

OttoQL-Eine einfache Programmiersprache auf 2 Seiten

2 oder 3 Worte in Google oder $3.19997*2.876$ in einen Taschenrechner eingegeben oder eine Datenbankbedingung `ORT="Hadmersleben"` an eine Tabelle gerichtet sind streng genommen sehr einfache Programme, wobei das letzte Problem in SQL ausführlich betrachtet die folgende Gestalt hätte.

```
SELECT *
FROM PERSONEN
WHERE ORT="Hadmersleben"
```

Ziel von OttoQL ist es Anfragen vom Googletyp auch auf (strukturierte) Tabellen anzuwenden und Anfragen vom SQL-Typ auch bei Dokumenten zuzulassen. Ferner sollen die Möglichkeiten des Taschenrechners ausgebaut werden.

Folgende OttoQL-Programme sind selbsterklärend:

Programm 1:

```
3.19997*2.876
```

Programm 2:

```
FLAECHE_RECHTECK := 3.19997*2.876
```

Programm 3: Gesucht sind die Durchschnitte in Mathe und Informatik und der Durchschnitt der Durchschnitte.

```
DUR_MA := avg(1 2 1 3 2)
ext DUR_INF := avg(1 3 2 1 1 1)
ext DUR := avg(DUR_MA, DUR_INF)
```

ext = Extension = Erweiterung der gegebenen Tabelle um eine neue Spalte

Programm 4a: Gesucht sind die Durchschnitte mehrerer Fächer und der Gesamtdurchschnitt.

```
<<L (FACH, L (NOTE)) : :
```

```
MA      1 2 1 3 2
INF     1 3 2 1 1 1
DEU     3 2 4 2 >>
```

```
gib DUR, M (FACH, DUR) DUR:=avg (NOTE)
```

Die Tabelle mit Noten kann man auch in einer separaten Tabelle `Noten.tab` speichern. Dann hat das obige Programm folgende Gestalt:

Programm 4b:

```
aus Noten.tab
gib DUR, M (FACH, DUR) DUR:=avg (NOTE)
```

Sind größere Tabellen oder Dokumente oder Mengen von Dokumenten gegeben, so ist die Selektion von größter Bedeutung. Gegeben ist eine Datei, die 45 Flusstupel enthält:

Fluesse.xml: M(FLUSSTUP)

FLUSSTUP: FLUSS, LAENGE, QUELLE, MUENDUNG, L(LAND, L(BUNDESLAND)), L(NEBENFLUSS)

Programm 5: Anfrage In welchen Fluss, fließt die Bode?

```
aus fluesse.xml
mit <NEBENFLUSS> = Bode
gib FLUSS
```

Programm 6: Gesucht sind alle Flusstupel, in denen das Wort Bode vorkommt.

```
aus fluesse.xml
mit Bode
```

Programm 7: Gesucht sind alle Flüsse, die durch Sachsen-Anhalt fließen, nach der Länge sortiert.

```
aus fluesse.xml
mit <BUNDESLAND> = Sachsen-Anhalt
gib M(LAENGE, FLUSS)
```

Programm 8: Gesucht sind alle Flüsse, die durch Sachsen-Anhalt fließen, der Länge nach abwärts sortiert mit allen Bundesländern, durch die sie fließen.

```
aus fluesse.xml
mit Sachsen-Anhalt
gib M-(LAENGE, FLUSS, M(BUNDESLAND))
```

Programm 9: Gesucht sind alle Flüsse, die durch Sachsenanhalt aber nicht durch Sachsen fließen.

```
aus fluesse.xml
mit Sachsen-Anhalt
ohne Sachsen
```

Hierbei handelt es sich um zwei Selektionen, die hintereinander ausgeführt werden.

Programm 10: Gesucht sind zu jedem Bundesland alle längeren Flüsse, nach der Länge abwärts sortiert. Die Bundesländer sollen alphabetisch sortiert werden.

```
aus fluesse.xml
mit <LAENGE> > 100
gib M(BUNDESLAND, M-(LAENGE, FLUSS))
```

Programm 11: Wie viele Dollar sind 500 Euro?

```
euro2dollar(500)
```

oder

```
500
euro2dollar
```

Das Ergebnis ist vom Typ DOLLAR, DATE, TIME, wobei das Datum und die Zeit nicht die aktuellen Werte darstellen, sondern die Daten, wann der Umrechnungswert zuletzt ermittelt wurde.

Programm 12: Wie viel Mehrwertsteuer muss ich für die folgenden Artikel bezahlen, wobei Artikel mit 19% bzw. 7% versteuert. Die Steuern sind zu summieren.

```
<<M(STEU_SATZ, L(ARTIKEL, ANZ, STUECK_PREIS)) ::
  0.19      AUDI      5      30000
           HUYNDAI   4      12000
           DACIA     3      8000
  0.07      Brot     8000   2.80
           Apfel    4000   1.25  >>
```

```
gib STEU, M(STEU_SATZ, STEU, L(ARTIKEL, ANZ, STUECK_PREIS, STEU)) &&
  STEU:=sum(ANZ * STUECK_PREIS * STEU_SATZ)
```

Programm 13: Wie viel habe ich 2010 und wie viel 2012 für VODAFONE bezahlt.

```
aus konto2010.csv, konto2012.csv
mit VODAFONE
gib SUM10, SUM12 SUM10:=sum(konto2010/Betrag) &&
  SUM12:=sum(konto2012/Betrag)
```

Hierbei sind konto2010 und konto2012 vom gleichen Typ:

L(Auftragskonto, Buchungstag, Valutadatum, Buchungstext, Verwendungszweck, Begünstigter_Zahlungspflichtiger, Kontonummer, BLZ, Betrag, Waehrung, Info)