



Práctica 12 – Programación de XML con DOM

El objetivo ahora es aprender a procesar documentos XML a través de DOM y algún lenguaje de programación. Para poder llevar a cabo esta práctica, retoma los conceptos teóricos de las notas “Análisis Sintáctico de XML con DOM” y apóyate de la documentación de la clase DOMDocument de PHP que puedes encontrar en **php.net**.

Sitios de interés:

- Cargar XML desde una cadena:
<https://www.php.net/manual/es/domdocument.loadxml.php>
- Cargar XML desde un archivo:
<https://www.php.net/manual/es/domdocument.load.php>
- Ejemplo sobre carga y obtención de contenido:
https://www.w3schools.com/php/php_xml_dom.asp
- Obtención de elementos (por nombre de etiqueta):
<https://www.php.net/manual/es/domdocument.getelementsbytagname.php>
- Obtención de atributos:
<https://www.php.net/manual/es/domelement.getattribute.php>

1. Carga y lectura de un XML.

Nuevamente retoma el documento *10-catalogovod.xml* proporcionado en la Práctica 10 y utilízalo para programar en PHP un script que lo procese y muestre el contenido en una página XHTML.

- a. Se debe agregar un logotipo al inicio de la página.
- b. Se deben agregar algunas reglas CSS simples para darle formato al contenido.
- c. La lista de películas y programas, que se deben mostrar por perfil, deben estar tabuladas. El estilo (CSS) de las tablas, se deja a tu consideración.
- d. El resto de la información se puede estructurar con los elementos HTML de tu elección.

Puedes tomar como referencia el resultado de la transformación XSL de la Práctica 11.

2. Carga y modificación de un XML.

Programa en PHP un script que cargue el archivo *10-catalogovod.xml* y permita agregar un nodo perfil que contenga tus datos y tus gustos considerando la siguiente información:

- a. Un nodo de películas con al menos 2 géneros.
- b. Cada género de películas debe tener al menos dos títulos.
- c. Un nodo de programas con al menos 2 géneros.
- d. Cada género de películas debe tener al menos dos títulos.

El script debe permitir almacenar en un archivo el XML resultante.



3. Carga y validación de un XML.

Una vez almacenado el XML resultante del ejercicio anterior, utiliza el siguiente código para validarlo y obtener los errores en caso de que existan:

```
libxml_use_internal_errors(true);

$xml= new DOMDocument();

$documento = file_get_contents('ruta_del_xml_a_validar');

$xml->loadXML($documento, LIBXML_NOBLANKS);
// Or load if filename required

$xsd = 'ruta_de_xsd_de_Validacion';

if (!$xml->schemaValidate($xsd))
// Or schemaValidateSource if string used.
{
    $errors = libxml_get_errors();
    $noError = 1;
    $lista = '';

    foreach ($errors as $error)
    {
        $lista = $lista.'['.($noError++).']: '.$error->message.' ';
    }

    echo $lista;
}
```