

# JSON

JSON (JavaScript Object Notation) est principalement utilisé pour échanger les données d'une application web entre un navigateur et un serveur. Un fichier JSON est un document texte enregistré avec l'extension .json, qui stocke des données structurées en paires clé/valeur. C'est un format relativement lisible, qui représente les données structurées sous forme de paires clé/valeur. {« utilisateur":"Carnus"}. La valeur peut être une chaîne de caractères, un nombre, une valeur booléenne, la valeur null, un objet ou un tableau.

## Structure d'un fichier JSON

Un fichier JSON comporte un nom de fichier suivi de l'extension .json. Le contenu du fichier se présente sous forme de paires clé/valeur. Les règles de syntaxe sont les suivantes :

- Une paire est écrite entre accolades.
- Chaque clé et chaque valeur de type chaîne de caractères sont entre guillemets doubles.
- La valeur est séparée de sa clé par un deux-points.
- Les clés sont des chaînes de caractères. Les valeurs peuvent être « null », des chaînes de caractères, des booléens, des nombres, des objets ou des tableaux.
- Les éléments de données sont séparés par des virgules.
- Les espaces, les retours à la ligne et les tabulations sont indifférents.
- Les accolades {} désignent les objets.
- Les crochets [] désignent des tableaux.

Un objet JSON est un ensemble de paires clé/valeur. Les paires sont séparées par une virgule.

### Exemple :

```
{
  "utilisateur": {
    "prenom": "Charles",
    "nom": "Carnus",
    "age": 19,
    "ville": null
  }
}
```

Un tableau JSON est une suite de valeurs, sous forme de chaînes de caractères, associées à une clé unique. Les valeurs sont écrites entre crochets et séparées par une virgule.

### Exemple :

```
{
  "Matières": [
    "Informatique",
    "Physique",
    "Anglais"
  ]
}
```

Les tableaux JSON permettent de créer des listes. Dans l'exemple, la clé *Matières* stocke une liste de plusieurs valeurs.

Dans un fichier JSON, il est possible d'intégrer des tableaux dans des objets.

**Exemple :**

```
{
  "utilisateur": {
    "prenom": "Charles",
    "nom": "Carnus",
    "age": 19,
    "ville": null,
    "reseaux": [
      "Instagram",
      "Linkedin",
      "Facebook"
    ]
  }
}
```

L'objet utilisateur, dans cet exemple, permet de lire les données de prénom, de nom, d'âge et de ville, et de consulter une liste de données de réseaux sociaux.

**Chaîne de caractères :** Les valeurs de type chaîne sont des séquences définies de zéro ou plusieurs caractères Unicode, entourées de guillemets.

```
{"nom": "Carnus"}
```

**Nombre :** Un nombre dans JSON doit être un **nombre entier** ou un **nombre à virgule flottante (float)**.

```
{"age": 20}
```

**Booléen :** Les booléens contiennent les valeurs **vrai** ou **faux**.

```
{"sportif": false}
```

**Null :** Null est une valeur vide. C'est pour montrer qu'il n'y a pas d'information.

```
{"bloodType": null}
```

**Exemple 1**

```
{
  "classe": "CIEL-1",
  "annee": 2025,
  "telephone": null,
  "inscrit": true,
  "prof": {"nom": "Carnus", "prenom": "Charles"},
  "etudiants": [{
    "nom": "Nom1", "prenom": "prenom1"},
    {"nom": "nom2", "prenom": "prenom2"},
    {"nom": "nom3", "prenom": "prenom3"},
  ]
}
```

## Exemple 2

```
{
  "nomEquipe": "CIEL-1",
  "ville": "Rodez",
  "creation": 2024,
  "lieu": "Salle-215",
  "actif": true,
  "membres": [
    {
      "nom": "Nom1 Prenom1",
      "age": 19,
      "matieres": ["Anglais", "Informatique", "Physique"]
    },
    {
      "nom": "Nom2 prenom2",
      "age": 18,
      "powers": [
        "Informatique",
        "Physique",
        "CGE"
      ]
    },
    {
      "nom": "Nom3 Prenom3",
      "age": 20,
      "powers": [
        "Informatique",
        "Anglais",
        "Physique",
        "CGE"
      ]
    }
  ]
}
```

## Exercice

- Créer un fichier JSON nommé **fichier1.json** et copier-coller les données de l'exemple 1
- Visualiser ce fichier avec différents navigateurs
- Créer un fichier JSON nommé **fichier2.json** et copier-coller les données de l'exemple 2
- Visualiser ce fichier avec différents navigateurs