

Aufgabe 13.27 SQL

a) Während bei CHARACTER(20) die Länge auf 20 festgelegt ist, unabhängig davon, ob sie benötigt wird, kann bei CHARACTER VARYING(20) die tatsächliche Länge auch geringer sein. 20 ist hier nur eine Obergrenze.

```
b)1 CREATE TABLE Belegung (
2     Matrikel DECIMAL(7)REFERENCES Student(Matrikel)
3     ON DELETE SET NULL
4     ON UPDATE CASCADE,
5     VBez CHARACTER(9)
6     ON DELETE SET NULL
7     ON UPDATE CASCADE,
8     VSemester CHARACTER(6)
9     ON DELETE SET NULL
10    ON UPDATE CASCADE,
11    Punkte DECIMAL(3),
12    FOREIGN KEY (VBez, VSemester) REFERENCES Veranstaltung(VBez, VSemester)
13 );
```

```
c)1 INSERT INTO Student (Matrikel,SName, SVorname)
2     VALUES ('1234567','Clever','Claus');
3 INSERT INTO Professor (PNr,PName,PVorname)
4     VALUES ('1','Theel','Oliver');
5 INSERT INTO Mitarbeiter (MNr,MName,MVorname,PNr)
6     VALUES ('1','Hahn','Philipp','1');
7 INSERT INTO Veranstaltung (VBez,VSemester,VTitel,VArt)
8     VALUES ('2.1.005','WS2003','Praktische Informatik','V3U1');
9 INSERT INTO Belegung (Matrikel,VBez,VSemester,Punkte)
10    VALUES ('1234567','2.1.005','WS2003',100);
```

```
d)1 UPDATE Student SET Matrikel = '7654321'
2     WHERE SName = 'Clever' AND SVorname = 'Claus';
```

Der Name ist nicht zwangsläufig eindeutig. Es werden so die Matrikelnummern aller Studenten mit diesem Namen geändert. Zudem könnte die neue Matrikelnummer bereits vergeben sein. In diesem Fall würde die Änderung hier abgelehnt werden.

```
e)1 UPDATE Student SET SSemester = SSemester + 1;
```

```
f)1 SELECT PName, PVorname, Abteilung FROM Professor
2     ORDER BY Abteilung;
```

```
g)1 SELECT Mitarbeiter.MName, Mitarbeiter.MVorname, Professor.Abteilung FROM Mitarbeiter, Professor
2     WHERE Mitarbeiter.PNr = Professor.PNr
3     GROUP BY Abteilung;
```

```
h)1 SELECT Matrikel FROM Belegung
2     WHERE VSemester = 'SS2003' AND Punkte > 85;
```

```
i)1 SELECT SAdresse AS Adresse FROM Student
2     UNION SELECT PAdresse AS Adresse FROM Professor
3     UNION SELECT MAdresse AS Adresse FROM Mitarbeiter;
```

```
j)1 SELECT Mitarbeiter.MNr, Mitarbeiter.MName, Mitarbeiter.MVorname
2     FROM Mitarbeiter, Veranstaltung, Belegung, Student
3     WHERE Mitarbeiter.MNr = Veranstaltung.MNr
4     AND Belegung.VBez = Veranstaltung.VBez
5     AND Belegung.VSemester = Veranstaltung.VSemester
6     AND Belegung.Matrikel = Student.Matrikel
7     AND Student.SName = 'M"uller'
8     AND Student.SVorname = 'Claudia';
```

```
k)1 SELECT Student.Matrikel, Student.SName, Student.SVorname
2     FROM Student, Belegung AS Belegung1, Belegung AS Belegung2
3     WHERE Student.Matrikel = Belegung1.Matrikel
```

```

4          AND Student.Matrikel = Belegung2.Matrikel
5          AND Belegung1.Punkte < 40
6          AND Belegung2.Punkte >= 40;

l)1  SELECT Veranstaltung1.VTitel
2     FROM Veranstaltung AS Veranstaltung1, Veranstaltung AS Veranstaltung2
3     WHERE Veranstaltung1.VBez = Veranstaltung2.VBez
4     AND Veranstaltung1.PNr <> Veranstaltung2.PNr;

m)1  SELECT MAX(SSemester) AS MaxSSemester FROM Student;

n)1  SELECT VBez, COUNT(VSemester) AS CountVSemester FROM Veranstaltung
2     GROUP BY VBez
3     HAVING COUNT(VSemester) > 1;

o)1  SELECT VSemester, MAX(Punkte) AS AVGPunkte FROM Belegung
2     WHERE Matrikel = '1111111'
3     GROUP BY VSemester;

p)1  SELECT Matrikel
2     FROM Belegung
3     WHERE VSemester = 'SS2003'
4     GROUP BY Matrikel
5     HAVING AVG(Punkte) > 75;

```

Aufgabe 13.28 HDLC/LAP B

- a) Bei den zu übertragenden Daten wird vom Sender bei jeder Folge von 5 Einsen eine Null angehängt. Der Empfänger entfernt jede Null, welche direkt auf fünf Einsen folgt. Die Bitfolge 01111110 kann somit in den Übertragenen Daten nicht mehr auftreten.
- b) – Der Sendefolgenzähler $V(S)$ gibt die Nummer an, mit welcher der nächste Datenblock gesendet wird.
 – Der Folgennummerzähler $VO(S)$ gibt die Nummer des letzten von der Gegenstation empfangen Blocks an.
 – Der Empfängerfolgezähler $V(R)$ gibt die Nummer des als nächstes erwarteten Blocks an.
 (aus: Kowalk, W., „Rechnernetze“,
<http://einstein.kowalk.informatik.uni-oldenburg.de/rechnernetze/seite61.htm>)
- c) $V(R)$ und $V(S)$ verändern ihre Werte durch fortlaufendes zählen. $VO(S)$ durch das Kopieren des Wertes $N(R)$ des letzten empfangen Blocks.
- d) Die momentane Fenstergröße ergibt sich aus dem Abstand von $V(S)$ und $VO(S)$. Sie lässt sich errechnen durch $V(S) - VO(S)$, falls $V(S) > VO(S)$ und $(V(S) + 8) - VO(S)$ sonst.
 Die Nummer des zuletzt empfangenen Pakets lässt sich errechnen durch $V(R) - 1$.
 (siehe auch: Kowalk, W., „Rechnernetze“,
<http://einstein.kowalk.informatik.uni-oldenburg.de/rechnernetze/seite61.htm>)