

JSON

UT1

Bases de Datos XML

JSON (JavaScript Object Notation)

- Para el intercambio de ficheros/información entre diferentes sistemas/formatos/fabricantes, XML puede ser una opción, en ocasiones compleja.
- ***JSON es un formato ligero de intercambio de datos independiente de los lenguajes de programación.***
 - ❑ Tiene formato de texto plano, sencillo de generar, leer y escribir.
 - ❑ Ocupa menos espacio que el XML
 - ❑ Muchos lenguajes lo “soportan”: Java, C++, .NET, PHP, Ruby...
 - ❑ Sirve para representar objetos en el lado del cliente

Bases de Datos XML

- Se puede usar si una aplicación Java, necesita atender peticiones representadas en JSON, tratar los datos y devolver una respuesta en JSON.
- Es un ejemplo de lo que hacen los servicios REST o los webSockets .

Notación:

- ✓ **Object-clase** => Empieza con llave de apertura y finaliza con llave de cierre. Las propiedades se separan por comas, y su nombre y su valor se separan por dos puntos (:)

```
{nombre:"Juan",  
telefono:"6666666666"}
```
- ✓ **Arrays** => Las colecciones ordenadas de objetos comienzan y finalizan por corchetes y se separan igualmente por comas.
- ✓ Las formas String, boolean, etc... similares a Java.

Bases de Datos XML

Ejemplos:

XML:

```
new-document 1  EmpleJson  new-document 2  cons
1 <Empleados>
2   <empleado>
3     <id>1</id>
4     <nombre>David</nombre>
5     <dep>10</dep>
6     <salario>501</salario>
7   </empleado>
8   <empleado>
9     <id>2</id>
10    <nombre>Eduardo</nombre>
11    <dep>2</dep>
12    <salario>2.0</salario>
13  </empleado>
14 </Empleados>
```

JSON:

```
New  New XQuery  Open  Save  Close  Eval  Run
new-document 1  EmpleJson  new-document 2  consulta3*  consulta2*
1 { "Empleados": [
2   { "empleado": { "id": "1", "nombre": "David", "dep": "10", "salario": "501" } },
3   { "empleado": { "id": "2", "nombre": "Ana", "dep": "20", "salario": "600" } }
4 ] }
```

Bases de Datos XML

¿Cómo se puede usar JSON?:

- ❖ Mediante librerías (Javascript...)
- ❖ Transformando Objetos a cadenas de texto y viceversa
- ❖ Personalizando las transformaciones

- *Existen muchas librerías de terceros que permiten la serialización de ficheros JSON. En esta Unidad, utilizaremos **GSON**, que es de Google.*