Hadoop Architecture and SQL

Course ID#: 1190-600-ZZ-W

Hours: 21

Course Content

Course Description:

In this course, students will learn the Hadoop Architecture and SQL starting at the most basic level and going to the most advanced level with many examples. At the completion of this course, students will have a deeper knowledge and understanding of the Hadoop Architecture and SQL and how to write it.

Topics include:

- Basic SQL Functions
- The WHERE Clause
- Distinct Vs. Group By
- Aggregation Function
- Join Functions
- Date Functions
- OLAP Functions
- Temporary Tables
- Sub-query Functions
- Strings
- Interrogating the Data
- View Functions
- Set Operators
- Data Manipulation Language (DML)
- Statistical Aggregate Functions

Target Student:

This course is designed for anyone who has a desire to learn the Hadoop Architecture and SQL from beginners to an advanced audience. This course is completely customizable by the client.

Prerequisites:

None

Hadoop Architecture and SQL

Course ID#: 1190-600-ZZ-W

Hours: 21

Topics:

Introduction

- History of Data Warehousing
- The Growth of Computer Data and Use of Databases
- Definition of Enterprise Data vs. Big Data
- Why is Big Data Important?
- Why Enhance Your Company's Data Warehousing Capabilities? ("We already have standard reports we run monthly.")
- Benefits of Big Data for Your Company
- Management Considerations
- Customer Considerations
- An Industry Example
- What does it mean to be data driven?

Module 1: The Concepts of Hadoop

- What is Hadoop All About?
- There is a Named Node and Up to 4000 Data Nodes
- The Named Node's Directory Tree
- The Data Nodes
- Hive MetaStore
- Data Layout and Protection – Step 1
- Data Layout and Protection – Step 2
- Data Layout and Protection – Step 3
- Data Layout and Protection – Step 4
- How are Blocks Distributed Amongst the Cluster?
- What is Parallel Processing?
- The Basics of a Single Computer
- Data in Memory is Fast as Lightning
- Parallel Processing Of Data
- Introduction to Hive
- Commodity Hardware Servers are Configured for Hadoop
- Commodity Hardware Allows Nodes to Scale Forever (Linear)
- The Named Node
- The Data Node's Responsibilities
- All Reducers, Some Reducers or a Single Reducer
- A Table has Columns and Rows
- Hadoop has Linear Scalability
- The Architecture of a Hadoop Data Warehouse
- How to Find All Databases in the System
- Setting Your Default Database With the USE Command
- List the Tables in a Database With the Show Tables Command
- Show Basic Table Information with the Describe Command
- Show Detailed Table Information Using Describe Extended
- The Show Functions Command Lists all System Functions
- Describe Function Command Provides Function Information
- Describe Function Extended Command Provides Details

Module 2: The Basics of SQL

- Introduction
- SELECT * (All Columns) in a Table
- SELECT Specific Columns in a Table
- Commas in the Front or Back?
- Place your Commas in front for better Debugging Capabilities
- Sort the Data with the ORDER BY Keyword
- ORDER BY Can Use the Column Number
- SORT BY Can Be Used Instead of ORDER BY
- Changing the ORDER BY to Descending Order
- Using the SORT BY in DESC Mode
- Major Sort vs. Minor Sorts
- SORT BY Using Major and Minor Sorts
- SORT BY Defaults to Ascending
- Sorts are Alphabetical, NOT Logical
- Using A CASE Statement to Sort Logically

Module 2: The Basics of SQL continued
- How to ALIAS a Column Name
- A Missing Comma can by Mistake become an Alias
- Comments using Double Dashes are Single Line Comments
- Comments for Multi-Lines
- Comments for Multi-Lines as Double Dashes per Line
- A Great Technique for Comments to Look for SQL Errors

Module 3: The WHERE Clause
- The WHERE Clause limits Returning Rows
- Case Sensitivity is Important
- Character Data needs Single Quotes, but Numbers Don't
- You Cannot Use the Alias in the Where Clause
- NULL means NO DATA so Equals Null Returns No Rows
- Use IS NULL or IS NOT NULL for Null Values
- NULL is UNKNOWN DATA so NOT Equal won't Work
- Use IS NULL or IS NOT NULL when dealing with NULLs
- Using Greater Than
- Using Greater Than or Equal To (>=)
- AND in the WHERE Clause
- Troubleshooting AND
- OR in the WHERE Clause
- Troubleshooting Or
- Troubleshooting Character Data
- Using Different Columns in an AND Statement
- Quiz – How many rows will return?
- Answer to Quiz – How many rows will return?
- What is the Order of Precedence?
- Using Parentheses to change the Order of Precedence
- An IN List is Another Technique
- Using an IN List in place of OR
- The IN List Can Use Character Data
- Using a NOT IN List
- BETWEEN is Inclusive
- NOT BETWEEN is Also Inclusive
- LIKE uses Wildcards Percent '%' and Underscore '_'
- LIKE command Underscore is Wildcard for one Character
- Quiz –Who Has the Letter 'n' in the 3rd Position
- Answer - Who Has the Letter 'n' in the 3rd Position
- LIKE is Case Sensitive
- LIKE Command to Find the Last Character of a Last_Name
- LIKE Command to Find Multiple Characters
- LIKE Command to Find Either Character
- Answer – What Data is Left Justified and What is Right?
- An Example of Data with Left and Right Justification
- A Visual of CHARACTER Data vs. VARCHAR Data
- Use the TRIM command to remove spaces on CHAR Data Escape Characters
- RTRIM command Removes Trailing spaces on CHAR Data

Module 4: Distinct, Group By, Limit and Sample
- The Distinct Command
- The Distinct Command
- Distinct vs. GROUP BY
- Quiz – How many rows come back from the Distinct?
- Answer – How many rows come back from the Distinct?
- Limit Will Limit the Returning Rows
- Limit Works Brilliantly with ORDER BY
- TABLESAMPLE
- Collect_List and Collect_Set

Module 5: Aggregation

- Quiz – You calculate the Answer Set in your own Mind
- Answer – You calculate the Answer Set in your own Mind
- There are Five Aggregates
- Quiz – How many Columns and Rows come back?
- Answer – How Many Columns and Rows Come Back?
- Quiz – What Happens With This Query?
- Answer To Quiz – What Happens With This Query?
- GROUP BY when Aggregates and Normal Columns Mix
- GROUP BY Delivers one row per Group
- Limiting Rows and Improving Performance with WHERE
- Keyword HAVING tests Aggregates after they are Totaled
- Getting the Average Values Per Column
- Average Values Per Column For all Columns in a Table

Module 6: Join Functions

- Hadoop Joins The picture below #3 sentence is a run on sentence break up
- The Shuffle Join (1 of 2)
- The Shuffle Join (2 of 2)
- Shuffle
- Map Strategy
- Duplication of the Smaller Table across All-AMPs
- Using Buckets For Table Joins
- Sort-Merge Bucket Join Needs No Shuffling or Duplication
- A Two-Table Join Using Traditional Join Syntax
- A Two-Table Join Using ANSI Syntax
- Traditional Join Using a Table Alias
- ANSI Join Using a Table Alias
- ANSI Join Using a Table Alias With Keyword AS
- ANSI Join Using the Keyword JOIN Instead of INNER JOIN
- You Can Fully Qualify All Columns for Clarity
- A Two-Table Join in Action
- Quiz – Can You Finish the Join Syntax?
- Answer to Quiz – Can You Finish the Join Syntax?
- Another Way to Write a Join
- A Cartesian Product Join
- Quiz – Can You Find the Error?
- Answer to Quiz – Can You Find the Error?
- Super Quiz – Can You Find the Difficult Error?
- Answer to Super Quiz – Can You Find the Difficult Error?
- Quiz – Which rows from both tables Won't Return?
- Answer to Quiz – Which rows from both tables Won't Return?
- LEFT OUTER JOIN
- LEFT OUTER JOIN Results
- A LEFT SEMI JOIN Replaces a Subquery
- What is a LEFT SEMI JOIN?
- RIGHT OUTER JOIN
- RIGHT OUTER JOIN Example and Results
- FULL OUTER JOIN
- FULL OUTER JOIN Results
- Which Tables are the Left and Which are the Right?
- Answer - Which Tables are the Left and Which are the Right?
- INNER JOIN with an Additional WHERE Clause
- INNER JOIN with an Additional AND Clause
- OUTER JOIN with Additional WHERE Clause
- OUTER JOIN with Additional AND Clause
- OUTER JOIN with Additional AND Clause Results
- Quiz – Why is this Considered an INNER JOIN?
- Evaluation Order For Outer Queries
- Cartesian Product Join
- The CROSS JOIN With a WHERE Clause

Module 6: Join Functions continued

- The CROSS JOIN With a WHERE Clause Answer Set
- The CROSS JOIN With an ON Clause
- The CROSS JOIN With an ON Clause Answer Set
- The Self Join
- How would you Join these two tables?
- An Associative Table is a Bridge that Joins Two Tables
- Quiz – Can you Write the 3-Table Join?
- Answer – Can you Write the 3-Table Join?
- Quiz – Can you Place the ON Clauses at the End?
- Answer – Can you Place the ON Clauses at the End?
- The 5-Table Join – Logical Insurance Model
- Quiz - Write a Five Table Join Using ANSI Syntax
- Answer - Write a Five Table Join Using ANSI Syntax
- Quiz –Re-Write this putting the ON clauses at the END
- Answer –Re-Write this putting the ON clauses at the END
- The Nexus Query Chameleon Writes the SQL for Users.

Module 7: Sub-query Functions

- An IN List is much like a Subquery
- An IN List Never has Duplicates – Just like a Subquery
- The Subquery
- The Three Steps of How a Basic Subquery Works
- These are Equivalent Queries
- The Final Answer Set from the Subquery
- Quiz- Answer the Difficult Question
- Answer to Quiz- Answer the Difficult Question
- Should you use a Subquery or a Join?
- Quiz- Write the Subquery
- Answer to Quiz- Write the Subquery
- Quiz- Write the More Difficult Subquery
- Answer to Quiz- Write the More Difficult Subquery
- Quiz- Write the Subquery with an Aggregate
- Answer to Quiz- Write the Subquery with an Aggregate
- Quiz- Write the Correlated Subquery
- Answer to Quiz- Write the Correlated Subquery
- Quiz- Write the NOT Subquery
- Answer to Quiz- Write the NOT Subquery
- Quiz- Write the Subquery using a WHERE Clause
- Answer - Write the Subquery using a WHERE Clause
- Quiz – How many rows return on a NOT IN with a NULL?
- Answer – How many rows return on a NOT IN with a NULL?
- How to handle a NOT IN with potential NULL Values
- Using a Correlated Exists
- How a Correlated Exists matches up
- The Correlated NOT Exists

Module 8: Date Functions

- Current_Date and Current_Timestamp Functions
- Extracted the Date From a Time Data Type
- Adding and Subtracting Days from a Time Column
- Adding Days and Providing a Discount
- Getting the Date Extracted From a Time Data Type
- Getting the Date Extracted From a Time Using Substring
- Getting the Date Extracted From a Time Using Concat
- Getting the Date Extracted in Day-Month-Year Format
- Getting the Date Extracted in Day-Month-Year Format
- The Date in Perfect Day-Month-Year Format With CASE
- Getting a Count of All Orders Per Year Per Month

Module 8: Date Functions continued

- Extracting the Year, Month and Day From a Time Data Type
- Extracting the Hour, Minute and Second From Time Data
- The ADD_MONTHS Command
- The ADD_MONTHS Command to Add Years
- Using Cast to Change a Data Type
- The Months_Between Command
- NEXT_DAY Command Finds a Future Day of the Week
- NEXT_DAY Command Finds a Future Day of the Week
- Interval Day
- Hadoop Calendar Knows Leap Year
- Interval Day, Month, Year Plus Cast

Module 9: OLAP Functions

- The Row_Number Command
- Quiz – How did the Row_Number Reset?
- Quiz – How did the Row_Number Reset?
- Using a Derived Table and Row_Number
- Ordered Analytics OVER
- RANK and DENSE RANK
- RANK Defaults to Ascending Order
- Getting RANK to Sort in DESC Order
- RANK() OVER and PARTITION BY
- PERCENT_RANK() OVER
- PERCENT_RANK() OVER with 14 rows in Calculation
- PERCENT_RANK() OVER with 21 rows in Calculation
- CSUM – Rows Unbounded Preceding Explained
- CSUM – Making Sense of the Data
- CSUM – Making Even More Sense of the Data
- CSUM – The Major and Minor Sort Key(s)
- The ANSI CSUM – Getting a Sequential Number
- Reset with a PARTITION BY Statement
- PARTITION BY only Resets a Single OLAP not ALL of them
- CURRENT ROW AND UNBOUNDED FOLLOWING
- Different Windowing Options
- Moving Sum has a Moving Window
- How ANSI Moving SUM Handles the Sort
- Quiz – How is that Total Calculated?
- Answer to Quiz – How is that Total Calculated?
- Moving SUM every 3-rows Vs a Continuous Average
- PARTITION BY Resets an ANSI OLAP
- The Moving Window is Current Row and Preceding
- Moving Average
- Moving Average Using a CAST Statement
- Moving Average every 3-rows Vs a Continuous Average
- PARTITION BY Resets an ANSI OLAP
- COUNT OVER for a Sequential Number
- COUNT OVER without Rows Unbounded Preceding
- Quiz – What caused the COUNT OVER to Reset?
- Answer to Quiz – What caused the COUNT OVER to Reset?
- The MAX OVER Command
- MAX OVER with PARTITION BY Reset
- The MIN OVER Command
- MIN OVER without Rows Unbounded Preceding
- The CSUM for Each Product_Id and the Next Start Date
- How Ntile Works
- Ntile
- Ntile Continued
- Ntile Percentile
- Another Ntile Example
- Using Quantiles (Partitions of Four)
- NTILE With a Single Sort Key
- NTILE Using a Value of 10
- NTILE With a Partition
- Using FIRST_VALUE
- FIRST_VALUE

Module 9: OLAP Functions continued

- FIRST_VALUE After Sorting by the Highest Value
- FIRST_VALUE with Partitioning
- FIRST_VALUE Combined with Row_Number
- FIRST_VALUE And Row_Number with Different Sort
- Using LAST_VALUE
- LAST_VALUE
- Using LAG and LEAD
- LEAD
- LEAD
- LEAD With Partitioning
- Finding the First Occurrence
- Finding the Last Occurrence
- Using LEAD
- Using LEAD with an Offset of 2
- Using LAG
- Using LAG with an Offset of 2
- LAG
- LAG with Partitioning
- CUME_DIST
- CUME_DIST With a Partition
- SUM(SUM(n))

Module 10: Temporary Tables

- There are two types of Temporary Tables
- CREATING A Derived Table
- CREATING A Derived Table using the WITH Command
- The Same Derived Query shown Two Different Ways
- Most Derived Tables Are Used To Join To Other Tables
- The Three Components of a Derived Table
- Visualize This Derived Table
- Our Join Example with A Different Column Aliasing Style
- Column Aliasing Can Default For Normal Columns
- Our Join Example With the WITH Syntax
- Quiz - Answer the Questions
- Answer to Quiz - Answer the Questions
- Clever Tricks on Aliasing Columns in a Derived Table
- Two Derived Tables Joining to a Permanent Table
- The Key to Multiple WITH Tables
- Joining Two WITH Tables to a Permanent Table
- Using a Derived Table and Row_Number
- LEAD
- Finding the First Occurrence
- Finding the Last Occurrence
- Creating a Temporary Table
- Creating, Populating and Querying a Temporary Table
- Creating a Temporary Table Using the LIKE Keyword
- Creating a Temporary Table Using the LIKE Keyword
- Creating a Temporary Table and Populating it Simultaneously
- Creating a Temporary Table that Joins Multiple Tables
- Many Users Can Use the Same Temporary Table Name

Module 11: Strings

- The LENGTH Command Counts Characters
- The LENGTH Command – Spaces can Count too
- The LENGTH Command Doesn't Count Trailing Spaces
- UPPER and LOWER Commands
- Using the LOWER Command
- A LOWER Command Example
- Using the UPPER Command
- An UPPER Command Example
- Non-Letters are Unaffected by UPPER and LOWER
- SOUNDEX
- REGEXP_REPLACE
- Concatenation

Module 11: Strings continued

- The TRIM Command trims both Leading and Trailing Spaces
- SUBSTRING and SUBSTR are equal, but use different syntax
- How SUBSTRING Works with NO ENDING POSITION
- Using SUBSTRING to move Backwards
- How SUBSTRING Works with an Ending Position of 0
- An Example using SUBSTRING and LENGTH Together
- Concatenation and SUBSTRING
- The Context_Ngrams Function
- Sentences Function
- Explode Ngrams Sentences to Find the 5 Most Popular Words
- Explode Ngrams Sentences to Find the 5 Most Two-Words
- Explode Ngrams Sentences For the Top 5 Trigrams
- Explode Ngrams Sentences Finding Words Following a Phrase
- Explode Ngrams Sentences Finding Words Following a Phrase

Module 12: Interrogating the Data

- Quiz – Fill in the Answers for the NULLIF Command
- Quiz – Fill in the Answers for the NULLIF Command
- The COALESCE Command – Fill In the Answers
- COALESCE is Equivalent to This CASE Statement
- Some Great CAST (Convert and Store) Examples
- Quiz - The Basics of the CASE Statements
- Answer to Quiz - The Basics of the CASE Statements
- Using an ELSE in the Case Statement
- Using an ELSE as a Safety Net
- Rules for a Valued Case Statement
- Rules for a Searched Case Statement
- Valued Case Vs. A Searched Case
- The CASE Challenge
- The CASE Challenge Answer
- Combining Searched Case and Valued Case
- A Trick for getting a Horizontal Case
- Nested Case
- Put a CASE in the ORDER BY

Module 13: View Functions

- The Fundamentals of Views
- Creating a Simple View to Restrict Sensitive Columns
- Describe a View
- Describe Extended a View
- You SELECT From a View
- Creating Views to Protect Sensitive Columns and Rows
- Querying Sensitive Columns and Rows in a View
- Basic Rules for Views
- How to Modify a View
- An Exception to the ORDER BY Rule inside a View
- Views Are Sometimes CREATED for Formatting
- Creating a View to Join Tables Together
- How to Alias Columns in a View CREATE
- The Standard Way Most Aliasing is done
- What Happens When Both Aliasing Options Are Present
- Resolving Aliasing Problems in a View CREATE
- Answer to Resolving Aliasing Problems in a View CREATE
- Aggregates on View Aggregates
- Altering a Table After a View Has Been Created

Module 14: Creating Databases and Tables

- Creating a Database
- The Basics of Creating a Table
- The ROW FORMAT will be Delimited or Serde
- Hive Data Type Fundamentals

Module 14: Creating Databases and Tables continued

- An Example of a Table Using All Basic Data Types
- Settings so Hive can Automatically Partition a Table
- Creating a Partitioned Table
- Creating an External Table
- Creating an External Table With a Specific Location
- INSERT/SELECT is One Method of Loading Data
- Using Buckets For Table Joins
- Defining Skewed Tables
- Defining a Table Location
- Creating a Text File Table
- Distribute By for Loading Data
- Sort By on Data Loads
- Cluster By Distributes and Sorts by the Same Key
- Hive does Not Store Data, But HDFS Does in These Formats
- Creating Tables as a Text file
- Hive SerDes Means Serializer/Deserializer
- Creating a Table as a SERDE
- Creating Tables as a SERDE with Advanced Options
- Creating Tables as an RCFile
- Creating Tables as ORC files
- Altering a Table to Add a Column
- Renaming a Table
- Dropping a Table
- Creating a Table Using a CTAS
- Creating a Table Using a CTAS Join
- Creating a Temporary Table Using a CTAS
- Creating a Temporary Table Using a LIKE Command
- Collecting Statistics – Cost Based Optimization (CBO)
- Collecting Statistics on Particular Columns of a Table
- Best Practices for Hive Cost Based Optimization
- Setting the Following Properties to Enable CBO
- Vectorization
- Use the DESCRIBE FORMATTED Function to See Statistics
- Hadoop Numeric Data Types
- Hadoop Date/Time Data Types
- Hadoop String Data Types Continued
- Hadoop Miscellaneous Data Types Continued

Module 15: Data Manipulation Language (DML)

- INSERT Syntax # 1
- INSERT example with Syntax 1
- INSERT Syntax # 2
- INSERT example with Syntax 2
- INSERT/SELECT Command
- INSERT/SELECT example using All Columns (*)
- INSERT/SELECT example with Less Columns
- DELETE and TRUNCATE Examples

Module 16: Statistical Aggregate Functions

- Numeric Manipulation Functions
- Finding the Cube Root
- Ceiling Gets the Smallest Integer Not Smaller Than X
- Floor Finds the Largest Integer Not Greater Than X
- The Round Function and Precision
- The Conv Function
- The Stats Table
- Compute_Stats Function
- The STDDEV_POP Function
- A STDDEV_POP Example
- The STDDEV_SAMP Function
- A STDDEV_SAMP Example
- The VAR_POP Function
- A VAR_POP Example
- The VAR_SAMP Function
- A VAR_SAMP Example
- The VARIANCE Function
- A VARIANCE Example
- The CORR Function
- A CORR Example

Module 16: Statistical Aggregate Functions continued

- Another CORR Example so you can Compare
- The COVAR_POP Function
- A COVAR_POP Example
- Another COVAR_POP Example so you can Compare
- The COVAR_SAMP Function
- A COVAR_SAMP Example
- Another COVAR_SAMP Example so you can Compare
- Using GROUP BY

Module 17: Hadoop EXPLAIN

- There are Many Options to See an EXPLAIN Plan
- Explain Output has Three Parts
- EXPLAIN EXTENDED and the Abstract Syntax Tree
- EXPLAIN EXTENDED Stage Plans and Stage Dependencies
- EXPLAIN DEPENDENCY Keywords in an Explain
- EXPLAIN AUTHORIZATION Keywords in an Explain
- Using a WHERE Clause Explains a Predicate
- EXPLAIN With an ORDER BY Statement