

HURTOWNIE DANYCH

Laboratorium 5

Maciej Kopiński 254578

Zad. 1. Przygotowanie powtarzalności procesu ETL

Przygotować instrukcję usuwającą każdą z tabel utworzonych w trakcie pracy nad listą 4.

Uwaga: Instrukcja powinna być wykonana tylko pod warunkiem istnienia usuwanej tabeli. Należy sprawdzić, czy dana tabela istnieje, używając instrukcji IF oraz informacji zawartych w widoku systemowym INFORMATION_SCHEMA.TABLES.

Zad. 2. Wymiar czasowy

Przygotować wymiar czasowy: utworzyć i wypełnić danymi tabelę DIM_TIME. Tabela DIM_TIME powinna być tabelą zawierającą wymiar czasowy (klucze obce do tej tabeli znajdują się w tabeli faktów).

Tabela DIM_TIME powinna zawierać następujące kolumny:

- PK_TIME (klucz główny – liczba całkowita postaci yyyymmdd – format taki sam jak kolumn OrderDate, ShipDate)
- Rok
- Kwartał
- Miesiąc
- Miesiąc słownie (wykorzystać tabelę pomocniczą z 12 rekordami dokonać odpowiedniego złączenia)
- Dzień tygodnia słownie (wykorzystać tabelę pomocniczą z 7 rekordami dokonać odpowiedniego złączenia)
- Dzień miesiąca

Zad. 3. Elementarne czyszczenie danych

Zamienić wszystkie wartości NULL:

- w kolumnie Color (tabela DIM_PRODUCT) na „Unknown”,
- w kolumnie SubCategoryName (tabela DIM_PRODUCT) na „Unknown”.
- w kolumnie CountryRegionCode na 000,
- w kolumnie Group na „Unknown”.

Zad. 4. Proces Extact – Transform - Load

Używając Visual Studio utworzyć projekt typu Integration Services (wybierając z Menu File -> New Project) zawierający zapytania SQL opracowane w zadaniach 1-4.1 z listy 4 oraz w zadaniach 1-3 z listy 5.

Uwaga: Do umieszczenia instrukcji SQL w treści pakietu użyć zadania Execute SQL Task (Menu – View – Toolbox).

Utworzony pakiet powinien działać sekwencyjnie i wykonywać następujące zadania:

- a) Usunąć tabele z przedrostkiem DIM i FACT (oczywiście usunąć tylko te, które istnieją),
- b) Utworzyć tabele z przedrostkiem DIM i FACT,
- c) Wypełnić tabele danymi (instrukcje INSERT INTO),
- d) Dodać więzy integralności z zadania 4.1 z listy 4 (bez sprawdzania poprawności integralności),
- e) Obsłużyć błędy i wyjątki – zakładka Event Handlers,
- f) Wysłać informację o pozytywnie zakończonym procesie na swój adres mailowy,

Uruchomić pakiet i sprawdzić poprawność jego działania.

Zad. 5. ETL (prawie) bez SQLa

Przygotować proces ETL analogiczny do opisanego w zad. 4. Import danych przygotować korzystając z narzędzi dostępnych w zakładce Data Flow, m.in. OLE DB Source/Destination, Merge Join, Sort, Derived Column, Fuzzy Lookup, Fuzzy Grouping, itp.

Rozwiązania:

Ad.1

```
IF OBJECT_ID(N'tempdb..#Dni') IS NOT NULL
BEGIN
DROP TABLE #Dni
END

IF OBJECT_ID(N'tempdb..#Miesiace') IS NOT NULL
BEGIN
DROP TABLE #Miesiace
END

IF EXISTS(SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME = 'FACT_SALES' AND
TABLE_SCHEMA = 'Kopinski')
DROP TABLE Kopinski.FACT_SALES;

IF EXISTS(SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME = 'DIM_CUSTOMER'
AND TABLE_SCHEMA = 'Kopinski')
DROP TABLE Kopinski.DIM_CUSTOMER;

IF EXISTS(SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME = 'DIM_PRODUCT' AND
TABLE_SCHEMA = 'Kopinski')
DROP TABLE Kopinski.DIM_PRODUCT;

IF EXISTS(SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME = 'DIM SALESPERSON'
AND TABLE_SCHEMA = 'Kopinski')
DROP TABLE Kopinski.DIM SALESPERSON;

IF EXISTS(SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME = 'DIM_TIME' AND
TABLE_SCHEMA = 'Kopinski')
DROP TABLE Kopinski.DIM_TIME;
```

Ad.2

```
CREATE TABLE Kopinski.DIM_TIME (
    PK_TIME INT NOT NULL,
    "Year" INT NOT NULL,
    "Quarter" INT NOT NULL,
    "Month" INT NOT NULL,
    "MonthInWords" NVARCHAR(50) NOT NULL,
    "DayOfWeekInWords" NVARCHAR(50) NOT NULL,
    "DayOfMonth" INT NOT NULL,
);

SET DATEFIRST 1;
SELECT DISTINCT DATEPART(dw, OrderDate) Numer,
CASE
    WHEN DATEPART(dw, OrderDate) = 1 THEN 'Poniedziałek'
    WHEN DATEPART(dw, OrderDate) = 2 THEN 'Wtorek'
    WHEN DATEPART(dw, OrderDate) = 3 THEN 'Środa'
    WHEN DATEPART(dw, OrderDate) = 4 THEN 'Czwartek'
    WHEN DATEPART(dw, OrderDate) = 5 THEN 'Piątek'
    WHEN DATEPART(dw, OrderDate) = 6 THEN 'Sobota'
    WHEN DATEPART(dw, OrderDate) = 7 THEN 'Niedziela'
END Nazwa
INTO #Dni
FROM Sales.SalesOrderHeader;
```

```

SELECT DISTINCT MONTH(OrderDate) Numer,
CASE
    WHEN MONTH(OrderDate) = 1 THEN 'Styczeń'
    WHEN MONTH(OrderDate) = 2 THEN 'Luty'
    WHEN MONTH(OrderDate) = 3 THEN 'Marzec'
    WHEN MONTH(OrderDate) = 4 THEN 'Kwiecień'
    WHEN MONTH(OrderDate) = 5 THEN 'Maj'
    WHEN MONTH(OrderDate) = 6 THEN 'Czerwiec'
    WHEN MONTH(OrderDate) = 7 THEN 'Lipiec'
    WHEN MONTH(OrderDate) = 8 THEN 'Sierpień'
    WHEN MONTH(OrderDate) = 9 THEN 'Wrzesień'
    WHEN MONTH(OrderDate) = 10 THEN 'Październik'
    WHEN MONTH(OrderDate) = 11 THEN 'Listopad'
    WHEN MONTH(OrderDate) = 12 THEN 'Grudzień'
END Nazwa
INTO #Miesiace
FROM Sales.SalesOrderHeader;

```

```

DECLARE @D INT;
SET @D = (SELECT TOP 1 OrderDate FROM Kopinski.FACT_SALES ORDER BY 1);
DECLARE @COUNTER DATE;
SET @COUNTER = CONVERT(date, CAST(@D AS nvarchar));
DECLARE @END INT;
SET @END = (SELECT TOP 1 ShipDate FROM Kopinski.FACT_SALES ORDER BY 1 DESC);
WHILE (@D <= @END)
BEGIN
    INSERT INTO Kopinski.DIM_TIME VALUES
    (
        @D,
        YEAR(@COUNTER),
        DATEPART(QQ, @COUNTER),
        MONTH(@COUNTER),
        (SELECT Nazwa FROM #Miesiace WHERE Numer = MONTH(@COUNTER)),
        (SELECT Nazwa FROM #Dni WHERE Numer = DATEPART(DW, @COUNTER)),
        DAY(@COUNTER)
    );
    SET @COUNTER = DATEADD(DAY, 1, @COUNTER);
    SET @D = CAST(CONVERT(varchar(8), @COUNTER, 112) AS INT);
END;

```

Ad.3

```

SELECT * FROM Kopinski.DIM_PRODUCT
UPDATE Kopinski.DIM_PRODUCT
SET Color='Unknown'
WHERE Color IS NULL;

UPDATE Kopinski.DIM_PRODUCT
SET SubCategoryName='Unknown'
WHERE SubCategoryName IS NULL;

UPDATE Kopinski.DIM_CUSTOMER
SET CountryRegionCode='000'
WHERE CountryRegionCode IS NULL;

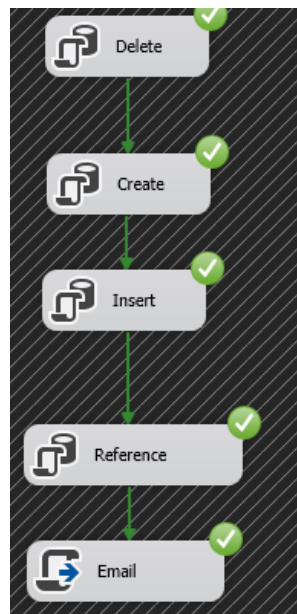
```

```
UPDATE Kopinski.DIM_CUSTOMER  
SET [Group]='Unknown'  
WHERE [Group] IS NULL;
```

```
UPDATE Kopinski.DIM SALESPERSON  
SET CountryRegionCode='000'  
WHERE CountryRegionCode IS NULL;
```

```
UPDATE Kopinski.DIM SALESPERSON  
SET [Group]='Unknown'  
WHERE [Group] IS NULL;
```

Ad.4

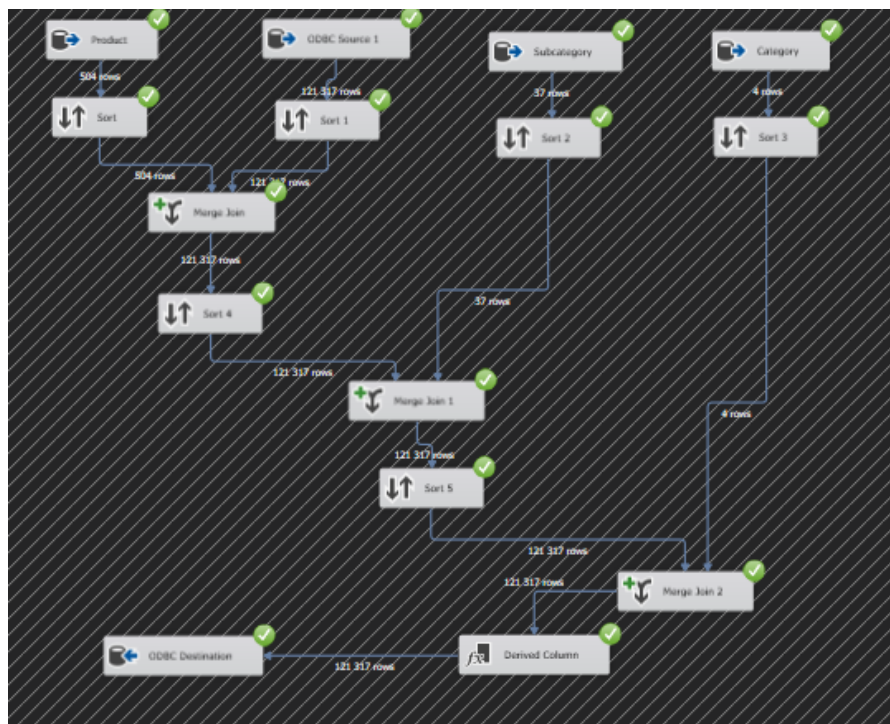
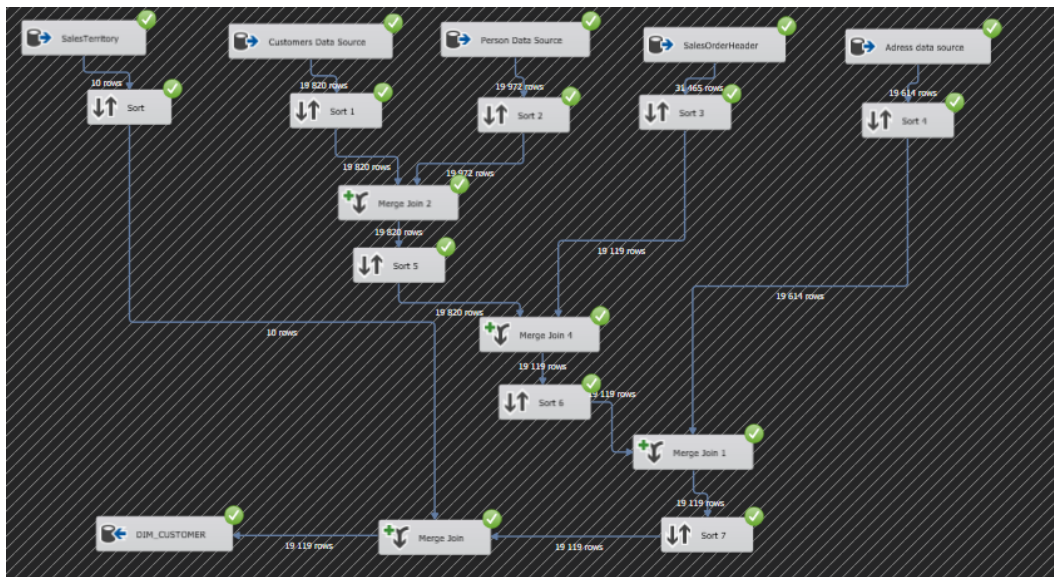
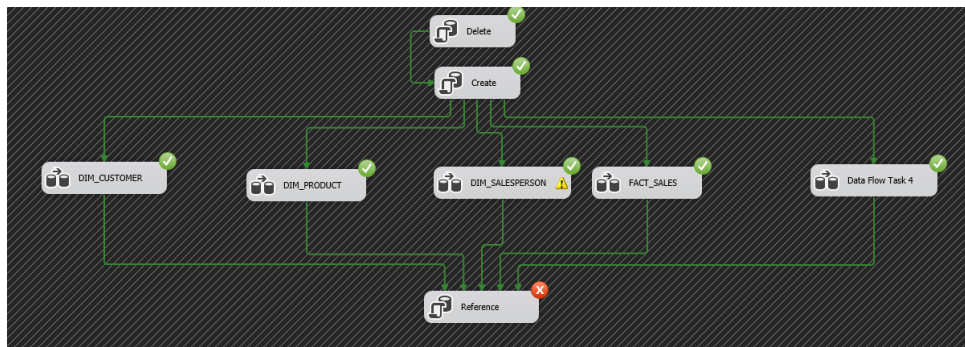


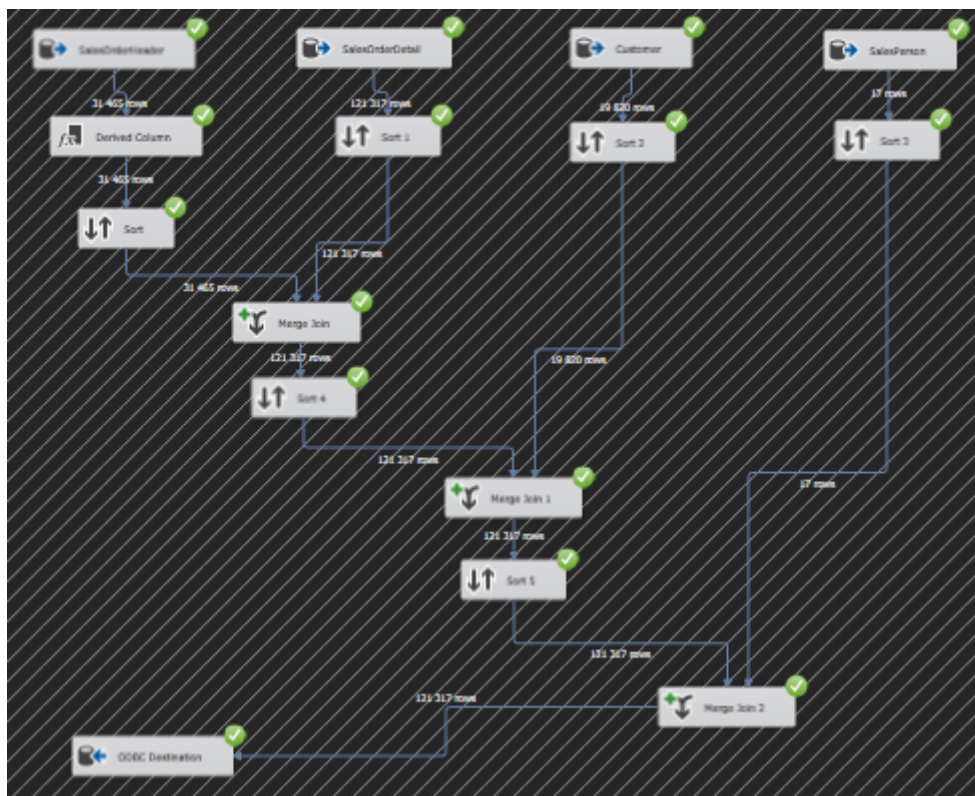
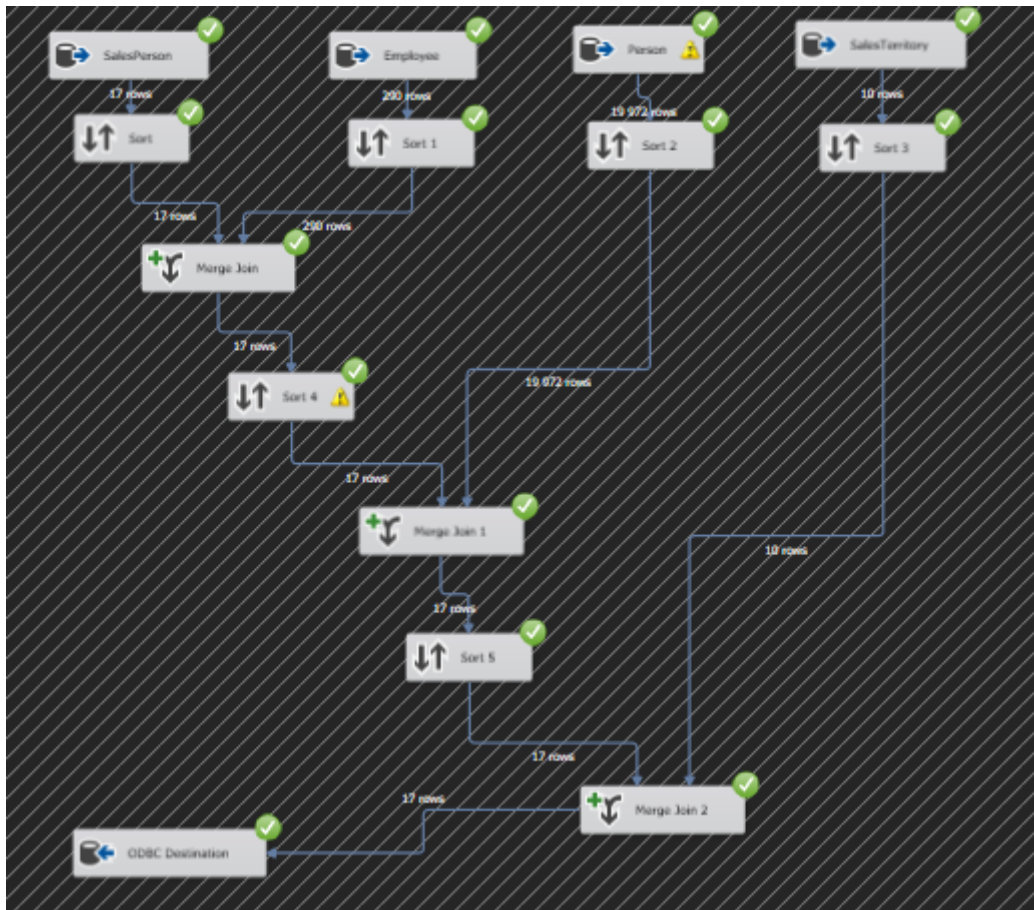
Hurtownie test [Z zewnątrz](#) [Odebrane x](#)



mackopin@gmail.com
Hurtownie test

Ad.5





Wnioski:

Zad.1

Do zadania 3 utworzyłem dodatkowe dwie tymczasowe tabele, więc sprawdzam czy one również istnieją. Jeśli tak – usuwam je.

Zad.2

W zadaniu 2 tworze dwie tymczasowe tabele, w celu umieszczenia w tabeli DIM_TIME nazw dni tygodnia oraz miesięcy. Dodawanie danych postanowiłem zrealizować poprzez przejście w pętli od pierwszej daty zamówienia do ostatniej daty dostawy, w przypadku gdy wystąpiłby dzień bez żadnych zamówień (Później sprawdziłem, że takich dni nie ma i wystarczy wziąć daty z SalesOrderHeader lub FACT_SALES).

Zad.3

- Istnieje 34 145 produktów bez koloru
- Nie ma produktów bez przypisanej podkategorii
- Nie ma klientów z pustym polem 'CountryRegionCode'
- Nie ma klientów z pustym polem 'Group'
- Istnieje 3 pracowników bez wpisanego pola 'Group' oraz 'CountryRegionCode'

Zad.4

W zadaniu 4 udało mi się usunąć, a następnie utworzyć, wypełnić, sprawdzić ograniczenia w tabelach oraz wysłać maila na swoje konto studenckie.

Zad.5

W zadaniu 5 nie udało mi się wypełnić danymi tabeli DIM_TIME. Było to spowodowane tym, że nie wiedziałem jak uzyskać dostęp do tabel tymczasowych, które wcześniej utworzyłem. Prawdopodobnie problem ten można rozwiązać tworząc permanentne tabele w bazie. Z tego powodu nie udaje się wykonać także ostatniej części polegającej na dodaniu kluczy głównych oraz obcych.