

Harry Kokal Kokal

1 ASIR B Lenguajes de Marcas

Ejercicio 1

Este es el documento xsl

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
3   <xsl:output method="html" indent="yes"/>
4   <xsl:template match="/">
5     <!--Harry Kokal Kokal 1ASIR B-->
6     <html>
7       <head>
8         <meta charset="UTF-8"/>
9         <title>Top 3 de libros con más puntuación</title>
10        <link rel="stylesheet" href="estilos.css"/>
11      </head>
12      <body>
13        <h1>Top 3 de libros con más puntuación</h1>
14        <h2>Modo Tabla</h2>
15        <table>
16          <tr>
17            <th class="negro-blanco">Posición</th>
18            <th class="negro-blanco">Titulo</th>
19            <th class="negro-blanco">Puntos</th>
20          </tr>
21          <!--Inicia un bucle sobre todos los elementos libro-->
22          <xsl:for-each select="//libro">
23            <!--Ordena los elementos libro por el valor de la etiqueta puntos en orden descendente (de mayor a menor)-->
24            <xsl:sort select="puntos" order="descending"/>
25            <!--Dentro del bucle, verifica si la posición del libro es menor o igual a 3. Solo se cojeran los tres primeros libros después de ordenar-->
26            <xsl:if test="position() &lt;= 3">
27              <tr>
28                <!-- Muestra la posición del libro en la tabla-->
29                <td><xsl:value-of select="position()"/></td>
30                <!--Muestra el titulo del libro-->
31                <td><xsl:value-of select="titulo"/></td>
32                <!--Muestra la puntuación del libro-->
33                <td><xsl:value-of select="puntos"/></td>
34              </tr>
35            </xsl:if>
36          </xsl:for-each>
37        </table>
38
39        <!--Harry Kokal Kokal 1ASIR B-->
40        <h2>Modo Lista</h2>
41        <dl>
42          <!--Inicia un bucle sobre todos los elementos libro-->
43          <xsl:for-each select="//libro">
44            <!--Ordena los elementos libro por el valor de la etiqueta puntos en orden descendente (de mayor a menor)-->
45            <xsl:sort select="puntos" order="descending"/>
46            <!--Dentro del bucle, verifica si la posición del libro es menor o igual a 3. Solo se cojeran los tres primeros libros después de ordenar-->
47            <xsl:if test="position() &lt;= 3">
48              <!-- Imprime la posición, el titulo y los puntos del elemento libro-->
49              <dd class="negro-blanco"><xsl:value-of select="position()"/> -
50                <xsl:value-of select="titulo"/> --> Puntos:
51                <xsl:value-of select="puntos"/></dd>
52            </xsl:if>
53          </xsl:for-each>
54        </dl>
55      </body>
56    </html>
57  </xsl:template>
</xsl:stylesheet>
```

Este es el css

```
1  h1{
2    padding: 5px 10px;
3    color: white;
4    background-color: black;
5    text-align: center;
6  }
7
8  h2{
9    color: white;
10   background-color: crimson;
11   text-align: center;
12 }
13
14 table{
15   margin: 0px;
16   border-collapse: collapse;
17   margin: auto;
18 }
19
20 td{
21   border: 1px solid black;
22   text-align: center;
23   padding: 5px 10px;
24 }
25
26 dl{
27   text-align: center;
28 }
29
30 .negro-blanco{
31   color: white;
32   background-color: black;
33 }
34
35 dd:nth-child(2n){
36   color: white;
37   background-color: green;
38 }
```

Este es el resultado

Top 3 de libros con más puntuación

Modo Tabla

Posición	Título	Puntos
1	La vida está en otra parte	5
2	Reina Roja	4.8
3	El señor de los anillos. El retorno del rey	4.75

Modo Lista

1 - La vida está en otra parte -> Puntos: 5

2 - Reina Roja -> Puntos: 4.8

3 - El señor de los anillos. El retorno del rey -> Puntos: 4.75

Ejercicio 2

Este es el documento xsl

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
3   <xsl:output method="html" indent="yes"/>
4   <xsl:template match="/">
5     <!--Harry Kokal Kokal IASIR B-->
6     <html>
7       <head>
8         <meta charset="UTF-8"/>
9         <title>Estadísticas</title>
10        <link rel="stylesheet" href="estilos.css"/>
11      </head>
12      <body>
13        <h1>Estadísticas</h1>
14        <dl>
15          <!--Inicia un bucle que selecciona todos los elementos biblioteca-->
16          <xsl:for-each select="//biblioteca">
17            <!--Para cada elemento biblioteca, cuenta el número de elementos libro dentro de ella y muestra este valor como Total Libro-->
18            <dd>Total Libro: <xsl:value-of select="count(libro)"/></dd>
19            <!--Calcula la puntuación media de todos los libros dentro de la biblioteca actual.Utiliza las funciones sum y count
20            junto con una variable para almacenar el resultado y luego lo muestra con un formato específico-->
21            <dd>Puntuación Media: <xsl:variable name="total_puntos" select="sum(libro/puntos)"/>
22            <xsl:variable name="total_libros" select="count(libro)"/>
23            <xsl:value-of select="format-number($total_puntos div $total_libros, '0.0')"/></dd>
24            <!--Cuenta los libros cuya fecha de publicación está en la década de los 60 y calcula el porcentaje en relación con el total
25            de libros en la biblioteca actual-->
26            <dd>Libros de la década de los 60: <xsl:value-of select="count(libro/fechaPublicacion[@año &gt;= 1960 and @año < 1970])"/>
27            (<xsl:value-of select="format-number((count(libro/fechaPublicacion[@año &gt;= 1960 and @año < 1970]) div count(libro)) * 100, '0.0')"/>%</dd>
28            <!--Para la década de los 70-->
29            <dd>Libros de la década de los 70: <xsl:value-of select="count(libro/fechaPublicacion[@año &gt;= 1970 and @año < 1980])"/>
30            (<xsl:value-of select="format-number((count(libro/fechaPublicacion[@año &gt;= 1970 and @año < 1980]) div count(libro)) * 100, '0.0')"/>%</dd>
31            <!--Para la década de los 80 o mayor-->
32            <dd>Libros de la década de los 80 o posterior: <xsl:value-of select="count(libro/fechaPublicacion[@año &gt;= 1980])"/>
33            (<xsl:value-of select="format-number((count(libro/fechaPublicacion[@año &gt;= 1980]) div count(libro)) * 100, '0.0')"/>%</dd>
34          </xsl:for-each>
35        </dl>
36      </body>
37    </html>
38  </xsl:template>
39 </xsl:stylesheet>
```

Este es el css

```
1 h1{
2   padding: 5px 10px;
3   color: white;
4   background-color: black;
5   text-align: center;
6 }
7
8 dl{
9   font-weight: bold;
10 }
```

Este es el resultado

Estadísticas
Total Libro: 15
Puntuación Media: 4.0
Libros de la década de los 60: 2 (1,3%)
Libros de la década de los 70: 3 (2,0%)
Libros de la década de los 80 o posterior: 5 (3,3%)

Ejercicio 3

Este es el documento xsl

```
5 <!--Harry Kokal Kokal IASIR B-->
6 <html>
7   <head>
8     <meta charset="UTF-8"/>
9     <title>Autor y Libros Escritos</title>
10    <link rel="stylesheet" href="estilos.css"/>
11  </head>
12  <body>
13    <h1>Autor y Libros Escritos</h1>
14    <table>
15      <tr>
16        <th class="negro-blanco">Autor</th>
17        <th class="negro-blanco">Total</th>
18        <th class="negro-blanco">Total (%)</th>
19      </tr>
20      <!--Crea una variable llamada $libros que contiene todos los nodos libro en el documento-->
21      <xsl:variable name="libros" select="//libro" />
22      <!--Inicia un bucle for-each que itera sobre los libros y selecciona solo aquellos cuyo autor no coincide con el autor
23      de su hermano anterior (es decir, el primer libro de cada autor)-->
24      <xsl:for-each select="$libros[not(author = preceding-sibling::libro/author)]">
25        <!--Ordena los libros en orden descendente según la cantidad de libros del autor actual-->
26        <xsl:sort select="count($libros[author = current()/author])" order="descending"/>
27        <!--Crea una variable $autor que almacena el valor del autor actual-->
28        <xsl:variable name="autor" select="author"/>
29        <!--Crea una variable $totalLibrosAutor que almacena la cantidad total de libros del autor actual-->
30        <xsl:variable name="totalLibros" select="count($libros[author = $autor])" />
31        <!--Crea una variable $porcentaje que almacena el porcentaje de libros del autor actual con respecto al total de libros-->
32        <xsl:variable name="porcentaje" select="format-number(($totalLibros div count($libros)) * 100, '0')"/>
33        <tr>
34          <!--Muestra el autor-->
35          <td><xsl:value-of select="$autor"/></td>
36          <!-- Muestra el valor del autor-->
37          <td><xsl:value-of select="$totalLibros"/></td>
38          <!--Muestra el porcentaje de libros del autor con respecto al total de libros-->
39          <td><xsl:value-of select="$porcentaje"/>%</td>
40        </tr>
41      </xsl:for-each>
42    </table>
43  </body>
44 </html>
45 </xsl:template>
```

Este es el css

```
1 h1{
2   padding: 5px 10px;
3   color: white;
4   background-color: black;
5   text-align: center;
6 }
7
8 table{
9   margin: 0px;
10  border-collapse: collapse;
11  margin: auto;
12 }
13
14 td{
15   border: 1px solid black;
16   text-align: center;
17   padding: 5px 10px;
18 }
19 .negro-blanco{
20   color: white;
21   background-color: black;
22 }
```

Este es el resultado

Autor y Libros Escritos

Autor	Total Total (%)	
Juan Gómez Jurado	3	20%
J.R.R Tolkien	3	20%
Mario Vargas Llosa	2	13%
Milan Kundera	1	7%
Gabriel García Márquez	1	7%
GFederico García Lorca	1	7%
Anthony Burgess	1	7%
Richard Dawkins	1	7%
George R.R. Martin	1	7%
Thomas Paine	1	7%