

# 5

## ERROS QUE QUEBRAM O SEU JSON

Guia prático para nunca mais errar no formato JSON



Programação de Aplicativos | Nível Iniciante | Curso Técnico SENAI

## Antes de começar: o que é JSON?

**JSON** (JavaScript Object Notation) é o formato de dados mais usado em APIs e aplicações web. É basicamente um jeito padronizado de escrever dados em texto para que qualquer linguagem consiga ler.

Ele parece muito com um objeto JavaScript, mas tem regras **bem mais rígidas**. É exatamente essas regras que causam os erros que vamos ver!

### ERRADO

```
// JavaScript (mais permissivo):  
let obj = {  
  nome: 'Ana',  
  idade: 20,  
}
```

JS aceita aspas simples, chave sem aspas e vírgula final

### CERTO

```
// JSON (regras estritas):  
{  
  "nome": "Ana",  
  "idade": 20  
}
```

JSON exige aspas duplas e sem vírgula no último item

**1**

## , A Vírgula Traidora (Trailing Comma)

ERRO #1

Vírgula no **último item** de um objeto ou array. É o erro mais comum de quem vem do JavaScript, porque o JS aceita. O JSON, não!

### ERRADO

```
{  
  "nome": "Joao",  
  "idade": 20,  
}  
  
[  
  "maca",  
  "banana",  
]
```

Vírgula depois de 20 e depois de 'banana' — ambas inválidas

### CERTO

```
{  
  "nome": "Joao",  
  "idade": 20  
}  
  
[  
  "maca",  
  "banana"  
]
```

O último item nunca termina com vírgula

**2**

## ' Aspas Simples vs. Duplas

ERRO #2

JSON só aceita **aspas duplas** "" em chaves e valores de texto. Aspas simples ' ' são válidas em JavaScript, mas no JSON causam erro imediato.

**ERRADO**

```
{  
  'nome': 'Maria',  
  'cidade': 'Sao Paulo'  
}
```

Aspas simples em tudo — JSON não aceita!

**CERTO**

```
{  
  "nome": "Maria",  
  "cidade": "Sao Paulo"  
}
```

Aspas duplas em chaves e valores de texto

3



## Chave sem Aspas

ERRO #3

Em JavaScript você pode escrever `{ nome: 'Ana' }` sem aspas na chave. No JSON isso é inválido — **toda chave precisa estar entre aspas duplas**, sem exceção.

## ERRADO

```
{
  nome: "Pedro",
  idade: 25,
  ativo: true
}
```

nome e idade sem aspas — invalido em JSON

## CERTO

```
{
  "nome": "Pedro",
  "idade": 25,
  "ativo": true
}
```

Todas as chaves entre aspas duplas

4



## Estrutura Multinível Quebrada

ERRO #4

JSON pode ter objetos dentro de objetos e arrays dentro de objetos. Mas **cada { precisa de }** e **cada [ precisa de ]**. Esquecer de fechar é um dos erros mais difíceis de achar!

## ERRADO

```
{
  "contato": {
    "email": "leo@email.com"
  },
  "tags": ["js", "css"
}
```

O array 'tags' não foi fechado com ]

## CERTO

```
{
  "contato": {
    "email": "leo@email.com"
  },
  "tags": ["js", "css"]
}
```

{ fecha }, [ fecha ] — cada abertura tem fechamento

5



## Erro de Tipo de Dado

ERRO #5

JSON só aceita **6 tipos de dados**: string (texto), number (numero), boolean (true/false), null, array e object. Nada de funções, datas, undefined ou código executável!

ERRADO	CERTO
<pre>{   "idade": 20,   "saudacao": alert("Oi"),   "criado": pegarData(),   "valor": undefined }</pre>	<pre>{   "idade": 20,   "saudacao": "Oi",   "criado": "2025-01-15",   "valor": null }</pre>
Funcao, Date() e undefined sao invalidos em JSON	Texto, numero, data como string e null sao validos

## Tipos validos no JSON

Tipo	Exemplo	Observação
String	"texto entre aspas duplas"	Sempre aspas duplas
Number	42 / 3.14 / -10	Sem aspas, ponto decimal
Boolean	true / false	Minúsculo, sem aspas
Null	null	Minúsculo, sem aspas
Array	["a", "b", 3]	Colchetes, itens separados por vírgula
Object	{"chave": "valor"}	Chaves com aspas duplas

## Resumão: os 5 erros de uma vez so

#	Erro	ERRADO	CERTO
	Vírgula Traidora	<code>"idade": 20, }</code>	<code>"idade": 20 }</code>
2	Aspas Simples	<code>{ 'nome': 'Ana' }</code>	<code>{ "nome": "Ana" }</code>
	Chave sem Aspas	<code>{ nome: 'Pedro' }</code>	<code>{ "nome": "Pedro" }</code>
4	Estrutura Quebrada	<code>["a", "b" }</code>	<code>["a", "b"] }</code>
	Tipo Inválido	<code>"f": alert("oi")</code>	<code>"f": "oi"</code>

### Como validar seu JSON antes de usar:

1. Cole o JSON em [jsonlint.com](https://jsonlint.com) ou [jsonformatter.curiousconcept.com](https://jsonformatter.curiousconcept.com)
2. No VS Code, salve o arquivo como `.json` — o editor já destaca os erros
3. No console do navegador (F12), tente `JSON.parse('seu json aqui')` — se quebrar, ele mostra onde está o erro

**Macete para memorizar:** JSON é mais chato que JavaScript. Se você pode fazer no JS, não significa que pode no JSON. Quando der erro, confira na ordem: vírgula final, aspas, chaves fechadas.