CSE 5347/7347

Homework Assignment 9

Due: April 19: Distance April 21

XQUERY : Write XQUERY to do the following:

Question 1:

Based on the **persons.xml** file (see end of this assignment and as a file on BB), write an XQuery that uses the **family** element (from persons.xml) to locate all the persons with family name of "**Boss"** and returns their **given** and **family** names separated by space. Your output should look like:

OUTPUT:

```
<persons>
  <found>Big  Boss</found>
</persons>
```

Question 2: (requires two files: persons.xml and names.xml)

Using the persons.xml and the names.xml files, match the **who** element (from names.xml) with the **family** element (from persons.xml). If there is a match, for each match, output a **person** element that contains the name, email and manager name. The manager name can be found in the **manager** attribute of the **link** element. Wrap all person elements in a root element.

Output should look like:

```
<root>
 <person>
   <name>One Worker</name>
   <email> one@oxygenxml.com</email>
   <boss>Big.Boss</boss>
 <person>
 …
etc.
</root>
```

File1: persons.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<personnel>
    <person id="Big.Boss">
        <name>
            <family>Boss</family>
            <given>Big</given>
        </name>
        <email>chief@oxygenxml.com</email>
        <link subordinates="one.worker two.worker three.worker four.worker
five.worker"/>
    </person>
    <person id="one.worker">
        <name>
            <family>Worker</family>
            <given>One</given>
        </name>
        <email>one@oxygenxml.com</email>
        <link manager="Big.Boss"/>
    </person>
    <person id="two.worker">
        <name>
            <family>Worker</family>
            <given>Two</given>
        </name>
        <email>two@oxygenxml.com</email>
        <link manager="Big.Boss"/>
    </person>
    <person id="three.worker">
        <name>
            <family>Worker</family>
            <given>Three</given>
        </name>
        <email>three@oxygenxml.com</email>
        <link manager="Big.Boss"/>
    </person>
    <person id="four.worker">
        <name>
            <family>Worker</family>
            <given>Four</given>
        </name>
        <email>four@oxygenxml.com</email>
        <link manager="Big.Boss"/>
    </person>
    <person id="five.worker">
        <name>
            <family>Worker</family>
            <given>Five</given>
        </name>
        <email>five@oxygenxml.com</email>
        <link manager="Big.Boss"/>
    </person>
</personnel>
```

FILE2: names.xml ---------------------------

```
<names>
  <who>Worker</who>
</names>
```

Do:

- submit PDF to BB showing your XQuery and the Output captured.
- You may use any XQuery engine – Oxygen is recommended.

Notes: On Namespaces and XQUERY

When using Namespaces with XQuery…

If using a default namespace (*xmlns="http://www.e-mis.com/emisopen"*).

There are at least three ways you can handle this (choose the one that best fits your needs):
1. If you do want to refer the elements by name and namespace, declare and use a namespace prefix:
Code: Select all

```
declare namespace emis = "http://www.e-mis.com/emisopen";
for $x in doc("input.xml")/emis:openHealthRecord
return <a>{$x/emis:id}</a>
```

2. If you don't care about the specific namespace, and only want to specify the elements by local name use the "any"(*) namespace prefix:
Code: Select all

```
for $x in doc("input.xml")/*:openHealthRecord
return $x/*:id
```

3. If all the elements you work with (including output) are from the same namespace, you can simply declare the default element namespace at the top of the XQuery and use the rest as it is:
Code: Select all

```
declare default element namespace "http://www.e-mis.com/emisopen";
for $x in doc("input.xml")/openHealthRecord
return $x/id
```

2. and **3.** are the easy way out of this.