

JavaScript Côté Serveur

Introduction

JavaScript n'est pas seulement utilisé côté client, mais aussi côté serveur grâce à Node.js. Node.js permet de créer des applications web rapides et évolutives.

Table des Matières

1. [Introduction à Node.js](#)
2. [Installation et Configuration](#)
3. [Modules de Base](#)
4. [Gestion des Fichiers](#)
5. [Création d'un Serveur HTTP](#)
6. [Utilisation d'Express](#)
7. [Connexion à une Base de Données](#)
8. [Bonnes Pratiques](#)
9. [Ressources Utiles](#)

Introduction à Node.js

Node.js est un environnement d'exécution JavaScript qui permet d'exécuter du JavaScript côté serveur. Il est basé sur le moteur V8 de Google Chrome.

Installation et Configuration

Installation de Node.js

1. Télécharger Node.js depuis nodejs.org.
2. Installer-le en suivant les instructions.

Vérification de l'Installation

```
node -v  
npm -v
```

Création d'un Projet Node.js

```
mkdir mon-projet  
cd mon-projet  
npm init -y
```

Modules de Base

Importation de Modules

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

Création d'un Module

```
// monModule.js  
module.exports = {  
  saluer: function() {  
    return "Bonjour, monde!";  
  }  
};
```

Utilisation d'un Module

```
const monModule = require('./monModule');  
console.log(monModule.saluer());
```

Gestion des Fichiers

Lecture de Fichier

```
const fs = require('fs');  
  
fs.readFile('fichier.txt', 'utf8', (err, data) => {  
  if (err) throw err;  
  console.log(data);  
});
```

Écriture dans un Fichier

```
fs.writeFile('fichier.txt', 'Bonjour, monde!', err => {  
  if (err) throw err;  
  console.log('Le fichier a été sauvegardé!');  
});
```

Création d'un Serveur HTTP

```
const http = require('http');  
  
const serveur = http.createServer((req, res) => {  
  res.statusCode = 200;  
  res.setHeader('Content-Type', 'text/plain');  
  res.end('Bonjour, monde!');
```

```
});  
  
const port = 3000;  
serveur.listen(port, () => {  
  console.log(`Serveur en cours d'exécution sur le port ${port}`);  
});
```

Utilisation d'Express

Installation d'Express

```
npm install express
```

Création d'une Application Express

```
const express = require('express');  
const app = express();  
  
app.get('/', (req, res) => {  
  res.send('Bonjour, monde!');  
});  
  
const port = 3000;  
app.listen(port, () => {  
  console.log(`Serveur en cours d'exécution sur le port ${port}`);  
});
```

Connexion à une Base de Données

Utilisation de MongoDB avec Mongoose

1. Installer MongoDB et exécuter-le.
2. Installer Mongoose :

```
npm install mongoose
```

Connexion à MongoDB

```
const mongoose = require('mongoose');

mongoose.connect('mongodb://localhost/maBaseDeDonnees', { use
  useNewUrlParser: true, useUnifiedTopology: true });

const db = mongoose.connection;
db.on('error', console.error.bind(console, 'Erreur de connexion:'));
db.once('open', () => {
  console.log('Connecté à la base de données');
});
```

Définition d'un Schéma et d'un Modèle

```
const utilisateurSchema = new mongoose.Schema({
  prenom: String,
  age: Number
});

const Utilisateur = mongoose.model('Utilisateur', utilisateur
  Schema);
```

Création d'un Document

```
const utilisateur = new Utilisateur({ prenom: 'Damien', age:
33 });
utilisateur.save((err) => {
  if (err) return console.error(err);
  console.log('Utilisateur sauvegardé');
});
```

Ressources Utiles

- [Node.js Documentation](#)
- [Express Documentation](#)
- [Mongoose Documentation](#)
- [MDN Web Docs](#)