Les bases du SQL

Les bases du SQL

This course is designed to introduce students to Transact-SQL.
It is designed in such a way that the first two days can be taught as a course to students requiring the knowledge for other courses in the SQL Server curriculum.

Détails

- Code : DB-SQLB
- Durée : 2 jours (14 heures)

Public

- Administrateurs
- Administrateurs de bases de données
- Consultants
- Consultants informatiques
- Developpeurs
- Ingénieurs

Pré-requis

Objectifs

- Maitriser les fonctionnalités standards du langage SQL.
- oumettre des requêtes en interactif et interpréter les résultats.

Programme

Introduction to Microsoft SQL Server

- Description
  - This module introduces SQL Server, the versions of SQL Server, including cloud versions, and how to connect to SQL Server using SQL Server Management Studio
- Lessons
  - The Basic Architecture of SQL Server
  - SQL Server Editions and Versions
  - Getting Started with SQL Server Management Studio
  - Working with SQL Server Management Studio
  - Creating and Organizing T-SQL Scripts
- After completing this module, you will be able to:
  - Describe relational databases and Transact-SQL queries
  - Describe the on-premise and cloud-based editions and versions of SQL Server
  - Describe how to use SQL Server Management Studio (SSMS) to connect to an instance of SQL Server, explore the databases contained in the instance, and work with script files that contain T-SQL queries

Module 2: Introduction to T-SQL Querying

- Description
  - This module describes the elements of T-SQL and their role in writing queries
  - Describe the use of sets in SQL Server
  - Describe the use of predicate logic in SQL Server
  - Describe the logical order of operations in SELECT statements
- Lessons
  - Introducing T-SQL
  - Understanding Sets
  - Understanding Predicate Logic
  - Understanding the Logical Order of Operations in SELECT statements
- Lab : Introduction to T-SQL Querying
  - Executing Basic SELECT Statements
  - Executing Queries that Filter Data using Predicates
  - Executing Queries That Sort Data Using ORDER BY
- After completing this module, you will be able to:
  - Describe the role of T-SQL in writing SELECT statements
  - Describe the elements of the T-SQL language and which elements will be useful in writing queries
  - Describe the concepts of the set theory, one of the mathematical underpinnings of relational databases, and to help you apply it to how you think about querying SQL Server
  - Describe predicate logic and examine its application to querying SQL Server
  - Explain the elements of a SELECT statement, delineate the order in which the elements are evaluated, and then apply this understanding to a practical approach to writing queries

Module 3: Writing SELECT Queries

- Description
  - This module introduces the fundamentals of the SELECT statement, focusing on queries against a single table
- Lessons
  - Writing Simple SELECT Statements
  - Eliminating Duplicates with DISTINCT
  - Using Column and Table Aliases
  - Writing Simple CASE Expressions
- Lab : Writing Basic SELECT Statements

- Writing Simple SELECT Statements
  - Eliminating Duplicates Using DISTINCT
  - Using Column and Table Aliases
  - Using a Simple CASE Expression
- After completing this module, you will be able to:
  - Describe the structure and format of the SELECT statement, as well as enhancements that will add functionality and readability to your queries
  - Describe how to eliminate duplicates using the DISTINCT clause
  - Describe the use of column and table aliases
  - Understand and use CASE expressions

Module 4: Querying Multiple Tables

- Description
  - This module describes how to write queries that combine data from multiple sources in Microsoft SQL Server
- Lessons
  - Understanding Joins
  - Querying with Inner Joins
  - Querying with Outer Joins
  - Querying with Cross Joins and Self Joins
- Lab : Querying Multiple Tables
  - Writing Queries that use Inner Joins
  - Writing Queries that use Multiple-Table Inner Joins
  - Writing Queries that use Self-Joins
  - Writing Queries that use Outer Joins
  - Writing Queries that use Cross Joins
- After completing this module, you will be able to:
  - Explain the fundamentals of joins in SQL Server
  - Write inner join queries
  - Write queries that use outer joins
  - Use additional join types

Module 5: Sorting and Filtering Data

- Description
  - This module describes how to implement sorting and filtering
- Lessons
  - Sorting Data
  - Filtering Data with Predicates
  - Working with Unknown Values
- Lab : Sorting and Filtering Data
  - Writing Queries that Filter Data using a WHERE Clause
  - Writing Queries that Sort Data Using an ORDER BY Clause
  - Writing Queries that Filter Data Using the TOP Option
- After completing this module, you will be able to:
  - Explain how to add an ORDER BY clause to your queries to control the order of rows displayed in your query’s output
  - Explain how to construct WHERE clauses to filter out rows that do not match the predicate
  - Explain how to limit ranges of rows in the SELECT clause using a TOP option
  - Explain how three-valued logic accounts for unknown and missing values, how SQL Server uses NULL to mark missing values, and how to test for NULL in your queries

Module 6: Working with SQL Server Data Types

- Description
  - This module introduces the data types SQL Server uses to store data
- Lessons
  - Introducing SQL Server Data Types
  - Working with Character Data
  - Working with Date and Time Data
- After completing this module, you will be able to:
  - Explore many of the data types SQL Server uses to store data and how data types are converted between types
  - Explain the SQL Server character-based data types, how character comparisons work, and some common functions you may find useful in your queries
  - Describe data types that are used to store temporal data, how to enter dates and times so they will be properly parsed by SQL Server, and how to manipulate dates and times with built-in functions

Module 7: Using DML to Modify Data

- Description
  - This module describes how to create DML queries, and why you would want to
- Lessons
  - Adding Data to Tables
  - Modifying and Removing Data
  - Generating automatic column values
- Lab : Using DML to Modify Data
  - Inserting Records with DML
  - Updating and Deleting Records Using DML
- After completing this module, you will be able to:
  - Use INSERT and SELECT INTO statements
  - Use UPDATE, MERGE, DELETE, and TRUNCATE

Module 8: Using Built-In Functions

- Description
  - This module introduces some of the many built in functions in SQL Server
- Lessons
  - Writing Queries with Built-In Functions
  - Using Conversion Functions
  - Using Logical Functions
  - Using Functions to Work with NULL
- Lab : Using Built-In Functions
  - Writing Queries That Use Conversion Functions
  - Writing Queries that use Logical Functions
  - Writing Queries that Test for Nullability
- After completing this module, you will be able to:
  - Describe the types of functions provided by SQL
  - Server, and then focus on working with scalar functions
  - Explain how to explicitly convert data between types using several SQL Server functions
  - Describe how to use logical functions that evaluate an expression and return a scalar result
  - Describe additional functions for working with NULL

Module 9: Grouping and Aggregating Data

- Description

  - This module describes how to use aggregate functions
- Lessons
  - Using Aggregate Functions
  - Using the GROUP BY Clause
  - Filtering Groups with HAVING
- Lab : Grouping and Aggregating Data
  - Writing Queries That Use the GROUP BY Clause
  - Writing Queries that Use Aggregate Functions
  - Writing Queries that Use Distinct Aggregate Functions
  - Writing Queries that Filter Groups with the HAVING Clause
- After completing this module, you will be able to:
  - Describe the built-in aggregate function in SQL
  - Server and write queries using it
  - Write queries that separate rows using the GROUP BY clause
  - Write queries that use the HAVING clause to filter groups

Module 10: Using Subqueries

- Description
  - This module describes several types of subquery and how and when to use them
- Lessons
  - Writing Self-Contained Subqueries
  - Writing Correlated Subqueries
  - Using the EXISTS Predicate with Subqueries
- Lab : Using Subqueries
  - Writing Queries That Use Self-Contained Subqueries
  - Writing Queries That Use Scalar and Multi-Result Subqueries
  - Writing Queries That Use Correlated Subqueries and an EXISTS Clause
- After completing this module, you will be able to:
  - Describe where subqueries may be used in a SELECT statement
  - Write queries that use correlated subqueries in a SELECT statement
  - Write queries that use EXISTS predicates in a WHERE clause to test for the existence of qualifying rows
  - Use the EXISTS predicate to efficiently check for the existence of rows in a subquery

Module 11: Using Table Expressions

- Description
  - Previously in this course, you learned about using subqueries as an expression that returned results to an outer calling query
  - Like subqueries, table expressions are query expressions, but table expressions extend this idea by allowing you to name them and to work with their results as you would work with data in any valid relational table
  - Microsoft SQL Server supports four types of table expressions: derived tables, common table expression (CTEs), views, and inline table-valued functions (TVFs)
  - In this module, you will learn to work with these forms of table expressions and learn how to use them to help create a modular approach to writing queries
- Lessons
  - Using Views
  - Using Inline Table-Valued Functions
  - Using Derived Tables
  - Using Common Table Expressions
- Lab : Using Table Expressions
  - Writing Queries That Use Views
  - Writing Queries That Use Derived Tables
  - Writing Queries That Use Common Table Expressions (CTEs)
  - Writing Queries That Use Inline Table-Valued Expressions (TVFs)
- After completing this module, you will be able to:
  - Write queries that return results from views
  - Use the CREATE FUNCTION statement to create simple inline TVFs
  - Write queries that create and retrieve results from derived tables
  - Write queries that create CTEs and return results from the table expression

Module 12: Using Set Operators

- Description
  - This module introduces how to use the set operators UNION, INTERSECT, and EXCEPT to compare rows between two input sets
- Lessons
  - Writing Queries with the UNION operator
  - Using EXCEPT and INTERSECT
  - Using APPLY
- Lab : Using Set Operators
  - Writing Queries That Use UNION Set Operators and UNION ALL
  - Writing Queries That Use CROSS APPLY and OUTER APPLY Operators
  - Writing Queries That Use the EXCEPT and INTERSECT Operators
- After completing this module, students will be able to:
  - Write queries that use UNION to combine input sets
  - Write queries that use UNION ALL to combine input sets
  - Write queries that use the EXCEPT operator to return only rows in one set but not another
  - Write queries that use the INTERSECT operator to return only rows that are present in both sets
  - Write queries using the CROSS APPLY operator
  - Write queries using the OUTER APPLY operator

Module 13: Using Windows Ranking, Offset, and Aggregate Functions

- Description
  - This module describes the benefits to using window functions
  - Restrict window functions to rows defined in an OVER clause, including partitions and frames
  - Write queries that use window functions to operate on a window of rows and return ranking, aggregation, and offset comparison results
  - This module is not needed for most applications, so it will only be thaught when specifically requested
- Lessons
  - Creating Windows with OVER
  - Exploring Window Functions
- Lab : Using Windows Ranking, Offset, and Aggregate

  - Functions
  - Writing Queries that use Ranking Functions
  - Writing Queries that use Offset Functions
  - Writing Queries that use Window Aggregate Functions
- After completing this module, students will be able to:
  - Describe the T-SQL components used to define windows, and the relationships between them
  - Write queries that use the OVER clause, with partitioning, ordering, and framing to define windows
  - Write queries that use window aggregate functions
  - Write queries that use window ranking functions
  - Write queries that use window offset functions

Module 14: Pivoting and Grouping Sets

- Description
  - This module describes write queries that pivot and unpivot result sets
  - Write queries that specify multiple groupings with grouping sets
- Lessons
  - Writing Queries with PIVOT and UNPIVOT
  - Working with Grouping Sets
- Lab : Pivoting and Grouping Sets
  - Writing Queries that use the PIVOT Operator
  - Writing Queries that use the UNPIVOT Operator
  - Writing Queries that use the GROUPING SETS CUBE and ROLLUP Subclauses
- After completing this module, students will be able to:
  - Describe how pivoting data can be used in T-SQL queries
  - Write queries that pivot data from rows to columns using the PIVOT operator
  - Write queries that unpivot data from columns back to rows using the UNPIVOT operator
  - Write queries using the GROUPING SETS subclause
  - Write queries that use ROLLUP AND CUBE
  - Write queries that use the GROUPING_ID function

Module 15: Executing Stored Procedures

- Description
  - This module describes how to return results by executing stored procedures
  - Pass parameters to procedures
  - Create simple stored procedures that encapsulate a SELECT statement
  - Construct and execute dynamic SQL with EXEC and sp_executesql
- Lessons
  - Querying Data with Stored Procedures
  - Passing Parameters to Stored procedures
  - Creating Simple Stored Procedures
  - Working with Dynamic SQL
- Lab : Executing Stored Procedures
  - Using the EXECUTE statement to Invoke Stored Procedures
  - Passing Parameters to Stored procedures
  - Executing System Stored Procedures
- After completing this module, students will be able to:
  - Describe stored procedures and their use
  - Write T-SQL statements that execute stored procedures to return data
  - Write EXECUTE statements that pass input parameters to stored procedures
  - Write T-SQL batches that prepare output parameters and execute stored procedures
  - Use the CREATE PROCEDURE statement to write a stored procedure
  - Create a stored procedure that accepts input parameters
  - Describe how T-SQL can be dynamically constructed
  - Write queries that use dynamic SQL

Modalités

- Type d’action :Acquisition des connaissances
- Moyens de la formation :Formation présentielle – 1 poste par stagiaire – 1 vidéo projecteur – Support de cours fourni à chaque stagiaire
- Modalités pédagogiques :Exposés – Cas pratiques – Synthèse
- Validation :Exercices de validation – Attestation de stages