Exam 70-761: Querying Data with Transact-SQL – Skills Measured

Audience Profile

This exam is intended for SQL Server database administrators, system engineers, and developers with two or more years of experience who are seeking to validate their skills and knowledge in writing queries.

Skills Measured

NOTE: The bullets that appear below each of the skills measured are intended to illustrate how we are assessing that skill. This list is not definitive or exhaustive.

NOTE: In most cases, exams do NOT cover preview features, and some features will only be added to an exam when they are GA (General Availability).

Manage data with Transact-SQL (40–45%)

Create Transact-SQL SELECT queries

- identify proper SELECT query structure, write specific queries to satisfy business requirements, construct results from multiple queries using set operators, distinguish between UNION and UNION ALL behaviour, identify the query that would return expected results based on provided table structure and/or data

Query multiple tables by using joins

- write queries with join statements based on provided tables, data, and requirements; determine proper usage of INNER JOIN, LEFT/RIGHT/FULL OUTER JOIN, and CROSS JOIN; construct multiple JOIN operators using AND and OR; determine the correct results when presented with multi-table SELECT statements and source data; write queries with NULLs on joins

Implement functions and aggregate data

- construct queries using scalar-valued and table-valued functions; identify the impact of function usage to query performance and WHERE clause sargability; identify the differences between deterministic and non-deterministic functions; use built-in aggregate functions; use arithmetic functions, date-related functions, and system functions

Modify data

- write INSERT, UPDATE, and DELETE statements; determine which statements can be used to load data to a table based on its structure and constraints; construct Data Manipulation Language (DML) statements using the OUTPUT statement; determine the results of Data Definition Language (DDL) statements on supplied tables and data

Query data with advanced Transact-SQL components (30–35%)

Query data by using subqueries and APPLY

- determine the results of queries using subqueries and table joins, evaluate performance differences between table joins and correlated subqueries based on provided data and query plans, distinguish between the use of CROSS APPLY and OUTER APPLY, write APPLY statements that return a given data set based on supplied data

Query data by using table expressions

- identify basic components of table expressions, define usage differences between table expressions and temporary tables, construct recursive table expressions to meet business requirements

Group and pivot data by using queries

- use windowing functions to group and rank the results of a query; distinguish between using windowing functions and GROUP BY; construct complex GROUP BY clauses using GROUPING SETS, and CUBE; construct PIVOT and UNPIVOT statements to return desired results based on supplied data; determine the impact of NULL values in PIVOT and UNPIVOT queries

Query temporal data and non-relational data

- query historic data by using temporal tables, query and output JSON data, query and output XML data

Program databases by using Transact-SQL (25–30%)

Create database programmability objects by using Transact-SQL

- create stored procedures, table-valued and scalar-valued user-defined functions, triggers, and views; implement input and output parameters in stored procedures;

identify whether to use scalar-valued or table-valued functions; distinguish between deterministic and non-deterministic functions; create indexed views

Implement error handling and transactions

- determine results of Data Definition Language (DDL) statements based on transaction control statements, implement TRY…CATCH error handling with Transact-SQL, generate error messages with THROW and RAISERROR, implement transaction control in conjunction with error handling in stored procedures

Implement data types and NULLs

- evaluate results of data type conversions, determine proper data types for given data elements or table columns, identify locations of implicit data type conversions in queries, determine the correct results of joins and functions in the presence of NULL values, identify proper usage of ISNULL and COALESCE functions