Project Documentation: Adventure Works Sales Analytics Report 2022

---

1. Project Title and Objective

- **Project Title:** Adventure Works Sales Analytics Report 2022
- **Objective:** The project was created to analyze sales performance, customer segmentation, product profitability, and market trends. The report enables the identification of key metrics and supports business decision-making.

---

2. Data Description

- **Data Sources:** The data is sourced from the Adventure Works 2022 database, which includes orders, product details, customer information, and territorial data.
- **Time Scope:** The data covers the period from January 1, 2011, to December 31, 2014.
- **Data Structure:** Key tables:
  - **SalesOrderHeader:** Information about orders.
  - **SalesOrderDetail:** Order details, including prices and quantities.
  - **Customer:** Customer data (e.g., customer type, location).
  - **Product:** Product details.
- **Data Preparation:** SQL views were created for individual tables to avoid data redundancy. Data integration for analysis was performed in Power BI.

---

3. Project Scope

- **Key Analytical Questions:**
  - Which customers generate the highest revenue?
  - Which products are the most profitable?
  - How have sales evolved over the years?
  - How many products generated no sales?
- **Analysis Areas:**
  - Sales
  - Customers
  - Products
  - Transaction Details

---

4. Report Features

- **"Sales" Tab:** Analysis of revenue, order volume, and year-over-year (YoY) sales growth. It includes a summary of top-selling products and top customers.
- **"Customers" Tab:** Segmentation of customers (loyal, inactive, new, occasional, individual customer, company) and analysis of top customers by revenue and order volume.

- **"Products" Tab:** Detailed analysis of product profitability, sales by category, and identification of products with no sales.
- **Interactivity:**
    - Data filtering by dates, regions, customers, and products.
    - "Top" value filtering (e.g., top customers, products with the highest margins).
    - Drill-through to detailed transaction data.
    - Tooltips providing additional information about the data and its limitations.

---

5. Methods and Tools

- **Technologies:**
    - Power BI: Report creation and visualization.
    - DAX: Custom measures and metrics.
    - SQL: Data preparation and view creation.
- **Analysis Methods:**
    - Year-over-Year (YoY) growth analysis.
    - Ranking of customers and products.
    - Customer segmentation based on purchasing activity.

---

6. Measures and Metrics

- **Sample DAX Measures:**
    - **YoY Sales Growth:** Year-over-year sales growth.
    - **LoyalCustomers:** Number of loyal customers.
    - **ChurnedCustomers:** Customers who stopped purchasing in the last year.
    - **CasualBuyers:** Occasional customers.
    - **NoSalesProductsList:** List of products with no sales.
    - **AverageOrder:** Average order value.

---

7. Conclusions and Recommendations

Conclusions:

- The highest revenue is generated by occasional customers, indicating the company's struggle to increase the number of loyal and new customers.
- Over the last recorded year, the number of orders decreased by 2,000, leading to a disproportionate revenue drop of 24 million. This was due to the fact that in 2013, orders included a higher number of sales units, which, despite lower margins, generated higher revenue.
- Australia sold the fewest products below the margin, possibly indicating less competition in the market or a better pricing strategy.
- The company achieves the highest margins in the Bicycles group, while Accessories generate the highest sales volume.
- Some of the most active customers stopped making purchases in May/June.

- **Recommendations:**
  - Increase loyal customer engagement through discounts and loyalty programs.
  - Promote less popular products with high margins.
  - Optimize the product portfolio to improve sales performance.

---

8. Challenges and Solutions

- **Challenges:**
  - Difficulty in filtering "Top\Flop" values in reports.
- **Solutions:**
  - Implementation of DAX measures enabling dynamic filtering.

---

9. Future Development Opportunities

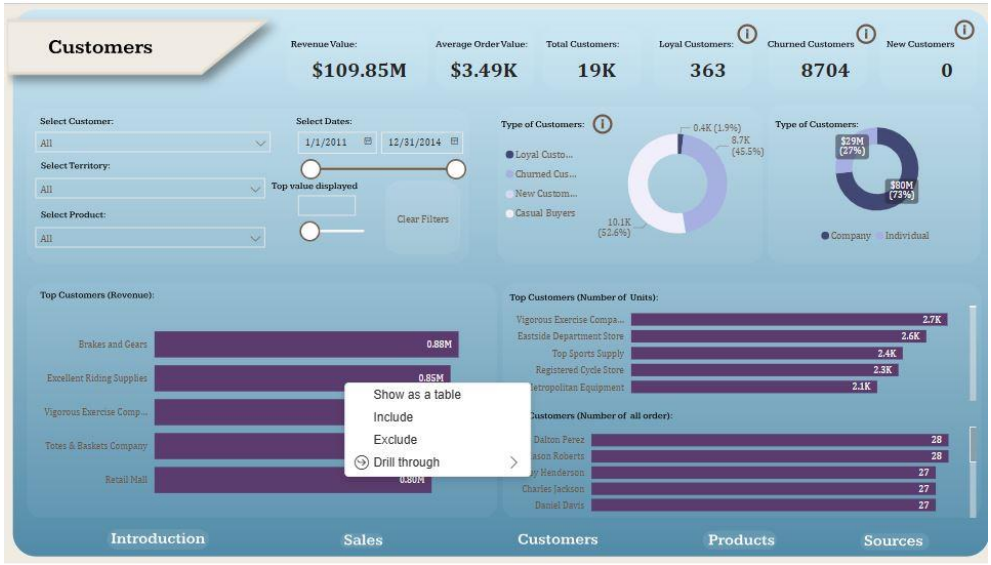
- Adding an analysis of logistics costs and profitability.
- Introducing sales forecasting using statistical modeling.
- Expanding the analysis scope to cover additional years.

---

10. Attachments

- **Screenshots:** Visualizations from the report (e.g., "Sales," "Customers," "Products" tabs).
- **DAX Code:** Example Measures Used in the Project.
- **Sample SQL Queries:** Queries used for data preparation.

Screenshots: Visualizations from the Report



List of Example DAX Measures Used in the Power BI Project:

The project utilized a variety of DAX measures that support the analysis of sales, customers, products, and growth dynamics. Below is a complete list of measures along with their descriptions.

Customer-Related Measures

1. **Casual Buyers**

- **Code:**

```
CasualBuyers =
COALESCE(
    CALCULATE(
        DISTINCTCOUNT('Customer'[CustomerID]),
        FILTER(
            'Customer',
            VAR FirstPurchaseDate = CALCULATE(MIN('SalesOrderHeader'[OrderDate]))
            VAR LastPurchaseDate = CALCULATE(MAX('SalesOrderHeader'[OrderDate]))
            VAR OrderCount = CALCULATE(COUNT('SalesOrderHeader'[SalesOrderID]))
            VAR IsChurned = LastPurchaseDate < MAX('Calendar'[Date]) - 365
            RETURN
                NOT IsChurned &&  -- Wyklucz klientów, którzy są churned
                LastPurchaseDate >= MAX('Calendar'[Date]) - 365 &&
                LastPurchaseDate <= MAX('Calendar'[Date]) &&
                FirstPurchaseDate < MAX('Calendar'[Date]) - 30 &&
                OrderCount < 5
        )
    ),
    0
)
```

Description: Occasional customers who placed fewer than 5 orders in the last year.

1. **Active Customers**
    - **Code**:

```
ActiveCustomers =
COALESCE(
            CALCULATE(
            DISTINCTCOUNT('SalesOrderHeader'[CustomerID]),
            FILTER(
                'SalesOrderHeader',
                'SalesOrderHeader'[OrderDate] <= MAX('Calendar'[Date])
            )
        ),
        0)
```

Description: The number of active customers within a given period.

2. **Loyal Customers**
   - **Code**:

```
LoyalCustomers =
COALESCE(
    CALCULATE(
        DISTINCTCOUNT('Customer'[CustomerID]),
        FILTER(
            'Customer',
            VAR FirstPurchaseDate = CALCULATE(MIN('SalesOrderHeader'[OrderDate]))
            VAR LastPurchaseDate = CALCULATE(MAX('SalesOrderHeader'[OrderDate]))
            VAR OrderCount = CALCULATE(COUNT('SalesOrderHeader'[SalesOrderID]))
            RETURN
                LastPurchaseDate >= MAX('Calendar'[Date]) - 365 &&
                LastPurchaseDate <= MAX('Calendar'[Date]) &&
                FirstPurchaseDate < MAX('Calendar'[Date]) - 30 &&
                OrderCount >= 5
        )
    ),
    0
)
```

Description: Loyal customers who placed at least 5 orders in the last year.

3. **New Customers**
   - **Code**:

```
NewCustomers =
COALESCE(
    CALCULATE(
        DISTINCTCOUNT('Customer'[CustomerID]),
        FILTER(
            'Customer',
            VAR FirstPurchaseDate = CALCULATE(MIN('SalesOrderHeader'[OrderDate]))
            RETURN
                FirstPurchaseDate >= MAX('Calendar'[Date]) - 30 &&
                FirstPurchaseDate <= MAX('Calendar'[Date])
        )
    ),
    0
)

```

Description: New customers who made their first purchase within the last 30 days.

Sales-Related Measures

5. **Total Revenue**
   - **Code**:

```
Total revenue = COALESCE(SUM(SalesOrderHeader[SubTotal]),0)
```

Description: Total revenue generated from sales.

6. **YoY Sales Growth**
   - **Code**:

```
YoY Sales Growth =
DIVIDE(
    [Total revenue] - CALCULATE( [Total revenue], DATEADD('Calendar'[Date], -1, YEAR)),
    CALCULATE( [Total revenue], DATEADD('Calendar'[Date], -1, YEAR))
)
```

Description: Year-over-year revenue growth expressed as a percentage.

Product-Related Measures

7. **No Sales Products Count**
   - **Code**:

```
NoSalesProductsList =
CALCULATE(
    CONCATENATEX(
        FILTER(
            ALL('Product'),
            CALCULATE([NumberOfUnits], KEEPFILTERS('Product'[ProductID])) = BLANK()
            || CALCULATE([NumberOfUnits], KEEPFILTERS('Product'[ProductID])) = 0
        ),
        'Product'[ProductName],
        ", "
    )
)
```

Description: The number of products with no recorded sales.

8. **TopN Filtered Sales Order Clients**
   - **Code**:

```
TopN Filtered SalesOrderClients =
IF(
    [Client Order Rank] <= 'Parametr'[Parametr – wartość],
     [NumberOfOrders],
    BLANK()
)
```

Description: The top customers based on the number of orders, limited by the parameter N.

Example SQL Queries

SQL queries were used to prepare the data for analysis.

Product View:

```sql
CREATE OR ALTER VIEW vw_Product AS
SELECT
    p.ProductID,
    ISNULL(p.ProductSubcategoryID, 0) AS ProductSubcategoryID,
    p.Name AS ProductName
FROM Production.Product p;
```

Subcategory View:

```sql
CREATE OR ALTER VIEW vw_ProductSubcategory AS
SELECT
    ProductSubcategoryID,
    ProductCategoryID,
    Name AS SubcategoryName
FROM Production.ProductSubcategory

UNION ALL

SELECT
    0 AS ProductSubcategoryID,
    0 AS ProductCategoryID,
    'Unassigned' AS SubcategoryName;
```

Category View:

```sql
CREATE OR ALTER VIEW vw_ProductCategory AS
SELECT
    ProductCategoryID,
    Name AS CategoryName
FROM Production.ProductCategory

UNION ALL

SELECT
    0 AS ProductCategoryID,
    'Unassigned' AS CategoryName;
```

Transaction Details View:

```sql
CREATE OR ALTER VIEW vw_SalesOrderDetail AS
SELECT
    sod.SalesOrderID,
    sod.ProductID,
    sod.OrderQty,
    sod.UnitPrice,
    sod.UnitPriceDiscount,
    sod.LineTotal,
    p.StandardCost,
    (sod.LineTotal - (sod.OrderQty * p.StandardCost)) AS Margin
FROM Sales.SalesOrderDetail sod
JOIN Production.Product p ON sod.ProductID = p.ProductID;
```

General Transaction View:

```sql
CREATE OR ALTER VIEW vw_SalesOrderHeader AS
SELECT
    SalesOrderID,
    CustomerID,
    TerritoryID,
    OrderDate,
    SubTotal
FROM Sales.SalesOrderHeader;
```