

Transformación XSLT

Realiza los siguientes ejercicios de XSLT a partir del documento *infoAfuera.xml* a través de XML Copy Editor.

Ejercicio 1.- Muestra los nombres de todos los ciclos.

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">

  <xsl:template match="ciclo">

    <xsl:value-of select="nombre"/>

  </xsl:template>

</xsl:stylesheet>
```

Ejercicio 2.- Muestra en párrafos cada uno de los nombres de los ciclos insertados en un documento HTML.

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">

  <xsl:template match="/">

    <html>

      <body>

        <xsl:apply-templates/>

      </body>

    </html>

  </xsl:template>
```

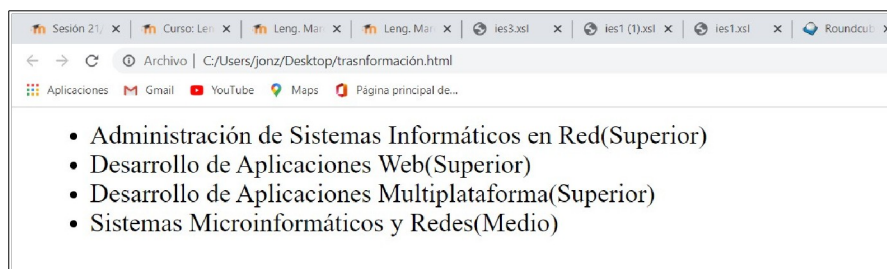
```
  <xsl:template match="ciclo">

    <p><xsl:value-of select="nombre"/></p>

  </xsl:template>

</xsl:stylesheet>
```

Ejercicio 3.- Muestra en una lista sin numerar los nombres de los ciclos y entre paréntesis su grado, insertados en un documento HTML.



```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">

  <xsl:template match="/">

    <html>

      <body>

        <list>

          <ul>

            <xsl:apply-templates/>

          </ul>

        </list>

      </body>

    </html>

  </xsl:template>

  <xsl:template match="ciclo">

    <li><xsl:value-of select="nombre"/>(<xsl:value-of select="grado"/>)</li>

  </xsl:template>

</xsl:stylesheet>
```

Ejercicio 4.- Muestra en una lista numerada los nombres de los ciclos ordenados alfabéticamente de manera descendente y entre paréntesis su grado.

- La etiqueta *xsl:sort* para ordenar elementos
- Puedes aplicar Plantillas de forma recursiva como en el siguiente ejemplo:

```
<xsl:template match="raiz">
...
<xsl:apply-templates select="padre"/>
...
</xsl:template>

<xsl:template match="padre">
...
<xsl:apply-templates select="hijos"/>
...
</xsl:template>

<xsl:template match="hijos">
...
</xsl:template>
```

**En este ejemplo, desde la raíz aplicamos ciertos valores o condiciones a la plantilla “Padre”, que a su vez aplica ciertos valores o condiciones a la plantilla “hijos”*

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:template match="/">
    <html>
    <body>
    <list>
    <ol>
    <xsl:apply-templates/>
    </ol>
    </list>
    </body>
    </html>
  </xsl:template>

  <xsl:template match="ciclos">
    <xsl:apply-templates select="ciclo"><xsl:sort select="nombre" order="descending"/></xsl:apply-templates>
  </xsl:template>

  <xsl:template match="ciclo">
    <li><xsl:value-of select="nombre"/>(<xsl:value-of select="grado"/>)</li>
  </xsl:template>
</xsl:stylesheet>
```