

Aufgabe 1

```
SELECT COUNT(kid) AS "Anzahl"  
FROM kunde  
WHERE ort = "Stans";
```

Aufgabe 2

```
SELECT COUNT(kid) AS "Anzahl"  
FROM kunde  
WHERE vorname LIKE "A%";
```

Aufgabe 3

```
SELECT COUNT(vid) AS "Anzahl"  
FROM verkauf  
WHERE datum = "2011-07-29";
```

Aufgabe 4

```
SELECT COUNT(vid) AS "Anzahl Verkäufe August"  
FROM verkauf  
WHERE datum >= "2011-08-01" AND datum <= "2011-08-31";
```

oder

```
SELECT COUNT(vid) AS "Anzahl Verkäufe August"  
FROM verkauf  
WHERE datum LIKE "2011-08-__";
```

oder

```
SELECT COUNT(vid) AS "Anzahl Verkäufe August"  
FROM verkauf  
WHERE datum BETWEEN "2011-08-01" AND "2011-08-31";
```

Aufgabe 5

```
SELECT COUNT(*) AS "Anzahl Verkäufe"  
FROM verkauf  
WHERE datum = "2011-06-08"  
      AND zeit >= "08:00:00"  
      AND zeit <= "11:00:00";
```

oder

```
SELECT count(vid) AS "Anzahl Verkäufe"
FROM verkauf
WHERE datum = "2011-06-08"
      AND zeit BETWEEN "08:00:00" AND "11:00:00";
```

(Die Angabe der Sekunden kann auch weggelassen werden.)

Aufgabe 6

```
SELECT bezeichnung, preis
FROM artikel
ORDER BY preis DESC;
```

Aufgabe 7

```
SELECT bezeichnung, lieferant.name
FROM artikel JOIN lieferant
      ON artikel.lid = lieferant.lid
ORDER BY bezeichnung;
```

kürzer aber „gefährlicher“:

```
SELECT bezeichnung, lieferant.name
FROM artikel NATURAL JOIN lieferant
ORDER BY bezeichnung;
```

Aufgabe 8

```
SELECT lieferant.name, count(artikel.aid)
      AS "Anzahl Artikel"
FROM lieferant JOIN artikel
      ON lieferant.lid = artikel.lid
GROUP BY lieferant.lid;
```

oder kürzer aber „gefährlicher“:

```
SELECT lieferant.name, count(artikel.aid)
      AS "Anzahl Artikel"
FROM lieferant JOIN artikel
      ON lieferant.lid = artikel.lid
GROUP BY lieferant.lid;
```

Aufgabe 9

```
SELECT bezeichnung, SUM(menge), einheit
FROM artikel JOIN verkauf
      ON artikel.aid = verkauf.aid
WHERE artikel.bezeichnung = "Schokolade";
```

Aufgabe 10

```
SELECT SUM(preis) AS 'Umsatz'  
FROM kunde JOIN verkauf JOIN artikel  
  ON kunde.kid = verkauf.kid  
  AND verkauf.aid = artikel.aid  
WHERE kunde.nachname = 'Fischer'  
  AND kunde.vorname = 'Nicole';
```

kürzer aber „gefährlich“:

```
SELECT SUM(preis) AS 'Umsatz'  
FROM kunde NATURAL JOIN verkauf NATURAL JOIN artikel  
WHERE kunde.nachname = 'Fischer' AND kunde.vorname = 'Nicole';
```

Aufgabe 11

```
SELECT kunde.nachname, kunde.vorname,  
  sum(artikel.preis) AS 'Umsatz'  
FROM artikel JOIN verkauf JOIN kunde  
  ON artikel.aid = verkauf.aid  
  AND verkauf.kid = kunde.kid  
GROUP BY kunde.kid  
ORDER BY Umsatz ASC;
```

Aufgabe 12

```
SELECT kunde.nachname, kunde.vorname,  
  sum(preis) AS "Umsatz"  
FROM kunde JOIN verkauf JOIN artikel  
  ON kunde.kid = verkauf.kid  
  AND verkauf.aid = artikel.aid  
GROUP BY kunde.kid  
HAVING sum(preis) > 80  
ORDER BY "Umsatz" DESC;
```

Aufgabe 13

```
SELECT bezeichnung, preis  
FROM artikel  
WHERE preis > (  
  SELECT avg(preis)  
  FROM artikel  
)  
ORDER BY preis DESC;
```

Aufgabe 14

```
SELECT bezeichnung, max("sales") AS "Umsatz"
FROM (
  SELECT bezeichnung, sum(artikel.preis)
  AS sales
  FROM verkauf JOIN artikel
  ON verkauf.aid = artikel.aid
  GROUP BY artikel.aid
);
```

Aufgabe 15

```
SELECT nachname, vorname, max(sales)
AS Umsatz
FROM (
  SELECT nachname, vorname,
  SUM(artikel.preis) AS sales
  FROM kunde JOIN verkauf JOIN artikel
  ON kunde.kid = verkauf.kid
  AND verkauf.aid = artikel.aid
  GROUP BY kunde.kid
);
```

Aufgabe 16

```
SELECT "Monat", "Umsatz"
FROM (
  SELECT "Juni" AS "Monat", sum(preis) AS "Umsatz"
  FROM verkauf NATURAL JOIN artikel
  WHERE verkauf.datum LIKE "____-06-__"
  UNION
  SELECT "Juli" AS "Monat", sum(preis) AS "Umsatz"
  FROM verkauf NATURAL JOIN artikel
  WHERE verkauf.datum LIKE "____-07-__"
  UNION
  SELECT "August" AS "Monat", sum(preis) AS "Umsatz"
  FROM verkauf NATURAL JOIN artikel
  WHERE verkauf.datum LIKE "____-08-__"
)
ORDER BY "Umsatz" DESC;
```

(Vermutlich gibt es noch elegantere Lösungen.)