Querying with Transact-SQL

Lab 9 – Modifying Data

Overview

In this lab, you will insert, update, and delete data in the **AdventureWorksLT** database.

Before starting this lab, you should view **Module 9 – Modifying Data** 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: Inserting Products

Each Adventure Works product is stored in the **SalesLT.Product** table, and each product has a unique **ProductID** identifier, which is implemented as an IDENTITY column in the **SalesLT.Product** table. Products are organized into categories, which are defined in the **SalesLT.ProductCategory** table. The products and product category records are related by a common **ProductCategoryID** identifier, which is an IDENTITY column in the **SalesLT.ProductCategory** table.

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

1. Insert a product

Adventure Works has started selling the following new product. Insert it into the **SalesLT.Product** table, using default or NULL values for unspecified columns:

Name	ProductNumber	StandardCost	ListPrice	ProductCategoryID	SellStartDate
LED Lights	LT-L123	2.56	12.99	37	*<Today>*

After you have inserted the product, run a query to determine the **ProductID** that was generated. Then run a query to view the row for the product in the **SalesLT.Product** table.

2. Insert a new category with two products

Adventure Works is adding a product category for 'Bells and Horns' to its catalog. The parent category for the new category is 4 (Accessories). This new category includes the following two new products:

Name	ProductNumber	StandardCost	ListPrice	ProductCategoryID	SellStartDate
Bicycle Bell	BB-RING	2.47	4.99	*<The new ID for Bells and Horns>*	*<Today>*
Bicycle Horn	BB-PARP	1.29	3.75	*<The new ID for Bells and Horns>*	*<Today>*

Write a query to insert the new product category, and then insert the two new products with the appropriate **ProductCategoryID** value.

After you have inserted the products, query the **SalesLT.Product** and **SalesLT.ProductCategory** tables to verify that the data has been inserted.

Challenge 2: Updating Products

You have inserted data for a product, but the pricing details are not correct. You must now update the records you have previously inserted to reflect the correct pricing.

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

1. Update product prices

The sales manager at Adventure Works has mandated a 10% price increase for all products in the Bells and Horns category. Update the rows in the **SalesLT.Product** table for these products to increase their price by 10%.

2. Discontinue products

The new LED lights you inserted in the previous challenge are to replace all previous light products. Update the **SalesLT.Product** table to set the **DiscontinuedDate** to today's date for all products in the Lights category (Product Category ID 37) other than the LED Lights product you inserted previously.

Challenge 3: Deleting Products

The Bells and Horns category has not been successful, and it must be deleted from the database.

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

1. Delete a product category and its products

Delete the records foe the Bells and Horns category and its products. You must ensure that you delete the records from the tables in the correct order to avoid a foreign-key constraint violation.

Next Steps

Well done! You've completed the lab, and you're ready to learn how to implement procedural logic in Transact-SQL by completing **Module 10 – Programming with Transact-SQL** in the Course *Querying with Transact-SQL.*