<oo>→<dh> Digital humanities

Maintained by: David J. Birnbaum (djbpitt@gmail.com) ![cc-by-nc-sa]
Last modified: 2015-03-23T03:47:54+0000

XQuery assignment #1

Use the 42 Shakespeare plays that have been uploaded to Obdurodon to do the following:

1. Find all of the titles of all of the Shakespeare texts in the corpus. You'll need to read our posting on the main course page on Obdurodon for information about how to address the collection of plays, and also about how to retrieve the full text of one of the plays so that you can look at it and see where the title is, which you'll need to know in order to construct the XPath to retrieve it. The simplest answer is a single XPath expression. The output should look something like (there are 42 of them):

   ```
   <title xmlns="http://www.tei-c.org/ns/1.0">Othello, the Moor of Venice</title>
   <title xmlns="http://www.tei-c.org/ns/1.0">The Second Part of King Henry the Fourth</title>
   <title xmlns="http://www.tei-c.org/ns/1.0">The Taming of the Shrew</title>
   ```

   Here are two important issues:
   - The Shakespeare texts are in the TEI namespace. To use namespaces in XQuery, you declare them as follows:

     ```
     declare namespace tei="http://www.tei-c.org/ns/1.0";
     ```

     This should go on the second line of your XQuery, just below the XQuery declaration. It declares that you will use the prefix `tei:` to refer to elements in the TEI namespace. That means, for example, that you should describe the TEI header as `tei:teiHeader`.
   - There are several `<title>` elements in the plays, and not all of them are titles of plays. Some, for example, may be titles of acts or scenes. You can find the titles of plays by using XPath, and you may want to examine a sample play to remind yourself of how the TEI encodes that type of title.
2. Modify your XPath above to return just the text of the titles, without the tags. You can do that by using `text()` or `data()` or `string()` (which you might want to look up in Kay or at w3schools). Your answer should look something like:

   ```
   Othello, the Moor of Venice
   The Second Part of King Henry the Fourth
   The Taming of the Shrew
   ```

3. Fourteen of the 42 plays have more than 40 unique speakers. Find those plays and return their titles. You will need to use `count()` and `distinct-values()` (and don't forget the TEI namespace!). Find the collection, drill down to the `<TEI>` elements in the collection (you know there are 42 of them), then filter them based on whether or not they contain more than 40 distinct `<speaker>` element values. Once you're getting the 14 plays that meet that description, you can add a path step to retrieve their titles.
4. Modify your solution to the preceding question #3 to return just the text of the play titles, without the `<title>` tags. You can take the same approach as you did for the transition from question #1 to question #2.

Copy and paste your XQuery expressions from eXide into a plain-text document (you can create one in <oXygen/> or in the plain-text editor of your choice) and upload that document as your homework submission. *Do not use Word, which may turn your well-formed straight apostrophes and quotation marks into curly ones, which don't have the same meaning and which will result in invalid XQuery code.* We do not need the results returned by your query; all we need is the XQuery expression itself.