Neo4j, prise en main et développement

Cours Pratique de 2 jours - 14h
Réf : NLP - Prix 2024 : 1 620€ HT

Neo4J est une base NoSQL orientée graphe où les données sont représentées comme des nœuds reliés par des arcs (chaque nœud étant constitué de couples clé-valeur). Durant cette formation, les développeurs apprendront à requêter directement la base de données puis à réaliser des applications qui exploitent Neo4J.

OBJECTIFS PÉDAGOGIQUES

À l'issue de la formation l'apprenant sera en mesure de :

Comprendre l'émergence des bases de données NoSQL face aux SGBDR

Savoir installer Neo4j et l'intégrer dans ses développements

Comprendre la structure des données et le vocabulaire associé dans Neo4j

Savoir manipuler des données sous forme de graphe

Améliorer les performances avec les index

Accéder aux données de Neo4j depuis des programmes tiers (Java, JavaScript, Python) et via les API REST ou GraphQL

LE PROGRAMME

dernière mise à jour : 06/2021

1) Introduction

- La domination historique des bases de données relationnelles.
- L'apparition des bases NoSQL et les différents types de bases NoSQL.
- Une présentation de Neo4j, son positionnement et ses avantages.

Démonstration : Des exemples d'utilisation de Neo4j.

2) Installation

- Les plateformes supportées et la configuration requise.
- Les installations sous Linux et sous Windows.
- La création de service, le lancement et la configuration.

Travaux pratiques : Installer Neo4j sur son poste de développement (avec Linux ou Windows).

3) La structure des données sous Neo4j

- Qu'est-ce qu'un graphe ?
- Les nœuds : les propriétés et les labels.
- Les relations : les propriétés, les types et les sens.
- Les propriétés : les types et les tableaux.
- Les chemins et les parcours de données.
- Les schémas, les index et les contraintes.

Travaux pratiques : Manipulations de différents graphes.

4) Le langage Cypher

- La syntaxe : les expressions, les variables, les paramètres, les opérateurs, les motifs, les tableaux, etc.
- Les clauses : les lectures, les écritures, les unions, les projections, etc.

PARTICIPANTS

Toute personne devant développer des solutions autour de Neo4j ou souhaitant découvrir le potentiel d'une base de données en graphe.

PRÉREQUIS

Connaissance des principes des bases de données ainsi que des langages de programmation Java, Python ou JavaScript.

COMPÉTENCES DU FORMATEUR

Les experts qui animent la formation sont des spécialistes des matières abordées. Ils ont été validés par nos équipes pédagogiques tant sur le plan des connaissances métiers que sur celui de la pédagogie, et ce pour chaque cours qu'ils enseignent. Ils ont au minimum cinq à dix années d'expérience dans leur domaine et occupent ou ont occupé des postes à responsabilité en entreprise.

MODALITÉS D'ÉVALUATION

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de QCM, mises en situation, travaux pratiques…
Le participant complète également un test de positionnement en amont et en aval pour valider les compétences acquises.

MOYENS PÉDAGOGIQUES ET TECHNIQUES

• Les moyens pédagogiques et les méthodes d'enseignement utilisés sont principalement : aides audiovisuelles, documentation et support de cours, exercices pratiques d'application et corrigés des exercices pour les stages pratiques, études de cas ou présentation de cas réels pour les séminaires de formation.
• À l'issue de chaque stage ou séminaire, ORSYS fournit aux participants un questionnaire d'évaluation du cours qui est ensuite analysé par nos équipes pédagogiques.
• Une feuille d'émargement par demi-journée de présence est fournie en fin de formation ainsi qu'une attestation de fin de formation si le stagiaire a bien assisté à la totalité de la session.

MODALITÉS ET DÉLAIS D'ACCÈS

L'inscription doit être finalisée 24 heures avant le début de la formation.

ACCESSIBILITÉ AUX PERSONNES HANDICAPÉES

Vous avez un besoin spécifique d'accessibilité ? Contactez Mme FOSSE, référente handicap, à l'adresse suivante psh-accueil@orsys.fr pour étudier au mieux votre demande et sa faisabilité.

- Les fonctions : les mathématiques, les agrégations, les scalaires, liées aux tableaux et aux chaînes, etc.
- Les index : la création, l'utilisation, la suppression, l'ajout de contraintes.
- L'optimisation de requêtes et plans d'exécution.
- L'importation de données.

Travaux pratiques : Importation de la base d'exemple et exécution de requêtes via l'interface Web.

5) Accéder à Neo4j depuis vos propres programmes

- Un aperçu de l'embarquement de Neo4j au sein d'un programme Java.
- Les drivers fournis par Neo4j (.Net, Java, JavaScript, Python).
- Un focus sur un driver au choix (Java, JavaScript, Spring ou Python).
- Les connexions au serveur et authentification, les requêtes en lectures/écritures et les transactions.

Travaux pratiques : Exécution de requêtes via le driver Java, JavaScript ou Python.

6) L'API REST et GraphQL

- L'API REST versus GraphQL.
- Racine et authentification. Flux JSON. Service transactionnel.
- La manipulation directe de nœuds et la manipulation directe de relations.
- Le service historique (le transactionnel et le non transactionnel).
- Le recensement des propriétés, les labels et les types de relations utilisés dans la base.
- La gestion de l'indexation et des contraintes.

Travaux pratiques : L'exécution de requêtes via l'API REST.

LES DATES

Nous contacter