

Hinweis: Die Lösungen entsprechen der SQL-Norm. Bitte beachten Sie Abweichungen in Oracle, SQL-Server und MySQL, auf die in Kapitel 5 ausführlich eingegangen wird.

Lösung zu Aufgabe 1

Relationen *Auftrag*, *Auftragsposten*, *Teilestruktur*, *Lieferung*: siehe Anhang in Kapitel 10.

```
CREATE TABLE Artikel
( ANr          INT          PRIMARY KEY,
  Bezeichnung  CHAR (35)    NOT NULL,
  Nettopreis   NUMERIC(7,2)  CHECK ( Nettopreis > 0 ),
  Steuer       NUMERIC(7,2)  CHECK ( Steuer > 0 ),
  Preis        NUMERIC(7,2)  CHECK ( Preis > 0 ),
  Farbe        CHAR (10),
  Mass         CHAR (15),
  Einheit      CHAR (2)      NOT NULL,
  Typ          CHAR (1)      NOT NULL CHECK ( Typ IN ('E','Z','F') )
);

CREATE TABLE Auftrag
( AuftrNr      INT          PRIMARY KEY,
  Datum        DATE,
  Kundnr       INT          NOT NULL REFERENCES Kunde
                                     ON DELETE NO ACTION ON UPDATE CASCADE,
  Persnr       INT          REFERENCES Personal
                                     ON DELETE SET NULL ON UPDATE CASCADE
);

CREATE TABLE Auftragsposten
( PosNr        INT          PRIMARY KEY,
  AuftrNr      INT          NOT NULL REFERENCES Auftrag
                                     ON DELETE CASCADE ON UPDATE CASCADE,
  ArtNr        INT          NOT NULL REFERENCES Artikel
                                     ON DELETE NO ACTION ON UPDATE CASCADE,
  Anzahl       SMALLINT,
  Gesamtpreis NUMERIC(10,2) CHECK (Gesamtpreis > 0),
  UNIQUE (AuftrNr, ArtNr)
);

CREATE TABLE Lieferung
( ANr          INT          REFERENCES Artikel
                                     ON DELETE CASCADE ON UPDATE CASCADE,
  LiefNr       INT          REFERENCES Lieferant
                                     ON DELETE CASCADE ON UPDATE CASCADE,
  Lieferzeit   SMALLINT,
  Nettopreis   NUMERIC(7,2),
  Bestellt     SMALLINT,
  PRIMARY KEY (ANr, LiefNr)
);

CREATE TABLE Teilestruktur
( ArtNr        INT          REFERENCES Artikel
                                     ON DELETE CASCADE ON UPDATE CASCADE,
  Einzelteilnr INT          REFERENCES Artikel
                                     ON DELETE NO ACTION ON UPDATE CASCADE,
  Anzahl       INT,
  Einheit      CHAR (2),
  PRIMARY KEY (ANr, Einzelteilnr)
);

CREATE TABLE Lieferant
( Nr          INT          PRIMARY KEY,
  Name        CHAR (30)    NOT NULL,
  Strasse     CHAR (30),
  PLZ         CHAR (5),
  Ort         CHAR (25),
  Sperre      CHAR ) ;
```

Analog wird die Relation *Kunde* erzeugt.

```
CREATE TABLE Personal
( PersNr          INT          PRIMARY KEY,
  Name            CHAR (30)    NOT NULL,
  Strasse         CHAR (30),
  PLZ             CHAR (5),
  Ort             CHAR (20),
  GebDatum        DATE        NOT NULL,
  Stand           CHAR (6),
  Vorgesetzt      INT          REFERENCES Personal
                                ON DELETE SET NULL ON UPDATE CASCADE,
  Gehalt          NUMERIC (10, 2),
  Beurteilung     CHAR,
  Aufgabe        CHAR (18)
);

CREATE TABLE Lager
( Artnr          INT          PRIMARY KEY REFERENCES Artikel
                                ON DELETE CASCADE ON UPDATE CASCADE,

  Lagerort        CHAR (6),
  Bestand         SMALLINT NOT NULL,
  Mindbest        SMALLINT,
  Reserviert      SMALLINT,
  Bestellt        SMALLINT
);

CREATE TABLE Reservierung
( Posnr  INT          REFERENCES Auftrag
                                ON DELETE NO ACTION ON UPDATE CASCADE,
  Artnr  INT          REFERENCES Artikel
                                ON DELETE NO ACTION ON UPDATE CASCADE,

  Anzahl SMALLINT,
  PRIMARY KEY (Posnr, Artnr)
);
```

Lösung zu Aufgabe 2

```
ALTER TABLE Auftragsposten
ADD COLUMN Einzelpreis Numeric (10,2) ;

UPDATE Auftragsposten
SET   Einzelpreis = Gesamtpreis / Anzahl
WHERE Anzahl IS NOT NULL
AND   Anzahl <> 0;
```

Die neue Relation ist in 2. NF.

Lösung zu Aufgabe 3

```
CREATE VIEW VPers AS
SELECT Persnr, Name, Strasse, PLZ, Ort, GebDatum, Stand, Vorgesetzt, Aufgabe
FROM   Personal
WHERE  Vorgesetzt IS NOT NULL ;
```

Die Sicht ist änderbar.

Lösung zu Aufgabe 4

```
CREATE VIEW VAuftragsPosten( PosNr, AuftrNr, Artnr, Anzahl,
                             Einzelpreis, Gesamtpreis) AS
SELECT PosNr, AuftrNr, Artnr, Anzahl, Gesamtpreis/Anzahl, Gesamtpreis
FROM   Auftragsposten ;
```

Die Sicht ist nicht änderbar.

Lösung zu Aufgabe 5

Das Hinzufügen von Schemaelementen zu einem Schema erfolgt einfach mittels Qualifizierung. Das Hinzufügen einer Sicht namens *VPers* zum Schema *Verkauf* geschieht beispielsweise wie folgt:

```
CREATE VIEW Verkauf.VPers AS
SELECT ...
```

Lösung zu Aufgabe 6

```
CREATE TRIGGER Artikelpreis_Trigger
BEFORE INSERT OR UPDATE
ON Artikel
REFERENCING NEW AS neu
FOR EACH ROW
BEGIN ATOMIC
    neu.Steuer = neu.Netto * 0.19;
    neu.Preis  = neu.Netto * 1.19;
END ;
```

Lösung zu Aufgabe 7

```
CREATE SEQUENCE Kundnr_Sequenz
START WITH 21
INCREMENT BY 2;
INSERT INTO Kunde (Nr, Name)
VALUES (NEXT VALUE FOR Kundnr_Sequenz, 'Dummy');
```

Lösung zu Aufgabe 8

```
CREATE SEQUENCE Persnr_Sequenz
START WITH 10
INCREMENT BY 1;
CREATE TRIGGER Persnr_Trigger
BEFORE INSERT
ON Personal
REFERENCES NEW AS neu
FOR EACH ROW
BEGIN ATOMIC
    neu.Persnr = NEXT VALUE FOR Persnr_Sequenz;
END ;
```

Lösung zu Aufgabe 9

Diese Aussage stimmt im Prinzip. Ein DBMS nimmt aber in der Regel Optimierung vor und hält die wichtigsten Daten im Cache. Damit gibt es meist doch nur einen Zugriff auf die Festplatte.

Lösung zu Aufgabe 10

```
GRANT      Select, Update (Bestand, Reserviert, Bestellt)
ON TABLE Lager
TO         Gast ;
```

Lösung zu Aufgabe 11

```
REVOKE      Select, Update
ON TABLE Lager
FROM        Gast ;
```

Lösung zu Aufgabe 12

```
CREATE VIEW VLager AS
SELECT Artnr, Lagerort, Bestand
FROM Lager
WHERE Bestand >= Mindbest + Reserviert ;
REVOKE      ALL PRIVILEGES
ON TABLE Lager
FROM        Gast ;
GRANT       Select
ON TABLE VLager
TO          Gast
WITH GRANT OPTION ;
```

Lösung zu Aufgabe 13

```
ALTER TABLE Personal
ADD CONSTRAINT PersIntegr CHECK ( GebDatum BETWEEN DATE '1970-01-01'
                                AND DATE '2010-12-31'
                                AND Stand IN ('led','verh','ges','verw'))
```

```
AND Gehalt BETWEEN 500 AND 6000  
AND Beurteilung BETWEEN 1 AND 10 ) ;
```