



## Formation Node.js Développement d'applications Web

**Code :** DW08

**Durée :** 3 jours

**Classe :** Présentiel / à distance

### Public

- Ingénieurs

### Prérequis

- Avoir suivi la formation "Développer des sites Web dynamiques avec JavaScript" (CE980) ou disposer de compétences équivalentes.

### Objectifs

- Découvrir et comprendre les principes et le fonctionnement de Node.js
- Se familiariser avec les composants de la plateforme
- Savoir mettre en oeuvre node.js pour tirer profit de la puissance du JavaScript côté serveur
- Connaître les meilleures pratiques de développement avec Node.js

### Programme détaillé

#### 1-INTRODUCTION

- Historique rapide
- Panorama rapide

#### PRINCIPES GÉNÉRAUX

- Fonctionnement interne
- Exemples d'applications
- Avantages et inconvénients
- Installation et découverte
- Application "hello world"

#### 2- Etablissement de la connectivité internet

- Compréhension des couches TCP/IP internet
- Adressage IB des sous-réseaux
- Compréhension des couches de transport TCP/IP
- Exploration des fonctions du routage
- Configuration d'un routeur Cisco
- Réalisation de l'installation et de la configuration initiale d'un routeur
- Exploration du processus de délivrance des paquets
- Configuration du routage statique
- Gestion du trafic avec les ACLs
- Configuration de la connexion internet

#### 3-LES MODULES

- Se servir des modules de base
- NPM

#### 4-SERVEUR WEB

- Ma première application web
- Gérer les requêtes et les réponses HTTP
- Routage des URLs
- Opérations bloquantes et nonbloquantes





## Formation Node.js Développement d'applications Web

**Code :** DW08

**Durée :** 3 jours

**Classe :** Présentiel / à distance

### 5-EXPRESS

- Paramétrage
- Gestion des requêtes HTTP
- Sessions
- Templating

### 6-BONNES PRATIQUES DE DÉVELOPPEMENT

- Organisation (modules, routes, etc.)
- Code

### 7-AUTRES COMPOSANTS

- Socket.IO
- Connect
- Async

### 8-TESTS

- Assert / expect.js
- Mocha / Zombie
- PhantomJS / CasperJS

### 9-PERFORMANCES

- Ssinglethread et eventloop
- Communication interprocessus
- Redis
- Le module "cluster"

