

XML-Manipulation in BaseX (Übungen)

Julia Beck | j.beck@ub.uni-frankfurt.de | @j4lib

11.12. und 18.12.2019



Praxislabor Digitale Geisteswissenschaften
Universitätsbibliothek Frankfurt am Main

Willkommen zurück!

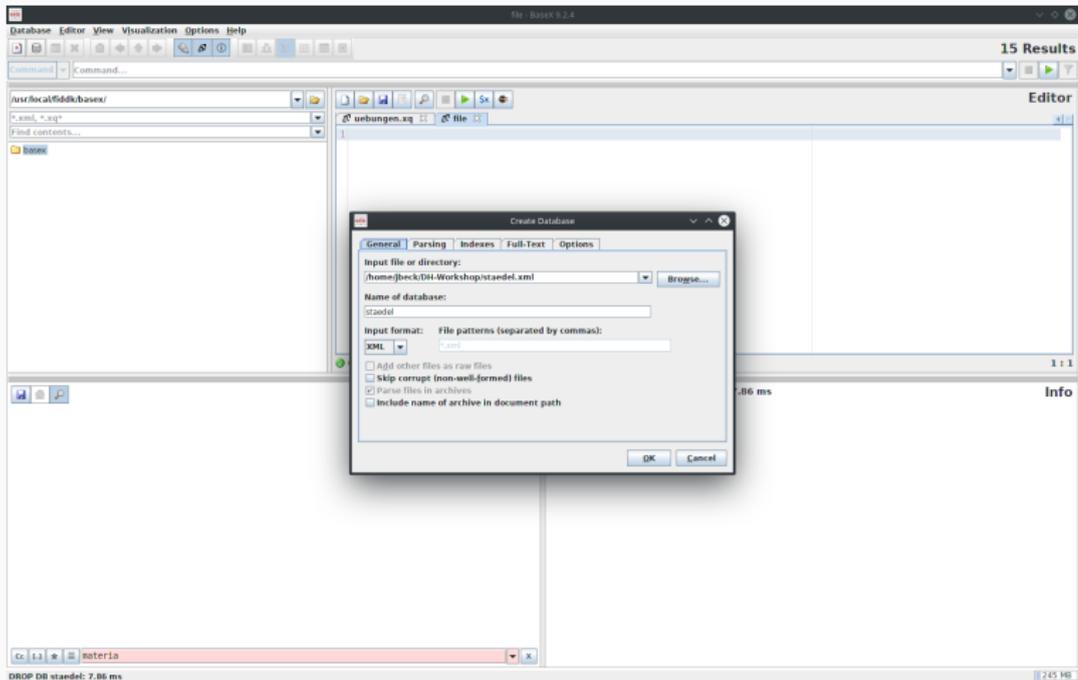
Hat die Installation von BaseX[1] geklappt?

Auf zu den Übungen! Ab Übung 3 werden die Musterlösungen auch als .xq im moodle verfügbar sein. Oft gibt es mehrere Wege zum Ziel.

Übung 1

Legen Sie eine neue Datenbank in BaseX mit dem Namen "staedel" auf Basis der Datei "staedel.xml" an!
Entweder mit Hilfe der GUI oder mit einem Kommando.

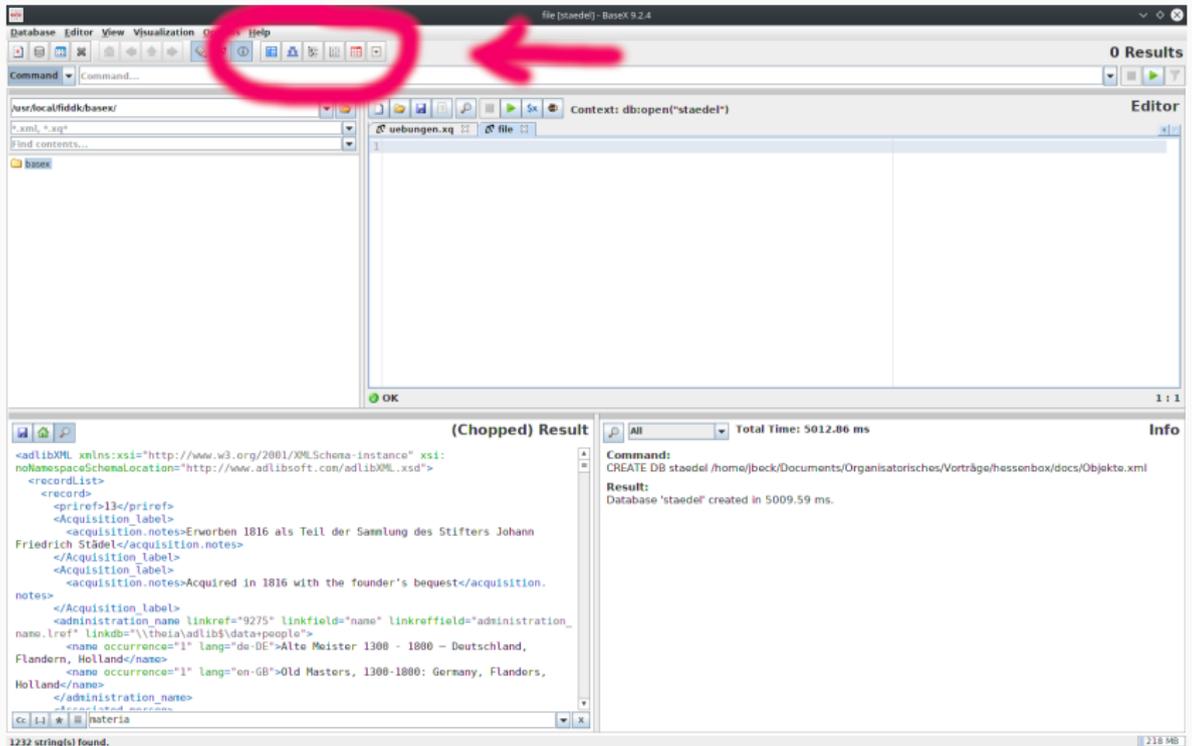
Übung 1 - Lösung



```
db:create("staedel", "/Pfad/zu/staedel.xml", "staedel.xml")
```

Machen Sie sich mit den Daten vertraut,
indem Sie sich die Daten genauer ansehen und die verschiedenen
Visualisierungsmöglichkeiten testen!

Übung 2 - Hilfe

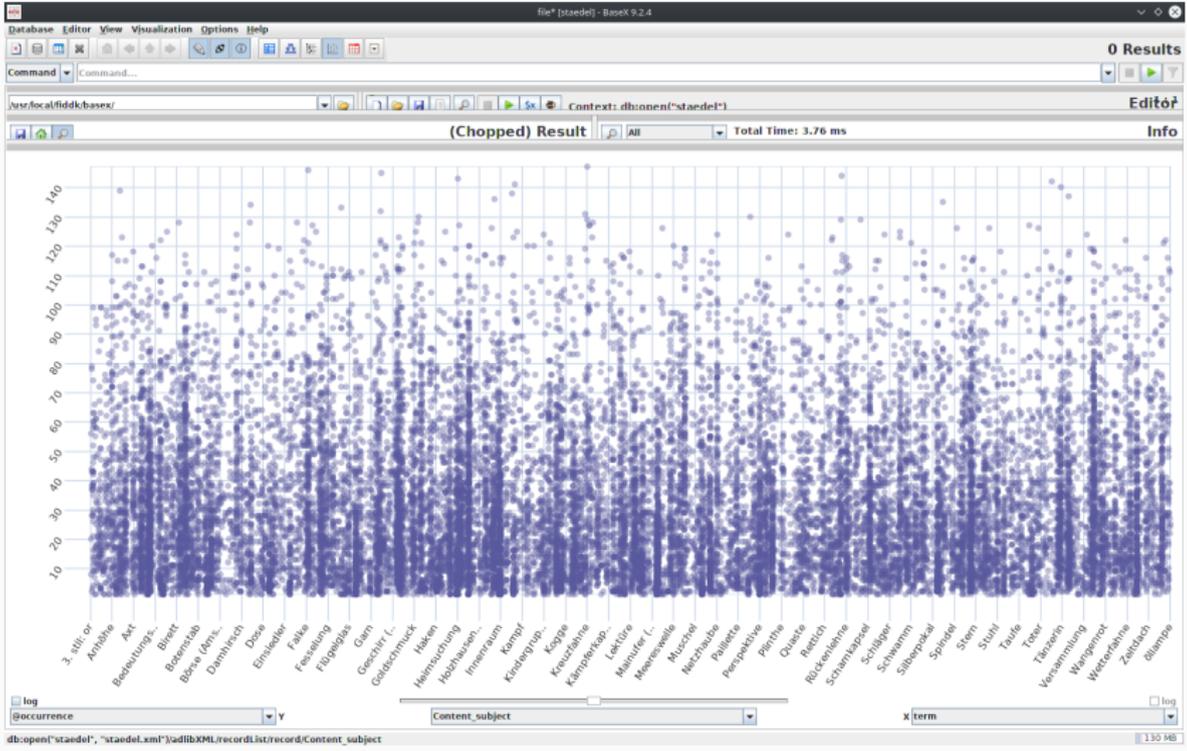


The screenshot shows the BaseX 9.2.4 interface. The top toolbar has a 'Help' menu highlighted with a red circle and a red arrow pointing to it. The main editor area shows a file named 'uebungen.xq' with the context 'db:open('staedel')'. Below the editor, the 'Chopped Result' window displays the following XML:

```
<adlibXML xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:
noNamespaceSchemaLocation="http://www.adlibsoft.com/adlibXML.xsd">
  <recordList>
    <record>
      <prifref>13</prifref>
      <Acquisition_label>
        <acquisition.notes>Erworben 1816 als Teil der Sammlung des Stifters Johann
Friedrich Stäedel</acquisition.notes>
      </Acquisition_label>
      <Acquisition_label>
        <acquisition.notes>Acquired in 1816 with the founder's bequest</acquisition.
notes>
      </Acquisition_label>
      <administration name linkref="9275" linkfield="name" linkreffield="administration_
name.lref" linkdb="\theiaadlib\data/people">
        <name occurrence="1" lang="de-DE">Alte Meister 1300 - 1800 - Deutschland,
Flandern, Holland</name>
        <name occurrence="1" lang="en-GB">Old Masters, 1300-1800: Germany, Flanders,
Holland</name>
      </administration_name>
    </record>
  </recordList>
</adlibXML>
```

The 'Chopped Result' window also shows the command: 'CREATE DB staedel /home/jbeck/Documents/Organisatorisches/Vorträge/hessenbox/docs/Objekte.xml' and the result: 'Database 'staedel' created in 5009.59 ms.' The bottom status bar indicates '1232 string(s) found.' and '218 MB'.

Übung 2 - "Lösung"



Übung 3

- Wie viele Records (<record>) gibt es?
- Wer malte das Bild "Zwei Mädchen"?
- Liste die Titel aller Bilder, die Gemälde aus Nussbaumholz sind!

Übung 3 - Hilfe

- Wie viele Records (`<record>`) gibt es?

Benutze `count(...)`

- Wer malte das Bild "Zwei Mädchen"?

Entweder als XPath oder FLWOR; Finde den Record, wo der Titel "Zwei Mädchen" ist. Gib von diesem Record den Creator zurück.

- Liste die Titel aller Bilder, die Gemälde aus Nussbaumholz sind

Entweder als XPath oder FLWOR; Finde alle Records vom Typ Gemälde, schränke dann ein, dass das Material Nussbaumholz sein soll. Gib von diesen Records jeweils den Titel zurück.

Übung 3 - Lösung

- Wie viele Records (<record>) gibt es?

```
count(db:open("staedel")//record)
```

- Wer malte das Bild "Zwei Mädchen"?

```
db:open("staedel")//record[Title/title/text()  
= "Zwei_Mädchen"]/Production/Creator/name/text()
```

- Liste die Titel aller Bilder, die Gemälde aus Nussbaumholz sind

```
db:open("staedel")//record[Object_name/  
object_name/term[@lang="de-DE"]/text() =  
"Gemälde" and Material/material/term  
[@lang="de-DE"][text() = "Nussbaumholz"]  
/Title/title/text()
```

- Liste die Titel aller Bilder, die Gemälde aus Nussbaumholz sind als FLWOR:

```
for $record in db:open("staedel")//record
where $record/Object_name/object_name/term
    [@lang="de-DE"]/text() = "Gemälde" and
    $record/Material/material/term[@lang =
        "de-DE"]/text() = "Nussbaumholz"
return $record/Title/title/text()
```

- Liste alle deutschen Begriffe für **Atmosphäre** (<Atmosphere>)...
- ... ohne Duplikate...
- ... und sortiere sie!

- Liste alle deutschen Begriffe für **Atmosphäre** (<Atmosphäre>)...
als XPath oder FLWOR
- ... ohne Duplikate...
als XPath oder FLWOR; nutze **distinct-values()** oder **group by**
- ... und sortiere sie!
nur als FLWOR; nutze **order by**

Übung 4 - Lösung

- Liste alle deutschen Begriffe für Atmosphäre (<Atmosphere>)...

```
distinct-values(db:open("staedel")  
//Atmosphere/atmosphere  
/term[@lang="de-DE"]/text())
```

- ... ohne Duplikate...
- ... und sortiere sie!

```
for $atmo in db:open("staedel")//Atmosphere  
/atmosphere/term[@lang="de-DE"]  
group by $atmotext := $atmo/text()  
let $count := count($atmo)  
order by $count descending  
return $atmotext
```

Aufbauend auf Übung 4

- Wie oft kommen die Begriffe jeweils vor? (Welcher Begriff kommt z.B. am häufigsten vor?)
- Probieren Sie das gleiche für "**Content_subject**" oder "**Emotion**", welche Begriffe sind dort besonders häufig vertreten?

- Wie oft kommen die Begriffe jeweils vor? (Welcher Begriff kommt z.B. am häufigsten vor?)

```
for $atmo in db:open("staedel")//Atmosphere
  /atmosphere/term[@lang="de-DE"]
group by $atmotext := $atmo/text()
let $count := count($atmo)
order by $count descending
return $atmotext || ":␣" || $count
```

- Probieren Sie das gleiche für "Content_subject" oder "Emotion", welche Begriffe sind dort besonders häufig vertreten?

```
for $atmo in db:open("staedel")
  //Content_subject/content.subject
  /term[@lang="de-DE"] ...
```

```
for $atmo in db:open("staedel")//Emotion
  /emotion/term[@lang="de-DE"] ...
```

Übung 6

- Füge jedem Record ein **beliebiges** Element hinzu!
- Lösche dieses Element wieder!
- Behalte nur diejenigen Elemente, die eine **bestimmte Sprache** haben (z.B. nur die deutschen oder nur die englischen Bezeichnungen)!

- Füge jedem Record ein beliebiges Element hinzu!

insert node ... into ...

- Lösche dieses Element wieder!

delete node ...

- Behalte nur diejenigen Elemente, die eine bestimmte Sprache haben

Es müssen nicht unbedingt alle sein. Suche zum Beispiel alle deutschen Terme und lösche diese.

Übung 6 - Lösung

- Füge jedem Record ein beliebiges Element hinzu!

```
for $record in db:open("staedel")//record
return insert node <size></size> into $record
```

- Lösche dieses Element wieder!

```
delete node db:open("staedel")//size
```

- Behalte nur diejenigen Elemente, die eine bestimmte Sprache haben; hier nur Beispiellösung für Terme

```
delete node db:open("staedel")//term[@lang
= "en-GB"]
```

- Welches ist das **älteste** Bild?
- Von wem sind die **meisten Bilder** der Sammlung?

- Welches ist das älteste Bild?

```
let $date1 := (for $date in db:open("staedel")
              //Production_date
              /production.date.start/text()
              order by $date ascending
              return $date)[1],
    $record := db:open("staedel")//record
              [Production_date
              /production.date.start/text()
              = $date1]
return $record/Title/title/text() ||
       "␣(" || $date1 || ")"
```

- Von wem sind die meisten Bilder der Sammlung?

```
(for $record in db:open("staedel")//record
group by $creator := $record/Production[1]
  /Creator/name[@lang = "de-DE"]/text()
let $count := count($record)
order by $count descending
return $creator || " mit " || $count
  || " Bildern")[1]
```

Happy Holidays!



Basex.

<http://basex.org/>.



M. Vogelgesang.

Metropolis beamer theme.

Metropolis is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.