Cloudera Data Analyst Training

32 hours

Take your knowledge to the next level with Cloudera's Apache Hadoop Training

Course objectives:

Data Analyst Training course will teach you to apply traditional data analytics and business intelligence skills to big data. This course presents the tools data professionals need to access, manipulate, transform, and analyze complex data sets using SQL and familiar scripting languages.

Advance Your Ecosystem Expertise

Apache Hive makes transformation and analysis of complex, multi-structured data scalable in Cloudera environments. **Apache Impala** enables real-time interactive analysis of the data stored in Hadoop using a native SQL environment. Together, they make multi-structured data accessible to analysts, database administrators, and others without Java programming expertise.

Who Should Attend:

This course is designed for data analysts, business intelligence specialists, developers, system architects, and database administrators. Some knowledge of SQL is assumed, as is basic Linux command-line familiarity. Prior knowledge of Apache Hadoop is not required.

Course Contents:

Hadoop Fundamentals

- The Motivation for Hadoop
- Hadoop Overview
- Data Storage: HDFS
- Distributed Data Processing: YARN, MapReduce, and Spark
- Data Processing and Analysis: Hive, and Impala
- Database Integration: Sqoop
- Other Hadoop Data Tools
- Exercise Scenario Explanation

Introduction to Apache Hive and Impala

- What Is Hive?
- What Is Impala?
- Why Use Hive and Impala?
- Schema and Data Storage
- Comparing Hive and Impala to Traditional Databases
- Use Cases

Querying with Apache Hive and Impala

- Databases and Tables
- Basic Hive and Impala Query Language Syntax
- Data Types

- Using Hue to Execute Queries
- Using Beeline (Hive's Shell)
- Using the Impala Shell

Common Operators and Built-In Functions

- Operators
- Scalar Functions
- Aggregate Functions

Data Management

- Data Storage
- Creating Databases and Tables
- Loading Data
- Altering Databases and Tables
- Simplifying Queries with Views
- Storing Query Results

Data Storage and Performance

- Partitioning Tables
- Loading Data into Partitioned Tables
- When to Use Partitioning
- Choosing a File Format
- Using Avro and Parquet File Formats

Working with Multiple Datasets

- UNION and Joins
- Handling NULL Values in Joins
- Advanced Joins

Analytic Functions and Windowing

- Using Common Analytic Functions
- Other Analytic Functions
- Sliding Windows

Complex Data

- Complex Data with Hive
- Complex Data with Impala

Analyzing Text

- Using Regular Expressions with Hive and Impala
- Processing Text Data with SerDes in Hive
- Sentiment Analysis and n-grams in Hive
-

Apache Hive Optimization

- Understanding Query Performance
- Cost-Based Optimization and Statistics
- Bucketing
- ORC File Optimizations

Apache Impala Optimization

- How Impala Executes Queries
- Improving Impala Performance

Extending Apache Hive and Impala

- Custom SerDes and File Formats in Hive
- Data Transformation with Custom Scripts in Hive
- User-Defined Functions
- Parameterized Queries

Choosing the Best Tool for the Job

- Comparing Hive, Impala, and Relational Databases
- Which to Choose?

Apache Kudu

- What Is Kudu?
- Kudu Tables
- Using Impala with Kudu