DAWAN

Plan de la formation

www.dawan.fr

Formation Node.js : Fondamentaux

Durée :	4 jours (28 heures)
Tarifs inter-entreprise :	2 475,00 € HT (standard) 1 980,00 € HT (remisé)
Public :	Développeurs Javascript
Pré-requis :	Bonne connaissance de javascript et des normes ecmascript
Objectifs :	Apprendre à utiliser NodeJs et créer un serveur
Modalités pédagogiques, techniques et d'encadrement :	• Formation synchrone en présentiel et distanciel. • Méthodologie basée sur l'Active Learning : 75 % de pratique minimum. • Un PC par participant en présentiel, possibilité de mettre à disposition en bureau à distance un PC et l'environnement adéquat. • Un formateur expert.
Modalités d'évaluation :	• Définition des besoins et attentes des apprenants en amont de la formation. • Auto-positionnement à l'entrée et la sortie de la formation. • Suivi continu par les formateurs durant les ateliers pratiques. • Évaluation à chaud de l'adéquation au besoin professionnel des apprenants le dernier jour de formation.
Sanction :	Attestation de fin de formation mentionnant le résultat des acquis
Référence :	SIT100450-F
Note de satisfaction des participants:	4,93 / 5
Contacts :	commercial@dawan.fr - 09 72 37 73 73
Modalités d'accès :	Possibilité de faire un devis en ligne (www.dawan.fr, moncompteformation.gouv.fr, maformation.fr, etc.) ou en appelant au standard.

Délais d'accès :	Variable selon le type de financement.
Accessibilité :	Si vous êtes en situation de handicap, nous sommes en mesure de vous accueillir, n'hésitez pas à nous contacter à referenthandicap@dawan.fr, nous étudierons ensemble vos besoins

Introduction

Présentation des versions de node
Principe d'un serveur Web
Comparaison entre serveur
Rappels de Javascript

Atelier : Télécharger et installer NodeJS

Présentation de Node.js

Gérer les versions de node (nvm vs n)
L'intérêt de javascript côté serveur
La single-threaded event loop
L'API non bloquante
Le moteur V8 de Google

Node.jset Ecmascript

Historique d'Ecmascript
Présentation du mode strict
Support d'ecmascript dans NodeJS (node.green)

Utiliser le gestionnaire de paquets

Présentation et description
npm: node package manager
Initialiser un projet avec npm
Présentation de la bibliothèque de module
Installation locale ou globale
Le fichier package.json
Gérer les dépendances (production et developpement)
npm et npx
Ajouter des scripts

Atelier : Créer différents scripts utiles à un projet

Les variables globales

La documentation de l’API de Node
Les classes Global, Console (log, errors, trace)
Les variables __filename et __dirname
Capter et émettre des évènements
Les Timers

Atelier : Executer notre premier script

Manipuler les fichiers

Présentation du module FileSystem
Gérer les chemins et les urls
Créer, supprimer, renommer des fichiers et dossiers
Lire le contenu d'un fichier

Atelier : Manipuler des fichiers

Callback et Promise

Présentation des callbacks (fonction de rappel)
Principe des Promises (les promesses)
Callbacks vs Promises
Implémentation des promesses
Transformation d'une fonction de rappel en promesse
Travailler les promesses avec la syntaxe async/await

Atelier : Utiliser les promesses

Créer des modules Node.js

Isoler des fonctionnalités
Exporter des fonctions
Inclure des modules dans un projet
Différence entre module CommonJS et EcmaScript

Atelier : Créer ses propres modules

Créer un serveur avec Node.js

Le module "http"
Request et Response
Modifier les entêtes
Parser une url
Afficher des contenus au format HTML, JSON, XML
Gérer l'affichage des fichiers css, image et js

Atelier : Créer un serveur de fichiers statiques

Créer un serveur avec Express

Présentation de "Express"
Créer un projet Express
Les middlewares
Gérer les verbes http
Ajouter un moteur de templates
Récupérer les paramètres de requête, d'url et de formulaire
Les modules Morgan, Helmet, Cors
Utiliser Express avec Typescript

Atelier : Utiliser Express pour créer une application Web ou une API REST avec authentification

Les bases de données

Se connecter à des bases de données SQL et NoSQL
Récupérer un connecteur de SGBD
Utiliser et configurer un ORM : TypeORM
Créer une classe d’entité
Executer des requêtes et récupérer les resultats
Mettre en place des relations entre deux entités

Atelier : Connecter une base de données

Tester une application

Présentation de Mocha
Les assertions dans Node.js : assert
Le module Chai
Les tests synchrones et asynchrones
Les hooks

Atelier : Lancer plusieurs tests