XML Lab

1) Create a Valid XML document

Open a text editor like Notepad++ and create an XML document which stores the following data about films. It's a good idea to save the XML to a file on your drive (e.g films.xml) before copying the text to the web link below.

FILMS:

Title	Classification	Stars	Genre
Joker	R	Joaquin Phoenix Robert De Niro	Crime Drama Thriller
Singing in the Rain	G	Gene Kelly Donald O'Connor Debbie Reynolds	Comedy Musical Romance
The Lion King	PG	Donald Glover Beyonce Seth Rogen	Animation Adventure Musical

Create your XML document following this template:

```
<?xml version="1.0" encoding="UTF-8"?>
 <films>
   <film>
     <title>…</title>
     …
  </film>
</films>
```

Check your XML
Use a web browser to open the following page: https://www.w3schools.com/xml/xml_validator.asp

Scroll down the page to the "Syntax-check Your XML" section and copy/paste your XML to validate it. If there is an error in the XML then check the debug information, and correct the XML.

Once you have completed this task you can close this webpage and text editor (e.g. Notepad++).

2) XPath

We’ll now take a look at using XPath to filter an XML document with information about music albums as follows:

Artist	Album Title	Genre	Released date	Bookmarked Tracks
Bill Evans	The Bill Evans Album	Jazz	1971	Sugar Plum Waltz for Debby
Queen	Jazz	Rock	1978	Don't Stop Me Now
Jamiroquai	Travelling Without Moving	Acid Jazz	1996	Virtual Insanity Cosmic Girl High Times
Muse	Drones	Rock	2015	Reapers The Handler

This dataset is available to download in XML from here: www.macs.hw.ac.uk/~pb56/f21df.xml

Take a look at the XML document structure:

```
<?xml version="1.0" encoding="UTF-8"?>
<albums>
   <album>
     <artist>…</artist>
     …
  </album>
</albums>
```

Copy across the XML data and write your XPath answers to the following questions here: https://www.freeformatter.com/xpath-tester.html

1. Get all the album titles
2. Show the artist for the last album on the list
3. Show all the details for the 'Jazz' genre albums
4. Show the album title for those released before 1980 (not inclusive of 1980)
5. Show the bookmarked tracks for all 'Rock' genre albums
6. Show a COUNT how many bookmarked tracks there are in total
7. Show the album title for records where 2 or more tracks have been bookmarked
8. Show a list of the album titles which include the word 'Album' in the title
9. Show a list of the album title and artist with the 'Rock' genre

HINTS:

Check the *lecture notes* for examples of XPath also check this *link*: https://www.w3schools.com/xml/xpath_syntax.asp

3) XSLT

Use the XML file from task 2 (music library) in conjunction with the following XSLT file. Copy both files (XML document and XSLT code) to the relevant sections on this link (clearing the text already in each box):

https://tiny.cc/f21df expands to: https://www.w3schools.com/xml/tryxslt.asp?xmlfile=cdcatalog&xsltfile=cdcatalog_if

The following XSLT can be downloaded from: http://www.macs.hw.ac.uk/~pb56/df_xslt.txt

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
  <html>
  <body>
  <h2>Albums</h2>
    <table border="1">
    <tr bgcolor="#9acd32">
     <th>Artist</th>
        <th>Album Title</th>
        <th>Genre</th>
        <th>Release Date</th>

      </tr>
      <xsl:for-each select="/albums/album">
      <tr>
     <td><xsl:value-of select="artist"/></td>
     <td><xsl:value-of select="album_title"/></td>
     <td><xsl:value-of select="genre"/></td>
     <td><xsl:value-of select="released_date"/></td>

      </tr>
      </xsl:for-each>
    </table>
  </body>
  </html>
</xsl:template>
</xsl:stylesheet>
```

1) Add a new column to the table which displays the number (count) of bookmarked tracks per album by editing the XSLT

2) Update your XSLT to sort the table by the ARTIST name (A-Z)

3) Update your XSLT to shade all pre-1980 albums with a 'yellow' background across the row

   HINTS: - use xsl:choose (check with Google for more details)
   - 'less than' in xsl requires you to use **&lt;**

4) DTD + XSchema

We'll now consider storing details for employees in XML. First by defining an XML Schema, and ending up by creating an XML document that matches the schema.

1) Build a simple XML schema of an employee. The employee element of the schema needs to contain the following elements: job title, employee number, phone and email address. Assume there may be multiple phone numbers

2) Give the document type definition (DTD) for the schema you have created.

3) Extend the schema to restrict the employee number to be less than 1000, restrict the job title to be one of the following: prof, associate prof or assistant prof.

4) Make an XML document that conforms to your schema.

==========END OF THE LAB==========