

## **XPATH**

**Obtén el nombre de todos los padres, y muestra el resultado obtenido.**

Solución: /censo/ciudadanos/ciudadano/padre

**Obtén el nombre de todos los países, y muestra el resultado obtenido.**

Solución: /censo/ciudadanos/ciudadano/nacimiento/@pais

**Obtén el nombre de todas las madres nacidas después de 1973 , y muestra el resultado obtenido.**

Solución: /censo/ciudadanos/ciudadano[nacimiento/@año>1973]/madre

**Obtén el nombre de todos los nacidos antes de 1970, y muestra el resultado obtenido.**

Solución: /censo/ciudadanos/ciudadano[nacimiento/@año<1970]/nombre

**Obtén el nombre de los hijos de Pedro Ramírez Dupont, y muestra el resultado obtenido.**

Solución: /censo/ciudadanos/ciudadano[padre="Pedro Ramírez Dupont"]/nombre

**Obtén el nombre de los nacidos en España y muestra el resultado obtenido.**

Solución: /censo/ciudadanos/ciudadano[nacimiento/@pais="España"]/nombre

**Obtén el nombre de aquellos que hayan nacido en Valencia, antes de 1970 y muestra el resultado obtenido.**

Solución: /censo/ciudadanos/ciudadano[nacimiento/@lugar="Valencia" and nacimiento/@año<1970]/nombre

**Obtén las personas que tengan abuelos y muestra el resultado obtenido.**

Solución: /censo/ciudadanos/ciudadano[abuelos]/nombre

**Obtén las personas que hayan nacido en Francia y muestra el resultado obtenido.**

Solución: /censo/ciudadanos/ciudadano[nacimiento/@pais="Francia"]

**Obtén las personas que hayan nacido después de mayo, y muestra el resultado obtenido.**

Solución: /censo/ciudadanos/ciudadano[nacimiento/@mes>05]

## **XQUERY**

**Genera la información de los libros del autor Stevens. Realiza además el cálculo del precio total y muestra el resultado obtenido.**

Solución: for \$libro in //libro

```
where $libro/autor/apellido="Stevens"
return
<libro>
<titulo>{$libro/titulo}</titulo>
<autor>{$libro/autor}</autor>
<precioTotal>{$libro/precio+$libro/precio}</precioTotal>
</libro>
```

**Listar año y título de los libros que no tienen autor, y muestra el resultado obtenido.**

```
Solución: for $libro in //libro
where $libro/editor
return
<libro>
<titulo>{$libro/titulo/text()}</titulo>
<anio>{$libro/@anyo}</anio>
</libro>
```

**Listar año y título de todos los libros, ordenados por el año, y muestra el resultado obtenido.**

```
Solución: for $libro in //libro
order by $libro/@anyo
return
<libro>
<titulo>{$libro/titulo/text()}</titulo>
<anio>{$libro/@anyo}</anio>
</libro>
```

**Por cada libro, listar agrupado en un elemento <result> su título y autores y muestra el resultado obtenido.**

```
Solución: for $libro in //libro
return <result>{$libro/titulo, $libro/autor}</result>
```

**Listar año y título de los libros publicados por Addison-Wesley después del año 1992, y muestra el resultado obtenido.**

```
Solución: for $libro in //libro
where $libro/editorial="Addison-Wesley" and $libro/@anyo>1992
return <libro>
<titulo>{$libro/titulo}</titulo>
<anio>{$libro/@anyo}</anio>
</libro>
```

**Listar los libros cuyo precio sea 65.95, y muestra el resultado obtenido.**

Solución: for \$libro in //libro  
where \$libro/precio=65.95  
return \$libro

**Listar los libros publicados antes del año 2000, y muestra el resultado obtenido.**

Solución: for \$libro in //libro  
where \$libro/@anyo < 2000  
return \$libro

**Por cada libro, obtener su título y el número de autores, agrupados en un elemento <libro> y muestra el resultado obtenido.**

Solución: for \$libro in //libro  
return <libro>  
<titulo>{\$libro/titulo}</titulo>  
<numAutores>{\$libro/count(autor)}</numAutores>  
</libro>

**Listar el título de todos los libros, y muestra el resultado obtenido.**

Solución: for \$libro in //libro  
return \$libro/titulo

**Mostrar los apellidos de los autores que aparecen en el documento, sin repeticiones, ordenados alfabéticamente y muestra el resultado obtenido.**

Solución: for \$libro in distinct-values(//libro/autor/apellido)  
order by \$libro  
return <apellido>{\$libro}</apellido>

**Listar año y título de los libros que tienen más de un autor, y muestra el resultado obtenido.**

Solución: for \$libro in //libro  
where \$libro/count(autor)>1  
return <libro>  
<titulo>{\$libro/titulo}</titulo>  
<anio>{\$libro/@anyo}</anio>  
</libro>

## **XPATH**

**Realiza el listado de los diccionarios, y un sublistado con sus palabras y sus definiciones. Realiza la consulta y muestra su resultado.**

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

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:array="http://www.w3.org/2005/xpath-functions/array"
  xmlns:map="http://www.w3.org/2005/xpath-functions/map"
  xmlns:math="http://www.w3.org/2005/xpath-functions/math"
  exclude-result-prefixes="#all"
  expand-text="yes"
  version="3.0">
```

```
<xsl:template match="/diccionarios">
```

```
  <html>
```

```
    <head/>
```

```
    <body>
```

```
      <ul>
```

```
        <xsl:for-each select="diccionario">
```

```
          <li>
```

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

```
          </li>
```

```
          <ol>
```

```
            <xsl:for-each select="definicion">
```

```
              <li>
```

```
                <xsl:value-of select="palabra"/>
```

```
                <xsl:value-of select="def"/>
```

```
              </li>
```

```
            </xsl:for-each>
```

```
          </ol>
```

```
        </xsl:for-each>
```

```
      </ul>
```

```
    </body>
```

```
  </html>
```

```
</xsl:template>
```

```
</xsl:stylesheet>
```

**Realiza el xslt sobre el fichero banco.xml que genere la siguiente salida:**

```
<datos>
```

```
  <cuentas>
```

```
    <cuenta dnititular="5671001D">
```

```
      <creacion>13-abril-2012</creacion>
```

```

    <titular>Ramon Perez</titular>
    <saldoactual>12000euros</saldoactual>
  </cuenta>
  <cuenta dnititular="39812341C">
    <creacion>15-febrero-2011</creacion>
    <titular>Carmen Diaz</titular>
    <saldoactual>1900euros</saldoactual>
  </cuenta>
</cuentas>
<fondos/>
</datos>

```

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

```

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:array="http://www.w3.org/2005/xpath-functions/array"
  xmlns:map="http://www.w3.org/2005/xpath-functions/map"
  xmlns:math="http://www.w3.org/2005/xpath-functions/math"
  exclude-result-prefixes="#all"
  expand-text="yes"
  version="3.0">
  <xsl:template match="/listado">
    <xsl:for-each select="cuenta">
      <xsl:element name="datos">
        <xsl:element name="cuentas">

          <xsl:element name="cuenta">
            <xsl:attribute name="dnititular" select="titular/@dni"/>
            <xsl:element name="creación">
              <xsl:value-of select="fechacreacion"/>
            </xsl:element>
            <xsl:element name="titular">
              <xsl:value-of select="titular"/>
            </xsl:element>
            <xsl:element name="saldoactual">
              <xsl:value-of select="saldoactual"/> <xsl:value-of
select="saldoactual/@moneda"/>
            </xsl:element>
          </xsl:element>
        <xsl:element name="fondos">

```

```
</xsl:element>
</xsl:element>
```

```
</xsl:element>
</xsl:for-each>
```

```
</xsl:template>
</xsl:stylesheet>
```

**Realiza el listado de los diccionarios, con el siguiente resultado.**

```
<?xml version="1.0" encoding="UTF-8"?>
<html>
  <ul>
    <li>DRAE: Diccionario de la Real Academia Española</li>
    <li>DMM: Diccionario de María Moliner</li>
  </ul>
</html>
```

Solución:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:array="http://www.w3.org/2005/xpath-functions/array"
  xmlns:map="http://www.w3.org/2005/xpath-functions/map"
  xmlns:math="http://www.w3.org/2005/xpath-functions/math"
  exclude-result-prefixes="#all"
  expand-text="yes"
  version="3.0">
```

```
<xsl:template match="/diccionarios">
  <html>
  <head/>
  <body>
    <ul>
      <xsl:for-each select="diccionario">
        <li>
          <xsl:value-of select="siglas"/> :<xsl:value-of select="nombre"/>
        </li>
      </xsl:for-each>
    </ul>
```

```
</body>
</html>
</xsl:template>
</xsl:stylesheet>
```

Realiza el listado de la palabras, que genere el siguiente resultado:

```
<?xml version="1.0" encoding="UTF-8"?>
<html>
  <h1>Palabras en diccionarios</h1>
  <table border="1">
    <caption>Diccionario de la Real Academia Española</caption>
    <tr>
      <th>ordenador, ra</th>
      <td>Que ordena.</td>
      <td>Jefe de una ordenación de pagos u oficina de cuenta y razón.</td>
      <td>Máquina electrónica ...</td>
    </tr>
    <tr>
      <th>monitor, ra.</th>
      <td>Persona que guía el aprendizaje deportivo, cultural, etc.</td>
      <td>Hombre que amonesta o avisa.</td>
      <td>Aparato receptor ...</td>
    </tr>
  </table>

  <table border="1">
    <caption>Diccionario de María Moliner</caption>
    <tr>
      <th>ordenador, ra</th>
      <td>Se aplica a lo que sirve para ordenar.</td>
      <td>Aplicado a personas, aficionado a poner las cosas en orden.</td>
      <td>Jefe de una ordenación (oficina).</td>
      <td>Máquina electrónica ...</td>
    </tr>
    <tr>
      <th>monitor, ra.</th>
      <td>Persona que amonesta, corrige o avisa. Consejero.</td>
      <td>Persona encargada de instruir a otras en determinadas actividades.</td>
      <td>Cualquier aparato dotado de una pantalla.</td>
    </tr>
  </table>
```

```
</tr>
</table>
</html>
```

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

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:array="http://www.w3.org/2005/xpath-functions/array"
  xmlns:map="http://www.w3.org/2005/xpath-functions/map"
  xmlns:math="http://www.w3.org/2005/xpath-functions/math"
  exclude-result-prefixes="#all"
  expand-text="yes"
  version="3.0">
```

```
<xsl:template match="/diccionarios">
  <html>
    <head>
      <h1>Palabras del diccionario</h1>
    </head>
    <table border="1">
      <xsl:for-each select="diccionario">
        <caption><xsl:value-of select="nombre"/></caption>
        <xsl:for-each select="definicion">
          <tr>
            <th><xsl:value-of select="palabra"/></th>
            <th><xsl:value-of select="def"/></th>
          </tr>
        </xsl:for-each>
      </xsl:for-each>
    </table>
  </html>
</xsl:template>
</xsl:stylesheet>
```

**Realiza el xslt sobre el fichero banco.xml que genere la siguiente salida:**

```
<datos>
  <cuentas>
```



```

<cuanta dnititular="5671001D">
  <creacion>13-abril-2012</creacion>
  <titular>Ramon Perez</titular>
  <saldoactual>12000euros</saldoactual>
</cuanta>
<cuanta dnititular="39812341C">
  <creacion>15-febrero-2011</creacion>
  <titular>Carmen Diaz</titular>
  <saldoactual>1900euros</saldoactual>
</cuanta>
</cuentas>
<fondos>
  <fondo cuentaasociada="20-A">
    <cantidaddepositada>20000</cantidaddepositada>
    <moneda>Euros</moneda>
  </fondo>
  <fondo cuentaasociada="21-DX">
    <cantidaddepositada>4800</cantidaddepositada>
    <moneda>Dolares</moneda>
  </fondo>
</fondos>
</datos>

```

Solución:

```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:array="http://www.w3.org/2005/xpath-functions/array"
xmlns:map="http://www.w3.org/2005/xpath-functions/map"
xmlns:math="http://www.w3.org/2005/xpath-functions/math"
exclude-result-prefixes="#all"
expand-text="yes"
version="3.0">

```

```

<xsl:template match="/listado">
  <xsl:for-each select="cuanta">
    <xsl:element name="cuentas">
      <xsl:element name="cuanta">
        <xsl:attribute name="dnititular" select="titular/@dni"/>
      </xsl:element>
    </xsl:for-each>
  </xsl:template>

```

```

<xsl:element name="creacion">
<xsl:value-of select="fechacreacion"/>
</xsl:element>
<xsl:element name="titular">
<xsl:value-of select="titular"/>
</xsl:element>
<xsl:element name="saldoactual">
<xsl:value-of select="saldoactual"/><xsl:value-of select="saldoactual/@moneda"/>
</xsl:element>
</xsl:element>
</xsl:for-each>
<xsl:for-each select="fondo">
<xsl:element name="fondos">
<xsl:element name="fondo">
<xsl:attribute name="cuentaasociada" select="fondo/cuentaasociada"/>
<xsl:element name="cantidaddepositada">
<xsl:value-of select="fondo/datos/cantidaddepositada"/>
</xsl:element>
<xsl:element name="moneda">
<xsl:value-of select="fondo/datos/moneda"/>
</xsl:element>
</xsl:element>
</xsl:element>
</xsl:for-each>
</xsl:template>

</xsl:stylesheet>

```

**Realiza el listado de los diccionarios, con el siguiente resultado.**

**DRAE: Diccionario de la Real Academia Española.**

**DMM: Diccionario de María Moliner.**

Solución:

```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:array="http://www.w3.org/2005/xpath-functions/array"
xmlns:map="http://www.w3.org/2005/xpath-functions/map"
xmlns:math="http://www.w3.org/2005/xpath-functions/math"

```

```
exclude-result-prefixes="#all"
expand-text="yes"
version="3.0">
```

```
<xsl:template match="/diccionarios">
<html>
<xsl:for-each select="diccionario">
<ul>
<li><xsl:value-of select="siglas"/>: <xsl:value-of select="nombre"/></li>
</ul>
</xsl:for-each>
</html>
</xsl:template>
</xsl:stylesheet>
```

**Realiza el xslt sobre el fichero banco.xml que genere la siguiente salida:**

```
<datos>
  <cuentas>
    <cuanta dnititular="5671001D">
      <titular>Ramon Perez</titular>
    </cuanta>
    <cuanta dnititular="39812341C">
      <titular>Carmen Diaz</titular>
    </cuanta>
  </cuentas>
  <fondos/>
</datos>
```

Solucion:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:array="http://www.w3.org/2005/xpath-functions/array"
xmlns:map="http://www.w3.org/2005/xpath-functions/map"
xmlns:math="http://www.w3.org/2005/xpath-functions/math"
exclude-result-prefixes="#all"
expand-text="yes"
version="3.0">
```

```
<xsl:template match="/listado">
<xsl:for-each select="cuenta">
<xsl:element name="cuentas">
<xsl:element name="cuenta">
<xsl:attribute name="dnititular" select="titular/@dni"/>
</xsl:element>
<xsl:element name="creacion">
<xsl:value-of select="fechacreacion"/>
</xsl:element>
<xsl:element name="titular">
<xsl:value-of select="titular"/>
</xsl:element>
<xsl:element name="saldoactual">
<xsl:value-of select="saldoactual"/><xsl:value-of select="saldoactual/@moneda"/>
</xsl:element>
</xsl:element>
<xsl:element name="fondos">
</xsl:element>
</xsl:for-each>
</xsl:template>

</xsl:stylesheet>
```