

Segundo trimestre

Lenguaje de marcas

Sergio Felipe García

Índice

Fase 1	2
Fase 2	7
Apartado 1	7
Apartado 2	10
Apartado 3	11
Apartado 4	11
Fase 3	14
Webgrafía	16

Fase 1

En la fase 1 tal y como se nos pide he hecho un documento xml en donde he puesto como elemento principal <pedidos>, que contiene información sobre todos los pedidos realizados durante el año, dentro del elemento <pedidos>, hay un elemento <anno> que tiene como atributo year con el valor "2021", que indica que los pedidos representados en este código son de ese año en particular.

El elemento <anno> también contiene otro elemento anidado llamado <trimestre> que representa el trimestre del año en el que se ha realizado el pedido, el elemento trimestre tiene un atributo llamado num que indica el número del trimestre.

El elemento <trimestre> contiene otro elemento anidado llamado <pedido> que representa un pedido en particular realizado por un cliente, el elemento <pedido> contiene información detallada sobre el cliente, la fecha en que se realizó el pedido, los detalles del pedido y los productos comprados.

A su vez el elemento <pedido> contiene un subelemento llamado <cliente> que proporciona información detallada sobre el cliente que realizó el pedido, como su nombre, apellidos, número de teléfono, dirección y correo electrónico.

El elemento <pedido> también contiene un subelemento llamado <fecha_inclusion> que indica la fecha en que se registró el cliente en el sistema de la tienda en línea, también el elemento <pedido> tiene otro subelemento <detalle_pedido> que contiene información detallada sobre el pedido realizado, incluido el número de pedido, la fecha de compra, la fecha de entrega, el total de la factura y los productos comprados.

En el elemento <detalle_pedido> hay un subelemento llamado <productos> que representa los productos comprados en el pedido, y dentro de <productos> hay un subelemento llamado <producto> que representa la información sobre el nombre del producto, su referencia, el precio y la cantidad comprada en unidades y así se repite para el resto de documento xml la estructura.

```

<pedidos>
  <anno year="2021">
    <trimestre num="1">
      <pedido>
        <cliente>
          <nombre>María</nombre>
          <apellidos>González Pérez</apellidos>
          <telefono>555-123-4567</telefono>
          <direccion>
            <calle>Calle Mayor, 1</calle>
            <ciudad>Madrid</ciudad>
            <codigo_postal>28001</codigo_postal>
            <provincia>Madrid</provincia>
          </direccion>
          <correo_electronico>maria.gonzalez@gmail.com</correo_electronico>
        </cliente>
        <fecha_inclusion>2021-01-02</fecha_inclusion>
        <detalle_pedido>
          <numero_pedido>10001</numero_pedido>
          <fecha_compra>2021-01-02</fecha_compra>
          <fecha_entrega>2021-01-05</fecha_entrega>
          <total_factura>29.99</total_factura>
          <productos>
            <producto>
              <nombre_producto>Camiseta de mujer</nombre_producto>
              <referencia>ABC123</referencia>
              <precio>29.99</precio>
              <unidades>1</unidades>
            </producto>
          </productos>
        </detalle_pedido>
      </pedido>
    </trimestre>
  </anno>
</pedidos>

```

Ahora explicaré la dtd que he hecho para que el xml pueda ser validado.

Primero he utilizado el `<!ELEMENT>` que me ayuda a definir un elemento y su contenido por ejemplo, en `<!ELEMENT anno (trimestre*)>` lo que hace es definir el elemento 'anno' para que pueda contener cero o más elementos trimestre y con el asterisco indicó que el elemento anno puede tener cero o más elementos trimestre como su contenido.

```
<!ELEMENT anno (trimestre*)>
```

Después tendríamos los `<!ATTLIST>` que con ellos defino qué atributos pueden tener los elementos, por ejemplo en `<!ATTLIST anno year CDATA #REQUIRED>` en esta línea defino que anno tiene como atributo year y cómo year es de tipo CDATA su valor puede ser cualquier cadena de caracteres y con el `#REQUIRED` estoy indicando que el atributo year es obligatorio y debe estar escrito en cualquier instancia del elemento anno en el documento XML y no debe estar vacío y lo mismo ocurre para

el trimestre que en su caso el atributo num es obligatorio que lo lleve escrito en cada instancia de trimestre .

```
<!ATTLIST trimestre num CDATA #REQUIRED>
```

Luego vendrían las líneas de código en donde especifico que elementos debe llevar cada etiqueta, por ejemplo en `<!ELEMENT pedido (cliente, fecha_inclusion, detalle_pedido)>` aquí lo que estoy haciendo es decir que en cada instancia de pedido en el documento xml deben estar los tres subelementos que en este caso son cliente, fecha_inclusion y detalle_pedido además de que si no tiene la estructura específica y estos tres subelementos el documento xml no es válido.

```
<!ELEMENT pedido (cliente, fecha_inclusion, detalle_pedido)>
```

Por último y no menos importante tenemos las líneas en donde se declaran los elementos por ejemplo, en `<!ELEMENT nombre (#PCDATA)>` aquí estoy definiendo el contenido del elemento nombre que sea de tipo (#PCDATA) para que contenga cualquier texto y no tenga elementos secundarios o hijos y así con todos los demás.

```
<!ELEMENT nombre (#PCDATA)>
```

Y así es como quedaría la dtd:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE pedidos[
<!ELEMENT pedidos (anno*)>
<!ELEMENT anno (trimestre*)>
<!ATTLIST anno year CDATA #REQUIRED>
<!ELEMENT trimestre (pedido+)>
<!ATTLIST trimestre num CDATA #REQUIRED>
<!ELEMENT pedido (cliente, fecha_inclusion, detalle_pedido)>
<!ELEMENT cliente (nombre, apellidos, telefono, direccion, correo_electronico)>
<!ELEMENT nombre (#PCDATA)>
<!ELEMENT apellidos (#PCDATA)>
<!ELEMENT telefono (#PCDATA)>
<!ELEMENT direccion (calle, ciudad, codigo_postal, provincia)>
<!ELEMENT calle (#PCDATA)>
<!ELEMENT ciudad (#PCDATA)>
<!ELEMENT codigo_postal (#PCDATA)>
<!ELEMENT provincia (#PCDATA)>
<!ELEMENT correo_electronico (#PCDATA)>
<!ELEMENT fecha_inclusion (#PCDATA)>
<!ELEMENT detalle_pedido (numero_pedido, fecha_compra, fecha_entrega, total_factura, productos)>
<!ELEMENT numero_pedido (#PCDATA)>
<!ELEMENT fecha_compra (#PCDATA)>
<!ELEMENT fecha_entrega (#PCDATA)>
<!ELEMENT total_factura (#PCDATA)>
<!ELEMENT productos (producto+)>
<!ELEMENT producto (nombre_producto, referencia, precio, unidades)>
<!ELEMENT nombre_producto (#PCDATA)>
<!ELEMENT referencia (#PCDATA)>
<!ELEMENT precio (#PCDATA)>
<!ELEMENT unidades (#PCDATA)>
]>
```

Fase 2

Apartado 1

Para el primer apartado he utilizado javascript para poder vincular el documento xml al proyecto html , en el código he puesto comentarios especificando que hace cada línea de código para poder crear la tabla de pedidos.

```
<!DOCTYPE html>
<html lang="es-es">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Aplicación Pedidos</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <h2>Pedidos</h2>

  <section>

</section>
<script>
  if (window.XMLHttpRequest)
  {
    xmlhttp=new XMLHttpRequest();
  }
  else
  {
    // comprueba si el navegador admite XMLHttpRequest. Si es así, se crea un objeto XMLHttpRequest.
    // si no crea un objeto ActiveXObject para Internet Explorer. Sirve para hacer solicitudes HTTP
    // desde un script del lado del cliente y actualizar partes de una página web sin tener que volver a cargar la página completa.

    xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
  }
}
```



```

// Luego, se abre el archivo "documento1.xml" mediante una solicitud GET
// utilizando el objeto XMLHttpRequest. La opción "false" indica que la solicitud es síncrona,
// lo que significa que el script se ejecuta hasta que se recibe una respuesta.

xmlhttp.open("GET","documento1.xml",false);
xmlhttp.send();
// Con el metodo responseXML nos traemos todo el contenido del archivo XML que tenemos en el backend y lo almacenamos en una variable xmlDoc
// Ojo hay que tener cuidado con los tiempos de respuesta si vamos a traernos un fichero xml muy grande.
xmlDoc=xmlhttp.responseXML;
//Con la línea de código const section = document.querySelector('section'); lo que hace es seleccionar
// el primer elemento HTML que coincide con el selector CSS section y lo almacena en la variable section
const section = document.querySelector('section');
document.write("<table class='table' border='10'>");
// obtenemos el conjunto de elementos "pedido" del archivo XML utilizando el método getElementsByTagName. Se almacenan en la variable "pedidos".
// si miramos el fichero <pedido> datos del cliente y los detalles del pedido con los productos y sus importes.
var pedidos=xmlDoc.getElementsByTagName("pedido");
//En esta parte estoy creando una tabla con 8 columnas y una fila que contiene los encabezados de cada columna.
//El primer document.write("<tr>"); crea una nueva fila en la tabla y las siguientes líneas crean los encabezados
//La última fila el document.write("</tr>"); cierra los encabezados
document.write("<tr>");
document.write("<th>Cliente</th>");
document.write("<th>Fecha de compra</th>");
document.write("<th>Fecha de entrega</th>");
document.write("<th>Producto</th>");
document.write("<th>Referencia</th>");
document.write("<th>Precio</th>");
document.write("<th>Unidades</th>");
document.write("<th>Total de Factura</th>");
document.write("</tr>");

```

Utilizó el bucle for para recorrer cada pedido y agregar una nueva fila a la tabla HTML para mostrar la información de cada pedido en su respectiva columna.

```

//A continuación, se utiliza un bucle for para iterar sobre los elementos "pedido". En cada iteración,
//la variable i toma el valor del índice correspondiente en el arreglo pedidos
//y se sigue ejecutando hasta que llega a los 80 pedidos

for(var i=0;i<pedidos.length;i++)
{
    //En esta línea se extrae el valor del elemento "nombre" dentro del elemento "pedido" en el archivo XML
    //Aquí "pedidos" es un array que contiene todos los elementos "pedido" del archivo XML.
    //Con getElementsByTagName("nombre") devuelve un array de elementos con los que tengan el nombre de
    //"nombre" dentro del elemento "pedido" y
    // con "[0]" selecciona el primer elemento del array de elementos con nombre "nombre" dentro del elemento "pedido".
    //Con "childNodes[0]" lo que hace es seleccionar el primer nodo hijo del elemento "nombre"
    // Y ".nodeValue" devuelve el valor del nodo de texto seleccionado,
    //que en este caso es el valor del campo "nombre" del pedido.
    //Y así con todos los demás.
    nombre=pedidos[i].getElementsByTagName("nombre")[0].childNodes[0].nodeValue
    apellidos=pedidos[i].getElementsByTagName("apellidos")[0].childNodes[0].nodeValue
    fecha_compra=pedidos[i].getElementsByTagName("fecha_compra")[0].childNodes[0].nodeValue
    fecha_entrega=pedidos[i].getElementsByTagName("fecha_entrega")[0].childNodes[0].nodeValue
    nombre_producto=pedidos[i].getElementsByTagName("nombre_producto")[0].childNodes[0].nodeValue
    referencia=pedidos[i].getElementsByTagName("referencia")[0].childNodes[0].nodeValue
    precio=pedidos[i].getElementsByTagName("precio")[0].childNodes[0].nodeValue
    unidades=pedidos[i].getElementsByTagName("unidades")[0].childNodes[0].nodeValue
    total_factura=pedidos[i].getElementsByTagName("total_factura")[0].childNodes[0].nodeValue
}

```

```

// En este codigo lo que hago es crear la tabla para mostrar los pedidos
//En esta linea utilizo el metodo document.write("<tr><td>"); para definir una fila
//en la tabla y el document.write(nombre + " " + apellidos); escribe en cada fila
// de la columna clientes el nombre y apellido de cada cliente además concateno las dos
// variables con el espacio y el +
//Con las demás variables sigue la misma formula se ponen en la columna que le corresponde por la
//estructura hecha en la tabla y muestran los datos en cada fila
document.write("<tr><td>");
document.write(nombre + " " + apellidos);
document.write("</td><td>");
document.write(fecha_compra);
document.write("</td><td>");
document.write(fecha_entrega);
document.write("</td><td>");
document.write(nombre_producto);
document.write("</td><td>");
document.write(referencia);
document.write("</td><td>");
document.write(precio);
document.write("</td><td>");
document.write(unidades);
document.write("</td><td>");
document.write(total_factura);
document.write("</td>");
}
document.write("</table>");// Con esto cierro la tabla
</script>

```

Este sería el resultado:

Pedidos

Cliente	Fecha de compra	Fecha de entrega	Producto	Referencia	Precio	Unidades	Total de Factura
María González Pérez	2021-01-02	2021-01-05	Camiseta de mujer	ABC123	29.99	1	29.99
Pedro González Gomez	2021-02-03	2021-02-05	Zapatos caballero casual planos	ABC156	65.95	1	65.95
Paco Domingez Gomez	2021-02-07	2021-02-09	Zapato cómodo	ABC178	20.99	1	20.99
Sergio Felipe García	2021-02-10	2021-02-12	deportivas de niño	ABC200	26.99	1	26.99
María González Pérez	2021-01-25	2021-01-27	Vestido camisero con cinturón para anudar	ZSD209	19.99	1	19.99
Paco Jimenez Lopez	2021-02-11	2021-02-13	deportivo de trekking para hombre	ABC180	20.99	1	20.99
Encarna Perez Delgado	2021-02-08	2021-02-10	Pantalón de felpa con puños en el bajo	SDE321	9.99	2	19.98
Victor García Fernandez	2021-02-10	2021-02-12	mocasín castellano de piel	OLÑ618	53.99	1	53.99
Encarna Perez Delgado	2021-02-26	2021-02-28	Jersey tricot manga larga acabados en canalé	KGQ671	12.99	1	12.99
Encarna Perez Delgado	2021-03-03	2021-03-05	Pijama camiseta manga corta	GVÑ058	12.99	1	12.99
Natalia Ramirez Benito	2021-04-02	2021-04-04	Vestido evasé punto tricot manga larga	FÑC401	19.99	1	19.99
Natalia Ramírez Benito	2021-04-06	2021-04-08	Chaquetón de lana liso para mujer	LXZ345	80.99	1	80.99

Para el apartado dos y tres se ha hecho lo mismo solo se han cambiado los datos que queremos mostrar en las tablas

Apartado 2

El resultado de la tabla clientes:

Clientes

Cientes	Telefonos	Correo electronico	Direcciones	Ciudades	Provincias	Codigo postal	Fecha de Inclusion
María González Pérez	555-123-4567	maria.gonzalez@gmail.com	Calle Mayor, 1	Madrid	Madrid	28001	2021-01-02
Pedro González Gomez	556-122-4568	pedro.gonzalez@gmail.com	Anillo Periférico	Ciudad de México	México	28002	2021-02-02
Paco Domingez Gomez	123-122-4564	paco.domingez@gmail.com	Puerta del Sol.	Madrid	Madrid	28003	2021-02-06
Sergio Felipe García	654-987-4564	sergio.felipe@gmail.com	Puerta del Sol.	Madrid	Madrid	28004	2021-02-06
María González Pérez	555-123-4567	maria.gonzalez@gmail.com	Calle Mayor, 1	Madrid	Madrid	28001	2021-01-02
Paco Jimenez Lopez	677-122-4564	paco.jimenez@gmail.com	Puerta del Sol.	Madrid	Madrid	28005	2021-02-10
Encarna Perez Delgado	766-122-4564	encarna.perez@gmail.com	Calle Huertas	Madrid	Madrid	28006	2021-02-08
Victor García Fernandez	879-361-4564	victor.garcia@gmail.com	Puerta del Sol.	Madrid	Madrid	28012	2021-02-10

Apartado 3

El resultado de la tabla de facturas:

Facturas							
Cientes	fecha de compra	fecha de entrega	Numero de Pedido	Producto	Unidades	Precio	Total de Factura
María González Pérez	2021-01-02	2021-01-05	10001	Camiseta de mujer	1	29.99	29.99
Pedro González Gomez	2021-02-03	2021-02-05	10002	Zapatos caballero casual planos	1	65.95	65.95
Paco Domingez Gomez	2021-02-07	2021-02-09	10003	Zapato cómodo	1	20.99	20.99
Sergio Felipe García	2021-02-10	2021-02-12	10004	deportivas de niño	1	26.99	26.99
María González Pérez	2021-01-25	2021-01-27	10009	Vestido camisero con cinturón para anudar	1	19.99	19.99
Paco Jimenez Lopez	2021-02-11	2021-02-13	10006	deportivo de trekking para hombre	1	20.99	20.99
Encarna Perez Delgado	2021-02-08	2021-02-10	10007	Pantalón de felpa con puños en el bajo	2	9.99	19.98
Victor García Fernandez	2021-02-10	2021-02-12	10028	mocasin castellano de piel	1	53.99	53.99
Encarna Perez Delgado	2021-02-26	2021-02-28	10075	Jersey tricot manga larga acabados en canalé	1	12.99	12.99
Encarna Perez Delgado	2021-03-03	2021-03-05	10097	Pijama camiseta manga corta	1	12.99	12.99

Apartado 4

Para el apartado cuatro el cual nos pide que mostremos los productos vendidos en el primer trimestre de 2021 y en el último de 2022 he tenido que agregar una condición más específicamente un if para poder completar este apartado

```
for(var i=0;i<pedidos.length;i++)
{
    // el valor de los atributos "num" y "year" de los nodos "padre" y "padre del padre" del elemento actual.
    //Estos valores se almacenan en las variables "trimestre" y "anno", respectivamente.

    var trimestre=pedidos[i].parentNode.getAttribute("num");
    var anno=pedidos[i].parentNode.parentNode.getAttribute("year");
    //En esta línea se verifica si el trimestre es "1" y el año es "2021", o si el trimestre es "4" y el año es "2022"
    // si se cumple la condicion se muestran los pedidos vendidos en esos trimestres
    if ((trimestre === "1" && anno === "2021") || (trimestre === "4" && anno === "2022")) {
        nombre_producto=pedidos[i].getElementsByTagName("nombre_producto")[0].childNodes[0].nodeValue
        referencia=pedidos[i].getElementsByTagName("referencia")[0].childNodes[0].nodeValue
        precio=pedidos[i].getElementsByTagName("precio")[0].childNodes[0].nodeValue
        unidades=pedidos[i].getElementsByTagName("unidades")[0].childNodes[0].nodeValue

        document.write("<tr>");
    }
}
```

También le he añadido css a las tablas

```

/* Estilo para la tabla */
table {
    font-family: Times , sans-serif; /* establezco la fuente de texto para la tabla.*/
    font-size: 16px; /*Aquí establezco el tamaño de texto para la tabla*/
    border-collapse: collapse; /*Con border-collapse puedo unir las celdas de la tabla*/
    width: 100%; /*Aquí con width hago que el ancho que ocupe sea del 100%*/
    background-color: #f9f9f9;
}

/* Estilo para las celdas encabezado */
th {
    background-color: #4CAF50;
    color: white; /*El texto del encabezado lo pongo de color blanco*/
    text-align: center; /*Centro el texto */
    padding: 8px; /*Establezco un espacio entre el borde y contenido de una celda*/
}

/* Estilo para las celdas de datos */
td {
    border: 1px solid #ddd; /*Le doy estilo a los bordes de las celdas de los datos*/
    padding: 8px;
    color: chocolate;
}

/* Estilo para las filas pares */
tr:nth-child(even) {
    background-color: #f2f2f2;
}

/* Estilo para las filas impares */
tr:nth-child(odd) {
    background-color: #fff;
}
h2{
    text-align: center;
}

```

```

/*@media permite consultar el tipo de pantalla que esta accediendo al contenido html y configurar dicho contenido para que se vea bien 'responsive'
  Todo lo que va entre {} es lo que cambia. Por ejemplo si la pantalla es mayor o igual a 350px y menor o igual a 414px 'dispositivos móviles'
  adapta todos los elementos a las propiedades indicadas. Por ejemplo el tamaño de fuente se establece en 14px para todo el elemento body.*/
@media only screen and (max-width: 414px) and (min-width:350px) {
  body {
    font-size: 14px;
  }

  h2 {
    text-align: center;
    padding: 0%;
    margin: 0%;
  }

  td {
    width: 70%;
  }
  .table {
    width: 70%;
  }
}

```

El resultado de la tabla producto sería:

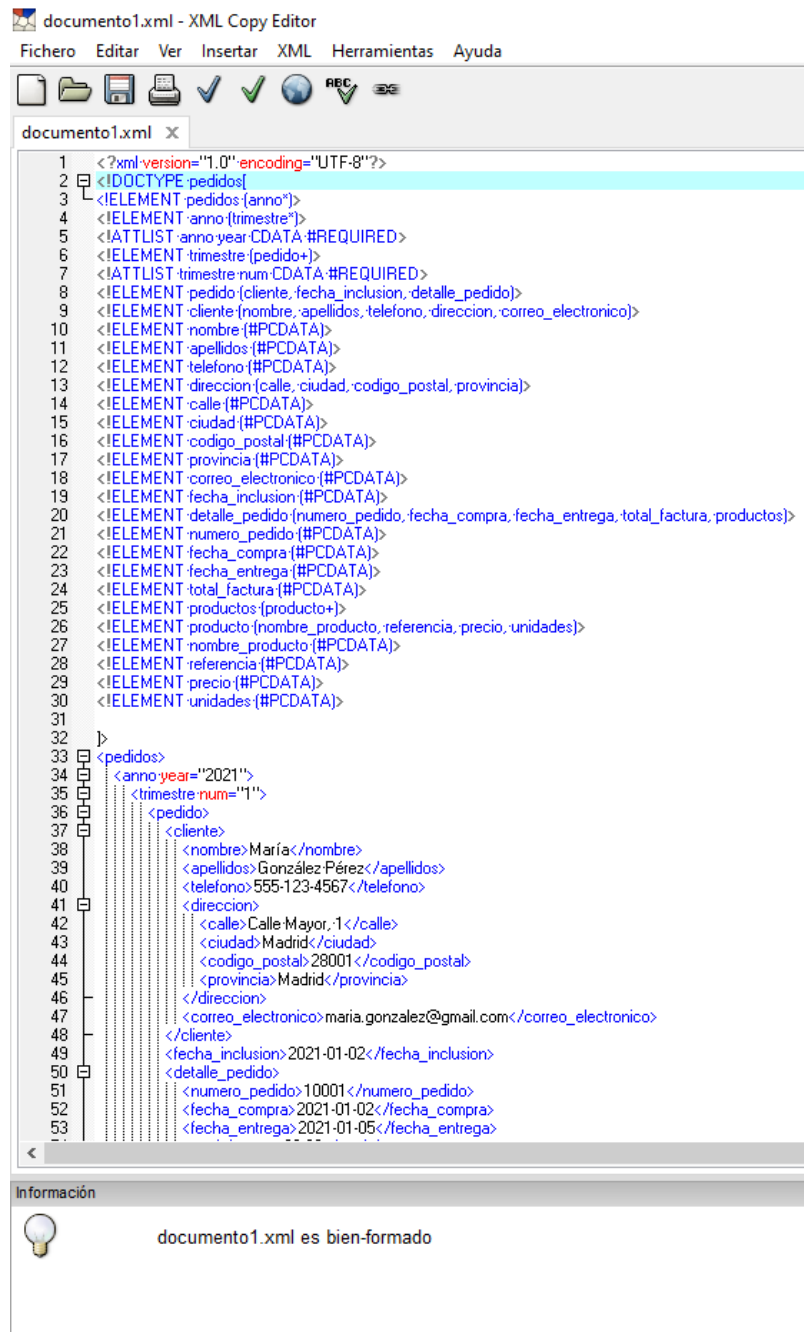
Productos

Producto	Referencia	Precio	Unidades
Camiseta de mujer	ABC123	29.99	1
Zapatos caballero casual planos	ABC156	65.95	1
Zapato cómodo	ABC178	20.99	1
deportivas de niño	ABC200	26.99	1
Vestido camisero con cinturón para anudar	ZSD209	19.99	1
deportivo de trekking para hombre	ABC180	20.99	1
Pantalón de felpa con puños en el bajo	SDE321	9.99	2
mocasín castellano de piel	OLÑ618	53.99	1
Jersey tricot manga larga acabados en canalé	KGQ671	12.99	1
Pijama camiseta manga corta	GVÑ058	12.99	1
Camiseta algodón orgánico azul marino	ZEW012	14.99	1

Fase 3

Para poder ver si es válido el documento xml lo voy a validar con el xml copy editor

En primer lugar vemos si está bien formado el documento xml:



The screenshot shows the XML Copy Editor interface with the file 'documento1.xml' open. The XML document is displayed with line numbers 1 through 53. The document structure is as follows:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE pedidos[
3   <ELEMENT pedidos-(anno*)>
4   <ELEMENT anno-(trimestre*)>
5   <!ATTLIST anno-year-CDATA #REQUIRED>
6   <ELEMENT trimestre-(pedido*)>
7   <!ATTLIST trimestre-num-CDATA #REQUIRED>
8   <ELEMENT pedido-(cliente, fecha_inclusion, detalle_pedido)>
9   <ELEMENT cliente-(nombre, apellidos, telefono, direccion, correo_electronico)>
10  <ELEMENT nombre-{#PCDATA}>
11  <ELEMENT apellidos-{#PCDATA}>
12  <ELEMENT telefono-{#PCDATA}>
13  <ELEMENT direccion-{calle, ciudad, codigo_postal, provincia}>
14  <ELEMENT calle-{#PCDATA}>
15  <ELEMENT ciudad-{#PCDATA}>
16  <ELEMENT codigo_postal-{#PCDATA}>
17  <ELEMENT provincia-{#PCDATA}>
18  <ELEMENT correo_electronico-{#PCDATA}>
19  <ELEMENT fecha_inclusion-{#PCDATA}>
20  <ELEMENT detalle_pedido-(numero_pedido, fecha_compra, fecha_entrega, total_factura, productos)>
21  <ELEMENT numero_pedido-{#PCDATA}>
22  <ELEMENT fecha_compra-{#PCDATA}>
23  <ELEMENT fecha_entrega-{#PCDATA}>
24  <ELEMENT total_factura-{#PCDATA}>
25  <ELEMENT productos-(producto*)>
26  <ELEMENT producto-(nombre_producto, referencia, precio, unidades)>
27  <ELEMENT nombre_producto-{#PCDATA}>
28  <ELEMENT referencia-{#PCDATA}>
29  <ELEMENT precio-{#PCDATA}>
30  <ELEMENT unidades-{#PCDATA}>
31 ]>
32 <pedidos>
33   <anno year="2021">
34     <trimestre num="1">
35       <pedido>
36         <cliente>
37           <nombre>María</nombre>
38           <apellidos>González-Pérez</apellidos>
39           <telefono>555-123-4567</telefono>
40           <direccion>
41             <calle>Calle Mayor, 1</calle>
42             <ciudad>Madrid</ciudad>
43             <codigo_postal>28001</codigo_postal>
44             <provincia>Madrid</provincia>
45           </direccion>
46           <correo_electronico>maria.gonzalez@gmail.com</correo_electronico>
47         </cliente>
48         <fecha_inclusion>2021-01-02</fecha_inclusion>
49         <detalle_pedido>
50           <numero_pedido>10001</numero_pedido>
51           <fecha_compra>2021-01-02</fecha_compra>
52           <fecha_entrega>2021-01-05</fecha_entrega>
53         </detalle_pedido>
54       </pedido>
55     </trimestre>
56   </anno>
57 </pedidos>
```

The bottom of the window shows an 'Información' (Information) panel with a lightbulb icon and the message: 'documento1.xml es bien-formado' (documento1.xml is well-formed).

Y por último pero no menos importante le daremos a que valide el documento para saber si es válido :

```
documento1.xml X
1  <?xml:version="1.0" encoding="UTF-8"?>
2  <!DOCTYPE pedidos[
3  <ELEMENT pedidos-{anno}*>
4  <ELEMENT anno-{trimestre}*>
5  <!ATTLIST anno-year-CDATA #REQUIRED>
6  <ELEMENT trimestre-{pedido}*>
7  <!ATTLIST trimestre-num-CDATA #REQUIRED>
8  <ELEMENT pedido-{cliente, fecha_inclusion, detalle_pedido}>
9  <ELEMENT cliente-{nombre, apellidos, telefono, direccion, correo_electronico}>
10 <ELEMENT nombre-{#PCDATA}>
11 <ELEMENT apellidos-{#PCDATA}>
12 <ELEMENT telefono-{#PCDATA}>
13 <ELEMENT direccion-{calle, ciudad, codigo_postal, provincia}>
14 <ELEMENT calle-{#PCDATA}>
15 <ELEMENT ciudad-{#PCDATA}>
16 <ELEMENT codigo_postal-{#PCDATA}>
17 <ELEMENT provincia-{#PCDATA}>
18 <ELEMENT correo_electronico-{#PCDATA}>
19 <ELEMENT fecha_inclusion-{#PCDATA}>
20 <ELEMENT detalle_pedido-{numero_pedido, fecha_compra, fecha_entrega, total_factura, productos}>
21 <ELEMENT numero_pedido-{#PCDATA}>
22 <ELEMENT fecha_compra-{#PCDATA}>
23 <ELEMENT fecha_entrega-{#PCDATA}>
24 <ELEMENT total_factura-{#PCDATA}>
25 <ELEMENT productos-{producto}*>
26 <ELEMENT producto-{nombre_producto, referencia, precio, unidades}>
27 <ELEMENT nombre_producto-{#PCDATA}>
28 <ELEMENT referencia-{#PCDATA}>
29 <ELEMENT precio-{#PCDATA}>
30 <ELEMENT unidades-{#PCDATA}>
31 ]>
32 <pedidos>
33   <anno year="2021">
34     <trimestre num="1">
35       <pedido>
36         <cliente>
37           <nombre>María</nombre>
38           <apellidos>González-Pérez</apellidos>
39           <telefono>555-123-4567</telefono>
40           <direccion>
41             <calle>Calle Mayor, 1</calle>
42             <ciudad>Madrid</ciudad>
43             <codigo_postal>28001</codigo_postal>
44             <provincia>Madrid</provincia>
45           </direccion>
46           <correo_electronico>maria.gonzalez@gmail.com</correo_electronico>
47         </cliente>
48         <fecha_inclusion>2021-01-02</fecha_inclusion>
49         <detalle_pedido>
50           <numero_pedido>10001</numero_pedido>
51           <fecha_compra>2021-01-02</fecha_compra>
52           <fecha_entrega>2021-01-05</fecha_entrega>
53         </detalle_pedido>
54       </pedido>
55     </trimestre>
56   </anno>
57 </pedidos>
```

Información



documento1.xml is valid



Webgrafía

La web que he utilizado para poner los nombres de los productos y los precios es esta

<https://www.venca.es/e/1595/zapatillas-y-bambas-hombre>