

JSON

JavaScript Object Notation

Format structuré de données Syntaxe et Usages

C@fé technique B@O Renatis – 19 mai 2026

philippe.houdry@inist.fr

Plan de la présentation

1. Caractéristiques générales de JSON
2. Syntaxe JSON, caractéristiques et exemples
3. JSON, un vrai exemple commenté
4. Validateurs JSON
5. JSON Lines



Caractéristiques générales de JSON

- Créé par [Douglas Crockford](#) en 1999
- Spécification officielle : [rfc8259](#) et sur le [web](#) (en français)
- Format texte, structuré et arborescent, schéma JSON possible
- Encodage par défaut en [UTF-8](#), extension de fichier « .json »
- Plus léger que [XML](#)
- Lisible par un humain
- ... mais facilite les échanges de données entre applications
- Indépendant des langages de programmation
- ... mais est cependant un sous-ensemble de [Javascript](#)
- Stable dans le temps depuis sa création, largement adopté depuis

Six types de données de base : 4 scalaires et 2 composés

Scalaires (ou primitifs)

- chaîne de caractères (string) : texte (éventuellement vide)
- nombre (number) : pas de distinction entier / flottant
- booléen (boolean) : true / false
- null : absence de valeur

Composés (ou complexes)

- objet (object) : paire clé - valeur (scalaire et/ou composée)
- tableau (array) : liste ordonnée de valeurs (scalaires et/ou composées)

A noter : pas de commentaires possibles en JSON.

Six types de données de base : 4 scalaires et 2 composés

Scalaires (ou primitifs)

- string : `"mot1 \n mot2 \n"` ou vide `""`
- nombre : un entier comme `42` ou un décimal comme `3.14`
- booléen : `true` `false`
- null : `null`

\n pour fin
de ligne

Décimal avec
un point

Valeurs en
minuscules

Composés (ou complexes)

- objet : `{ "key1" : "valeur texte", "key2" : 123 }`
- tableau : `[ 1,2,3 ]` `[ "a","b","c","",10,20,3.14,true,null ]`

Chaîne vide
avec `""`

Utilisation plus complexes de ces types de données

Tableau de tableau (ou matrice)

```
[ ["nom", "prénom"], ["Crockford", "Douglas"], ["Houdry",  
"Philippe" ] ]
```

Tableau d'objets (ou collections)

```
[  
  { "nom" : "Crockford", "prénom" : "Douglas" },  
  { "nom" : "Houdry", "prénom" : "Philippe" }  
]
```

Pas de virgule
à la fin

Syntaxe JSON : un vrai exemple - 1/3

```
{
  "id": "https://openalex.org/W2993763064",
  "doi": "https://doi.org/10.3917/i2d.192.0109",
  "title": "La question des revues dans la science ouverte : une approc",
  "display_name": "La question des revues dans la science ouverte : une",
  "publication_year": 2019,
  "publication_date": "2019-11-26",
  "ids": {
    "openalex": "https://openalex.org/W2993763064",
    "doi": "https://doi.org/10.3917/i2d.192.0109",
    "mag": "2993763064"
  },
  "language": "fr",
  "primary_location": {
    "id": "doi:10.3917/i2d.192.0109",
    "is_oa": false,
    "landing_page_url": "https://doi.org/10.3917/i2d.192.0109",
    "pdf_url": null,
    "source": {
      "id": "https://openalex.org/S4210225659",
      "display_name": "I2D - Information données & documents",
      "issn_l": "2428-2111",
      "issn": [
        "2428-2111",
        "2431-3467"
      ],
      "is_oa": false,
      "is_in_doaj": false,
      "is_core": false,
      "host_organization": "https://openalex.org/P4310311120",
      "host_organization_name": "Association des professionnels de",
      "host_organization_lineage": [
        "https://openalex.org/P4310311120"
      ]
    }
  }
}
```

Début du
JSON : {

Exploration d'une notice
OpenAlex en JSON.

Objets
imbriqués

Valeurs
booléennes

Syntaxe JSON : un vrai exemple - 2/3

```
"institutions": [],
"countries_distinct_count": 1,
"institutions_distinct_count": 2,
"corresponding_author_ids": [
  "https://openalex.org/A5032597808"
],
"corresponding_institution_ids": [
  "https://openalex.org/I2279609970"
],
"apc_list": null,
"apc_paid": null,
"fwci": 0.4841,
"has_fulltext": true,
"cited_by_count": 4,
"citation_normalized_percentile": {
  "value": 0.74106454,
  "is_in_top_1_percent": false,
  "is_in_top_10_percent": false
},
"cited_by_percentile_year": {
  "min": 90,
  "max": 95
},
"biblio": {
  "volume": "n° 2",
  "issue": "2",
  "first_page": "109",
  "last_page": "127"
},
"is_retracted": false,
"is_paratext": false,
"is_xpac": false,
"primary_topic": {
  "id": "https://openalex.org/T10102",
  "display_name": "scientometrics and bibliometrics research",
  "score": 0.9952999949455261,
```

Tableau vide
[]

Booléens et
numériques

Nombres en
chaînes car.

Syntaxe JSON : un vrai exemple - 3/3

```

    "host_organization": "https://openalex.org/I1294671590",
    "host_organization_name": "Centre National de la Recherche",
    "host_organization_lineage": [
      "https://openalex.org/I1294671590"
    ],
    "host_organization_lineage_names": [],
    "type": "repository"
  },
  "license": "other-oa",
  "license_id": "https://openalex.org/licenses/other-oa",
  "version": "submittedVersion",
  "is_accepted": false,
  "is_published": false,
  "raw_source_name": "I2D - Information, données & documents, 20",
  "raw_type": "Journal articles"
},
"sustainable_development_goals": [],
"awards": [],
"funders": [],
"has_content": {
  "pdf": true,
  "grobid_xml": true
},
"content_urls": {
  "pdf": "https://content.openalex.org/works/W2993763064.pdf",
  "grobid_xml": "https://content.openalex.org/works/W2993763064."
},
"referenced_works_count": 27,
"referenced_works": [
  "https://openalex.org/W1488718213",
  "https://openalex.org/W1655306493",
  "https://openalex.org/W2014937475",
  "https://openalex.org/W2302501749",
  "https://openalex.org/W2572440112"

```

Objets et
tableaux

Voir arbo-
rescence

Fin du JSON
avec }

```

  },
  "updated_date": "2026-05-07T13:39:58.223016",
  "created_date": "2025-10-10T00:00:00"
}

```

Il en existe de nombreux en ligne.
Par exemple jsonlint.com

To format and validate your JSON, just copy + paste it below:

```
1 [
2   { "nom" : "Crockford", "prénom" : "Douglas" },
3   { "nom" : "Houdry", "prénom" : "Philippe" },
4 ]
```

, erreur après
Philippe

Si on préfère éviter des solutions en ligne, de nombreux éditeurs comme [Notepad++](#) (avec l'extension JSON Viewer) ou [Vscode](#) peuvent faire ces mêmes contrôles en local.

✓ Validate

Clear

Compress

Sort Keys

Copy

Try Fix

✗ Invalid JSON

JSON.parse: unexpected character at line 4 column 1 of the JSON data (Line 1)

💡 Your JSON contains trailing commas which are not allowed in standard JSON. Use "Try Fix" to remove them.

Message
d'erreur

En JSON Lines, un document JSON valide tient sur une seule ligne.

L'intérêt de ce format est qu'il convient très bien aux traitements qui demandent un document à la fois car lire un fichier ligne par ligne est très rapide et efficace pour de nombreux outils ou langages (OpenRefine ou Python par exemple). L'extension d'un tel fichier est « .jsonl ».

```
[\n  { "nom" : "Crockford", "prénom" : "Douglas" },\n  { "nom" : "Houdry", "prénom" : "Philippe" }\n]\n
```



```
[{"nom" : "Crockford", "prénom" : "Douglas"}, {"nom" : "Houdry", "prénom" : "Philippe"}]\n
```

\n pour fin
de ligne