Apache Spark avancé : traitement Big Data

Référence : **BDDAPA001** Durée : **3 jours (21 heures)** Certification : **Aucune**

CONNAISSANCES PRÉALABLE

- Avoir de bonnes connaissances en Python/Scala, SQL et Big Data

PROFIL DES STAGIAIRES

- Data engineers, data scientists

OBJECTIFS

- Optimiser les traitements Spark
- Utiliser les RDD et DataFrame efficacement
- Gérer les ressources et la parallélisation
- Déboguer et profiler des applications Big Data

CERTIFICATION PRÉPARÉE

- Aucune

MÉTHODES PÉDAGOGIQUES

- Mise à disposition d'un poste de travail par stagiaire
- Remise d'une documentation pédagogique numérique pendant le stage
- La formation est constituée d'apports théoriques interactif, démonstrations en direct, travaux dirigés, ateliers collaboratif, mises en situation professionnelles, projet fil rouge, études de cas réels, d'ateliers d'optimisation et de tuning, d'ateliers de diagnostic et de profiling
- Le suivi de cette formation donne lieu à la signature d'une feuille d'émargement par demi-journée de formation

FORMATEUR

- Consultant-formateur expert Data

MÉTHODES D'ÉVALUATION DES ACQUIS

- Validation des prérequis envoyée en amont de la formation. En l'absence de réponse d'un ou plusieurs participants, un temps sera consacré en ouverture de session pour prendre connaissance du positionnement de chaque stagiaire sur les objectifs pédagogiques évalués
- Validation progressive des compétences tout au long de la formation via exercices pratiques, quiz techniques, études de cas, travaux d'optimisation, analyses de performances et restitution des travaux réalisés dans le cadre du projet fil rouge
- Auto-évaluation des acquis par le stagiaire via un questionnaire
- Attestation des compétences acquises envoyée au stagiaire
- Attestation de fin de stage adressée avec la facture

ACCESSIBILITÉ DE LA FORMATION

- Si vous avez besoin d'un aménagement pour suivre votre formation dans de bonnes conditions, nous vous invitons à contacter notre référent handicap au 01 71 19 70 30 ou à l'adresse referent.handicap@edugroupe.com. Un échange en amont nous permettra d'étudier avec vous les adaptations possibles.

CONTENU DU COURS

1. JOUR 1 - Comprendre et optimiser les fondations de Spark (7h)

2. Architecture avancée d'Apache Spark (2h)

- Écosystème Apache Spark
- Architecture Driver / Executor
- Cluster Manager (Standalone, YARN, Kubernetes)
- DAG Scheduler
- Task Scheduler
- Lazy Evaluation
- Jobs, Stages et Tasks
- Fault Tolerance
- 💡 *Mises en pratique : Analyse d'une architecture Spark ; Visualisation du cycle d'exécution d'un traitement ; Exploration de l'interface Spark UI*

3. Maîtriser les RDD et leur optimisation (2h30)

- Création et manipulation de RDD
- Transformations et actions
- Partitions
- Persistence et Cache
- Checkpointing
- Gestion de la sérialisation
- Optimisation des traitements distribués
- 💡 *Mises en pratique : Création de traitements RDD ; Analyse des performances ; Optimisation des temps d'exécution*

4. Exploiter efficacement les DataFrames et DataSets (2h30)

- DataFrames et DataSets
- Catalyst Optimizer
- Tungsten Engine
- Schémas de données
- Lecture et écriture de données
- Formats Parquet, ORC, JSON et CSV
- Bonnes pratiques de développement
- 💡 *Mises en pratique : Traitement de jeux de données volumineux ; Comparaison RDD / DataFrame ; Analyse des plans d'exécution*

5. JOUR 2 - Optimisation des performances et gestion des ressources (7h)

6. Optimiser les traitements distribués (3h)

- Shuffle Operations
- Broadcast Variables
- Broadcast Joins
- Partitionnement
- Repartition et Coalesce
- Gestion des skewed data
- Optimisation des jointures
- Stratégies de traitement
- 💡 *Mises en pratique : Optimisation de traitements existants ; Réduction des temps de calcul ; Analyse des gains de performance*

7. Gérer les ressources du cluster Spark (2h)

- Gestion mémoire Spark
- Allocation dynamique
- Configuration des Executors
- CPU et parallélisme
- Paramètres de tuning
- Gestion des ressources YARN et Kubernetes
- Capacity Planning
- 💡 *Mises en pratique : Configuration d'un cluster Spark ; Ajustement des ressources ; Analyse de consommation mémoire*

8. Traiter des données massives avec Spark SQL (2h)

- Architecture Spark SQL
- Optimisation des requêtes
- Fenêtres analytiques
- Agrégations distribuées
- Jointures avancées
- Catalogues de données
- Gestion des métadonnées
- 💡 *Mises en pratique : Développement de requêtes analytiques ; Optimisation de traitements SQL ; Analyse des plans physiques et logiques*

9. JOUR 3 - Débogage, monitoring et optimisation avancée (7h)

10. Déboguer et profiler des applications Spark (2h30)

- Spark UI
- Logs d'exécution
- Monitoring des jobs
- Analyse des stages
- Analyse des tâches
- Profiling des traitements
- Gestion des erreurs et exceptions
- 💡 *Mises en pratique : Analyse d'applications dégradées ; Diagnostic de problèmes de performance ; Identification des causes racines*

11. Industrialiser les traitements Big Data (2h)

- Structuration des applications Spark
- Gestion des environnements
- Déploiement des jobs
- Gestion des dépendances
- Journalisation
- Supervision
- Bonnes pratiques DevOps
- 💡 *Mises en pratique : Préparation d'un traitement pour la production ; Mise en place de mécanismes de supervision ; Gestion des incidents*

12. Projet fil rouge : optimisation d'une plateforme Big Data Spark (2h30)

- Analyse d'un pipeline Big Data
- Étude des performances
- Optimisation des traitements
- Ajustement des ressources
- Débogage
- Industrialisation
- Synthèse des recommandations
- 💡 *Mises en pratique : Audit complet d'une application Spark ; Optimisation des traitements et du cluster ; Réduction des temps d'exécution ; Présentation des résultats et débriefing collectif*