Data Analyst Training Programme

Window Functions & Complex JOINs

Introduction to Advanced SQL Techniques

Beyond Basic Aggregation

- Window functions provide row-by-row calculations
- Complex JOINs handle sophisticated data relationships
- Advanced analytical capabilities for enterprise reporting

Window Functions vs GROUP BY

- GROUP BY collapses rows into summaries
- Window functions preserve individual rows whilst adding calculations
- Enables ranking, running totals, and comparative analysis

Complex JOIN Scenarios

- Multiple table relationships
- Self-joins for hierarchical data
- Cross joins for cartesian products
- Advanced filtering and conditioning

Ranking and Row Number Functions

Essential Ranking Functions

- **ROW_NUMBER():** unique sequential numbers
- **RANK():** tied values get same rank, gaps in sequence
- **DENSE_RANK():** tied values get same rank, no gaps
- **NTILE():** divide data into equal buckets

PARTITION BY Clause

- Creates separate ranking groups
- Resets ranking for each partition
- Enables comparative analysis within categories

Business Applications

- Top performers by region
- Product rankings within categories
- Customer segmentation and quartiles
- Sales league tables

Aggregate Window Functions

Running Calculations

- **SUM() OVER:** cumulative totals
- **AVG() OVER:** moving averages
- **COUNT() OVER:** running counts
- **MIN()/MAX() OVER:** running extremes

Window Frame Specification

- **ROWS BETWEEN:** physical row boundaries
- **RANGE BETWEEN:** logical value boundaries
- **UNBOUNDED PRECEDING/FOLLOWING:** from start/to end
- **CURRENT ROW:** reference point

Business Intelligence Applications

- Sales trend analysis
- Customer lifetime value tracking
- Performance against moving benchmarks
- Growth rate calculations

LAG, LEAD, and Analytical Functions

Value Access Functions

- **LAG():** access previous row values
- **LEAD():** access next row values
- **FIRST_VALUE():** first value in window
- **LAST_VALUE():** last value in window

Growth and Change Analysis

- Period-over-period comparisons
- Growth rate calculations
- Trend direction identification
- Sequential pattern analysis

Advanced Analytical Scenarios

- Customer retention analysis
- Product lifecycle tracking
- Seasonal comparison studies
- Performance trajectory mapping

Complex JOIN Patterns

Self-JOINs for Hierarchical Data

- Connect tables to themselves
- Employee-manager relationships
- Product comparison analysis
- Sequential record connections

Multiple Table JOINs with Conditions

- Chain multiple relationships
- Filter at different JOIN levels
- Conditional JOIN criteria
- Performance optimisation strategies

Advanced JOIN Applications

- Basket analysis with self-joins
- Comparative product analysis
- Customer relationship mapping
- Time-series data connections

Assignment

Complete these advanced SQL challenges using the Northwind database:

Window Function Fundamentals

- Rank customers by total spending within each country
- Calculate running totals of monthly sales revenue
- Find the top 3 products by revenue in each category using window functions

Advanced Analytical Functions

- Create a month-over-month growth analysis showing revenue changes and growth percentages
- Build a customer lifetime value analysis with cumulative spending and average order values
- Develop a sales trend analysis comparing each month to the same month in the previous year

Complex JOIN Scenarios

- Perform a market basket analysis to find products frequently bought together
- Create a customer analysis joining multiple tables with conditional criteria
- Build a product comparison report using self-JOINs to identify similar products within categories

Integrated Advanced Analysis

- Design a sales performance dashboard combining window functions and complex JOINs
- Create a customer segmentation analysis using quartiles, growth rates, and purchasing patterns
- Build a product lifecycle analysis showing performance trends, competitive positioning, and market share
- Develop an executive summary report combining all advanced techniques to provide strategic business insights

Until Next Week Sunday...

See you next week on Sunday, **[student name]**.

Window functions and complex JOINs represent the advanced techniques that separate expert data analysts from beginners, enabling sophisticated time-series analysis and multi-dimensional business intelligence. You now possess the complete SQL toolkit to handle enterprise-level analytical challenges and generate strategic insights that drive executive decision-making.

Thank you, [student name].

Any Questions?