

CSE 344 Introduction to Data Management

Section 6: XML

TA: Alok Krishnan (alokak@uw.edu)

- Announcements
 - HW4 due tonight at 11.59pm
 - HW5 posted and is due on 20th February at 11:59pm
- Today
 - Saxon setup
 - Practice with Xml and Xpath

Saxon setup

- Download Saxon:

<http://sourceforge.net/projects/saxon/files/>

- Download the Mondial XML dataset and Mondial DTD

- Make sure that you have Java VM installed



Saxon setup

- You may want to use xmllint for pretty-printing XML

Installing xmllint on Cygwin -- Select the following packages

libxml

Libxslt

- Optional -- use Xray XML editor for editing

http://download.cnet.com/XRay-XML-Editor/3000-7241_4-10166163.html

- Make sure that `saxonhe9.jar`, `mondial.xml` and `mondial.dtd` files are in the same directory as the `query` files

Practice with Saxon

- Open a text editor of your own choice and save the following in q1.xq

```
<result>  
    Xpath expression  
</result>
```

- Run the program

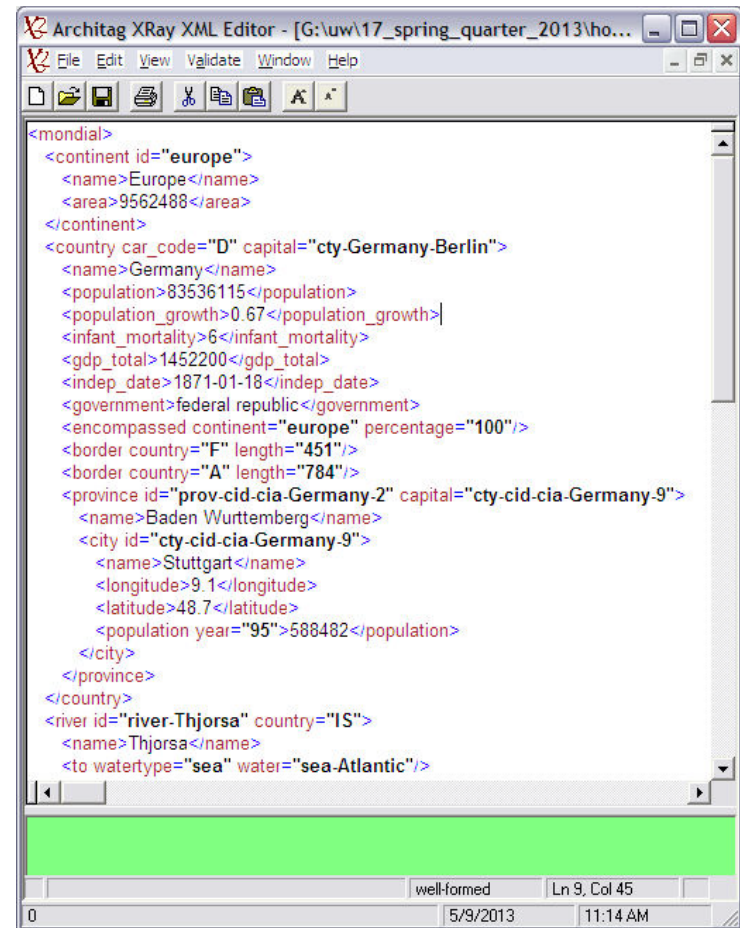
```
$ java -cp saxon9he.jar net.sf.saxon.Query q1.xq
```

- To pretty print, do this:

```
$ java -cp saxon9he.jar net.sf.saxon.Query q1.xq > a1.xml  
$ xmllint --format a1.xml
```

Practice with Xpath

We will use the Mondial database, which you are using in homework 5, due next week. Note that for homework 5, you will use XQuery instead of XPath, but every XPath expression is also an XQuery query. Here is a simplified example of Mondial:



```
<mondial>
  <continent id="europe">
    <name>Europe</name>
    <area>9562488</area>
  </continent>
  <country car_code="D" capital="cty-Germany-Berlin">
    <name>Germany</name>
    <population>83536115</population>
    <population_growth>0.67</population_growth>
    <infant_mortality>6</infant_mortality>
    <gdp_total>1452200</gdp_total>
    <indep_date>1871-01-18</indep_date>
    <government>federal republic</government>
    <encompassed continent="europe" percentage="100"/>
    <border country="F" length="451"/>
    <border country="A" length="784"/>
    <province id="prov-cid-cia-Germany-2" capital="cty-cid-cia-Germany-9">
      <name>Baden Wurttemberg</name>
      <city id="cty-cid-cia-Germany-9">
        <name>Stuttgart</name>
        <longitude>9.1</longitude>
        <latitude>48.7</latitude>
        <population year="95">588482</population>
      </city>
    </province>
  </country>
  <river id="river-Thjorsa" country="IS">
    <name>Thjorsa</name>
    <to watertype="sea" water="sea-Atlantic"/>
  </river>
</mondial>
```

well-formed Ln 9, Col 45
0 5/9/2013 11:14 AM

Practice Questions

1. Give a list of all the countries in XML
2. Give a list of the countries that Germany borders
3. Give the names of all the countries with population at least 10 million
4. Find all cities located in countries that are partially or fully part of Europe (The cities themselves don't have to be in Europe)
5. Find the names of all rivers that start north of the equator (at a positive latitude)
6. Find the names of all rivers that start in Iceland
7. Get the names of all countries in both Asia and Europe

Q1. Give a list of all the countries in XML.

```
<result>  
  { doc("mondial.xml")//country }  
</result>
```


Q2. Give a list of the countries that Germany borders.

```
<result>
```

```
{ doc("mondial.xml")//country[@car_code="D"]/border }
```

```
</result>
```

Q3. Give the names of all the countries with population at least 10 million.

```
<result>  
{ doc("mondial.xml")//country[population/text() >=  
10000000]/name }  
</result>
```

Q4. Find all cities located in countries that are partially or fully part of Europe. (The cities themselves don't have to be in Europe.)

```
<result>
{
  doc("mondial.xml")//country[encompassed/@continent="europe"]//city
}
</result>
```

Q5. Find the names of all rivers that start north of the equator (at a positive latitude).

```
<result>  
{ doc("mondial.xml")//river[source/latitude >  
0.0]/name }  
</result>
```

Q6. Find the names of all rivers that start in Iceland.

```
<result>  
{ doc("mondial.xml")//river[source/@country =  
  (//country[ name='Iceland']/@car_code)]/name }  
</result>
```

Q7. Get the names of all countries in both Asia and Europe.

```
<result>
{
  doc("mondial.xml")//country[encompassed/@continent='europe']
  [encompassed/@continent='asia']/name
}
</result>
```