Course Title: Advanced SQL for Data Analysis & Business Intelligence

- **Duration:** 5 Days (approximately 35-40 instructional hours)
- **Course Goal:** To provide data analysts, BI professionals, and data-savvy business users with a deep, practical mastery of advanced SQL techniques. This course will move beyond simple SELECT statements to cover complex joins, subqueries, window functions, and Common Table Expressions (CTEs), enabling participants to solve sophisticated analytical problems, prepare data for BI tools, and write clean, efficient, and powerful SQL code.
- **Core Focus:** Mastering the advanced querying and data manipulation capabilities of SQL for complex analytical tasks.

Learning Objectives:

Upon successful completion of this workshop, participants will be able to:

- **Write** and deconstruct complex, multi-level SQL queries.
- **Master** all types of JOINs, including self-joins and complex multi-table joins.
- **Utilize** Common Table Expressions (CTEs) to write clean, readable, and modular SQL.
- **Master** the use of powerful Window Functions for ranking, aggregation over a window, and calculating running totals and moving averages.
- **Apply** advanced filtering and aggregation techniques using GROUP BY, HAVING, and grouping sets.
- **Write** and interpret sophisticated subqueries (scalar, multi-row, correlated).
- **Perform** advanced data manipulation and transformation using CASE statements and data type conversions.
- **Understand** and apply techniques for performance tuning basic queries.
- **Prepare** complex datasets for ingestion into BI tools like Power BI or Tableau.
- **Solve** complex, real-world business problems using only SQL.

Target Audience:

- Data Analysts, Business Intelligence (BI) Analysts, and BI Developers.
- Data Scientists and Machine Learning Engineers.
- Marketing, Finance, and Operations Analysts who frequently work with data.
- Anyone who has a solid foundation in basic SQL and wants to take their skills to the next level.

Prerequisites:

- **Solid, demonstrable experience with basic SQL is essential.** Participants MUST be comfortable with SELECT, FROM, WHERE, basic JOINs, GROUP BY, HAVING, and ORDER BY. This is *not* an introductory course.
- Familiarity with a relational database environment.

Teaching Methodology:

This is an intensely hands-on, "code-along" workshop. The methodology includes:

- **Interactive Lectures:** Introducing advanced concepts with clear explanations and syntax breakdowns.
- **Live Coding Demonstrations:** The instructor will write and deconstruct complex queries live.
- **Intensive, Hands-on "Query Labs":** The core of the course. Participants will spend the majority of their time writing SQL to solve a series of progressively difficult business problems using a rich, multi-table database.
- **"SQL Puzzles" & Challenges:** Short, challenging problems designed to test understanding of specific functions.
- **Code Review & Optimization:** Participants will share their solutions and discuss different, more efficient ways to write the same query.

ADEPT TRAINING CENTER

Materials Provided:

- Comprehensive digital or printed workbook with advanced SQL concepts, syntax guides, and detailed lab instructions.
- Access to a dedicated database (e.g., PostgreSQL, SQL Server, BigQuery) pre-loaded with a complex, realistic dataset (e.g., e-commerce, financial transactions).
- A "Advanced SQL Playbook" with scripts and templates for common analytical patterns.
- A SQL style guide for writing readable and maintainable code.
- Certificate of Completion.

Detailed Day-by-Day Curriculum

Day 1: Mastering Joins & Set Operators

- **Module 1: Advanced SQL Foundations & Course Overview (Approx. 1.5 hours)**
    - Welcome, introductions, and course objectives.
    - Brief recap of foundational SQL (SELECT, WHERE, basic JOIN, GROUP BY).
    - Introduction to the course database schema and the business context.
    - The mindset of an analytical SQL user: Thinking in sets.
- **Module 2: Deep Dive into JOINs (Approx. 3.5 hours)**
    - Review of INNER JOIN and LEFT JOIN.
    - Mastering RIGHT JOIN and FULL OUTER JOIN and their use cases.
    - **Self-Joins:** Joining a table to itself to analyze hierarchical data or sequential events.
    - **CROSS JOINs:** Understanding the Cartesian product and when it can be useful (or dangerous).
    - Joining on multiple conditions and non-key columns.
    - **Intensive Hands-on Lab:** A series of complex join exercises to answer questions like "Find all customers who have *never* purchased a certain product" or "Find all employees who report to the same manager."
- **Module 3: Combining Result Sets with Set Operators (Approx. 2.5 hours)**
    - UNION: Stacking results and removing duplicates.
    - UNION ALL: Stacking results and keeping duplicates.
    - INTERSECT: Finding common rows between two queries.
    - EXCEPT / MINUS: Finding rows that exist in the first query but not the second.
    - Understanding the rules for using set operators (e.g., column count and data types).
    - **Hands-on Lab:** Using set operators to compare customer lists and analyze data from different time periods.
    - **End of Day 1 Recap & Q&A.**

Day 2: Advanced Aggregation & Subqueries

- **Module 4: Advanced GROUP BY Techniques (Approx. 2.5 hours)**
    - Grouping by multiple columns.
    - Advanced aggregate functions (e.g., COUNT(DISTINCT ...)).
    - Introduction to GROUPING SETS, CUBE, and ROLLUP for generating multiple levels of subtotals in a single query.
- **Module 5: Deep Dive into Subqueries (Approx. 3 hours)**
    - Recap of basic subqueries in the WHERE clause.
    - **Scalar Subqueries:** Subqueries that return a single value.
    - **Multi-row Subqueries:** Using operators like IN, NOT IN, ANY, ALL.
    - **Subqueries in the FROM clause (Derived Tables):** A powerful technique for pre-processing or summarizing data before joining.

ADEPT TRAINING CENTER

- **Correlated Subqueries:** Subqueries that depend on the outer query for their values. Understanding how they work and their performance implications.

- **Module 6: Subquery & Join Integration Lab (Approx. 2 hours)**
  - **Intensive Hands-on Lab:** Solving a series of complex business problems that require a combination of joins, aggregations, and different types of subqueries.
  - Comparing solutions that use subqueries vs. those that use joins.
  - **End of Day 2 Recap & Q&A.**

Day 3: The Power of Common Table Expressions (CTEs) & Window Functions

- **Module 7: Writing Cleaner Code with Common Table Expressions (CTEs) (Approx. 3 hours)**
  - The problem with deeply nested subqueries: They are hard to read and debug.
  - Introduction to the WITH clause to define CTEs.
  - Using CTEs to break down a complex query into logical, readable steps.
  - Using multiple CTEs in a single query.
  - Recursive CTEs for hierarchical data (e.g., organizational charts) - overview.
  - **Workshop:** "Rewrite the Subquery" – Participants take complex queries written with subqueries from Day 2 and rewrite them using CTEs.

- **Module 8: Introduction to Window Functions (The Game Changer) (Approx. 4 hours)**
  - What are Window Functions? Performing calculations across a set of table rows that are somehow related to the current row.
  - The OVER() clause: The key to window functions.
  - **Aggregation as Window Functions:** SUM() OVER (...), COUNT() OVER (...), etc. (e.g., showing a total alongside individual rows).
  - The PARTITION BY clause: Defining the "window" or group over which to calculate.
  - **Hands-on Lab:** Writing queries to calculate running totals, percent of total, and compare a row's value to the group's average, all without using GROUP BY.
  - **End of Day 3 Recap & Q&A.**

Day 4: Mastering Window Functions & Advanced Data Manipulation

- **Module 9: Ranking & Ordering with Window Functions (Approx. 3 hours)**
  - **Ranking Functions:** RANK(), DENSE_RANK(), ROW_NUMBER(). Understanding the subtle differences.
  - **The ORDER BY clause within OVER()**.
  - Use cases: Finding the "Top N" items per category (e.g., top 3 selling products in each region).
  - **Intensive Hands-on Lab:** A series of "Top N" challenges requiring the use of CTEs and ranking functions.

- **Module 10: LAG, LEAD, and Positional Window Functions (Approx. 2.5 hours)**
  - **LAG():** Accessing data from a previous row in the window.
  - **LEAD():** Accessing data from a subsequent row in the window.
  - Use cases: Calculating period-over-period changes, analyzing user session data.
  - Other positional functions: FIRST_VALUE, LAST_VALUE.

- **Module 11: Advanced Data Manipulation with CASE (Approx. 2 hours)**
  - The power of the CASE statement for conditional logic within a query.
  - Using CASE for:
    - Creating custom categories or segments.
    - Handling conditional aggregations.
    - Creating flags and indicators.
  - Combining CASE with aggregate functions.

ADEPT TRAINING CENTER

- **Hands-on Lab:** Writing queries that use CASE statements to perform customer segmentation and create complex conditional summaries.
- **End of Day 4 Recap & Q&A.**

Day 5: Performance Tuning, Preparing Data for BI & Capstone

- **Module 12: SQL Performance Tuning for Analysts (Basics) (Approx. 2.5 hours)**
  - Why query performance matters.
  - Understanding the EXPLAIN plan (conceptual).
  - The critical role of **Indexes**.
  - Common performance killers: Inefficient joins, correlated subqueries, filtering after aggregation.
  - Best practices for writing more efficient SQL.
- **Module 13: Preparing & Shaping Data for BI Tools (Approx. 2 hours)**
  - The ideal data structure for tools like Power BI and Tableau (tall, denormalized tables).
  - Using SQL to pivot and unpivot data.
  - Creating aggregated summary tables in the database to improve dashboard performance.
  - The role of SQL Views in BI.
- **Module 14: Capstone Project: The End-to-End Analysis (Approx. 3 hours)**
  - **Final Project:** Participants are given a new, complex business problem that requires them to integrate skills from the entire week.
  - They must write a single, clean, well-documented query (likely using CTEs and window functions) to answer the problem.
  - Example problem: "For each month, find the top 10 customers by sales, and for each of those customers, show their total sales for the month, and their sales from the previous month."
  - Participants present and explain their final query.
- **Module 15: Course Review & The Modern Data Stack (Approx. 0.5 hours)**
  - Review of key advanced SQL functions.
  - A brief overview of where SQL fits into the modern data stack (dbt, data warehouses).
  - Open Q&A, course evaluation, and closing.

ADEPT TRAINING CENTER