

Nicht korrespondierende Unterabfragen, die einen Wert liefern

Wir möchten als erstes Beispiel alle Schüler und das Geburtsdatum des jüngsten Schülers ausgeben. Dies klingt einmal einfach, stellt uns aber vor ein Problem.

Einerseits erfolgt die Ausgabe der Schüler mit folgendem normalen SQL Statement:

```
SELECT s.S_Nr, s.S_Zuname, s.S_Vorname
FROM Schueler s;
```

Das Ermitteln des jüngsten Schülers ist auch keine schwierige Sache, es ist eine einfache *MAX()* Funktion.

```
SELECT MAX(s.S_Gebdatum) FROM Schueler s;
```

Die Schwierigkeit ist die Kombination der beiden Ergebnisse. Denn wenn wir *MAX()* verwenden, gruppieren wir die Daten. Es entsteht 1 Datensatz pro Tabelle. Andererseits möchten wir aber alle Schülerdaten ausgeben.

Die Lösung bieten Unterabfragen. Sie können - je nach ihrem Rückgabewert - in 3 Kategorien eingeteilt werden:

- Unterabfragen, die genau einen Wert liefern.
- Unterabfragen, die eine Liste von Werten (1 Spalte, mehrere Zeilen) liefern.
- Unterabfragen, die ganze Tabellen liefern.

Die Verwendung von Unterabfragen, die genau einen Wert liefern, ist sehr leicht. Sie werden dort eingesetzt, wo normalerweise Spalten stehen. Unser Beispiel sieht dann so aus:

```
SELECT
    s.S_Nr, s.S_Zuname, s.S_Vorname,
    (SELECT MAX(s.S_Gebdatum) FROM Schueler s) AS Aeltester
FROM Schueler s;
```

S_Nr	S_Zuname	S_Vorname	Aeltester
1000	Pfannerstill	Ricardo	2005-08-30
1001	Mohr	Alyssa	2005-08-30
1002	Schneider	Lyle	2005-08-30
...

Hinweis: Die Unterabfrage muss immer eingeklammert sein!

Vergleich mit Programmcode

Hilfreich ist auch diese Analogie in den Programmiersprachen. Da die Unterabfrage keinerlei Daten von der äußeren Abfrage braucht, wird sie sozusagen vorher ausgeführt und wie als Variable gespeichert.

```
var value = MySubQuery();
foreach (Abteilung a in db.Abteilungen)
{
    Console.WriteLine($"{a.AbtNr} {a.AbtLeiter} {value}")
}
```

Verwendung mehrerer Unterabfragen

Natürlich können auch mehrere Unterabfragen verwendet werden. Somit kann z. B. das Geburtsdatum des ältesten und jüngsten Schülers ermittelt werden:

```
SELECT
    s.S_Nr, s.S_Zuname, s.S_Vorname,
    (SELECT MAX(s.S_Gebdatum) FROM Schueler s) AS Aeltester,
    (SELECT MIN(s.S_Gebdatum) FROM Schueler s) AS Juengster
FROM Schueler s;
```

S_Nr	S_Zuname	S_Vorname	Aeltester	Juengster
1000	Pfannerstill	Ricardo	2005-08-30	1989-09-14
1001	Mohr	Alyssa	2005-08-30	1989-09-14
1002	Schneider	Lyle	2005-08-30	1989-09-14
...

Unterabfragen in Ausdrücken

Da die angegebenen Unterabfragen nur 1 Wert zurückliefern, können sie auch in Ausdrücken verwendet werden. Hier wird die Altersdifferenz in Tagen zum ältesten Schüler berechnet. Die Funktion *JULIANDAY()* ist speziell für SQLite, denn sonst würde die Differenz in Jahren berechnet werden.

```
SELECT
    s.S_Nr, s.S_Zuname, s.S_Vorname,
    JULIANDAY(s.S_Gebdatum) - JULIANDAY((SELECT MIN(s.S_Gebdatum) FROM Schueler s)) AS DiffZuAeltester
FROM Schueler s;
```

S_Nr	S_Zuname	S_Vorname	DiffZuAeltester
1000	Pfannerstill	Ricardo	1704
1001	Mohr	Alyssa	2221
1002	Schneider	Lyle	2979
...

Unterabfragen in Filterkriterien (WHERE und HAVING)

Möchten wir alle Schüler ausgeben, die im selben Jahr wie der älteste Schüler geboren sind, so verwenden wir unsere Unterabfrage einfach in *WHERE*. Die Funktion *STRFTIME()* ist speziell für SQLite und gibt Teile (hier das Jahr) des Datumswertes zurück. Beachte die Klammerung der Unterabfrage im Argument von *STRFTIME*!

```
SELECT    s.S_Nr, s.S_Zuname, s.S_Vorname, s.S_Klasse, s.S_Gebdatum
FROM      Schueler s
WHERE     STRFTIME('%Y', s.S_Gebdatum) = STRFTIME('%Y', (SELECT MIN(s.S_Gebdatum) FROM Schueler s))
ORDER BY  s.S_Klasse, s.S_Nr;
```

S_Nr	S_Zuname	S_Vorname	S_Klasse	S_Gebdatum
1917	Rippin	Josh	1AVIF	1989-10-31
1835	Gibson	Wallace	1DVIF	1989-12-20
1848	Kunde	Amos	1DVIF	1989-10-20
1010	Auer	Bethany	1EVIF	1989-10-02
...

Nun wollen wir die Klassen herausfinden, wo der älteste Schüler der Klasse im selben Jahr wie der älteste Schüler der Schule geboren wurde. Im Gegensatz zur vorigen Abfrage wird jetzt jede Klasse nur 1x ausgegeben.

```
SELECT    s.S_Klasse, MIN(s.S_Gebdatum) AS Aelterster
FROM      Schueler s
GROUP BY  s.S_Klasse
HAVING    MIN(s.S_Gebdatum) = (SELECT MIN(s.S_Gebdatum) FROM Schueler s);
```

S_Klasse	Aelterster
1AVIF	1989-10-31
1DVIF	1989-10-20
1EVIF	1989-10-02

S_Klasse	Aelterster
3ACIF	1989-10-22
3AKIF	1989-11-11
3BAIF	1989-12-19
3BKIF	1989-10-26
3CAIF	1989-11-12
5ACIF	1989-09-25
5BBIF	1989-12-25
5CAIF	1989-11-27
7ACIF	1989-11-30
7BBIF	1989-09-14

Zusammenfassung: Unterabfragen, die einen Wert liefern, lassen sich wie Variablen behandeln. Sie können überall dort eingesetzt werden, wo Spalten oder fixe Werte stehen können.

Übungen

Bearbeiten Sie die folgenden Abfragen. Die korrekte Lösung ist in der Tabelle darunter, die erste Spalte (#) ist allerdings nur die Datensatznummer und kommt im Abfrageergebnis nicht vor. Die Bezeichnung der Spalten, die Formatierung und die Sortierung muss nicht exakt übereinstimmen.

(1) Welche Lehrer sind neu bei uns, haben also das maximale Eintrittsjahr?

#	LNr	LName	LVorname	LEintrittsjahr
1	BAM	Balluch	Manfred	2019
2	BOA	Bohn	Adele	2019
3	CO	Coufal	Klaus	2019
4	DOB	Dormayer	Bernd	2019
5	FEM	Felix	Mario	2019
6	GA	Gschaidner	Andreas	2019
7	MAY	Mayer	Sonja	2019
8	MOS	Moser	Gabriele	2019
9	NAI	Naimer	Eva Maria	2019
10	OEM	Öhlknecht	Martin	2019
11	POD	Poppel	Dominik	2019
12	SAB	San	Berg	2019
13	SAC	Schachner	Christine	2019
14	SCM	Schrammel	Manuela	2019
15	SIL	Siller	Waltraud	2019
16	WEM	Wessely	Mario	2019
17	ZIP	Zippel	Erich	2019
18	ZOC	Zöchbauer	Christian	2019

(2) Geben Sie die Klassen der Abteilung AIF und die Anzahl der gesamten Klassen und Schüler der Schule aus.

#	Klasse	KlassenGesamt	SchuelerGesamt
1	2AAIF	116	2462
2	2BAIF	116	2462
3	2CAIF	116	2462
4	2DAIF	116	2462
5	3BAIF	116	2462
6	3CAIF	116	2462
7	4BAIF	116	2462
8	4CAIF	116	2462

#	Klasse	KlassenGesamt	SchuelerGesamt
9	5BAIF	116	2462
10	5CAIF	116	2462
11	6BAIF	116	2462
12	6CAIF	116	2462

(3) Geben Sie bei allen Lehrern, die 2018 eingetreten sind (Spalte *L_Eintrittsjahr*), das Durchschnittsgehalt (gerechnet über alle Lehrer der Schule) aus.

#	LNr	LName	LVorname	LEintrittsjahr	LGehalt	AvgGehalt
1	AH	Auinger	Harald	2018	2083	3126.67
2	BIE	Bierbamer	Peter	2018	2225	3126.67
3	CAM	Camrda	Christian	2018		3126.67
4	HY	Horny	Christian	2018	2224	3126.67
5	KEM	Keminger	Alexander	2018	2138	3126.67
6	KMO	Kmyta	Olga	2018	2122	3126.67
7	MC	Marek	Clemens	2018	2158	3126.67
8	PEC	Pemöller	Christoph	2018		3126.67
9	SE	Schmid	Erhard	2018	2064	3126.67
10	ZLA	Zlabinger	Walter	2018	2256	3126.67

(4) Als Ergänzung geben Sie nun bei diesen Lehrern die Abweichung vom Durchschnittsgehalt aus. Zeigen Sie dabei nur die Lehrer an, über 1000 Euro unter diesem Durchschnittswert verdienen.

#	LNr	LName	LVorname	LEintrittsjahr	LGehalt	AvgGehalt	Abweichung
1	AH	Auinger	Harald	2018	2083	3126.67	-1043.67
2	KMO	Kmyta	Olga	2018	2122	3126.67	-1004.67
3	SE	Schmid	Erhard	2018	2064	3126.67	-1062.67

(5) Geben Sie die Prüfungen aus, die maximal 3 Tage vor der letzten Prüfung stattfanden.

#	PDatumZeit	PPruefer	PNote	Zuname	Vorname
1	31.05.2020 20:10:00	SAB	1	Kuhlman	Frances
2	31.05.2020 17:20:00	PC		Hammes	Danny
3	31.05.2020 12:55:00	SJ		Balistreri	Irene
4	31.05.2020 09:10:00	SGC	2	Quitzon	Sue
5	30.05.2020 20:55:00	SPN	3	Rohan	Tracy
6	30.05.2020 20:10:00	VOG	2	Hahn	Oliver
7	30.05.2020 18:25:00	HOH	2	Sanford	Everett
8	30.05.2020 17:25:00	BEC		Nicolas	Erika
9	30.05.2020 16:05:00	KNT	4	Klocko	Kristie
10	30.05.2020 15:30:00	WAG	3	Frami	Timothy
11	30.05.2020 11:00:00	JAD	3	Robel	Drew
12	30.05.2020 10:15:00	PT	3	Zemlak	Katie
13	29.05.2020 12:20:00	HAU		Prohaska	Ross
14	29.05.2020 11:05:00	GAL	2	O'Hara	Hubert

(6) Geben Sie die Räume mit der meisten Kapazität (Spalte *R_Plaetze*) aus. Hinweis: Das können auch mehrere Räume sein.

#	RId	RPlaetze	RArt
1	AH.32	36	Naturwissenschaftlicher Raum
2	B5.09	36	Klassenraum

(7) Gibt es Räume, die unter einem Viertel der Plätze als der größte Raum haben?

#	RId	RPlaetze	RArt
1	A3.04	3	Multifunktionsraum Medien
2	AH.21	6	Bibliothek
3	BH.08W	8	Maschentechnik
4	BH.09aW	8	Gewebetechnik
5	BH.10	8	Schweisstechnik
6	BH.11W	8	Mechanik
7	BLA	8	Betriebslaboratorium
8	DE.04L	8	Instrumentelle Analytik
9	DE.05L	8	Analytik und Prüftechnik
10	DE.06L	8	Umweltlabor
11	DE.09L	8	Nasschemisches Labor
12	DE.10W	8	Reinigungs- und Facilitytechnik
13	DE.11W	8	Färbe-, Veredlungs- und Verfahrenstechnik
14	DE.12aW	8	Farblabor
15	DE.13aW	8	Drucktechnik
16	DE.15L	8	Versuchsanstalt Prüflabor
17	DE.19L	8	Versuchsanstalt Grünbereich

(8) Welche Klasse hat mehr weibliche Schüler (S_Geschlecht ist 2) als die 5BAIF? Hinweis: Gruppieren Sie die Schülertabelle und vergleichen die Anzahl mit dem ermittelten Wert aus der 5BAIF.

#	Klasse	AnzWeibl
1	1AFITN	15
2	1AHBGM	15
3	1AHKUI	18
4	1AHMNA	19
5	1AHWIT	17
6	1AO	15
7	1BFITN	18
8	1BHBGM	18
9	1BHWIT	17
10	1CVIF	19
11	1DVIF	15
12	1EVIF	15
13	2AFITN	17
14	2AHBGM	19
15	2AHKUI	18
16	2BHWIT	18
17	2CHWIT	19
18	3ACIF	17
19	3AKKUI	16
20	3CAIF	16
21	4AFITM	15
22	4AHWIT	17
23	4BHBGM	19
24	5ACMNA	15
25	5AHBGM	15
26	5AKIF	16

#	Klasse	AnzWeibl
27	5CAIF	17

(9) Geben Sie die Klassen der Abteilung BIF sowie die Anzahl der Schüler in dieser Abteilung aus. Hinweis: Verwenden Sie GROUP BY, um die Schüleranzahl pro Klasse zu ermitteln. Achten Sie auch darauf, dass Klassen mit 0 Schülern auch angezeigt werden. Danach schreiben Sie eine Unterabfrage, die die Schüler der BIF Abteilung zählt.

#	Klasse	SchuelerKlasse	SchuelerBIF
1	2ABIF	0	105
2	3BBIF	30	105
3	4BBIF	0	105
4	5BBIF	24	105
5	6BBIF	0	105
6	7BBIF	25	105
7	7CBIF	26	105
8	8BBIF	0	105
9	8CBIF	0	105