Hadoop Developer with Spark

Duration: 4 Days

Prerequisites:

- Basic understanding of programming principles and data structures.
- Familiarity with any high-level programming language, preferably Java, Scala, or Python, as Spark examples may be given in these languages.
- Basic knowledge of Linux or Unix-based systems for navigating through the command line.
- Fundamental understanding of database concepts and query language (SQL).
- An introductory comprehension of big data and distributed systems.
- Willingness to learn new technologies and adapt to the Hadoop ecosystem.
- Please note that while prior experience with Hadoop or Spark is beneficial, it is not mandatory. This course is designed to introduce participants to Apache Hadoop and Spark, and it will cover the necessary components and tools throughout the training modules.

Course Description:

The Hadoop Developer with Spark course is designed to equip learners with the skills needed to build big data processing applications using Apache Hadoop and Apache Spark. It is an excellent pathway for those preparing for the CCA 175 certification, as it covers the necessary topics and provides hands-on experience. Throughout the course, participants will explore the Hadoop ecosystem, understand HDFS architecture, and work with YARN for resource management. The course delves into the basics of Apache Spark, DataFrame operations, and Spark SQL for querying data, which are crucial for the CCA 175 certification. Learners will also gain practical knowledge of RDDs, data persistence, and Spark streaming, all of which are part of the CCA 175 exam syllabus. By the end of the course, participants will be proficient in writing, configuring, and running Spark applications, setting them on the path to becoming certified Hadoop professionals with a focus on Spark.

Target Audience:

- Data Engineers
- Software Developers with a focus on big data
- Big Data Analysts
- System Administrators interested in big data infrastructure
- IT professionals looking to specialize in data processing
- Data Scientists who want to add big data processing skills
- Technical Leads managing big data projects
- Database Professionals transitioning to big data roles
- Graduates aiming to build a career in big data
- IT Architects designing big data solutions systems

Target Audience:

- Understand the fundamental concepts of Apache Hadoop and its role in the big data ecosystem.
- Gain proficiency in HDFS architecture, data ingestion, storage operations, and cluster components.
- Learn distributed data processing using YARN and develop the capability to work with YARN applications.
- Acquire hands-on experience with Apache Spark, including Spark Shell, Datasets, DataFrames, RDDs, and Spark SQL.
- Master data transformation, querying, and aggregation techniques using Spark's core abstractions and APIs.
- Develop and configure robust Spark applications, understanding deployment modes and application tuning.
- Grasp the concept of distributed processing, including partitioning strategies and job execution planning.
- Learn data persistence methods and storage levels within Spark for optimized data handling.
- Explore common data processing patterns, including iterative algorithms and machine learning with Spark's MLlib.
- Dive into real-time data processing with Apache Spark Streaming, understanding DStreams, window operations, and integrating with sources like Apache Kafka.

Course Outlines:

Module 1: Introduction to Apache Hadoop and the Hadoop Ecosystem
- Apache Hadoop Overview
- Data Ingestion and Storage
- Data Processing
- Data Analysis and Exploration
- Other Ecosystem Tools
- Introduction to the Hands-On Exercises

Module 2: Apache Hadoop File Storage
- Apache Hadoop Cluster Components
- HDFS Architecture
- Using HDFS

Module 3: Distributed Processing on an Apache Hadoop Cluster
- YARN Architecture
- Working With YARN

Module 4: Apache Spark Basics
- What Is Apache Spark?
- Starting the Spark Shell
- Using the Spark Shell
- Getting Started with Datasets and DataFrames
- DataFrame Operations

Module 5: Working with DataFrames and Schemas
- Introduction to Data-rames
- Exercise: Introducing DataFrames
- Exercise: Reading and Writing DataFrames
- Exercise: Working with Columns
- Exercise: Working with Complex Types
- Exercise: Combining and Splitting Datarrames
- Exercise: Summarizing and Grouping DF
- Exercise: Working with UDFs
- Exercise: Working with Windows
- Eager and Lazy Execution

COURSE OUTLINE

Module 6: Analyzing Data with DataFrame Querles
- Querying DataFrames Using Column Exp.
- Grouping and Aggregation Queries
- Joining DataFrames

Module 7: Introduction to Apache Hive
- About Hive
- Transforming data with Hive QL

Module 8: 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

Module 9: Hive and Spark Integration
- Hive and Spark Integration
- Exercise: Spark integration with Hive

Module 10: RDD Overview
- RDD Overview
- RDD Data Sources
- Creating and Saving RDDs
- RDD Operations

Module 11: Transforming Data with RDDs
- Writing and Passing Transformation Functions
- Transformation Execution
- Converting Between RDDs and DataFrames

Module 12: Aggregating Data with Pair RDDs
- Key-Value Pair RDDs
- Map-Reduce
- Other Pair RDD Operations

Module 13: Querying Tables and Views with Apache Spark SQL
- Querying Tables in Spark Using SQL
- Querying Files and Views
- The Catalog API
- Comparing Spark SQL, Apache Impala, and Apache Hive-on-Spark

Module 14: Working with Datasets in Scala
- Datasets and DataFrames
- Creating Datasets
- Loading and Saving Datasets
- Dataset Operations

Module 15: Writing, Configuring, and Running Apache Spark Applications
- Writing a Spark Application
- Building and running an application
- Application Deployment Mode
- The Spark Application Web UI
- Configuring Application Properties

Module 16: Distributed Processing
- Review: Apache Spark on a Cluster
- RDD Partitions
- Example: Partitioning in Queries
- Stages and Tasks
- Job Execution Planning
- Example: Catalyst Execution Plan
- Example: RDD Execution Plan

Module 17: Distributed Processing Challenges
- Shuffle
- Skew
- Order

Module 18: Distributed Data Persistence
- DataFrame and Dataset Persistence
- Persistence Storage Leveis
- Viewing Persisted RDDs

Module 19: Machine Learning with Spark ML
- Common Apache Spark Use Cases
- Iterative Algorithms in Apache Spark: Machine Learning, Graph Processing
- Introduction to MLlib- Varlous ML algorithms supported by Miib
- ML model with Spark ML
- Exercise: Implement Linear regression
- Exercise: Implement logistic regression
- Exercise: Implement Random Forest
- Exercise: Implement k-means

Module 20: Apache Spark Streaming: Introduction to DStreams
- Apache Spark Streaming Overview
- Example: Streaming Request Count
- Streams
- Developing Streaming Applications

Module 21: Apache Spark Streaming: Processing Multiple Batches
- Multi-Batch Operations
- Time Slicing
- State Operations
- Sliding Window Operations
- Preview: Structured Streaming

Module 22: Apache Spark Streaming: Data Sources
- Streaming Data Source Overview
- Apache Flume and Apache Kafka Data Sounces
- •Example: Using a Katka Direct Data Source