

DSS – THIRD ASSIGNMENT

Domenico Sacco

Thiago da Silva Alves

REPORT INDEX

1.SALES INFORMATION.....	2
1.1 BEST CUSTOMERS.....	2
1.2 BEST PRODUCTS	2
1.3 BEST STORES	2
2.COMPANY PROFITS THROUGH TIME.....	3
2.1 NET REVENUES THROUGH TIME.....	3
2.2 NET REVENUES IN TRIMESTERS	3
2.3 ORDER QUANTITY THROUGH TIME	4
3.DATA MINING STRUCTURES	4
4. PREDICTIONS	4
4.1 PROFITS PREDICTION	4
4.2 HIGHEST DEMAND PREDICTION	4

1.SALES INFORMATIONS

1.1 BEST CUSTOMERS

We define the best clients as those generating the highest revenues for AdventureWorks.

The list of the top 10 customers (excluding store owners) was obtained with the following SQL query:

```
select top 10 cs.CustomerID, cs.Name, sum(sd.LineTotal - sd.StandardCostHistory) as total
from Sales.SalesOrderDetail as sd
inner join sales.Customer as cs on sd.FK_Customer = cs.CustomerID
where sd.FK_Store = 0
group by cs.CustomerID, cs.Name
order by sum(sd.LineTotal - sd.StandardCostHistory) desc
```

The generated report BestClients.rdl contains all informations about the best customers

1.2 BEST PRODUCTS

Likewise, we define the best products as those generating the highest earnings for the company, the best 20 were obtained with the following SQL query:

```
select top 20 pr.ProductID, pr.ProductName, pr.ModelName, sum(sd.LineTotal - sd.StandardCostHistory) as total
from Sales.SalesOrderDetail as sd
inner join Production.Product as pr on sd.FK_ProductID = pr.ProductID
group by pr.ProductID, pr.ProductName, pr.ModelName
order by sum(sd.LineTotal - sd.StandardCostHistory) desc
```

The generated report BestProducts.rdl contains all informations about the best customers

1.3 BEST STORES

We define the best customer stores as those where highest revenues were generated.

The SQL query for the top 20 stores follows:

```
select top 20 st.BusinessEntityID, st.Name, sum(sd.LineTotal - sd.StandardCostHistory) as total
from Sales.SalesOrderDetail as sd
inner join sales.Store as st on sd.FK_Store = st.BusinessEntityID
where st.BusinessEntityID != 0
group by st.BusinessEntityID, st.Name
order by sum(sd.LineTotal - sd.StandardCostHistory) desc
```

The generated report BestStores.rdl contains all informations about the best stores

2.COMPANY PROFITS THROUGH TIME

2.1 NET REVENUES THROUGH TIME

The following MDX query shows net revenues for each month of activity:

WITH

MEMBER [Measures].Revenues

AS

[Measures].[Line Total] -[Measures].[Standard Cost History],

FORMAT_STRING = "Currency",

*FORE_COLOR = 32768 /*Green*/,*

VISIBLE = 1

SELECT NON EMPTY { [Measures].[Revenues] } ON COLUMNS, NON EMPTY { ([Time].[Year].[Year].ALLMEMBERS,[Time].[Month].[Month].ALLMEMBERS) } DIMENSION PROPERTIES MEMBER_CAPTION, MEMBER_UNIQUE_NAME ON ROWS FROM [Our Work] CELL PROPERTIES VALUE, BACK_COLOR, FORE_COLOR, FORMATTED_VALUE, FORMAT_STRING, FONT_NAME, FONT_SIZE, FONT_FLAGS

Further informations can be found on the generated report RevenuesPerYear.rdl

2.2 NET REVENUES IN TRIMESTERS

With this MDX query, we determined company revenues for the first, the second and the third trimester, comparing them between the years:

WITH

MEMBER [Measures].Revenues

AS

[Measures].[Line Total] -[Measures].[Standard Cost History],

FORMAT_STRING = "Currency",

*FORE_COLOR = 32768 /*Green*/,*

VISIBLE = 1

SELECT NON EMPTY { [Measures].[Revenues] } ON COLUMNS, NON EMPTY { ([Time].[Year].[Year].ALLMEMBERS) } DIMENSION PROPERTIES MEMBER_CAPTION, MEMBER_UNIQUE_NAME ON ROWS FROM [Our Work] WHERE ([Time].[Trimester Of Year].&[1]) CELL PROPERTIES VALUE, BACK_COLOR, FORE_COLOR, FORMATTED_VALUE, FORMAT_STRING, FONT_NAME, FONT_SIZE, FONT_FLAGS

SELECT NON EMPTY { [Measures].[Revenues] } ON COLUMNS, NON EMPTY { ([Time].[Year].[Year].ALLMEMBERS) } DIMENSION PROPERTIES MEMBER_CAPTION, MEMBER_UNIQUE_NAME ON ROWS FROM [Our Work] WHERE ([Time].[Trimester Of Year].&[2]) CELL PROPERTIES VALUE, BACK_COLOR, FORE_COLOR, FORMATTED_VALUE, FORMAT_STRING, FONT_NAME, FONT_SIZE, FONT_FLAGS

SELECT NON EMPTY { [Measures].[Revenues] } ON COLUMNS, NON EMPTY { ([Time].[Year].[Year].ALLMEMBERS) } DIMENSION PROPERTIES MEMBER_CAPTION, MEMBER_UNIQUE_NAME ON ROWS FROM [Our Work] WHERE ([Time].[Trimester Of Year].&[3]) CELL PROPERTIES VALUE, BACK_COLOR, FORE_COLOR, FORMATTED_VALUE, FORMAT_STRING, FONT_NAME, FONT_SIZE, FONT_FLAGS

Further informations can be found on the generated report ByTrimester.rdl

2.3 ORDER QUANTITY THROUGH TIME

Using the following MDX query, company total profits for each month of the year were calculated:

```
SELECT NON EMPTY { [Measures].[Order Qty] } ON COLUMNS, NON EMPTY { ([Time].[Month].[Month].ALLMEMBERS * [Time].[Year - Trimester - Month - Date].[Year].ALLMEMBERS ) } DIMENSION PROPERTIES MEMBER_CAPTION, MEMBER_UNIQUE_NAME ON ROWS FROM [Our Work] CELL PROPERTIES VALUE, BACK_COLOR, FORE_COLOR, FORMATTED_VALUE, FORMAT_STRING, FONT_NAME, FONT_SIZE, FONT_FLAGS
```

Further informations can be found on the generated report QtyByYear.rdl

3.DATA MINING STRUCTURES

Different mining structures were created to accomplish the assigned tasks:

- vPredictOrder.dmm

It is a time series to predict the total sales quantity in the next three months of the products having the highest demand.

- PredictProfits.dmm

The purpose of this mining structure is to predict the profits of the top products in the next three months.

4. PREDICTIONS

4.1 PROFITS PREDICTION

Using the informations available in the DataWarehouse, profits of the top products in the next three months were predicted.

Informations can be found in the report named ThreeMonthsProfit.rdl

4.2 HIGHEST DEMAND PREDICTION

Also, products that will have the highest demand in the next three months have been predicted.

Informations can be found in the report named ThreeMonths.rdl