

## Hurtownie Danych - laboratorium

### Lista 4

#### *Wielowymiarowy model danych*

##### Wstęp teoretyczny

W ramach niniejszej listy wykorzystane zostaną podstawowe skrypty DDL oraz DML.

Tworzenie schematu:

```
CREATE SCHEMA schema_name [ <schema_element> [ ...n ] ]
```

Tworzenie tabeli:

```
CREATE TABLE  
  { database_name.schema_name.table_name. |  
    schema_name.table_name | table_name }  
  ( { <column_definition> } [ ,...n ] )
```

Wypełnianie tabeli danymi:

```
[ WITH <common_table_expression> [ ,...n ] ]  
INSERT  
{  
  [ TOP ( expression ) [ PERCENT ] ]  
  [ INTO ]  
  { <object> | rowset_function_limited  
    [ WITH ( <Table_Hint_Limited> [ ...n ] ) ]  
  }  
  {  
    [ ( column_list ) ]  
    [ <OUTPUT Clause> ]  
    { VALUES ( { DEFAULT | NULL | exp } [ ,...n ] )  
      | derived_table  
      | execute_statement  
      | <dml_table_source>  
      | DEFAULT VALUES  
    }  
  }  
}
```

Dodawanie więzów integralności:

```
ALTER TABLE table_name  
ADD CONSTRAINT constraint_name  
{  
  [ NULL | NOT NULL ]  
  { PRIMARY KEY | UNIQUE }  
  [ CLUSTERED | NONCLUSTERED ]
```

```
[ WITH FILLFACTOR = fillfactor ]
[ WITH ( index_option [, ...n ] ) ]
[ ON { partition_scheme_name (partition_column_name)
    | filegroup | "default" } ]
FOREIGN KEY attribute_name
REFERENCES [ schema_name . ] referenced_table_name
    [ ( ref_column ) ]
[ ON DELETE { NO ACTION | CASCADE | SET NULL | SET DEFAULT } ]
    [ NOT FOR REPLICATION ]
    | CHECK [ NOT FOR REPLICATION ] ( logical_expression )
}
```

#### Źródła:

<https://docs.microsoft.com/en-us/sql/t-sql/statements/create-schema-transact-sql?view=sql-server-ver15>  
<https://docs.microsoft.com/en-us/sql/t-sql/statements/create-table-transact-sql?view=sql-server-ver15>  
<https://docs.microsoft.com/en-us/sql/t-sql/statements/insert-transact-sql?view=sql-server-ver15>  
<https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-table-column-constraint-transact-sql?view=sql-server-ver15>

## Zadania do wykonania

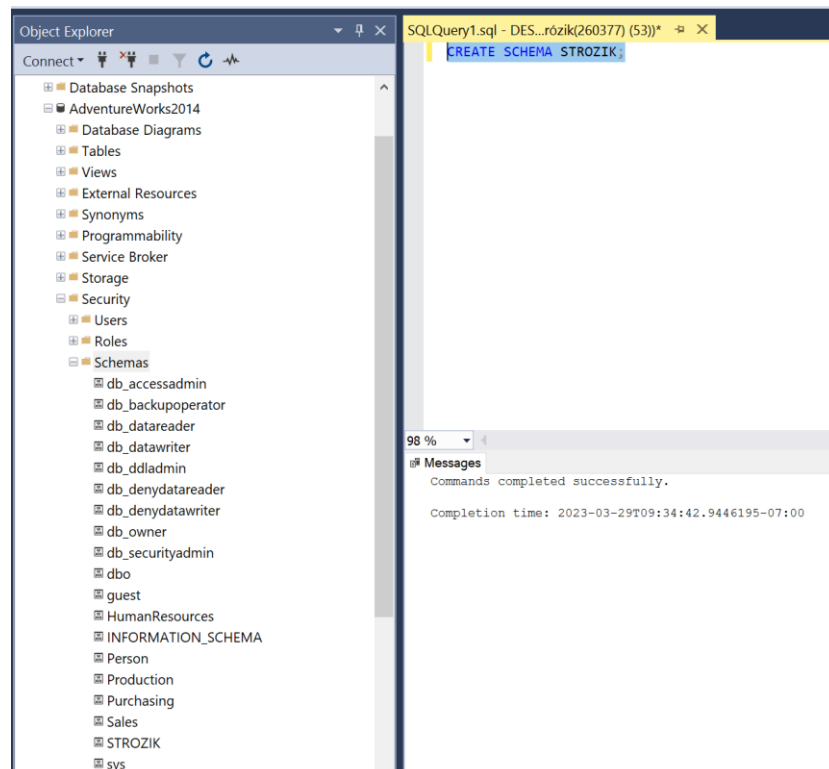
Baza danych: **AdventureWorks**

### Zad 1. Przygotowanie schematu

W bazie danych należy utworzyć schemat, którego nazwa będzie odpowiadać nazwisku wykonującego ćwiczenie (zapisać zapytanie tworzące ten schemat).

#### Rozwiązanie:

**CREATE SCHEMA** STROZIK;



**Zad 2. Tworzenie tabel wymiarów i tabeli faktów**

W nowo utworzonym schemacie utworzyć tabele wymiarów: klienta, produktu i sprzedawcy (zapisać skrypt `CREATE TABLE`), opisane w następujących schematach:

- **DIM\_CUSTOMER** (CustomerID, FirstName, LastName, Title, City, TerritoryName, CountryRegionCode, Group)
- **DIM\_PRODUCT** (ProductID, Name, ListPrice, Color, SubCategoryName, CategoryName, Weight, Size, IsPurchased)
- **DIM SALESPERSON** (SalesPersonID, FirstName, LastName, Title, Gender, CountryRegionCode, Group)

oraz tabelę faktów:

- **FACT\_SALES** (ProductID, CustomerID, SalesPersonID, OrderDate, ShipDate, OrderQty, UnitPrice, UnitPriceDiscount, LineTotal)

*Uwaga 1: Proszę sprawdzić, dlaczego niektóre atrybuty mają wartość NULL. Czy zawsze wynika to z braku danych?*

*Uwaga 2: Kolumny OrderDate oraz ShipDate przechowują dane typu całkowitego, gdzie cztery pierwsze cyfry oznaczają rok, dwie następne miesiąc, a dwie ostatnie dzień. Do pobrania poszczególnych części daty użyć funkcji datepart.*

*Uwaga 3: Atrybuty OrderQty oraz LineTotal będą miarami w naszej hurtowni.*

**--2.1**

```
CREATE TABLE STROZIK.DIM_CUSTOMER
(
    CustomerID INT IDENTITY(1,1) PRIMARY KEY NOT NULL,
    FirstName NVARCHAR(50) NOT NULL,
    LastName NVARCHAR(50) NOT NULL,
    Title NVARCHAR(8),
    City NVARCHAR(30) NOT NULL,
    TerritoryName NVARCHAR(50) NOT NULL,
    CountryRegionCode NVARCHAR(3) NOT NULL,
    [Group] NVARCHAR(50) NOT NULL
);
GO
```

| DESKTOP-B5JBJO...IK.DIM_CUSTOMER |                   |              |                                     |
|----------------------------------|-------------------|--------------|-------------------------------------|
|                                  | Column Name       | Data Type    | Allow Nulls                         |
| ✖                                | CustomerID        | int          | <input type="checkbox"/>            |
|                                  | FirstName         | nvarchar(50) | <input type="checkbox"/>            |
|                                  | LastName          | nvarchar(50) | <input type="checkbox"/>            |
|                                  | Title             | nvarchar(8)  | <input checked="" type="checkbox"/> |
|                                  | City              | nvarchar(30) | <input type="checkbox"/>            |
|                                  | TerritoryName     | nvarchar(50) | <input type="checkbox"/>            |
|                                  | CountryRegionCode | nvarchar(3)  | <input type="checkbox"/>            |
|                                  | [Group]           | nvarchar(50) | <input type="checkbox"/>            |
|                                  |                   |              | <input type="checkbox"/>            |

Tutaj aby poprawnie wstawić rekordy w dalszej części listy musimy ustawić, że atrybut CustomerID będzie automatycznie generowany, zaczynając od 1 oraz inkrementując o 1, ponieważ złączenia generujące wyniki do wstawienia w nowoutworzone wyniki zawierają powtórzenia CustomerID spowodowane niejednoznacznością wskazywaniem adresu związanego z klientem. Dla jednego Klienta może być wiele adresów. Jest to spowodowane budową modelu tej bazy danych.

--2.2

```
CREATE TABLE STROZIK.DIM_PRODUCT
(
    ProduktID INT NOT NULL PRIMARY KEY,
    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
);
GO
```

DESKTOP-B5JEBJO...ZIK.DIM\_PRODUCT

|   | Column Name     | Data Type     | Allow Nulls                         |
|---|-----------------|---------------|-------------------------------------|
| 🔍 | ProduktID       | int           | <input type="checkbox"/>            |
|   | Name            | nvarchar(50)  | <input type="checkbox"/>            |
|   | ListPrice       | money         | <input type="checkbox"/>            |
|   | Color           | nvarchar(15)  | <input checked="" type="checkbox"/> |
|   | SubCategoryName | nvarchar(50)  | <input checked="" type="checkbox"/> |
|   | CategoryName    | nvarchar(50)  | <input checked="" type="checkbox"/> |
|   | Weight          | decimal(8, 2) | <input checked="" type="checkbox"/> |
|   | Size            | nvarchar(5)   | <input checked="" type="checkbox"/> |
|   | IsPurchased     | bit           | <input checked="" type="checkbox"/> |
|   |                 |               | <input type="checkbox"/>            |

--2.3

```
CREATE TABLE STROZIK.DIM SALESPERSON
(
    SalesPersonID INT NOT NULL PRIMARY KEY,
    FirstName NVARCHAR(50) NOT NULL,
    LastName NVARCHAR(50) NOT NULL,
    Title NVARCHAR(8),
    Gender NCHAR(1),
    CountryRegionCode NVARCHAR(3),
    [Group] NVARCHAR(50)
);
GO
```

| DESKTOP-B5JEBJO...DIM_SALESPERSON |                   |              |                                     |
|-----------------------------------|-------------------|--------------|-------------------------------------|
|                                   | Column Name       | Data Type    | Allow Nulls                         |
| ▼                                 | SalesPersonID     | int          | <input type="checkbox"/>            |
|                                   | FirstName         | nvarchar(50) | <input type="checkbox"/>            |
|                                   | LastName          | nvarchar(50) | <input type="checkbox"/>            |
|                                   | Title             | nvarchar(8)  | <input checked="" type="checkbox"/> |
|                                   | Gender            | nchar(1)     | <input checked="" type="checkbox"/> |
|                                   | CountryRegionCode | nvarchar(3)  | <input checked="" type="checkbox"/> |
|                                   | [Group]           | nvarchar(50) | <input checked="" type="checkbox"/> |
|                                   |                   |              | <input type="checkbox"/>            |

--2.4

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

| DESKTOP-B5JEBJO....ROZIK.FACT_SALES |                |                                     |
|-------------------------------------|----------------|-------------------------------------|
| Column Name                         | Data Type      | Allow Nulls                         |
| ProductID                           | int            | <input type="checkbox"/>            |
| CustomerID                          | int            | <input type="checkbox"/>            |
| SalesPersonID                       | int            | <input checked="" type="checkbox"/> |
| OrderDate                           | datetime       | <input type="checkbox"/>            |
| ShipDate                            | datetime       | <input type="checkbox"/>            |
| OrderQty                            | int            | <input type="checkbox"/>            |
| UnitPrice                           | money          | <input type="checkbox"/>            |
| UnitPriceDiscount                   | money          | <input type="checkbox"/>            |
| LineTotal                           | numeric(38, 6) | <input type="checkbox"/>            |

Tak jak poniżej w zadaniu 2, pozwalamy na wartości NULL dla sprzedawcy.

### Wnioski do Uwag:

1. Niektóre atrybuty mają wartość NULL, a wynika to z tego faktu że są one kluczami obcymi odwołującym się do innych tabel. jeżeli nie istnieje referencja danego rekordu do innego rekordu w innej tabeli, to klucz obcy będzie miał wartość NULL. Głównie wynika to ze sposobu zamodelowania tabel i relacji między nimi w taki sposób aby odzwierciedlić wycięty kawałek rzeczywistości.
2. Akurat w przypadku tej wersji Adventure Works 2014, typy OrderDate i ShipDate są typami przedstawiającymi datę. Być może w innych wersjach tej bazy ten przypadek jest prawdziwy.
3. OrderQty oraz LineTotal są typami, które nadają się jako miary naszej hurtowni, ponieważ są miarami liczbowymi.

**Zad. 3. Wypełnianie danych – denormalizacja źródłowej bazy**

Wypełnić nowoutworzone tabele danymi znajdującymi się w tabelach źródłowych. Do wypełnienia użyć instrukcji `INSERT INTO`. Proszę sprawdzić liczbę skopiowanych rekordów.

*Uwaga 1. Do tabeli DIM\_PRODUCT należy także skopiować produkty, które nie mają przypisanej podkategorii.*

*Uwaga 2. Do tabeli FACT\_SALES należy skopiować również transakcje, które nie mają sprzedawcy.*

**Rozwiązania:**

--3.1

```
SET IDENTITY_INSERT STROZIK.DIM_CUSTOMER ON
INSERT INTO
    STROZIK.DIM_CUSTOMER (
        CustomerID,
        FirstName,
        LastName,
        Title,
        City,
        TerritoryName,
        CountryRegionCode,
        [Group]
    )
SELECT
    DISTINCT C.CustomerID,
    P.FirstName,
    P.LastName,
    P.Title,
    A.City,
    T.[Name],
    T.CountryRegionCode,
    T.[Group]
FROM Sales.Customer C JOIN Person.Person P on C.PersonId = P.BusinessEntityID
JOIN Sales.SalesTerritory T on C.TerritoryID = T.TerritoryID
JOIN Sales.SalesOrderHeader H on C.CustomerID = H.CustomerID
JOIN Person.Address A on A.AddressID=H.ShipToAddressID;
GO
SET IDENTITY_INSERT STROZIK.DIM_CUSTOMER OFF;
```

|    | CustomerID | FirstName | LastName | Title | City            | TerritoryName | CountryRegionCode | Group   |
|----|------------|-----------|----------|-------|-----------------|---------------|-------------------|---------|
| 1  | 11000      | Jon       | Yang     | NULL  | Rockhampton     | Australia     | AU                | Pacific |
| 2  | 11001      | Eugene    | Huang    | NULL  | Seaford         | Australia     | AU                | Pacific |
| 3  | 11002      | Ruben     | Torres   | NULL  | Hobart          | Australia     | AU                | Pacific |
| 4  | 11003      | Christy   | Zhu      | NULL  | North Ryde      | Australia     | AU                | Pacific |
| 5  | 11004      | Elizabeth | Johnson  | NULL  | Wollongong      | Australia     | AU                | Pacific |
| 6  | 11005      | Julio     | Ruiz     | NULL  | East Brisbane   | Australia     | AU                | Pacific |
| 7  | 11006      | Janet     | Alvarez  | NULL  | Matraville      | Australia     | AU                | Pacific |
| 8  | 11007      | Marco     | Mehta    | NULL  | Warrnambool     | Australia     | AU                | Pacific |
| 9  | 11008      | Rob       | Verhoff  | NULL  | Bendigo         | Australia     | AU                | Pacific |
| 10 | 11009      | Shannon   | Carlson  | NULL  | Hervey Bay      | Australia     | AU                | Pacific |
| 11 | 11010      | Jacquelyn | Suarez   | NULL  | East Brisbane   | Australia     | AU                | Pacific |
| 12 | 11011      | Curtis    | Lu       | NULL  | East Brisbane   | Australia     | AU                | Pacific |
| 13 | 11012      | Lauren    | Walker   | NULL  | Bremerton       | Northwest     | US                | Nort... |
| 14 | 11013      | Ian       | Jenkins  | NULL  | Lebanon         | Northwest     | US                | Nort... |
| 15 | 11014      | Sydney    | Bennett  | NULL  | Redmond         | Northwest     | US                | Nort... |
| 16 | 11015      | Chloe     | Young    | NULL  | Burbank         | Southwest     | US                | Nort... |
| 17 | 11016      | Wyatt     | Hill     | NULL  | Imperial Bea... | Southwest     | US                | Nort... |

```
SELECT COUNT(*) FROM STROZIK.DIM_CUSTOMER;
```

```
SELECT COUNT(*) FROM Sales.Customer;
```

| Results |                  |
|---------|------------------|
|         | (No column name) |
| 1       | 19119            |

---

| Results |                  |
|---------|------------------|
|         | (No column name) |
| 1       | 19820            |

--3.2

INSERT INTO

STROZIK.DIM\_PRODUCT

SELECT

p.ProductID,  
 p.Name,  
 p.ListPrice,  
 p.Color,  
 sc.Name,  
 c.Name,  
 p.Weight,  
 p.Size,  
 ~p.MakeFlag

FROM Production.Product p

LEFT JOIN Production.ProductSubcategory sc ON p.ProductSubcategoryID = sc.ProductSubcategoryID

LEFT JOIN Production.ProductCategory c ON sc.ProductCategoryID = c.ProductCategoryID;

|    | ProduktID | Name                  | ListPrice | Color  | SubCategoryName | CategoryName | Weight | Size | IsPurchased |
|----|-----------|-----------------------|-----------|--------|-----------------|--------------|--------|------|-------------|
| 1  | 1         | Adjustable Race       | 0.00      | NULL   | NULL            | NULL         | NULL   | NULL | 1           |
| 2  | 2         | Bearing Ball          | 0.00      | NULL   | NULL            | NULL         | NULL   | NULL | 1           |
| 3  | 3         | BB Ball Bearing       | 0.00      | NULL   | NULL            | NULL         | NULL   | NULL | 0           |
| 4  | 4         | Headset Ball Bearings | 0.00      | NULL   | NULL            | NULL         | NULL   | NULL | 1           |
| 5  | 316       | Blade                 | 0.00      | NULL   | NULL            | NULL         | NULL   | NULL | 0           |
| 6  | 317       | LL Crankarm           | 0.00      | Black  | NULL            | NULL         | NULL   | NULL | 1           |
| 7  | 318       | ML Crankarm           | 0.00      | Black  | NULL            | NULL         | NULL   | NULL | 1           |
| 8  | 319       | HL Crankarm           | 0.00      | Black  | NULL            | NULL         | NULL   | NULL | 1           |
| 9  | 320       | Chainring Bolts       | 0.00      | Silver | NULL            | NULL         | NULL   | NULL | 1           |
| 10 | 321       | Chainring Nut         | 0.00      | Silver | NULL            | NULL         | NULL   | NULL | 1           |
| 11 | 322       | Chainring             | 0.00      | Black  | NULL            | NULL         | NULL   | NULL | 1           |
| 12 | 323       | Crown Race            | 0.00      | NULL   | NULL            | NULL         | NULL   | NULL | 1           |
| 13 | 324       | Chain Stays           | 0.00      | NULL   | NULL            | NULL         | NULL   | NULL | 0           |
| 14 | 325       | Decal 1               | 0.00      | NULL   | NULL            | NULL         | NULL   | NULL | 1           |
| 15 | 326       | Decal 2               | 0.00      | NULL   | NULL            | NULL         | NULL   | NULL | 1           |
| 16 | 327       | Down Tube             | 0.00      | NULL   | NULL            | NULL         | NULL   | NULL | 0           |
| 17 | 328       | Mountain End Caps     | 0.00      | NULL   | NULL            | NULL         | NULL   | NULL | 0           |
| 18 | 329       | Road End Caps         | 0.00      | NULL   | NULL            | NULL         | NULL   | NULL | 0           |

```
SELECT COUNT(*) FROM STROZIK.DIM_PRODUCT;
```

```
SELECT COUNT(*) FROM Production.Product p;
```

| Results |                  |
|---------|------------------|
|         | (No column name) |
|         | 504              |

---

| Results |                  |
|---------|------------------|
|         | (No column name) |
|         | 504              |

Jak widać poprzez poprawnie zdefiniowania ograniczeń dla wartości, które wskazują na klucz obcy, wyświetlamy produkty, które nie posiadają żadnej podkategorii, a co za tym też nie posiadają kategorii.

--3.3

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

|    | SalesPersonID | FirstName | LastName          | Title | Gender | CountryRegionCode | Group         |
|----|---------------|-----------|-------------------|-------|--------|-------------------|---------------|
| 1  | 274           | Stephen   | Jiang             | NULL  | M      | NULL              | NULL          |
| 2  | 275           | Michael   | Blythe            | NULL  | M      | US                | North America |
| 3  | 276           | Linda     | Mitchell          | NULL  | F      | US                | North America |
| 4  | 277           | Jillian   | Carson            | NULL  | F      | US                | North America |
| 5  | 278           | Garrett   | Vargas            | NULL  | M      | CA                | North America |
| 6  | 279           | Tsvi      | Reiter            | NULL  | M      | US                | North America |
| 7  | 280           | Pamela    | Ansman-Wolfe      | NULL  | F      | US                | North America |
| 8  | 281           | Shu       | Ito               | NULL  | M      | US                | North America |
| 9  | 282           | José      | Saraiva           | NULL  | M      | CA                | North America |
| 10 | 283           | David     | Campbell          | NULL  | M      | US                | North America |
| 11 | 284           | Tete      | Mensa-Annan       | Mr.   | M      | US                | North America |
| 12 | 285           | Syed      | Abbas             | Mr.   | M      | NULL              | NULL          |
| 13 | 286           | Lynn      | Tsoflias          | NULL  | F      | AU                | Pacific       |
| 14 | 287           | Amy       | Alberts           | NULL  | F      | NULL              | NULL          |
| 15 | 288           | Rachel    | Valdez            | NULL  | F      | DE                | Europe        |
| 16 | 289           | Jae       | Pak               | NULL  | F      | GB                | Europe        |
| 17 | 290           | Ranjit    | Varkey Chudukatil | NULL  | M      | FR                | Europe        |

```
SELECT COUNT(*) FROM Sales.SalesPerson;
SELECT COUNT(*) FROM STROZIK.DIM_SALESPERSON;
```

|   | (No column name) |
|---|------------------|
| 1 | 17               |

|   | (No column name) |
|---|------------------|
| 1 | 17               |

Wynik lekko się różni od tego z rozwiązania przykładowego, ale rozumiem, że jest to kwestia wersji użytej bazy danych.



--3.4

```
INSERT INTO STROZIK.FACT_SALES
SELECT
```

```
    DISTINCT
    SOD.ProductID,
    SOH.CustomerID,
    SOH.SalesPersonID,
    SOH.OrderDate,
    SOH.ShipDate,
    SOD.OrderQty,
    SOD.UnitPrice,
    SOD.UnitPriceDiscount,
    SOD.LineTotal
```

```
FROM Sales.SalesOrderHeader SOH
```

```
LEFT JOIN Sales.SalesOrderDetail SOD ON SOD.SalesOrderID = SOH.SalesOrderID;
```

|    | ProductID | CustomerID | SalesPersonID | OrderDate               | ShipDate                | OrderQty | UnitPrice | UnitPriceDiscount | LineTotal |
|----|-----------|------------|---------------|-------------------------|-------------------------|----------|-----------|-------------------|-----------|
| 1  | 707       | 11007      | NULL          | 2013-08-19 00:00:00.000 | 2013-08-26 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 2  | 707       | 11078      | NULL          | 2013-07-18 00:00:00.000 | 2013-07-25 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 3  | 707       | 11093      | NULL          | 2013-11-08 00:00:00.000 | 2013-11-15 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 4  | 707       | 11097      | NULL          | 2013-12-24 00:00:00.000 | 2013-12-31 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 5  | 707       | 11103      | NULL          | 2013-11-06 00:00:00.000 | 2013-11-13 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 6  | 707       | 11142      | NULL          | 2013-12-16 00:00:00.000 | 2013-12-23 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 7  | 707       | 11147      | NULL          | 2013-09-29 00:00:00.000 | 2013-10-06 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 8  | 707       | 11148      | NULL          | 2014-05-05 00:00:00.000 | 2014-05-12 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 9  | 707       | 11170      | NULL          | 2014-05-31 00:00:00.000 | 2014-06-07 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 10 | 707       | 11177      | NULL          | 2014-03-16 00:00:00.000 | 2014-03-23 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 11 | 707       | 11180      | NULL          | 2014-02-25 00:00:00.000 | 2014-03-04 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 12 | 707       | 11185      | NULL          | 2013-08-17 00:00:00.000 | 2013-08-24 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 13 | 707       | 11185      | NULL          | 2014-06-02 00:00:00.000 | 2014-06-09 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 14 | 707       | 11201      | NULL          | 2013-12-01 00:00:00.000 | 2013-12-08 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 15 | 707       | 11212      | NULL          | 2014-06-19 00:00:00.000 | 2014-06-26 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 16 | 707       | 11223      | NULL          | 2013-08-22 00:00:00.000 | 2013-08-29 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 17 | 707       | 11223      | NULL          | 2013-10-01 00:00:00.000 | 2013-10-08 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 18 | 707       | 11223      | NULL          | 2013-12-06 00:00:00.000 | 2013-12-13 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |
| 19 | 707       | 11241      | NULL          | 2013-05-31 00:00:00.000 | 2013-06-07 00:00:00.000 | 1        | 34.99     | 0.00              | 34.990000 |

```
SELECT FROM STROZIK.FACT_SALES;
SELECT COUNT(*) FROM Sales.SalesOrderHeader SOH LEFT JOIN Sales.SalesOrderDetail SOD ON SOD.SalesOrderID = SOH.SalesOrderID;
SELECT COUNT(*) FROM STROZIK.FACT_SALES;
```

118 %

Results Messages

(No column name)

1 121317

---

(No column name)

1 121307

Pozwolono na zapis transakcji bez sprzedawcy.

**Zad. 4. Więzy integralności**

1. Dodać integralność referencyjną i klucze główne do tabel już zdefiniowanych.

Klucze główne udało się dodać już podczas zadania związanego z samym tworzeniem tabel. Ctu o tabele DIM\_CUSTOMER, DIM\_PRODUCT, DIM SALESPERSON. Brakuje klucza głównego dla FACT\_SALES. Będzie to klucz złożony z kluczy obcych odpowiednio ProductID, CustomerID, SalesPersonID:

```
ALTER TABLE STROZIK.FACT_SALES
ADD CONSTRAINT FK_ProductID FOREIGN KEY (ProductID) REFERENCES STROZIK.DIM_PROD-
UCT(ProductID),
    CONSTRAINT FK_CustomerID FOREIGN KEY (CustomerID) REFERENCES STROZIK.DIM_CUS-
TOMER(CustomerID),
    CONSTRAINT FK_SalesPersonID FOREIGN KEY (SalesPersonID) REFERENCES
STROZIK.DIM_SALESPERSON(SalesPersonID);
```

W ten sposób dodaliśmy zadaliśmy o integralność między tabelą faktów, a tabelami wymiarów.

2. Przygotować instrukcję INSERT INTO, która sprawdzi poprawność integralności referencyjnej oraz klucze główne.

--4.2

```
SET IDENTITY_INSERT STROZIK.DIM_CUSTOMER ON;
INSERT INTO STROZIK.DIM_CUSTOMER (CustomerID, FirstName, LastName, Title, City, Terri-
toryName, CountryRegionCode, [Group])
VALUES (11001, 'Adam', 'Kowalski', 'Pan', 'Warszawa', 'Mazowsze', 'PL', 'Mazowieckie');
```

```
Msg 2627, Level 14, State 1, Line 280
Violation of PRIMARY KEY constraint 'PK_DIM_CUST_44AE64B8B7F6F6A4'. Cannot insert duplicate key in object 'STROZIK.DIM_CUSTOMER'. The duplicate key value is (11001).
The statement has been terminated.

Completion time: 2023-03-31T11:16:12.7813261-07:00
```

Rekord nie został dodany, ponieważ nie mogą występować duplikaty klucza głównego.

```
INSERT INTO STROZIK.DIM_CUSTOMER (CustomerID, FirstName, LastName, Title, City, Terri-
toryName, CountryRegionCode, [Group])
VALUES (30119, 'Ewa', 'Nowak', 'Pani', 'Kraków', 'Małopolska', 'PL', 'Klient');
```

(1 row affected)

Completion time: 2023-03-31T11:29:33.5010661-07:00

Rekord został dodany, ponieważ nie ma takiego rekordu z atrybutem CustomerID o takiej wartości.

```
INSERT INTO STROZIK.DIM_PRODUCT (ProduktID, Name, ListPrice, Color, SubCategoryName,
CategoryName, Weight, Size, IsPurchased)
VALUES (1, 'Koszula', 89.99, 'Biały', 'Odzież', 'Moda', 0.2, 'XL', 1);
```

```
Msg 2627, Level 14, State 1, Line 286
Violation of PRIMARY KEY constraint 'PK_DIM_PROD_F1FF3022C151C5D4'. Cannot insert duplicate key in object 'STROZIK.DIM_PRODUCT'. The duplicate key value is (1).
The statement has been terminated.

Completion time: 2023-03-31T11:49:24.9346701-07:00
```

Rekord nie został dodany, ponieważ nie mogą występować duplikaty klucza głównego.

```
INSERT INTO STROZIK.DIM_PRODUCT (ProduktID, Name, ListPrice, Color, SubCategoryName,
CategoryName, Weight, Size, IsPurchased)
VALUES (1000, 'Koszula', 89.99, 'Biały', 'Odzież', 'Moda', 0.2, 'XL', 1);
```

(1 row affected)

Completion time: 2023-03-31T11:56:31.8098074-07:00

```
INSERT INTO STROZIK.DIM SALESPERSON (SalesPersonID, FirstName, LastName, Title, Gen-
der, CountryRegionCode, [Group])
VALUES (274, 'Filip', 'Strózik', 'Pan', 'M', 'PL', 'Dolnyslask');
```

Msg 2627, Level 14, State 1, Line 292  
Violation of PRIMARY KEY constraint 'PK\_DIM\_SALE\_7A591C1812A7D721'. Cannot insert duplicate key in object 'STROZIK.DIM SALESPERSON'. The duplicate key value is (274).  
The statement has been terminated.

Completion time: 2023-03-31T12:01:52.0125916-07:00

Podobnie jak powyżej.

```
INSERT INTO STROZIK.DIM SALESPERSON (SalesPersonID, FirstName, LastName, Title, Gen-
der, CountryRegionCode, [Group])
VALUES (1, 'Filip', 'Strózik', 'Pan', 'M', 'PL', 'Dolnyslask');
```

(1 row affected)

Completion time: 2023-03-31T12:10:54.1065642-07:00

## Wnioski:

Modelowanie wielowymiarowe jest kluczowe dla hurtowni danych i wymaga analizy dziedziny problemowej, identyfikacji procesów biznesowych oraz oceny dostępności i jakości źródeł danych.

W trakcie wykonywania zadań na bazie Adventure Works, należało stworzyć schemat o nazwie własnego nazwiska oraz table wymiarów i faktów. Tabele wymiarów dotyczyły klientów, produktów i sprzedawców, natomiast tabela faktów zawierała dane dotyczące sprzedaży.

Aby poprawnie utworzyć tabele, należało wykonać odpowiednie polecenia CREATE TABLE, które zawierały informacje typach atrybutów oraz dotyczących ich ograniczeń. W przypadku tak dużej bazy danych jak Adventure Works, wymagało to przestudiowania modelu projektu tabel, sprawdzenia zgodności typów oraz upewnienia się co do możliwości wstawienia wartości NULL. W przypadku wymiaru DIM\_CUSTOMER, w celu umożliwienia wprowadzania danych, należało ustawić atrybut CustomerID na automatycznie generowany, zaczynając od 1 i inkrementując o 1. Było to wymagane przez listę zapytaniem, które jako wynik dawało nierozróżnialne krotki ze względu na klucz główny.

Tworzenie schematu, tabel i dodawanie więzów integralności to podstawowe skrypty DDL i DML, które są niezbędne w procesie projektowania i tworzenia hurtowni danych.

Dodawanie więzów integralności, takich jak klucze obce czy ograniczenia unikalności, jest kluczowe w zapewnieniu integralności danych w hurtowni.

Wykonanie zadań z listy pozwoliło na zapoznanie się z podstawowymi skryptami DDL i DML oraz na zastosowanie ich podczas tworzenia tabel wymiarów i faktów hurtowni danych.

***Uwaga!***

- Sprawozdanie, bez wniosków podsumowujących aspekt zagadnień analizowanych na zajęciach laboratoryjnych i zawartych w sprawozdaniu, jest automatycznie oceniane negatywnie!