XQuery practice

a. From bib.xml, List books published by Addison-Wesley after 1991, including their year and title. You should get:

```
<bib>
   <book year="1994">
      <title>TCP/IP Illustrated</title>
   </book>
   <book year="1992">
      <title>Advanced Programming in the Unix environment</title>
   </book>
</bib>
```

SOLUTION QUERY

```
<bib>
 {
  for $b in /bib/book
  where $b/publisher = "Addison-Wesley" and $b/@year
> 1991
  return
    <book year="{ $b/@year }">
     { $b/title }
    </book>
 }
  </bib>
```

b. List the titles and years of all books in bib.xml published by Addison-Wesley after 1991, in alphabetic order. You should get:

```
<bib>
   <book year="1992">
      <title>Advanced Programming in the Unix environment</title>
   </book>
   <book year="1994">
      <title>TCP/IP Illustrated</title>
   </book>
</bib>
```

SOLUTION QUERY

```
<bib>
 {
  for $b in //book
```

```
    where $b/publisher = "Addison-Wesley" and $b/@year > 1991
    order by $b/title
    return
      <book>
        { $b/@year }
        { $b/title }
      </book>
  }
</bib>
```

c. With root /bib find all titles that contain the word "the", regardless of the level of nesting. You should get:

```
<results>
  <title>Advanced Programming in the Unix environment</title>
    <title>Data on the Web</title>
</results>
```

SOLUTION QUERY

```
<results>
  {
    for $t in /bib//title
    where contains($t/text(), "the")
    return $t
  }
  </results>
```

d. For each book, representing bookstore called “bstore1” [(bib.xml)] and representing bookstore called “bstore2” [(reviews.xml)], list the title of the book and its price from each source. You should get:

```
<books-with-prices>
  <book-with-prices>
    <title>TCP/IP Illustrated</title>
    <price-bstore2>65.95</price-bstore2>
    <price-bstore1>65.95</price-bstore1>
  </book-with-prices>
  <book-with-prices>
    <title>Advanced Programming in the Unix environment</title>
    <price-bstore2>65.95</price-bstore2>
    <price-bstore1>65.95</price-bstore1>
  </book-with-prices>
  <book-with-prices>
    <title>Data on the Web</title>
    <price-bstore2>34.95</price-bstore2>
    <price-bstore1>39.95</price-bstore1>
  </book-with-prices>
</books-with-prices>
```

SOLUTION QUERY

```
<books-with-prices>
 {
  for $b in doc("bib.xml")/bib/book, $a in doc("reviews.xml")//entry
  where $b/title = $a/title
  return
    <book-with-prices>
      { $b/title }
      <price-bstore2>{ $a/price/text() }</price-bstore2>
      <price-bstore1>{ $b/price/text() }</price-bstore1>
    </book-with-prices>
 }
</books-with-prices>
```

e. Under the root /summary_prices (“overview_prices.xml”), find the minimum price for each book, in the form of a "minprice" element with the book title as its title attribute.

```
<results>
  <minprice title="Advanced Programming in the Unix environment">
    <price>65.95</price>
  </minprice>
  <minprice title="TCP/IP Illustrated">
    <price>65.95</price>
  </minprice>
  <minprice title="Data on the Web">
    <price>34.95</price>
  </minprice>
</results>
```

SOLUTION QUERY

```
<results>
  {
    let $doc := doc("prices.xml")
    for $t in distinct-values($doc//book/title)
    let $p := $doc//book[title = $t]/price
    return
      <minprice title="{ $t }">
        <price>{ min($p) }</price>
      </minprice>
  }
  </results>
```

f. For each book under root bib (“bib.xml”) that has at least one author, list the title and first two authors, and an empty "et-al" element if the book has additional authors.

```
<bib>
  <book>
    <title>TCP/IP Illustrated</title>
```

```
      <author>
         <last>Stevens</last>
         <first>W.</first>
      </author>
   </book>
   <book>
      <title>Advanced Programming in the Unix environment</title>
      <author>
         <last>Stevens</last>
         <first>W.</first>
      </author>
   </book>
   <book>
      <title>Data on the Web</title>
      <author>
         <last>Abiteboul</last>
         <first>Serge</first>
      </author>
      <author>
         <last>Buneman</last>
         <first>Peter</first>
      </author>
      <et-al/>
   </book>
</bib>
```

SOLUTION QUERY

```
<bib>
 {
   for $b in doc("bib.xml")//book
   where count($b/author) > 0
   return
      <book>
         { $b/title }
         {
            for $a in $b/author[position()<=2]
            return $a
         }
         {
            if (count($b/author) > 2)
             then <et-al/>
             else ()
         }
      </book>
 }
</bib>
```