

Ejercicios 1

Actividad 1.1

Esta es la actividad 1.1 del tema, en la cual se pedía que añadiésemos una materia extra además de las ya dadas y con la misma estructura que en el ejemplo.

```

1  <?xml version="1.0" encoding="iso-8859-1"?>
2  <!--Actividad 1.1-->
3
4  <!--Creamos exámenes, y dentro de este creamos materia -->
5  <exámenes>
6    <materia>
7      <!--Ahora creamos dentro nombre, aprobados, suspensos y unidades. Este paso lo vamos a repetir 2 veces más -->
8      <nombre>Lenguaje de marcas</nombre>
9      <aprobados>20</aprobados>
10     <suspensos>6</suspensos>
11     <unidades>4</unidades>
12   </materia>
13   <materia>
14     <nombre>Sistemas operativos</nombre>
15     <aprobados>7</aprobados>
16     <suspensos>20</suspensos>
17     <unidades>5</unidades>
18   </materia>
19   <materia>
20     <nombre>Programación</nombre>
21     <aprobados>18</aprobados>
22     <suspensos>9</suspensos>
23     <unidades>6</unidades>
24   </materia>
25 </exámenes>
26 <!--En esta practica tras validarlo no ha dado errores-->

```

Este es la captura de la validación del código anteriormente mostrado.

XML Validator



Como podemos apreciar, el validador no nos muestra ningún error.

Actividad 1.2

En esta actividad se nos pedía que comprobásemos en un navegador que tuviésemos instalado si el siguiente código estaba correctamente escrito.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <!--Actividad 1.2-->
4
5 <!--Creamos una agenda-->
6 <agenda>
7 <!--Dentro de la agenda creamos contactos-->
8 <contactos>
9 <!--De cada contacto vamos a poner su nombre, su ciudad, su email y si está casado o no-->
10 <nombre>Alumno</nombre>
11 <ciudad cp="15500">Ciudad Real </ciudad>
12 <email>alumno@gmail.com</email>
13 <casado v="no"/>
14 </contactos>
15 </agenda>
```

Este es la captura de la validación del código anteriormente mostrado.

XML Validator



Como podemos apreciar en la imagen anterior, el código de la actividad 1.2 no presenta errores tras haberlo validado.

Actividad 1.3

Tras haber finalizado la actividad anterior, hemos copiado unas líneas de código extra al anterior código con el objetivo de que comprobemos si está bien formulado el código.

```
1  <?xml version="1.0" encoding="utf-8"?>
2
3  <!--Actividad 1.3-->
4
5  <agenda>
6      <!--Vamos a crear un contacto-->
7      <contactos>
8          <nombre>Alumno</nombre>
9          <Ciudad cp="15500">Ciudad Real</ciudad>
10         <telefono n="612345678"/>
11         <email>alumno@gmail.com</email>
12         <casado v="no">
13             <@AsignaturaPreferida>Matemáticas</@>
14         </contactos>
15 </agenda>
```

Este es la captura de la validación del código anteriormente mostrado,

XML Validator



The screenshot shows an XML Validator interface. The XML code from the previous block is pasted into the editor. On the left, a list of lines from 1 to 16 is shown. Lines 8, 9, 10, 11, 12, and 13 are highlighted with red error markers. At the bottom, a red error message box states: "Error : InvalidTag", "Line : 9", and "Message : Closing tag 'Ciudad' is expected inplace of 'ciudad'". Below the editor, there are buttons for "Auto", "Validate XML", "Choose a file..", "Load URL", and "Download".

Y como podemos observar, hay algunos fallos que vamos a nombrar y corregir a continuación:

- En la línea 9, la etiqueta de apertura <Ciudad> no coincide en sintaxis con la de cierre <ciudad>. Para solucionarlo se debe ponerlas igual, así que yo he rectificado la de apertura y la he puesto como <ciudad>
- En la línea 12, la expresión <casado v="no">, no está cerrada, así que añadimos "/" entre el final de las comillas y el cierre, quedando finalmente así: <casado v="no"/>
- En la línea 13, donde pone <@AsignaturaPreferida>Matemáticas</@>, no tienen la misma sintaxis la etiqueta de apertura y de cierre. Como solución a este error he quitado las @ y en la etiqueta de cierre he puesto lo mismo que en la de apertura

El código arreglado quedaría así:

```

1  <?xml version="1.0" encoding="utf-8"?>
2
3  <!--Actividad 1.3-->
4
5  <agenda>
6      <!--Vamos a crear un contacto-->
7      <contactos>
8          <nombre>Alumno</nombre>
9          <ciudad cp="15500">Ciudad Real</ciudad>
10         <teléfono n="612345678"/>
11         <email>alumno@gmail.com</email>
12         <casado v="no"/>
13         <AsignaturaPreferida>Matemáticas</AsignaturaPreferida>
14     </contactos>
15 </agenda>

```

La validación de este código quedaría de la siguiente manera:

XML Validator



The screenshot shows a web-based XML Validator interface. At the top, there's a 'Sample' button and a 'Add to Fav' icon. The main area displays the XML code from the previous block. Below the code, there's a status bar showing 'Ln: 16 Col: 0' and 'size: 395 B'. At the bottom, there are four buttons: 'Auto' (checked), 'Validate XML' (highlighted in green), 'Choose a file..', 'Load URL', and 'Download'.

Podemos observar que la validación de este código no presenta ningún fallo.