



## Übung zur Vorlesung *Einsatz und Realisierung von Datenbanken im SoSe24*

Alice Rey, Maximilian Bandle, Michael Jungmair (i3erdb@in.tum.de)

<http://db.in.tum.de/teaching/ss24/impldb/>

### Blatt Nr. 11

**Hinweise** Die Aufgaben können auf <http://xquery.db.in.tum.de/> getestet werden. Die Daten für das Unischema können mit `doc('uni2')` geladen werden. Zur Lösung der Aufgaben können Sie die folgenden XQuery-Funktionen verwenden:

`max(NUM)`, `count(X)`, `tokenize(STR,SEP)`, `sum(NUM)`, `contains(HAY,NEEDLE)`

1. `max(NUMBERS)` - Returns largest number from list
2. `count(LIST)` - Return the number of elements in the list
3. `tokenize(STR,SEP)` - Splits up the string at the separator
4. `sum(NUMBERS)` - Returns sum of all numbers in list
5. `contains(HAY,NEEDLE)` - Checks if the search string (NEEDLE) is contained in the string (HAY)
6. `distinct-values(LIST)` - Returns the distinct values from the list

### Hausaufgabe 1

Führen Sie die folgenden Abfragen in der Spark-Shell aus. Als Grundlage für die Abfragen dient das TPC-H Schema. Laden Sie dazu die TPC-H Daten wie in der Vorlesung gezeigt in die Spark-Shell.

- (a) Ermitteln Sie pro Marktsegment die Anzahl der Bestellungen in 1997.
- (b) Ermitteln Sie die Zahl der Kunden und Lieferanten pro Land.
- (c) Ermitteln Sie die Stückzahlen der verschiedenen Bauteile in Deutschland.
- (d) Ermitteln Sie, welche Kunden kein *goldenrod lavender spring chocolate lace* bestellt haben.

### Hausaufgabe 2

Führen Sie die folgenden Abfragen in der Spark-Shell aus. Als Grundlage für die Abfragen dient das TPC-H Schema. Laden Sie dazu die TPC-H Daten wie in der Vorlesung gezeigt in die Spark-Shell.

- (a) Laden Sie die `region.tbl` Datei als DataFrame Objekt in die Spark-Shell.
- (b) Ermitteln Sie die Namen aller Regionen.
- (c) Ermitteln Sie die Zahl der Länder die nicht in Europa liegen.
- (d) Ermitteln Sie die größte Bestellung aus dem Jahr 1996.
- (e) Ermitteln Sie welcher europäische Kunde im Jahr 1996 am meisten Geld ausgegeben hat.

(f) Ermitteln Sie welche Unternehmen keine Kunden in Europa haben.

### Hausaufgabe 3

Formulieren Sie die zuvor in SQL bearbeiteten Anfragen zur Universitätsdatenbank in XQuery. Erstellen Sie insbesondere XQuery-Anfragen, um folgende Fragestellungen zu beantworten:

- a) Suchen Sie die Professoren, die Vorlesungen halten.
- b) Finden Sie die Studenten, die alle Vorlesungen gehört haben.
- c) Finden Sie die Studenten mit der größten Semesterzahl unter Verwendung von Aggregatfunktionen.
- d) Berechnen Sie die Gesamtzahl der Semesterwochenstunden, die die einzelnen Professoren erbringen. Dabei sollen auch die Professoren berücksichtigt werden, die keine Vorlesungen halten.
- e) Finden Sie die Studenten, die alle vierstündigen Vorlesungen gehört haben.
- f) Finden Sie die Namen der Studenten, die in keiner Prüfung eine bessere Note als 3.0 hatten.
- g) Berechnen Sie den Umfang des Prüfungsstoffes jedes Studenten. Es sollen der Name des Studenten und die Summe der Semesterwochenstunden der Prüfungsvorlesungen ausgegeben werden.
- h) Finden Sie Studenten, deren Namen den eines Professors enthalten.
- i) Ermitteln Sie den Bekanntheitsgrad der Professoren unter den Studenten, wobei wir annehmen, dass Studenten die Professoren nur durch Vorlesungen oder Prüfungen kennen lernen.

### Hausaufgabe 4

Schreiben Sie eine Anfrage, die folgendes Ergebnis zurückgibt:

```
<Universitaet>
  <Fakultaet Name="Philosophie" AnzahlAssistenten="3">
    <Professor Name="Sokrates" AnzahlAssistenten="2"/>
    <Professor Name="Russel" AnzahlAssistenten="1"/>
  </Fakultaet>
  <Fakultaet Name="Physik" AnzahlAssistenten="2">
    <Professor Name="Kopernikus" AnzahlAssistenten="2"/>
  </Fakultaet>
  <Fakultaet Name="Theologie" AnzahlAssistenten="1">
    <Professor Name="Augustinus" AnzahlAssistenten="1"/>
  </Fakultaet>
</Universitaet>
```

### Hausaufgabe 5

Datenbanksysteme erlauben JSON-Objekte eingebettet als Attribute in Tabellen. Der zugehörige Syntax ist seit 2017 standardisiert<sup>1</sup> und zum Beispiel in PostgreSQL integriert<sup>2</sup>. Das nachfolgende Statement erstellt eine Hilfstabelle, die einen Ausschnitt des Uni-Schemas als JSON-Objekt enthält (und lässt sich in `hyper-db.de` eingeben).

<sup>1</sup>[https://standards.iso.org/ittf/PubliclyAvailableStandards/c067367\\_ISO\\_IEC\\_TR\\_19075-6\\_2017.zip](https://standards.iso.org/ittf/PubliclyAvailableStandards/c067367_ISO_IEC_TR_19075-6_2017.zip)

<sup>2</sup><https://www.postgresql.org/docs/current/functions-json.html>

```

with uni_json (name, doc) as (values ('VirtU', '{
  "Name": "Virtuelle Universitaet der Grossen Denker",
  "UniLeitung": {"Rektor": "Sokrates", "Kanzler": "Erhard"},
  "Fakultaeten": [
    { "Name": "Philosophie", "Professoren": [
      { "PersNr": 2125, "Name": "Sokrates", "Rang": "C4",
        "Vorlesungen": [ {"VorlNr": 5041, "Titel": "Ethik", "SWS": 4},
          {"VorlNr": 5049, "Titel": "Maeeutik", "SWS": 2},
          {"VorlNr": 4052, "Titel": "Logik", "SWS": 4}]
        }
      ]
    }
  ]
}'))

```

1. Geben Sie in SQL den Namen der jeweils ersten Fakultät in `uni_json` aus.
2. Geben Sie in SQL die Personalnummer (`PersNr`) des ersten Professors der jeweils ersten Fakultät aus.
3. Joinen Sie diese mit der SQL-Relation `pruefen` und `Studenten`, um die Namen aller von ihm geprüften Studenten auszugeben.