UNIVERSITE VIRTUELLE DU SENEGAL

foo nekk foofu la

Conversion entre JSON et JavaScript

Le format de données JSON

El hadji Mamadou NGUER Enseignant chercheur en Informatique à l'UVS



2020

Chapitre 3 : Conversion entre JSON et JavaScript

Objectifs spécifiques : A la suite de ce chapitre, l'étudiant doit être capable de:

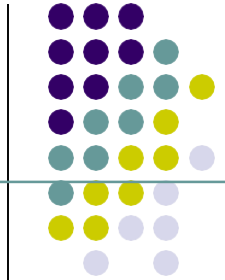
1. Convertir une données JSON en objet JavaScript
2. Convertir un objet JavaScript en une donnée JSON

Chapitre 3 : Conversion entre JSON et JavaScript

Plan du chapitre :

1. Introduction
2. Conversion JSON vers JavaScript
3. Conversion JavaScript vers JSON

Introduction



- Une utilisation courante de JSON consiste à échanger des données avec un serveur Web.
- Lors de la réception de données d'un serveur Web, les données sont toujours de type chaîne.
- Et pour les analyser, il faudra les convertir en objet JavaScript.
- De même, lors de l'envoi de données vers un serveur Web, les données sont toujours de type chaîne.
- Et s'il s'agit d'objet JavaScript, ils doivent être convertis en JSON.
- Nous allons dans la suite, voir comment effectuer ces conversions.

Conversion JSON vers JavaScript



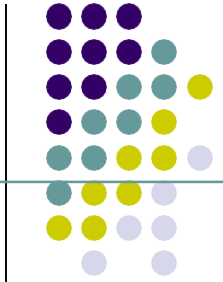
- Une utilisation courante de JSON consiste à échanger des données avec un serveur Web.
- Lors de la réception de données d'un serveur Web, les données sont toujours de type chaîne.
- Et pour les analyser, il faudra les convertir en objet JavaScript en utilisant la fonction JSON.parse().

Cas de données provenant d'une variable :

```
var myJSON = '{"nom":"Samba", "age":31, "ville":"Dakar"}';
var myObj = JSON.parse(myJSON);
document.getElementById("demo").innerHTML = myObj.nom;
```

Attention : Il faut s'assurer que le texte soit écrit au format JSON, sinon une erreur de syntaxe sera obtenue.

Conversion JSON vers JavaScript



Cas de données JSON provenant d'un serveur :

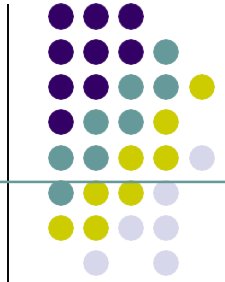
- On peut obtenir des données JSON à partir d'un serveur web à l'aide d'une requête AJAX
- Tant que la réponse du serveur est écrite au format JSON, il est possible d'analyser la chaîne dans un objet JavaScript.

```
var xmlhttp = new XMLHttpRequest();
xmlhttp.onreadystatechange = function() {
  if (this.readyState == 4 && this.status == 200) {
    var myObj = JSON.parse(this.responseText);
    document.getElementById("demo").innerHTML = myObj.nom;
  }
};
xmlhttp.open("GET", "json_pers.txt", true);
xmlhttp.send();
```

- L'objet myObj est un objet JavaScript car la chaîne obtenue est un objet JSON.
- Le contenu du fichier json_pers.txt est présenté ci-contre.

```
{
  "nom":"Samba",
  "age":31,
  "œuvres":[
    { "livre":"roman", "nom":"Bamba" },
    { "livre":"poeme", "nom":"Xarnu bi" },
    { "livre":"theatre", "nom":"Xànju" }
  ]
}
```

Conversion JSON vers JavaScript



Cas de données provenant d'un serveur :

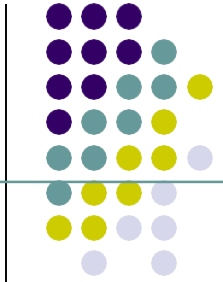
- Lors de l'utilisation de JSON.parse() sur un fichier JSON contenant un tableau, la méthode renvoie un tableau JavaScript au lieu d'un objet JavaScript.

```
var xmlhttp = new XMLHttpRequest();
xmlhttp.onreadystatechange = function() {
  if (this.readyState == 4 && this.status == 200) {
    var myArr = JSON.parse(this.responseText);
    document.getElementById("demo").innerHTML = myArr[0];
  }
};
xmlhttp.open("GET", "json_hop_array.txt", true);
xmlhttp.send();
```

- Contenu du fichier json_hop_array.txt

```
[ "Fann", "Hoggy", "Principal", "Matlaboul Fawzeyni" ]
```

Conversion JSON vers JavaScript



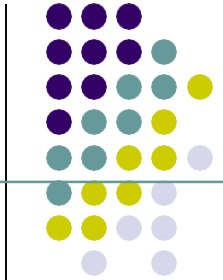
Quelques exceptions :

- **Parser une date :**
  - Les objets date ne sont pas autorisés dans JSON.
  - Pour inclure une date, on l'écrit sous forme de chaîne.
  - Il pourra le reconvertir plus tard en un objet JavaScript date:

```
var text = '{ "nom":"Samba", "dateNais":"1986-12-14", "ville":"Dakar"}';
var obj = JSON.parse(text);
obj.dateNais = new Date(obj.dateNais);

document.getElementById("demo").innerHTML = obj.nom + ", " + obj.dateNais;
```

Conversion JSON vers JavaScript



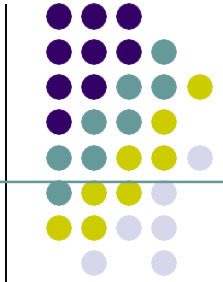
Quelques exceptions :

- **Parser une date : (Utilisation de la fonction reviver)**
  - On peut aussi utiliser le second paramètre de JSON.parse() appelé fonction reviver.
  - Le paramètre reviver est une fonction qui vérifie chaque propriété, avant de renvoyer la valeur.

```
var text = '{ "nom":"Samba", "dateNais":"1986-12-14", "ville":"Dakar"}';
var obj = JSON.parse(text, function (key, value) {
  if (key == "dateNais") {
    return new Date(value);
  } else {
    return value;
  }
});

document.getElementById("demo").innerHTML = obj.nom + ", " + obj.dateNais;
```

Conversion JSON vers JavaScript



Quelques exceptions :

- **Parser une fonction :**
  - Les fonctions ne sont pas autorisées dans JSON.
  - Si vous devez inclure une fonction, écrivez-la sous forme de chaîne.
  - Vous pouvez le reconvertir après en fonction JavaScript.

```
var text = '{"nom":"Samba", "age":"function () {return 30;}", "ville":"Dakar"}';
var obj = JSON.parse(text);
obj.age = eval("(" + obj.age + ")");

document.getElementById("demo").innerHTML = obj.nom + ", " + obj.age();
```

Il faut éviter d'utiliser des fonctions dans JSON, les fonctions perdront leur portée et vous devrez les utiliser eval() pour les reconvertir en fonctions.

Conversion JSON vers JavaScript

La prise en charge du parseur par les navigateurs :

- La fonction JSON.parse() est incluse dans tous les principaux navigateurs et dans la dernière norme ECMAScript (JavaScript).
- Le tableau ci-dessous spécifie les versions des navigateurs à partir des quelles la fonction JSON.parse() est prise entièrement en charge :

Yes	8.0	3.5	4.0	10.0

- Pour les navigateurs les plus anciens, une bibliothèque JavaScript est disponible sur https://github.com/douglascrockford/JSON-js .

Conversion JavaScript vers JSON



- Une utilisation courante de JSON consiste à échanger des données avec un serveur Web.
- Lors de l'envoi de données vers un serveur Web, les données sont toujours de type chaîne.
- S'il s'agit d'objet JavaScript, il doit être converti en JSON avec la fonction JSON.stringify().

- **Conversion d'un objet JavaScript en JSON :**

Exemple :

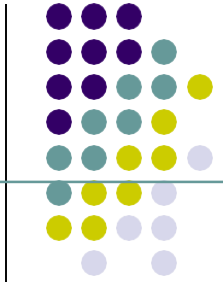
```
var obj = { nom: "Samba", age: 30, ville: "Dakar" };
var myJSON = JSON.stringify(obj);
document.getElementById("demo").innerHTML = myJSON;
```

- **Conversion d'un tableau JavaScript en JSON :**

Exemple :

```
var arr = [ "Samba", "Pathé", "Saly", "Jatu" ];
var myJSON = JSON.stringify(arr);
document.getElementById("demo").innerHTML = myJSON;
```

Conversion JavaScript vers JSON



Quelques exceptions

- **Conversion d'un objet date en JSON :**

  En JSON, les objets date ne sont pas autorisés mais il est possible de les convertir en objet JSON en utilisant la fonction JSON.stringify().

Exemple :

```
var obj = { nom: "Samba", today: new Date(), ville : "Dakar" };
var myJSON = JSON.stringify(obj);

document.getElementById("demo").innerHTML = myJSON;
```

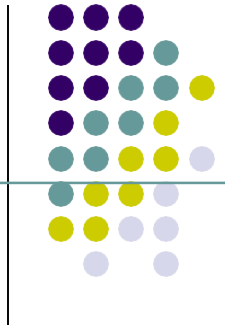
- **Conversion d'une fonction JavaScript en JSON :**
  - En JSON, les fonctions ne sont pas autorisées en tant que valeurs d'objet.
  - La fonction JSON.stringify() supprime toutes fonctions dans un objet JavaScript, à la fois la clé et la valeur :

Exemple :

```
var obj = { nom: "Samba", age: function () {return 30;}, ville: "Dakar"};
var myJSON = JSON.stringify(obj);

document.getElementById("demo").innerHTML = myJSON;
```

Conversion JavaScript vers JSON



Quelques exceptions

- **Conversion d'une fonction JavaScript en JSON :**

  - Cependant il est possible de l'éviter en convertissant la fonction en une chaîne avant l'appel de JSON.stringify() :

Exemple :

```
var obj = { nom: "Samba", age: function () {return 30;}, ville: "Dakar" };
obj.age = obj.age.toString();
var myJSON = JSON.stringify(obj);

document.getElementById("demo").innerHTML = myJSON;
```

Les fonctions envoyées à l'aide de JSON perdent leur portée. Il faudra utiliser eval () pour les reconvertir en fonctions.

Conversion JavaScript vers JSON

La prise en charge de JSON.stringify() par les navigateurs :

- La fonction JSON.stringify() est incluse dans JavaScript (ECMAScript) et est prise en charge dans tous les principaux navigateurs.
- Le tableau ci-dessous spécifie les versions des navigateurs à partir des quelles la fonction JSON.stringify() est prise entièrement en charge :

				
Yes	8.0	3.5	4.0	10.0