

Node.js

Description

Durant ce cours de 5 jours, les participants apprendront à concevoir, construire, tester et déployer une API RESTful en utilisant Node.js et Mongo. Les sujets suivants seront abordés : travailler avec un middleware, tester dans Node.js l'organisation d'applications, la modélisation de données, l'interrogation de données avec Mongoose, l'utilisation de JSON Web Token, la sécurisation des routes et enfin le déploiement.

Contenu du cours

Module 1 : Programme d'installation

- Leçon 1 : Introduction
- Leçon 2 : Installation de Node
- Leçon 3 : Qu'est-ce que Node?
- Leçon 4 : Pourquoi devrais-je utiliser Node?

Module 2 : Notions de base de Node.js

- Leçon 1 : Utiliser Require
- Leçon 2 : Require de vos propres fichiers
- Leçon 3 : Utiliser des modules tiers
- Leçon 4 : Redémarrage de l'application avec Nodemon
- Leçon 5 : Obtenir des saisies de l'utilisateur
- Leçon 6 : Entrée simplifiée avec Yargs
- Leçon 7 : Travailler avec JSON
- Leçon 8 : Refactoring pour la réutilisabilité
- Leçon 9 : Notes de lecture et réutilisabilité
- Leçon 10 : Débogage des applications Node.js
- Leçon 11 : Débogage via Chrome Dev Tools
- Leçon 12 : Identifier les Notes
- Leçon 13 : Exiger des arguments et des arguments avancés
- Leçon 14 : Fonctions de flèche (arrow)

Module 3 : Node.js asynchrone

- Leçon 1 : Notions de base asynchrones
- Leçon 2 : Pile d'appel et boucle d'événement
- Leçon 3 : Fonctions de callback et API
- Leçon 4 : Objets d'impression
- Leçon 5 : De quoi est faite une requête HTTP?
- Leçon 6 : Codage de l'entrée de l'utilisateur
- Leçon 7 : Erreurs de callback
- Leçon 8 : Résumé des rappels
- Leçon 9 : Câblage de la météo
- Leçon 10 : Chaînage des rappels ensemble
- Leçon 11 : Intro aux promesses ES6 (Promises)
- Leçon 12 : Promesses avancées
- Leçon 13 : Fonctionnalités supplémentaires

Module 4 : Serveurs Web et déploiement d'applications

- Leçon 1 : Créer un serveur Web
- Leçon 2 : Rendu de modèles avec des données
- Leçon 3 : Templates avancés
- Leçon 4 : Express Middleware
- Leçon 5 : Ajouter un contrôle de version (Git)
- Leçon 6 : Configuration des clés GitHub et SSH
- Leçon 7 : Déployer vos applications
- Leçon 8 : Ajouter une nouvelle fonctionnalité et déployer

Module 5 : Test de vos applications

- Leçon 1 : Mocha et tests de base
- Leçon 2 : Tests de surveillance et de redémarrage automatique
- Leçon 3 : Utilisation d'une bibliothèque d'assertion
- Leçon 4 : Test du code asynchrone
- Leçon 5 : Test des applications Express: Partie I
- Leçon 6 : Test des applications Express: Partie II
- Leçon 7 : Test d'organisation avec describe()

Module 6 : API MongoDB, Mongoose et REST

- Leçon 1 : Installation de MongoDB et Robomongo (Windows)
- Leçon 2 : Construire un vocabulaire NoSQL
- Leçon 3 : Connexion à Mongo et écriture de données
- Leçon 4 : L'ObjectId
- Leçon 5 : Récupération des données
- Leçon 6 : Mise en place du Repo
- Leçon 7 : Supprimer des documents
- Leçon 8 : Mise à jour des données
- Leçon 9 : L'ORM Mongoose
- Leçon 10 : Mise en place de Mongoose
- Leçon 11 : Validateurs, types et valeurs par défaut
- Leçon 12 : Installation de Postman
- Leçon 13 : Point de fin de la création de ressources - POST / todos
- Leçon 14 : Requêtes Mongoose et validation d'ID
- Leçon 15 : Déployer l'API sur Heroku

Module 7 : Sécurité et authentification

- Leçon 1 : Configuration du modèle utilisateur
- Leçon 2 : JWT et Hashing
- Leçon 3 : Génération de jetons Auth et définition des en-têtes
- Leçon 4 : Routes privées et middleware Auth
- Leçon 5 : Hashing mots de passe
- Leçon 6 : Seeding Base de données de test avec les utilisateurs
- Leçon 7 : Amélioration de la configuration de l'application

Module 8 : Applications Web en temps réel avec Socket.io

- Leçon 1 : Créer un nouveau projet

- Leçon 2 : Ajouter Socket.io à une application
- Leçon 3 : Émettre et écouter des événements
- Leçon 4 : Émettre et écouter des événements (Solution)
- Leçon 5 : Événements de broadcasting
- Leçon 6 : Générateur de messages et tests
- Leçon 7 : Reconnaissance des événements
- Leçon 8 : Formulaire de message et jQuery
- Leçon 9 : Géolocalisation Partie I
- Leçon 10 : Géolocalisation Partie II
- Leçon 11 : Timestamps et formatage avec Moment
- Leçon 12 : Impression des Timestamps des messages
- Leçon 13 : Moustache.js
- Leçon 14 : Autoscrolling
- Leçon 15 : Ajouter une page
- Leçon 16 : Données de passage
- Leçon 17 : Socket.io Rooms
- Leçon 18 : Stockage des utilisateurs avec des classes ES6: Partie I
- Leçon 19 : Stockage des utilisateurs avec des classes ES6: Partie II
- Leçon 20 : Liste des utilisateurs
- Leçon 21 : Envoyer des messages seulement à la Room
- Leçon 22 : Nouvelles idées de fonctionnalités

Module 9 : Async / Await

- Leçon 1 : Async / Await Project Setup
- Leçon 2 : Async / Await Basics
- Leçon 3 : Un exemple du monde réel
- Leçon 4 : Gestion des erreurs et attente de la fonction asynchrone
- Leçon 5 : Utilisation d'Async / Await dans l'API Todo

Lab / Exercices

- Pendant le cours, les participants sont encouragés à participer activement à l'expérience d'apprentissage en exécutant des exemples de fichiers et en effectuant des tâches de codage pendant les travaux de lab. Chaque session de lab vous permet de comparer votre solution à celle de l'instructeur

Documentation

- Support de cours numérique inclus

Profils des participants

- Personne cherchant à lancer ses propres applications Node ou souhaitant devenir un développeur Node freelance

Connaissances Préalables

- Bonne compréhension de base JavaScript standard (variables, instructions if, fonctions de base, objets de base)

Objectifs

- Construire et déployer une API entièrement fonctionnelle pour une application avec authentification
- Créer leurs propres API

- Prétendre à un poste de développeur Node professionnel capable de développer, tester et déployer des applications de production dans le monde réel

Niveau

Fondamental

Prix de l'inscription en Présentiel (CHF)

3800

Prix de l'inscription en Virtuel (CHF)

3550

Durée (Nombre de Jours)

5

Reference

NODEJS