tyler technologies Empowering people who serve the public™

SQL Scripting Basics – Lab Module 1

OVERVIEW

In this lab, you will use some basic SELECT queries to retrieve data from the **AdventureWorksLT** database.

Before starting this lab, you should view **Module 1 – Introduction** in the Course *Querying with TransactSQL*. 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 CUSTOMER DATA

Adventure Works Cycles sells directly to retailers, who then sell products to consumers. Each retailer that is an Adventure Works customer has provided a named contact for all communication from Adventure Works. The sales manager at Adventure Works has asked you to generate some reports containing details of the company's customers to support a direct sales campaign.

Tip: Review the documentation for the SELECT statement in the Transact-SQL Reference.

1. Retrieve customer details

Familiarize yourself with the Customer table by writing a Transact-SQL query that retrieves all columns for all customers.

2. Retrieve customer name data

Create a list of all customer contact names that includes the title, first name, middle name (if any), last name, and suffix (if any) of all customers.

3. Retrieve customer names and phone numbers

Each customer has an assigned salesperson. You must write a query to create a call sheet that lists:

- The salesperson
- A column named **CustomerName** that displays how the customer contact should be greeted (for example, "Mr Smith")
- The customer's phone number.

CHALLENGE 2: RETRIEVE CUSTOMER AND SALES DATA

As you continue to work with the Adventure Works customer data, you must create queries for reports that have been requested by the sales team.

Tip: Review the documentation for Conversion Functions in the Transact-SQL Reference.

1. Retrieve a list of customer companies

You have been asked to provide a list of all customer companies in the format <*Customer* ID> : <*Company Name*> - for example, **78: Preferred Bikes**.

2. Retrieve a list of sales order revisions

The **SalesLT.SalesOrderHeader** table contains records of sales orders. You have been asked to retrieve data for a report that shows:

- The sales order number and revision number in the format <*Order Number*> (<*Revision*>) – for example **SO71774 (2)**.
- The order date converted to ANSI standard format (yyyy.mm.dd – for example **2015.01.31**).

CHALLENGE 3: RETRIEVE CUSTOMER CONTACT DETAILS

Some records in the database include missing or unknown values that are returned as NULL. You must create some queries that handle these NULL fields appropriately.

Tip: Review the documentation for the ISNULL function and Expressions in the Transact-SQL Reference.

1. Retrieve customer contact names with middle names if known

You have been asked to write a query that returns a list of customer names. The list must consist of a single field in the format <*first name*> <*last name*> (for example **Keith Harris**) if the middle name is unknown, or <*first name*> <*middle name*> <*last name*> (for example **Jane M. Gates**) if a middle name is stored in the database.

2. Retrieve primary contact details

Customers may provide adventure Works with an email address, a phone number, or both. If an email address is available, then it should be used as the primary contact method; if not, then the phone number should be used. You must write a query that returns a list of customer IDs in one column, and a second column named **PrimaryContact** that contains the email address if known, and otherwise the phone number.

IMPORTANT: In the sample data provided in **AdventureWorksLT**, there are no customer records without an email address. Therefore, to verify that your query works as expected, run the following UPDATE statement to remove some existing email addresses before creating your query (don't worry, you'll learn about UPDATE statements later in the course).

```
UPDATE SalesLT.Customer
SET EmailAddress = NULL WHERE CustomerID % 7 = 1;
```

3. Retrieve shipping status

You have been asked to create a query that returns a list of sales order IDs and order dates with a

column named **ShippingStatus** that contains the text “Shipped” for orders with a known ship date, and “Awaiting Shipment” for orders with no ship date.

IMPORTANT: In the sample data provided in **AdventureWorksLT**, there are no sales order header records without a ship date. Therefore, to verify that your query works as expected, run the following UPDATE statement to remove some existing ship dates before creating your query (don’t worry, you’ll learn about UPDATE statements later in the course).

```
UPDATE SalesLT.SalesOrderHeader
SET ShipDate = NULL
WHERE SalesOrderID > 71899;
```