

Exercise 1: Using the oXygen XML Editor

James Cummings – @jamescummings

2016-07-04

1 Learning Outcomes

When you successfully complete this exercise you should be able to:

- mark up an XML declaration
- mark up basic features of a poem
- create a well-formed XML document

2 Summary

This exercise will walk you through creating an XML document in the oXygen editor and introduce a variety of ways to mark this document up. You will first start a new document, then insert some unmarked up text into the editor, and then mark up the stanzas or line-groups (lg) and lines (l). You will learn to check that your document is well-formed or not.

Note: read each step before you do it.

3 Starting A New XML File

Let's start a new XML file by following the following steps:

- 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 you may wish to close any additional popup windows, and optionally the 'views' on the left and right-hand side. This will give you more space.
- From the 'File' menu select 'New' and double click on 'New Document' select 'XML Document' This should open up a blank document with an XML Declaration added.
- An XML Declaration looks like:

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

and the XML declaration in the element tells anything processing your XML file, including the editor, that this is an XML file and what version of XML you are using through the @version attribute. It also conveys which characters the program may expect in attribute @encoding. XML version 1.0 is a W3C recommendation from 2008. UTF-8 (Universal Character Set Transformation Format - 8 bit) contains most characters from all human writing systems. The XML declaration needs no closing tag as it takes the form of a special processing-instruction that starts and ends with an angle-bracket and a question mark.

4 Creating a Division

Let's create a division of a text using the <div> element. This is a generic division or section element.

- On the line below the XML declaration type: <div>

5 INSERTING SOME TEXT

- Notice what happens when you type the final '>'. oXygen is trying to help you and inserts in the closing `</div>` tag. This is because it knows the rules of XML, and knows that if you type an opening `<div>` you are required to have a closing `</div>` sooner or later.
- We haven't said what *type* of division this is, which means it is just a generic division. Let's categorise it as 'verse' by adding a `@type` attribute. (Note: although I say '@type' the attribute's name is 'type'. The at-sign, '@' is used as a convention to denote an attribute in prose.) Move the cursor back until you're just after the letter 'v' in the opening tag. Press space, and then type: `type="` and notice what happens when you type the quotation mark. oXygen is again trying to help you by putting the closing quotation mark, because it knows that attribute values must always be quoted.
- In between the quotation marks type `verse` to categorise our division as being verse.
- Move back until you are directly in between the opening `<div>` and closing `</div>`. Press 'enter' a couple of times to give yourself some space inside the element.
- We are going to put one or more sonnets by William Shakespeare inside the `<div>`. Add a `<head>` element inside the `<div>` with the heading 'My Favourite Sonnets'. Your file should now look like:

```
<div type="verse">
  <head>My Favourite Sonnets</head>
  <!-- We will add the sonnet here -->
</div>
```

5 Inserting Some Text

We are going to use the sonnet by William Shakespeare *Sonnet 18* as an example for this exercise. This uses:

- Sonnet: <http://tei.it.ox.ac.uk/Talks/2016-07-dhoxss/materials/sonnet18.txt>
- Image: <http://tei.it.ox.ac.uk/Talks/2016-07-dhoxss/materials/sonnet18.jpg>
- Make sure your cursor is in-between the opening `<div>` and the closing `</div>` underneath the `<head>` element. Go to <http://tei.it.ox.ac.uk/Talks/2016-07-dhoxss/materials/sonnet18.txt> in your browser, and highlight the entire poem (click, then control-a) and copy (control-c) and then in the oxygen window paste it (control-v).
- Your division should now look like:

```
<div type="verse">
  <head>My Favourite Sonnets</head>
  18.

  Shall I compare thee to a Summer's day?
  Thou art more lovely and more temperate:
  Rough winds do shake the darling buds of May,
  And Summer's lease hath all too short a date:

  Sometime too hot the eye of heaven shines,
```

```
And often is his gold complexion dimm'd,  
And every faire from faire some-time declines,  
By chance, or nature's changing course, vntrimm'd:
```

```
But thy eternall Sommer shall not fade,  
Nor loose possession of that faire thou ow'st;  
Nor shall death brag thou wander'st in his shade,  
When in eternall lines to time thou grow'st,
```

```
So long as men can breathe or eyes can see,  
So long lives this, and this giues life to thee,
```

```
</div>
```

- Now we are going to use the line-group element `<lg>` not only to group the sonnet as a whole but the individual quatrains. Highlight the text of the poem including the '18.' but no other elements. Either press control-e as a shortcut key, or right-click and under 'Refactoring' select 'Surround with Tags'. A box should pop up and type `lg` into it and press enter. Notice how oXygen helps you again by putting the opening tag before what you had highlighted and the close tag afterwards. This should surround your entire sonnet with the `<lg>` element. Add a `@type` attribute to this `<lg>` with the value of 'sonnet'.
- Use the same surround-with-tag shortcut to highlight and surround '18.' with a `<head>` element
- It should now look like:

```
<div type="verse">  
  <head>My Favourite Sonnets</head>  
  <lg type="sonnet">  
    <head>18.</head>  
    Shall I compare thee to a Summer's day?  
    Thou art more louely and more temperate:  
    Rough windes do shake the darling buds of Maie,  
    And Sommers lease hath all too short a date:  
  
    <!-- rest of poem -->  
  </lg>  
</div>
```

6 Encoding the lines (using surround-with-tag and split element)

- The surround-with-tag shortcut is really useful. There is another one called split-element which will split the current element that you are inside ending the current element and starting a new one where the cursor is. This can be used to mark up lots of elements very quickly. In our case we want to wrap each of the lines in the metrical line element `<l>`.
- Highlight from the first word 'Shall' all the way down to the last word 'thee,' (yes, the last punctuation character in the sonnet is a misprinted comma) then use the control-e surround-with-tag shortcut to wrap the whole sonnet in a single `<l>` element. This means that our encoding is claiming that the whole sonnet is one metrical line. This is clearly false.
- Place the cursor at the start of the second line, just before 'Thou' and press alt-shift-D to split the line element that you are in. This will put `</l><l>` at your current position.

Don't worry about the closing tag being on the wrong line for now. After you've done the first couple lines it should look like:

```
<div type="verse">
  <head>My Favourite Sonnets</head>
  <lg type="sonnet">
    <head>18.</head>
    <l> Shall I compare thee to a Summer's day?
    </l>
    <l>Thou art more louely and more temperate:
    </l>
    <l>Rough windes do shake the darling buds of Maie,
      And Sommers lease hath all too short a date:

  <!-- rest of poem -->
  </l>
</lg>
</div>
```

Do the rest of the poem.

7 Format and Indent

Our poem is marked up, but some of the markup might be a bit messy because of the split-element shortcut we used.

- Make sure that your file is 'well-formed'. You'll be able to tell it is well-formed because oXygen will have a *happy green square* in the upper right-hand corner. If it is an *angry red square*, you better find the problem (where a red bar on the right-hand side is) and correct the mistake! (Note: if you are red/green colour blind, ask me and I'll show you how to change this to colours you can differentiate between!)
- Now let's format and indent our file. This tidies up some of the whitespace and indents elements based on their place in the hierarchy. Either select the 'Format and Indent' icon from the toolbar (it looks like some indented lines), or go to the menus: 'Document' -> 'Source' -> 'Format and Indent'.
- Formatting and indenting your markup is not necessary, it could all be on one big long line, but it makes it much easier for people to read.
- You can delete the extra space at the end of each line, but it is optional.

8 Adding the stanzas

Now we have each `<l>` marked up we want to mark up the stanzas. A shakespearean sonnet is made of three quatrains and a final couplet.

- Around each set of four lines and the last two wrap an `<lg>` element. Use the surround-with-tag shortcut again.
- Give each `<lg>` a `@type` attribute with the values of 'quatrain' or 'couplet' (for the last one).
- Your poem should now look like:

```

<div type="verse">
  <head>My Favourite Sonnets</head>
  <lg type="sonnet">
    <head>18.</head>
    <lg type="quatrain">
      <l>Shall I compare thee to a Summer's day?</l>
      <l>Thou art more louely and more temperate:</l>
      <l>Rough windes do shake the darling buds of Maie,</l>
      <l>And Sommers lease hath all too short a date:</l>
    </lg>
    <lg type="quatrain">
      <l>Sometime too hot the eye of heauen shines,</l>
      <l>And often is his gold complexion dimm'd,</l>
      <l>And euey faire from faire some-time declines,</l>
      <l>By chance, or nature's changing course, vntrimm'd:</l>
    </lg>
    <lg type="quatrain">
      <l>But thy eternall Sommer shall not fade,</l>
      <l>Nor loose possession of that faire thou ow'st;</l>
      <l>Nor shall death brag thou wander'st in his shade,</l>
      <l>When in eternall lines to time thou grow'st,</l>
    </lg>
    <lg type="couplet">
      <l>So long as men can breathe or eyes can see,</l>
      <l>So long liues this, and this giues life to thee,</l>
    </lg>
  </lg>
</div>

```

9 Adding more markup

We've marked up the structure of this sonnet, but not much else. Let's add more markup.

- The `<rhyme>` element has a `@label` attribute and is used to mark up the rhyme word of each line and what letter it is in the rhyme scheme. Shakeapearean sonnets usually have an **abab cdcd efef gg** rhyme scheme. Using the surround-with-tag shortcut highlight the last word in a line and wrap the `<rhyme>` element around it. Then add a `@label` attribute for the part it plays in the rhyme scheme. For example the first quatrain should look like:

```

<lg type="quatrain">
  <l>Shall I compare thee to a Summer's <rhyme label="a">day</rhyme>?</l>
  <l>Thou art more louely and more <rhyme label="b">temperate</rhyme>:</l>
  <l>Rough windes do shake the darling buds of <rhyme label="a">Maie</rhyme>,</l>
  <l>And Sommers lease hath all too short a <rhyme label="b">date</rhyme>:</l>
</lg>

```

Note how we've not included the punctuation in the `<rhyme>` element. (This is just an editorial decision.)

- What about the original rendering of the sonnet? If you look at : <http://tei.it.ox.ac.uk/Talks/2016-07-dhoxss/materials/sonnet18.jpg> you'll see that the first 'S' is a drop capital, and the last two lines are indented.
- Surround the first 'S' and wrap an `<hi>` element (for general highlighting) and give this a `@rend` attribute value of 'dropcap'.
- For the final two lines, add a `@rend` attribute to each one with a value of 'indent'.

10 The result

In the end your file should look something like:

```
<div type="verse">
  <head>My Favourite Sonnets</head>
  <lg type="sonnet">
    <head>18.</head>
    <lg type="quatrain">
      <l>
        <hi rend="dropcap">S</hi>hall I compare thee to a Summer's <rhyme la-
bel="a">day</rhyme>?</l>
        <l>Thou art more louely and more <rhyme label="b">temperate</rhyme>:</l>
        <l>Rough windes do shake the darling buds of <rhyme label="a">Maie</rhyme>,</l>
        <l>And Sommers lease hath all too short a <rhyme label="b">date</rhyme>:</l>
      </lg>
    <lg type="quatrain">
      <l>Sometime too hot the eye of heauen <rhyme label="c">shines</rhyme>,</l>
      <l>And often is his gold complexion <rhyme label="d">dim'd</rhyme>,</l>
      <l>And every faire from faire some-time <rhyme label="c">declines</rhyme>,</l>
      <l>By chance, or nature's changing course, <rhyme label="d">vntrimm'd</rhyme>:</l>
    </lg>
    <lg type="quatrain">
      <l>But thy eternall Sommer shall not <rhyme label="e">fade</rhyme>,</l>
      <l>Nor loose possession of that faire thou <rhyme label="f">ow'st</rhyme>;</l>
      <l>Nor shall death brag thou wander'st in his <rhyme label="e">shade</rhyme>,</l>
      <l>When in eternall lines to time thou <rhyme label="f">grow'st</rhyme>,</l>
    </lg>
    <lg type="couplet">
      <l rend="indent">So long as men can breathe or eyes can <rhyme la-
bel="g">see</rhyme>,</l>
      <l rend="indent">So long liues this, and this giues life to <rhyme la-
bel="g">thee</rhyme>,</l>
    </lg>
  </lg>
</div>
```

If you didn't manage to finish the exercise in time, don't worry, there is a pre-completed version at: [view-source:http://tei.it.ox.ac.uk/Talks/2016-07-dhoxss/spoilers/ex01-spoiler.xml?style=raw](http://tei.it.ox.ac.uk/Talks/2016-07-dhoxss/spoilers/ex01-spoiler.xml?style=raw) or <http://tei.it.ox.ac.uk/Talks/2016-07-dhoxss/spoilers/ex01-spoiler.xml?style=raw> (and then right-click to view source).

11 Checking well-formedness?

We are not using a particular schema at the moment, so we can't check validity, but we can check whether it is well-formed or not. That is, if our file follows the rules of XML. If you have a happy green square in the upper right-hand corner of your oxygen scroll-bar then you are well-formed! If not, and you have an angry red square, you must look at the error message at the bottom, at what is underlined in red, and try to solve the problem. (We can help you!) If you are well-formed then delete the first part of any closing tag and see the error produced. Once you are comfortable you know how to tell if your document is well-formed or not, ensure that it is and that you have a happy green square.

12 Saving Your Work

Let's save our work:

- 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 'exercise01.xml' or another name of your choice.

13 Self-Assessment

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

- How do you start a new XML document in oXygen?
- What is an XML declaration?
- What is a well-formed document?
- How do I 'Surround with tag' and repeat that action quickly?
- Why might using the 'Split element' approach be useful?
- What is the function of each element and attribute in your current file? Why have we encoded it this way?
- What is the advantage of formatting and indenting your markup?

14 Next?

Your XML file may be well-formed but it is not yet **valid** because it doesn't validate against a particular schema (such as those which are customisations of the TEI). Next we will have a short introduction to the structure of TEI documents and some of the most frequently-used elements. 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.html>. In particular you might want to look at the *Elements* appendix for reference pages for individual elements. Consider looking up all the elements you've used in this file to see how they are defined.