

Mario Fentler 5CHIT

27.10.2018

Dokumentation AU01

Aufgabenstellung

Die Mitschrift bzw. das DDL-Script für "Restaurant" soll um die beiden Funktionen

- preis99() - Schreibt die Nachkommastelle auf 99 um auf drei verschiedene Arten !
- loescheRechnung()

wie in der Vortragsunterlage ergänzt werden.

Inkl. INSERT bzw. Aufruf vorher/nachher...

Aufgabendurchführung

Um die Aufgabe zu lösen wird eine Debian VM mit einer Postgres Datenbank verwendet.

Die Datenbank wird durch das SQL-File aus dem E-Learning Kurs erstellt. Um sicherzustellen, dass die Funktionen auch korrekt funktionieren, wird die Tabelle Speise für jeden Task neu erstellt.

```
DROP TABLE IF EXISTS speise CASCADE;
CREATE TABLE speise (
    snr            INTEGER,
    bezeichnung    VARCHAR(255),
    preis          DECIMAL(6,2),
    PRIMARY KEY (snr)
);

INSERT INTO speise VALUES (1, 'Heisse Liebe',      3);
INSERT INTO speise VALUES (2, 'Schoko Palatschinken', 4);
INSERT INTO speise VALUES (3, 'Pute gebacken',      7);
INSERT INTO speise VALUES (4, 'Pute natur',          8);
INSERT INTO speise VALUES (5, 'Puten-Cordon',        9);
INSERT INTO speise VALUES (6, 'Menue fuer 2',       15);
INSERT INTO speise VALUES (7, 'Menue fuer 3',       19);
INSERT INTO speise VALUES (8, 'Menue fuer 4',       22);
```

Erstelle Funktionen ausführen

Um die Funktionen, die man erstellt hat, auszuführen schreibt man:

```
SELECT * FROM <Funktionsname>();
```

Task 1 - Preis

Dazu gibt es 3 verschiedene Arten die Aufgabenstellung umzusetzen.

- Preis aufrunden, - 1 Cent
- Preis abrunden, + 99 Cent
- Nachkommastelle vom Preis abschneiden, + 0.99

Variante 1

Der Preis wird aufgerundet und danach wird 1 Cent abgezogen.

```
CREATE FUNCTION preis_var1() RETURNS VOID AS '
UPDATE speise SET preis = ceil(preis) - 0.01; ' LANGUAGE SQL;
```

Davor/danach:

```
restaurant=# select * from speise;
 snr |      bezeichnung      | preis
-----+-----+-----
  1 | Heisse Liebe          |  3.00
  2 | Schoko Palatschinken |  4.00
  3 | Pute gebacken         |  7.00
  4 | Pute natur            |  8.00
  5 | Puten-Cordon          |  9.00
  6 | Menue fuer 2          | 15.00
  7 | Menue fuer 3          | 19.00
  8 | Menue fuer 4          | 22.00
(8 rows)

restaurant=# CREATE FUNCTION preis_var1() RETURNS VOID AS '
restaurant'#      UPDATE speise SET preis = ceil(preis) - 0.01; ' LANGUAGE SQL;
CREATE FUNCTION
restaurant=# select * from preis_var1();
 preis_var1
-----
(1 row)

restaurant=# select * from speise;
 snr |      bezeichnung      | preis
-----+-----+-----
  1 | Heisse Liebe          |  2.99
  2 | Schoko Palatschinken |  3.99
  3 | Pute gebacken         |  6.99
  4 | Pute natur            |  7.99
  5 | Puten-Cordon          |  8.99
  6 | Menue fuer 2          | 14.99
  7 | Menue fuer 3          | 18.99
  8 | Menue fuer 4          | 21.99
(8 rows)

restaurant=#
```

Variante 2

Der Preis wird abgerundet und dann werden 99 Cent dazu addiert.

```
CREATE FUNCTION preis_var2() RETURNS VOID AS '
UPDATE speise SET preis = floor(preis) + 0.99; ' LANGUAGE SQL;
```

Davor/danach:

```
restaurant=# select * from speise;
 snr |      bezeichnung      | preis
-----+-----+-----
  1 | Heisse Liebe          |  3.00
  2 | Schoko Palatschinken |  4.00
  3 | Pute gebacken         |  7.00
  4 | Pute natur            |  8.00
  5 | Puten-Cordon          |  9.00
  6 | Menue fuer 2          | 15.00
  7 | Menue fuer 3          | 19.00
  8 | Menue fuer 4          | 22.00
(8 rows)

restaurant=# CREATE FUNCTION preis_var2() RETURNS VOID AS '
restaurant'#      UPDATE speise SET preis = floor(preis) + 0.99; ' LANGUAGE SQL;
CREATE FUNCTION
restaurant=# select * from preis_var2();
 preis_var2
-----
(1 row)

restaurant=# select * from speise;
 snr |      bezeichnung      | preis
-----+-----+-----
  1 | Heisse Liebe          |  3.99
  2 | Schoko Palatschinken |  4.99
  3 | Pute gebacken         |  7.99
  4 | Pute natur            |  8.99
  5 | Puten-Cordon          |  9.99
  6 | Menue fuer 2          | 15.99
  7 | Menue fuer 3          | 19.99
  8 | Menue fuer 4          | 22.99
(8 rows)

restaurant=# █
```

Variante 3

Die Nachkommastellen werden abgeschnitten und anschließend werden 99 Cent dazu addiert.

```
CREATE FUNCTION preis_var3() RETURNS VOID AS '
UPDATE speise SET preis = trunc(preis) + 0.99; ' LANGUAGE SQL;
```

Davor/danach:

```
restaurant=# select * from speise;
 snr |      bezeichnung      | preis
-----+-----+-----
  1 | Heisse Liebe          |  3.00
  2 | Schoko Palatschinken |  4.00
  3 | Pute gebacken         |  7.00
  4 | Pute natur            |  8.00
  5 | Puten-Cordon         |  9.00
  6 | Menue fuer 2         | 15.00
  7 | Menue fuer 3         | 19.00
  8 | Menue fuer 4         | 22.00
(8 rows)

restaurant=# CREATE FUNCTION preis_var3() RETURNS VOID AS '
restaurant'#      UPDATE speise SET preis = trunc(preis) + 0.99; ' LANGUAGE SQL;
CREATE FUNCTION
restaurant=# select * from preis_var3();
 preis_var3
-----
(1 row)

restaurant=# select * from speise;
 snr |      bezeichnung      | preis
-----+-----+-----
  1 | Heisse Liebe          |  3.99
  2 | Schoko Palatschinken |  4.99
  3 | Pute gebacken         |  7.99
  4 | Pute natur            |  8.99
  5 | Puten-Cordon         |  9.99
  6 | Menue fuer 2         | 15.99
  7 | Menue fuer 3         | 19.99
  8 | Menue fuer 4         | 22.99
(8 rows)

restaurant=# █
```

Task 2 - Rechnung loeschen

Der Task laut Angabe ist es die Rechnungen zu löschen, zu denen es keine Bestellung gibt.

-> Ich schließe daraus, dass die Rechnungen gelöscht werden sollen, die schon bezahlt wurden.

Um diesen Task zu lösen muss die **Syntax** für die Erstellung einer Funktion **geändert** werden. Denn so, wie sie bis jetzt erstellt wurde kann man in der Funktion **keinen String** angeben. Man muss allerdings nach dem String 'bezahlt' suchen.

-> Neue Syntax "\$\$... \$\$"

```
CREATE FUNCTION loeschePreis() RETURNS VOID AS $$
DELETE FROM rechnung WHERE status='bezahlt'; $$
LANGUAGE SQL;
```

Davor/danach:

```

restaurant=# CREATE FUNCTION loeschePreis() RETURNS VOID AS $$
restaurant$#      DELETE FROM rechnung WHERE status='bezahlt'; $$
restaurant=#      LANGUAGE SQL;
CREATE FUNCTION
restaurant=# select * from rechnung;
 rnr |   datum   | tisch | status | knr
-----+-----+-----+-----+-----
  1 | 2013-03-07 |     1 | bezahlt | 1
  2 | 2013-03-07 |     2 | offen  | 2
  3 | 2013-03-07 |     1 | gedruckt | 3
  4 | 2013-03-07 |     1 | gedruckt | 1
  5 | 2013-03-07 |     1 | bezahlt | 1
  6 | 2013-03-07 |     2 | offen  | 2
(6 rows)

restaurant=# select * from loeschePreis();
 loeschepreis
-----
(1 row)

restaurant=# select * from rechnung;
 rnr |   datum   | tisch | status | knr
-----+-----+-----+-----+-----
  2 | 2013-03-07 |     2 | offen  | 2
  3 | 2013-03-07 |     1 | gedruckt | 3
  4 | 2013-03-07 |     1 | gedruckt | 1
  6 | 2013-03-07 |     2 | offen  | 2
(4 rows)

restaurant=# █

```

Quellen

- [1] <https://www.postgresql.org/docs/9.1/static/functions-math.html>
- [2] <http://www.postgresqltutorial.com/postgresql-delete/>