

ACCESO A DATOS DISTANCIA

UD1: MANEJO DE FICHEROS

ENUNCIADO 18: Leer XML

Escribe el código necesario para poder leer un documento XML. En este caso será el fichero clase.xml que lo situaremos en el paquete Datos y que contiene la siguiente estructura

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<clase>
  <alumno numero = "393">
    <nombre> Luis </nombre>
    <apellido> Luna </apellido>
    <apodo> Na </apodo>
    <marcas> 85 </marcas>
  </alumno>
  <alumno numero = "493">
    <nombre> Antonio </nombre>
    <apellido> Alvarez </apellido>
    <apodo> Avez </apodo>
    <marcas> 95 </marcas>
  </alumno>
  <alumno numero = "593">
    <nombre> Juan </nombre>
    <apellido> Juz </apellido>
    <apodo> jazz </apodo>
    <marcas> 90 </marcas>
  </alumno>
</clase>
```

ENUNCIADO 19: Leer XML

En este ejercicio se visualizar el mismo fichero que el enunciado 18 pero mostrando los datos por empleado en horizontal. Nota del profesor:” *Preferiblemente prefiero este criterio de visualización en horizontal. Que no te importe si no está bien tabulado. Es tema de la consola*”

ENUNCIADO 20: Leer XML con elementos con un número de atributos variable y elementos opcionales.

Escribe el código necesario para poder leer un documento XM . En este caso será el fichero clase2.xml que lo situaremos en el paquete Datos y que contiene la siguiente estructura:

ACCESO A DATOS DISTANCIA

UD1: MANEJO DE FICHEROS

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<clase>
  <alumno numero = "393" dni="11111111A">
    <nombre> Luis </nombre>
    <apellido> Luna </apellido>
    <apodo> Na </apodo>
    <marcas> 85 </marcas>
  </alumno>
  <alumno numero = "493">
    <nombre> Antonio </nombre>
    <apellido> Alvarez </apellido>
    <apodo> Avez </apodo>
    <marcas> 95 </marcas>
  </alumno>
  <alumno numero = "593">
    <nombre> Juan </nombre>
    <apellido> Juz </apellido>
    <apodo> jazz </apodo>
    <marcas> 90 </marcas>
    <admitido/>
  </alumno>
</clase>
```

En el punto 1: observa como no todos los alumnos tienen el mismo número de atributos.
En el punto 2: observa que hay etiquetas que no están en todos los alumnos.

ENUNCIADO 21: Escribir XML

Escribe un programa que permita escribir un fichero XML (Ejercicio21.xml) utilizando DOM. El fichero tendrá este formato.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?><root>
  <elemento1 id="valor del atributo"/>
  <elemento2>Contenido del elemento 2</elemento2>
</root>
```

ACCESO A DATOS DISTANCIA

UD1: MANEJO DE FICHEROS

ENUNCIADO 22: Escribir XML con un número de atributos variable y con subelementos.

Escribe un programa que permita escribir un fichero XML (Ejercicio22.xml) utilizando DOM. El fichero final tendrá que ser idéntico al clase2.xml que ya hemos leído en el enunciado 20.

Nota: En la solución propuesta no se hace todo el ejercicio pero se indican las pautas para poder crearlo. A mayores se da la posibilidad de utilizar o no un método crearElemento que podría simplificar el código elaborado. Este método sigue tal que así:

```
private static void crearElemento(String dato, String valor, Element raiz, Document document) {  
    Element elem = document.createElement(dato);  
    elem.setTextContent(valor);  
    raiz.appendChild(elem);  
}
```

El primer parámetro es el nombre de la etiqueta a crear, el segundo el valor que tendría (null si es vacío), el tercer parámetro cual es su elemento padre y por último el documento con el que trabajamos.

ENUNCIADO 23: Leer SAX

Escribe un programa que permita leer el fichero clase.xml y obtener esta salida.

```
-----  
Comienzo del documento XML  
Numero :393      nombre: Luis      apellido: Luna      apodo: Na      marcas: 85  
Numero :493      nombre: Antonio      apellido: Alvarez      apodo: Avez      marcas: 95  
Numero :593      nombre: Juan      apellido: Juz      apodo: jazz      marcas: 90  
Total Alumnos: 3
```

ENUNCIADO 24: Leer STAX. API CURSOR

Escribe un programa que permita leer el fichero clase.xml y obtener esta salida.

```
numero393      nombre: Luis      apellido: Luna      apodo: Na      marcas: 85  
numero493      nombre: Antonio      apellido: Alvarez      apodo: Avez      marcas: 95  
numero593      nombre: Juan      apellido: Juz      apodo: jazz      marcas: 90  
Total Alumnos 3
```

Nota: no se ha echo un ejemplo con API event porque la idea de leer un xml es la misma.

ACCESO A DATOS DISTANCIA

UD1: MANEJO DE FICHEROS

ENUNCIADO 25: Crear un XML. JAXB. PROCESO MARSHALL

Escribe un programa que permita crear un fichero empleadosjaxb.xml a partir del empleadosjaxb.xsd que se te proporciona. El fichero tendrá esta estructura.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<empleados>
  <empleado>
    <id>100</id>
    <nombre>Patricia</nombre>
    <dep>88</dep>
    <salario>1000.1</salario>
  </empleado>
  <empleado>
    <id>200</id>
    <nombre>Lucia</nombre>
    <dep>99</dep>
    <salario>1000.2</salario>
  </empleado>
</empleados>
```

```
JAXBContext contexto = JAXBContext.newInstance(Empleados.class);
Marshaller marshallerObj = contexto.createMarshaller();
marshallerObj.marshal(empleados, new File("./src/Datos/empleadosjaxb.xml"));
```

Recuerda que previamente debes crear un enlace jaxb para poder trabajar con él. Debes crear una lista en memoria y posteriormente con el marshall generar el fichero.

ENUNCIADO 26: Leer un XML. JAXB. PROCESO UNMARSHALL

Lee el fichero previamente creado. Para ello realizarás el proceso UNMARSHALL que pasa el fichero a memoria y lo lee.

```
JAXBContext contexto=JAXBContext.newInstance(Empleados.class);
Unmarshaller u=contexto.createUnmarshaller();
Empleados listaEmpleados = (Empleados) u.unmarshal(new File ("./src/Datos/empleadosjaxb.xml"));
```

ENUNCIADO 27: Operaciones con JAXB. Insertar, Borrar, Modificar

Realiza un programa que a través de un menú permita operar con el fichero empleadosjaxb.xml y permita insertar un empleado, borrar un empleado y modificar un empleado.