Exercise 2: Creating a Valid TEI Document

1   Learning Outcomes

When you successfully complete the first part of this exercise you should be able to:

- discern the elements and attributes needed for a minimum valid TEI XML file

- associate a TEI XML file with a schema

- have used the TEI namespace

- create a minimum TEI header and text body

- check for both validity and well-formedness

2   Summary

This part of this exercise will walk you through creating a TEI XML file and inserting the work you did previously into it. You'll learn about the required aspects of the `<teiHeader>` and the basic structure of a TEI file.

3   Start a New XML File

Follow the same steps you did for the first exercise to start a new blank XML file. Although we could start a file with a TEI P5 template, for this particular exercise that would be cheating!

- Load up the oXygen XML Editor if it isn't already loaded by using the Windows Start Menu, or double-clicking the icon on the desktop.

- Once the editor has fully loaded from the 'File' menu select 'New' and under 'New Document' select 'XML Document'. This should open up a blank file with an XML Declaration added.

- An XML Declaration looks like:

```
<?xml version="1.0" encoding="UTF-8"?>
```

4   Inserting a <TEI> Element

All TEI files start either with a `<TEI>` element or a `<teiCorpus>` element. In most cases you'll want a `<TEI>` element. These elements have a special psuedo-attribute called 'xmlns' that indicates the namespace a set of elements are from. This is inherited by any elements inside it (unless overridden). This is how we can be sure we're talking about, say, a `<title>` element from the TEI rather than any other schema.

- Add a `<TEI>` element and then add it to the TEI namespace (http://www.tei-c.org/ns/1.0). Maybe add a few blanks line between the starting and closing tag. Your file should look like:

```
 <?xml version="1.0" encoding="UTF-8"?>
<TEI xmlns="http://www.tei-c.org/ns/1.0">


</TEI>
```

- Notice what happens in oXygen and how it helps you input this. Also notice that your file may now have an angry red square rather than a happy green one! Is your file well-formed? (yes, it is!) Why is this red then?

- If it is red it is because your version of oXygen is prepackaged with all sorts of TEI goodness, and in this case it recognises that files starting with **<TEI>** in the TEI namespace are to be associated automatically with a TEI schema that it has stored. It is complaining that you do not have a **<teiHeader>** in your file because all valid TEI files must have this.

5   Adding a **<teiHeader>**

Inside the **<TEI>** element we need to add a **<teiHeader>** element.

- Put the cursor between the starting and closing **<TEI>** element and type in a **<tei-Header>** element. Notice that oXygen provides the closing **</teiHeader>** element. If the correct option is set in oXygen, it understands the TEI schema and knows that certain content is required inside a **<teiHeader>**. It can automatically provide that markup. If not, you'll have to type it in! The resulting file should look like:

```
 <?xml version="1.0" encoding="UTF-8"?>
<TEI xmlns="http://www.tei-c.org/ns/1.0">
<teiHeader>
<fileDesc>
<titleStmt>
<title> </title>
</titleStmt>
<publicationStmt/>
<sourceDesc/>
</fileDesc>
</teiHeader></TEI>
```

- Notice that your file still has an angry red square rather than a happy green square. This is because there is still stuff needed even though you've added some markup. First, add a title of something like "My 'Strange Meeting' document" by adding this text between the starting and closing **<title>** tags. There are other elements which are allowed here in **<titleStmt>** such as **<author>** (Wilfred Owen), **<editor>** (Jon Stallworthy), that you could add but aren't really required for this exercise. You could use the more general **<respStmt>** (with a **<name>** element with your name and a **<resp>** element with something like 'TEI P5 Encoding' in it) to record your own work if you wish, but as with the other embellishments this isn't necessary for this exercise.

- Then add a paragraph **<p>** inside the **<publicationStmt>** with some text to record what this file is for, perhaps something like "An exercise for learning TEI."

- Inside sourceDesc we should add a **<p>** with some text like: "The primary resource of this file is Strange Meeting from Jon Stallworthy's edition, available on the First World War Poetry Digital Archive." To make this even better, we might surround the

title 'Strange Meeting' with a `<ref>` element with a *@target* attribute with a value of 'http://www.oucs.ox.ac.uk/ww1lit/collections/item/3350' because that is URL from which we got this text.

- Your `<teiHeader>` should now look something like:

```
<teiHeader>
 <fileDesc>
  <titleStmt>
   <title>My 'Strange Meeting' document</title>
  </titleStmt>
  <publicationStmt>
   <p>An exercise for learning TEI.</p>
  </publicationStmt>
  <sourceDesc>
   <p> The primary resource of this file is <ref
      target="http://www.oucs.ox.ac.uk/ww1lit/collections/item/3350">Strange
        Meeting</ref> from Jon Stallworthy's edition, available on the First
World
        War Poetry Digital Archive. </p>
  </sourceDesc>
 </fileDesc>
</teiHeader>
```

- Notice that even though this is a complete `<teiHeader>` with all the required aspects, our file as a whole isn't valid.

6 Add a `<text>`

All TEI files, in addition to a `<teiHeader>` with `<fileDesc>` containing a `<titleStmt>`, `<publicationStmt>`, and `<sourceDesc>`, need to follow the header with at least one of: `<sourceDoc>`, `<facsimile>`, or `<text>`. In our case we're going to add a `<text>` element. To do this:

- Add a couple of blank lines after the closing `</teiHeader>`.

- Insert a `<text>` element and inside that a `<body>` element. (The `<text>` element requires a `<body>` element because if you don't have a text body, what are you encoding?)

- The `<text>` section of the file should look something like:

```
<text>
 <body>
<!--We will put our poem here -->
 </body>
</text>
```

7 Adding Our Poem

This is a good start but we need to put something inside the body. Luckily, we have already encoded a poem in the previous exercise, so we can use that!

- With the cursor in between the opening and closing `<body>` tags go to the 'Document' menu on the menu bar, and select 'File', and 'Insert File'. Select the file you saved earlier if you finished the first exercise. If you didn't then in the spoilers directory there is a file called 'ex01.xml' which has the completed first exercise.

- But wait, as soon as you've added this we get a bit of a problem! oXygen will complain that we've got an XML declaration in the middle of our file. Delete this redundant XML declaration!

- Your document should now be valid and have a happy green square in the upper right-hand corner! If it isn't, try to solve the problem by looking at the error message that is provided.

8   Saving Your Work

Let's save our work:

- Have you formatted and indented your work automatically?

- Is your work well-formed? Do you have a happy green square or an angry red one?

- From the 'File' menu select 'Save' or click on the Save icon (looks like an old-style 3.5" disk)

- Save the file using the name 'exercise02.xml' or another name of your choice.

9   Self-Assessment

Check if you understand some of the core principles of this part of the exercise by answering the following questions:

- Which elements and attributes do you need for a minimum valid TEI XML document?

- What three parts of the `<teiHeader>` are required in all TEI conformant documents?

- Where are these elements and attributes allowed?

- What is the function of each element and attribute you've used?

- Why do you think these elements and attributes are required in TEI XML?

10   Next and Further Reading

These exercises should have given you some experience editing XML and making a valid TEI file.

- If you are finished early you may wish to browse through the TEI Guidelines online at http://www.tei-c.org/release/doc/tei-p5-doc/en/html/index-toc.html.

- In particular you might want to look at the Elements appendix of reference pages for individual elements. Consider looking up all the elements you've used in this file to see how they are defined.

- What other elements are allowed inside the `<text>` element? What would you use them for?

- What other parts of the `<teiHeader>` are there? What are they for?

- You may wish to read the chapters on Default Text Structure http://www.tei-c.org/release/doc/tei-p5-doc/en/html/DS.html or Elements Available to All TEI Documents http://www.tei-c.org/release/doc/tei-p5-doc/en/html/CO.html.

- If you haven't already, look up the main elements in the `<teiHeader>` on the TEI-C website and see what they are allowed to contain.

- You could also have a look at the TEI Guidelines chapter of the Header at http://www.tei-c.org/release/doc/tei-p5-doc/en/html/HD.html.