT) auf den Auftrag bekommt. Ein Kunde kann mehrere Bankverbindungen haben, die eindeutig durch die Bankleitzahl (BLZ) als auch durch die Kontonummer (KNR) identifiziert sind. Bei Bankverbindungen wird zusätzlich der Name der Bank (BANK) abgespeichert und es ist zu bemerken, dass eine Bankverbindung immer zu exakt einem Kunden gehört. Ein Kundenbetreuer berät und unterstützt jeweils eine beliebige Anzahl von Kunden im Umfeld einer ebenfalls beliebigen Anzahl an Themen. Bei Themen sind jeweils der eindeutige Titel (TITEL) sowie auch eine Beschreibung (BESCHREIBUNG) bekannt. Bei auftretenden Schadensfällen können Kunden eine Schadensmeldung einreichen, die identifiziert ist durch eine eindeutige ID. Weiters werden eine Detailinformation (INFO), das Datum (DATUM) und der vermutliche Betrag des Schadens (BETRAG) bei der Schadensmeldung gespeichert. Zur Abwicklung eines Schadensfalles wird noch dessen Bearbeitung in der Datenbank gespeichert. Eine Bearbeitung ist hierbei eindeutig identifiziert durch den Schadensfall, dem sie zugeordnet ist. Als Eigenschaften einer Bearbeitung werden sowohl das Datum (DATUM), eine Anmerkung (ANMERKUNG) als auch der tatsächlich anerkannte Schadensbetrag (BETRAG) gespeichert. Für die Nachverfolgung der Bearbeitung muss auch die verwendete Bankverbindung und der für die jeweilige Bearbeitung zuständige Mitarbeiter persistiert werden.

---

1)

Geben Sie jene aktiven (d.h. nicht gekündigten) Vertreter aus, deren Nachname mit 'Bau' beginnt. Ordnen Sie die Ergebnisse absteigend nach der Sozialversicherungsnummer.

```
SELECT * FROM Mitarbeiter
WHERE mnr IN (SELECT mnr FROM Vertreter)
```

```
AND mnr NOT IN (SELECT mnr FROM gekuendigt)
AND nname LIKE 'Bau%'
ORDER BY svnr DESC;
```

```
SELECT Mitarbeiter.*
FROM Mitarbeiter JOIN Vertreter ON Vertreter.mnr = Mitarbeiter.mnr
WHERE Mitarbeiter.mnr NOT IN (SELECT mnr
                             FROM gekuendigt)
AND nname LIKE 'Bau%'
ORDER BY svnr DESC;
```

```
SELECT Mitarbeiter.*
FROM Mitarbeiter JOIN Vertreter ON Vertreter.mnr = Mitarbeiter.mnr
WHERE NOT EXISTS (SELECT mnr
                 FROM gekuendigt
                 WHERE gekuendigt.mnr = Mitarbeiter.mnr)
AND nname LIKE 'Bau%'
ORDER BY svnr DESC;
```

---

2)

Geben Sie Kundennummer, Vor- und Nachname aller Kunden aus, die vor dem 01-01-1965 geboren wurden und noch nie eine Schadensmeldung eingereicht haben. Verwenden Sie dazu die Funktion TO\_DATE und folgendes Datumsformat: DD-MM-YYYY. Ordnen Sie die Ergebnisse aufsteigend nach dem Geburtsdatum.

```
SELECT knr, vname, nname
FROM Kunde WHERE gebdat < TO_DATE('01-01-1965', 'DD-MM-YYYY')
AND knr NOT IN (SELECT kunde
               FROM Schadensmeldung)
ORDER BY GebDat ASC;
```

```
SELECT knr, vname, nname
FROM Kunde
WHERE gebdat < TO_DATE('01-01-1965', 'DD-MM-YYYY')
AND NOT EXISTS (SELECT kunde
               FROM Schadensmeldung
               WHERE kunde = knr)
ORDER BY GebDat ASC;
```

---

3.1)

Geben Sie die Anzahl aller (unterschiedlichen) Themen aus.

```
SELECT COUNT(*) as anzahl
FROM Thema;
```

---

3.2)

Geben Sie die Mitarbeiternummer jener Kundenbetreuer aus, die Themen betreuen, sowie die Anzahl aller unterschiedlichen Themen, die sie betreuen.

```
SELECT mnr, COUNT(DISTINCT thema) as anzahl
FROM Kundenbetreuer JOIN betreut ON betreut.betreuer = Kundenbetreuer.mnr
GROUP BY Kundenbetreuer.mnr;
```

---

3.3)

Geben Sie die Mitarbeiternummer, Vor- und Nachnamen jener Kundenbetreuer aus, die Themen betreuen, sowie die Anzahl aller unterschiedlichen Themen, die sie betreuen.

```
SELECT Mitarbeiter.mnr, vname, nname, COUNT(DISTINCT thema) as anzahl
FROM Kundenbetreuer JOIN Mitarbeiter ON Mitarbeiter.mnr = Kundenbetreuer.mnr
JOIN betreut ON betreut.betreuer = Kundenbetreuer.mnr
GROUP BY Mitarbeiter.mnr, vname, nname;
```

```
SELECT mnr, vname, nname, COUNT(DISTINCT thema) as anzahl
FROM Kundenbetreuer NATURAL JOIN Mitarbeiter JOIN betreut ON betreut.betr
euer = mnr
GROUP BY mnr, vname, nname;
```

---

3.4)

Geben Sie die Mitarbeiternummer, Vor- und Nachnamen der Mitarbeiter aus, die am meisten

unterschiedliche Themen betreuen.

```
SELECT mnr, vname, nname
FROM Kundenbetreuer NATURAL JOIN Mitarbeiter JOIN betreut ON betreut.betr
euer = mnr
GROUP BY mnr, vname, nname
HAVING COUNT(DISTINCT thema) = (SELECT MAX(themen)
FROM (SELECT COUNT(DISTINCT thema) themen
FROM Kundenbetreuer JOIN betreut ON betreut.betreuer =
Kundenbetreuer.mnr
GROUP BY Kundenbetreuer.mnr));
```

```
SELECT mnr, vname, nname
FROM Kundenbetreuer NATURAL JOIN Mitarbeiter JOIN betreut ON betreut.betr
euer = mnr
GROUP BY mnr, vname, nname
HAVING COUNT(DISTINCT thema) >= ALL(SELECT COUNT(DISTINCT thema)
FROM Kundenbetreuer JOIN betreut ON betreut.betreuer =
Kundenbetreuer.mnr
GROUP BY Kundenbetreuer.mnr);
```

---

4.1)

Geben Sie Kundennummer und Nachname der ältesten Kunden aus.

```
SELECT knr, nname
FROM Kunde
WHERE gebdat <= ALL(SELECT gebdat
FROM Kunde);
```

```
SELECT knr, nname
FROM Kunde
WHERE gebdat = (SELECT MIN(gebdat)
FROM Kunde);
```

---

4.2)

Geben Sie Kundennummer und Nachname der ältesten Kunden aus, die zu mindestens einem Thema betreut

werden.

```
SELECT DISTINCT knr, nname
FROM Kunde JOIN betreut ON betreut.kunde = Kunde.knr
WHERE gebdat <= ALL(SELECT gebdat
                    FROM Kunde JOIN betreut ON betreut.kunde = Kunde.knr);
```

```
SELECT DISTINCT knr, nname
FROM Kunde JOIN betreut ON betreut.kunde = Kunde.knr
WHERE gebdat = (SELECT MIN(gebdat)
               FROM Kunde JOIN betreut ON betreut.kunde = Kunde.knr);
```

```
SELECT DISTINCT knr, nname
FROM Kunde
WHERE knr IN (SELECT kunde FROM betreut)
AND gebdat <= ALL(SELECT gebdat
                  FROM Kunde
                  WHERE knr IN (SELECT kunde
                              FROM betreut));
```

```
SELECT DISTINCT knr, nname
FROM Kunde
WHERE knr IN (SELECT kunde FROM betreut)
AND gebdat = (SELECT MIN(gebdat)
              FROM Kunde
              WHERE knr IN (SELECT kunde FROM betreut));
```

---

5)

Geben Sie Mitarbeiternummer und Nachname aller Mitarbeiter aus. In einer zusätzlichen Spalte soll vermerkt werden, um welchen Typ von Mitarbeiter es sich handelt, also entweder 'Vertreter' oder 'Kundenbetreuer'. Sie können davon ausgehen, dass ein Mitarbeiter immer nur maximal einem Typ zugeordnet ist. Ist die betreffende Person weder Vertreter noch Kundenbetreuer, soll der Wert der Spalte 'Nicht zugeordnet' lauten.

```
SELECT mnr, nname, 'Kundenbetreuer'
FROM Kundenbetreuer NATURAL JOIN Mitarbeiter
UNION
```

```
SELECT mnr, nname, 'Vertreter'
FROM Vertreter NATURAL JOIN Mitarbeiter
UNION
SELECT mnr, nname, 'Nicht zugeordnet'
FROM Mitarbeiter
WHERE mnr NOT IN (SELECT mnr FROM Kundenbetreuer UNION SELECT mnr
FROM Vertreter);
```

---

6)

Geben Sie alle Kunden aus, die Produkte aus jeder Kategorie gekauft haben.

```
SELECT * FROM Kunde WHERE NOT EXISTS
(SELECT * FROM Kategorie WHERE katid NOT IN
(SELECT Produkt.katid FROM Auftrag JOIN Produkt ON Produkt.pid = Auftr
ag.pid WHERE Auftrag.knr = Kunde.knr));
```

```
SELECT * FROM Kunde
WHERE (SELECT COUNT(DISTINCT katid)
FROM Auftrag JOIN Produkt ON Produkt.pid = Auftrag.pid
WHERE Auftrag.knr = Kunde.knr) = (SELECT COUNT(*)
FROM Kategorie);
```

---

7.1)

Geben Sie Information, Datum und Betrag aller Schadensmeldungen aus, die der Kunde mit dem Nachnamen 'Steinreich' eingereicht hat.

```
SELECT info, datum, betrag
FROM Schadensmeldung JOIN Kunde ON Kunde.knr = Schadensmeldung.kunde
WHERE nname = 'Steinreich';
```

---

7.2)

Geben Sie die Kundennummer, Nachname und Gesamtschadenssumme aller Schadensmeldungen jedes Kunden aus. Wenn kein Schadensfall vorliegt, soll die Schadenssumme 0 betragen. (Verwenden Sie dazu die Funktion COALESCE.)

g.kunde  
SELECT Kunde.knr, nname, SUM(COALESCE(betrag, 0)) as summe  
FROM Schadensmeldung RIGHT JOIN Kunde ON Kunde.knr = Schadensmeldung.knr  
GROUP BY Kunde.knr, nname;

```
SELECT Kunde.knr, nname, COALESCE((SELECT SUM(betrag)
FROM Schadensmeldung
WHERE kunde = knr), 0) as summe
FROM Kunde;
```

---

7.3)  
Geben Sie eine Liste ALLER Kunden aus zusammen mit der Schadenssumme, die für Autounfälle eingereicht wurden.

```
SELECT Kunde.knr, nname, SUM(COALESCE(betrag, 0)) as summe
FROM Schadensmeldung RIGHT JOIN Kunde ON Kunde.knr = Schadensmeldung.knr
AND info = 'Autounfall'
GROUP BY Kunde.knr, nname;
```

```
SELECT Kunde.knr, nname, COALESCE((SELECT SUM(betrag)
FROM Schadensmeldung
WHERE kunde = knr
AND info = 'Autounfall'), 0) as summe
FROM Kunde;
```

---

8)  
Geben Sie für Kunden die Kundennummer und die Gesamtkosten für alle gekauften Versicherungsprodukte aus. Es sollen nur die Kunden aufscheinen, die durchschnittlich weniger als 250 Geldeinheiten Umsatz mit ihren gekauften Produkten gemacht haben. Beachten Sie in diesem Beispiel den gewährten Rabatt. (Doppelkäufe werden hierbei ebenfalls gezählt. Wurden keine Produkte gekauft, sollen die Gesamtkosten 0 betragen.)

```
SELECT Kunde.knr, SUM(COALESCE(preis * ((100 - rabatt) / 100), 0)) as koste
FROM Kunde LEFT JOIN Auftrag ON Auftrag.knr = Kunde.knr LEFT JOIN Produk
```

```
t ON Produkt.pid = Auftrag.pid
GROUP BY Kunde.knr
HAVING AVG(COALESCE(preis * ((100 - rabatt) / 100), 0)) < 250;
```

```
SELECT knr, COALESCE((SELECT SUM(preis * ((100 - rabatt) / 100))
FROM Auftrag NATURAL JOIN Produkt
WHERE Auftrag.knr = Kunde.knr), 0) as kosten
FROM Kunde
WHERE COALESCE((SELECT AVG(preis * ((100 - rabatt) / 100))
FROM Auftrag NATURAL JOIN Produkt
WHERE Auftrag.knr = Kunde.knr), 0) < 250;
```

---

9)

Finden Sie die am häufigsten verwendeten Nachnamen aller Mitarbeiter aus.

```
SELECT nname
FROM Mitarbeiter
GROUP BY nname
HAVING COUNT(nname) >= ALL(SELECT COUNT(nname)
FROM Mitarbeiter
GROUP BY nname);
```

```
SELECT nname
FROM Mitarbeiter
GROUP BY nname
HAVING COUNT(nname) = (SELECT MAX(anzahl)
FROM (SELECT COUNT(nname) anzahl
FROM Mitarbeiter
GROUP BY nname));
```

---

10)

Geben Sie die Mitarbeiternummer aller nicht gekündigten Vertreter aus, die schon mehr als 20% aller Produkte vertrieben haben.

```
SELECT mnr
FROM Vertreter
```



```
WHERE mnr NOT IN (SELECT mnr FROM gekuendigt)
AND (SELECT COUNT(DISTINCT produkt)
     FROM vertreibt
     WHERE vertreter = mnr) > (SELECT COUNT(pid)/5
                              FROM produkt);
```