

**Considerando el siguiente documento xml:**

```
<?xml version="1.0" encoding="UTF-8"?>
<empresa nombre="Estudios MJSV" web="http://www.estudios_mjsv.com/" >
  <empleados>
    <empleado id="ASG">
      <nombre>Antonio Sánchez Gutiérrez</nombre>
      <puesto>Director</puesto>
      <incorporacion año="2015" />
    </empleado>
    <empleado id="ARD">
      <nombre>Ana Reyes Descalzo</nombre>
      <puesto>Comercial</puesto>
      <incorporacion año="2016" />
    </empleado>
    <empleado id="JLA">
      <nombre>Juan López Álvarez</nombre>
      <puesto>Comercial</puesto>
      <incorporacion año="2021" />
    </empleado>
  </empleados>
</empresa>
```

**Nota: no considerar las líneas en blanco como diferencia respecto de los resultados a obtener**

**EJERCICIO 1**

Escribir el fichero con las reglas de transformación XSLT (.xsl) que permiten obtener la respuesta indicada a continuación:

```
<?xml version="1.0" encoding="UTF-8"?>
Antonio Sánchez Gutiérrez
Ana Reyes Descalzo
Juan López Álvarez
```

**SOLUCIÓN**

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

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

</xsl:stylesheet>
```

**SOLUCIÓN ALTERNATIVA 2**

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

  <xsl:template match="empleado">
    <xsl:apply-templates select="nombre"/>
  </xsl:template>

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

```

    <xsl:value-of select="."/>
</xsl:template>

</xsl:stylesheet>

```

**SOLUCIÓN ALTERNATIVA 3** (se elimina la plantilla para nombre y muestra lo mismo ya que hace lo que hace la plantilla por defecto que es la aplica si no existe una plantilla referenciada)

```

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

  <xsl:template match="empleado">
    <xsl:apply-templates select="nombre"/>
  </xsl:template>

</xsl:stylesheet>

```

## EJERCICIO 2

Escribir el fichero con las reglas de transformación XSLT (.xsl) que permiten obtener la respuesta indicada a continuación (notar que es similar a la anterior pero para permitir obtener párrafos html):

```

<?xml version="1.0" encoding="UTF-8"?>
<html>
  <p>Antonio Sánchez Gutiérrez</p>
  <p>Ana Reyes Descalzo</p>
  <p>Juan López Álvarez</p>
</html>

```

## SOLUCIÓN

```

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

  <xsl:template match="empleados">
    <html>
      <xsl:apply-templates select="empleado"/>
    </html>
  </xsl:template>

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

</xsl:stylesheet>

```

## SOLUCIÓN ALTERNATIVA 2

```

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

  <xsl:template match="empresa">

```

```

        <html>
        <xsl:apply-templates select="empleados/empleado"/>
        </html>
</xsl:template>

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

</xsl:stylesheet>

```

### EJERCICIO 3

Escribir el fichero con las reglas de transformación XSLT (.xsl) que permiten obtener la respuesta indicada a continuación (notar que es similar a las anteriores pero para obtener una lista ul html):

```

<?xml version="1.0" encoding="UTF-8"?>
<html>
  <ul>
    <li>Antonio Sánchez Gutiérrez</li>
    <li>Ana Reyes Descalzo</li>
    <li>Juan López Álvarez</li>
  </ul>
</html>

```

### SOLUCIÓN

```

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

<xsl:template match="empleados">
  <html>
    <ul>
      <xsl:apply-templates select="empleado"/>
    </ul>
  </html>
</xsl:template>

<xsl:template match="empleado">
  <li><xsl:value-of select="nombre"/></li>
</xsl:template>

</xsl:stylesheet>

```

### EJERCICIO 4

Escribir el fichero con las reglas de transformación XSLT (.xsl) que permiten obtener la respuesta indicada a continuación (notar que es similar a las anteriores con información adicional de la empresa antes de la lista: nombre y enlace a página web):

```

<?xml version="1.0" encoding="UTF-8"?>
<html>
  <h1>Estudios MJSV</h1>

```

```

    <p>Página web: <a href="http://www.estudios_mjsv.com">
http://www.estudios_mjsv.com </a></p>
    <ul>
        <li>Antonio Sánchez Gutiérrez</li>
        <li>Ana Reyes Descalzo</li>
        <li>Juan López Álvarez</li>
    </ul>
</html>

```

### SOLUCIÓN

```

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

<xsl:template match="empresa">
    <html>
        <h1><xsl:value-of select="@nombre"/></h1>
        <p>Página web: <a>
            <xsl:attribute name="href"><xsl:value-of
select="@web"/></xsl:attribute>
            <xsl:value-of select="@web"/></a></p>
        <ul>
            <xsl:apply-templates select="empleados/empleado"/>
        </ul>
    </html>
</xsl:template>

<xsl:template match="empleado">
    <li><xsl:value-of select="nombre"/></li>
</xsl:template>

</xsl:stylesheet>

```

Código HTML que se muestra en el navegador como:



### EJERCICIO 5

Escribir el fichero con las reglas de transformación XSLT (.xsl) que permiten obtener la respuesta indicada a continuación (notar que es similar a las anteriores con más información y en formato de tabla):

```

<?xml version="1.0" encoding="UTF-8"?>
<html>
    <h1>Estudios MJSV</h1>

```

```

    <p>Página web: <a href="http://www.estudios_mjsv.com">
http://www.estudios_mjsv.com </a></p>
    <table border="1">
    <tr>
        <th>Nombre del empleado</th>
        <th>Posición</th>
        <th>Año de incorporación</th>
    </tr>
    <tr>
        <td>Antonio Sánchez Gutiérrez</td>
        <td>Director</td>
        <td>2015</td>
    </tr>
    <tr>
        <td>Ana Reyes Descalzo</td>
        <td>Comercial</td>
        <td>2016</td>
    </tr>
    <tr>
        <td>Juan López Álvarez</td>
        <td>Comercial</td>
        <td>2021</td>
    </tr>
</table>
</html>

```

### SOLUCIÓN

```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">
    <xsl:strip-space elements="*" />

    <xsl:template match="empresa">
        <html>
            <h1><xsl:value-of select="@nombre" /></h1>
            <p>Página web: <a >
                <xsl:attribute name="href">
                    <xsl:value-of select="@web" />
                </xsl:attribute>
                <xsl:value-of select="@web" /></a></p>
            <table border="1">
                <tr>
                    <th>Nombre del empleado</th>
                    <th>Posición</th>
                    <th>Año de incorporación</th>
                </tr>
                <xsl:apply-templates />
            </table>
        </html>
    </xsl:template>

    <xsl:template match="empleado">
        <tr>
            <td><xsl:value-of select="nombre"/></td>
            <td><xsl:value-of select="puesto"/></td>
            <td><xsl:value-of select="incorporacion/@año"/></td>
        </tr>
    </xsl:template>

```

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

## EJERCICIO 6

Escribir el fichero con las reglas de transformación XSLT (.xsl) que permiten generar un nuevo fichero xml en el que se conviertan las etiquetas (elementos hijos) de elemento empleado en atributos:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet href="ejercicio_5.xsl" type="text/xsl"?>
<empresa nombre="Estudios MJSV" web="http://www.estudios_mjsv.com/" >
  <empleados>
    <empleado id="ASG" nombre="Antonio Sánchez Gutiérrez"
      puesto="Director" incorporacion_año="2015" />
    <empleado id="ARD" nombre="Ana Reyes Descalzo" puesto="Comercial"
      incorporacion_año="2016" />
    <empleado id="JLA" nombre="Juan López Álvarez" puesto="Comercial"
      incorporacion_año="2021" />
  </empleados>
</empresa>
```

## SOLUCIÓN

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

  <xsl:template match="/">
    <empresa nombre="Estudios MJSV" web="http://www.estudios_mjsv.com/"
    >
      <empleados>
        <xsl:apply-templates select="empresa/empleados/empleado"/>
      </empleados>
    </empresa>
  </xsl:template>

  <xsl:template match="empleado">
    <empleado>
      <xsl:attribute name="id"><xsl:value-of
      select="@id"/></xsl:attribute>
      <xsl:attribute name="nombre"><xsl:value-of
      select="nombre"/></xsl:attribute>
      <xsl:attribute name="puesto"><xsl:value-of
      select="puesto"/></xsl:attribute>
      <xsl:attribute name="incorporacion_año"><xsl:value-of
      select="incorporacion/@año"/></xsl:attribute>
    </empleado>
  </xsl:template>

</xsl:stylesheet>
```

## EJERCICIO 7

Escribir el fichero con las reglas de transformación XSLT (.xsl) que permiten cambiar la estructura del documento, es decir permiten obtener un nuevo documento xml en el que aparece para los elementos empleado primero el puesto, luego el año de incorporación y luego el nombre.

```

<?xml version="1.0" encoding="UTF-8"?>
<empresa nombre="Estudios MJSV" web="http://www.estudios_mjsv.com/" >
  <empleados>
    <empleado id="ASG">
      <puesto>Director</puesto>
      <incorporacion año="2015"/>
      <nombre>Antonio Sánchez Gutiérrez</nombre>
    </empleado>
    <empleado id="ARD">
      <puesto>Comercial</puesto>
      <incorporacion año="2016"/>
      <nombre>Ana Reyes Descalzo</nombre>
    </empleado>
    <empleado id="JLA">
      <puesto>Comercial</puesto>
      <incorporacion año="2021"/>
      <nombre>Juan López Álvarez</nombre>
    </empleado>
  </empleados>
</empresa>

```

### SOLUCIÓN

```

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

  <xsl:template match="/">
    <empresa nombre="Estudios MJSV" web="http://www.estudios_mjsv.com/"
    >
      <empleados>
        <xsl:apply-templates select="empresa/empleados/empleado"/>
      </empleados>
    </empresa>
  </xsl:template>

  <xsl:template match="empleado">
    <empleado>
      <xsl:attribute name="id"><xsl:value-of select="@id"
      /></xsl:attribute>
      <puesto><xsl:value-of select="puesto" /></puesto>
      <incorporacion><xsl:attribute name="año"><xsl:value-of
      select="incorporacion/@año" /></xsl:attribute></incorporacion>
      <nombre><xsl:value-of select="nombre" /></nombre>
    </empleado>
  </xsl:template>

</xsl:stylesheet>

```

### EJERCICIO 8

Escribir el fichero con las reglas de transformación XSLT (.xsl) que permiten cambiar el atributo id del elemento empleado por un elemento hijo de empleado:

```

<?xml version="1.0" encoding="UTF-8"?>
<empresa nombre="Estudios MJSV" web="http://www.estudios_mjsv.com/" >
  <empleados>
    <empleado>

```

```

        <id>ASG</id>
        <nombre>Antonio Sánchez Gutiérrez</nombre>
        <puesto>Director</puesto>
        <incorporacion año="2015" />
    </empleado>
    <empleado>
        <id>ARD</id>
        <nombre>Ana Reyes Descalzo</nombre>
        <puesto>Comercial</puesto>
        <incorporacion año="2016" />
    </empleado>
    <empleado>
        <id>JLA</id>
        <nombre>Juan López Álvarez</nombre>
        <puesto>Comercial</puesto>
        <incorporacion año="2021" />
    </empleado>
</empleados>
</empresa>

```

### SOLUCIÓN

```

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

    <xsl:template match="/">
        <empresa nombre="Estudios MJSV" web="http://www.estudios_mjsv.com/"
        >
            <empleados>
                <xsl:apply-templates select="empresa/empleados/empleado"/>
            </empleados>
        </empresa>
    </xsl:template>

    <xsl:template match="empleado">
        <empleado>
            <xsl:element name="id"><xsl:value-of select="./@id"
            /></xsl:element>
            <nombre><xsl:value-of select="nombre" /></nombre>
            <puesto><xsl:value-of select="puesto" /></puesto>
            <incorporacion><xsl:attribute name="año"><xsl:value-of
            select="incorporacion/@año" /></xsl:attribute></incorporacion>
        </empleado>
    </xsl:template>

</xsl:stylesheet>

```

### **EJERCICIO 9**

Escribir el fichero con las reglas de transformación XSLT (.xsl) que permiten obtener la respuesta indicada a continuación utilizando el elemento <xsl:for-each>:

```

<?xml version="1.0" encoding="UTF-8"?>
<html>
    <ul>
        <li>Antonio Sánchez Gutiérrez</li>
    </ul>

```



```

        <li>Ana Reyes Descalzo</li>
        <li>Juan López Álvarez</li>
    </ul>
</html>

```

**Nota:** se pide lo mismo que en el ejercicio 3, utilizando el elemento `<xsl:for-each>`

### SOLUCIÓN

```

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

<xsl:template match="empleados">
    <html>
        <ul>
            <xsl:for-each select="empleado">
                <li><xsl:value-of select="nombre"/></li>
            </xsl:for-each>
        </ul>
    </html>
</xsl:template>

</xsl:stylesheet>

```

**Nota:** comparar solución con solución en base a uso de plantillas (ejercicio 3) que se incluye a continuación. Se trata de una solución en principio más sencilla.

```

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

<xsl:template match="empleados">
    <html>
        <ul>
            <xsl:apply-templates select="empleado"/>
        </ul>
    </html>
</xsl:template>

<xsl:template match="empleado">
    <li><xsl:value-of select="nombre"/></li>
</xsl:template>

</xsl:stylesheet>

```

### **EJERCICIO 10**

Escribir el fichero con las reglas de transformación XSLT (.xsl) que permiten obtener la respuesta indicada a continuación utilizando el elemento `<xsl:for-each>` y elementos de selección o condicionales:

```

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

```

```

<h1>Estudios MJSV</h1>
<p>Página web: <a href="http://www.estudios_mjsv.com">
http://www.estudios_mjsv.com </a></p>
<table border="1">
<tr>
<th>Nombre del empleado</th>
<th>Posición</th>
<th>Año de incorporación</th>
</tr>
<tr>
<td>Antonio Sánchez Gutiérrez</td>
<td>Director</td>
<td>2015</td>
</tr>
<tr>
<td>Ana Reyes Descalzo</td>
<td>Comercial</td>
<td>2016</td>
</tr>
<tr>
<td>Juan López Álvarez</td>
<td>Comercial</td>
<td>2021</td>
</tr>
</table>
</html>

```

**Nota:** se pide lo mismo que en el ejercicio 5, utilizando el elemento `<xsl:for-each>` y elementos de selección o condicionales

### SOLUCIÓN

```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">
  <xsl:strip-space elements="*" />

  <xsl:template match="empresa">
    <html>
      <h1><xsl:value-of select="@nombre" /></h1>
      <p>Página web: <a >
        <xsl:attribute name="href">
          <xsl:value-of select="@web" />
        </xsl:attribute>
        <xsl:value-of select="@web" /></a></p>
      <table border="1">
        <tr>
          <th>Nombre del empleado</th>
          <th>Posición</th>
          <th>Año de incorporación</th>
        </tr>

        <xsl:for-each select="empleados/empleado">
          <tr>
            <xsl:for-each select="*">
              <xsl:choose>
                <xsl:when test="!=''">
                  <td> <xsl:value-of select="." /> </td>
                </xsl:when>

```

```
        <xsl:otherwise>
            <xsl:if test="@año!=''">
                <td> <xsl:value-of select="@año" /> </td>
            </xsl:if>
        </xsl:otherwise>
    </xsl:choose>
</xsl:for-each>
</tr>
</xsl:for-each>

</table>
</html>
</xsl:template>

</xsl:stylesheet>
```