



# Teradata SQL v15

Course ID#: 1190-110-15-W

Hours: 35

## Course Content

### Course Description:

In this course, students will learn the Teradata SQL v15 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 Teradata SQL v15 and how to write it.

In this course you will cover the following:

- 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 Teradata SQL v15 from beginners to an advanced audience. This course is completely customizable by the client.

### Prerequisites:

None



## Topics:

### Module 1: Basic SQL Functions

- Introduction
- SELECT \* (All Columns) in a Table
- SELECT Specific Columns in a Table
- Using the Best Form for Writing SQL
- Commas in the Front or in the Back?
- Place your Commas in front for better Debugging Capabilities
- Sort the Data with the ORDER BY Keyword
- ORDER BY Defaults to Ascending
- Use the Name or the Number in your ORDER BY Statement
- Two Examples of ORDER BY using Different Techniques
- Changing the ORDER BY to Descending Order
- NULL Values sort First in Ascending Mode (Default)
- NULL Values sort Last in Descending Mode (DESC)
- Major Sort vs. Minor Sorts
- Multiple Sort Keys using Names vs. Numbers
- Sorts are Alphabetical, NOT Logical
- Using A CASE Statement to Sort Logically
- How to ALIAS a Column Name
- A Missing Comma can by Mistake become an Alias
- The Title Command and Literal Data
- 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 2: The WHERE Clause

- The WHERE Clause limits Returning Rows
- Using a Column ALIAS throughout the SQL
- Double Quoted Aliases are for Reserved Words and Spaces
- Character Data needs Single Quotes in the WHERE Clause
- Character Data needs Single Quotes, but Numbers Don't
- NULL means UNKNOWN DATA so Equal (=) won't Work
- Use IS NULL or IS NOT NULL when dealing with NULLs
- NULL is UNKNOWN DATA so NOT Equal won't Work
- Use IS NULL or IS NOT NULL when dealing with NULLs
- Using Greater Than OR Equal To (>=)
- Using GE as Greater Than or Equal To (>=)
- AND in the WHERE Clause
- Troubleshooting AND
- OR in the WHERE Clause
- Troubleshooting OR
- OR must utilize the Column Name Each Time
- 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
- Using an IN List in place of OR
- The IN List is an Excellent Technique
- IN List vs. OR brings the same Results
- Using a NOT IN List
- A Technique for Handling Nulls with a NOT IN List



# Teradata SQL v15

Course ID#: 1190-110-15-W

Hours: 35

## Module 2 Continued: The WHERE Clause

- An IN List with the Keyword ANY
- A NOT IN List with the Keywords NOT = ALL
- BETWEEN is Inclusive
- BETWEEN Works for Character Data
- LIKE uses Wildcards Percent '%' and Underscore '\_'
- LIKE command Underscore is Wildcard for one Character
- LIKE ALL means ALL conditions must be Met
- LIKE ANY means ANY of the Conditions can be Met
- IN ANSI Transaction Mode Case Matters
- In Teradata Transaction Mode Case Doesn't Matter
- LIKE Command Works Differently on Char Vs. Varchar
- Troubleshooting LIKE Command on Character Data
- Introducing the TRIM Command
- Quiz – Which Data is Left Justified and Which is Right?
- Numbers are Right Justified and Character Data is Left
- Answer – Which Data is Left Justified and Which 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
- TRIM Eliminates Leading and Trailing Spaces
- Escape Character in the LIKE Command changes Wildcards
- Escape Characters Turn off Wildcards in the LIKE Command
- Quiz – Turn off that Wildcard
- ANSWER – To Find that Wildcard

## Module 3: Distinct Vs. Group By

- The Distinct Command
- Distinct vs. GROUP BY
- Rules of Thumb for DISTINCT vs. GROUP BY
- GROUP BY Vs. DISTINCT – Good Advice
- Quiz – How many rows come back from the Distinct?
- Answer – How many rows come back from the Distinct?

## Module 4: The TOP Command

- TOP Command
- TOP Command is brilliant when ORDER BY is used!
- The TOP Command WITH TIES
- How the TOP Command WITH TIES Decides
- The TOP Command will NOT work with Certain Commands

## Module 5: Review

- Testing Your Knowledge 1
- Testing Your Knowledge 2
- Testing Your Knowledge 3
- Testing Your Knowledge 4
- Testing Your Knowledge 5
- Testing Your Knowledge 6
- Testing Your Knowledge 7

## Module 6: HELP and SHOW

- Determining the Release of your Teradata System
- Basic HELP Commands
- Other HELP Commands
- HELP DATABASE
- HELP USER
- HELP TABLE
- Adding a Comment to a Table
- Adding a Comment to a View
- SELECT SESSION
- USER Information Functions
- HELP SESSION
- HELP SQL



# Teradata SQL v15

Course ID#: 1190-110-15-W

Hours: 35

- A HELP SQL Example
- Show Commands
- SHOW Table command for Table DDL
- SHOW View command for View Create Statement
- SHOW Macro command for Macro Create Statement
- SHOW Trigger command for Trigger Create Statement

## Module 7: Aggregation Function

- Quiz – You calculate the Answer Set in your own Mind
- Answer – You calculate the Answer Set in your own Mind
- The 3 Rules of Aggregation
- There are Five Aggregates
- Quiz – How many rows come back?
- Troubleshooting Aggregates
- GROUP BY when Aggregates and Normal Columns Mix
- GROUP BY Delivers one row per Group
- GROUP BY Dept\_No or GROUP BY 1 the same thing
- Limiting Rows and Improving Performance with WHERE
- WHERE Clause in Aggregation limits unneeded Calculations
- Keyword HAVING tests Aggregates after they are Totaled
- Keyword HAVING is like an Extra WHERE Clause for Totals
- Getting the Average Values per Column
- Average Values per Column for All Columns in a Table
- Three types of Advanced Grouping
- GROUP BY Grouping Sets
- GROUP BY Rollup
- GROUP BY Rollup Result Set
- GROUP BY Cube
- GROUP BY CUBE Result Set

- Use the Nexus for all Groupings
- Testing Your Knowledge – Basic Aggregation
- Testing Your Knowledge – Multiple Aggregates
- Testing Your Knowledge- Group By
- Testing Your Knowledge – Using a Where Clause
- Testing Your Knowledge- Using Having
- Final Answer to Test Your Knowledge on Aggregates

## Module 8: Join Functions

- A two-table join using Non-ANSI Syntax
- A two-table join using Non-ANSI Syntax with Table Alias
- Aliases and Fully Qualifying Columns
- A two-table join using ANSI Syntax
- Both Queries have the same Results and Performance
- Quiz – Can You Finish the Join Syntax?
- Answer to Quiz – Can You Finish the Join Syntax?
- Quiz – Can You Find the Error?
- Answer to Quiz – Can You Find the 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 Brings Back All Rows in the Left Table
- RIGHT OUTER JOIN
- RIGHT OUTER JOIN Brings Back All Rows in the RIGHT Table
- FULL OUTER JOIN
- FULL OUTER JOIN Brings Back All Rows in All Tables
- Which Tables are the Left and which are the Right?
- Answer - Which Tables are the Left and which are the Right?
- INNER JOIN with Additional AND Clause



# Teradata SQL v15

Course ID#: 1190-110-15-W

Hours: 35

## Module 8 Continued: Join Functions

- ANSI INNER JOIN with Additional AND Clause
- ANSI INNER JOIN with Additional WHERE Clause
- OUTER JOIN with Additional WHERE Clause
- OUTER JOIN with Additional AND Clause
- Results from OUTER JOIN with Additional AND Clause
- Quiz – Why is this considered an INNER JOIN?
- The DREADED Product Join
- Result Set of the DREADED Product Join
- The Horrifying Cartesian Product Join
- The ANSI Cartesian Join will ERROR
- Quiz – Do these Joins Return the Same Answer Set?
- Answer – Do these Joins Return the Same Answer Set?
- The CROSS JOIN
- The CROSS JOIN Answer Set
- The Self Join
- The Self Join with ANSI Syntax
- Quiz – Will both queries bring back the same Answer Set?
- Answer – Will both queries bring back the same Answer Set?
- Quiz – Will both queries bring back the same Answer Set?
- Answer – Will both queries bring back the same Answer Set?
- How would you Join these two tables?
- How would you Join these two tables? You Can't Yet!
- An Associative Table is a Bridge that Joins Two Tables
- Quiz – Can you Write the 3-Table Join?
- Answer to Quiz – Can you Write the 3-Table Join?
- Quiz – Can you Write the 3-Table Join to ANSI Syntax?
- Answer – Can you Write the 3-Table Join to ANSI Syntax?

- 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 - Write a Five Table Join Using ANSI Syntax
- Answer - Write a Five Table Join Using ANSI Syntax
- Quiz - Write a Five Table Join Using Non-ANSI Syntax
- Answer - Write a Five Table Join Using Non-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 9: Date Functions

- Date, Time, and Current\_Timestamp Keywords
- Dates are stored internally as INTEGERS from a Formula
- Displaying Dates for INTEGERDATE and ANSIDATE
- DATEFORM
- Changing the DATEFORM in Client Utilities such as BTEQ
- Date, Time, and Timestamp Recap
- Timestamp Differences
- Finding the Number of Hours between Timestamps
- Troubleshooting Timestamp
- Add or Subtract Days from a date
- A Summary of Math Operations on Dates
- Using a Math Operation to find your Age in Years
- Find What Day of the week you were Born



## Module 9 Continued: Date Functions

- The ADD\_MONTHS Command
- Using the ADD\_MONTHS Command to Add 1 Year
- Using the ADD\_MONTHS Command to Add 5 Years
- The EXTRACT Command
- EXTRACT from DATES and TIME
- CURRENT\_DATE and EXTRACT or Current\_Date and Math
- CAST the Date of January 1, 2011 and the Year 1800
- The System Calendar
- Using the System Calendar in Its Simplest Form
- How to really use the Sys\_Calendar.Calendar
- Storing Dates Internally
- Storing Time Internally
- Storing TIME with TIME\_ZONE Internally
- Storing Timestamp Internally
- Storing Timestamp with TIME\_ZONE Internally
- Storing Date, Time, and Timestamp with Zone Internally
- Time Zones
- Setting Time Zones
- Seeing your Time Zone
- Creating a Sample Table for Time Zone Examples
- Inserting Rows in the Sample Table for Time Zone Examples
- Selecting the Data from our Time Zone Table
- Normalizing our Time Zone Table with a CAST
- Intervals for Date, Time and Timestamp
- Interval Data Types and the Bytes to Store Them
- The Basics of a Simple Interval
- Troubleshooting the Basics of a Simple Interval
- Interval Arithmetic Results
- A Date Interval Example
- A Time Interval Example
- A - DATE Interval Example
- A Complex Time Interval Example using CAST
- A Complex Time Interval Example using CAST
- The OVERLAPS Command

- An OVERLAPS Example that Returns No Rows
- The OVERLAPS Command using TIME
- The OVERLAPS Command using a NULL Value

## Module 10: Format Functions

- The FORMAT Command
- The Basics of the FORMAT Command
- Quiz – How will the Date Appear after Formatting
- Answer to Quiz – How will the Date Appear after Formatting
- Quiz – How will the Date Appear after Formatting
- Answer to Quiz – How will the Date Appear after Formatting
- Formatting with MMM for the Abbreviated Month
- Answer to Quiz – How will the Date Appear after Formatting
- Formatting with MMMM for the Full Month Name
- Formatting with MMMM for the Full Month
- Formatting with DDD for the Julian Day
- Formatting with DDD for the Julian Day
- Formatting with EEE or EEEE for the Day of the Week
- EEEE for the Abbreviated or Full Day of the Week
- Placing Spaces inside your Formatting Commands with a B
- Formatting Spaces with B or b
- Formatting with 9
- Formatting with 9 Results
- Troubleshooting when Formatted Data Overflows
- Troubleshooting when Formatted Data Overflows
- Formatting with X or x
- Formatting with Z
- Formatting with Z Visual



# Teradata SQL v15

Course ID#: 1190-110-15-W

Hours: 35

## Module 10 Continued: Format Functions

- Formatting with 9
- Formatting with 9 Visual
- Formatting with \$
- Formatting with \$ Visual
- Formatting with \$ and Commas
- Formatting with \$ and Commas Visual
- Formatting with \$ and Commas and 9
- Formatting with \$ and Commas and 9 with Zero Dollars
- A Great Formatting Example
- A Great Formatting Example for Day, Month, and Year
- A Trick to get SQL Assistant to Format Data
- Using the CASESPECIFIC (CS) Command in Teradata Mode
- Using NOT CASESPECIFIC (CS) in ANSI Mode
- Using the LOWER Command
- Using the UPPER Command

## Module 11: OLAP Functions

- On-Line Analytical Processing (OLAP) or Ordered Analytics
- Cumulative Sum (CSUM) Command and how OLAP Works
- OLAP Commands always Sort (ORDER BY) in the Command
- Calculate the Cumulative Sum (CSUM) after Sorting the Data
- The OLAP Major Sort Key
- The OLAP Major Sort Key and the Minor Sort Key(s)
- Troubleshooting OLAP – My Data isn't coming back correct
- GROUP BY in Teradata OLAP Syntax Resets on the Group
- CSUM the Number 1 to get a Sequential Number
- A Single GROUP BY Resets each OLAP with Teradata Syntax
- A Better Choice – The ANSI Version of CSUM

## Module 11 Continued: OLAP Functions

- The ANSI Version of CSUM – The Sort Explained
- The ANSI CSUM – Rows Unbounded Preceding Explained
- The ANSI CSUM – Making Sense of the Data
- The ANSI CSUM – Making Even More Sense of the Data
- The ANSI CSUM – The Major and Minor Sort Key(s)
- The ANSI CSUM – Getting a Sequential Number
- Troubleshooting the ANSI OLAP on a GROUP BY
- The ANSI OLAP – Reset with a PARTITION BY Statement
- PARTITION BY only Resets a Single OLAP not ALL of them
- The Moving SUM (MSUM) and Moving Window
- How the Moving Sum is calculated
- How the Sort works for Moving SUM (MSUM)
- GROUP BY in the Moving SUM does a Reset
- Quiz – Can you make the Advanced Calculation in your mind?
- Answer to Quiz for the Advanced Calculation in your mind?
- Quiz – Write that Teradata Moving Average in ANSI Syntax
- Both the Teradata Moving SUM and ANSI Version
- The ANSI Moving Window is Current Row and Preceding
- How ANSI Moving Average 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 Average (MAVG) and Moving Window
- How the Moving Average is calculated



## Module 11 Continued: OLAP Functions

- How the Sort works for Moving Average (MAVG)
- GROUP BY in the Moving Average does a Reset
- Quiz – Can you make the Advanced Calculation in your mind?
- Answer to Quiz for the Advanced Calculation in your mind?
- Quiz – Write that Teradata Moving Average in ANSI Syntax
- Both the Teradata Moving Average and ANSI Version
- The ANSI Moving Window is Current Row and Preceding
- How ANSI Moving Average Handles the Sort
- Quiz – How is that Total Calculated?
- Answer to Quiz – How is that Total Calculated?
- Quiz – How is that 4th Row Calculated?
- Answer to Quiz – How is that 4th Row Calculated?
- Moving Average every 3-rows Vs. a Continuous Average
- Partition BY Resets an ANSI OLAP
- The Moving Difference (MDIFF)
- Moving Difference (MDIFF) Visual
- Moving Difference using ANSI Syntax
- Moving Difference using ANSI Syntax with Partition By
- Trouble Shooting the Moving Difference (MDIFF)
- Using the RESET WHEN Option in Teradata (V13)
- How Many Months per Product\_ID has Revenue Increased?
- The RANK Command
- How to get Rank to Sort in Ascending Order
- Two ways to get Rank to Sort in Ascending Order
- RANK using ANSI Syntax Defaults to Ascending Order
- Getting RANK using ANSI Syntax to Sort in DESC Order

## Module 11 Continued: OLAP Functions

- RANK () OVER and PARTITION BY
- RANK () OVER and QUALIFY
- RANK () OVER and PARTITION BY with a QUALIFY
- QUALIFY and WHERE
- Quiz – How can you simplify the QUALIFY Statement
- Answer to Quiz –Can you simplify the QUALIFY Statement
- The QUALIFY Statement without Ties
- The QUALIFY Statement with Ties
- The QUALIFY Statement with Ties Brings back Extra Rows
- Mixing Sort Order for QUALIFY Statement
- Quiz – What Caused the RANK to Reset?
- Answer to Quiz – What Caused the RANK to Reset?
- Quiz – Name those Sort Orders
- Answer to Quiz – Name those Sort Orders
- PERCENT\_RANK () OVER
- PERCENT\_RANK () OVER with 14 rows in Calculation
- PERCENT\_RANK () OVER with 21 rows in Calculation
- Quiz – What Cause the Product\_ID to Reset
- Answer to Quiz – What Causes the Product\_ID to Reset
- Answer to Quiz – What Causes the Product\_ID to Reset
- COUNT OVER for a Sequential Number
- Troubleshooting COUNT OVER
- 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
- Troubleshooting MAX OVER
- The MIN OVER Command
- Troubleshooting MIN OVER





# Teradata SQL v15

Course ID#: 1190-110-15-W

Hours: 35

## Module 11 Continued: OLAP Functions

- Finding a Value of a Column in the Next Row with MIN
- Finding a Value of a Date in the Next Row with MIN
- Finding Gaps between Dates
- The CSUM for Each Product\_ID for the First 3 Days
- Quiz – Fill in the Blank
- Answer to Quiz – Fill in the Blank
- The Row\_Number Command
- Quiz – How did the Row\_Number Reset?
- Quiz – How did the Row\_Number Reset?
- Row\_Number with Qualify to get the Typical Rows per Value
- A Second Typical Rows per Value Query on Sale\_Date
- Testing Your Knowledge
- Testing Your Knowledge
- Testing Your Knowledge
- Testing Your Knowledge

## Module 12: The Quantile Function

- The Quantile Function and Syntax
- A Quantile Example
- A Quantile Example using DESC Mode
- QUALIFY to find Products in the top Partitions
- QUALIFY to find Products in the top Partitions Sorted DESC
- QUALIFY to find Products in the top Partitions Sorted ASC
- QUALIFY to find Products in top Partitions with Tiebreaker
- Using Tertiles (Partitions of Four)
- How Quantile Works

## Module 13: Temporary Tables

- There are three types of Temporary Tables
- CREATING A Derived Table
- Naming the Derived Table

## Module 13 Continued: Temporary Tables

- Aliasing the Column Names in the Derived Table
- Most Derived Tables Are Used To Join To Other Tables
- Multiple Ways to Alias the Columns in a Derived Table
- Our Join Example with a Different Column Aliasing Style
- Column Aliasing Can Default for Normal Columns
- CREATING a Derived Table using the WITH Command
- Our Join Example With the WITH Syntax
- The Same Derived Query shown Three Different Ways
- Quiz - Answer the Questions
- Answer to Quiz - Answer the Questions
- Clever Tricks on Aliasing Columns in a Derived Table
- A Derived Table lives only for the lifetime of a single query
- An Example of Two Derived Tables in a Single Query
- WITH RECURSIVE Derived Table
- Defining the WITH Recursive Derived Table
- Looping Through the WITH Recursive Derived Table
- Looping Through the WITH Recursive Derived Table
- Looping Through the WITH Recursive Derived Table
- Looping Through the WITH Recursive Derived Table
- Looping Through the WITH Recursive Derived Table
- Creating a Volatile Table
- You Populate a Volatile Table with an INSERT/SELECT
- The Three Steps to Use a Volatile Table
- Why Would You Use the ON COMMIT DELETE ROWS?



## Module 13 Continued: Temporary Tables

- The HELP Volatile Table Command Shows your Volatiles
- A Volatile Table with a Primary Index
- The Joining of Two Tables Using a Volatile Table
- You Can Collect Statistics on Volatile Tables
- The New Teradata V14 Way to Collect Statistics
- Four Examples of Creating a Volatile Table Quickly
- Four Advanced Examples of Creating a Volatile Table Quickly
- Creating Partitioned Primary Index (PPI) Volatile Tables
- Using a Volatile Table to Get Rid of Duplicate Rows
- Using a Simple Global Temporary Table
- Two Brilliant Techniques for Global Temporary Tables
- The Joining of Two Tables Using a Global Temporary Table
- CREATING A Global Temporary Table

## Module 14: Sub-query Functions

- An IN List is much like a Subquery
- An IN List Never has Duplicates – Just like a Subquery
- An IN List Ignores Duplicates
- The Subquery
- How a Basic Subquery Works
- The Final Answer Set from the Subquery
- Quiz- Answer the Difficult Question
- Answer to Quiz- Answer the Difficult Question
- Should you use a Subquery of 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

## Module 14 Continued: Sub-query Functions

- Answer to Quiz- Write the Subquery with an Aggregate
- Quiz- Write the Correlated Subquery
- Answer to Quiz- Write the Correlated Subquery
- The Basics of a Correlated Subquery
- The Top Query always runs first in a Correlated Subquery
- The Bottom Query runs last in a Correlated Subquery
- Quiz- Who is coming back in the Final Answer Set?
- Answer- Who is coming back in the Final Answer Set?
- Correlated Subquery Example vs. a Join with a Derived Table
- Quiz- A Second Chance to Write a Correlated Subquery
- Answer - A Second Chance to Write a Correlated Subquery
- Quiz- A Third Chance to Write a Correlated Subquery
- Answer - A Third Chance to Write a Correlated Subquery
- Quiz- Last Chance to Write a Correlated Subquery
- Answer – Last Chance to Write a Correlated Subquery
- Correlated Subquery that Finds Duplicates
- 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- Write the Subquery with Two Parameters
- Answer to Quiz- Write the Subquery with Two Parameters
- How the Double Parameter Subquery Works



## Module 14 Continued: Sub-query Functions

- More on how the Double Parameter Subquery Works
- Quiz – Write the Triple Subquery
- Answer to Quiz – Write the Triple Subquery
- Quiz – How many rows return on a NOT IN with a NULL?
- How to handle a NOT IN with Potential NULL Values
- IN is equivalent to =ANY
- Using a Correlated Exists
- How a Correlated Exists matches up
- The Correlated NOT Exists
- The Correlated NOT Exists Answer Set
- Quiz – How many rows come back from this NOT Exists?
- Answer – How many rows come back from this NOT Exists?

## Module 15: Substrings and Positioning Functions

- The CHARACTERS Command Counts Characters
- The CHARACTERS Command – Spaces can Count too
- The CHARACTERS Command and Char (20) Data
- Troubleshooting the CHARACTERS Command
- TRIM for Troubleshooting the CHARACTERS Command
- CHARACTERS and CHARACTER\_LENGTH equivalent
- OCTET\_LENGTH
- The TRIM Command trims both Leading and Trailing Spaces
- Trim and Trailing is Case Sensitive
- Trim and Trailing works if Case right
- Trim Combined with the CHARACTERS Command
- How to TRIM only the Trailing Spaces

## Module 15 Continued: Substrings and Positioning Functions

- How to TRIM Trailing Letters
- How to TRIM Trailing Letters and use CHARACTER\_Length
- The SUBSTRING Command
- How SUBSTRING Works with NO ENDING POSITION
- Using SUBSTRING to move Backwards
- How SUBSTRING Works with a Starting Position of -1
- How SUBSTRING Works with an Ending Position of 0
- An Example using SUBSTRING, TRIM and CHAR Together
- SUBSTRING and SUBSTR are equal, but use different syntax
- The POSITION Command finds a Letters Position
- The POSITION Command is brilliant with SUBSTRING
- Quiz – Name that SUBSTRING Starting and For Length
- The POSITION Command is brilliant with SUBSTRING
- Quiz – Name that SUBSTRING Starting and For Length
- Answer to Quiz – Name that Starting and For Length
- Answer to Quiz – Name that Starting and For Length
- Using the SUBSTRING to Find the Second Word On
- Quiz – Why did only one Row Return
- Answer to Quiz – Why Did only one Row Return
- Concatenation
- Concatenation and SUBSTRING
- Four Concatenations Together
- Troubleshooting Concatenation



## Module 16: Interrogating the Data

- Quiz – What would the Answer be?
- Answer to Quiz – What would the Answer be?
- The NULLIFZERO Command
- Quiz – Fill in the Blank Values in the Answer Set
- Answer to Quiz – Fill in the Blank Values in the Answer Set
- Answer to Quiz – Fill in the Blank Values in the Answer Set
- Quiz – Fill in the Answers for the NULLIF Command
- Quiz – Fill in the Answers for the NULLIF Command
- The ZEROIFNULL Command
- Answer to the ZEROIFNULL Question
- The COALESCE Command
- The COALESCE Answer Set
- The Coalesce Quiz
- Answers to the Coalesce Quiz
- The Basics of CAST (Convert and Store)
- Some Great CAST (Convert and Store) Examples
- Some Great CAST (Convert and Store) Examples
- Some Great CAST (Convert and Store) Examples
- A Teradata Extension – The Implied Cast
- The Basics of the CASE Statements
- The Basics of the CASE Statement shown visually
- Valued Case vs. Searched Case
- Quiz - Valued Case Statement
- Answer - Valued Case Statement
- Quiz - Searched Case Statement
- Answer - Searched Case Statement
- Quiz - When NO ELSE is present in CASE Statement
- Answer - When NO ELSE is present in CASE Statement
- When an ELSE is present in CASE Statement
- When NO ELSE is present in CASE Statement
- When an Alias is NOT used in a CASE Statement
- When an Alias is NOT used in a CASE Statement
- When NO ELSE is present in CASE Statement

- Combining Searched Case and Valued Case
- A Trick for getting a Horizontal Case
- Nested Case
- Put a CASE in the ORDER BY

## Module 17: View Functions

- Creating a Simple View
- Basic Rules for Views
- How to Modify a View
- Exceptions to the ORDER BY Rule inside a View
- How to Get HELP with a View
- Views sometimes CREATED for Formatting or Row Security
- Another Way to Alias Columns in a View CREATE
- Resolving Aliasing Problems in a View CREATE
- Resolving Aliasing Problems in a View CREATE
- Resolving Aliasing Problems in a View CREATE
- CREATING Views for Complex SQL such as Joins
- WHY certain columns need Aliasing in a View
- Aggregates on View Aggregates
- Locking Row for Access
- Creating Views for Temporal Tables
- Altering a Table
- Altering a Table after a View has been created
- A View that errors After an ALTER
- Troubleshooting a View
- Updating Data in a Table through a View
- Maintenance Restrictions on a Table through a View



## Module 18: Macro Functions

- The 14 rules of Macros
- CREATING and EXECUTING a Simple Macro
- Multiple SQL Statements inside a Macro
- Complex Joins inside a Macro
- Passing an INPUT Parameter to a Macro
- Troubleshooting a Macro with INPUT Parameters
- Troubleshooting a Macro with INPUT Parameters
- An UPDATE Macro with Two Input Parameters
- Executing a Macro with Named (Not Positional) Parameters
- Troubleshooting a Macro

## Module 19: Set Operators Functions

- Rules of Set Operators
- INTERSECT Explained Logically
- INTERSECT Explained Logically
- UNION Explained Logically
- UNION Explained Logically
- UNION ALL Explained Logically
- UNION Explained Logically
- EXCEPT Explained Logically
- EXCEPT Explained Logically
- Minus Explained Logically
- Minus Explained Logically
- Testing Your Knowledge
- Testing Your Knowledge
- An Equal Amount of Columns in both SELECT List
- Columns in the SELECT list should be from the same Domain
- The Top Query handles all Aliases
- The Bottom Query does the ORDER BY (a Number)
- Great Trick: Place your Set Operator in a Derived Table
- UNION vs. UNION ALL
- UNION vs. UNION ALL Example

## Module 19 Continued: Set Operators Functions

- Using UNION ALL and Literals
- A Great Example of how EXCEPT works
- USING Multiple SET Operators in a Single Request
- Changing the Order of Precedence with Parentheses
- Using UNION ALL for speed in Merging Data Sets
- Using UNION to be same as GROUP BY GROUPING SETS
- Using UNION to be same as GROUP BY ROLLUP
- Using UNION to be the same as GROUP BY Cube
- Using UNION to be same as GROUP BY Cube
- Using UNION to be same as GROUP BY Cube

## Module 20: Creating Tables, Secondary Indexes, and Join Indexes

- Creating a Table with a Unique Primary Index
- Creating a Table with a Non-Unique Primary Index
- Creating a Table without entering a Primary Index
- Creating a Table with NO Primary Index
- Creating a Set Table
- Creating a Multiset Table
- Creating a Set Table with a Unique Primary Index
- Creating a Set Table with a Unique Secondary Index
- Creating a Table with an UPI and USI
- Creating a Table with a Multicolumn Primary Index
- Creating a Unique Secondary Index (USI) after a table is created
- Creating a Non-Unique Secondary Index (NUSI) after a table is created
- Creating a Value-Ordered NUSI
- Data Types



## Module 20 Continued: Creating Tables, Secondary Indexes, and Join Indexes

- Data Types Continued
- Data Types Continued
- Major Data Types and the number of Bytes they take up
- Making an exact copy a Table
- Making a NOT-So-Exact Copy a Table
- Copying a Table
- Troubleshooting Copying and Changing the Primary Index
- Copying only specific columns of a table
- Copying a Table and Keeping the Statistics
- Copying a Table with Statistics
- Copying a table Structure with NO Data but Statistics
- Creating a Table with Fallback
- Creating a Table with No Fallback
- Creating a Table with a Before Journal
- Creating a Table with a Dual Before Journal
- Creating a Table with an After Journal
- Creating a Table with a Dual After Journal
- Creating a Table with a Journal Keyword Alone
- Why Use Journaling?
- Why Use Journaling?
- Creating a Table with Customization of the Data Block Size
- Creating a Table with Customization with FREESPACE Percent
- Creating a QUEUE Table
- Example of how a Queue Table Works
- Example of how a Queue Table Works
- Creating a Columnar Table
- Creating a Columnar Table with Multi-Column Containers
- Creating a Columnar Table with a Row Hybrid
- Creating a Columnar Table with both Row and Column Partitions
- How to Load into a Columnar Table

## Module 20 Continued: Creating Tables, Secondary Indexes, and Join Indexes

- Creating a Columnar Table with NO AUTO COMPRESS
- CREATING a Bi-Temporal Table
- Explaining Bi-Temporal PERIOD Data Types
- Creating a PPI Table with Simple Partitioning
- Creating a PPI Table with RANGE\_N Partitioning per Day
- Creating a PPI Table with RANGE\_N Partitioning per Month
- A Visual of One Year of Data with Range\_N per Month
- Creating a PPI Table with RANGE\_N Partitioning per Week
- A Clever Range\_N Option
- Creating a PPI Table with CASE\_N
- A Visual of Case\_N Partitioning
- Number of PPI Partitions Allowed
- NO CASE and UNKNOWN Partitions Together
- Combining Older Data and Newer Data in PPI
- A Visual for Combining Older Data and Newer Data in PPI
- Multi-Level Partitioning Combining Range\_N and Case\_N
- A Visual of Multi-Level Partitioning
- NON-Unique Primary Indexes (NUPI) in PPI
- PPI Table with a Unique Primary Index (UPI)
- Tricks for Non-Unique Primary Indexes (NUPI)
- A Brilliant Technique for a Unique Secondary Index
- A Brilliant Technique for a Non-Unique Secondary Index
- Character Based PPI for RANGE\_N
- Character-Based PPI for CASE\_N
- Dates and Character-Based Multi-Level PPI
- TIMESTAMP Partitioning
- Using CURRENT\_DATE to define a PPI
- ALTER to CURRENT\_DATE the next year
- ALTER to CURRENT\_DATE with Save



## Module 20 Continued: Creating Tables, Secondary Indexes, and Join Indexes

- Altering a PPI Table to Add or Drop Partitions
- Deleting a Partition
- Deleting a Partition and saving its contents
- Using the PARTITION Keyword in your SQL
- SQL for RANGE\_N
- SQL for CASE\_N
- SQL – User Defined Functions (UDF)
- User Defined Functions
- Creating a Multi-Table Join Index
- Visual of a Join Index
- Outer Join Multi-Table Join Index
- Visual of a Left Outer Join Index
- Compressed Multi-Table Join Index
- Creating a Single-Table Join Index
- Compressed Single-Table Join Index
- Aggregate Join Index
- Sparse Join Index
- A Global Multi-Table Join Index
- Creating a Hash Index

## Module 21: Data Manipulation Language (DML)

- INSERT Syntax # 1
- INSERT Example with Syntax 1
- INSERT Syntax # 2
- INSERT Example with Syntax 2
- INSERT Example with Syntax 3
- Using NULL for Default Values
- INSERT/SELECT Command
- INSERT/SELECT Example using All Columns (\*)
- INSERT/SELECT Example with Less Columns
- INSERT/SELECT to Build a Data Mart
- Fast Path INSERT/SELECT
- NOT quite the Fast Path INSERT/SELECT
- UNION for the Fast Path INSERT/SELECT
- BTEQ for the Fast Path INSERT/SELECT
- The UPDATE Command Basic Syntax
- Two UPDATE Examples
- Subquery UPDATE Command Syntax

## Module 21 Continued: Data Manipulation Language (DML)

- Example of Subquery UPDATE Command
- Join UPDATE Command Syntax
- Example of an UPDATE Join Command
- Fast Path UPDATE
- The DELETE Command Basic Syntax
- Two DELETE Examples to DELETE ALL Rows in a Table
- A DELETE Example Deleting only Some of the Rows
- Subquery and Join DELETE Command Syntax
- Example of Subquery DELETE Command
- Example of Join DELETE Command
- Fast Path DELETE
- Fast Path DELETE Example # 1
- Fast Path DELETE Example # 2
- Fast Path DELETE Example # 3
- MERGE INTO
- MERGE INTO Example that Matches
- MERGE INTO Example that does NOT Match
- OReplace

## Module 22: Stored Procedure Functions

- Stored Procedures vs. Macros
- Creating a Stored Procedure
- How you CALL a Stored Procedure
- Label all BEGIN and END statements except the first ones
- How to Declare a Variable
- How to Declare a Variable and then SET the Variable
- An IN Variable is passed to the Procedure during the CALL
- The IN, OUT and INOUT Parameters
- Using IF inside a Stored Procedure
- Example of two Stored Procedures with different techniques
- Using Loops in Stored Procedures



## Module 22 Continued: Stored Procedure Functions

- You can Name the First Begin and End if you choose
- Using Keywords LEAVE vs. UNTIL for LEAVE vs. REPEAT
- Stored Procedure Basic Assignment
- Answer - Stored Procedure Basic Assignment
- Stored Procedure Advanced Assignment
- Answer - Stored Advanced Assignment

## Module 23: Trigger Functions

- The Fundamentals of Triggers
- CREATING A Trigger
- FOR EACH STATEMENT vs. FOR EACH ROW
- Using ORDER when Similar Triggers Exist

## Module 24: Math Functions

- What is the Order of Precedents?
- What is the Answer to this Math Question?
- What is the Answer to this Math Question?
- What is the Answer to this Math Question?

## Module 25: Sample

- The SAMPLE Function and Syntax
- SAMPLE Function Examples
- A SAMPLE Example that asks for Multiple Samples
- A SAMPLE Example with the SAMPLEID
- A SAMPLE Example WITH REPLACEMENT
- A SAMPLE Example with Four 10% Samples
- A Randomized SAMPLE
- A SAMPLE with Conditional Logic
- Aggregates and A SAMPLE using a Derived Table
- Random Number Generator
- Using Random to SELECT a Percentage of Rows
- Using Random and Aggregations

## Module 26: Statistical Aggregate Functions

- The Stats Table
- The KURTOSIS Function
- A Kurtosis Example
- The SKEW Function
- A SKEW Example
- 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 CORR Function
- A CORR Example
- Another CORR Example so you can compare
- The COVAR\_POP Function
- A COVAR\_POP Example
- Another COVAR\_POP Example so you can Compare
- The REGR\_INTERCEPT Function
- A REGR\_INTERCEPT Example
- Another REGR\_INTERCEPT Example so you can compare
- The REGR\_SLOPE Function
- A REGR\_SLOPE Example
- Another REGR\_SLOPE Example so you can compare
- Using GROUP BY
- No Having Clause vs. Use of HAVING





# Teradata SQL v15

Course ID#: 1190-110-15-W

Hours: 35

## Module 27: Explain

- EXPLAIN Keywords
- EXPLAIN Keywords Continued
- Explain Example – Full Table Scan
- Explain Example – Unique Primary Index (UPI)
- Explain Example – Non-Unique Primary Index (NUPI)
- Explain Example – Unique Secondary Index (USI)
- Explain Example – Redistributed to All-AMPs
- Explain Example – Row Hash Match Scan
- Explain Example – Duplicated on All-AMPs
- Explain Example – Low Confidence
- Explain Example – High Confidence
- Explain Example – Product Join
- Explain Example – BMSMS
- Explain Terminology for Partitioned Primary Index Tables
- Explain Example – From a Single Partition
- Explain Example – From N Partitions
- Explain Example – Partitions and Current\_Date

## Module 28: Collect Statistics

- The Teradata Parsing Engine (Optimizer) is Cost Based
- The Purpose of Collect Statistics
- When Teradata Collects Statistics, it creates a Histogram
- The Interval of the Collect Statistics Histogram
- Histogram Quiz
- Answers to Histogram Quiz
- What to COLLECT STATISTICS On?
- Why Collect Statistics?
- How do you know if Statistics were collected on a Table?
- A Huge Hint that No Statistics Have Been Collected
- The Basic Syntax for COLLECT STATISTICS
- COLLECT STATISTICS Examples for a better Understanding

## Module 28 Continued: Collect Statistics

- The New Teradata V14 Way to Collect Statistics
- Where Does Teradata Keep the Collected Statistics?
- The Official Syntaxes for COLLECT STATISTICS
- How to Recollect STATISTICS on a Table
- Teradata Always Does a Random AMP Sample
- Random Sample is kept in the Table Header in FSG Cache
- Multiple Random AMP Samplings
- How a Random AMP gets a Table Row count
- Random AMP Estimates for NUSI Secondary Indexes
- USI Random AMP Samples are Not Considered
- There's No Random AMP Estimate for Non-Indexed Columns
- Summary of the PE Plan if No Statistics Were Collected
- Stale Statistics Detection and Extrapolation
- Extrapolation for Future Dates
- How to Copy a Table with Data and the Statistics?
- COLLECT STATISTICS Directly From another Table
- How to Copy a Table with NO Data and the Statistics?
- When to COLLECT STATISTICS Using only a SAMPLE
- Examples of COLLECT STATISTICS Using only a SAMPLE
- Examples of COLLECT STATISTICS for V14
- How to Collect Statistics on a PPI Table on the Partition
- Teradata V12 and V13 Statistics Enhancements
- Teradata V14 Statistics Enhancements
- Teradata V14 Summary Statistics
- Teradata V14 MaxValueLength
- Teradata V14 MaxIntervals
- Teradata V14 Sample N Percent
- Teradata Statistics Wizard



# Teradata SQL v15

Course ID#: 1190-110-15-W

Hours: 35

## Module 29: Hashing Functions

- Hashing Functions on Teradata
- The HASHROW Function
- The HASHROW Function in a real-world Example
- The HASHBUCKET Function
- The HASHBUCKET Function in a real-world Example
- The HASHAMP Function
- The HASHAMP Function in a real-world Example
- A Great HASHAMP Function for Large Tables
- The HASHBAKAMP Function
- A Real-World HASBAKHAMP Function Example
- A Great way to see distribution for Primary and Fallback rows

## Module 30: BTEQ – Batch Teradata Query

- BTEQ – Batch Teradata Query Tool
- How to Logon to BTEQ in Interactive Mode
- Running Queries in BTEQ in Interactive Mode
- BTEQ Commands vs. BTEQ SQL Statements
- WITH BY Command for Subtotals
- WITH Command for a Grand Total
- WITH and WITH BY Together for Subtotals and Grand Totals
- How to Logon to BTEQ in a SCRIPT
- Running Queries in BTEQ through a Batch Script
- Running a BTEQ Batch Script through the Command Prompt
- Running a BTEQ Batch Script through the Run Command
- Using Nexus to Build Your BTEQ Scripts
- Using Nexus to Build Your BTEQ Scripts
- Using BTEQ Scripts to IMPORT Data
- What Keywords Mean in a BTEQ Script
- Creating a BTEQ IMPORT for a Comma Separated Value File

## Module 30 Continued: BTEQ – Batch Teradata Query

- Four Great Examples/Ways to Run a Teradata BTEQ Script
- BTEQ Export – Four types of Export Variations
- Creating a BTEQ Export Script in Record Mode
- Creating a BTEQ Export Script in Report Mode
- The Appearance of Record Mode Vs Report Mode Data
- Using Report Mode to Create a Comma Separated Report
- Creating a BTEQ IMPORT for a Comma Separated Value File
- Using Multiple Sessions in BTEQ
- BTEQ Fast Path Inserts
- BTEQ Can Use Conditional Logic
- Using a BTEQ Export and Setting a Limit In a UNIX System

## Module 31: Top SQL Commands Cheat Sheet

- SELECT All Columns from a Table and Sort
- Select Specific Columns and Limiting the Rows
- Changing your Default Database
- Keywords that describe you
- Select TOP Rows in a Rank Order
- A Sample number of rows
- Getting a Sample Percentage of rows
- Find Information about a Database
- Find information about a Table
- Using Aggregates
- Performing a Join
- Performing a Join using ANSI Syntax
- Using Date, Time and Timestamp
- Using Date Functions
- Using the System Calendar
- Using the System Calendar in a Query
- Formatting Data
- Using Rank
- Using a Derived Table
- Using a Subquery



# Teradata SQL v15

Course ID#: 1190-110-15-W

Hours: 35

## Module 31 Continued: Top SQL Commands

### Cheat Sheet

- Correlated Subquery
- Using Substring
- Basic CASE Statement
- Advanced CASE Statement
- Using an Access Lock in your SQL
- Collect Statistics
- CREATING a Volatile Table with a Primary Index
- CREATING a Volatile Table that is Partitioned (PPI)
- CREATING a Volatile Table that is deleted after the Query
- Finding the Typical Rows per Value for specific column
- Finding out how much Space you have
- How much Space you have Per AMP
- Finding your Space
- Finding Space Skew in Tables in a Database
- Finding the Number of rows per AMP for a Column
- Finding Account Information
- Ordered Analytics