

HTML CSS Javascript

Javascript – Webservices

Objectif

- Découvrir le format JSON
- Manipuler des callbacks
- Solliciter des WebServices

JSON

JSON est un acronyme pour « JavaScript Object Notation ».

Il s'agit d'un format d'échange de données respectant l'écriture JavaScript.

Aujourd'hui, il est très utilisé pour envoyer des informations d'une application à une autre.

JSON

Il existe deux éléments principaux en JSON : les « objets » et les « tableaux ».

Les objets sont délimités par des accolades { }

Les tableaux sont délimités par des crochets []

JSON

Il existe deux éléments principaux en JSON : les « objets » et les « tableaux ».

Les objets sont délimités par des accolades { }

Les tableaux sont délimités par des crochets []

Observons les schémas du site : <https://www.json.org/json-en.html>

JSON

Il existe deux éléments principaux en JSON : les « objets » et les « tableaux ».

Les objets sont délimités par des accolades { }

Les tableaux sont délimités par des crochets []

Observons les schémas du site : <https://www.json.org/json-en.html>

Utilisons la fonction *fetch* de javascript pour lire un fichier local !

Callbacks

En Javascript, il est possible de stocker des fonctions dans des variables.

Callbacks

En Javascript, il est possible de stocker des fonctions dans des variables.

➔ Regardons l'intérêt avec la calculatrice avec choix d'opérateur (exo 1 revisité)

Callbacks

En Javascript, il est possible de stocker des fonctions dans des variables.

➔ Regardons l'intérêt avec la calculatrice avec choix d'opérateur (exo 1 revisité)

Les callbacks sont un cas particulier de cette utilisation.

Un callback est une fonction passée en paramètre d'une autre fonction, qui sera invoquée par cette dernière à un moment donné du traitement.

source : MDN

Callbacks

En Javascript, il est possible de stocker des fonctions dans des variables.

➔ Regardons l'intérêt avec la calculatrice avec choix d'opérateur (exo 1 revisité)

Les callbacks sont un cas particulier de cette utilisation.

Un callback est une fonction passée en paramètre d'une autre fonction, qui sera invoquée par cette dernière à un moment donné du traitement.

source : MDN

L'objectif est de pouvoir définir des comportements (fonctions) qui s'exécuteront à la fin d'un traitement long, ou à des moments clés de celui-ci.

Callbacks

En Javascript, il est possible de stocker des fonctions dans des variables.

➔ Regardons l'intérêt avec la calculatrice avec choix d'opérateur (exo 1 revisité)

Les callbacks sont un cas particulier de cette utilisation.

Un callback est une fonction passée en paramètre d'une autre fonction, qui sera invoquée par cette dernière à un moment donné du traitement.

source : MDN

L'objectif est de pouvoir définir des comportements (fonctions) qui s'exécuteront à la fin d'un traitement long, ou à des moments clés de celui-ci.

Démos : afficher un message de validation à la fin d'un traitement long; mettre à jour une barre de chargement.

Solliciter des WebServices

Un WebService est une application permettant d'échanger des données avec d'autres applications.
(*Oracle*)

Solliciter des WebServices

Un WebService est une application permettant d'échanger des données avec d'autres applications.
(*Oracle*)

Aujourd'hui, la très grande majorité des WebServices partagent ces données dans un format de fichier appelé JSON.

Solliciter des WebServices

Un WebService est une application permettant d'échanger des données avec d'autres applications.
(*Oracle*)

Aujourd'hui, la très grande majorité des WebServices partagent ces données dans un format de fichier appelé JSON.

Il nous est possible de solliciter ces WebServices pour afficher les données d'autres sites dans notre propre application.

Solliciter des WebServices

Un WebService est une application permettant d'échanger des données avec d'autres applications.
(Oracle)

Aujourd'hui, la très grande majorité des WebServices partagent ces données dans un format de fichier appelé JSON.

Il nous est possible de solliciter ces WebServices pour afficher les données d'autres sites dans notre propre application.

(On parle aussi souvent d'API pour Application Programming Interface) – dans ce contexte, une API ou un WebService désignent la même chose)

Solliciter des WebServices

En Javascript, c'est l'outil XMLHttpRequest qui nous permettra de solliciter ces WebServices.

```
let xhr = new XMLHttpRequest();

xhr.onreadystatechange = function() {
    if (this.readyState == 4 && this.status == 200) {
        console.log(xhr.responseText);
    }
};

xhr.open("GET", "http://uneurl.com/", true);
xhr.send();
```


Solliciter des WebServices

La réponse est au format texte (string). Afin de la manipuler comme un objet Javascript, il sera nécessaire de préciser au programme que le texte respecte le format JSON :

```
let xhr = new XMLHttpRequest();

xhr.onreadystatechange = function() {
    if (this.readyState == 4 && this.status == 200) {
        console.log(xhr.responseText);
        let contenuJSON = JSON.parse(xhr.responseText);
        console.log(contenuJSON);
    }
};

xhr.open("GET", "http://uneurl.com/", true);
xhr.send();
```

Solliciter des WebServices

La réponse est au format texte (string). Afin de la manipuler comme un objet Javascript, il sera nécessaire de préciser au programme que le texte respecte le format JSON :

```
let xhr = new XMLHttpRequest();

xhr.onreadystatechange = function() {
    if (this.readyState == 4 && this.status == 200) {
        console.log(xhr.responseText);
        let contenuJSON = JSON.parse(xhr.responseText);
        console.log(contenuJSON);
    }
};

xhr.open("GET", "http://uneurl.com/", true);
xhr.send();
```

Testons avec <https://api.gouv.fr> !

HTML CSS Javascript

Javascript – Webservices