Course Outline

Intermediate/Advanced SQL Training (Custom for TRS of Texas)

Course Summary

Description

This course covers ANSI/ISO standard SQL, with examples in SQL Server, Oracle, DB2 LUW and z/OS, and MySQL, with emphasis on SQL Server and Oracle.

Objectives

By the end of this course, students will be able to:

- Sample a very large database to have accurate knowledge of the structure and contents
- Use several tools to monitor and possibly improve the performance of complex queries on large databases, including using the query execution plan, optimizer hints, and other tools
- Know when and how to use temporary tables, query-scope tables, Common Table Expressions, and subqueries of various types to solve complex problems
- Handle all logical and performance issues with joins and subqueries on large tables
- Handle recursive relationships with recursive With, the Oracle Connect By, and/or the SQL Server HierarchyID data type
- Handle character data with built-in functions, Soundex, full-text searches, and Regular Expressions
- Use Case logic to control how sorting, grouping and other operations work
- Summarize data, including handling missing values, creating pivot reports using Pivot or Case logic, using Rollup and Cube, and using basic analytic (OLAP) functions
- Understand basic relational database design principles, and how tables and other objects are created and maintained
- Follow best practices to improve security of data
- Create and manage complex views, and handle updating of views, the Check Option, materialized views, and virtual columns
- Create and manage indexes, including knowing when to create, and when not to create, indexes, and the use of partitioning, clustering, filtered, full-text, and other types of indexes
- Create stored procedures, including the use of required and optional parameters, variables, conditional statements, looping, cursors, exception handling, transactions, and debugging
- Create user defined functions, including the use of parameters, and all language features; create scalar and table functions, and understand the differences between table functions, stored procedures, and views
- Create triggers, including table, database and server triggers, before, instead of, and after triggers, handling various events, using the "transition" data involved, and dealing with transaction issues

Topics

- Introduction
- Multiple tables
- Text handling issues
- Case logic
- SQL summarization
- Stored Procedures
- User-defined Functions
- Performance considerations
- Temp tables

Course Outline

Intermediate/Advanced SQL Training (Custom for TRS of Texas)

Course Summary (cont.)

Audience

This course is designed for application developers.

Prerequisite

Before taking this course, students should take an SQL Basics course, or have equivalent work experience, and had several weeks of experience using SQL.

Duration

Three or Four Days

- Three days of intense, fast pace training based on the course outline.
- Four days of training would allow for a slower pace, more question and answer time allowing for deeper discussions and additional hands-on time to complete all of the activities.

The format is lecture/questions about a topic followed by a lab activity to reinforce the student's understanding. To ensure even greater understanding, the instructor will add a demonstration to review with the group. If time allows, the students will be asked to type the code along with the instructor, so they get more hands-on time. Since all students work at a different pace, some of them will complete the activities and some won't. The instructor will then review and move on to the next topic. The SQL Intermediate portion covers a lot of ground in a short amount of time because the basics of SQL are assumed to be known.

Course Outline

Intermediate/Advanced SQL Training (Custom for TRS of Texas)

Course Outline

I. Introduction

II. Multiple tables

A. Review Inner joins
B. Inner vs outer joins
C. Left, Right, and Full outer joins
D. Join performance consideration
E. Recursive joins and complex relationships
F. Difference between "Not Exists" and "Not IN"
G. Set operators: Union, Union All

III. Text handling issues

A. Text handling functions
B. Unicode
C. Regular expressions
D. Creating a full-text index, using the contains function

IV. Case logic

A. Review of simple and searched When clauses
B. Cases used in:
   1. Functions and expressions
   2. From clauses
   3. Order by clause
   4. Update
   5. Nested Selects
   6. In Group by

V. SQL summarization

A. Review Group by
B. Filtering groups with Having
C. Using Pivot clause & Case logic
D. Analytic (OLAP) functions
E. Performance issues

VI. Stored Procedures

A. T-SQL
B. Parameters: Input, output, optional, use of default values
C. Variables, conditions, looping
D. Returning a result set
E. Transaction handling
F. Debugging

VII. User-defined Functions

A. Scalar functions
B. Table-value functions

VIII. Performance considerations

A. Minimize locking
B. Maximize index use
C. Viewing a query execution plan
D. Getting performance help
E. Optimizer hints

IX. Temp tables

A. Creating temporary and query-scope tables and indexes
B. Common Table Expressions (CTEs)