

Academia

Selectores básicos

Anonymous · 08/10/2026

Tabla de contenidos

Selectores básicos	3
<i>Selector simple</i>	4
<i>Selector anidado</i>	5
<i>Selector con OR</i>	6
<i>Selectores con IF ternario</i>	7

Academia » Leer datos de JSON con Selectores » Selectores básicos

Selectores básicos

Anonymous · 08/10/2026

Selectores básicos

Última actualización el 08/10/2026

Academia » Leer datos de JSON con Selectores » Selectores básicos » Selector simple

Selector simple

Selector Simple

Anonymous · 08/10/2026

Selector Simple

Un selector simple accede a un valor directamente por su clave.

- **Formato:** clave
- **Ejemplo:** Para obtener el valor de la clave nombre en { "nombre": "Juan" }, se usa el selector "nombre".

Ejemplos

Acceso Directo a una Clave:

- Dado el JSON: { "nombre": "Ana" }
- Selector: nombre
- Resultado: "Ana"

Acceso Directo a una Clave:

- Dado el JSON: { "cliente_rut": "1-9" }
- Selector: cliente_rut
- Resultado: "1-9"

Este último selector permite extraer los datos del RUT (VAT number en Chile) y asignarlo en un campo en la configuración de BillMySales. En este caso sería en el campo "Identificador fiscal (VAT number)". En la configuración de una pasarela de facturación, sección origen de datos, se vería así:

Campos del cliente

Campo de destino en el documento normalizado

Identificador fiscal

`billing_address.vat_number`

Campo de origen en los datos de Shopify

cliente.rut



i Se puede usar el campo `billing_address.company` existente en los datos de facturación.

Última actualización el 08/10/2026

Academia » Leer datos de JSON con Selectores » Selectores básicos » Selector anidado

Selector anidado

Selector Anidado Los selectores anidados acceden a valores dentro de estructuras JSON anidadas.

Anonymous · 08/10/2026

Selector Anidado

Los selectores anidados acceden a valores dentro de estructuras JSON anidadas.

- **Formato:** clave1.clave2...claveN
- **Ejemplo:** Para acceder al valor en { "grupo": { "subgrupo": { "clave": "valor" } } }, se usa el selector "grupo.subgrupo.clave".

Ejemplos

Acceso a una Clave Anidada:

- **Dado el JSON:** { "usuario": { "nombre": "Carlos", "edad": 30 } }
- **Selector:** usuario.nombre
- **Resultado:** "Carlos"

Selector Anidado:

- **Dado el JSON:** { "cliente": { "rut": "1-9" } }
- **Selector:** cliente.rut
- **Resultado:** "1-9"

Última actualización el 08/10/2026

Academia » Leer datos de JSON con Selectores » Selectores básicos » Selector con OR

Selector con OR

Selectores con OR

Anonymous · 08/10/2026

Selectores con OR

Permiten especificar múltiples opciones, retornando el valor de la primera opción no vacía.

- **Formato:** selector1||selector2
- **Ejemplo:** "claveInexistente||claveExistente" devuelve el valor de claveExistente si claveInexistente no existe o su valor es vacío.

Ejemplos

Selector con Alternativa:

- **Dado el JSON:** {"nombre": "Laura", "apodo": "Lau"}
- **Selector:** nombreInexistente||apodo
- **Resultado:** "Lau"

Cadena Literal como Alternativa:

- **Dado el JSON:** {"nombre": "Pedro"}
- **Selector:** apodo||"Desconocido"
- **Resultado:** "Desconocido"

Selector con Valor por Defecto:

- **Selector:** (selector_inexistente)||"Valor por defecto"
- **Resultado:** "Valor por defecto"

Nota

Aunque los operadores como OR ofrecen gran flexibilidad, su uso excesivo puede hacer que los selectores sean difíciles de entender y mantener.

Última actualización el 08/10/2026

Selectores con IF ternario

Selectores con IF ternario

Anonymous · 08/10/2026

Selectores con IF ternario

Permiten seleccionar un valor basado en una condición.

- **Formato:** ((selector_condición) operador "valor_condición" ? (selector_si_verdadero) : (selector_si_falso))
- **Ejemplo:** ((edad) >= "18" ? "adulto" : "menor") devuelve "adulto" si la edad es mayor o igual a 18, de lo contrario devuelve "menor".

Operadores disponibles en IF ternario

Los selectores con IF ternarios soportan varios operadores:

- Igualdad: =, ==
- Desigualdad: !=, <>
- Mayor que: >
- Menor que: <
- Mayor o igual que: >=
- Menor o igual que: <=
- Contiene (para listas y cadenas): contains
- Longitud (para listas y cadenas): length

Ejemplos

Condición Simple:

- Dado el JSON: {"edad": 20}
- Selector: ((edad) < "18" ? "Menor" : "Adulto")
- Resultado: "Adulto"

Condición con Contiene en Lista:

- Dado el JSON: {"frutas": ["manzana", "banana", "naranja"]}
- Selector: ((frutas) contains "banana" ? "Encontrada" : "No Encontrada")
- Resultado: "Encontrada"

Selector con Condiciones IF:

- Dado el JSON: {"valor": 100}
- Selector: ((valor) > "50" ? "Alto" : "Bajo")
- Resultado: "Alto"

Nota

Aunque las condiciones IF ofrecen gran flexibilidad, su uso excesivo puede hacer que los selectores sean difíciles de entender y mantener.

Última actualización el 08/10/2026