
PDB (Teil 2) – 40 Punkte

Aufgabenblatt Nr. 6 – SQL

Abgabe: bis Sonntag 03.07.2022 bis 09:00 Uhr!

Setup: Wir gehen für alle Aufgaben (auch für die weiteren SQL-Aufgaben in kommenden Aufgabenblättern) davon aus, dass in der **Datenbank „krankenhaus“** gearbeitet wird. Sie sollten in Ihrem DBMS diese Datenbank vorab anlegen (`create schema krankenhaus;`).

Für die Aufgabe werden die Dateien: `blatt6_template.sql` und `blatt6_reset.sql` online zur Verfügung gestellt. Diese werden für den SQL Checker benötigt.

Nutzen Sie als Abgabe nur die vom SQL Checker erstellte .txt-Datei! Andere Abgabeformate werden nicht akzeptiert bzw. bringen keine Punkte!

Denken Sie daran:

1. Nur ein SQL-Statement pro Teilaufgabe!
2. Nur mit Kommentaren können Teilpunkte vergeben werden, insb. wenn das Verständnis der Aufgabenstellung von der unseren abweichen sollte.
3. Nicht im Team einreichen (nur eine Person im SQL Checker eintragen).

Sie sind selbst verantwortlich dafür die erstellten Anfragen vernünftig zu testen. Dazu gehört auch, dass die gegebenen Daten aus dem Reset-Script erweitert werden müssen. Bei der Evaluation ihrer Abgabe werden andere Daten verwendet als die vom Reset-Script (hierbei kann es auch vorkommen, dass die Tabellenstruktur durch Alter-Table Statements verändert wird, Ihre Anfragen sollten trotzdem stabil laufen und die geforderten Ergebnisse liefern)!

Die erwartete Ergebnisstruktur und die Ergebnisse bezogen auf die Daten des Reset-Scripts sind bei den Teilaufgaben angegeben. Diese Ergebnisstruktur ist zwingend einzuhalten, auch was die Groß- und Kleinschreibung angeht! Gleiches gilt für die Verwendung von Tabellen und Attributnamen innerhalb der Statements!

Aufgabe 1 – SQL - Funktionen (24 Punkte – je 6 Punkte):

Schreiben Sie jeweils eine Funktion für die gegebene Aufgabenstellung. Achten Sie auch darauf in der Abgabe **KEINE Delimiter-Angaben** und **KEINE DEFINER-Angabe** anzugeben.

Da mit MariaDB evaluiert wird, ist ebenfalls keine „characteristic“ Angabe nötig, also ob die Funktion DETERMINISTIC ist oder nicht. Es ist jeweils die ganze Signatur inklusive des Rückgabetypes einzuhalten!

Allgemein gilt: Sollten **NULL-Values** übergeben werden, wird auch **NULL** als Rückgabe erwartet.

Ein gültiger Eintrag im SQL-Checker sollte also ungefähr so aussehen (nur halt mit dem richtigen Funktionsnamen, der Logik in der Mitte und der richtigen Rückgabe):

```
CREATE FUNCTION new_function ()  
RETURNS ...  
BEGIN  
#... Logik  
RETURN ...;  
END
```

a) Erstellen Sie die Funktion:

quersumme(value int) RETURNS int

Die Funktion soll für den gegebenen Integer die entsprechende Quersumme ausgeben. Werte außerhalb des Wertebereiches müssen nicht behandelt werden.

Beispielaufwurf: **select quersumme(-123), quersumme(23), quersumme(null);**

Ergebnis:

quersumme(-123)	quersumme(23)	quersumme(null)
6	5	NULL

b) Erstellen Sie die Funktion:

statistic_value(name varchar(100), gebdat date) RETURNS varchar(5)

Die Funktion soll anonymisierte Werte für spätere Auswertungen erzeugen (wie weit dies den Regeln für Anonymisierungen entspricht steht auf einem anderen Blatt). Vom Namen soll der erste Buchstabe (kein Leerzeichen) und vom Geburtstag das Geburtsjahr kombiniert werden. Ist einer der übergebenen Werte NULL soll auch NULL zurückgegeben werden.

Beispielaufwurf: **select name, gebdat, statistic_value(name, gebdat) from patient limit 4;**

Ergebnis:

name	gebdat	statistic_value(name, gebdat)
Max Mustermann	1993-07-15	M1993
Mika Mustermann	1990-10-03	M1990
Moritz Petersen	1982-12-21	M1982
Hanna Mai	1992-04-09	H1992

c) Erstellen Sie die Funktion:

verschreibungenMonat(patnr int, monat date) RETURNS varchar(500)

Die Funktion soll für den gegebenen Patienten alle Medikamente (alphabetisch sortiert nach deren Namen) auflisten, welche in dem übergebenen Monat verschrieben wurden. Der Wochentag des übergebenen Monats ist dabei unerheblich, ob also '2022-06-20' oder '2022-06-02' ist unerheblich und es soll immer für '2022-06' berechnet werden.

Beispielaufwurf: **select verschreibungenMonat(1, '2022-06-20');**

Ergebnis:

verschreibungenMonat(1, '2022-06-20')
Iberogast,Ibuprophen

d) Erstellen Sie die Funktion:

medikament_bestell(name varchar(100)) RETURNS varchar(100)

welche für Medikamente, prüft, wie diese in den letzten 90 Tagen seit der Ausführung der Funktion verbraucht wurden (also entsprechend Eintrag der Dosierung in der Tabelle *verschreibt*). Weiterhin soll der aktuell Bestand angegeben werden. Zusätzlich soll am Anfang ein Bestellstatus angegeben werden: "Status: 0 - dringend|Status: 1 bestellen|Status: 2 – vorrätig“. Der Bestellstatus 0: Wenn der aktuelle Bestand minus dem Verbrauch kleiner gleich 10 ist. Bestellstatus 1: Wenn der aktuelle Bestand minus dem Verbrauch kleiner gleich 100 ist. Bestellstatus 2: Wenn der aktuelle Bestand minus dem Verbrauch größer als 100 ist. Die Angaben zum Bestellstatus, dem Verbrauch und dem Bestand sollen durch Semikolons getrennt sein.

Beispielaufwurf:

select name, medikament_bestell(name) from medikament order by 2;

Ergebnis:

name	medikament_bestell(name)
Sergos	Status: 0 - dringend; Verbrauch: 0; Bestand: 8
Iberogast	Status: 0 - dringend; Verbrauch: 9; Bestand: 10
Sinupret	Status: 1 - bestellen; Verbrauch: 0; Bestand: 100
Nasentropfen	Status: 1 - bestellen; Verbrauch: 0; Bestand: 100
Aspirin Complex	Status: 1 - bestellen; Verbrauch: 0; Bestand: 100
Nasenspray	Status: 1 - bestellen; Verbrauch: 0; Bestand: 50
Insulin	Status: 1 - bestellen; Verbrauch: 23; Bestand: 100
Ibuprophen	Status: 1 - bestellen; Verbrauch: 44; Bestand: 120
Aspirin	Status: 2 - vorrätig; Verbrauch: 0; Bestand: 200

Aufgabe 2 – SQL – 12 Punkte (je 4)

- a) Eine Anfrage, die die Namen der Medikamente auflistet, welche in den ersten vier Zeichen (inkl. eventueller Leerzeichen und unabhängig von der Groß-Klein-Schreibung) übereinstimmen. (Beachten Sie, dass jedes Medikamentenpaar auch gespiegelt auftauchen sollte.) Sortieren Sie das Ergebnis nach *name2* aufsteigend.

name1	name2
Aspirin Complex	Aspirin
Aspirin	Aspirin Complex
Nasentropfen	Nasenspray
Nasenspray	Nasentropfen

- b) Eine Anfrage, welche für die Mitarbeiter deren PLZ mit „6“ beginnt, die angegebenen Mitarbeiterdaten ausgibt und falls der Mitarbeiter ein Arzt ist auch dessen Fachgebiet.

pnr	name	gebdat	fachgebiet
12	Caroline	1963-04-11	Orthopaedie
7	albert Einst...	1879-03-14	Innere Medizin
10	Hans	1955-12-03	Chirurgie
17	Lukas	1967-01-01	NULL
19	Tibor	1977-03-25	NULL
16	Alexander	1982-08-30	NULL

- c) Eine Anfrage, welche für die Mitarbeiter deren Geburtstag vor dem Jahr 1900 liegt, die angegebenen Mitarbeiterdaten ausgibt und falls der Mitarbeiter ein Pfleger ist auch den Namen der Station, in welcher der Pfleger arbeitet.

pnr	name	gebdat	StationsName
1	Albert Eins...	1879-03-14	HNO
2	Marie Curie	1867-11-07	Neurologie
4	Ada Lovelace	1815-12-10	NULL
7	albert Einst...	1879-03-14	NULL

- d) Eine Anfrage, welche alle Fachnamen der eingetragenen Krankheiten angibt und in einer Symptomliste die entsprechenden Symptome auflistet (jedes Symptom pro Liste max. 1 mal, die Symptomliste soll **aufsteigend sortiert** sein). Sind keine Symptome vorhanden, soll NULL angegeben werden. Das Ergebnis soll aufsteigend nach dem Fachnamen der Krankheit sortiert sein.

fachname	Symptomliste
Apoplex	Laehmmung,Hautausschlag
Arderum	Vergesslichkeit,Schwellung
Commotio cerebri	NULL
Fussi Juckus	NULL
Gastroenteritis	Kopfschmerzen,Husten,Erbrechen
Lebensmittelintoxikation	Erbrechen