**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.

1. Retrieve customer details
   * Familiarize yourself with the **SalesLT.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 data for transportation reports**

The logistics manager at Adventure Works has asked you to generate some reports containing details of the company’s customers to help to reduce transportation costs.

1. Retrieve a list of cities
   * Initially, you need to produce a list of all of your customers’ locations. Write a Transact-SQL query that queries the **SalesLT.Address** table and retrieves the values for **City** and **StateProvince**, removing duplicates and sorted in ascending order of city.
2. Retrieve the heaviest products
   * Transportation costs are increasing and you need to identify the heaviest products. Retrieve the names of the top ten percent of products by weight.

**Challenge 3: Retrieve product data**

The Production Manager at Adventure Works would like you to create some reports listing details of the products that you sell.

1. Retrieve product details for product model 1
   * Initially, you need to find the names, colors, and sizes of the products with a product model ID 1.
2. Filter products by color and size
   * Retrieve the product number and name of the products that have a color of *black*, *red*, or *white* and a size of *S* or *M*.
3. Filter products by product number
   * Retrieve the product number, name, and list price of products whose product number begins *BK-*
4. Retrieve specific products by product number
   * Modify your previous query to retrieve the product number, name, and list price of products whose product number begins *BK-* followed by any character other than *R*, and ends with a *-* followed by any two numerals.

**Challenge 4: Generate invoice reports**

Adventure Works Cycles sells directly to retailers, who must be invoiced for their orders. You have been tasked with writing a query to generate a list of invoices to be sent to customers.

1. Retrieve customer orders
   * As an initial step towards generating the invoice report, write a query that returns the company name from the **SalesLT.Customer** table, and the sales order ID and total due from the **SalesLT.SalesOrderHeader** table.
2. Retrieve customer orders with addresses
   * Extend your customer orders query to include the *Main Office* address for each customer, including the full street address, city, state or province, postal code, and country or region
   * **Tip**: Note that each customer can have multiple addressees in the **SalesLT.Address** table, so the database developer has created the **SalesLT.CustomerAddress** table to enable a many-to-many relationship between customers and addresses. Your query will need to include both of these tables, and should filter the results so that only *Main Office* addresses are included.

**Challenge 5: Retrieve customer data**

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

1. Retrieve a list of all customers and their orders
   * The sales manager wants a list of all customer companies and their contacts (first name and last name), showing the sales order ID and total due for each order they have placed. Customers who have not placed any orders should be included at the bottom of the list with NULL values for the order ID and total due.
2. Retrieve a list of customers with no address
   * A sales employee has noticed that Adventure Works does not have address information for all customers. You must write a query that returns a list of customer IDs, company names, contact names (first name and last name), and phone numbers for customers with no address stored in the database.

**Challenge 6: Retrieve order shipping information**

The operations manager wants reports about order shipping based on data in the **SalesLT.SalesOrderHeader** table.

1. Retrieve the order ID and freight cost of each order.
   * Write a query to return the order ID for each order, together with the the **Freight** value rounded to two decimal places in a column named **FreightCost**.
2. Add the shipping method.
   * Extend your query to include a column named **ShippingMethod** that contains the **ShipMethod** field, formatted in lower case.
3. Add shipping date details.
   * Extend your query to include columns named **ShipYear**, **ShipMonth**, and **ShipDay** that contain the year, month, and day of the **ShipDate**. The **ShipMonth** value should be displayed as the month name (for example, *June*)

**Challenge 7: Aggregate product sales**

The sales manager would like reports that include aggregated information about product sales.

1. Retrieve total sales by product
   * Write a query to retrieve a list of the product names from the **SalesLT.Product** table and the total revenue for each product calculated as the sum of **LineTotal** from the **SalesLT.SalesOrderDetail** table, with the results sorted in descending order of total revenue.
2. Filter the product sales list to include only products that cost over 1,000
   * Modify the previous query to include sales totals for products that have a list price of more than 1000.
3. Filter the product sales groups to include only total sales over 20,000
   * Modify the previous query to only include only product groups with a total sales value greater than 20,000.