

6. sprawozdanie z laboratorium Hurtownie Danych

Mikołaj Kubś, 272662

4 maja 2025

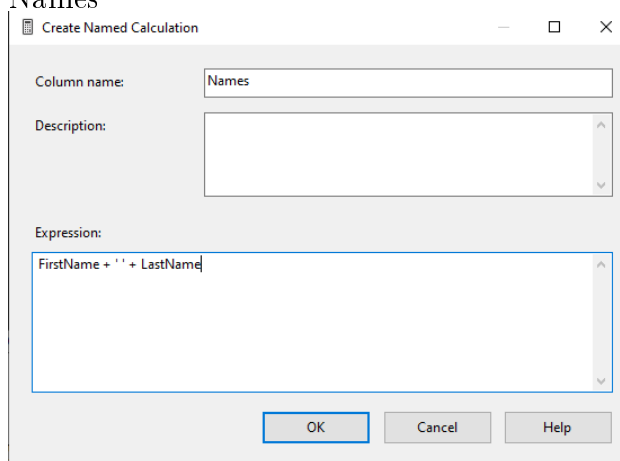
1 Zad. 1. Modyfikacja wymiarów i tabeli faktów

Bazując na kostce utworzonej przy realizacji listy 4, należy:

1.1 Podpunkt a

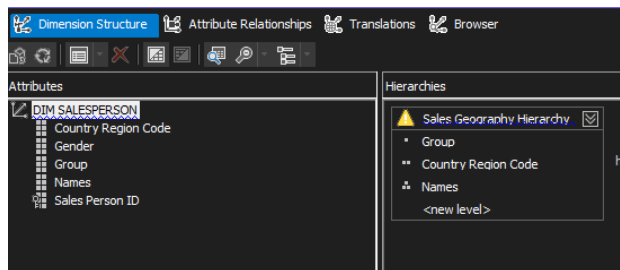
Zmodyfikować definicję wymiarów tak, aby:

1. W wymiarach CUSTOMER i SALESPERSON nie można było korzystać z atrybutów FirstName oraz LastName. W zamian dodać atrybut Names

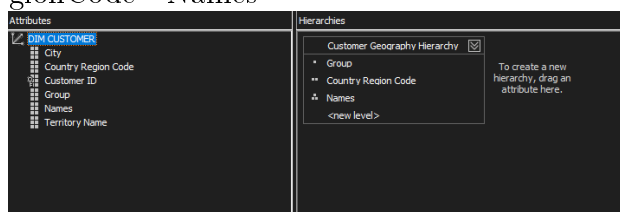


The screenshot shows a 'Create Named Calculation' dialog box. It has three main input fields: 'Column name' with the value 'Names', 'Description' which is empty, and 'Expression' with the formula 'FirstName + '' + LastName'. At the bottom, there are three buttons: 'OK', 'Cancel', and 'Help'.

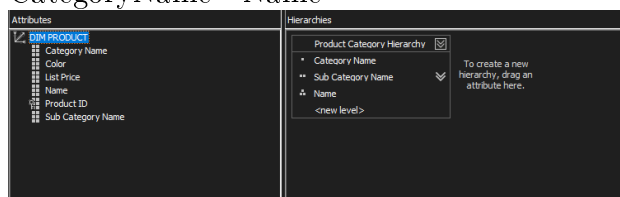
2. W wymiarze SALESPERSON pojawiła się hierarchia Group - CountryRegionCode - Names



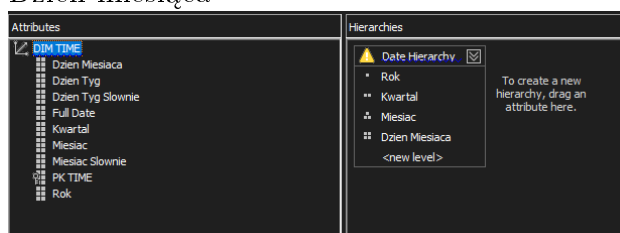
3. W wymiarze CUSTOMER pojawiła się hierarchia Group - CountryRegionCode - Names



4. W wymiarze PRODUCT pojawiła się hierarchia CategoryName - Sub-CategoryName - Name



5. W wymiarze TIME pojawiła się hierarchia Rok - Kwartał - Miesiąc - Dzień miesiąca



1.2 Podpunkt b

Dla każdego atrybutu kluczowego wymiaru, którego wartościami są liczby całkowite, zmodyfikować właściwości (Properties). Zmodyfikować parametr NameColumn, tak aby nazwy kolejnych elementów wymiaru nie były liczbami. (Przykładowo dla wymiaru dotyczącego Produktu można wykorzystać atrybut Name).

Source	
CustomRollupColumn	(none)
CustomRollupPropertiesColumn	(none)
KeyColumns	DIM_CUSTOMER.CustomerID (Integer)
NameColumn	DIM_CUSTOMER.Names (WChar)
ValueColumn	(none)

Rysunek 1: Widok Properties dla DIM_Salesperson

Source	
CustomRollupColumn	(none)
CustomRollupPropertiesColumn	(none)
KeyColumns	DIM_CUSTOMER.CustomerID (Integer)
NameColumn	DIM_CUSTOMER.Names (WChar)
ValueColumn	(none)

Rysunek 2: Widok Properties dla DIM_Customer

Source	
CustomRollupColumn	(none)
CustomRollupPropertiesColumn	(none)
KeyColumns	DIM_PRODUCT.ProductID (Integer)
NameColumn	DIM_PRODUCT.Name (WChar)
ValueColumn	(none)

Rysunek 3: Widok Properties dla DIM_Product

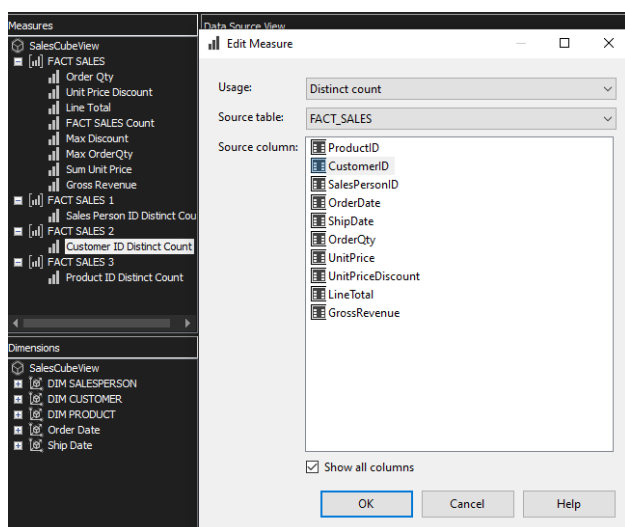
Source	
CustomRollupColumn	(none)
CustomRollupPropertiesColumn	(none)
KeyColumns	DIM_TIME.PK_TIME (Integer)
NameColumn	DIM_TIME.FullDate (WChar)
ValueColumn	(none)

Rysunek 4: Widok Properties dla DIM_Time

1.3 Podpunkt c

Utworzyć nowe miary, które będą odzwierciedlać:

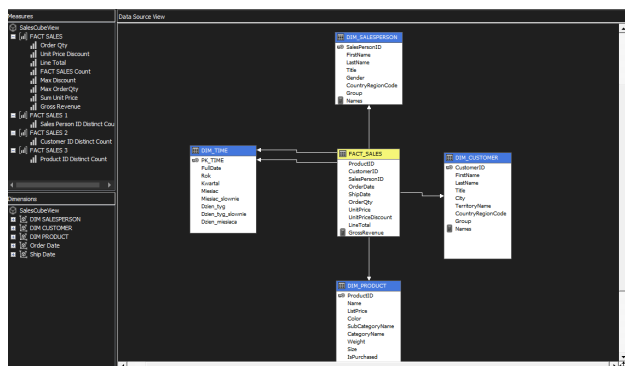
- Liczbę różnych klientów (aggregatedFunction: distinct count)
- Liczbę różnych produktów
- Maksymalną wartość rabatu (aggregatedFunction: max)
- Maksymalną liczbę zamówionych produktów
- Liczbę różnych sprzedawców realizujących zamówienia



Rysunek 5: Miara dotycząca liczby różnych klientów

1.4 Podpunkt d

Wdrożyć i przetworzyć kostkę.



Rysunek 6: Widok przetworzonej kostki

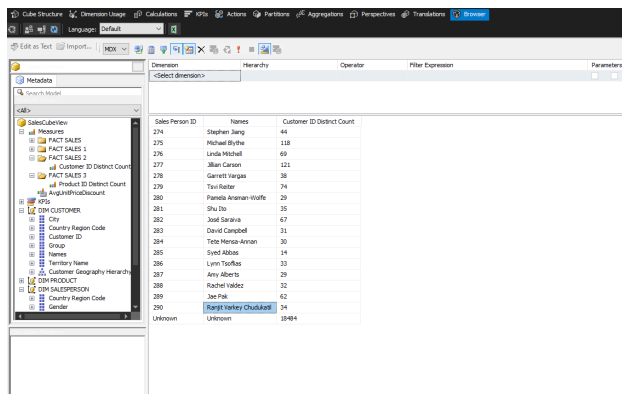
2 Zad. 2. Przegląd danych i tworzenie zestawień

Przy użyciu zakładki Browser:

2.1 Podpunkt a

Sprawdzić, czy dane zapisane w kostce zgadzają się z danymi zapisanymi w tabelach, przeciągając za pomocą myszy:

- atrybuty wymiarów w region wierszy
- miary w część centralną widoku

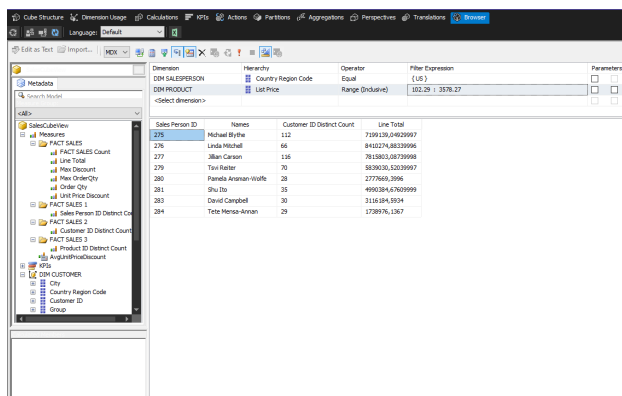


Sales Person ID	Names	Customer ID District Count
274	Stephen Jung	44
275	Michael Blythe	118
276	Linda Mitchell	69
277	Sellan Carson	121
278	Garnett Vargas	38
279	Toni Kaler	74
280	Pamela Ansman-Wolfe	29
281	Shu-Ita	35
282	Joel Saraisa	67
283	David Campbell	71
284	Tate Henss-Arman	30
285	Syed Abbas	14
286	Lynn Tsofas	33
287	Amy Alberts	29
288	Rachel Valdez	32
289	Jan-Pi	42
290	Rajesh Kulkarni	24
Unknown	Unknown	18484

Rysunek 7: Widok przykładowej kwerendy w Browser

2.2 Podpunkt b

Przetestować możliwości przeglądarki (Browser) - operator wyboru danych (Operator), wyrażenia filtrujące dane (Filter Expression) itp.



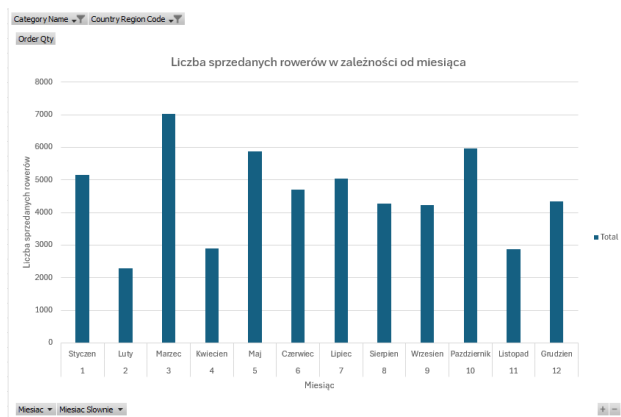
Sales Person ID	Names	Customer ID District Count	Line Total
275	Michael Blythe	112	7199126,04229997
276	Linda Mitchell	66	8402274,38239996
277	Sellan Carson	125	7615852,08739995
279	Toni Kaler	70	5839626,52339997
280	Pamela Ansman-Wolfe	28	2777669,3996
281	Shu-Ita	35	4990284,67620999
283	David Campbell	30	3116184,9524
284	Tate Henss-Arman	29	1738976,1367

Rysunek 8: Widok przykładowej kwerendy z dwoma różnymi rodzajami fil-trów (Operator i Filter Expression)

2.3 Podpunkt c

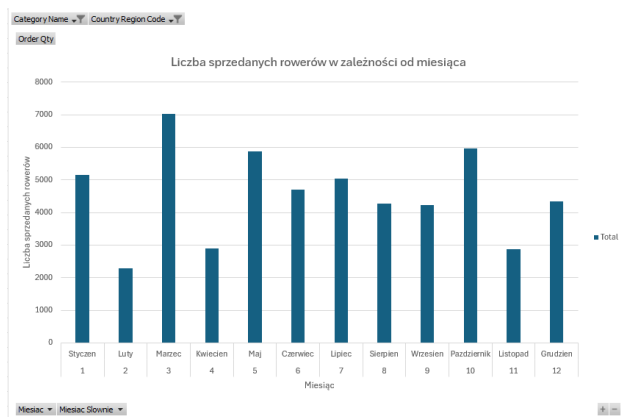
Przygotować przykładowe tabele i wykresy przestawne oraz zinterpretować uzyskane wyniki (proszę zapisać wnioski!)

2.3.1 Rowery



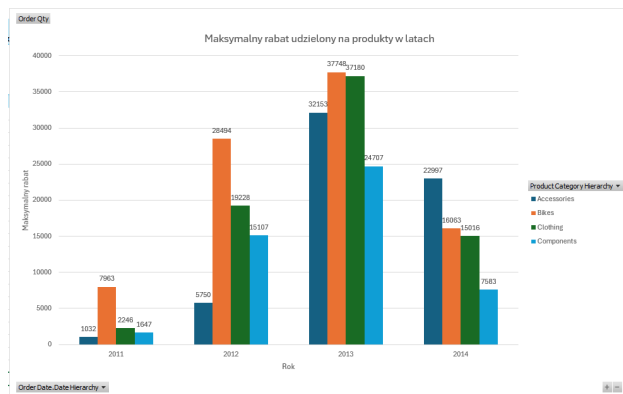
Rysunek 9: Wykres

2.3.2 Rowery



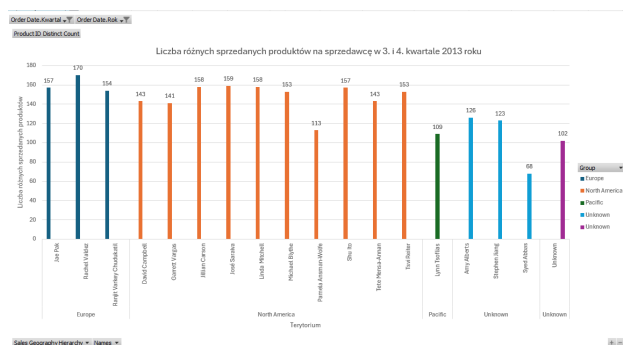
Rysunek 10: Wykres

2.3.3 Zniżka



Rysunek 11: Wykres

2.3.4 Sprzedawca



Rysunek 12: Wykres

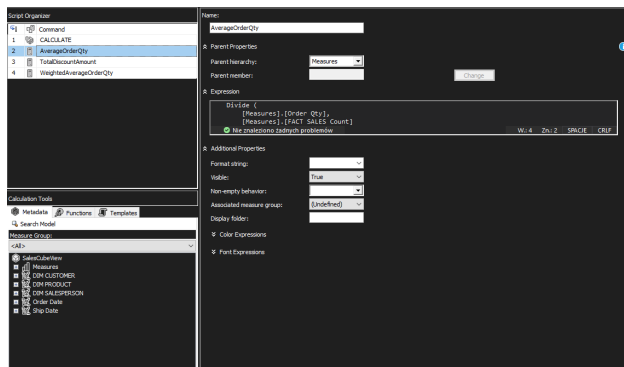
3 Zad. 3. Miary kalkulowane

W zakładce Calculations dodać dwie miary kalkulowane (ang. calculated members):

- średnią liczbę zamówionych towarów na zamówienie
- średnią ważoną liczbę towarów na zamówienie. Jako wagę należy wybrać cenę danego produktu.

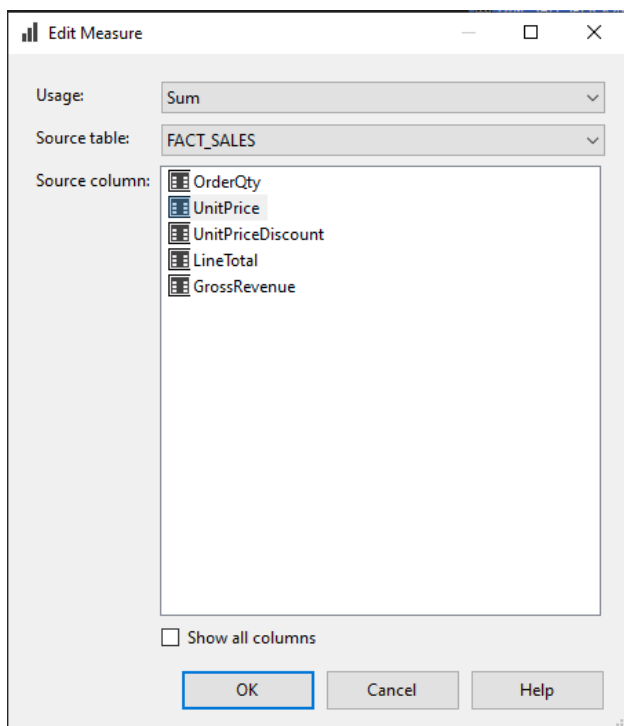
Wskazówka: w celu utworzenia wyżej wymienionej średniej ważonej można posłużyć się nową kolumną zdefiniowaną w widoku źródła danych (lub w

tabeli). Kolumna ta powinna definiować miarę pomocniczą, która pozwoli uzyskać fragment wyrażenia odpowiadającego średniej ważonej.

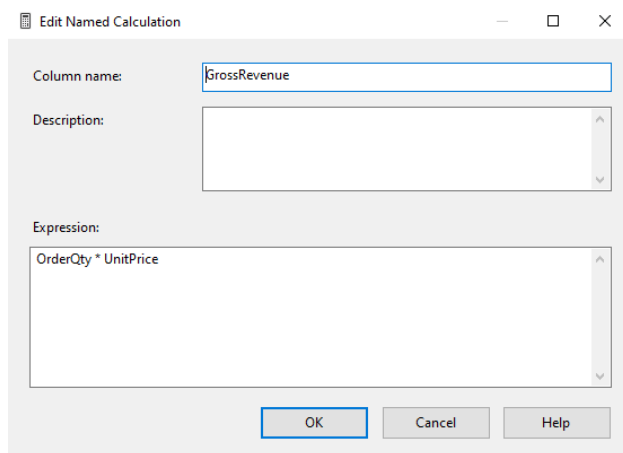


Rysunek 13: Sposób obliczania miary ze zwykłą średnią

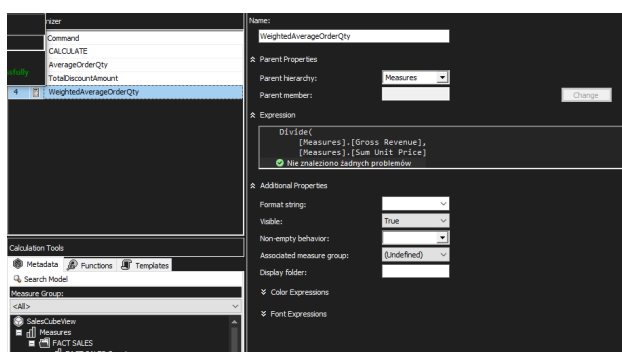
Do obliczenia średniej ważonej należało dodać miarę obliczającą sumę ceny jednostkowej i drugą miarę, będącą iloczynem ceny jednostkowej i liczby zamówionego produktu (LineTotal prawie to spełniał, ale miał w sobie czasem zniżkę).



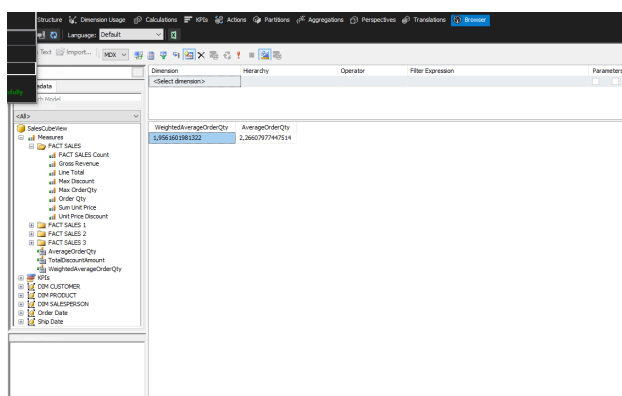
Rysunek 14: Sposób obliczania miary z sumą ceny jednostkowej



Rysunek 15: Sposób obliczania miary zysku brutto



Rysunek 16: Sposób obliczania miary średniej ważonej



Rysunek 17: Wynik średnich dla całego zbioru danych

4 Zad. 4. Partycje

Podzielić zawartość kostki na partycje (zakładka Partitions). Każda partycja powinna odzwierciedlać jeden rok. Istnieją dwa podstawowe sposoby podziału partycjonowania kostek:

- dane do zasilania poszczególnych partycji znajdują się w osobnych tabelach
- dane do zasilania poszczególnych partycji znajdują się w tej samej tabeli, zaś każda z partycji ma przypisanie zapytanie SQL, którego wynik służy do jej zasilenia.

Proszę przygotować partycje na dwa sposoby i znaleźć uzasadnienie dla każdej opcji.

4.1 Sposób pierwszy

Wymaga to stworzenia 4 nowych tabeli o schemacie identycznym do FACT_SALES i wypełnienie ich odpowiednimi danymi.

```
1 CREATE TABLE Kubs.FACT_SALES_2011 (  
2     ProductID INT FOREIGN KEY REFERENCES Kubs.DIM_PRODUCT(ProductID),  
3     CustomerID INT FOREIGN KEY REFERENCES Kubs.DIM_CUSTOMER(  
4         CustomerID),  
5     SalesPersonID INT FOREIGN KEY REFERENCES Kubs.DIM_SALESPERSON(  
6         SalesPersonID),  
7     OrderDate INT NOT NULL,  
8     ShipDate INT NULL,  
9     OrderQty SMALLINT NOT NULL,  
10    UnitPrice MONEY NOT NULL,  
11    UnitPriceDiscount DECIMAL(8, 4) NOT NULL,  
12    LineTotal DECIMAL(19, 4) NOT NULL  
13 );  
14  
15 CREATE TABLE Kubs.FACT_SALES_2012 (  
16     ProductID INT FOREIGN KEY REFERENCES Kubs.DIM_PRODUCT(ProductID),  
17     CustomerID INT FOREIGN KEY REFERENCES Kubs.DIM_CUSTOMER(  
18         CustomerID),  
19     SalesPersonID INT FOREIGN KEY REFERENCES Kubs.DIM_SALESPERSON(  
20         SalesPersonID),  
21     OrderDate INT NOT NULL,  
22     ShipDate INT NULL,  
23     OrderQty SMALLINT NOT NULL,
```

```

20     UnitPrice MONEY NOT NULL,
21     UnitPriceDiscount DECIMAL(8, 4) NOT NULL,
22     LineTotal DECIMAL(19, 4) NOT NULL
23 );
24
25 CREATE TABLE Kubs.FACT_SALES_2013 (
26     ProductID INT FOREIGN KEY REFERENCES Kubs.DIM_PRODUCT(ProductID),
27     CustomerID INT FOREIGN KEY REFERENCES Kubs.DIM_CUSTOMER(
28     CustomerID),
29     SalesPersonID INT FOREIGN KEY REFERENCES Kubs.DIM_SALESPERSON(
30     SalesPersonID),
31     OrderDate INT NOT NULL,
32     ShipDate INT NULL,
33     OrderQty SMALLINT NOT NULL,
34     UnitPrice MONEY NOT NULL,
35     UnitPriceDiscount DECIMAL(8, 4) NOT NULL,
36     LineTotal DECIMAL(19, 4) NOT NULL
37 );
38
39 CREATE TABLE Kubs.FACT_SALES_2014 (
40     ProductID INT FOREIGN KEY REFERENCES Kubs.DIM_PRODUCT(ProductID),
41     CustomerID INT FOREIGN KEY REFERENCES Kubs.DIM_CUSTOMER(
42     CustomerID),
43     SalesPersonID INT FOREIGN KEY REFERENCES Kubs.DIM_SALESPERSON(
44     SalesPersonID),
45     OrderDate INT NOT NULL,
46     ShipDate INT NULL,
47     OrderQty SMALLINT NOT NULL,
48     UnitPrice MONEY NOT NULL,
49     UnitPriceDiscount DECIMAL(8, 4) NOT NULL,
50     LineTotal DECIMAL(19, 4) NOT NULL
51 );
52
53 with Sales1 AS (
54     SELECT
55     Kubs.FACT_SALES.ProductID,
56     Kubs.FACT_SALES.CustomerID,
57     Kubs.FACT_SALES.SalesPersonID,
58     Kubs.FACT_SALES.OrderDate,
59     Kubs.FACT_SALES.ShipDate,
60     Kubs.FACT_SALES.OrderQty,
61     Kubs.FACT_SALES.UnitPrice,
62     Kubs.FACT_SALES.UnitPriceDiscount,

```

```

59     Kubs.FACT_SALES.LineTotal
60     FROM Kubs.FACT_SALES
61     WHERE OrderDate >= 20110101 AND OrderDate < 20120000
62 )
63 INSERT INTO Kubs.FACT_SALES_2011
64
65 SELECT * FROM Sales1;
66
67 with Sales2 AS (
68     SELECT
69         Kubs.FACT_SALES.ProductID,
70         Kubs.FACT_SALES.CustomerID,
71         Kubs.FACT_SALES.SalesPersonID,
72         Kubs.FACT_SALES.OrderDate,
73         Kubs.FACT_SALES.ShipDate,
74         Kubs.FACT_SALES.OrderQty,
75         Kubs.FACT_SALES.UnitPrice,
76         Kubs.FACT_SALES.UnitPriceDiscount,
77         Kubs.FACT_SALES.LineTotal
78     FROM Kubs.FACT_SALES
79     WHERE OrderDate >= 20120101 AND OrderDate < 20130000
80 )
81 INSERT INTO Kubs.FACT_SALES_2012
82
83 SELECT * FROM Sales2;
84
85 with Sales3 AS (
86     SELECT
87         Kubs.FACT_SALES.ProductID,
88         Kubs.FACT_SALES.CustomerID,
89         Kubs.FACT_SALES.SalesPersonID,
90         Kubs.FACT_SALES.OrderDate,
91         Kubs.FACT_SALES.ShipDate,
92         Kubs.FACT_SALES.OrderQty,
93         Kubs.FACT_SALES.UnitPrice,
94         Kubs.FACT_SALES.UnitPriceDiscount,
95         Kubs.FACT_SALES.LineTotal
96     FROM Kubs.FACT_SALES
97     WHERE OrderDate >= 20130101 AND OrderDate < 20140000
98 )
99 INSERT INTO Kubs.FACT_SALES_2013
100
101 SELECT * FROM Sales3;

```

```

102
103 with Sales4 AS (
104     SELECT
105         Kubs.FACT_SALES.ProductID,
106         Kubs.FACT_SALES.CustomerID,
107         Kubs.FACT_SALES.SalesPersonID,
108         Kubs.FACT_SALES.OrderDate,
109         Kubs.FACT_SALES.ShipDate,
110         Kubs.FACT_SALES.OrderQty,
111         Kubs.FACT_SALES.UnitPrice,
112         Kubs.FACT_SALES.UnitPriceDiscount,
113         Kubs.FACT_SALES.LineTotal
114     FROM Kubs.FACT_SALES
115     WHERE OrderDate >= 20140101
116 )
117 INSERT INTO Kubs.FACT_SALES_2014

```

Listing 1: Tworzenie i wypełnianie tabeli DIM_TIME.

Następnie należało dodać każdą z tabel do projektu, a potem do partycji w kostce.

Item	Partition Name	Source	Estimated Rows	Storage Mode	Aggregation Design
1	FACT_SALES_2011	FACT_SALES_2011	0	HOLAP	
2	FACT_SALES_2012	FACT_SALES_2012	2089	HOLAP	AggregationDesign
3	FACT_SALES_2013	FACT_SALES_2013	0	HOLAP	
4	FACT_SALES_2014	FACT_SALES_2014	0	HOLAP	

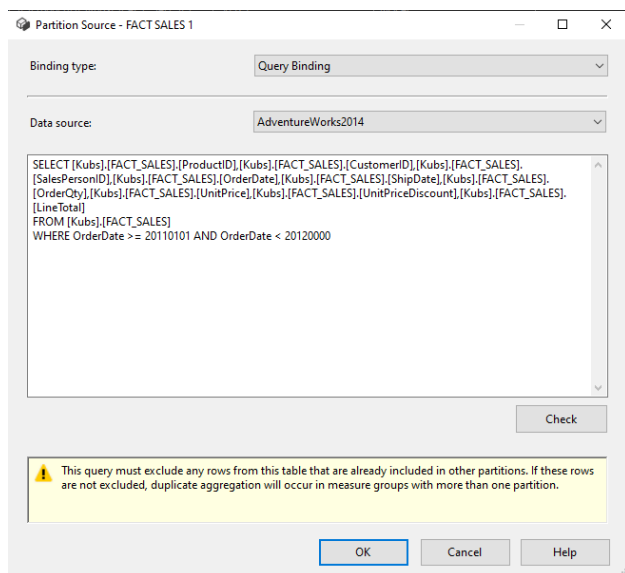
Rysunek 18: Dodane partycje

Rak	FACT_SALES Count
2011	5718
2012	2089
2013	5575
2014	3739

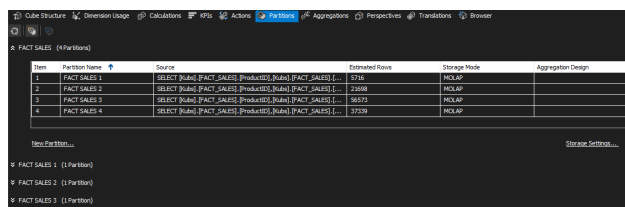
Rysunek 19: Wynik

4.2 Sposób drugi

To podejście było teoretycznie prostsze - wystarczyło zmienić kod SQL w nowo dodawanych partycjach



Rysunek 20: Zmieniony kod SQL w partycji - dodana klauzula WHERE ograniczająca daty



Rysunek 21: Dodane partycje

Rok	FACT SALES Count
2011	5716
2012	12089
2013	95573
2014	37339

Rysunek 22: Wynik

5 Zad. 5. * Definiowanie KPI

6 Wnioski

7 Wnioski