

Objectifs / Durée de la formation

Durée: 2 jours, soit 14 heures

- Connaître l'architecture et le fonctionnement de NodeJS
- Savoir installer et configurer NodeJS et son écosystème de plugins
- Comprendre les concepts d'une architecture asynchrone
- Savoir créer des applications NodeJS
- Mettre en oeuvre les meilleures pratiques de développement et d'utilisation de NodeJS

Participants / Pré-requis

- Développeur Java et JavaScript, Architecte
- Formation initiale ou expérience en développement informatique requise
- Connaissance et pratique régulière de javascript et XML

Moyens pédagogiques

- Formateur expert dans le domaine
- Mise à disposition d'un ordinateur, support de cours remis à chaque participant, vidéo projecteur, tableau blanc et paperboard
- Feuille de présence signée en demi-journée, évaluation des acquis tout au long de la formation, questionnaire de satisfaction, attestation de stage

Programme

1. Rappel des bonnes pratiques javascript

- Visibilité des variables
- Structuration de code
- Les Closures
- Objets et prototypes
- Underscore.js

2. Introduction à nodejs

- Origine du projet
- Le moteur d'interprétation Chrome V8
- La notion de gestion d'événements
- Les différents usages
- Overview de l'écosystème de plugins
- Installation
- Un tout premier exemple

3. Architecture de nodejs

- Asynchronisme
- Programmation par callbacks
- l'event loop

4. Modules et gestion de dépendances

- l'approche modulaire
- NPM et manipulation des modules
- Le fichier package.json en détail
- Modularisation de son code
- Modules noyaux: console, process, os, fs, path et util
- Publier un module sur NPM

5. Node et le Web : HTTP, Connect & Express

- Périmètre du module HTTP
- Connect et ses middlewares
- Serveur Web avec Express
- Générer un serveur Express generator
- Router les requêtes
- Gestion d'une requête et création d'une réponse
- Utiliser un moteur de template
- Ouverture à Passport et Hapi

6. L'asynchrone en détails

- Callback Hell & Pyramid of Doom
- Le module async
- Les promesses
- Exemples d'enchaînement et de parallélisation
- Gestion des erreurs

7. Communication temps réel

- Intégration des websockets HTML5
- Socket.io
- Gestion des transports
- Intégration côté client et côté serveur

8. La gestion des streams

- Description de Streams2
- Buffers
- Type de flux