Lire des fichiers avec Node.js

1. ## Objectifs

   - Être capable d'utiliser node.js pour lire un fichier

2. ## Présentation

   L'une des choses les plus courantes que vous voudrez faire avec à peu près n'importe quel langage de programmation est d'ouvrir et de lire un fichier. Avec la plupart des langages, c'est assez simple, mais pour les vétérans de JavaScript, cela peut sembler un peu étrange. Pendant tant d'années, JavaScript n'était disponible que dans le navigateur, de sorte que les développeurs frontaux ne connaissent peut-être que l' API FileReader ou similaire.



   - Il existe **deux fonctions** pour lire les fichiers dans le nœud js.
     - La première fonction `readFile ()` est une fonction lit les données du fichier de manière **asynchrone**.
     - La deuxième fonction `readFileSync ()` est une fonction lit les données du fichier de manière **synchrone**.

3. ## contenu du fichier fichier_text.txt

   - Mon site apcpedagogie.com est un site de formation
   - Les cours suivants appartiennent à la catégorie Cours de programmation:
     - php

- java
- nodejs
- javascript
- ruby
- python

4. Fonctions de lecture de fichier Node Js

1. Fonction `readFile()`

- La fonction `readFile()` lit les données du fichier de manière asynchrone.
- Le moyen le plus simple de lire l'intégralité du contenu d'un fichier est avec `fs.readFile`, comme suit:
- **syntaxe**
  - **`fs.readFile (fichier [, options], rappel)`**
- **Paramètres:**
  - **`encoding`**
    - **`encoding`** est un paramètre facultatif qui spécifie le type de codage pour lire le fichier.
    - Les encodages possibles sont "ascii", "utf8" et "base64".
    - Si aucun encodage n'est fourni, la valeur par défaut est null.
  - **`callback`**
    - callback est une fonction à appeler lorsque le fichier a été lu et que le contenu est prêt
    - Cette fonction admet deux arguments, `error` et `data`.
    - S'il n'y a pas d'erreur, **error** sera **null** et data contiendra le contenu du fichier; sinon error contient le message d'erreur.
- **Exemples:**
  - **Fichier: fichier_text.txt**

```
Mon site apcpedagogie.com est un site de formation
Les cours suivants appartiennent à la catégorie Cours de programmation:
php
java
nodejs
javascript
ruby
python
```

apcpedagogie.com

- **Fichier: read_file.js**

```
const http = require('http')
const fs = require('fs')
var path = "fichier_text.txt"

var server = http.createServer((req,res)=>{
    res.writeHead(200,{"Content-Type":"text/html;charset=utf8"})
    fs.readFile(path,'utf8',(err,data)=>{
        if(err) throw err
        res.write(data)
        res.end()
    })
})
console.log('Connected')
server.listen(1988)
```

2. **Fonction `readFileSync()`**

- La méthode `fs.readFileSync()` est une interface de programmation d'application intégrée du module **fs** qui est utilisée pour lire le fichier et renvoyer son contenu.
- Dans la méthode `fs.readFile()`, nous pouvons lire un fichier de manière asynchrone non bloquante, mais dans la méthode `fs.readFileSync()`, nous pouvons lire les fichiers de manière synchrone, c'est-à-dire que nous disons à **node.js** de bloquer d'autres processus parallèles et effectuez le processus de lecture du fichier en cours.
    - **`fs.readFileSync (fichier [, options])`**
- **Paramètres:**
    - **`fichier (chemin)`**
        - **`file`** il prend le chemin relatif du fichier texte. Le chemin peut être de type URL. Le fichier peut également être un descripteur de fichier. Si les deux fichiers sont dans le même dossier, indiquez simplement le nom du fichier entre guillemets.
    - **`options`**
        - C'est un paramètre facultatif qui contient le codage et le drapeau, le codage contient la spécification des données. Sa valeur par défaut est null qui renvoie le tampon brut et l'indicateur contient une indication des opérations dans le fichier. Sa valeur par défaut est "**r**".
- **Exemples:**

apcpedagogie.com

```
const fs = require('fs')

try {
  const data = fs.readFileSync('fichier_text.txt', 'utf8')
  console.log(data)
} catch (err) {
  console.error(err)
}
```

Sommaire du cours Node.js

Source : https://apcpedagogie.com/lire-des-fichiers-avec-node-js · téléchargé le 05/11/2026 · © APC Pédagogie · apcpedagogie.com

apcpedagogie.com