

Amazon Redshift Architecture and SQL

Course Summary

Description

In this course, students will learn the Amazon Redshift Architecture and SQL starting at the most basic level and going to the most advanced level with many examples.

Objectives

By the end of this course, students will have a deeper knowledge and understanding of the Amazon Redshift Architecture and SQL and how to write it.

Topics

- 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

Audience

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

Prerequisites

There are no prerequisites for this course.

Duration

Two to three days

Amazon Redshift Architecture and SQL

Course Outline

- I. What is Columnar?**
 - A. What is Parallel Processing?
 - B. The Basics of a Single Computer
 - C. Data in Memory is Fast as Lightning
 - D. Parallel Processing Of Data
 - E. A Table has Columns and Rows
 - F. Each Parallel Process Organizes the Rows inside a Data Block
 - G. Moving Data Blocks is Like Checking In Luggage
 - H. Facts That Are Disturbing
 - I. Why Columnar?
 - J. Row Based Blocks vs. Columnar Based Blocks
 - K. As Row-Based Tables Get Bigger, the Blocks Split
 - L. Data Blocks Are Processed One at a Time Per Unit
 - M. Columnar Tables Store Each Column in Separate Blocks
 - N. Visualize the Data – Rows vs. Columns
 - O. The Architecture of Redshift
 - P. Redshift has Linear Scalability
 - Q. Distribution Styles
 - R. Distribution Key Where the Data is Unique
 - S. Another Way to Create A Table
 - T. Distribution Key Where the Data is Non-Unique
 - U. Distribution Key is ALL
 - V. Even Distribution Key
 - W. Matching Distribution Keys for Co-Location of Joins
 - X. Big Table / Small Table Joins
 - Y. Fact and Dimension Table Distribution Key Designs
 - Z. Improving Performance By Defining a Sort Key
 - AA. Sort Keys Help Group By, Order By and Window Functions
 - BB. Each Block Comes With Metadata
 - CC. How Data Might Look On A Slice
 - DD. Question – How Many Blocks Move Into Memory?
 - EE. Answer – How Many Blocks Move Into Memory?
 - FF. Quiz – Master that Query With the Metadata
 - GG. Answer to Quiz – Master that Query With the Metadata
- HH. The ANALYZE Command Collects Statistics**
- II. Redshift Automatically ANALYZES Some Create Statements**
- JJ. What is a Vacuum?**
- KK. When is a Good Time to Vacuum?**
- LL. The VACUUM Command Grooms a Table**
- MM. Database Limits**
- NN. Creating a Database**
- OO. Creating a User**
- PP. Dropping a User**
- QQ. Inserting Into a Table**
- RR. Renaming a Table or a Column**
- SS. Adding and Dropping a Column to a Table**
- II. Best Practices For Table Design**
 - A. Converting Table Structures to Redshift
 - B. Converting Table Structures to Redshift Finale
 - C. Best Practices for Designing Tables
 - D. Choose the Best Sort Key
 - E. Each Block Comes With Metadata
 - F. Creating a Sort Key
 - G. Sort Keys Help Group By, Order By and Window Functions
 - H. Choose a Great Distribution Key
 - I. Distribution Key Where the Data is Unique
 - J. Matching Distribution Keys for Co-Location of Joins
 - K. Big Table / Small Table Joins
 - L. Define Primary Key and Foreign Key Constraints
 - M. Primary Key and Foreign Key Examples
 - N. Use The Smallest Column Size When Creating Tables
 - O. Use Date/Time Data Types for Date Columns
 - P. Specify Redundant Predicates on the Sort Column
 - Q. Setting the Statement_Timeout to Abort Long Queries
- III. System Tables**
 - A. Amazon Redshift System Tables
 - B. Trouble Shooting Catalog Table pg_table_def

Amazon Redshift Architecture and SQL

Course Outline (cont'd)

- C. Seeing the System Tables in your Nexus Tree
 - D. Catalog Table pg_table_def
 - E. Checking Tables for Skew (Poor Distribution)
 - F. Checking All Statements That Used the Analyze Command
 - G. Checking Tables for Skew (Poor Distribution)
 - H. Checking For Details About the Last Copy Operation
 - I. Checking When a Table Has Last Been Analyzed
 - J. Checking For Column Information on a Table
 - K. System tables for troubleshooting data loads
 - L. Determining Whether a Query is Writing to Disk
- IV. Compression**
- A. Compression Types
 - B. Byte Dictionary Compression
 - C. Delta Encoding
 - D. LZO Encoding
 - E. Mostly Encoding
 - F. Runlength encoding
 - G. Text255 and Text32k Encodings
 - H. ANALYZE COMPRESSION
 - I. Copy
- V. Temporary Tables**
- A. Create Table Syntax
 - B. Basic Temporary Table Examples
 - C. More Advanced Temporary Table Examples
 - D. Advanced Temporary Table Examples
 - E. Table Limits and CTAS
 - F. Performing a Deep Copy
 - G. Deep Copy Using the Original DDL
 - H. Deep Copy Using A CTAS
 - I. Deep Copy Using A Create Table LIKE
 - J. Deep Copy By Creating a Temp Table and Truncating Original
 - K. CREATING A Derived Table
 - L. The Three Components of a Derived Table
 - M. Naming the Derived Table
- VI. Explain**
- A. Three Ways to Run an EXPLAIN
 - B. EXPLAIN – Steps, Segments and Streams
 - C. EXPLAIN Terms For Scans and Joins
 - D. EXPLAIN Terms For Aggregation and Sorts
 - E. EXPLAIN Terms For Set Operators and Miscellaneous Terms
 - F. EXPLAIN Example and the Cost
 - G. EXPLAIN Example and the Rows
 - H. EXPLAIN Example and the Width
 - I. Simple EXPLAIN Example and the Costs
 - J. EXPLAIN Join Example Using DS_BCAST_INNER
 - K. EXPLAIN Join Example Using DS_DIST_NONE
- N. Aliasing the Column Names in The Derived Table
 - O. Visualize This Derived Table
 - P. Most Derived Tables Are Used To Join To Other Tables
 - Q. Multiple Ways to Alias the Columns in a Derived Table
 - R. Our Join Example With A Different Column Aliasing Style
 - S. Column Aliasing Can Default For Normal Columns
 - T. CREATING A Derived Table using the WITH Command
 - U. Our Join Example With The WITH Syntax
 - V. WITH Statement That Uses a SELECT *
 - W. A WITH Clause That Produces Two Tables
 - X. The Same Derived Query shown Three Different Ways
 - Y. Quiz - Answer the Questions
 - Z. Answer to Quiz - Answer the Questions
 - AA. Clever Tricks on Aliasing Columns in a Derived Table
 - BB. A Derived Table lives only for the lifetime of a single query
 - CC. An Example of Two Derived Tables in a Single Query
 - DD. Connecting To Redshift Via Nexus

Amazon Redshift Architecture and SQL

Course Outline (cont'd)

- L. EXPLAIN Showing DS_DIST_NONE Visually
 - M. EXPLAIN With a Warning
 - N. EXPLAIN For Ordered Analytics Such as CSUM
 - O. EXPLAIN For Scalar Aggregate Functions
 - P. EXPLAIN For HashAggregate Functions
 - Q. EXPLAIN Using Limit, Merge and Sort
 - R. EXPLAIN Using a WHERE Clause Filter
 - S. EXPLAIN Using the Keyword Distinct
 - T. EXPLAIN for Subqueries
- VII. Basic SQL Functions**
- A. Finding the Current Schema on the Leader Node
 - B. Getting Things Setup in Your Search Path
 - C. Five Details You Need To Know About The Search_Path
 - D. Introduction
 - E. SELECT * (All Columns) in a Table
 - F. SELECT Specific Columns in a Table
 - G. Commas in the Front or Back?
 - H. Place your Commas in front for better Debugging Capabilities
 - I. Sort the Data with the ORDER BY Keyword
 - J. ORDER BY Defaults to Ascending
 - K. Use the Name or the Number in your ORDER BY Statement
 - L. Two Examples of ORDER BY using Different Techniques
 - M. Changing the ORDER BY to Descending Order
 - N. NULL Values sort First in Ascending Mode (Default)
 - O. NULL Values sort Last in Descending Mode (DESC)
 - P. Major Sort vs. Minor Sorts
 - Q. Multiple Sort Keys using Names vs. Numbers
 - R. Sorts are Alphabetical, NOT Logical
 - S. Using A CASE Statement to Sort Logically
 - T. How to ALIAS a Column Name
- U. A Missing Comma can by Mistake become an Alias
 - V. Comments using Double Dashes are Single Line Comments
 - W. Comments for Multi-Lines
 - X. Comments for Multi-Lines As Double Dashes Per Line
 - Y. A Great Technique for Comments to Look for SQL Errors
- VIII. The WHERE Clause**
- A. Using Limit to bring back a Sample
 - B. Using Limit With an Order By Statement
 - C. The WHERE Clause limits Returning Rows
 - D. Using a Column ALIAS throughout the SQL
 - E. Double Quoted Aliases are for Reserved Words and Spaces
 - F. Character Data needs Single Quotes in the WHERE Clause
 - G. Character Data needs Single Quotes, but Numbers Don't
 - H. NULL means UNKNOWN DATA so Equal (=) won't Work
 - I. Use IS NULL or IS NOT NULL when dealing with NULLs
 - J. NULL is UNKNOWN DATA so NOT Equal won't Work
 - K. Use IS NULL or IS NOT NULL when dealing with NULLs
 - L. Using Greater Than Or Equal To (>=)
 - M. AND in the WHERE Clause
 - N. Troubleshooting AND
 - O. OR in the WHERE Clause
 - P. Troubleshooting Or
 - Q. Troubleshooting Character Data
 - R. Using Different Columns in an AND Statement
 - S. Quiz – How many rows will return?
 - T. Answer to Quiz – How many rows will return?
 - U. What is the Order of Precedence?
 - V. Using Parentheses to change the Order of Precedence
 - W. Using an IN List in place of OR
 - X. The IN List is an Excellent Technique
 - Y. IN List vs. OR brings the same Results

Amazon Redshift Architecture and SQL

Course Outline (cont'd)

- Z. Using a NOT IN List
- AA. A Technique for Handling Nulls with a NOT IN List
- BB. Another Technique for Handling Nulls with a NOT IN List
- CC. BETWEEN is Inclusive
- DD. NOT BETWEEN is Also Inclusive
- EE. LIKE command Underscore is Wildcard for one Character
- FF. LIKE Command Works Differently on Char Vs Varchar
- GG. The Ilike Command Is NOT Case Sensitive
- HH. Troubleshooting LIKE Command on Character Data
- II. Introducing the TRIM Command
- JJ. Quiz – What Data is Left Justified and What is Right?
- KK. Numbers are Right Justified and Character Data is Left
- LL. Answer – What Data is Left Justified and What is Right?
- MM. An Example of Data with Left and Right Justification
- NN. A Visual of CHARACTER Data vs. VARCHAR Data
- OO. Use the TRIM command to remove spaces on CHAR Data
- PP. Like and Your Escape Character of Choice
- QQ. Like and the Default Escape Character
- RR. Similar To Operators
- SS. Similar To Example With Lower Case Letters
- TT. Similar To Example With Lower and Upper Case Letters
- UU. Similar To Example With Multiple Occurrences
- VV. Multiple Occurrences Must Be Consecutive
- IX. Distinct Vs Group By AND TOP**
 - A. The Distinct Command
 - B. Distinct vs. GROUP BY
 - C. Quiz – How many rows come back from the Distinct?
 - D. Answer – How many rows come back from the Distinct?
 - E. TOP Command
 - F. TOP Command is brilliant when ORDER BY is Used!
 - G. What is the Difference Between TOP and LIMIT?
- X. Aggregation**
 - A. Quiz – You calculate the Answer Set in your own Mind
 - B. Answer – You calculate the Answer Set in your own Mind
 - C. The 3 Rules of Aggregation
 - D. There are Five Aggregates
 - E. Quiz – How many rows come back?
 - F. Answer – How many rows come back?
 - G. Troubleshooting Aggregates
 - H. GROUP BY when Aggregates and Normal Columns Mix
 - I. GROUP BY Delivers one row per Group
 - J. GROUP BY Dept_No or GROUP BY 1 the same thing
 - K. Limiting Rows and Improving Performance with WHERE
 - L. WHERE Clause in Aggregation limits unneeded Calculations
 - M. Keyword HAVING tests Aggregates after they are Totaled
 - N. Keyword HAVING is like an Extra WHERE Clause for Totals
- XI. Join Functions**
 - A. A Two-Table Join Using Traditional Syntax
 - B. A two-table join using Non-ANSI Syntax with Table Alias
 - C. You Can Fully Qualify All Columns
 - D. A two-table join using ANSI Syntax
 - E. Both Queries have the same Results and Performance
 - F. Quiz – Can You Finish the Join Syntax?
 - G. Answer to Quiz – Can You Finish the Join Syntax?
 - H. Quiz – Can You Find the Error?
 - I. Answer to Quiz – Can You Find the Error?
 - J. Super Quiz – Can You Find the Difficult Error?
 - K. Answer to Super Quiz – Can You Find the Difficult Error?

Amazon Redshift Architecture and SQL

Course Outline (cont'd)

- L. Quiz – Which rows from both tables Won't Return?
 - M. Answer to Quiz – Which rows from both tables Won't Return?
 - N. LEFT OUTER JOIN
 - O. LEFT OUTER JOIN Results
 - P. Left Outer Joins Compatible with Oracle
 - Q. RIGHT OUTER JOIN
 - R. RIGHT OUTER JOIN Example and Results
 - S. Right Outer Joins Compatible with Oracle
 - T. FULL OUTER JOIN
 - U. FULL OUTER JOIN Results
 - V. Which Tables are the Left and Which are the Right?
 - W. Answer - Which Tables are the Left and Which are the Right?
 - X. INNER JOIN with Additional AND Clause
 - Y. ANSI INNER JOIN with Additional AND Clause
 - Z. ANSI INNER JOIN with Additional WHERE Clause
 - AA. OUTER JOIN with Additional WHERE Clause
 - BB. OUTER JOIN with Additional AND Clause
 - CC. OUTER JOIN with Additional AND Clause Results
 - DD. Quiz – Why is this Considered an INNER JOIN?
 - EE. The DREADED Product Join
 - FF. The DREADED Product Join Results
 - GG. The Horrifying Cartesian Product Join
 - HH. The ANSI Cartesian Join will ERROR
 - II. Quiz – Do these Joins Return the Same Answer Set?
 - JJ. Answer – Do these Joins Return the Same Answer Set?
 - KK. The CROSS JOIN
 - LL. The CROSS JOIN Answer Set
 - MM. The Self Join
 - NN. The Self Join with ANSI Syntax
 - OO. Quiz – Will both queries bring back the same Answer Set?
 - PP. Answer – Will both queries bring back the same Answer Set?
 - QQ. Quiz – Will both queries bring back the same Answer Set?
 - RR. Answer – Will both queries bring back the same Answer Set?
 - SS. How would you Join these two tables?
 - TT. An Associative Table is a Bridge that Joins Two Tables
 - UU. Quiz – Can you Write the 3-Table Join?
 - VV. Answer to Quiz – Can you Write the 3-Table Join?
 - WW. Quiz – Can you Write the 3-Table Join to ANSI Syntax?
 - XX. Answer – Can you Write the 3-Table Join to ANSI Syntax?
 - YY. Quiz – Can you Place the ON Clauses at the End?
 - ZZ. Answer – Can you Place the ON Clauses at the End?
 - AAA. The 5-Table Join – Logical Insurance Model
 - BBB. Quiz - Write a Five Table Join Using ANSI Syntax
 - CCC. Answer - Write a Five Table Join Using ANSI Syntax
 - DDD. Quiz - Write a Five Table Join Using Non-ANSI Syntax
 - EEE. Answer - Write a Five Table Join Using Non-ANSI Syntax
 - FFF. Quiz –Re-Write this putting the ON clauses at the END
 - GGG. Answer –Re-Write this putting the ON clauses at the END
- XII. Date Functions**
- A. Current_Date
 - B. TIMEOFDAY()
 - C. SYSDATE Returns a Timestamp With Microseconds
 - D. GETDATE Returns a Timestamp Without Microseconds
 - E. Add or Subtract Days from a date
 - F. The ADD_MONTHS Command Returns a Timestamp
 - G. The ADD_MONTHS Command With Trunc Removes Time
 - H. ADD_MONTHS Command to Add 1-Year or 5-Years

Amazon Redshift Architecture and SQL

Course Outline (cont'd)

- | | | | |
|-----|--|-----------------------------|--|
| I. | Dateadd Function And Add_Months Function are Different | TT. | Using Intervals |
| J. | The EXTRACT Command | UU. | Troubleshooting The Basics of a Simple Interval |
| K. | EXTRACT from DATES and TIME | VV. | Interval Arithmetic Results |
| L. | EXTRACT with DATE and TIME Literals | WW. | A Date Interval Example |
| M. | EXTRACT of the Month on Aggregate Queries | XX. | A Time Interval Example |
| N. | The Datediff command | YY. | A DATE Interval Example |
| O. | The Datediff Function on Column Data | ZZ. | A Complex Time Interval Example using CAST |
| P. | The Date_Part Function Using a Date | AAA. | A Complex Time Interval Example using CAST |
| Q. | The Date_Part Function Using a Time | BBB. | The OVERLAPS Command |
| R. | Date_Part Abbreviations | CCC. | An OVERLAPS Example that Returns No Rows |
| S. | The to_char command | DDD. | The OVERLAPS Command using TIME |
| T. | Conversion Functions | EEE. | The OVERLAPS Command using a NULL Value |
| U. | Conversion Function Templates | | |
| V. | Conversion Function Templates Continued | XIII. OLAP Functions | |
| W. | Formatting A Date | A. | CSUM |
| X. | A Summary of Math Operations on Dates | B. | CSUM – The Sort Explained |
| Y. | Using a Math Operation to find your Age in Years | C. | CSUM – Rows Unbounded Preceding Explained |
| Z. | Date Related Functions | D. | CSUM – Making Sense of the Data |
| AA. | A Side Title example with Reserved Words as an Alias | E. | CSUM – Making Even More Sense of the Data |
| BB. | Implied Extract of Day, Month and Year | F. | CSUM – The Major and Minor Sort Key(s) |
| CC. | DATE_PART Function | G. | Reset with a PARTITION BY Statement |
| DD. | DATE_PART Function using an ALIAS | H. | PARTITION BY only Resets a Single OLAP not ALL of them |
| EE. | DATE_TRUNC Function | I. | ANSI Moving Window is Current Row and Preceding n Rows |
| FF. | DATE_TRUNC Function using TIME | J. | How ANSI Moving SUM Handles the Sort |
| GG. | MONTHS_BETWEEN Function | K. | Quiz – How is that Total Calculated? |
| HH. | MONTHS_BETWEEN Function in Action | L. | Answer to Quiz – How is that Total Calculated? |
| II. | ANSI TIME | M. | Moving SUM every 3-rows Vs a Continuous Average |
| JJ. | ANSI TIMESTAMP | N. | Partition By Resets an ANSI OLAP |
| KK. | Redshift TIMESTAMP Function | O. | Moving Average |
| LL. | Redshift TO_TIMESTAMP Function | P. | The Moving Window is Current Row and Preceding |
| MM. | Redshift NOW() Function | Q. | How Moving Average Handles the Sort |
| NN. | Redshift TIMEOFDAY Function | R. | Quiz – How is that Total Calculated? |
| OO. | Redshift AGE Function | | |
| PP. | Time Zones | | |
| QQ. | Setting Time Zones | | |
| RR. | Using Time Zones | | |
| SS. | Intervals for Date, Time and Timestamp | | |

Amazon Redshift Architecture and SQL

Course Outline (cont'd)

- S. Answer to Quiz – How is that Total Calculated?
 - T. Quiz – How is that 4th Row Calculated?
 - U. Answer to Quiz – How is that 4th Row Calculated?
 - V. Moving Average every 3-rows Vs a Continuous Average
 - W. Partition By Resets an ANSI OLAP
 - X. RANK Defaults to Ascending Order
 - Y. RANK Defaults to Ascending Order
 - Z. Getting RANK to Sort in DESC Order
 - AA. RANK() OVER and PARTITION BY
 - BB. RANK() OVER And LIMIT
 - CC. PERCENT_RANK() OVER
 - DD. PERCENT_RANK() OVER with 14 rows in Calculation
 - EE. PERCENT_RANK() OVER with 21 rows in Calculation
 - FF. Quiz – What Causes the Product_ID to Reset?
 - GG. Answer to Quiz – What Cause the Product_ID to Reset?
 - HH. COUNT OVER for a Sequential Number
 - II. Quiz – What caused the COUNT OVER to Reset?
 - JJ. Answer to Quiz – What caused the COUNT OVER to Reset?
 - KK. The MAX OVER Command
 - LL. MAX OVER with PARTITION BY Reset
 - MM. The MIN OVER Command
 - NN. Quiz – Fill in the Blank
 - OO. Answer – Fill in the Blank
 - PP. The Row_Number Command
 - QQ. Quiz – How did the Row_Number Reset?
 - RR. Quiz – How did the Row_Number Reset?
 - SS. Standard Deviation Functions Using STDDEV / OVER
 - TT. Standard Deviation Functions and STDDEV / OVER Syntax
 - UU. STDDEV / OVER Example
 - VV. VARIANCE / OVER Syntax
 - WW. Variance Functions Using VARIANCE / OVER
 - XX. Using VARIANCE with PARTITION BY Example
 - YY. Using FIRST_VALUE and LAST_VALUE
 - ZZ. Using FIRST_VALUE
 - AAA. Using LAST_VALUE
 - BBB. Using LAG and LEAD
 - CCC. Using LEAD
 - DDD. Using LEAD With and Offset of 2
 - EEE. Using LAG
 - FFF. Using LAG With an Offset of 2
- XIV. Temporary Tables**
- A. CREATING A Derived Table
 - B. The Three Components of a Derived Table
 - C. Naming the Derived Table
 - D. Aliasing the Column Names in The Derived Table
 - E. Visualize This Derived Table
 - F. Most Derived Tables Are Used To Join To Other Tables
 - G. Multiple Ways to Alias the Columns in a Derived Table
 - H. Our Join Example With A Different Column Aliasing Style
 - I. Column Aliasing Can Default For Normal Columns
 - J. CREATING A Derived Table using the WITH Command
 - K. Our Join Example With The WITH Syntax
 - L. WITH
 - M. A WITH Clause That Produces Two Tables
 - N. The Same Derived Query shown Three Different Ways
 - O. Quiz - Answer the Questions
 - P. Answer to Quiz - Answer the Questions
 - Q. Clever Tricks on Aliasing Columns in a Derived Table
 - R. A Derived Table lives only for the lifetime of a single query
 - S. An Example of Two Derived Tables in a Single Query
 - T. Create Table Syntax
 - U. Basic Temporary Table Examples
 - V. More Advanced Temporary Table Examples
 - W. Advanced Temporary Table Examples
 - X. Performing a Deep Copy

Amazon Redshift Architecture and SQL**Course Outline (cont'd)**

- Y. Deep Copy Using the Original DDL
 - Z. Deep Copy Using A CTAS
 - AA. Deep Copy Using A Create Table LIKE
 - BB. Deep Copy By Creating a Temp Table and Truncating Original
- XV. Sub-query Functions**
- A. An IN List is much like a Subquery
 - B. An IN List Never has Duplicates – Just like a Subquery
 - C. An IN List Ignores Duplicates
 - D. The Subquery
 - E. The Three Steps of How a Basic Subquery Works
 - F. These are Equivalent Queries
 - G. The Final Answer Set from the Subquery
 - H. Quiz- Answer the Difficult Question
 - I. Answer to Quiz- Answer the Difficult Question
 - J. Should you use a Subquery of a Join?
 - K. Quiz- Write the Subquery
 - L. Answer to Quiz- Write the Subquery
 - M. Quiz- Write the More Difficult Subquery
 - N. Answer to Quiz- Write the More Difficult Subquery
 - O. Quiz- Write the Subquery with an Aggregate
 - P. Answer to Quiz- Write the Subquery with an Aggregate
 - Q. Quiz- Write the Correlated Subquery
 - R. Answer to Quiz- Write the Correlated Subquery
 - S. The Basics of a Correlated Subquery
 - T. The Top Query always runs first in a Correlated Subquery
 - U. Correlated Subquery Example vs. a Join with a Derived Table
 - V. Quiz- A Second Chance To Write a Correlated Subquery
 - W. Answer - A Second Chance to Write a Correlated Subquery
 - X. Quiz- A Third Chance To Write a Correlated Subquery
 - Y. Answer - A Third Chance to Write a Correlated Subquery
 - Z. Quiz- Last Chance To Write a Correlated Subquery
 - AA. Answer – Last Chance to Write a Correlated Subquery
 - BB. Quiz- Write the NOT Subquery
 - CC. Answer to Quiz- Write the NOT Subquery
 - DD. Quiz- Write the Subquery using a WHERE Clause
 - EE. Answer - Write the Subquery using a WHERE Clause
 - FF. Quiz- Write the Subquery with Two Parameters
 - GG. Answer to Quiz- Write the Subquery with Two Parameters
 - HH. How the Double Parameter Subquery Works
 - II. More on how the Double Parameter Subquery Works
 - JJ. Quiz – Write the Triple Subquery
 - KK. Answer to Quiz – Write the Triple Subquery
 - LL. Quiz – How many rows return on a NOT IN with a NULL?
 - MM. Answer – How many rows return on a NOT IN with a NULL?
 - NN. How to handle a NOT IN with Potential NULL Values
 - OO. Using a Correlated Exists
 - PP. How a Correlated Exists matches up
 - QQ. The Correlated NOT Exists
 - RR. The Correlated NOT Exists Answer Set
 - SS. Quiz – How many rows come back from this NOT Exists?
 - TT. Answer – How many rows come back from this NOT Exists?
- XVI. Substrings and Positioning Functions**
- A. The TRIM Command trims both Leading and Trailing Spaces
 - B. A Visual of the TRIM Command Using Concatenation
 - C. Trim and Trailing is Case Sensitive
 - D. How to TRIM Trailing Letters
 - E. The SUBSTRING Command
 - F. How SUBSTRING Works with NO ENDING POSITION
 - G. Using SUBSTRING to move Backwards

Amazon Redshift Architecture and SQL

Course Outline (cont'd)

- H. How SUBSTRING Works with a Starting Position of -1
 - I. How SUBSTRING Works with an Ending Position of 0
 - J. The POSITION Command finds a Letters Position
 - K. Quiz – Find that SUBSTRING Starting Position
 - L. Answer to Quiz – Find that SUBSTRING Starting Position
 - M. Using the SUBSTRING to Find the Second Word On
 - N. Quiz – Why Did only one Row Return
 - O. Answer to Quiz – Why Did only one Row Return
 - P. Concatenation
 - Q. Concatenation and SUBSTRING
 - R. Four Concatenations Together
 - S. Troubleshooting Concatenation
 - T. Declaring a Cursor
- XVII. Interrogating the Data**
- A. Quiz – What would the Answer be?
 - B. Answer to Quiz – What would the Answer be?
 - C. The NULLIFZERO Command
 - D. Quiz – Fill in the Blank Values in the Answer Set
 - E. Answer to Quiz – Fill in the Blank Values in the Answer Set
 - F. Quiz – Fill in the Answers for the NULLIF Command
 - G. Quiz – Fill in the Answers for the NULLIF Command
 - H. The ZEROIFNULL Command
 - I. Answer to the ZEROIFNULL Question
 - J. The COALESCE Command
 - K. The COALESCE Answer Set
 - L. The Coalesce Quiz
 - M. Answer – The Coalesce Quiz
 - N. The Basics of CAST (Convert And Store)
 - O. Some Great CAST (Convert And Store) Examples
 - P. The Basics of the CASE Statements
 - Q. The Basics of the CASE Statement
 - R. Valued Case Vs. A Searched Case
 - S. Quiz - Valued Case Statement
- XVIII. View Functions**
- A. Creating a Simple View to Restrict Sensitive Columns
 - B. Creating a Simple View to Restrict Rows
 - C. Creating a View to Join Tables Together
 - D. You Select From a View
 - E. Basic Rules for Views
 - F. An ORDER BY Example Inside of a View
 - G. An ORDER BY Inside of a View That is Queried Differently
 - H. Creating a View With Ordered Analytics
 - I. Creating a View With The TOP Command
 - J. Creating a View With The LIMIT Command
 - K. Altering A Table
 - L. Altering A Table After a View has been Created
 - M. A View that Errors After An ALTER
 - N. Troubleshooting a View
 - O. Updating Data in a Table through a View
- XIX. Set Operators Functions**
- A. Rules of Set Operators
 - B. INTERSECT Explained Logically
 - C. UNION Explained Logically
- T. Answer - Valued Case Statement
 - U. Quiz - Searched Case Statement
 - V. Answer - Searched Case Statement
 - W. Quiz - When NO ELSE is present in CASE Statement
 - X. Answer - When NO ELSE is present in CASE Statement
 - Y. When an ELSE is present in CASE Statement
 - Z. Answer - When an ELSE is present in CASE Statement
 - AA. When an Alias is NOT used in a CASE Statement
 - BB. Answer - When an Alias is NOT used in a CASE Statement
 - CC. Combining Searched Case and Valued Case
 - DD. Nested Case
 - EE. Put a CASE in the ORDER BY

Amazon Redshift Architecture and SQL

Course Outline (cont'd)

- D. UNION ALL Explained Logically
 - E. EXCEPT Explained Logically
 - F. Minus Explained Logically
 - G. Testing Your Knowledge
 - H. Answer - Testing Your Knowledge
 - I. An Equal Amount of Columns in both SELECT List
 - J. Columns in the SELECT list should be from the same Domain
 - K. The Top Query handles all Aliases
 - L. The Bottom Query does the ORDER BY (a Number)
 - M. Great Trick: Place your Set Operator in a Derived Table
 - N. UNION vs. UNION ALL
 - O. A Great Example of how EXCEPT works
- XX. Statistical Aggregate Functions**
 - A. The Stats Table
 - B. STDDEV
 - C. Casting STDDEV_SAMP and SQRT (VAR_SAMP)
 - D. The STDDEV_POP Function
 - E. A STDDEV_POP Example
 - F. The STDDEV_SAMP Function
 - G. A STDDEV_SAMP Example
 - H. The VAR_POP Function
 - I. A VAR_POP Example
 - J. The VAR_SAMP Function
 - K. A VAR_SAMP Example