

# Node.js – Développement d'applications Web

3 jours (21 heures)

Initiation

Présentiel ou classe à distance

à partir de

**1905 € HT** par personne (TVA en vigueur) Repas inclus en présentiel

Contenu mis à jour le 25/09/2026. Document téléchargé le 03/10/2026.

## Objectifs

A l'issue de cette formation, vous serez capable de :

- Anticiper et développer asynchrone dans un environnement multi-utilisateurs
- Identifier les API fondamentales fournies par Node.js
- Décrire NPM et la modularité, de façon approfondie
- Accéder aux données depuis Node.js
- Utiliser les modules Express
- Réaliser des API REST et les tester
- Identifier les dernières fonctionnalités de Node.js
- Tester une application Node.js.

Les compétences visées par cette formation font l'objet d'une évaluation.

## À qui s'adresse cette formation ?

Développeurs, architectes et chefs de projets techniques.

### Prérequis

Avoir une connaissance avancée de JavaScript ou avoir suivi le cours JVS-AV "JavaScript – Programmation avancée".

# Programme

## 1 Pendant votre formation

### Node.js – Développement d'applications Web

Jour 1 – Matin

#### Impacts architecturaux

- Rappels JavaScript
- Présentation rapide de NPM (Node Package Manager)
- Les modules dans Node.js
- Apports de N-API dans les dernières versions
- Présentation de Node.js
- Utilisation de REPL (Read Eval Print Loop)
- Intérêts de JavaScript côté serveur
- Asynchronisme, programmation événementielle
- Stateless

#### Asynchronisme avec Node

- Présentation de l'Event Loop
- Logique asynchrone
- Des callbacks partout
- Nested callbacks
- Gestion d'erreurs
- Patterns asynchrones :
  - Les promesses
  - Le pattern Async / Await

#### Exemples de travaux pratiques (à titre indicatif)

- Conception d'un premier programme Node.js pour comprendre l'aspect asynchrone du langage

Jour 1 – Après-midi

#### Node API

- Global objects
- Event et EventEmitter
- Process objects
- Gestion des processus fils
- Timers
- Les Worker Threads

*Exemples de travaux pratiques (à titre indicatif)*

- *Traitement des évènements avec Node.js*

**Modules**

- Inclusion de modules avec require
- Résolution des noms, chemins et fichiers
- Modules populaires avec :
  - Console
  - Util
  - Timers...

*Exemples de travaux pratiques (à titre indicatif)*

- *Présentation et utilisation des modules les plus utiles*

Jour 2 – Matin

**Programmation serveur**

- Rappels de la programmation asynchrone
- Modules HTTP et NET
- Modules DNS et URL
- Module FS (File System)
- Connect
- Ressources statiques
- Moteurs de templates
- Templates avec EJS

*Exemples de travaux pratiques (à titre indicatif)*

- *Utilisation des modules HTTP, NET, DNS et URL*
- *Utilisation du module FS*
- *Création d'une application Web avec EJS*

Jour 2 – Après-midi

**Back end**

- Modes de connexion
- Accès MongoDB
- Accès Redis
- Accès PostgreSQL
- ORM pour Node.js

*Exemples de travaux pratiques (à titre indicatif)*

- *Ajout de l'accès à une base de données PostgreSQL depuis Node.js*
- *Accès via l'API*

- Accès via l'ORM Sequelize

## Node Package Manager

- Commandes NPM
- Packager un module
- Installation locale ou globale

### Exemples de travaux pratiques (à titre indicatif)

- Utilisation de diverses commandes importantes de NPM en local et global

Jour 3 – Matin

## Express Web Framework

- Configuration
- API Requêtes et Réponses
- API Routes
- Présentation de REST
- Création d'une API REST

### Exemples de travaux pratiques (à titre indicatif)

- Mise en oeuvre d'une application serveur Web avec Express Web
- Création d'API REST accédant à la base PostgreSQL et répondant en JSON à un client JavaScript

Jour 3 – Après-midi

### Exemples de travaux pratiques (à titre indicatif)

- Création et déploiement d'une application

## Tests

- Test avec Mocha
- Assertions

## Les nouveautés

- Node 23 et Node 24 :
  - Require avec les modules ES
  - Exécution de code JS en ligne de commande
  - LocalStorage et SessionStorage
  - Autres fonctionnalités expérimentales et gain de performance
- Node 26 :
  - Temporal API activée par défaut
  - Support TypeScript natif
  - Le Permission Model
  - Le runner natif node:test
  - Autres fonctionnalités expérimentales et gain de performance

## Financement

Plusieurs dispositifs de financement sont accessibles via les OPCO (Opérateurs de Compétences), organismes agréés par le ministère du Travail dont le rôle est d'accompagner, collecter et gérer les contributions des entreprises au titre du financement de la formation professionnelle.

Pour plus d'information, une équipe de gestionnaires SKOLAE Formation spécialisée vous accompagne dans le choix de vos formations et la gestion administrative.

[Nous contacter pour en savoir plus →](#)

## Modalités pédagogiques

Formation délivrée en présentiel ou distanciel\* (blended-learning, e-learning, classe virtuelle, présentiel à distance).

Le formateur alterne entre méthode\*\* démonstrative, interrogative et active (via des travaux pratiques et/ou des mises en situation).

Variables suivant les formations, les moyens pédagogiques mis en oeuvre sont :

- Ordinateurs Mac ou PC (sauf pour certains cours de l'offre Management), connexion internet fibre, tableau blanc ou paperboard, vidéoprojecteur ou écran tactile interactif (pour le distanciel)
- Environnements de formation installés sur les postes de travail ou en ligne
- Supports de cours et exercices

En cas de formation intra sur site externe à M2I, le client s'assure et s'engage également à avoir toutes les ressources matérielles pédagogiques nécessaires (équipements informatiques...) au bon déroulement de l'action de formation visée conformément aux prérequis indiqués dans le programme de formation communiqué.

*\* nous consulter pour la faisabilité en distanciel*

*\*\* ratio variable selon le cours suivi*

### Accessibilité aux personnes en situation de handicap

## Détails de la session

Du 19 octobre 2026 au 21 octobre 2026 (3 jours / 21 heures)

Rennes

Du 19 octobre 2026 au 21 octobre 2026

Rennes

- 19 octobre 2026 ..... 9h00 – 16h00
- 20 octobre 2026 ..... 9h00 – 16h00
- 21 octobre 2026 ..... 9h00 – 16h00