tyler technologies Empowering people who serve the public™

Lab 7 – Using Table Expressions

OVERVIEW

In this lab, you will use views, temporary tables, variables, table-valued functions, derived tables, and common table expressions to retrieve data from the **AdventureWorksLT** database.

Before starting this lab, you should view **Module 7 – Using Table Expressions** in the Course *Querying with Transact-SQL*. Then, if you have not already done so, follow the instructions in the **Getting Started** document for this course to set up the lab environment.

If you find some of the challenges difficult, don't worry – you can find suggested solutions for all of the challenges in the **Lab Solution** folder for this module.

WHAT YOU'LL NEED

- An Azure SQL Database instance with the **AdventureWorksLT** sample database. Review the **Getting Started** document for information about how to provision this.

CHALLENGE 1: RETRIEVE PRODUCT INFORMATION

Adventure Works sells many products that are variants of the same product model. You must write queries that retrieve information about these products

1. Retrieve product model descriptions

Retrieve the product ID, product name, product model name, and product model summary for each product from the **SalesLT.Product** table and the **SalesLT.vProductModelCatalogDescription** view.

2. Create a table of distinct colors

Tip: Review the documentation for Variables in Transact-SQL Language Reference.

Create a table variable and populate it with a list of distinct colors from the **SalesLT.Product** table. Then use the table variable to filter a query that returns the product ID, name, and color from the **SalesLT.Product** table so that only products with a color listed in the table variable are returned.

3. Retrieve product parent categories

The **AdventureWorksLT** database includes a table-valued function named **dbo.ufnGetAllCategories**, which returns a table of product categories (for example 'Road Bikes') and parent categories (for example 'Bikes'). Write a query that uses this function to return a list of all products including their parent category and category.

CHALLENGE 2: RETRIEVE CUSTOMER SALES REVENUE

Each Adventure Works customer is a retail company with a named contact. You must create queries that return the total revenue for each customer, including the company and customer contact names.

Tip: Review the documentation for the WITH common_table_expression syntax in the Transact-SQL language reference.

1. Retrieve sales revenue by customer and contact

Retrieve a list of customers in the format *Company* (*Contact Name*) together with the total revenue for that customer. Use a derived table or a common table expression to retrieve the details for each sales order, and then query the derived table or CTE to aggregate and group the data.

NEXT STEPS

Well done! You've completed the lab, and you're ready to learn how to summarize data by specifying grouping sets and pivoting data in **Module 8 – Grouping Sets and Pivoting Data** in the Course *Querying with Transact-SQL*.