

INTERACTIVE OBJECT

Formation Web temps réel avec NodeJS

Objectifs :

Dans cette formation, vous apprendrez à développer des applications temps réel avec Node.js. Vous mettrez en oeuvre les concepts de programmation événementielle et asynchrone. Vous utiliserez les API websocket avec Node.js et assurerez la persistance de vos données dans une base NoSQL et la scalabilité de votre architecture.

Pré-requis :

- Connaissances de base du développement web (HTML, CSS, JavaScript).
- Connaissances des principes de développement côté serveur

Durée de formation :

2 jours

Prix indicatifs :

1000€ par personne

**Possibilité de financement avec la DIF, CIF, plan de formation et période de professionnalisation*

Programme

JOUR 1

1. Introduction à NodeJS

- a. Qu'est ce que NodeJS ?
- b. Que pourriez-vous construire?
- c. Qu'est ce que NodeJS n'est pas ?
- d. Bloquant vs non-Bloquant
- e. Les callback
- f. Hello world avec NodeJS
- g. Event Loop
- h. Pourquoi Javascript ?
- i. Timeline des callback

2. Les événements (Events)

- a. Les évènements dans le DOM
- b. Les évènements dans NodeJs.
- c. Émetteur d'évènement personnalisé.
- d. Exemple de serveur Echo.

1. Les flux (Streams)

- 1. Qu'est ce qu'un Stream ?
- 2. Réponse en Streaming
- 3. Comment lire un stream depuis un object Request ?
- 4. Réécrire du serveur d'Echo avec les stream
- 5. Lecteur et écrire de fichier
- 6. Exemple de cas d'utilisation du STREAMING
- 7. Problème du 'Backpressure'
- 8. Les PIPE pour résoudre le BACKPRESSURE
- 9. Exemple d'upload fichier avec progressions

JOUR 2

1. Modules

- a. Utilisation des modules
- b. Création d'un module
- c. Exporter multiple de fonctions
- d. Encapsulation d'une fonction
- e. Gestionnaire des modules NPM
- f. Installation du module avec NPM
- g. Local VS Global
- h. Recherche de modules existants
- i. Définir les dépendances
- j. Versionning sémantique.

2. ExpressJS

- a. Introduction
- b. Les routes avec expressJs
- c. Express + HTML
- d. Les templates avec Express
- e. Gestion des layoutings.
- f. Rest API avec ExpressJS

3. Websocket

- a. Introduction au protocole Websocket
- b. Les librairies pour websocket : FayeJS/ Socket.IO/ SockJS
- c. Fallbacks avec le protocol bayeux
- d. Envoi de message au serveur
- e. Broadcast des messages au clients
- f. Mécanisme du pub/sub
- g. Exemple de serveur d'application de Chat

4. Persistance de donnée

- a. Introduction
- b. Persistance des données avec MongoDB
- c. Scalabilité du serveur websocket server avec Redis