

TP_5_JSON

Objectif : JSON, JSON Schema, JSON Path

Exercice 1:

- **Proposer un fichier JSON représentant une liste d'étudiants, chaque élément étudiant doit contenir les informations suivantes :**

- identifiant
- nom
- prenom
- dateNaissance
- lieuNaissance
- adresse
 - numRue
 - nomRue
 - codePostal
 - ville
- mail
- coursSuivis sous forme de tableau

NB : Valideur JSON online : <http://jsonlint.com/>

- **Appliquer les règles suivantes :**
 - ✓ La liste doit contenir au moins 3 étudiants
 - ✓ propriétés identifiant , nom , prenom et mail sont obligatoires
 - ✓ dateNaissance doit respecter le format JJ/MM/AA
 - ✓ codePostal doit contenir 5 chiffres
 - ✓ Si coursSuivis est présent, il doit contenir au moins un item
- **Vérifier la bonne conformité de votre fichier en ligne**

Exercice 2:

- **Écrire le schéma permettant de valider le document JSON suivant :**

```
{
  "codePerm": "DOEJ01020300",
  "nom": "Doe",
  "prenom": "John",
  "cours": [
    {
      "sigle": "INF0101",
      "credits": 3
    },
    {
      "sigle": "INF0102",
      "credits": 3,
      "reprise": true
    }
  ]
}
```

- **Règles à appliquer:**
 - ✓ Cours contient un tableau d'objets représentants des cours

- ✓ chaque cours est un objet contenant toujours les propriétés sigle et credits
- ✓ la propriété reprise est facultative

NB : Valideur JSON Schema online : <https://json-schema-validator.herokuapp.com/>

Exercice 3:

- **Écrire le schéma permettant de valider le document JSON suivant:**

```
{
  "facID": "123456",
  "total": 163.0,
  "A213": {
    "nom": "Canard en plastique",
    "qt": 3,
    "prix": 1.0
  },
  "P214": {
    "nom": "Flamant rose en platre",
    "qt": 1,
    "prix": 10.0
  },
  "X215": {
    "nom": "Ours en bois",
    "qt": 1,
    "prix": 150
  }
}
```

- **Règles à appliquer:**

- ✓ une facture contient toujours un facID et un total
- ✓ les autres propriétés sont des articles ajoutés à la facture
- ✓ un article est un objet maintenu sous une clé de la forme AXXX , où A est une lettre et XXX un nombre à trois chiffres
- ✓ chaque article contient un nom, un prix et une quantité qt

Exercice 4:

- **Considérons le document JSON suivant:**

```
{ "store": {
  "book": [
    { "category": "reference",
      "author": "Nigel Rees",
      "title": "Sayings of the Century",
      "price": 8.95
    },
    { "category": "fiction",
      "author": "Evelyn Waugh",
      "title": "Sword of Honour",
      "price": 12.99
    },
    { "category": "fiction",
      "author": "Herman Melville",
```

```
    "title": "Moby Dick",
    "isbn": "0-553-21311-3",
    "price": 8.99
  },
  { "category": "fiction",
    "author": "J. R. R. Tolkien",
    "title": "The Lord of the Rings",
    "isbn": "0-395-19395-8",
    "price": 22.99
  }
],
  "bicycle": {
    "color": "red",
    "price": 19.95
  }
}
```

- **Décrire les chemins JSON Path permettant de sélectionner les éléments suivants :**
 - Tous les membres de la structure JSON
 - Les auteurs de tous les livres
 - Tous les livres
 - Les prix de tous les livres
 - Le 1er livre
 - Les deux derniers livres
 - Tous les livres avec le numéro isbn
 - Tous les livres plus chers que 10

NB: *JSON Path evaluator online JSON Schema online :*