



# INTRODUCCIÓN A JSON

Primero DAW/DAM

**Descripción breve**

Primera toma de contacto con JSON

**Antº Javier Miras Llamas**

# Tabla de contenido

¿QUÉ ES JSON? .....	2
SINTAXIS JSON .....	3
TIPOS DE DATOS .....	5
JSON Object .....	5
JSON Array .....	6
String.....	7
Number.....	8
true / false .....	9
null .....	9
OBJECT.....	10
JSON vs XML .....	11
COMPRESIÓN.....	12
EJERCICIOS .....	13

# ¿QUÉ ES JSON<sup>1</sup>?

JSON (JavaScript Object Notation) es un estándar utilizado para intercambiar información de forma estructurada. Está en formato texto y su estructura está basada en pares de valores clave, que veremos más adelante en el apartado de sintaxis.

JSON es independiente del lenguaje de programación y se utiliza para comunicaciones, como web services.

Normalmente los ficheros que contiene un JSON terminan con la extensión ".json".

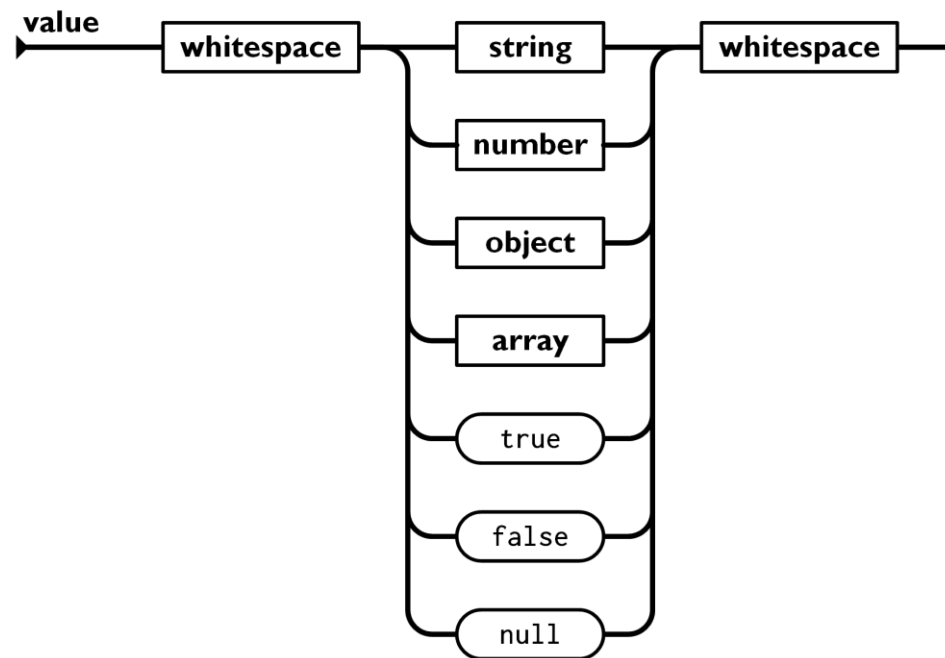
---

<sup>1</sup> Sitio web oficial: <https://json.org/>

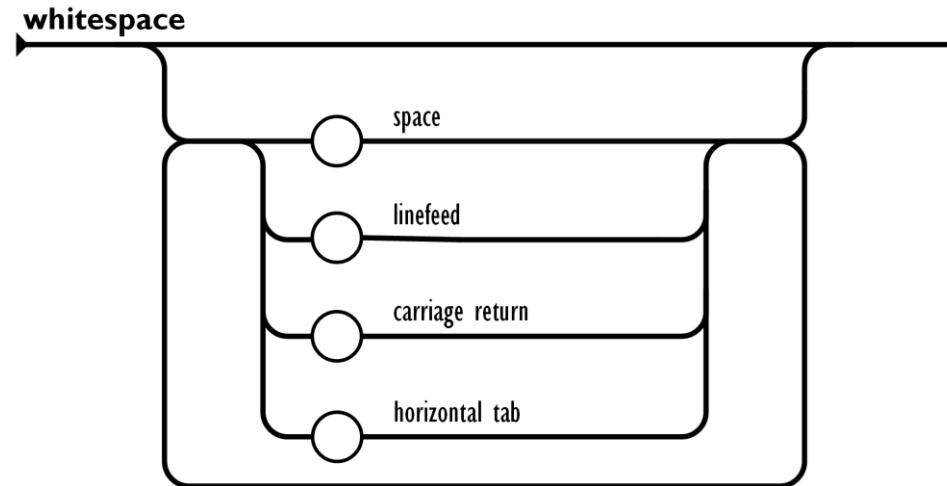
# SINTAXIS JSON

La regla fundamental de la sintaxis JSON es que los datos vienen en pares clave/valor.

La clave será un string entre comillas dobles. Para separar la clave del valor se utilizará los dos puntos ":", y el valor puede ser cualquiera de los indicados en el siguiente punto.



Los espacios en blanco se codifican de la siguiente forma.



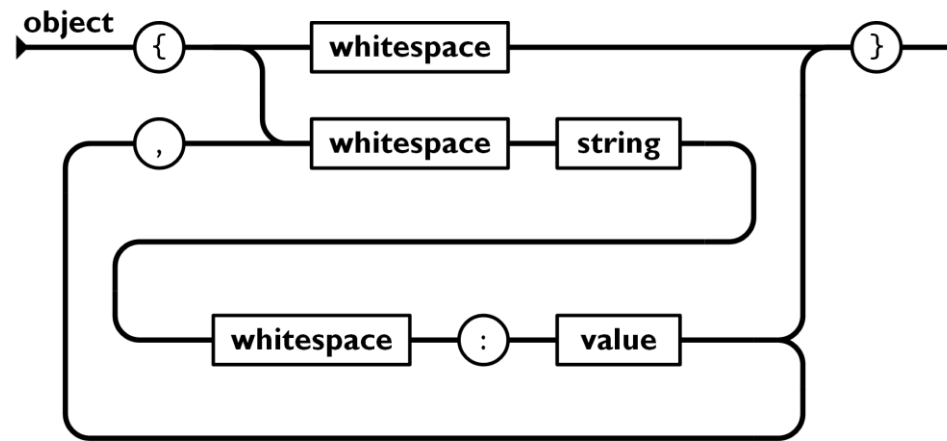
Ejemplo 1. El siguiente ejemplo muestra tres pares clave/valor. Las claves son "nombre", "apellidos" y "edad". Los valores son "Javier", "Miras" y null, respectivamente.

```
"nombre" : "Javier"  
"apellidos" : "Miras"  
"edad" : null
```

## TIPOS DE DATOS

### JSON Object

Un objeto en JSON se representa mediante dos llaves { }.

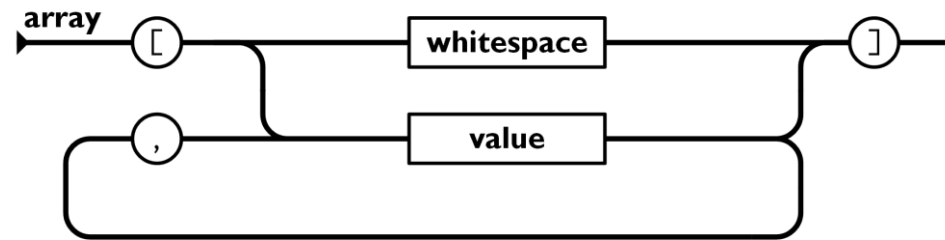


Ejemplo 2. El JSON propuesto es un objeto con tres elementos, cada uno de ellos separados por coma. Cada elemento está formado por las claves: "nombre", "apellidos" y "edad", y cuyos valores son: "Javier", "Miras" y null.

```
{
  "nombre" : "Javier",
  "apellidos" : "Miras",
  "edad" : null
}
```

### JSON Array

Un array en JSON es similar a los arrays de los lenguajes de programación. Son una colección ordenada de valores separados por comas y encerrados entre corchetes [ ].



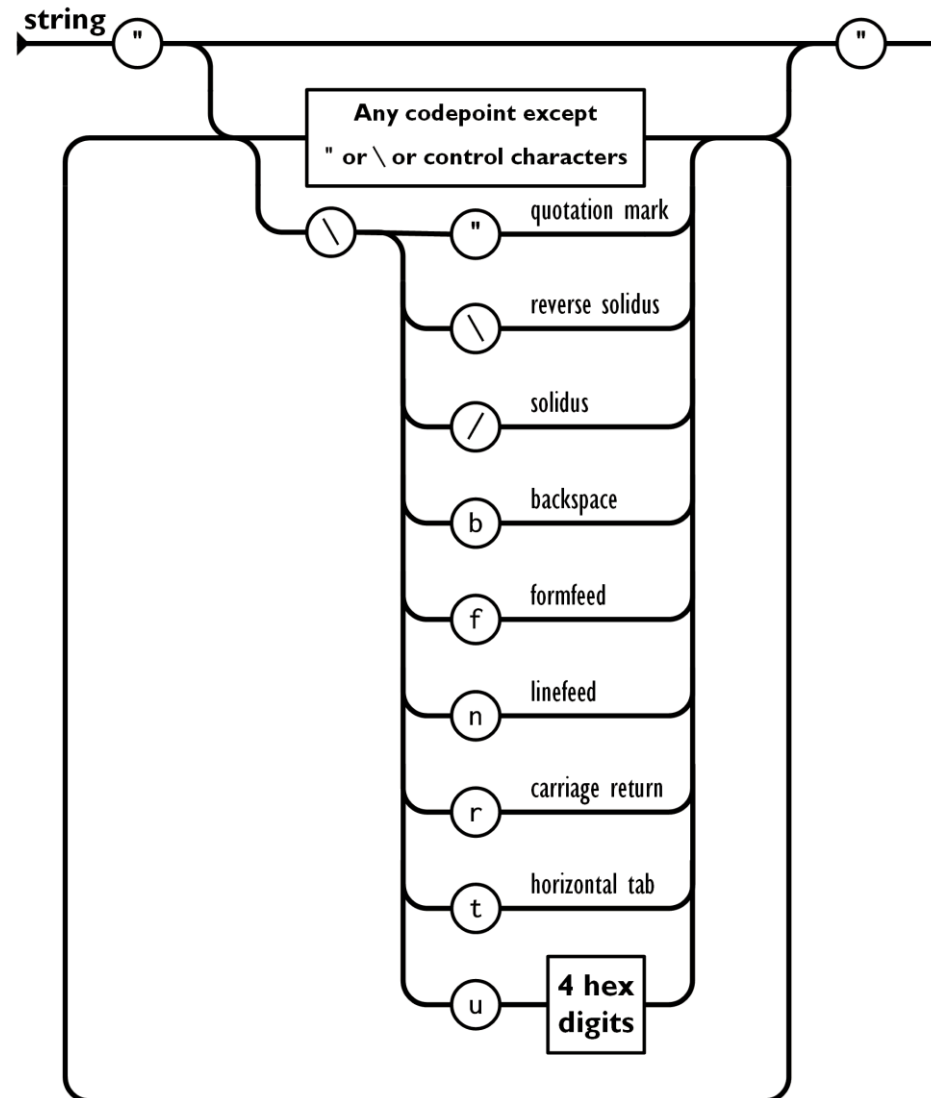
Ejemplo 3. El JSON propuesto es un vector con tres elementos, cada uno de compuesto por un objeto de dos elementos. Cada elemento está formado por las claves: "id" (formato numérico) y "nombre" (formato string).

```
[
  {
    "id" : 17,
    "nombre": "Programación"
  },
  {
    "id" : 19,
    "nombre": "Base de datos"
  },
  {
    "id" : 23,
    "nombre": "Inglés"
  }
]
```

Dentro de un array podremos tener tanto objetos como arrays.

## String

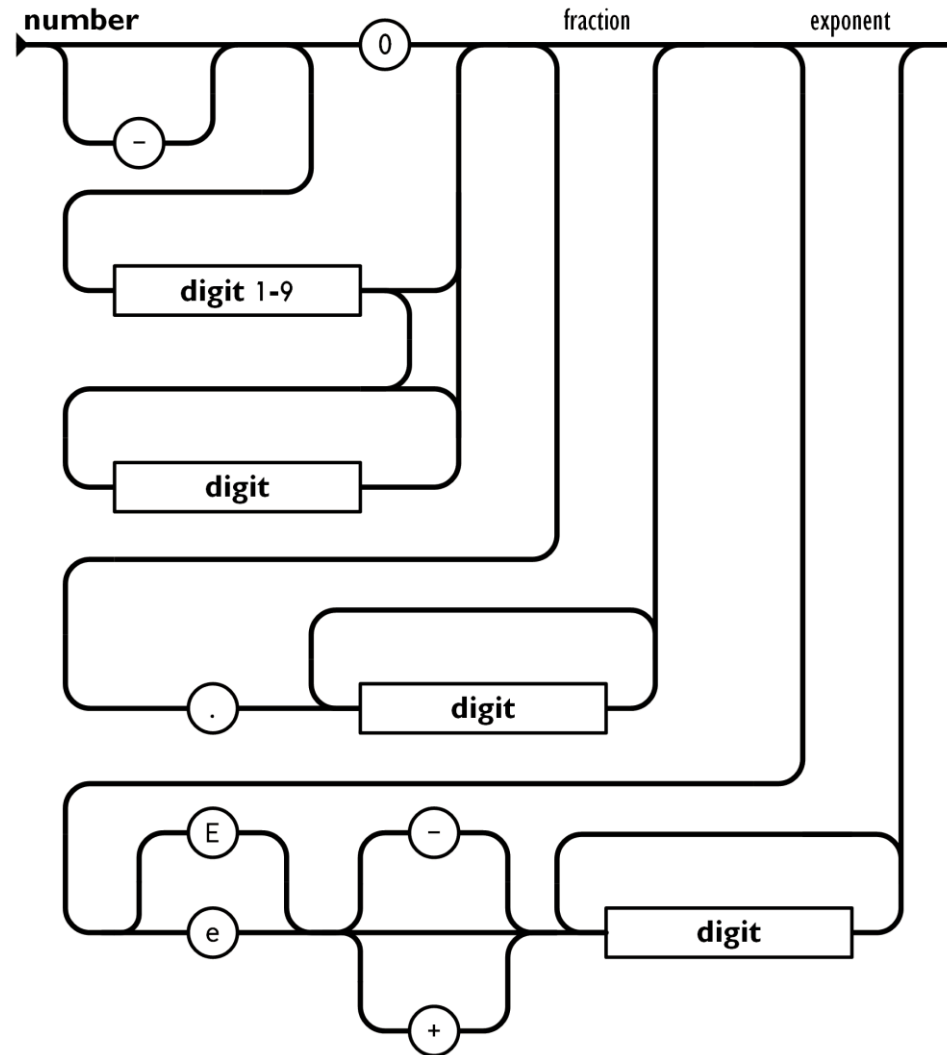
Un string es una cadena de caracteres entre doble comillas. El tipo carácter de los lenguajes de programación se representa como string.





## Number

Cualquier valor numérico decimal. Los formatos octal y hexadecimal no se utilizan en JSON.



Ejemplo 4. El JSON propuesto son tres elementos numéricos

```
{  
  "id" : 17,  
  "precio" : 10.25,  
  "peso" : 10e-5  
}
```

### true / false

Los valores booleanos están representados por true o false.

Ejemplo 5. El JSON propuesto muestra el uso de los valores true/false

```
{  
  "activo" : true,  
  "casado" : false  
}
```

### null

Los valores nulos se representan con null.

Ejemplo 6. El JSON propuesto muestra el uso del valor null.

```
{  
  "hijos" : null  
}
```

# OBJECT

La sintaxis de JSON nos permite mezclar objetos y arrays para representar modelos de datos más complejos.

Ejemplo 7. Observar que en el ejemplo tenemos un objeto y dentro de él tenemos otro objeto "contacto" y un array "asignaturas".

```
{
  "academia" : "IG Formación",
  "contacto" : {
    "telefono" : "965 120 039",
    "email" : "info@igformacion.com"
  },
  "direccion" : "Enriqueta Ortega nº 17",
  "cp" : "03005",
  "asignaturas" : [
    {
      "id" : 17,
      "nombre": "Programación"
    },
    {
      "id" : 19,
      "nombre": "Base de datos"
    },
    {
      "id" : 23,
      "nombre": "Inglés"
    }
  ]
}
```

## JSON vs XML

En la siguiente tabla se muestran las similitudes y diferencias entre JSON y XML

SIMILITUDES	DIFERENCIAS
Tanto JSON como XML son autodescriptivos y fáciles de entender.	XML utiliza etiquetas de cierre, mientras que en JSON no es necesario.
Ambos se pueden utilizar para intercambio de información.	JSON es más rápido.
La mayoría de los programas y lenguajes de guiones soportan tanto JSON como XML.	JSON tiene soporte nativo para los arrays.
Tanto JSON como XML tienen una estructura jerarquizada.	XML soporta atributos y nombre de espacios en los elementos. Esto proporciona más información.
	XML soporta los comentarios.

Ejemplo 8. A continuación, se muestra un ejemplo de información en ambos formatos.

FORMATO JSON	FORMATO XML
<pre>{   "alumnos" : [     {       "matricula" : 125,       "nombre": "José"     },     {       "matricula" : 126,       "nombre": "Josefina"     }   ] }</pre>	<pre>&lt;alumnos&gt;   &lt;alumno id="125"&gt;     &lt;nombre&gt;José&lt;/ nombre &gt;   &lt;/alumno&gt;   &lt;alumno id="126"&gt;     &lt;nombre&gt;Josefina&lt;/ nombre &gt;   &lt;/alumno&gt; &lt;/alumnos&gt;</pre>

# COMPRESIÓN

Cuando el JSON sea muy grande y lo queremos mandar como respuesta a una petición, este se puede comprimir quitando los espacios y saltos de línea.

Ejemplo 9. JSON del Ejemplo 7. Observar que en el ejemplo tenemos un objeto y dentro de él tenemos otro objeto "contacto" y un array "asignaturas". sin espacios y saltos de línea.

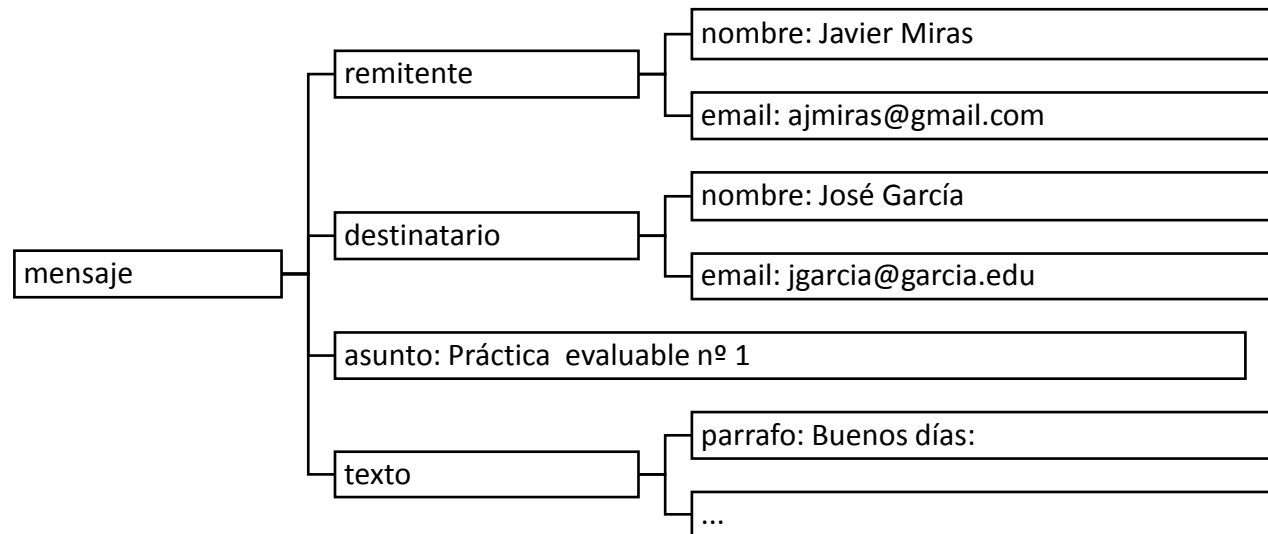
```
{"academia":"IGFormación","contacto":{"telefono":"965120039","email":"info@igformacion.com"},"direccion":"Enriqueta Ortega nº 17","cp":"03005","asignaturas":[{"id":17,"nombre":"Programación"}, {"id":19,"nombre":"Base de datos"}, {"id":23,"nombre":"Inglés"}]}
```

## EJERCICIOS

Ejercicio 1. Identifica las distintas partes del siguiente JSON.

```
{
  "saludo" : "Hola",
  "nombre" : "Pedro",
  "edad" : 35,
  "casado" : true,
  "hijos" : [
    { "nombre" : "Juan", "edad" : 5 },
    { "nombre" : "Juana", "edad" : 6 }
  ],
  "altura" : 1.75
}
```

Ejercicio 2. Crea un JSON que represente la siguiente estructura



Ejercicio 3. Crea aun JSON para mandar las notas de una asignatura. Deberá contener la siguiente información:

- Nombre de la asignatura: "Base de Datos".
- Fecha: actual.
- Evaluación: 1, 2 ó 3.
- Nombre de cada alumno.
- Nota de cada alumno.

Poner la nota de 4 compañeros.