Data warehouses – Specification of business process

Gracjan Grzech
193579

Paweł Pstrągowski
193473

grupa 1 · informatyka · semestr 5 · rok akademicki 2024/2025

1. KPI

1.1. Increase annual ratios returns/sold.

To achieve the given goal, the company assumes annual return/sold ratio below 5%.

- Value Expression

```
WITH
-- Calculate the annual ratio of returns to sold products
MEMBER [Measures].[ReturnSoldRatio] AS
    'IIF([Measures].[ProductSales] > 0,
         [Measures].[Returns_] / [Measures].[ProductSales],
         NULL)'
```

- Goal Expression

```
-- Define the company's goal for the ratio (below 5%)
MEMBER [Measures].[ReturnSoldGoal] AS
    '0.05'
```

- Status Expression

```
-- Check whether the current ratio meets the goal
MEMBER [Measures].[ReturnSoldStatus] AS
    'IIF([Measures].[ReturnSoldRatio] <= [Measures].[ReturnSoldGoal],
         "On Target",
         "Above Target")'
```

- Trend Expression

```
-- Compare the ratio with the previous year to determine improvement or
deterioration
MEMBER [Measures].[ReturnSoldTrend] AS
    '([Measures].[ReturnSoldRatio], [Date].[Year].CURRENTMEMBER) -
     ([Measures].[ReturnSoldRatio], [Date].[Year].PREVMEMBER)'
```

```
SELECT
    {[Measures].[ReturnSoldRatio], [Measures].[ReturnSoldGoal],
     [Measures].[ReturnSoldStatus], [Measures].[ReturnSoldTrend]} ON COLUMNS,
    [Date].[Year].MEMBERS ON ROWS
FROM [Data Warehouse]
```

1.2. Increase of 8% of annual sale.

To achieve the given goal, the company assumes annual sale increase of 8%.

- Value Expression

```
WITH
-- Sum of ProductSales for the Year
MEMBER [Measures].[AnnualSales] AS
    'SUM([Date].[Year].CURRENTMEMBER, [Measures].[ProductSales])'
```

- Goal Expression

```
-- 8% Increase of Annual Sales
MEMBER [Measures].[SalesGoal] AS
    '[Measures].[AnnualSales] * 1.08'
```

- Status Expression

```
-- Whether Sales Meet the Goal
MEMBER [Measures].[Status] AS
    'IIF([Measures].[AnnualSales] >= [Measures].[SalesGoal], "On Track", "Below
Target")'
```

- Trend Expression

```
-- Change in Annual Sales Over Time
MEMBER [Measures].[SalesTrend] AS
    '([Measures].[AnnualSales], [Date].[Year].CURRENTMEMBER) -
     ([Measures].[AnnualSales], [Date].[Year].PREVMEMBER)'
```

```
SELECT
    {[Measures].[AnnualSales], [Measures].[SalesGoal], [Measures].[Status],
[Measures].[SalesTrend]} ON COLUMNS,
    [Date].[Year].MEMBERS ON ROWS
FROM [Data Warehouse]
```

2. MDX queries

2.1. Did the quality of any of building materials used in products released in the last year deteriorate?

```
WITH
-- Total number of products sold with each material in the current year

MEMBER [Measures].[CurrentYearTotalSold] AS
    'SUM(
       FILTER([Date].[Year].MEMBERS, [Date].[Year].CURRENTMEMBER IS [Measures].
[2024]),
        [Measures].[ProductSales]
    )'

-- Total number of returns for each material in the current year
MEMBER [Measures].[CurrentYearTotalReturns] AS
    'SUM(
        FILTER([Date].[Year].MEMBERS, [Date].[Year].CURRENTMEMBER IS [2024]),
```

```
        [Measures].[Returns_]
    )'

-- Return rate for the current year
MEMBER [Measures].[CurrentYearReturnRate] AS
    'IIF([Measures].[CurrentYearTotalSold] > 0,
                        [Measures].[CurrentYearTotalReturns]   /   [Measures].
[CurrentYearTotalSold],
         NULL)'

-- Average repair cost for the current year
MEMBER [Measures].[CurrentYearAvgRepairCost] AS
    'IIF([Measures].[CurrentYearTotalReturns] > 0,
         SUM([Measures].[CompanyCost], [Date].[Year].CURRENTMEMBER) /
         [Measures].[CurrentYearTotalReturns],
         NULL)'

-- Same metrics for the previous year
MEMBER [Measures].[PreviousYearTotalSold] AS
    'SUM(
        FILTER([Date].[Year].MEMBERS, [Date].[Year].CURRENTMEMBER IS [2023]),
        [Measures].[ProductSales]
    )'

MEMBER [Measures].[PreviousYearTotalReturns] AS
    'SUM(
        FILTER([Date].[Year].MEMBERS, [Date].[Year].CURRENTMEMBER IS [2023]),
        [Measures].[Returns_]
    )'

MEMBER [Measures].[PreviousYearReturnRate] AS
    'IIF([Measures].[PreviousYearTotalSold] > 0,
                       [Measures].[PreviousYearTotalReturns]   /   [Measures].
[PreviousYearTotalSold],
         NULL)'

MEMBER [Measures].[PreviousYearAvgRepairCost] AS
    'IIF([Measures].[PreviousYearTotalReturns] > 0,
         SUM([Measures].[CompanyCost], [Date].[Year].PREVMEMBER) /
         [Measures].[PreviousYearTotalReturns],
         NULL)'

-- Change in return rate between years
MEMBER [Measures].[ReturnRateChange] AS
    '[Measures].[CurrentYearReturnRate] - [Measures].[PreviousYearReturnRate]'

-- Change in average repair cost between years
MEMBER [Measures].[AvgRepairCostChange] AS
                   '[Measures].[CurrentYearAvgRepairCost]     -     [Measures].
[PreviousYearAvgRepairCost]'

SELECT
```

```
    {[Measures].[CurrentYearReturnRate], [Measures].[PreviousYearReturnRate],
     [Measures].[ReturnRateChange], [Measures].[CurrentYearAvgRepairCost],
     [Measures].[PreviousYearAvgRepairCost], [Measures].[AvgRepairCostChange]}
ON COLUMNS,
    [Material].[MaterialType].MEMBERS ON ROWS
FROM [Data Warehouse]
WHERE [Date].[Year].[2024]
```

2.2. What is the average amount of time a product lasts for (how long it takes before a return is issued)?

```
WITH
-- Calculate the time difference for each returned product
MEMBER [Measures].[TimeBeforeReturn] AS
                'IIF([Measures].[Returns_]    >    0    AND    [Returns].
[wasReturnedOnWarranty].CURRENTMEMBER IS [Yes],
         DATEDIFF("d",
                  [Measures].[ProductSales.Date],
                  [Measures].[Returns.ProcessingStarted]),
         NULL)'

-- Average the time difference for all returned products
MEMBER [Measures].[AvgTimeBeforeReturn] AS
    'AVG(
        FILTER([Products].[Products].MEMBERS,
               [Returns].[wasReturnedOnWarranty].CURRENTMEMBER IS [Yes]),
        [Measures].[TimeBeforeReturn]
    )'

SELECT
    {[Measures].[AvgTimeBeforeReturn]} ON COLUMNS,
    [Products].[Products].MEMBERS ON ROWS
FROM [Data Warehouse];
```

2.3. Which products that were returned this year, broke down before their warranty ran out?

```
WITH
-- Calculate the time difference between sale date and return date for each
returned product
MEMBER [Measures].[TimeBeforeReturn] AS
    'DATEDIFF(
        [Measures].[ProductSales.Date],
        [Measures].[Returns_.ProcessingStartedDate]
    )'

-- Check if the product broke down before its warranty expired
MEMBER [Measures].[BrokeBeforeWarranty] AS
    'IIF(
                           [Measures].[TimeBeforeReturn]   <=   [Measures].
[ProductCatalogue.WarrantyPeriod],
        1,
        0
```

```
    )'

-- Total count of products returned within warranty this year

MEMBER [Measures].[TotalBrokenWithinWarranty] AS
    'SUM(
        FILTER(
            [Products].[Products Id].MEMBERS,
            [Measures].[BrokeBeforeWarranty] = 1 AND
            [Date].[Year].CURRENTMEMBER IS YEAR(NOW())
        ),
        [Measures].[Returns_]
    )'

SELECT
    {[Measures].[TimeBeforeReturn],
     [Measures].[BrokeBeforeWarranty],
     [Measures].[TotalBrokenWithinWarranty]} ON COLUMNS,
    [Products].[Products Id].MEMBERS ON ROWS
FROM [Data Warehouse]
WHERE [Date].[Year].[2024]
```

2.4. Compare the number of returns in prior years.

```
WITH
-- Count of returns for the current year
MEMBER [Measures].[CurrentYearReturns] AS
    '([Measures].[Returns_], [Time].[Year].CURRENTMEMBER)'

-- Count of returns for the previous year
MEMBER [Measures].[PreviousYearReturns] AS
    '([Measures].[Returns_], [Time].[Year].PREVMEMBER)'

-- Difference in the number of returns between current and previous year
MEMBER [Measures].[ReturnDifference] AS
    '[Measures].[CurrentYearReturns] - [Measures].[PreviousYearReturns]'

-- Percentage change in the number of returns
MEMBER [Measures].[ReturnPercentageChange] AS
    'IIF([Measures].[PreviousYearReturns] > 0,
         ([Measures].[ReturnDifference] / [Measures].[PreviousYearReturns])
* 100,
         NULL)'

SELECT
    {[Measures].[CurrentYearReturns], [Measures].[PreviousYearReturns],
     [Measures].[ReturnDifference], [Measures].[ReturnPercentageChange]} ON
COLUMNS,
    [Date].[Year].MEMBERS ON ROWS
FROM [Data Warehouse]
```

2.5. Calculate the average cost of a return this year and compare it with previous years.

```
WITH
-- Average cost of a return for the current year
MEMBER [Measures].[CurrentYearAvgReturnCost] AS
    'IIF([Measures].[Returns_] > 0,
        SUM([Measures].[CompanyCost], [Date].[Year].CURRENTMEMBER) / [Measures].
[Returns_],
         NULL)'

-- Average cost of a return for the previous year
MEMBER [Measures].[PreviousYearAvgReturnCost] AS
    'IIF([Measures].[Returns_] > 0,
         SUM([Measures].[CompanyCost], [Date].[Year].PREVMEMBER) /
         ([Measures].[Returns_], [Date].[Year].PREVMEMBER),
         NULL)'

-- Difference in the average cost of a return
MEMBER [Measures].[AvgReturnCostDifference] AS
                   '[Measures].[CurrentYearAvgReturnCost]     -     [Measures].
[PreviousYearAvgReturnCost]'

-- Percentage change in the average cost of a return
MEMBER [Measures].[AvgReturnCostPercentageChange] AS
    'IIF([Measures].[PreviousYearAvgReturnCost] > 0,
                        ([Measures].[AvgReturnCostDifference]   /   [Measures].
[PreviousYearAvgReturnCost]) * 100,
         NULL)'

SELECT
                       {[Measures].[CurrentYearAvgReturnCost],      [Measures].
[PreviousYearAvgReturnCost],
                         [Measures].[AvgReturnCostDifference],     [Measures].
[AvgReturnCostPercentageChange]} ON COLUMNS,
    [Date].[Year].MEMBERS ON ROWS
FROM [Data Warehouse]
```

2.6. Which type of build material incurs the highest repair bill.

```
WITH
-- Total repair cost for each MaterialType
MEMBER [Measures].[TotalRepairCostByMaterial] AS
    'SUM(
        [Material].[MaterialType].MEMBERS,
        [Measures].[CompanyCost]
    )'

-- Identify the MaterialType with the highest repair cost
MEMBER [Measures].[HighestRepairMaterial] AS
    'TOPCOUNT(
        [Material].[MaterialType].MEMBERS,
        1,
```

```
        [Measures].[TotalRepairCostByMaterial]
    ).ITEM(0).NAME'

-- Retrieve the highest repair cost
MEMBER [Measures].[HighestRepairCost] AS
    'MAX(
        [Material].[MaterialType].MEMBERS,
        [Measures].[TotalRepairCostByMaterial]
    )'

SELECT
    {[Measures].[TotalRepairCostByMaterial],
     [Measures].[HighestRepairMaterial],
     [Measures].[HighestRepairCost]} ON COLUMNS,
    [Material].[MaterialType].MEMBERS ON ROWS
FROM [Data Warehouse]
```

2.7. What percent of annual return requests are denied?

```
WITH
-- Count of denied return requests for the current year
MEMBER [Measures].[DeniedReturns] AS
    'SUM(
        FILTER([Junk].[Status_].MEMBERS, [Junk].[Status_].CURRENTMEMBER IS
[Denied]),
        [Measures].[Returns_]
    )'

-- Percentage of return requests that are denied
MEMBER [Measures].[DeniedPercentage] AS
    'IIF([Measures].[Returns_] > 0,
         ([Measures].[DeniedReturns] / [Measures].[Returns_]) * 100,
         NULL)'

SELECT
        {[Measures].[Returns_],  [Measures].[DeniedReturns],  [Measures].
[DeniedPercentage]} ON COLUMNS,
    [Date].[Year].MEMBERS ON ROWS
FROM [Data Warehouse]
```

2.8. Compare the ratio of returned/sold products.

```
WITH
-- Total returned products
MEMBER [Measures].[TotalReturns] AS
    'SUM([Date].[Year].CURRENTMEMBER, [Measures].[Returns_])'

-- Total sold products
MEMBER [Measures].[TotalSold] AS
    'SUM([Date].[Year].CURRENTMEMBER, [Measures].[ProductSales])'

-- Ratio of returned to sold products
MEMBER [Measures].[ReturnToSoldRatio] AS
```

```
    'IIF([Measures].[TotalSold] > 0,
         [Measures].[TotalReturns] / [Measures].[TotalSold],
         NULL)'

SELECT
    {[Measures].[TotalReturns], [Measures].[TotalSold], [Measures].
[ReturnToSoldRatio]} ON COLUMNS,
    [Date].[Year].MEMBERS ON ROWS
FROM [Data Warehouse]
```

2.9. Check if there exists such product, whose repair was more expensive than issuing a new unit.

```
WITH
-- Check if any product's repair cost exceeds its price
MEMBER [Measures].[RepairMoreExpensive] AS
    'IIF(
        MAX(
            [Products].[Products Id].MEMBERS,
            [Measures].[CompanyCost] - [Measures].[Price]
        ) > 0,
        "Yes",
        "No"
    )'

SELECT
    {[Measures].[RepairMoreExpensive]} ON COLUMNS,
    {} ON ROWS
FROM [Data Warehouse]
```

2.10. Check whether there has been a large number of returns of the same products in a short period of time (potentially defective batch of goods).

```
WITH
-- Count the number of returns per product within the current year
MEMBER [Measures].[AnnualReturnCount] AS
    'SUM(
        [Date].[Day].MEMBERS,
        [Measures].[Returns_]
    )'

-- Threshold for a "large number of returns" (arbitrarily set to 10)
MEMBER [Measures].[LargeReturnThreshold] AS 10

-- Check if any product exceeds the return threshold
MEMBER [Measures].[DefectiveBatch] AS
    'IIF(
        MAX(
            [Products].[Products Id].MEMBERS,
            [Measures].[AnnualReturnCount]
        ) > [Measures].[LargeReturnThreshold],
```

```
        "Yes",
        "No"
    )'

SELECT
    {[Measures].[AnnualReturnCount], [Measures].[DefectiveBatch]} ON COLUMNS,
    [Products].[PProducts Id].MEMBERS ON ROWS
FROM [Data Warehouse]
WHERE [Date].[Year].CURRENTMEMBER
```

2.11. Show 3 best selling products.

```
WITH
-- Rank products based on their total sales
MEMBER [Measures].[TotalSales] AS
    'SUM([Measures].[ProductSales])'

-- Select the top 3 best-selling products
MEMBER [Measures].[Top3BestSellers] AS
    'TOPCOUNT(
        [Products].[Products Id].MEMBERS,
        3,
        [Measures].[TotalSales]
    ).ITEM(0).NAME'

-- Retrieve the total sales for the top 3 best-selling products
SELECT
    {[Measures].[TotalSales],
     [Measures].[Top3BestSellers]} ON COLUMNS,
    [Products].[Products Id].MEMBERS ON ROWS
FROM [Data Warehouse]
```

2.12. Show the number of returns in each year.

```
WITH
-- Total number of returns for each year
MEMBER [Measures].[TotalReturns] AS
    'SUM([Measures].[Returns_])'

SELECT
    {[Measures].[TotalReturns]} ON COLUMNS,
    [Date].[Year].MEMBERS ON ROWS
FROM [Data Warehouse]
```