Hurtownie danych
Laboratorium Czw 11:15

Lista 4

Kajetan Pynka 254495

Zad 1

```
CREATE SCHEMA Pynka;
```

Zad 2

```
CREATE TABLE Pynka.DIM_CUSTOMER (
    CustomerID INT NOT NULL,
    FirstName NVARCHAR(50),
    LastName NVARCHAR(50),
    Title NVARCHAR(8),
    City NVARCHAR(30),
    TerritoryName NVARCHAR(50),
    CountryRegionCode NVARCHAR(3),
    [Group] NVARCHAR(50)
);

CREATE TABLE Pynka.DIM_PRODUCT (
    ProductID INT NOT NULL,
    Name NVARCHAR(50) NOT NULL,
    ListPrice MONEY NOT NULL,
    Color NVARCHAR(15),
    SubCategoryName NVARCHAR(50),
    CategoryName NVARCHAR(50),
    Weight DECIMAL(8,2),
    Size NVARCHAR(5),
    IsPurchased BIT NOT NULL
);

CREATE TABLE Pynka.DIM_SALESPERSON (
    SalesPersonID INT NOT NULL,
    FirstName NVARCHAR(50) NOT NULL,
    LastName NVARCHAR(50) NOT NULL,
    Title NVARCHAR(8),
    Gender NCHAR(1) NOT NULL,
    CountryRegionCode NVARCHAR(3),
    [Group] NVARCHAR(50)
);
```

```
CREATE TABLE Pynka.FACT_SALES (
    ProductID INT NOT NULL,
    CustomerID INT NOT NULL,
    SalesPersonID INT,
    OrderDate INT NOT NULL,
    ShipDate INT NOT NULL,
    OrderQty SMALLINT NOT NULL,
    UnitPrice MONEY NOT NULL,
    UnitPriceDiscount MONEY NOT NULL,
    LineTotal NUMERIC(38,6) NOT NULL
);
```

Wszelkie typy danych oraz obligatoryjność zostały zaczerpnięte z oryginalnych danych (za wyjątkiem dat, które zostały sklejone w liczby całkowite). Klucze główne i obce dodaję dopiero w ramach zadania czwartego.

Zad 3

```
INSERT INTO Pynka.DIM_CUSTOMER
SELECT DISTINCT C.CustomerID, P.FirstName, P.LastName, P.Title,
MIN(A.City), MIN(ST.Name), MIN(ST.CountryRegionCode),
MIN(ST.[Group])
    FROM Sales.Customer C
    JOIN Person.Person P ON P.BusinessEntityID=C.PersonID
    LEFT JOIN Person.BusinessEntityAddress BEA ON
BEA.BusinessEntityID=P.BusinessEntityID
    LEFT JOIN Person.Address A ON A.AddressID=BEA.AddressID
    LEFT JOIN Sales.SalesTerritory ST ON
ST.TerritoryID=C.TerritoryID
    GROUP BY C.CustomerID, P.FirstName, P.LastName, P.Title;

INSERT INTO Pynka.DIM_PRODUCT
SELECT DISTINCT P.ProductID, P.Name, P.ListPrice, P.Color,
PSC.Name, PC.Name, P.Weight, P.Size,
    CASE
        WHEN (SELECT COUNT(*) FROM Sales.SalesOrderDetail SOD
WHERE SOD.ProductID=P.ProductID) > 0 THEN 1 ELSE 0
    END "IsPurchased"
    FROM Production.Product P
    LEFT JOIN Production.ProductSubcategory PSC ON
PSC.ProductSubcategoryID=P.ProductSubcategoryID
```

```
    LEFT JOIN Production.ProductCategory PC ON
PC.ProductCategoryID=PSC.ProductCategoryID
    WHERE (SELECT COUNT(*) FROM Sales.SalesOrderDetail SOD WHERE
SOD.ProductID=P.ProductID) > 0;

INSERT INTO Pynka.DIM_SALESPERSON
SELECT DISTINCT SP.BusinessEntityID, P.FirstName, P.LastName,
P.Title, E.Gender, ST.CountryRegionCode, ST.[Group]
    FROM Sales.SalesPerson SP
    LEFT JOIN Person.Person P ON
P.BusinessEntityID=SP.BusinessEntityID
    LEFT JOIN HumanResources.Employee E ON
E.BusinessEntityID=SP.BusinessEntityID
    LEFT JOIN Sales.SalesTerritory ST ON
ST.TerritoryID=SP.TerritoryID;

INSERT INTO Pynka.FACT_SALES
SELECT DISTINCT SOD.ProductID, SOH.CustomerID, SOH.SalesPersonID,
    CAST(FORMAT(SOH.OrderDate, 'yyyy') + FORMAT(SOH.OrderDate,
'MM') + FORMAT(SOH.OrderDate, 'dd') AS INT),
    CAST(FORMAT(SOH.ShipDate, 'yyyy') + FORMAT(SOH.ShipDate, 'MM')
+ FORMAT(SOH.ShipDate, 'dd') AS INT),
    SOD.OrderQty, SOD.UnitPrice, SOD.UnitPriceDiscount,
SOD.LineTotal
    FROM Sales.SalesOrderHeader SOH
    LEFT JOIN Sales.SalesOrderDetail SOD ON
SOD.SalesOrderID=SOH.SalesOrderID;
```

Liczba wstawionych rekordów (po kolei dla: DIM_CUSTOMER, DIM_PRODUCT, DIM_SALESPERSON, FACT_SALES):

```
(19119 rows affected)

(266 rows affected)

(17 rows affected)

(121307 rows affected)
```

Ignoruję klientów bez podanych danych osobistych oraz produkty, które nie zostały ani razu sprzedane.

Zad 4.1

```
ALTER TABLE Pynka.DIM_CUSTOMER
ADD CONSTRAINT dim_cust_pk PRIMARY KEY (CustomerID);

ALTER TABLE Pynka.DIM_PRODUCT
ADD CONSTRAINT dim_prod_pk PRIMARY KEY (ProductID);

ALTER TABLE Pynka.DIM_SALESPERSON
ADD CONSTRAINT dim_salespers_pk PRIMARY KEY (SalesPersonID);

ALTER TABLE Pynka.FACT_SALES
ADD CONSTRAINT fact_sales_fk_prod FOREIGN KEY (ProductID)
REFERENCES Pynka.DIM_PRODUCT(ProductID);

ALTER TABLE Pynka.FACT_SALES
ADD CONSTRAINT fact_sales_fk_cust FOREIGN KEY (CustomerID)
REFERENCES Pynka.DIM_CUSTOMER(CustomerID);

ALTER TABLE Pynka.FACT_SALES
ADD CONSTRAINT fact_sales_fk_salesper FOREIGN KEY (SalesPersonID)
REFERENCES Pynka.DIM_SALESPERSON(SalesPersonID);
```

Zad 4.2

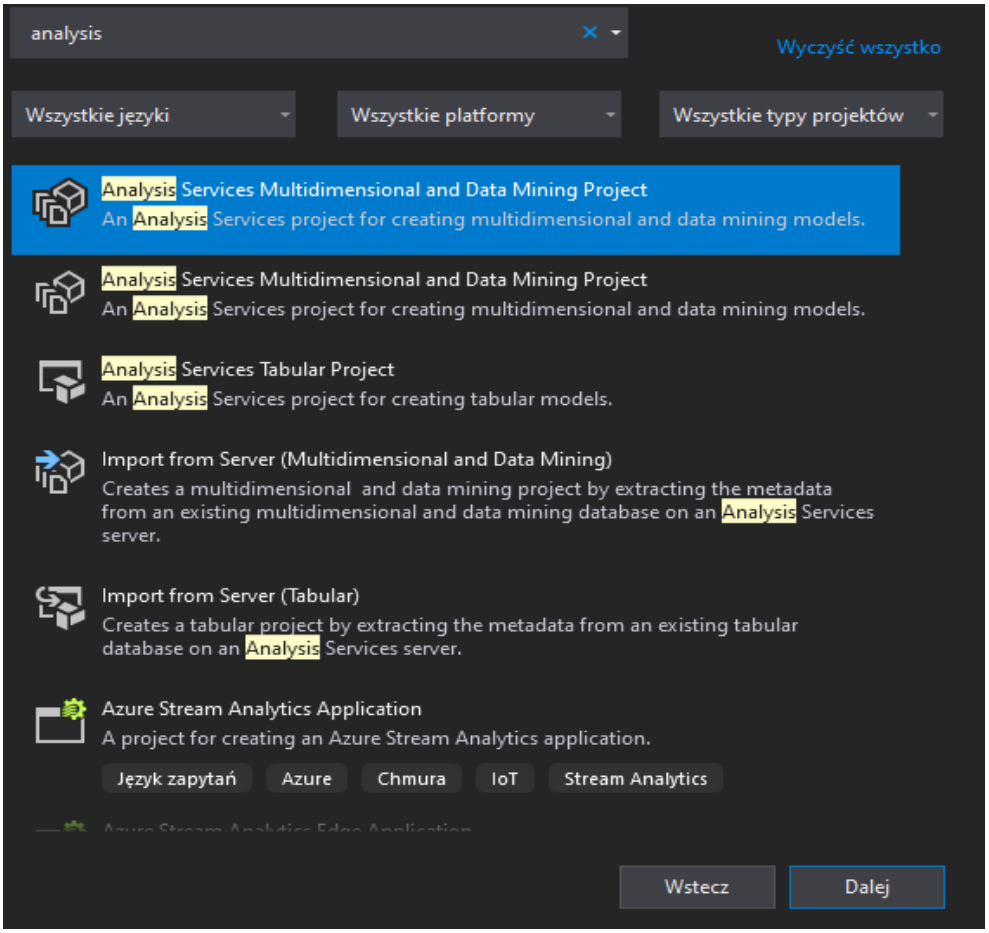
```
INSERT INTO Pynka.DIM_CUSTOMER VALUES(11242, 'Josh', 'Test', NULL,
'Baltimore', 'Central', 'US', 'North America');
INSERT INTO Pynka.DIM_PRODUCT VALUES(710, 'TEST', 123.40,
'Silver', NULL, NULL, 3.50, 40, 1);
INSERT INTO Pynka.DIM_SALESPERSON VALUES(280, 'John', 'Brooks',
'Mr.', 'M', 'US', 'North America');

INSERT INTO Pynka.FACT_SALES VALUES(NULL, 11185, NULL, 20160220,
20160323, 2, 34.90, 0.00, 34.90);
INSERT INTO Pynka.FACT_SALES VALUES(707, NULL, NULL, 20160123,
20160131, 2, 34.90, 0.00, 34.90);
INSERT INTO Pynka.FACT_SALES VALUES(1500, 11185, NULL, 20150404,
20150416, 2, 34.90, 0.00, 34.90);
INSERT INTO Pynka.FACT_SALES VALUES(707, 40000, NULL, 20130712,
20130720, 2, 34.90, 0.00, 34.90);
```

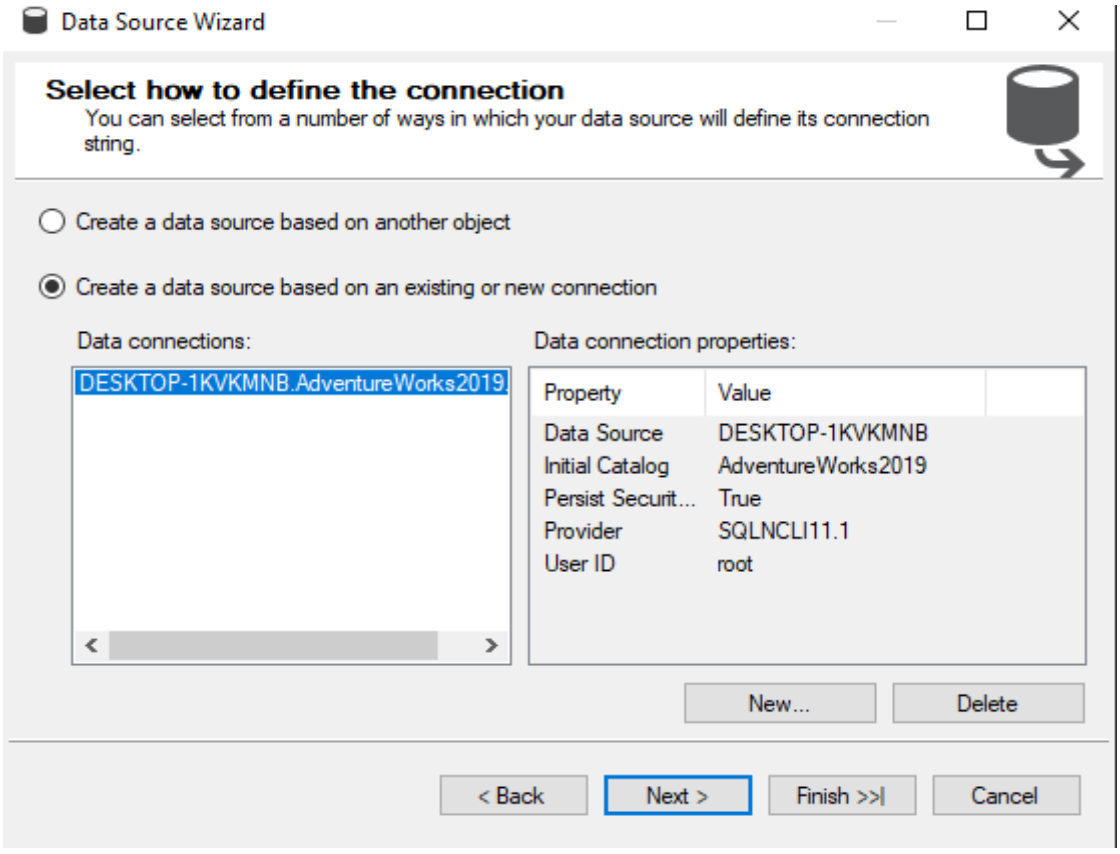
Komunikaty zwrócone przez SZBD (dla powyższych zapytań INSERT INTO):

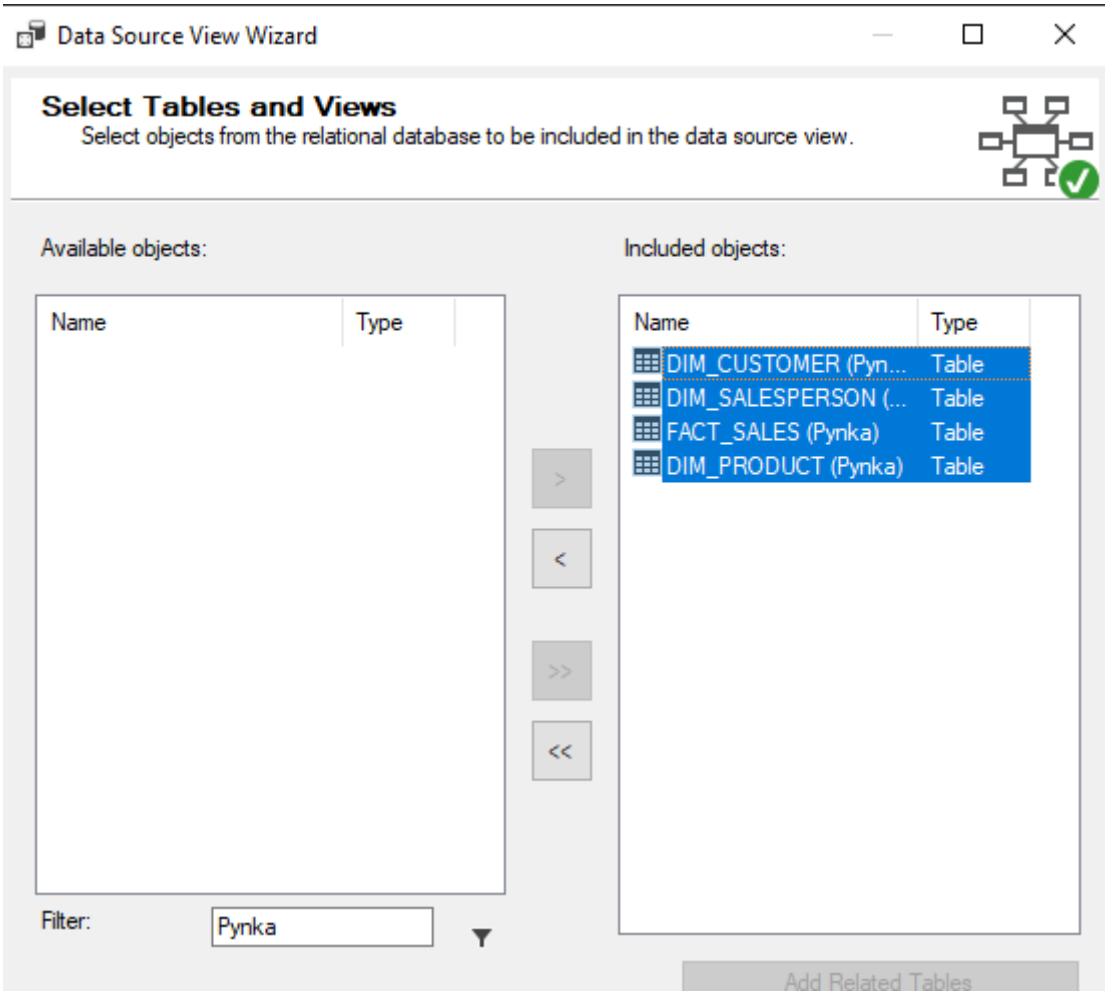
```
Msg 2627, Level 14, State 1, Line 1
Violation of PRIMARY KEY constraint 'dim_cust_pk'. Cannot insert duplicate key in obj
The statement has been terminated.
Msg 2627, Level 14, State 1, Line 2
Violation of PRIMARY KEY constraint 'dim_prod_pk'. Cannot insert duplicate key in obj
The statement has been terminated.
Msg 2627, Level 14, State 1, Line 3
Violation of PRIMARY KEY constraint 'dim_salespers_pk'. Cannot insert duplicate key i
The statement has been terminated.
Msg 515, Level 16, State 2, Line 5
Cannot insert the value NULL into column 'ProductID', table 'AdventureWorks2019.Pynka
The statement has been terminated.
Msg 515, Level 16, State 2, Line 6
Cannot insert the value NULL into column 'CustomerID', table 'AdventureWorks2019.Pynk
The statement has been terminated.
Msg 547, Level 16, State 0, Line 7
The INSERT statement conflicted with the FOREIGN KEY constraint "fact_sales_fk_prod".
The statement has been terminated.
Msg 547, Level 16, State 0, Line 8
The INSERT statement conflicted with the FOREIGN KEY constraint "fact_sales_fk_cust".
The statement has been terminated.
```

Wniosek: Wszystkie ograniczenia zostały poprawnie nałożone.

Zad 5




Data Source Wizard
Select how to define the connection
You can select from a number of ways in which your data source will define its connection string.
Create a data source based on another object
Create a data source based on an existing or new connection
Data connections:
DESKTOP-1KVKMNB.AdventureWorks2019
Data connection properties:
Property
Value
Data Source
DESKTOP-1KVKMNB
Initial Catalog
AdventureWorks2019
Persist Securit...
True
Provider
SQLNCLI11.1
User ID
root
New...
Delete
< Back
Next >
Finish >>|
Cancel


Data Source View Wizard
Select Tables and Views
Select objects from the relational database to be included in the data source view.
Available objects:
Name
Type
Included objects:
Name
Type
DIM_CUSTOMER (Pyn...
Table
DIM_SALESPERSON (...
Table
FACT_SALES (Pynka)
Table
DIM_PRODUCT (Pynka)
Table
>
<
>>
<<
Filter:
Pynka
Add Related Tables

Select Creation Method

Cubes can be created by using existing tables, creating an empty cube, or generating tables in the data source.

How would you like to create the cube?

- (•) Use existing tables
- ( ) Create an empty cube
- ( ) Generate tables in the data source

Template:

(None)

Description:

Create a cube based on one or more tables in a data source.

< Back | Next > | Finish >>| | Cancel

Select Measure Group Tables

Select a data source view or diagram and then select the tables that will be used for measure groups.

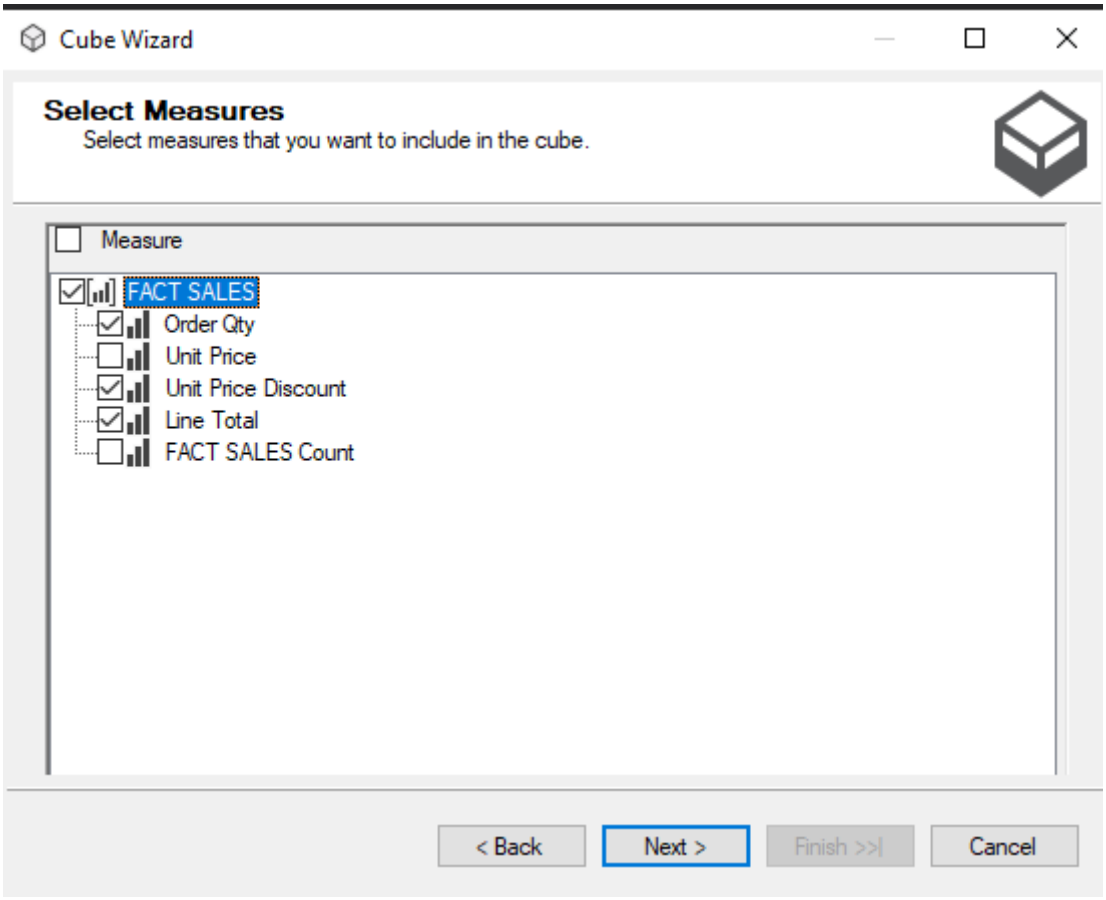
Data source view:

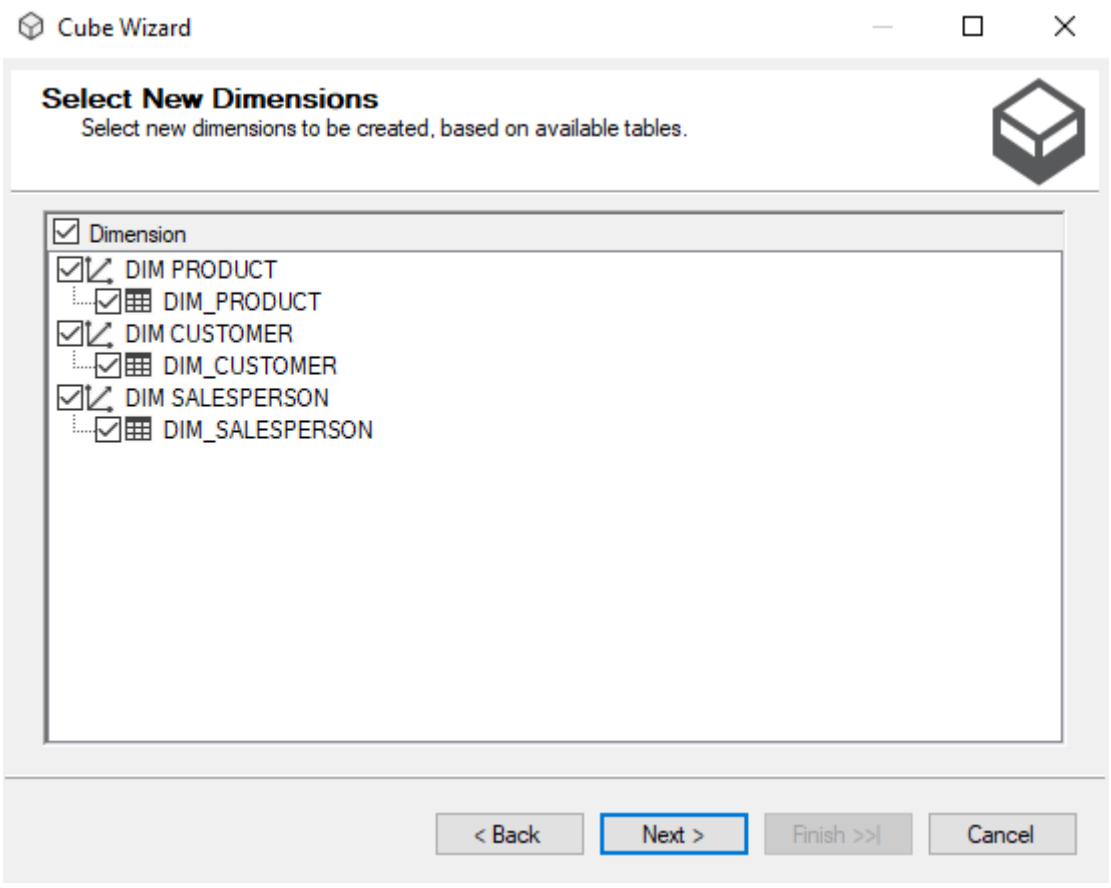
Adventure Works2019

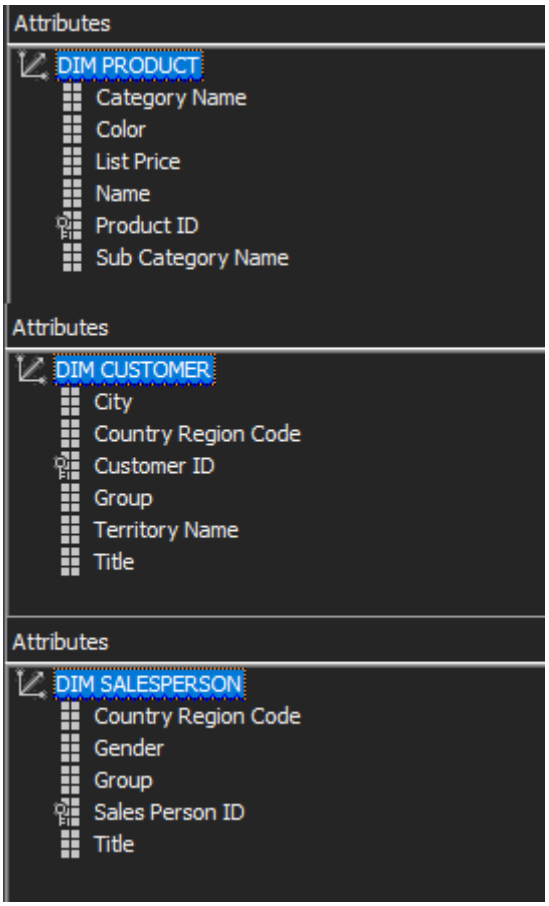
Measure group tables: Suggest

- [ ] DIM_CUSTOMER
- [ ] DIM_SALESPERSON
- [x] FACT_SALES
- [ ] DIM_PRODUCT

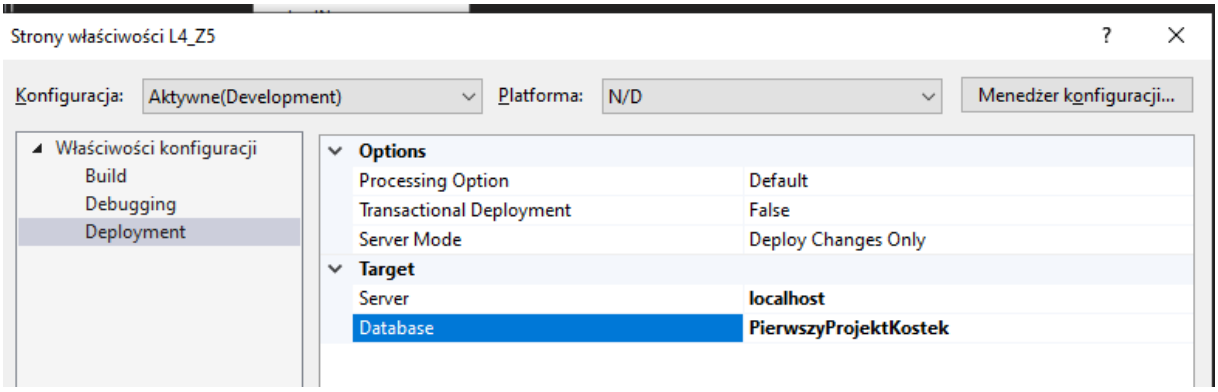
< Back | Next > | Finish >>| | Cancel

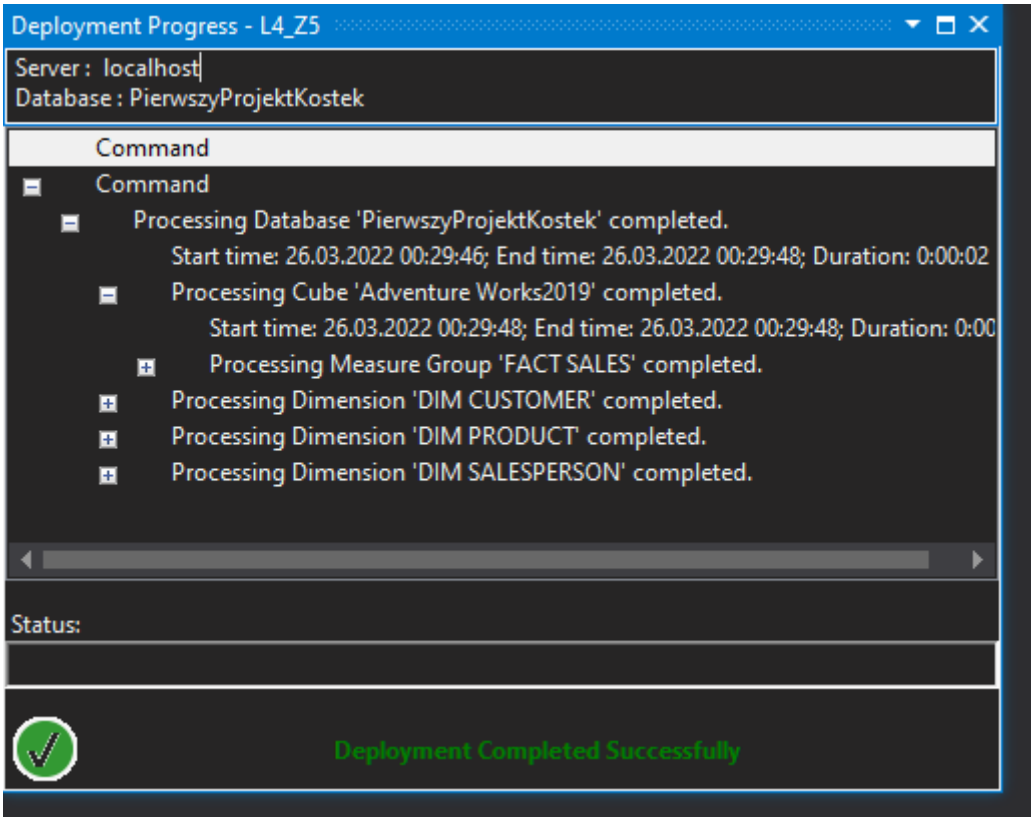

Cube Wizard
Select Measures
Select measures that you want to include in the cube.
Measure
FACT SALES
Order Qty
Unit Price
Unit Price Discount
Line Total
FACT SALES Count
< Back
Next >
Finish >>|
Cancel

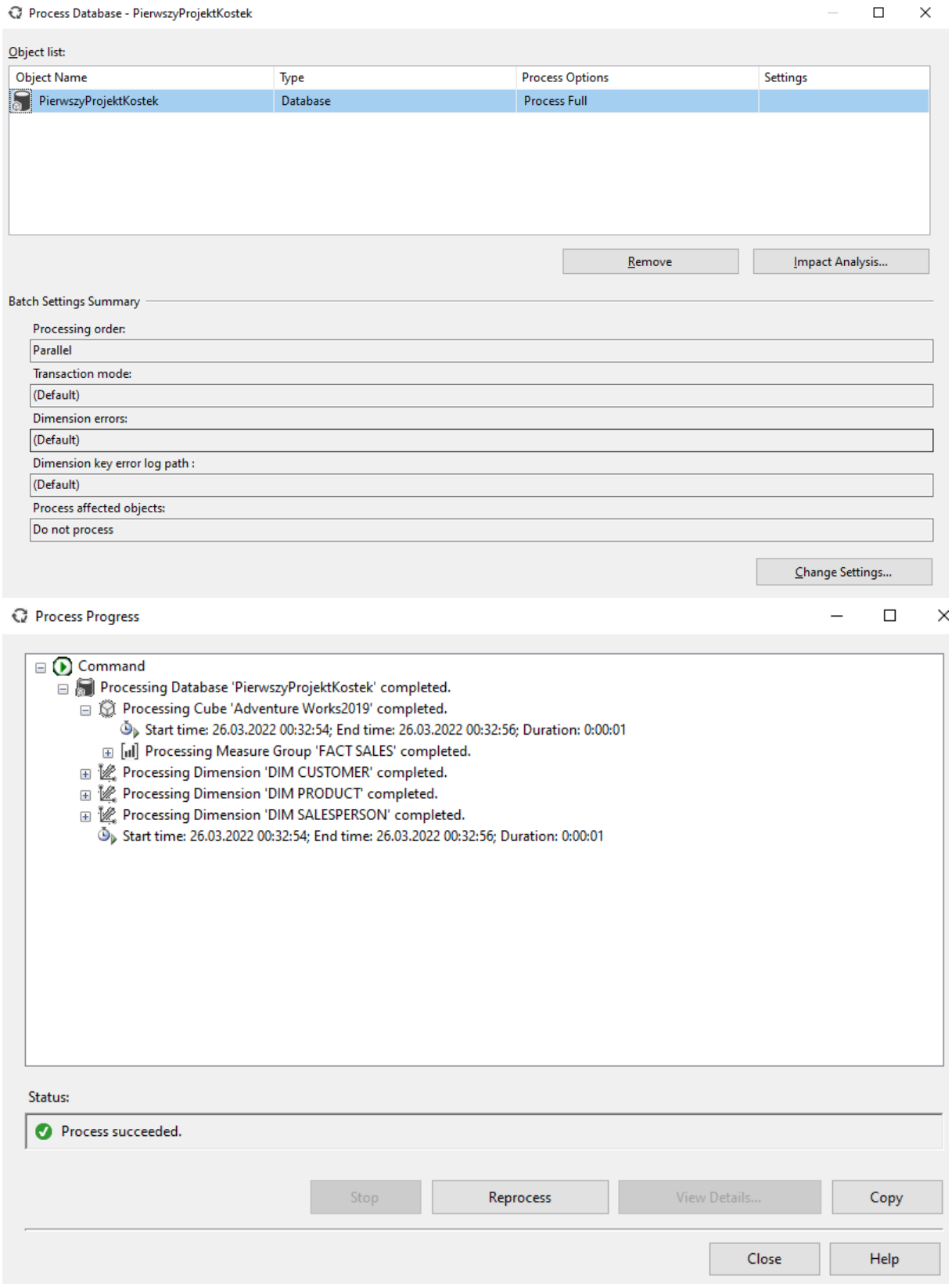

Cube Wizard
Select New Dimensions
Select new dimensions to be created, based on available tables.
Dimension
DIM PRODUCT
DIM_PRODUCT
DIM CUSTOMER
DIM_CUSTOMER
DIM SALESPERSON
DIM_SALESPERSON
< Back
Next >
Finish >>|
Cancel


Attributes
DIM PRODUCT
Category Name
Color
List Price
Name
Product ID
Sub Category Name
Attributes
DIM CUSTOMER
City
Country Region Code
Customer ID
Group
Territory Name
Title
Attributes
DIM SALESPERSON
Country Region Code
Gender
Group
Sales Person ID
Title

Zad 6

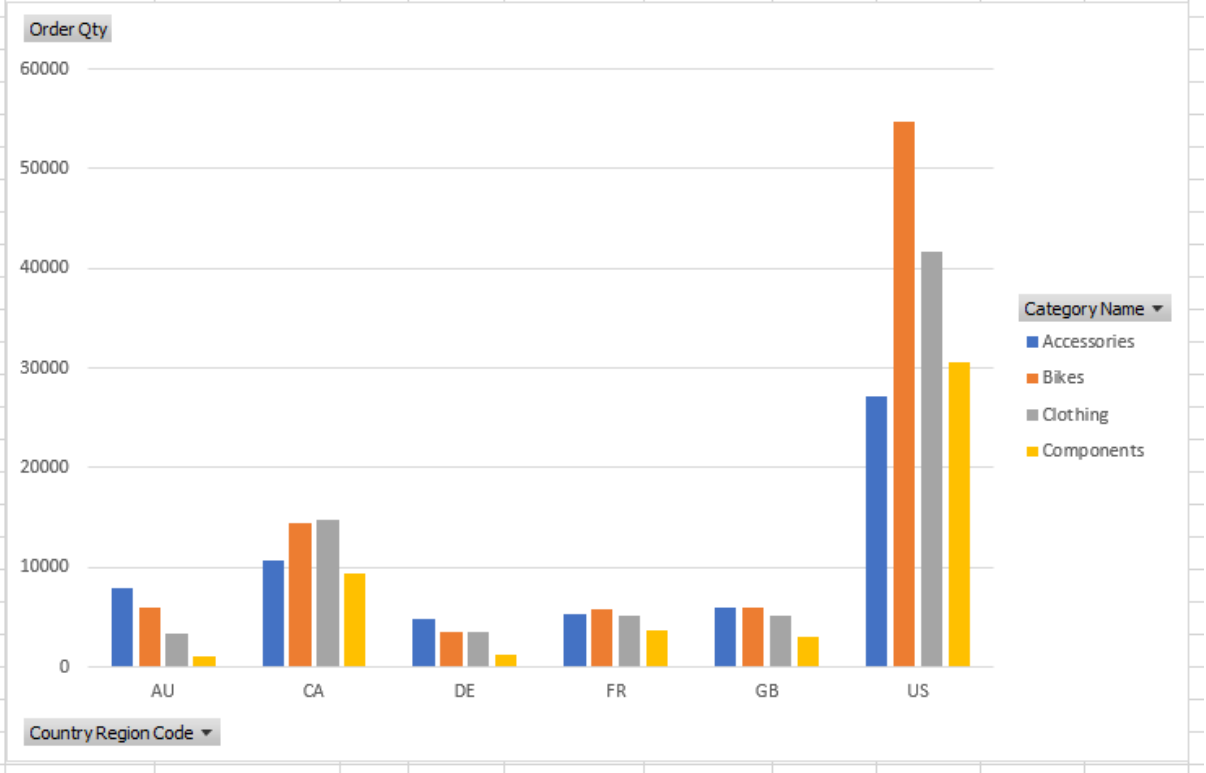

Strony właściwości L4_Z5
Konfiguracja: Aktywne(Development)
Platforma: N/D
Menedżer konfiguracji...
Właściwości konfiguracji
Build
Debugging
Deployment
Options
Processing Option Default
Transactional Deployment False
Server Mode Deploy Changes Only
Target
Server localhost
Database PierwszyProjektKostek


Deployment Progress - L4_Z5
Server : localhost
Database : PierwszyProjektKostek
Command
Command
Processing Database 'PierwszyProjektKostek' completed.
Start time: 26.03.2022 00:29:46; End time: 26.03.2022 00:29:48; Duration: 0:00:02
Processing Cube 'Adventure Works2019' completed.
Start time: 26.03.2022 00:29:48; End time: 26.03.2022 00:29:48; Duration: 0:00
Processing Measure Group 'FACT SALES' completed.
Processing Dimension 'DIM CUSTOMER' completed.
Processing Dimension 'DIM PRODUCT' completed.
Processing Dimension 'DIM SALESPERSON' completed.
Status:
Deployment Completed Successfully


Process Database - PierwszyProjektKostek
Object list:
Object Name
Type
Process Options
Settings
PierwszyProjektKostek
Database
Process Full
Remove
Impact Analysis...
Batch Settings Summary
Processing order:
Parallel
Transaction mode:
(Default)
Dimension errors:
(Default)
Dimension key error log path :
(Default)
Process affected objects:
Do not process
Change Settings...
Process Progress
Command
Processing Database 'PierwszyProjektKostek' completed.
Processing Cube 'Adventure Works2019' completed.
Start time: 26.03.2022 00:32:54; End time: 26.03.2022 00:32:56; Duration: 0:00:01
Processing Measure Group 'FACT SALES' completed.
Processing Dimension 'DIM CUSTOMER' completed.
Processing Dimension 'DIM PRODUCT' completed.
Processing Dimension 'DIM SALESPERSON' completed.
Start time: 26.03.2022 00:32:54; End time: 26.03.2022 00:32:56; Duration: 0:00:01
Status:
Process succeeded.
Stop
Reprocess
View Details...
Copy
Close
Help

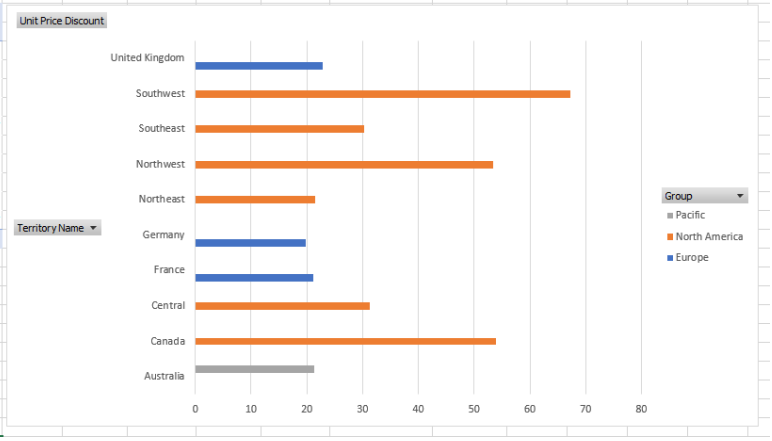
Zad 7

Order Qty	Column Labels				
Row Labels	Accessories	Bikes	Clothing	Components	Grand Total
AU	7867	6028	3379	1019	18293
CA	10759	14414	14808	9396	49377
DE	4809	3510	3517	1299	13135
FR	5389	5738	5153	3625	19905
GB	5905	5940	5192	3061	20098
US	27196	54638	41614	30644	154092
Grand Total	**61925**	**90268**	**73663**	**49044**	**274900**



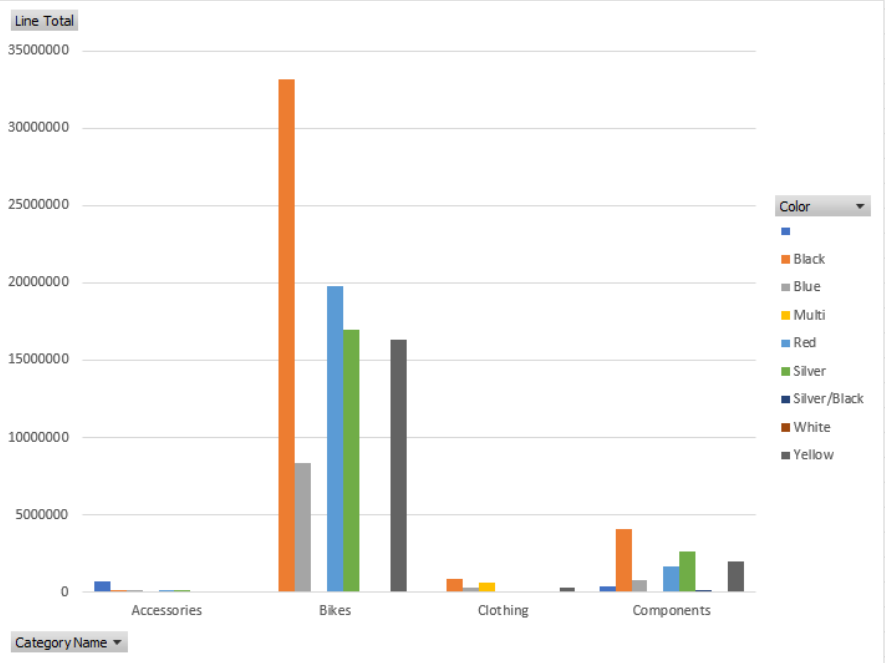
Wniosek: Najwięcej sztuk produktów sprzedawanych jest w Stanach Zjednoczonych, z czego najrzadziej kupowane są akcesoria. W pozostałych regionach ogólnie najrzadziej kupowane są komponenty.

Unit Price Discount	Column Labels			
Row Labels	Europe	North America	Pacific	Grand Total
Australia			21,36	21,36
Canada		53,9		53,9
Central		31,23		31,23
France	21,13			21,13
Germany	19,77			19,77
Northeast		21,56		21,56
Northwest		53,43		53,43
Southeast		30,31		30,31
Southwest		67,29		67,29
United Kingdom	22,87			22,87
Grand Total	**63,77**	**257,72**	**21,36**	**342,85**



Wniosek: Na podstawie tych danych można zastanawiać się nad użytecznością miary UnitPriceDiscount. Wiadomo, że sumarycznie najwięcej przecen pojawiło się na zachodzie Stanów Zjednoczonych i Kanadzie. Nie uwzględnia to jednak faktu, że w tych regionach było sprzedawanych najwięcej produktów (być może wypadałoby to uśrednić). Większy sens wydaje się mieć liczenie przecen (gdy UnitPriceDiscount > 0) niż sumowanie procentów.

Line Total	Column Labels								
Row Labels		Black	Blue	Multi	Red	Silver	Silver/Black	White	Yellow	Grand Total
Accessories	682114,7763	160869,5178	165371,627		157772,3944	105826,4183				1271954,734
Bikes		33135870,15	8374313,877		19784772,19	17005080,84			16351135,65	94651172,7
Clothing		860261,5778	259488,3685	649737,6023				29745,12758	321198,2906	2120430,967
Components	417120,9712	4090017,383	803642,0957		1680724,955	2666432,69	147483,9098		1997171,282	11802593,29
Grand Total	**1099235,748**	**38247018,63**	**9602815,968**	**649737,6023**	**21623269,54**	**19777339,95**	**147483,9098**	**29745,12758**	**18669505,22**	**109846151,7**



Wniosek: W każdej kategorii najlepiej sprzedają się produkty w kolorze czarnym (jeśli zignorujemy produkty bez oznaczenia koloru). Prawdopodobnie najlepszym posunięciem byłoby wycofanie niebieskich rowerów i zwiększenie sprzedaży rowerów w pozostałych kolorach (ponieważ to rowery przynoszą ogólnie największe zyski).

Wnioski:

- Sporządzenie tabel wymiarów i faktów jest stosunkowo proste i sprowadza się do zwykłych zapytań SQL'owych. Istnieją dedykowane narzędzia i rozszerzenia, które pozwolą nam następnie utworzyć kostkę w bardzo prosty sposób.
- Przetworzoną kostkę możemy wykorzystać do analizowania posiadanych danych ze względu na dowolne wybrane przez nas wymiary i dowolne miary. Pozwala to wysuwać wnioski co do powiązania pewnych wartości i zależności występujących w hurtowni.