

Aufgabensammlung SQL – SW4

1. Einfache Anfragen

Buch: Kapitel 4.6 und 4.7.

Datenbank: Die folgenden Anfragen beziehen sich auf die Universitätsdatenbank des Buches. Alle Umlaute werden umschrieben (hören → hoeren)

Aufgabe 1: Geben Sie den Raum aus von Prof. Curie.

```
SELECT Raum
FROM Professoren
WHERE Name = 'Curie'
```

```
Raum
-----
36
```

(1 row(s) affected)

Aufgabe 2: Geben Sie die Nummern aller Vorlesungen aus, die der Student mit der Matrikelnummer 29120 hört.

```
SELECT VorlNr
FROM hoeren
WHERE MatrNr = 29120
```

```
VorlNr
-----
5001
5041
5049
```

(3 row(s) affected)

Aufgabe 3: Geben Sie die Nummer und den Titel aller Vorlesungen aus. Sortieren Sie diese aufsteigend nach dem Titel.

```
SELECT VorlNr, Titel
FROM Vorlesungen
ORDER BY Titel
```

```
VorlNr      Titel
-----
5216        Bioethik
5259        Der Wiener Kreis
4630        Die 3 Kritiken
5043        Erkenntnistheorie
5041        Ethik
5022        Glaube und Wissen
5001        Grundzuege
4052        Logik
5049        Maeeutik
5052        Wissenschaftstheorie
```

(10 row(s) affected)

Aufgabe 4: Geben Sie den Titel und PersNr des lesenden Professors aller Vorlesungen aus. Sortieren Sie diese aufsteigend nach der PersNr. Vorlesungen, die vom gleichen Professor gelesen werden, sollen absteigend nach dem Titel sortiert ausgegeben werden.

```
SELECT Titel, PersNr
FROM Vorlesungen AS v
INNER JOIN Professoren AS p ON p.PersNr = v.gelesenVon
ORDER BY PersNr, Titel DESC
```

Titel	PersNr
Maeeutik	2125
Logik	2125
Ethik	2125
Wissenschaftstheorie	2126
Erkenntnistheorie	2126
Bioethik	2126
Der Wiener Kreis	2133
Glaube und Wissen	2134
Grundzuege	2137
Die 3 Kritiken	2137

(10 row(s) affected)

Aufgabe 5: Geben Sie die Namen der Assistenten und den Namen ihrer Bosse aus (aufsteigend nach den Namen der Assistenten sortiert).

```
SELECT a.Name AS AssistentenName, p.Name AS ProfessorName
FROM Assistenten AS a
INNER JOIN Professoren AS p ON p.PersNr = a.Boss
ORDER BY a.Name
```

AssistentenName	ProfessorName
Aristoteles	Sokrates
Newton	Kopernikus
Platon	Sokrates
Rhetikus	Kopernikus
Spinoza	Augustinus
Wittgenstein	Russel

(6 row(s) affected)

Aufgabe 6: Geben Sie die Namen der Professoren aus, die eine Vorlesung lesen, die vom Studenten „Carnap“ besucht wird. Geben Sie jeden Professor nur einmal aus.

```
SELECT DISTINCT p.Name AS ProfessorName
FROM Professoren AS p
INNER JOIN Vorlesungen AS v ON v.gelesenVon = p.PersNr
INNER JOIN hoeren AS h ON h.Vor1Nr = v.Vor1Nr
WHERE h.MatrNr = (SELECT MatrNr
                  FROM Studenten
                  WHERE Name = 'Carnap')
```

ProfessorName
Popper
Russel
Sokrates

(3 row(s) affected)

Aufgabe 7: Geben Sie die Matrikelnummern aller Studierenden aus, die mindestens eine Vorlesung besuchen. Dabei soll jede Matrikelnummer nur einmal ausgegeben werden.

```
SELECT DISTINCT MatrNr
FROM hoeren
```

MatrNr

25403

26120

27550

28106

29120

29555

(6 row(s) affected)

Aufgabe 8: Geben Sie die Namen aller Studierenden aus, die mindestens eine Vorlesung hören. Dabei soll jeder Name nur einmal ausgegeben werden.

```
SELECT DISTINCT s.Name
FROM Studenten AS s
INNER JOIN hoeren as h ON h.MatrNr = s.MatrNr
```

Name

Carnap

Feuerbach

Fichte

Jonas

Schopenhauer

Theophrastos

(6 row(s) affected)

Aufgabe 9: Geben Sie die Namen der Studenten aus, welche die Vorlesung „Bioethik“ hören.

```
SELECT DISTINCT s.Name
FROM Studenten AS s
INNER JOIN hoeren as h ON h.MatrNr = s.MatrNr
INNER JOIN Vorlesungen as v ON v.VorlNr = h.VorlNr
WHERE v.Titel = 'Bioethik'
```

Name

Carnap

(1 row(s) affected)

Aufgabe 10: Geben Sie die Namen aller Studenten aus, die eine Vorlesung bei „Russel“ hören.

```
SELECT DISTINCT s.Name
FROM Studenten AS s
INNER JOIN hoeren AS h ON h.MatrNr = s.MatrNr
INNER JOIN Vorlesungen AS v ON v.VorlNr = h.VorlNr
INNER JOIN Professoren AS p ON p.PersNr = v.gelesenVon
WHERE p.Name = 'Russel'
```

Name

Carnap

(1 row(s) affected)

Aufgabe 11: Geben Sie die Namen der Professoren zusammen mit den Namen der Studenten aus, die eine Vorlesung bei ihnen hören.

```
SELECT p.Name AS ProfessorName, s.Name AS StudentName
FROM Professoren AS p
  INNER JOIN Vorlesungen AS v ON v.gelesenVon = p.PersNr
  INNER JOIN hoeren AS h ON h.VorlNr = v.VorlNr
  INNER JOIN Studenten AS s ON s.MatrNr = h.MatrNr
ORDER BY p.Name, s.Name
```

ProfessorName	StudentName
Augustinus	Feuerbach
Augustinus	Jonas
Kant	Feuerbach
Kant	Fichte
Kant	Schopenhauer
Kant	Theophrastos
Popper	Carnap
Russel	Carnap
Russel	Carnap
Sokrates	Carnap
Sokrates	Schopenhauer
Sokrates	Theophrastos
Sokrates	Theophrastos

(13 row(s) affected)

2. Weitergehende Anfragen

Buch: Kapitel 4.8 bis 4.9 und 4.12

Datenbank: Die Universitätsdatenbank [U] aus dem Buch und die Zug-Datenbank [Z].

Die Zug-Datenbank siehe Skript zugDb.sql bzw. zugDbPostgre.sql und Beschreibung unten.

Aufgabe 12: Z: Welche Bahnhöfe heissen so wie Städte?

```
USE zug
SELECT b.Name
FROM Bahnhof AS b
INNER JOIN Stadt AS s ON s.Name = b.Name;
```

Name

Baden

Horw

Luzern

Oltén

Thalwil

Zug

(6 row(s) affected)

Aufgabe 13: Z: Welche Bahnhöfe haben einen anderen Namen als Städte?

```
USE zug
SELECT b.Name
FROM Bahnhof AS b
WHERE b.Name NOT IN (SELECT Name FROM Stadt)
```

Name

Basel Bad

Basel SBB

Zuerich HB

(3 row(s) affected)

Aufgabe 14: U: Finden Sie die Professoren, die keine Assistierenden haben.

a) mit „exists“

```
USE uni
SELECT p.Name
FROM Professoren AS p
WHERE NOT EXISTS (SELECT *
                  FROM Assistenten AS a
                  WHERE a.Boss = p.PersNr)
```

Name

Popper

Curie

Kant

(3 row(s) affected)

b) mit dem Mengenvergleich

```
USE uni
SELECT p.Name
FROM Professoren AS p
WHERE p.PersNr NOT IN (SELECT Boss
                       FROM Assistenten)
```

Name

Popper

Curie

Kant

(3 row(s) affected)

Aufgabe 15: U: Finden Sie den Namen des Studierenden (der Studierenden, sofern es mehrere gibt), der die beste Prüfungsnote1 erreicht hat.

```
USE uni
SELECT s.Name
FROM Studenten AS s
INNER JOIN pruefen AS pr ON pr.MatrNr = s.MatrNr
WHERE pr.Note = 1
```

Name

Carnap

(1 row(s) affected)

Aufgabe 16: U: Finden Sie die Namen der Studierenden, die Prüfungen mit einer schlechteren Note als der Durchschnitt abgeschlossen haben.

```
USE uni
SELECT s.Name
FROM Studenten AS s
WHERE s.MatrNr IN (SELECT MatrNr
                  FROM pruefen
                  WHERE Note > (SELECT AVG(Note)
                               FROM pruefen)
                  )
```

Name

Jonas

Schopenhauer

(2 row(s) affected)

Aufgabe 17: U: Finden Sie die Namen der Professoren, die noch keine Prüfung abgenommen haben.

```
USE uni
SELECT Name
FROM Professoren
WHERE PersNr NOT IN (SELECT PersNr
                    FROM pruefen)
```

Name

Kopernikus

Popper

Augustinus

Curie

(4 row(s) affected)

Aufgabe 18: Z: Geben Sie die durchschnittliche Anzahl Gleise aus.

```
USE zug
SELECT AVG(Gleise) AS AvgGleise
FROM Bahnhof
```

```
AvgGleise
-----
14
```

(1 row(s) affected)

Aufgabe 19: Z: Geben Sie die Namen der Bahnhöfe aus, die weniger als die durchschnittliche Anzahl Gleise haben.

```
USE zug
SELECT Name
FROM Bahnhof
WHERE Gleise < (SELECT AVG(Gleise)
                FROM Bahnhof)
```

```
Name
-----
Basel Bad
Thalwil
Horw
Zug
Baden
```

(5 row(s) affected)

Aufgabe 20: Z: Mit welchen Zügen haben Sie in Zürich HB einen Anschluss, wenn Sie mit dem Zug 1 nach Zürich fahren? (Welche Züge fahren von Zürich HB nach Ankunft des Zuges 1 in Zürich ab).

```
USE zug
SELECT Zug, Abfahrt
FROM Zughalt
WHERE Halt = 9 -- 9 = Zürich HB
      AND Abfahrt > (SELECT Ankunft
                    FROM Zughalt
                    WHERE Halt = 9 -- 9 = Zürich HB
                    AND Zug = 1) -- 1 = Zug 1
```

```
Zug      Abfahrt
-----
2        13:15:00.0000000
7        13:25:00.0000000
```

(2 row(s) affected)

Aufgabe 21: Z: Finden Sie alle Züge (Nr. und Abfahrt) in der Relation "verbindet" (zusätzlich neu: zugHalt), die nach dem letzten Zug (Ankunft in Zürich) von Zürich HB abfahren. (Sie kommen mit dem letzten Zug, der eingetragen ist, in Zürich an. Mit welchem Zug (Nr. und Abfahrt) können Sie dann weiterfahren?)

```
USE zug
SELECT Zug, Abfahrt
FROM Zughalt
WHERE Halt = 9 -- 9 = Zürich HB
      AND Abfahrt > (SELECT MAX(Ankunft)
                    FROM Zughalt
                    WHERE Halt = 9
                    AND (Ankunft IS NOT NULL)) -- 9 = Zürich HB
```

```
Zug      Abfahrt
-----
```

(0 row(s) affected)

→ Der erste Zug am nächsten Tag..

Aufgabe 22: Z: Finden Sie die ZugNr der Züge, die max. einen Zwischenhalt zwischen Start- und Ziel-Bahnhof haben.

```
USE zug
SELECT Zug
FROM Zughalt
GROUP BY Zug
HAVING COUNT(Halt) <= 3
```

```
Zug
-----
3
4
5
6
```

(4 row(s) affected)

Aufgabe 23: Z: Geben Sie die ZugNr, den Start- und den Zielort an für alle Züge, die nicht in Olten halten.

```
SELECT z.Zug, b.Name
FROM Zughalt AS z
INNER JOIN Bahnhof AS b ON b.BahnhofId = z.Halt
WHERE Zug NOT IN (SELECT Zug FROM Zughalt WHERE Halt = (SELECT BahnhofId FROM Bahnhof
WHERE Name = 'Olten'))
AND (Ankunft IS NULL OR Abfahrt IS NULL)
```

```
Zug      Name
-----
1        Luzern
1        Zuerich HB
2        Luzern
2        Zuerich HB
3        Luzern
3        Horw
4        Luzern
4        Horw
7        Basel Bad
7        Zuerich HB
8        Basel Bad
8        Zuerich HB
```

(12 row(s) affected)

Aufgabe 24: Z: Geben Sie alle Züge (ZugNr, Abfahrtszeit und nächster Halt) aus, die zwischen 13:00 und 13:30 in Zürich losfahren.

Aufgabe 25: Z: Geben Sie alle Züge aus (ZugNr, Bahnhof, nächster Halt, Abfahrtszeit), die aus Basel losfahren.

3. Komplexere Anfragen

Buch: Kapitel 4.10, 4.11 und 4.13

Datenbanken: Universitätsdatenbank [U] und Züge-Datenbank [Z]

Aufgabe 26: U: Finden Sie die Studierenden, die alle Vorlesungen mit dem Begriff "Ethik" (like '%thik') im Titel hören. Im Kemper-Buch 4.10 Seite 125 Mitte ist ein entsprechendes Beispiel.

Aufgabe 27: U: Finden Sie die Studierenden (Name und Matrikelnummer), die alle Vorlesungen mit dem Begriff "Ethik" (like '%thik') im Titel hören, mit Hilfe von count. Im Kemper-Buch 4.10 Seite 125 unten ist ein entsprechendes Beispiel.

Aufgabe 28: U: Darauf basierend können Sie nun die Aufgabe 4.6 in Ihrem Buch beantworten. [Hinweis: Die Antwortmenge in der aktuellen Ausprägung der Unidatenbank ist leer.]

Aufgabe 29: U: Geben Sie den Namen der Studierenden und den Titel der von ihnen gehörten Vorlesungen aus. Verwenden Sie hierzu einmal den join-operator und einmal die entsprechende Einschränkung in der where Klausel.

Aufgabe 30: U: Geben Sie eine Liste aller Professoren (alle Attribute) aus. Sofern ein Professor einen Assistenten hat, soll dessen Name und Fachgebiet ebenfalls ausgegeben werden.

Aufgabe 31: U: Geben Sie eine Liste aller Studenten (MatrNr und Name) und aller Vorlesungen (VorlNr und Titel), die sie hören, aus. Sofern ein Student keine Vorlesung hört oder eine Vorlesung nicht besucht wird, sollen die entsprechenden Informationen des Studenten trotzdem ausgegeben werden.

Aufgabe 32: Z: Geben Sie alle Bahnhöfe (Name) sowie die Nummer und das Ziel der Züge, die in diesen Bahnhöfen starten, aus. Bahnhöfe, in denen keine Züge starten, sind trotzdem auszugeben.

Aufgabe 33: Z: Geben Sie alle Informationen der Tabelle Züge und alle „verbindet“ Informationen aus. Die Zuginformation soll immer bei der ersten Information der Relation „verbindet“ eines Zuges stehen (mit dem entsprechenden Start-Bahnhof).

Aufgabe 34: U: Geben Sie die Anfrage an, die in der aktuellen Datenbankausprägung folgendes Ergebnis erzeugt. (Es gibt syntaktisch zwei Möglichkeiten)

MatrNr	Name	Semester	VorlNr	Note
24002	Xenokrates	18		
25403	Jonas	12	5041	2
26120	Fichte	10		
26830	Aristoxenos	8		
27550	Schopenhauer	6	4630	2
28106	Carnap	3	5001	1
29120	Theophrastos	2		
29555	Feuerbach	2		

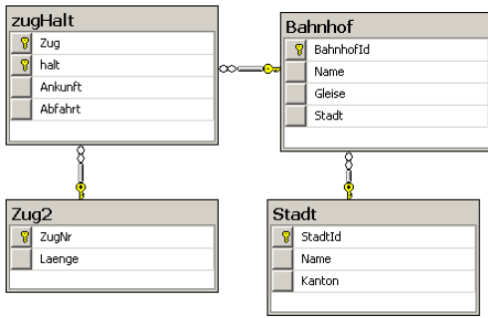
Aufgabe 35: U: Geben Sie die Anfrage an, die in der aktuellen Datenbankausprägung folgendes Ergebnis erzeugt.

VorNr	Titel	SWS	gelesenVon	Vorgänger	Nachfolger	VorNr	Titel	SWS	gelesenVon
						4052	Logik	4	2125
						4630	Die 3 Kritiken	4	2137
						5001	Grundzuege	4	2137
						5022	Glaube und Wissen	2	2134
5001	Grundzuege	4	2137	5001	5041	5041	Ethik	4	2125
5001	Grundzuege	4	2137	5001	5043	5043	Erkenntnistheorie	3	2126
5001	Grundzuege	4	2137	5001	5049	5049	Maeeutik	2	2125
5043	Erkenntnistheorie	3	2126	5043	5052	5052	Wissenschaftstheorie	3	2126
5041	Ethik	4	2125	5041	5052	5052	Wissenschaftstheorie	3	2126
5041	Ethik	4	2125	5041	5216	5216	Bioethik	2	2126
5052	Wissenschaftstheorie	3	2126	5052	5259	5259	Der Wiener Kreis	2	2133
4630	Die 3 Kritiken	4	2137						
5049	Maeeutik	2	2125						
5022	Glaube und Wissen	2	2134						
4052	Logik	4	2125						
5216	Bioethik	2	2126						
5259	Der Wiener Kreis	2	2133						

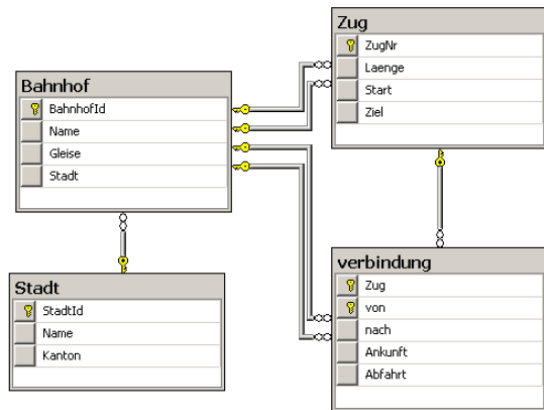
Aufgabe 36: U: Beschreiben Sie das Ergebnis der Anfrage von Aufgabe 35: in Ihren eigenen Worten.

Aufgabe 37: U: Reduzieren Sie die Tabelle aus Aufgabe 35:in einer Anfrage auf die Angabe der Vorlesungsnummern und der Titel.

Physisches ER-Modell für Zug-DB:



zugHalt.halt ist der Verweis auf einen Bahnhof. Beim Startbahnhof ist die Ankunft unbekannt (null), beim Endbahnhof ist die Abfahrt unbekannt (null). Die Reihenfolge der Zughalte ergibt sich aus den Abfahrt- und Ankunftszeiten. Mit dem Skripts erzeugen Sie sich obige Tabellen und füllen sie auch mit Beispieldaten.



Im Kemper/Eickel-Buch Abb. 3.11 ist ein ER-Modell für „Zugverbindungen“, in dem von-/nach-Bahnhof explizit modelliert ist. Start- und Ziel-Bahnhof sind auch explizit modelliert.

Hier ist ein (verkürztes) physisches ER-Modell dafür: Start- und Zielbahnhof sind nicht mehr explizit modelliert. Eine Verbindung ohne Ankunftszeit entspricht dem Start, eine Verbindung ohne Abfahrtszeit (und ohne nach-Bahnhof) dem Ziel.

Sie können für Ihre select-Übungen auch von diesem Modell ausgehen – im Skript sind entsprechende views kreiert.

Beispieldaten:
Stadt

StadtId	Name	Kanton
7	Baden	Aargau
5	Basel	Basel Stadt
2	Horw	Luzern
1	Luzern	Luzern
6	Olten	Solothurn
8	Thalwil	Zuerich
4	Zuerich	Zuerich
3	Zug	Zug

Bahnhof

BahnhofId	Name	Gleise	Stadt
1	Luzern	15	1
2	Basel SBB	17	5
3	Basel Bad	4	5
4	Thalwil	5	8
5	Olten	18	6
6	Horw	3	2
7	Zug	8	3
8	Baden	7	7
9	Zuerich HB	52	4

Zughalt (Fremdschlüssel halt durch bahnhof.name ersetzt):

zug	name	Ankunft	abfahrt
1	Luzern	NULL	12:12:00...
1	Zug	12:35:00...	12:37:00...
1	Thalwil	12:50:00...	12:52:00...
1	Zuerich HB	13:05:00...	NULL
2	Zuerich HB	NULL	13:15:00...
2	Thalwil	13:22:00...	13:23:00...
2	Zug	13:40:00...	13:42:00...
2	Luzern	14:00:00...	NULL
3	Luzern	NULL	14:10:00...
3	Horw	14:14:00...	NULL
4	Horw	NULL	14:21:00...
4	Luzern	14:25:00...	NULL
5	Luzern	NULL	14:12:00...
5	Olten	14:42:00...	14:45:00...
5	Basel SBB	15:08:00...	NULL
6	Basel SBB	NULL	15:33:00...
6	Olten	15:12:00...	16:05:00...
6	Luzern	15:36:00...	NULL
7	Zuerich HB	NULL	13:25:00...
7	Baden	13:37:00...	13:39:00...
7	Basel SBB	14:05:00...	14:08:00...
7	Basel Bad	14:12:00...	NULL
8	Basel Bad	NULL	14:26:00...
8	Basel SBB	14:30:00...	14:32:00...
8	Baden	15:05:00...	15:07:00...
8	Zuerich HB	15:20:00...	NULL

Zug2

ZugNr	Laenge
1	45
2	48
3	25
4	25
5	60
6	60
7	60
8	60