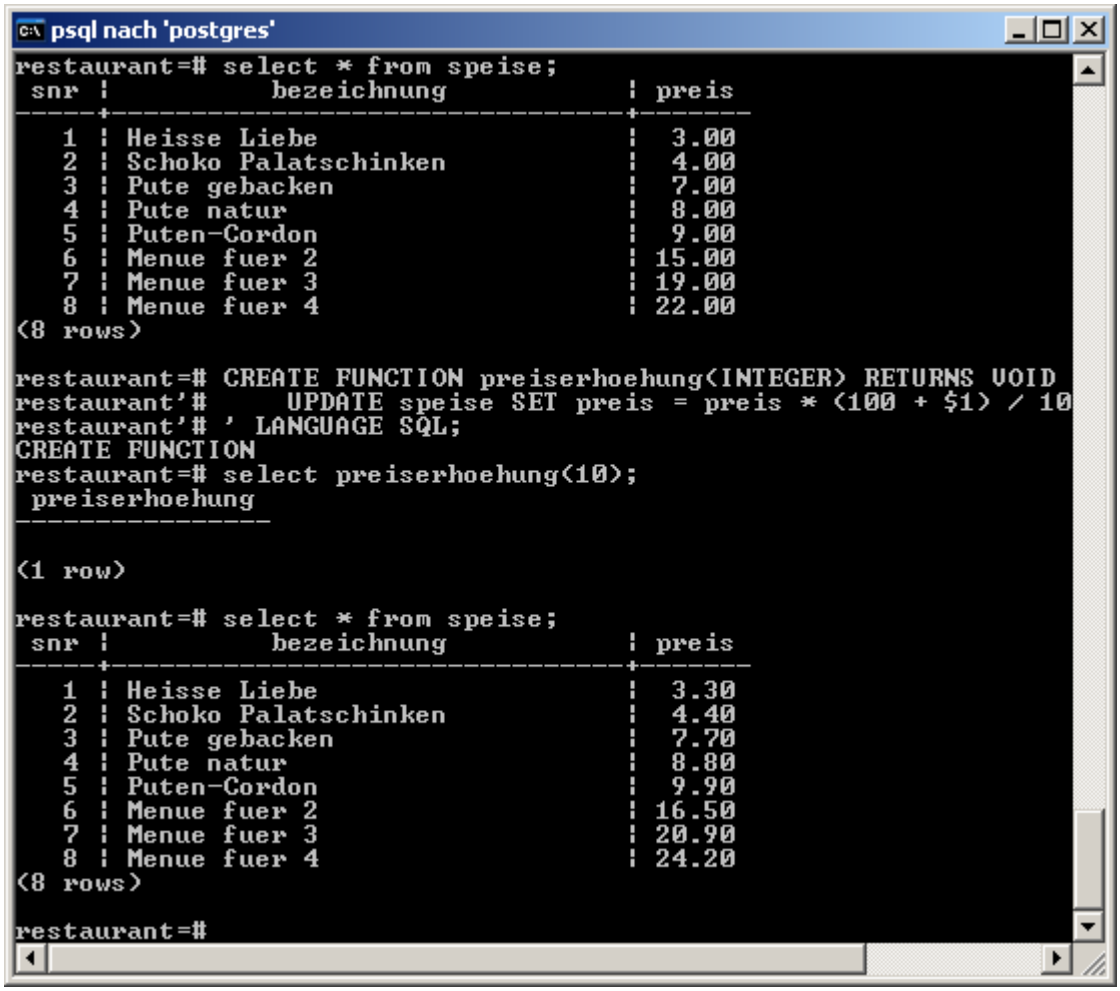
Funktionen mit Parameter / ohne Rückgabewert

Beispiel:
Die Preise aller Speisen sollen um einen variablen Prozentsatz erhöht werden.

```
CREATE FUNCTION preiserhoehung(INTEGER) RETURNS VOID AS '
    UPDATE speise SET preis = preis * (100 + $1) / 100;
' LANGUAGE SQL;
```

- `preiserhoehung(INTEGER)` definiert einen ganzzahligen Funktionsparameter.
- Mit `$1` kann auf diesen Wert im Funktionskörper zugegriffen werden.
  Auf weitere Parameter könnte mit `$2`, `$3` etc. zugegriffen werden.
- `DROP FUNCTION preiserhoehung(INTEGER)` löscht die Funktionsdefinition, d.h. man muss beim Löschen die Parameter der zu löschenden Funktion angeben.
  Alternative dazu: `CREATE OR REPLACE …`



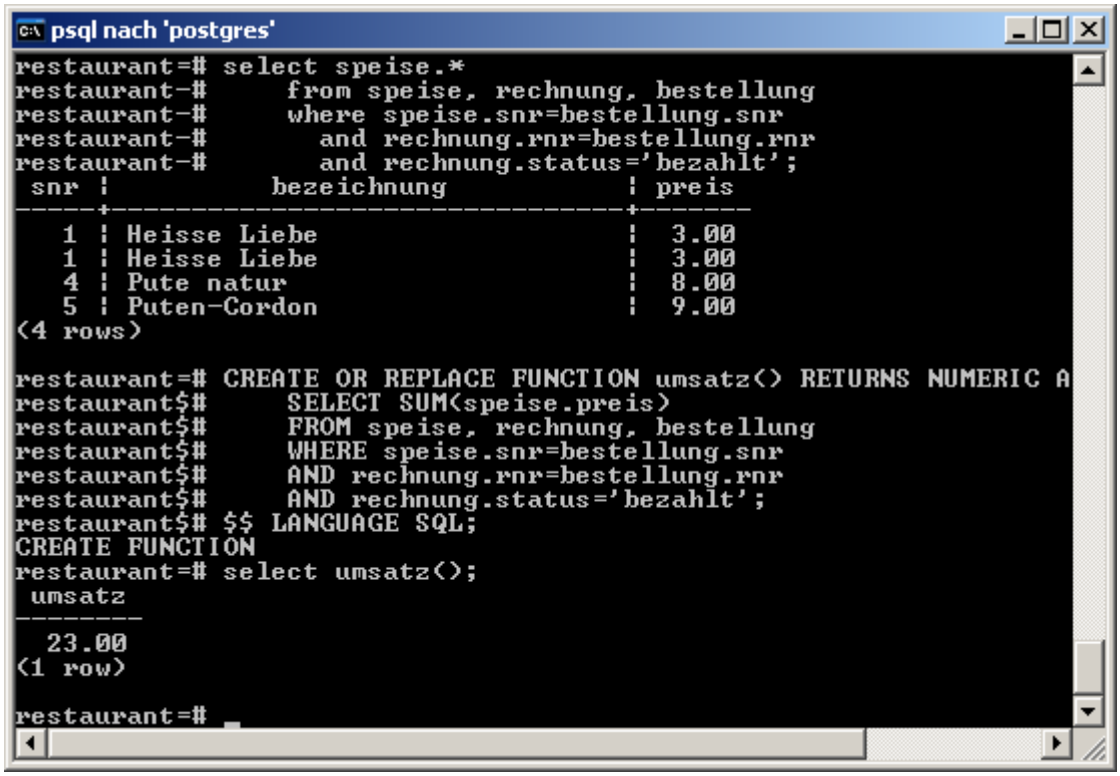
Übung:
Erstelle eine Funktion mit zwei Parametern: Alle Speisen die billiger als der Durchschnittspreis aller Speisen sind, sollen um einen fixen Betrag erhöht werden, alle Speisen die teurer sind, sollen um einen Prozentwert erhöht werden.

Funktionen ohne Parameter / mit Rückgabewert

Beispiel:
Der Umsatz der bezahlten Rechnungen soll ermittelt werden.

```
CREATE OR REPLACE FUNCTION umsatz() RETURNS NUMERIC AS $$
    SELECT SUM(speise.preis*bestellung.anzahl)
    FROM speise, rechnung, bestellung
    WHERE speise.snr=bestellung.snr
    AND rechnung.rnr=bestellung.rnr
    AND rechnung.status='bezahlt';
$$ LANGUAGE SQL;
```

- Die Definition `OR REPLACE` bewirkt, dass keine Fehlermeldung ausgegeben wird, wenn eine gleichnamige Funktion existiert, die nun überschrieben werden soll.
- Mit `RETURNS NUMERIC` wird definiert, dass die Funktion einen Wert vom Typ `NUMERIC` (d.h. Fließkomma) an den Aufrufer der Funktion zurückliefern soll.
  `SELECT SUM(speise.preis)` ist die letzte Anweisung im Funktionskörper und berechnet einen Fließkommawert. Dieser Wert ist der Rückgabewert.
- Die Angabe `$$` ist erforderlich, da innerhalb der Funktionsdefinition einfache Hochkommas für einen Stringvergleich verwendet werden müssen und diese die Funktionsdefinition unterbrechen würden.



Übung:
Der Tagesumsatz für einen bestimmten Kellner soll für den aktuellen Tag ermittelt werden (Kellner-Nr via Parameter, aktueller Tag via `CURRENT_DATE`).