

Storm : Flux de données

Référence : PYCB038

Durée : 2 jours

Certification : Aucune

CONNAISSANCES PREALABLES

- Connaissance d'un langage de programmation comme Java ou Python.

PROFIL DES STAGIAIRES

- Chefs de projet. • Data scientists. • Développeurs.

OBJECTIFS

- Savoir mettre en œuvre Storm pour le traitement de flux de données.

CERTIFICATION PREPAREE

Aucune

METHODES PEDAGOGIQUES

- Mise à disposition d'un poste de travail par stagiaire
- Remise d'une documentation pédagogique papier ou numérique pendant le stage
- La formation est constituée d'apports théoriques, d'exercices pratiques, de réflexions et de retours d'expérience
- Le suivi de cette formation donne lieu à la signature d'une feuille d'émargement

FORMATEUR

Consultant-Formateur expert Bigdata

METHODE D'EVALUATION DES ACQUIS

- Auto-évaluation des acquis par le stagiaire via un questionnaire
- Attestation de fin de stage adressée avec la facture

CONTENU DU COURS

Introduction

- Présentation de Storm : fonctionnalités, architecture, langages supportés
- Définitions : spout, bolt, topology

Architecture

- Etude des composants d'un cluster Storm : master node 'nimbus' et worker nodes
- Positionnement par rapport à un cluster Hadoop. Le modèle de données. Différents types de flux

Premiers pas

- Configuration d'un environnement de développement
- Installation d'un cluster Storm. Travaux pratiques sur le projet storm-starter

Flux de données

- Définition du nombre de flux dans un nœud, création de topologies regroupant des flux entre différents nœuds, communication entre flux en JSON, lecture de flux d'origines diverses (JMS, Kafka, ...)

Haute disponibilité

- Tolérance aux pannes: principe de fiabilisation des master node, workers node, nimbus
- Garantie de traitement des flux: principe, paramètres TOPOLOGY_MESSAGE_TIMEOUT_SECS, TOPOLOGY_ACKERS
- Traitements temps réel avec Trident. Scalabilité : parallélisme dans un cluster storm, ajouts de nœuds, commande 'storm rebalance'