CLOUDERA

DENG-254: Preparing with Cloudera Data Engineering

Cloudera Data Engineering powered by Apache Spark, Hive, and Airflow

Course Overview

Course Type

Instructor-led training course

Level

Intermediate

Duration

4 days

Platform

CDP Public Cloud
Data Engineering Service

Topics Covered

- HDFS
- Apache YARN
- Apache Spark
- Apache Hive

About This Training

This four-day hands-on training course delivers the key concepts and knowledge developers need to use Apache Spark to develop high-performance, parallel applications on the Cloudera Data Platform (CDP).

Hands-on exercises allow students to practice writing Spark applications that integrate with CDP core components. Participants will learn how to use Spark SQL to query structured data, how to use Hive features to ingest and denormalize data, and how to work with “big data” stored in a distributed file system.

After taking this course, participants will be prepared to face real-world challenges and build applications to execute faster decisions, better decisions, and interactive analysis, applied to a wide variety of use cases, architectures, and industries.

What Skills You Will Gain

During this course, you will learn how to:

- Distribute, store, and process data in a CDP cluster
- Write, configure, and deploy Apache Spark applications
- Use the Spark interpreters and Spark applications to explore, process, and analyze distributed data
- Query data using Spark SQL, DataFrames, and Hive tables
- Deploy a Spark application on the Data Engineering Service

Who Should Take This Course?

This course is designed for developers and data engineers. All students are expected to have basic Linux experience, and basic proficiency with either Python or Scala programming languages. Basic knowledge of SQL is helpful. Prior knowledge of Spark and Hadoop is not required.

Other Training That Might Interest You

- *Apache Spark Application Performance Tuning*

DENG-254: Preparing with Cloudera Data Engineering

Training Outline (Page 2 of 2)

HDFS Introduction
- HDFS Overview
- HDFS Components and Interactions
- Additional HDFS Interactions
- Ozone Overview
- Exercise: Working with HDFS

YARN Introduction
- YARN Overview
- YARN Components and Interaction
- Working with YARN
- Exercise: Working with YARN

Working with RDDs
- Resilient Distributed Datasets (RDDs)
- Exercise: Working with RDDs

Working with DataFrames
- Introduction to DataFrames
- Exercise: Introducing DataFrames
- Exercise: Reading and Writing DataFrames
- Exercise: Working with Columns
- Exercise: Working with Complex Types
- Exercise: Combining and Splitting DataFrames
- Exercise: Summarizing and Grouping DataFrames
- Exercise: Working with UDFs
- Exercise: Working with Windows

Introduction to Apache Hive
- About Hive
- Transforming data with Hive QL

Working with Apache Hive
- Exercise: Working with Partitions
- Exercise: Working with Buckets
- Exercise: Working with Skew
- Exercise: Using Serdes to Ingest Text Data
- Exercise: Using Complex Types to Denormalize Data

Hive and Spark Integration
- Hive and Spark Integration
- Exercise: Spark Integration with Hive

Distributed Processing Challenges
- Shuffle
- Skew
- Order

Spark Distributed Processing
- Spark Distributed Processing
- Exercise: Explore Query Execution Order

Spark Distributed Persistence
- DataFrame and Dataset Persistence
- Persistence Storage Levels
- Viewing Persisted RDDs
- Exercise: Persisting DataFrames

Data Engineering Service
- Create and Trigger Ad-Hoc Spark Jobs
- Orchestrate a Set of Jobs Using Airflow
- Data Lineage using Atlas
- Auto-scaling in Data Engineering Service

Workload XM
- Optimize Workloads, Performance, Capacity
- Identify Suboptimal Spark Jobs

Appendix: Working with Datasets in Scala
- Working with Datasets in Scala
- Exercise: Using Datasets in Scala