

ACTIVIDAD PRÁCTICA 6 (AP6)

Título

Introducción a JSON

Objetivos

- Conocer la sintaxis JSON como alternativa complementaria a XML en el intercambio de datos, probando la conversión de datos entre formatos, especialmente entre XML y JSON.
- Introducirse en el conocimiento del OpenData (Datos Abiertos)
- Conocer la manipular de objetos JSON mediante ejemplos de uso y casos prácticos.
- Saber crear documentos JSON válidos con una sintaxis correcta y óptima.

Temporalización

Previsto: Tres semanas aproximadamente.

Proceso de desarrollo

INTRODUCCIÓN.

JSON significa **JavaScript Object Notation**. JSON almacena e intercambia información de texto, de una manera similar a XML. Ocupa menos espacio que **XML**, además de ser más rápido y más fácil de analizar.

JSON está desplazando al **XML** en las aplicaciones web, debido a que los servicios online importantes (p.e. Foursquare o Twitter –actualmente X-) han orientado sus APIs hacia este formato, que se ha convertido en el estándar, debido a su facilidad de uso, portabilidad y legibilidad. Por el contrario, XML es necesario y más apropiado en sistemas que requieren una mayor fiabilidad de los datos y donde su corrección/validez (comprobable a través de XSD) sea de gran relevancia.

JSON se considera un subconjunto de la notación literal para representar objetos, arrays, cadenas, booleanos y números en Javascript. Su sintaxis específica se puede consultar en: <https://www.json.org/json-es.html>

Un ejemplo de sintaxis JSON es:

```
{
  "empleados": [
    { "nombre": "Jorge", "apellido": "Mesa", "edad": 28 },
    { "nombre": "Ana", "apellido": "Sánchez", "edad": 21 },
    { "nombre": "Pedro", "apellido": "Lee", "edad": 44 }
  ]
}
```

*El objeto **empleados** es un arreglo/array de 3 objetos de tipo empleado, que guarda la información del nombre, el apellido y la edad de cada empleado.*

Algunas de las particularidades o reglas del formato JSON a tener en cuenta son:

- Los pares nombre-valor van siempre delimitados por comillas, independientemente de si se tratan de nombres válidos en Javascript que podrían aparecer sin ellas.
- JSON puede representar seis tipos de valores: objetos, arrays, números, cadenas/strings, booleanos y null.
- Las fechas no son reconocidas como un tipo de objeto propio.
- Los números en JSON no pueden ir precedidos de ceros salvo en el caso de notación decimal (ejemplo: 0.001).
- JSON es considerado un lenguaje independiente, aunque su origen esté basado en el uso de objetos nativos en lenguaje JavaScript.

El planteamiento para trabajar este bloque temático será desarrollar las siguientes **CUATRO TAREAS**.

TAREA 1)

VISUALIZACION DE VIDEOS EXPLICATIVOS DE BLOQUE TEMÁTICO.

Visualiza, en el orden establecido, los siguientes videos disponibles en la sección de RECURSOS DIDÁCTICOS
→ Transparencias / videos explicativos del bloque temático:

- [Video 1](#). Introducción a JSON (Conceptos básicos).
- [Video 2](#). Ejemplos de JSON y equivalencias con XML.
- [Video 3](#). Utilización de conversores JSON / XML
- [Video 4](#). Utilización de validadores de JSON.
- [Video 5](#). Utilización de editores online para JSON.
- [Video 6](#). Ejemplo 1 - utilización formato JSON – Servicio WEB (OpenWeatherMap)
- [Video 7](#). Ejemplo 2 - utilización formato JSON – Servicio WEB (Aemet OpenData)
- [Video 8](#). Ejemplo 3 - utilización formato JSON – Servicio WEB (datos abiertos – Gobierno de España)
- [Video 9](#). Ejemplos Básicos de uso de JSON (análisis del código).

En la misma sección puedes descargar las transparencias utilizadas en los vídeos explicativos para que puedas seguir los enlaces y links comentados.

Ten en cuenta las siguientes cuestiones y advertencias previas:

- Según vayas realizando la visualización de los **videos 1 al 5** debes de ir probando los ejemplos y herramientas online presentadas (validadores, editores, formateadores, etc...) de tal forma que te familiarices con su conocimiento y uso básico. El objetivo de estos videos es que conozcas la base del formato JSON y conozcas las posibilidades de conversión con respecto a otros formatos (XML, CVS, YALM, etc...).
- Los **videos 6 y 7** presentan diferentes servicios WEB que hacen uso y proporcionan información en formato JSON. Te introducen en el uso y conocimiento de los servicios WEB a través de dos ejemplos reales. Respecto al **video 6**, verás que se analiza un servicio web - OpenWeatherMap – sobre obtención de datos meteorológicos en varios formatos (JSON, XML, etc...). Lamentablemente, aunque en el video os indique que probéis el servicio, lo han actualizado y ahora es de suscripción obligatoria con necesidad de suministrar un medio de pago, por lo que no es necesario que os suscribáis y lo probéis, aunque yo en el video os anime a probarlo. No obstante, no os preocupéis ya que la intención del video 6 es que al visualizarlo podáis lograr un entendimiento básico y conocimiento de las posibilidades que nos proporciona un servicio WEB que sirve y atiende peticiones de datos en diversos formatos (principalmente JSON y XML).
- Los **videos 7 y 8** te introduce en “el fabuloso mundo” de los Datos Abiertos. Es importante el conocimiento y entendimiento de los sistemas basados en Datos Abiertos (OpenData) fundamentales para el desarrollo de aplicaciones y su relación con los formatos de datos (JSON, XML, etc...).
- El **video 9** presenta diferentes ejemplos de uso y accesos al formato JSON en ejemplos básicos de código. Su objetivo es que percibas de una forma global diferentes formas para acceder y consumir información en formato JSON desde un programa / aplicación o página Web.

TAREA 2)

CONVERSIÓN DE ARCHIVOS (XML → JSON)

Recupera tu archivo XML que aportaste a la Actividad de Evaluación 2 (AEV2), donde diseñaste un archivo XML con información de tu Currículum Vitae que debía ser validado por un XSD, también construido por ti.

Partiendo de dicho archivo XML (con la información de tu Currículum Vitae), convierte dicha información a un formato JSON. Para ello debes utilizar alguna de las diversas herramientas de conversión de formatos

analizadas en los videos de la Tarea 1. Puedes probar diversas herramientas de conversión, con sus diferentes opciones para obtener tu JSON transformado, seleccionando aquel resultado que creas más adecuado y óptimo.

El resultado final de la tarea debe contener los dos ficheros trabajados. El primer fichero será tu CV en formato XML y el segundo fichero el resultante de la transformación a JSON. Debes nombrarlos como **cv.xml** y **cv.json**, comprimirlos en un formato ZIP o RAR para subirlos como resultado de la tarea 2. Denomina a este archivo **cv.rar** o **cv.zip**.

TAREA 3)

TRATAMIENTO DE UN CONJUNTO DE DATOS JSON CON JAVASCRIPT.

Realizar una sencilla aplicación en Javascript que visualice por terminal la siguiente información correspondiente al **servicio de estaciones de bicicletas (Valenbisi)** existentes en la ciudad de Valencia:

- Nombre de la estación.
- Dirección de la estación.
- Coordenadas de situación de la estación (longitud y latitud).

Se desea que la aplicación realizada sólo visualice los datos indicados anteriormente para aquellas estaciones que tengan 20 bornes o más. Un borne es un aparcamiento individual para bicicleta.

Además, al finalizar la visualización de la información de las estaciones, la aplicación deberá informarnos también del número total de estaciones con 20 bornes o más existentes en la ciudad.

Para poder resolver el problema se dan las siguientes recomendaciones:

1. Deberás descargar (*botón derecho* → *guardar como...*) el conjunto de datos en formato JSON suministrado en el siguiente enlace ([conjunto de datos valenbisi.json](#)). Dichos datos corresponden a parte de la información ofrecida por el servicio de datos abiertos (<http://datos.gob.es/>) sobre las estaciones del **servicio Valenbisi** en la ciudad de Valencia. Recordarás que se hizo mención a él en el video 8 la tarea 1. Se recomienda llamar al archivo **valenbisi.json**.
2. Crea un archivo **valenbisi.js** donde realices la programación demandada. Con tus conocimientos actuales aprendidos en la asignatura de PROGRAMACIÓN, seguro estarás en disposición de crear el código adecuado. Además, para poder tratar y leer los datos del archivo te proporciono el siguiente código de base del cual podrás partir para presentar tu solución. Si lo copias y ejecutas podrás observar cómo se comporta el código y te servirá de pista para resolver el problema. El archivo de datos **valenbisi.json** deberá estar en el mismo directorio que el fichero fuente (**valenbisi.js**):

```
1  'use strict';
2
3  // Utilización de la librería FileSystem
4  const fs = require('fs');
5  let rawdata = fs.readFileSync('valenbisi.json');
6  // Carga del fichero en un objeto JSON de JavaScript
7  let estaciones = JSON.parse(rawdata);
8
9  // Ejemplo de visualización de todo el conjunto de datos
10 console.log(estaciones);
11 // Ejemplo de visualización de las características de la primera estación
12 console.log(estaciones.features[0]);
13 // Ejemplo de visualización de las coordenadas (posición) de la estación 11.
14 console.log("Coordenada X: ",estaciones.features[10].geometry.coordinates[0]);
15 console.log("Coordenada Y: ",estaciones.features[10].geometry.coordinates[1]);
16 // Ejemplo de visualización del total de bornes de la estación 24.
17 console.log("Bornes: ", estaciones.features[23].properties.total);
```

A modo de ejemplo del resultado final previsto, a continuación, se muestra una posible visualización de las dos últimas estaciones. En una ejecución correcta salen aproximadamente 158 estaciones con 20 bornes o más:

```
ESTACION: 167_PAVIA 3
  Direccion: Pavía - Acequia de la Cadena
  Longitud: 730126.267
  Latitud: 4372896.898

ESTACION: 169_PAVIA 4
  Direccion: Pavía (Instituto Isabel de Villena)
  Longitud: 730110.471
  Latitud: 4373327.171

Cantidad de estaciones con 20 bornes o más: 158
```

TAREA 4)

CREACIÓN DE DOCUMENTOS EN FORMATO JSON

Se propone la creación y desarrollo de diferentes documentos/archivos en formato JSON que recojan y estructuren diversa información. No todos los ejercicios planteados en esta tarea son de obligada entrega. En la sección (entrega de tareas) se especifica cuáles deberás haber completado correctamente y deberás entregar como resultado de tu actividad.

EJERCICIO 1)

Escribe un documento JSON que estructure la siguiente información:

Pedido para el señor Juan Delgado Martínez. El pedido se compone de una bicicleta A2023. A entregar en la calle Barco 4, tercer piso, letra A, el día 19-5-2000.

EJERCICIO 2)

Escribe un documento JSON que representa la estructura y los datos del siguiente diagrama:



Completa el contenido del documento JSON para que el texto completo sea el siguiente (tres párrafos en total):

*¿Hola, qué tal?
Hace mucho tiempo que no escribes. A ver si llamas y quedamos pronto.
Un saludo.*

EJERCICIO 3)

Escribir un JSON que represente la siguiente información sobre la carta del menú de desayunos de un restaurante:

Nombre	Precio (en euros)	Descripción	Calorías
Gofres Belgas	5.95	Dos de nuestros famosos Gofres belgas con abundante sirope	650
Gofres Belgas con fresas	7.95	Ligeros gofres belgas cubiertos de fresas y nata montada	900
Gofres Belgas con frutas del bosque	8.95	Ligeros gofres belgas cubiertos con frutas del bosque y nata montada	900
Tostada Francesa	4.50	Dos gruesas rebanadas de nuestro pan francés casero	600
Desayuno de la casa	6.95	Dos huevos, bacon o salchicha, tostada y patatas fritas	950

EJERCICIO 4)

Escribir un JSON que represente la siguiente información sobre la CDs de oferta de una tienda de música:

CDs EN OFERTA					
Título	Artista	País	Compañía discográfica	Precio	Año de publicación
Still got the blues	Gary Moore	UK	Virgin records	10.20	1990
One night only	Bee Gees	UK	Polydor	10.90	1998
When a man loves a woman	Percy Sledge	USA	Atlantic	8.70	1987
Big Willie style	Will Smith	USA	Columbia	9.90	1997
La canción de Juan Perro	Radio Futura	España	Sony Music-Ariola	9.90	1987
The dock of the bay	Otis Redding	USA	Atlantic	7.90	1987

EJERCICIO 5)

En una planta de envasado de líquidos, disponen de 5 tipos de envases, 3 tipos de contenedores para empaquetar grupos de envases y 10 tipos de líquidos que pueden ser almacenados en los envases. Sin embargo, hay ciertas restricciones en los líquidos que pueden ser almacenados en cada envase y los tamaños de los envases hacen que no se pueda almacenar todos los envases en todos los contenedores.

Tenemos contenedores:

- Grandes, acepta envases grandes
- Medianos, no acepta envases grandes
- Verticales, acepta envases altos

Tenemos envases:

- Grande ancho, válido para alimentos
- Grande alto, válido para alimentos
- Mediano ancho, válido para alimentos
- Mediano alto, resistente a sustancias corrosivas
- Pequeño, resistente a sustancias corrosivas

Tenemos líquidos:

- | | |
|------------|------------|
| • Aguarrás | • Agua |
| • Gasolina | • Leche |
| • Aceite | • Lejía |
| • Vinagre | • Amoniaco |
| • Alcohol | • Jabón |

Para estos líquidos, envases y contenedores, crear tanto un fichero JSON (denominado **envasado.json**), como un fichero XML que representen esta información relativa a lo que puede ser almacenado en cada uno de los tres niveles.

EJERCICIO 6)

Crea un archivo JSON (denominado **superhero.json**) que almacene la siguiente información:

/// SUPERHERO SQUAD ///		
Hometown: Metro City // Formed: 2016		
/// MOLECULE MAN ///	/// MADAME UPPERCUT ///	/// ETERNAL FLAME ///
Secret identity: Dan Jukes	Secret identity: Jane Wilson	Secret identity: Unknown
Age: 29	Age: 39	Age: 1000000
Superpowers:	Superpowers:	Superpowers:
<ul style="list-style-type: none">• Radiation resistance• Turning tiny• Radiation blast	<ul style="list-style-type: none">• Million tonne punch• Damage resistance• Superhuman reflexes	<ul style="list-style-type: none">• Immortality• Heat Immunity• Inferno• Teleportation• Interdimensional travel

EJERCICIO 7)

Escribir un documento JSON (denominado **arboles.json**) que represente la siguiente información sobre diferentes tipos de árboles:

- **Acer monspessulanum**
 - Nombre común: Arce de Montpellier, Arce menor
 - Vegetación: Caducifolio
 - Altura: De 6 a 10 metros
 - Forma y estructura: Copa esférica. Tronco principal recto con bifurcaciones. Ramaje colgante
 - Color en primavera: Haz verde brillante, envés verde blanquecino
 - Resistencia a las heladas: Heladas fuertes (hasta -15°C)
- **Olea europea**
 - Nombre común: Olivo
 - Vegetación: Perenne
 - Altura: De 8 a 15 metros
 - Forma y estructura: Copa irregular. Tronco principal irregular con bifurcaciones. Ramaje tortuoso
 - Color en primavera: Haz verde oscuro, envés verde plateado
 - Resistencia a las heladas: Heladas medias (hasta -10°C)
- **Platanus orientalis**
 - Nombre común: Plátano
 - Vegetación: Caducifolio
 - Altura: De 20 a 25 metros
 - Forma y estructura: Copa ovoidal. Tronco principal recto. Ramaje expandido
 - Color en primavera: Haz verde medio, envés verde claro
 - Color en otoño: Ocre
 - Resistencia a las heladas: Heladas fuertes (hasta -20°C)

- **Quercus ilex**
 - Nombre común: Encina
 - Vegetación: Perenne
 - Altura: En torno a 25 metros
 - Forma y estructura: Copa esférica o elíptica irregular. Tronco principal recto. Ramaje tortuoso
 - Color en primavera: Plateado en hojas jóvenes. En hojas antiguas, haz verde oscuro, envés plateado
 - Resistencia a las heladas: Heladas fuertes (hasta -15°C)

Convierte el documento a su correspondiente formato XML.

Entrega de la tarea

Como resultado final de esta tarea **deberás entregar los ficheros comprimidos especificados a continuación:**

Tarea 2:

- Un fichero comprimido (**cv.rar o cv.zip**) que contendrán tu CV en los dos formatos pedidos (**cv.xml** y **cv.json**)

Tarea 3:

- Un fichero comprimido (**valenbisi.rar o valenbisi.zip**) con los archivos (**valenbisi.json** y **valenbisi.js**).

Tarea 4:

- Un fichero comprimido (**misJson.rar o misJson.zip**) con los archivos (**envasado.json**, **superhero.json** y **arboles.json**).

Evaluación

Esta actividad práctica es una actividad de aprendizaje.

Aunque no es una actividad de evaluación (AEV) como tal, **debe ser entregada y obtener la valoración de APTA.**

En caso de ser valorada como NO APTA, en el feedback del profesor se indicarán los aspectos que deben ser mejorados para volver a realizar la entrega de la actividad una vez subsanados.

IMPORTANTE

La actividad debe ser entregada y valorada como APTA para poder presentarse al examen presencial del 1er cuatrimestre.

Recursos

Disponibles en Recursos didácticos del Bloque 3:

- Videos explicativos utilizados.
- Material de apoyo en FloridaOberta
- Web oficial de JSON: <http://www.json.org/json-es.html>