



## TAREA 05 - LMSGI



ANTONIO SANZ PANS  
IES MAESTRE DE CALATRAVA

## JUSTIFICACIÓN DE LAS DECISIONES TOMADAS EN LA TRANSFORMACIÓN

Lo primero que hice al empezar a generar el archivo .xslt fue añadir las etiquetas

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:output method="html" />
```

Después empecé creando la estructura básica del HTML que quería formar, generando el <body> que contendría la etiqueta <table> como si de un HTML se tratase.

Como se pedía en el primero de los puntos, para añadir el header de la tabla, utilicé las etiquetas <tr><th></th></tr> generando una estructura para cada uno de los campos necesarios: Código, Nombre, Año de nacimiento, Movimiento, Género, País y Página web.

```
<tr>
  <th>Código</th>
  <th>Nombre</th>
  <th>Año de nacimiento</th>
  <th>Movimiento</th>
  <th>Género</th>
  <th>País</th>
  <th>Página web</th>
</tr>
```

A continuación hice lo mismo para el cuerpo de la tabla, donde estaría todo el contenido, pero en vez de usar las etiquetas <tr> y <th> utilicé <tr> y <td>, añadiendo dentro de cada <td> los valores correspondientes con la estructura <xsl:value-of select="valor" />

```
<xsl:value-of select="@cod" />
<xsl:value-of select="nombreCompleto" />
<xsl:value-of select="nacimiento" />
<xsl:value-of select="movimiento" />
<xsl:value-of select="pais" />
```

Para los valores que necesitaban un tratamiento especial:

Género: Utilicé la etiqueta <xsl:choose> para poder indicarle que si el campo para género estaba vacío, lo cambiase por "Varios".

```
<xsl:choose>
  <xsl:when test="genero">
    <xsl:value-of select="genero" />
  </xsl:when>
  <xsl:otherwise>
    <xsl:text>Varios</xsl:text>
  </xsl:otherwise>
</xsl:choose>
```

En caso del enlace con la ficha completa (Página web), dentro del td añadí un <a href=""> con el nombre del campo que quería insertar en el href y como texto para que se mostrase "Saber más"

```
<a href="{fichaCompleta}">Saber más</a>
```

Para mostrar todas las filas de la tabla, utilicé un bucle foreach que filtraría por el país deseado, en este caso España

```
<xsl:for-each select="artistas/artista[pais='España']">
```

Y para ordenar a los artistas utilice la etiqueta <xsl:sort> indicando que lo hiciera por “movimiento”

```
<xsl:sort select="movimiento"/>
```