

Course duration

- 2 days

Course Benefits

- Learn to gain a deeper knowledge and understanding of the Amazon Redshift SQL and how to write it.

Course Outline

1. Basic SQL Functions
 1. Finding the Current Schema on the Leader Node
 2. Getting Things Setup in Your Search Path
 3. Five Details You Need To Know About The Search_Path
 4. Introduction
 5. SELECT * (All Columns) in a Table
 6. SELECT Specific Columns in a Table
 7. Commas in the Front or Back?
 8. Place your Commas in front for better Debugging Capabilities
 9. Sort the Data with the ORDER BY Keyword
 10. ORDER BY Defaults to Ascending
 11. Use the Name or the Number in your ORDER BY Statement
 12. Two Examples of ORDER BY using Different Techniques
 13. Changing the ORDER BY to Descending Order
 14. NULL Values sort First in Ascending Mode (Default)
 15. NULL Values sort Last in Descending Mode (DESC)
 16. Major Sort vs. Minor Sorts
 17. Multiple Sort Keys using Names vs. Numbers
 18. Sorts are Alphabetical, NOT Logical
 19. Using A CASE Statement to Sort Logically
 20. How to ALIAS a Column Name
 21. A Missing Comma can by Mistake become an Alias
 22. Comments using Double Dashes are Single Line Comments
 23. Comments for Multi-Lines
 24. Comments for Multi-Lines As Double Dashes Per Line
 25. A Great Technique for Comments to Look for SQL Errors
2. The WHERE Clause
 1. Using Limit to bring back a Sample
 2. Using Limit With an Order By Statement
 3. The WHERE Clause limits Returning Rows
 4. Using a Column ALIAS throughout the SQL

5. Double Quoted Aliases are for Reserved Words and Spaces
 6. Character Data needs Single Quotes in the WHERE Clause
 7. Character Data needs Single Quotes, but Numbers Don't
 8. NULL means UNKNOWN DATA so Equal (=) won't Work
 9. Use IS NULL or IS NOT NULL when dealing with NULLs
 10. NULL is UNKNOWN DATA so NOT Equal won't Work
 11. Use IS NULL or IS NOT NULL when dealing with NULLs
 12. Using Greater Than Or Equal To (>=)
 13. AND in the WHERE Clause
 14. Troubleshooting AND
 15. OR in the WHERE Clause
 16. Troubleshooting Or
 17. Troubleshooting Character Data
 18. Using Different Columns in an AND Statement
 19. What is the Order of Precedence?
 20. Using Parentheses to change the Order of Precedence
 21. Using an IN List in place of OR
 22. The IN List is an Excellent Technique
 23. IN List vs. OR brings the same Results
 24. Using a NOT IN List
 25. Null Values in a NOT IN List Bring Back No Rows
 26. Another Technique for Handling Nulls with a NOT IN List
 27. BETWEEN is Inclusive
 28. NOT BETWEEN is Also Inclusive
 29. LIKE uses Wildcards Percent '%' and Underscore '_'
 30. LIKE command Underscore is Wildcard for one Character
 31. LIKE Command Works Differently on Char Vs Varchar
 32. The Ilike Command Is NOT Case Sensitive
 33. Troubleshooting LIKE Command on Character Data
 34. Introducing the TRIM Command
 35. Numbers are Right Justified and Character Data is Left
 36. An Example of Data with Left and Right Justification
 37. A Visual of CHARACTER Data vs. VARCHAR Data
 38. Use the TRIM command to remove spaces on CHAR Data
 39. Like and Your Escape Character of Choice
 40. Like and the Default Escape Character
 41. Similar To Operators
 42. Similar To Operators
 43. Similar To Example With Lower Case Letters
 44. Similar To Example With Lower and Upper Case Letters
 45. Similar To Example With Multiple Occurrences
 46. Multiple Occurrences Must Be Consecutive
3. Distinct Vs Group By AND TOP
 1. The Distinct Command
 2. Distinct vs. GROUP BY
 3. TOP Command
 4. TOP Command is brilliant when ORDER BY is Used!

5. What is the Difference Between TOP and LIMIT?
4. Aggregation
 1. The Rules of Aggregation
 2. There are Five Aggregates
 3. Troubleshooting Aggregates
 4. GROUP BY when Aggregates and Normal Columns Mix
 5. GROUP BY Delivers one row per Group
 6. GROUP BY Dept_No or GROUP BY the same thing
 7. Limiting Rows and Improving Performance with WHERE
 8. WHERE Clause in Aggregation limits unneeded Calculations
 9. Keyword HAVING tests Aggregates after they are Totaled
 10. Keyword HAVING is like an Extra WHERE Clause for Totals
5. Join Functions
 1. A Two-Table Join Using Traditional Syntax
 2. A two-table join using Non-ANSI Syntax with Table Alias
 3. You Can Fully Qualify All Columns
 4. A two-table join using ANSI Syntax
 5. Both Queries have the same Results and Performance
 6. LEFT OUTER JOIN
 7. LEFT OUTER JOIN Results
 8. Left Outer Joins Compatible with Oracle
 9. RIGHT OUTER JOIN
 10. RIGHT OUTER JOIN Example and Results
 11. Right Outer Joins Compatible with Oracle
 12. FULL OUTER JOIN
 13. FULL OUTER JOIN Results
 14. Which Tables are the Left and Which are the Right?
 15. INNER JOIN with Additional AND Clause
 16. ANSI INNER JOIN with Additional AND Clause
 17. ANSI INNER JOIN with Additional WHERE Clause
 18. OUTER JOIN with Additional WHERE Clause
 19. OUTER JOIN with Additional AND Clause
 20. OUTER JOIN with Additional AND Clause Results
 21. The DREADED Product Join
 22. The DREADED Product Join Results
 23. The Horrifying Cartesian Product Join
 24. The ANSI Cartesian Join will ERROR
 25. The CROSS JOIN
 26. The CROSS JOIN Answer Set
 27. The Self Join
 28. The Self Join with ANSI Syntax
 29. How would you Join these two tables?
 30. An Associative Table is a Bridge that Joins Two Tables
 31. The 5-Table Join – Logical Insurance Model
6. Date Functions
 1. Current_Date
 2. TIMEOFDAY()

3. SYSDATE Returns a Timestamp With Microseconds
4. GETDATE Returns a Timestamp Without Microseconds
5. Add or Subtract Days from a date
6. The ADD_MONTHS Command Returns a Timestamp
7. The ADD_MONTHS Command With Trunc Removes Time
8. ADD_MONTHS Command to Add -Year or -Years
9. Dateadd Function And Add_Months Function are Different
10. The EXTRACT Command
11. EXTRACT from DATES and TIME
12. EXTRACT with DATE and TIME Literals
13. EXTRACT of the Month on Aggregate Queries
14. The Datediff command
15. The Datediff Function on Column Data
16. The Date_Part Function Using a Date
17. The Date_Part Function Using a Time
18. Date_Part Abbreviations
19. The to_char command
20. Conversion Functions
21. Conversion Function Templates
22. Conversion Function Templates Continued
23. Formatting A Date
24. A Summary of Math Operations on Dates
25. Using a Math Operation to find your Age in Years
26. Date Related Functions
27. A Side Title example with Reserved Words as an Alias
28. Implied Extract of Day, Month and Year
29. DATE_PART Function
30. DATE_PART Function using an ALIAS
31. DATE_TRUNC Function
32. DATE_TRUNC Function using TIME
33. MONTHS_BETWEEN Function
34. MONTHS_BETWEEN Function in Action
35. ANSI TIME
36. ANSI TIMESTAMP
37. Redshift TIMESTAMP Function
38. Redshift TO_TIMESTAMP Function
39. Redshift NOW() Function
40. Redshift TIMEOFDAY Function
41. Redshift AGE Function
42. Time Zones
43. Setting Time Zones
44. Using Time Zones
45. Intervals for Date, Time and Timestamp
46. Using Intervals
47. Troubleshooting The Basics of a Simple Interval
48. Interval Arithmetic Results
49. A Date Interval Example

50. A Time Interval Example
 51. A DATE Interval Example
 52. A Complex Time Interval Example using CAST
 53. A Complex Time Interval Example using CAST
 54. The OVERLAPS Command
 55. An OVERLAPS Example that Returns No Rows
 56. The OVERLAPS Command using TIME
 57. The OVERLAPS Command using a NULL Value
7. Window Functions
1. Cumulative Sum (CSUM)
 2. CSUM – The Sort Explained
 3. CSUM – Rows Unbounded Preceding Explained
 4. CSUM – Making Sense of the Data
 5. CSUM – Making Even More Sense of the Data
 6. CSUM – The Major and Minor Sort Key(s)
 7. Reset with a PARTITION BY Statement
 8. PARTITION BY only Resets a Single OLAP not ALL of them
 9. ANSI Moving Window is Current Row and Preceding n Rows
 10. How ANSI Moving SUM Handles the Sort
 11. Moving SUM every -rows Vs a Continuous Average
 12. Partition By Resets an ANSI OLAP
 13. Moving Average
 14. The Moving Window is Current Row and Preceding
 15. How Moving Average Handles the Sort
 16. Moving Average every -rows Vs a Continuous Average
 17. Partition By Resets an ANSI OLAP
 18. RANK Defaults to Ascending Order
 19. Getting RANK to Sort in DESC Order
 20. RANK() OVER and PARTITION BY
 21. RANK() OVER And LIMIT
 22. PERCENT RANK() OVER
 23. PERCENT_RANK() OVER with rows in Calculation
 24. PERCENT_RANK() OVER with rows in Calculation
 25. COUNT OVER for a Sequential Number
 26. The MAX OVER Command
 27. MAX OVER with PARTITION BY Reset
 28. The MIN OVER Command
 29. The Row_Number Command
 30. Standard Deviation Functions Using STDDEV / OVER
 31. Standard Deviation Functions and STDDEV / OVER Syntax
 32. STDDEV / OVER Example
 33. VARIANCE / OVER Syntax
 34. Variance Functions Using VARIANCE / OVER
 35. Using VARIANCE with PARTITION BY Example
 36. Using FIRST_VALUE and LAST_VALUE
 37. Using FIRST_VALUE
 38. FIRST_VALUE

39. FIRST_VALUE After Sorting by the Highest Value
40. FIRST_VALUE with Partitioning
41. FIRST_VALUE Combined with Row_Number
42. FIRST_FIRST_VALUE And Row_Number with Different Sort
43. Using LAG and LEAD
44. Using LEAD
45. Using LEAD with a PARTITION Statement
46. Using LEAD With an Offset of
47. Using LEAD With an Offset of and a PARTITION
48. Using LAG
49. Using LAG with a PARTITION Statement
50. Using LAG With an Offset of
51. Using LAG With an Offset of and a PARTITION
52. CUME_DIST
53. CUME_DIST With a Partition
54. RANK and DENSE RANK
55. LISTAGG Function
56. LISTAGG Basic Example
57. Another Example of LISTAGG
58. LISTAGG With a Pipe-Separated List
59. LISTAGG With a Comma-Separated List in Groups
60. MEDIAN Function
61. MEDIAN Example
62. MEDIAN with Partitioning and a WHERE Clause
63. MEDIAN with Partitioning
64. NTILE Function
65. How Ntile Works
66. Ntile
67. Ntile Continued
68. Ntile Percentile
69. Another Ntile Example
70. Using Tertiles (Partitions of Four)
71. NTILE
72. NTILE Using a Value of
73. NTILE With a Partition
74. NTH_VALUE Function and Syntax
75. NTH_VALUE Arguments
76. NTH_VALUE
77. NTH_VALUE With Partition
78. NTH_VALUE With Partition
79. PERCENTILE_CONT Function Description and Syntax
80. Final Result Information About PERCENTILE_CONT
81. PERCENTILE_CONT Function Arguments
82. PERCENTILE_CONT Example
83. PERCENTILE_CONT Example with Percentage Change
84. PERCENTILE_CONT With PARTITION Example
85. PERCENTILE_CONT With PARTITION and (.)

- 86. PERCENTILE_DISC Function Description and Syntax
- 87. PERCENTILE_DISC Function Arguments
- 88. PERCENTILE_DISC Example
- 89. PERCENTILE_DISC Example with Percentage Change
- 90. PERCENTILE_DISC With PARTITION Example
- 91. PERCENTILE_DISC With PARTITION and (.)
- 92. RATIO_TO_REPORT Function
- 93. RATIO_TO_REPORT Example
- 94. RATIO_TO_REPORT Example with Partitioning

8. Temporary Tables

- 1. CREATING A Derived Table
- 2. The Three Components of a Derived Table
- 3. Naming the Derived Table
- 4. Aliasing the Column Names in The Derived Table
- 5. Visualize This Derived Table
- 6. Most Derived Tables Are Used To Join To Other Tables
- 7. Multiple Ways to Alias the Columns in a Derived Table
- 8. Our Join Example With A Different Column Aliasing Style
- 9. Column Aliasing Can Default For Normal Columns
- 10. CREATING A Derived Table using the WITH Command
- 11. A Join Using the WITH Syntax
- 12. WITH
- 13. A WITH Clause That Produces Two Tables
- 14. Finding the First Occurrence of a Row using WITH
- 15. Finding the First Occurrence of a Row using a Derived Table
- 16. Finding the Last Occurrence Using a WITH Derived Table
- 17. Finding the Last Occurrence Using a Derived Table
- 18. The Same Derived Query shown Three Different Ways
- 19. Clever Tricks on Aliasing Columns in a Derived Table
- 20. A Derived Table lives only for the lifetime of a single query
- 21. An Example of Two Derived Tables in a Single Query
- 22. Create Table Syntax
- 23. Basic Temporary Table Examples
- 24. More Advanced Temporary Table Examples
- 25. Advanced Temporary Table Examples
- 26. Performing a Deep Copy
- 27. Deep Copy Using the Original DDL
- 28. Deep Copy Using A CTAS
- 29. Deep Copy Using A Create Table LIKE
- 30. Deep Copy By Creating a Temp Table and Truncating Original

9. Sub-query Functions

- 1. An IN List is much like a Subquery
- 2. An IN List Never has Duplicates – Just like a Subquery
- 3. An IN List Ignores Duplicates
- 4. The Subquery
- 5. The Three Steps of How a Basic Subquery Works
- 6. These are Equivalent Queries

7. The Final Answer Set from the Subquery
 8. The Basics of a Correlated Subquery
 9. The Top Query always runs first in a Correlated Subquery
 10. Correlated Subquery Example vs. a Join with a Derived Table
 11. How to handle a NOT IN with Potential NULL Values
 12. Using a Correlated Exists
 13. How a Correlated Exists matches up
 14. The Correlated NOT Exists
 15. Substrings and Positioning Functions
 16. The TRIM Command trims both Leading and Trailing Spaces
 17. A Visual of the TRIM Command Using Concatenation
 18. Trim and Trailing is Case Sensitive
 19. How to TRIM Trailing Letters
 20. The SUBSTRING Command
 21. How SUBSTRING Works with NO ENDING POSITION
 22. Using SUBSTRING to move Backwards
 23. How SUBSTRING Works with a Starting Position of -
 24. How SUBSTRING Works with an Ending Position of
 25. The POSITION Command finds a Letters Position
 26. Using the SUBSTRING to Find the Second Word On
 27. Concatenation
 28. Concatenation and SUBSTRING
 29. Four Concatenations Together
 30. Troubleshooting Concatenation
 31. Declaring a Cursor
10. Interrogating the Data
 1. The NULLIFZERO Command
 2. The ZEROIFNULL Command
 3. The COALESCE Command
 4. The COALESCE Answer Set
 5. The Coalesce Quiz
 6. The Basics of CAST (Convert And STore)
 7. Some Great CAST (Convert And STore) Examples
 8. Some Great CAST (Convert And STore) Examples
 9. Some Great CAST (Convert And STore) Examples
 10. The Basics of the CASE Statements
 11. The Basics of the CASE Statement
 12. Valued Case Vs. A Searched Case
 13. Combining Searched Case and Valued Case
 14. Nested Case
 15. Put a CASE in the ORDER BY
11. View Functions
 1. Creating a Simple View to Restrict Sensitive Columns
 2. Creating a Simple View to Restrict Sensitive Columns
 3. Creating a Simple View to Restrict Rows
 4. Creating a View to Join Tables Together
 5. You Select From a View

6. Basic Rules for Views
7. An ORDER BY Example Inside of a View
8. An ORDER BY Inside of a View That is Queried Differently
9. Creating a View With Ordered Analytics
10. Creating a View With The TOP Command
11. Creating a View With The LIMIT Command
12. Altering A Table
13. Altering A Table After a View has been Created
14. A View that Errors After An ALTER
15. Troubleshooting a View
16. Updating Data in a Table through a View
12. Set Operators Functions
 1. Rules of Set Operators
 2. INTERSECT Explained Logically
 3. INTERSECT Explained Logically
 4. UNION Explained Logically
 5. UNION Explained Logically
 6. UNION ALL Explained Logically
 7. UNION Explained Logically
 8. EXCEPT Explained Logically
 9. EXCEPT Explained Logically
 10. Minus Explained Logically
 11. Minus Explained Logically
 12. Testing Your Knowledge
 13. Testing Your Knowledge
 14. An Equal amount of Columns in both SELECT List
 15. Columns in the SELECT list should be from the same Domain
 16. The Top Query handles all Aliases
 17. The Bottom Query does the ORDER BY (a Number)
 18. Great Trick: Place your Set Operator in a Derived Table
 19. UNION vs. UNION ALL
 20. A Great Example of how EXCEPT works
13. Statistical Aggregate Functions
 1. The Stats Table
 2. STDDEV
 3. Casting STDDEV_SAMP and SQRT (VAR_SAMP)
 4. The STDDEV_POP Function
 5. A STDDEV_POP Example
 6. The STDDEV_SAMP Function
 7. A STDDEV_SAMP Example
 8. The VAR_POP Function
 9. A VAR_POP Example
 10. The VAR_SAMP Function
 11. A VAR_SAMP Example

Each student will receive a comprehensive set of materials, including course notes and all the class examples.