

Bachelier en Informatique de Gestion

Web : principes de base Projet de Développement Web

Enseignement supérieur économique de type court

Code FWB : 7534 29 U32 D1, 7534 30 U32 D3

Code ISFCE : 4IWPB, 4IPW3



Table des matières

Généralités

01. Introduction au web

03. Outils

05. Format XML

06. Format JSON

Front-End

12. Structure HTML

13. Formulaire HTML

14. Mise en forme CSS

15. Adaptabilité

17. Javascript

18. Bibliothèque jQuery

19. Composant Vue.js

Back-End

21. Middleware PHP

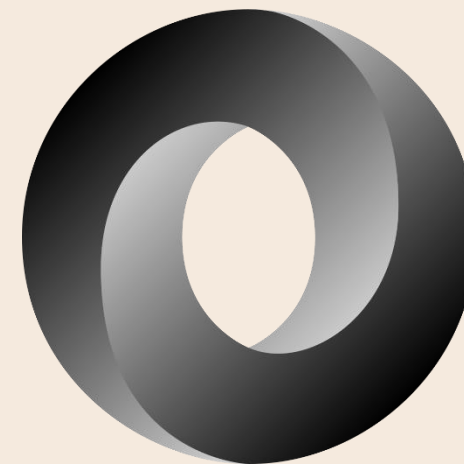
22. Traitement du
formulaire

23. Architecture MVC

24. Données SQL

25. Données NoSQL

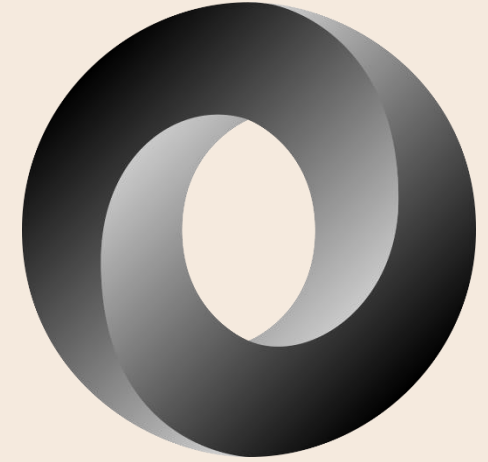
27. Requête asynchrone



06. Données JSON

Structure

JSON vs XML



JSON : Structure

JSON en quelques mots

- JavaScript Object Notation = format d'échange de données
- Facile à lire et à écrire pour les humains
- Facile à analyser et à générer pour les machines
- Basé sur JavaScript.
- Format "texte"
- Indépendant du langage, universel
- Basé sur des conventions familières aux programmeurs de la famille "C" (C, C++, C#, Java, JavaScript, Perl, PHP, ...)
- www.json.org

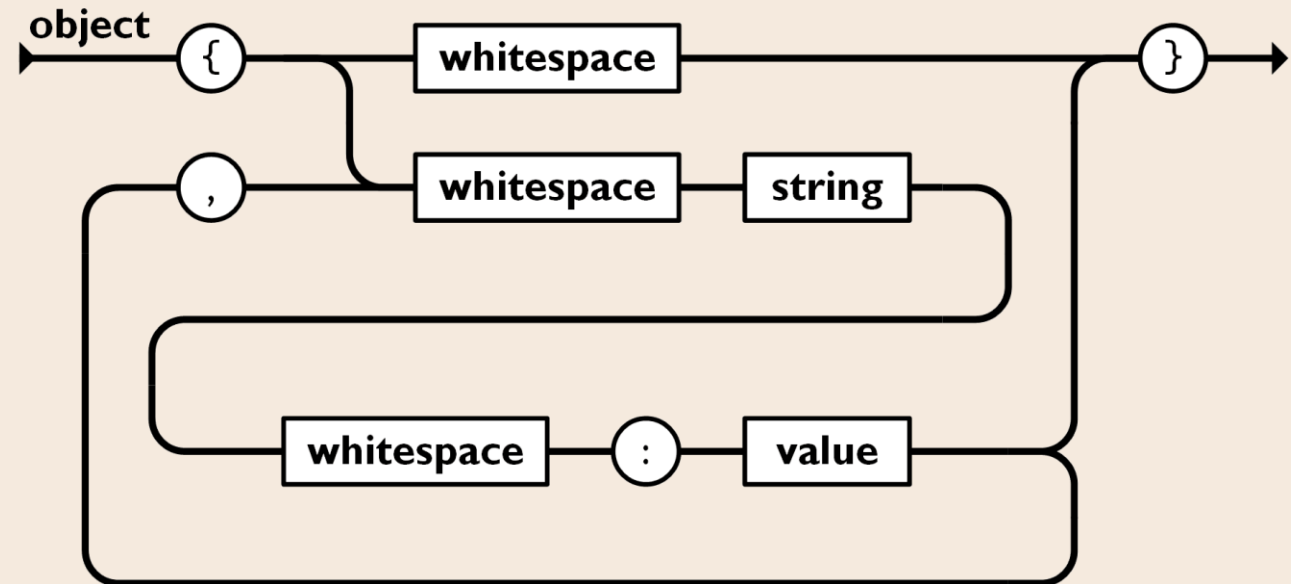


JSON : deux structures

1. Une collection de paires

<nom> / <valeur>

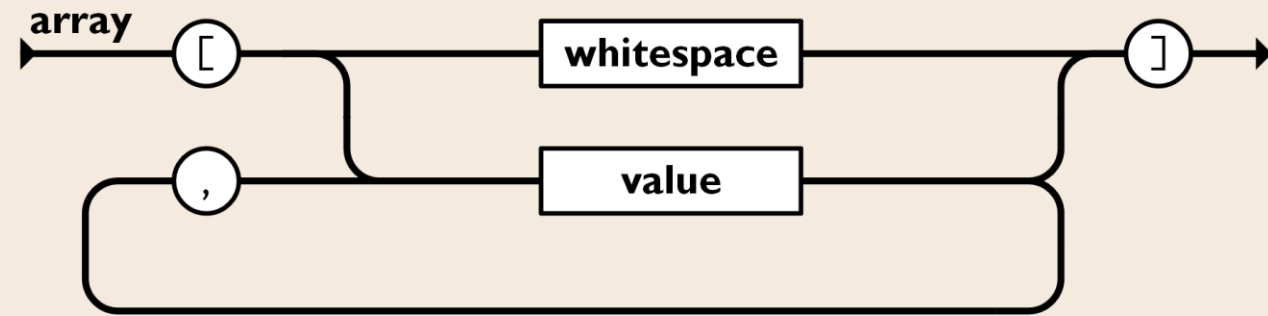
- // tableau associatif, dict.
- Début : {
- Fin : }
- **<nom> : <valeur>**
- Séparateur : ,



JSON : deux structures

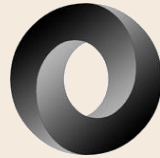
2. Une liste ordonnée de valeurs

- // tableau, liste
- Début : **[**
- Fin : **]**
- Séparateur : **,**



JSON vs. XML

```
{
  Menu : {
    id : "file",
    Value : "File",
    Popup : {
      menuitem: [
        {
          value: "New",
          onclick:"CreateNewDoc()"
        },
        {
          value: "Open",
          onclick: "OpenDoc()"
        },
        {
          value: "Close",
          onclick: "CloseDoc()"
        }
      ]
    }
  }
}
```



```
<menu
  id="file"
  value="File">
  <popup>
    <menuitem
      value="New"
      onclick="CreateNewDoc()"
    />
    <menuitem
      value="Open"
      onclick="OpenDoc()"
    />
    <menuitem
      value="Close"
      onclick="CloseDoc()"
    />
  </popup>
</menu>
```

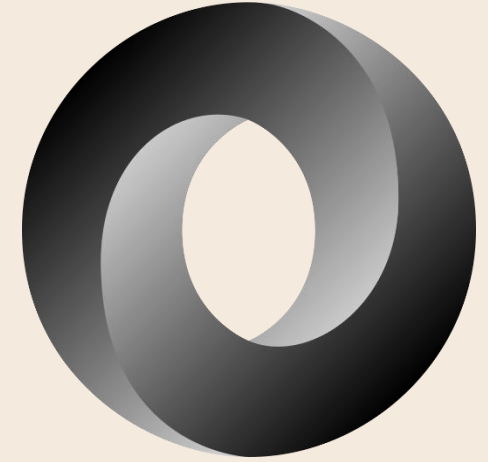


= BUROTIX ()

Autre exemple : exo05_cantine.json

```
{
  "semaine": 37,
  "année": 2025,
  "nomCantine": "Les Délices de Tante Gertrude (approuvé par 3 élèves sur 10)",
  "menus": [
    {
      "jour": "lundi",
      "platPrincipal": {
        "nom": "Mystère en sauce brune",
        "description": "Un plat avec une sauce. Toujours une sauce.",
        "noteMoyenne": 2.5,
        "commentaires": [
          "Goût de carton... mais c'est peut-être DU carton ? – Jean-Michel (5ème)",
          "La sauce a sauvé ma vie. – Marie (Terminale, désespérée)"
        ],
        "accompagnement": ["frites molles", "haricots verts (théoriquement)"]
      }
    },
    ...
  ],
  ...
}
```





JSON : Références

Références

■ Général

- <https://www.json.org/json-fr.html>
- https://www.w3schools.com/js/js_json_intro.asp

■ API

- <https://itds.fr/creer-une-api-rest-en-php/>
- <https://blog.nicolashachet.com/developpement-php/larchitecture-rest-expliquee-en-5-regles/>
- <https://blog.nicolashachet.com/developpement-php/exemples-api-rest-en-php/>

