

Acceso a Datos	CÓDIGO	AD2DAMUT1A1
UT1. Capítulo 1 – Manejo de Ficheros – EJERCICIOS ENTREGABLE OBLIGATORIO (xml , xsl, html)	FECHA	05/11/2025
	VERSIÓN	v.02

## Enunciado: Catálogo de videojuegos

Crear un programa en java que sea capaz de:

- Recoger los datos desde un fichero .txt que contiene toda la información asociada a los videojuegos.
- Crear un .xml a con las api's vistas en la clase con DOM, que contengan toda la información de ese .txt y la structure dentro del fichero .xml con el siguiente formato:

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <videojuegos>
3   <videojuego id="vg001">
4     <titulo>Super Mario Bros.</titulo>
5     <anio>1985</anio>
6     <compania>Nintendo</compania>
7     <consola>NES</consola>
8     <puntuacion>9.8</puntuacion>
9   </videojuego>
10  <videojuego id="vg002">
11    <titulo>The Legend of Zelda: Ocarina of Time</titulo>
12    <anio>1998</anio>
13    <compania>Nintendo</compania>
14    <consola>N64</consola>
15    <puntuacion>9.6</puntuacion>

```

- Opción de añadir una etiqueta llamada <recomendacion> que esté contenida dentro de <videojuego> que contenga un carácter (S/N, S en el caso de sí, y N, en el caso de No). Esa recomendación debe agregarse a todos los videojuegos. Y por defecto, se establecerá el valor S. Esta opción tendrá que modificar ese fichero .xml para agregar la etiqueta <recomendación>
- Mostrar un menú en el que pregunte el juego que **NO** desea recomendar, el usuario introducirá por teclado el id del videojuego. (Ejemplo: vg002) y entonces, el valor del atributo <recomendacion> cambiará a "N". Obviamente también se modificarán los cambios en el fichero .xml.

```

<videojuego id="vg095">
  <titulo>Death Stranding 2: On the Beach</titulo>
  <anio>2025</anio>
  <compania>Kojima Productions</compania>
  <consola>PS5, PC</consola>
  <puntuacion>9.0</puntuacion>
  <recomendacion>N</recomendacion>
</videojuego>

```

- Leer con a través de la api vista en clase SAX el fichero .xml para que lo recorra de arriba hasta abajo, y cuando encuentre la etiqueta <recomendación> cuyo contenido una N, almacene el valor de su etiqueta hermana <titulo> con la finalidad de que al acabar de recorrer el fichero .xml, muestre aquellos títulos que **NO** son recomendados.
- Deberá crear un .html adicional que se llame "catalogo\_recomendado\_top.html", el cual mostrará una tabla con los juegos que **SÍ** estén recomendados, y además los mostrará por orden de mayor a menor puntuación.

### Lista de Videojuegos

ID	Título	Año	Compañía	Consola	Puntuación
vg085	Skyrim	2011	Bethesda	PC	10
vg002	The Legend of Zelda: Ocarina of Time	1998	Nintendo	N64	9.9
vg044	Ghost of Tsushima	2020	Sucker Punch Productions	PS4	9.7
vg094	Split Fiction	2025	Kepler Interactive	PS5, Xbox Series X/S, PC, Switch 2	9.6
vg056	Half-Life 2	2004	Valve	PC	9.5

**Consejos:**

- Consultar los métodos vistos en clase.
- Revisar las diapositivas de clase, también los métodos de trabajo con .xml de DOM.



¿Y para leer un XML?

## FICHEROS XML

- ✓ Node getChild()
- ✓ Node getNextSibling()
- ✓ Node getParentNode()
- ✓ NodeList getChildNodes()
- ✓ int getLength()
- ✓ String getNodeValue()
- ✓ boolean hasAttributes()
- ✓ void setAttribute(String name, String value)
- ✓ Node appendChild(Node hijoNuevo)
- ✓ Node insertBefore(Node hijoNuevo, Node hijoReferencia)
- ✓ Node removeChild(Node hijoViejo)
- ✓ Node replaceChild(Node hijoNuevo, Node hijoViejo)

<https://docs.oracle.com/javase/7/docs/api/org/w3c/dom/package-summary.html>

- Recuerda cuando encuentres el contenido del atributo con el que interesa trabajar guardarlo en un string, y poder crear un array de string para posteriormente recorrerlo y mostrar su contenido.
- Para la entrada de datos por teclado como : “vg064”, o la S/N de recomendación se puede hacer empleando scanners, el programa de java no requiere de ningún tipo de datos como argumento, o parámetro de entrada.
- Ten a mano los programas en .java realizados en clase y que están colgados en Google Drive