

El formulario con casillas: qué es JSON

Datos de esta lección

Fase	F1 — Cimientos, seguridad y arquitectura del sistema
Módulo	M1.2 — Modelos mentales: radiografía de un agente telefónico
Puntos de profesionalidad al superar el test	15

Objetivo práctico

Al terminar esta lección podrás reconocer visualmente y leer con NVDA un fragmento sencillo de JSON, identificando claves y valores, sin sentir que es un lenguaje de programación inaccesible.

Resultado que obtendrás

Capacidad de leer, sin ayuda, un JSON breve como el que aparece en un Webhook de confirmación de cita, y de identificar qué dato corresponde a qué campo.

Dónde encaja dentro del sistema final

El JSON es el formato en el que casi todos los sistemas de este curso empaquetan la información que se envían entre sí: los datos de un Webhook, la respuesta de una API, la configuración exportada de un Blueprint. Es el «papel» sobre el que está escrita toda comunicación entre piezas del sistema.

Conocimientos previos

- API y Webhook (lecciones anteriores).

Herramientas necesarias

- Un navegador con NVDA activo.
- Opcional: el Bloc de notas de Windows u otro editor de texto simple, para copiar los ejemplos.

Posible coste

Esta lección es conceptual y no requiere ningún gasto.

Modelo mental

Piensa en un formulario de papel con casillas perfectamente identificadas:

Nombre: Juan

Teléfono: 600000000

Fecha: 15 de septiembre

JSON es exactamente esa misma idea —una etiqueta seguida de su valor— pero escrita con un formato muy concreto que los programas pueden leer sin ambigüedad.

Explicación

JSON son las siglas de *JavaScript Object Notation*. Es un formato de texto para representar datos organizados en pares «clave: valor», igual que las casillas de un formulario. Un ejemplo mínimo:

```
{  
  "nombre": "Juan",  
  "telefono": "600000000",  
  "fecha": "2026-09-15"  
}
```

Los elementos que forman un JSON son: las **llaves** { }, que marcan el principio y el final de un objeto (un conjunto de casillas); las **comillas**, que envuelven el texto (tanto las claves como los valores de texto); los **dos puntos**, que separan la clave de su valor; y las **comas**, que separan una casilla de la siguiente. Cuando un valor es una lista de varias cosas —por ejemplo, varios teléfonos— se usan **corchetes** [] en lugar de llaves, y eso se llama un *array*. Un valor puede ser un texto entre comillas (*string*), un número sin comillas, un valor verdadero/falso sin comillas (*booleano*: true o false), u otro objeto anidado dentro (unas llaves dentro de otras llaves).

Para leer JSON con NVDA, la estrategia más eficaz no es intentar «escuchar» la puntuación entera de golpe, sino avanzar línea por línea (con la flecha abajo) y, en cada línea, identificar primero la clave (lo que hay antes de los dos puntos) y después el valor (lo que hay después). No hace falta memorizar los símbolos: basta con reconocer el patrón «clave, dos puntos, valor, coma» repetido.

Vocabulario nuevo

JSON

Piénsalo así: Un formulario de papel con casillas etiquetadas.

JavaScript Object Notation: formato de texto estándar para representar datos organizados en pares clave-valor.

Ejemplo: `{"nombre": "Juan", "telefono": "6000000000"}`

Clave-valor

Piénsalo así: La etiqueta de la casilla («Nombre:») y lo escrito dentro de ella («Juan»).

Par formado por un nombre de campo (clave) y el dato que contiene (valor), la unidad básica de organización de un JSON.

Ejemplo: En `"fecha": "2026-09-15"`, la clave es `"fecha"` y el valor es `"2026-09-15"`.

Array

Piénsalo así: Una lista de varias casillas del mismo tipo, una debajo de otra.

Lista ordenada de valores dentro de un JSON, delimitada por corchetes `[ ]`.

Ejemplo: `"servicios": ["fontanería", "calefacción", "urgencias"]`

Preparación

Descarga «Ejemplos de JSON comentados» desde la página [Recursos](#) de esta plataforma antes de continuar. Es el mismo archivo que se usa con más profundidad en el módulo 3.1 («JSON y Webhooks»).

Procedimiento paso a paso

1. **Contexto:** Vas a descargar el recurso real de esta lección, «Ejemplos de JSON comentados», desde la página de Recursos.

Acción de teclado: Abre el enlace «Recursos» de la navegación principal (o ve a /recursos.php). Pulsa **H** hasta el encabezado de nivel 2 «Ejemplos JSON», y después **Tab** hasta el enlace de descarga. Confirma con **Enter**.

Respuesta esperada de NVDA: NVDA debería anunciar «Ejemplos de JSON comentados, enlace» antes de confirmar, y el navegador debería iniciar la descarga o abrir el archivo al pulsar Enter.

Qué significa: Ya tienes en tu equipo los cuatro ejemplos reales de JSON que se explican en el propio recurso, del más sencillo al más complejo.

Acción siguiente: Abre el archivo descargado (es un archivo de texto Markdown, legible con cualquier editor) y localiza el «Ejemplo 1».

2. **Contexto:** Vas a practicar la lectura línea a línea de un JSON real usando NVDA, siguiendo el método explicado en esta lección.

Acción de teclado: Con el archivo abierto en el Bloc de notas o en el navegador, sitúate al principio del bloque de código del «Ejemplo 1» y pulsa la flecha abajo repetidamente, una línea cada vez.

Respuesta esperada de NVDA: NVDA leerá cada línea completa: por ejemplo, «comillas nombre comillas dos puntos comillas Ana García comillas coma».

Qué significa: En cada línea puedes identificar primero la clave (antes de los dos puntos) y después el valor (después de los dos puntos), sin necesidad de procesar toda la puntuación de golpe.

Acción siguiente: Repite el mismo ejercicio con el «Ejemplo 3» del recurso, que incluye un objeto anidado, y observa cómo cambia la indentación en cada nivel.

Errores frecuentes y diagnóstico

Sentir que un JSON es «ilegible» por la cantidad de símbolos de puntuación.

Cómo localizarlo: Es una reacción habitual al principio; no es un problema de accesibilidad, sino de falta de costumbre con el patrón.

Solución: Leer línea por línea, ignorando de momento llaves y comas, y fijándose sólo en el patrón «clave: valor» de cada línea. La puntuación es importante para que el programa lo entienda, pero no es lo primero que necesitas entender tú.

Comprobación final

Lee este fragmento con NVDA, línea por línea, y di en voz alta cuántas «casillas» (pares clave-valor) contiene: `{"cliente": "Fontanería Ejemplo", "urgente": true, "telefono": "6000000000"}`. La respuesta correcta es tres. Si has llegado a esa cifra identificando cada clave y su valor, la comprobación es positiva.

Qué acabas de conseguir

Puedes leer un JSON sencillo sin sentir que es un obstáculo técnico insalvable, base necesaria para el módulo 3.1 (JSON y Webhooks) y para entender los Blueprints de automatización.

Ejercicio práctico

Escribe (en un editor de texto simple, no necesariamente correcto al cien por cien) un JSON de tres casillas con tu propio nombre, tu ciudad y un número de teléfono ficticio, siguiendo el patrón de los ejemplos de esta lección.

Resumen de lo imprescindible

- JSON representa datos en pares clave-valor, como un formulario con casillas.
- Las llaves { } delimitan un objeto; los corchetes [] delimitan una lista (array).
- Los dos puntos separan clave de valor; las comas separan una casilla de la siguiente.
- La estrategia de lectura con NVDA es avanzar línea por línea, identificando primero la clave y después el valor.

No necesitas aprender todavía

Todavía no necesitas escribir JSON complejo ni anidado a mano: este curso proporciona archivos JSON ya preparados (Blueprints, ejemplos) para cada caso práctico.

Estoy preparado para continuar si puedo...

- Identificar claves y valores en un JSON sencillo.
- Explicar la diferencia entre un objeto (llaves) y un array (corchetes).
- Leer un JSON breve con NVDA, línea por línea, sin sentirlo inaccesible.

Test de comprobación

Para registrar el resultado y obtener Puntos de profesionalidad, realiza este test desde la versión web de la lección. Las opciones se muestran aquí sin señalar la respuesta correcta.

1. En el fragmento `<code>{"nombre": "Ana", "servicios": ["fontanería", "calefacción"]}</code>`, ¿qué representa la parte entre corchetes `<code>[ ]</code>`?

 - a. Un objeto anidado con más pares clave-valor.
 - b. Un array: una lista de varios valores asociados a la misma clave "servicios".
 - c. Un error de formato, porque JSON no admite corchetes.

2. En el fragmento `<code>{"cliente": {"nombre": "Juan", "telefono": "600111222"}}</code>`, ¿qué describe mejor la relación entre la clave "cliente" y lo que aparece después de sus dos puntos?

 - a. El valor de "cliente" es, a su vez, otro objeto completo con sus propias claves y valores: es un objeto anidado.
 - b. Es un error de formato, porque JSON no permite llaves dentro de otras llaves.
 - c. El valor de "cliente" es simplemente el texto literal "nombre".

3. Al leer un JSON con NVDA, ¿cuál es la estrategia recomendada en esta lección para no sentirse abrumado por la puntuación?

 - a. Avanzar línea por línea, identificando primero la clave y después el valor de cada línea, sin intentar procesar toda la puntuación de golpe.
 - b. Memorizar de antemano el significado de cada símbolo antes de leer ningún ejemplo real.
 - c. Pedir siempre a otra persona que lea el JSON en voz alta, porque no es una tarea accesible con lector de pantalla.

Fuentes

- [Introducing JSON — JSON.org \(consultado el 2026-08-30\)](#)

Última actualización de este contenido: **2026-08-30**.