

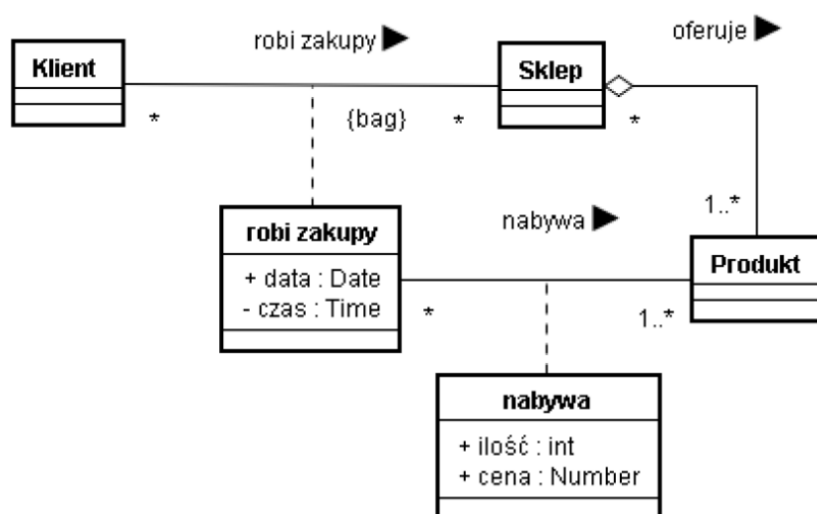
# Sprawozdanie z laboratorium Hurtownie Danych

Mikołaj Kubś, 272662

8 marca 2025

## 1 Zadanie 1

Analiza konceptualnego modelu danych "Usługi", który jest niekompletny, ale klasy i relacje między nimi reprezentują rozpatrywany wycinek rzeczywistości.



Rysunek 1: Konceptualny model danych "Usługi"

Reguły i ograniczenia dziedzinowe:

- Reg/01 - Klient może wielokrotnie robić zakupy w tym samym sklepie
- Reg/02 - W sklepie może robić zakupy dowolny klient

- Reg/03 - Każdy zakup realizowany jest przez klienta w sklepie w określonym dniu i godzinie
- Reg/04 - Sklep musi oferować co najmniej jeden produkt
- Reg/05 - ...

## 1.1 Weryfikacja i poprawa modelu danych

Ponieważ reguły są niekompletne i nie w pełni poprawne, zdecydowałem się wprowadzić szereg zmian.

Uznałem, że reguła "Reg/04 - Sklep musi oferować co najmniej jeden produkt" wprowadza niepotrzebną komplikację. Na przykład, według tej zasady, gdy sklep sprzedałby cały swój inwentarz, nie mógłby dalej istnieć w bazie danych.

Brakuje aktualnie informacji, jaka jest liczba dostępnego produktu w danym sklepie. Można wyciągnąć tą daną do nowej tabeli asocjacyjnej, do której można by dodatkowo dodać cenę produktu dla konkretnego sklepu, co zwiększa elastyczność na przyszłość i jest szeroko stosowaną praktyką w rzeczywistych sklepach. Tak więc sklep może oferować wiele produktów, każdy z własną ceną i ilością. Ponieważ tabela asocjacyjna "oferuje" ma cenę, można by usunąć cenę z tabeli asocjacyjnej "nabywa". Uznałem jednak, że ją zostawię, ponieważ cena oferty może się zmienić po zakupie produktu przez klienta. Sklepy mają często informację, że coś sprzedają, nawet jeśli nie ma tego chwilowo w magazynie, tak więc ustaliłem, że ilość oferowanego produktu to co najmniej 0, a nie koniecznie więcej niż zero, co wymuszałoby usunięcie oferty w przypadku braków w magazynie.

Można dodać parę atrybutów do encji. Do klienta dodam imię i nazwisko, a do produktu i do sklepu nazwę.

Warto również dodać standardowe ograniczenia wobec atrybutów encji. Zdecydowałem, że cena nie może być mniejsza od 0, ale może się mu równać - czasem są zaskakujące promocje w sklepach. Dodatkowo uznałem, że każdy produkt ma unikalną nazwę.

Oprócz tego dodałem reguły wynikające z diagramu klas, klaryfikujące, że:

1. Klient może robić zakupy w różnych sklepach
2. Ten sam produkt może być oferowany w wielu sklepach
3. Klient robiący zakupy musi zakupić przynajmniej 1 produkt

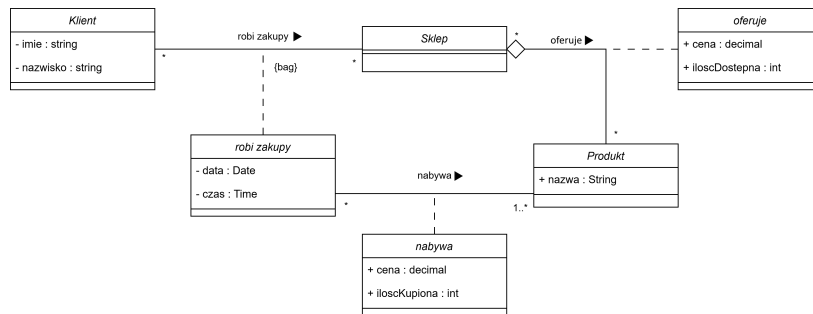
Dodatkowo zmieniłem typ atrybutów dotyczących kosztu na "decimal", a także zmieniłem atrybut "data" w "robi zakupy" na prywatny.

Na koniec dodałem regułę klaryfikującą, że wiele zakupów klienta nie może być zapisane do bazy danych w tym samym czasie.

## 1.2 Finalna postać reguł, ograniczeń i diagramu klas UML

Finalna postać reguł i ograniczeń:

- Reg/01 - Klient może wielokrotnie robić zakupy w tym samym sklepie
- Reg/02 - W sklepie może robić zakupy dowolny klient
- Reg/03 - Każdy zakup realizowany jest przez klienta w sklepie w określonym dniu i godzinie
- Reg/04 - Każdy sklep ustala własną cenę oraz ilość oferowanego produktu
- Reg/05 - Klient nabywając produkt w danym sklepie kupuje go za cenę oferowaną w sklepie, która zostaje zapamiętana
- Reg/06 - Sklep może zmienić cenę oferowanego produktu, co wpływa tylko na przyszłe zakupy
- Reg/07 - Klient może robić zakupy w różnych sklepach
- Reg/08 - Ten sam produkt może być oferowany w wielu sklepach
- Reg/09 - Klient robiąc zakupy musi nabyć co najmniej 1 produkt
- Reg/10 - Imię klienta nie może być puste
- Reg/11 - Nazwisko klienta nie może być puste
- Reg/12 - Ilość oferowanego produktu nie może być mniejsza od zera
- Reg/13 - Cena oferowanego produktu nie może być mniejsza od zera
- Reg/14 - Ilość nabytego produktu musi być większa od zera
- Reg/15 - Cena nabytego produktu nie może być mniejsza od zera
- Reg/16 - Nazwa produktu nie może się powtarzać
- Reg/17 - Klient nie może robić zakupów więcej niż raz o tym samym czasie i dacie.



Rysunek 2: Finalny model danych "Usługi"

### 1.3 Skrypt DDL SQL

```

1
2 CREATE DATABASE Uslugi;
3 GO
4
5 USE Uslugi;
6
7 DROP TABLE IF EXISTS Nabycie;
8 DROP TABLE IF EXISTS Zakup;
9 DROP TABLE IF EXISTS Oferta;
10 DROP TABLE IF EXISTS Produkt;
11 DROP TABLE IF EXISTS Sklep;
12 DROP TABLE IF EXISTS Klient;
13
14 CREATE TABLE Klient (
15     id INT PRIMARY KEY,
16     imie VARCHAR(255) NOT NULL,
17     nazwisko VARCHAR(255) NOT NULL
18 );
19
20 CREATE TABLE Sklep (
21     id INT PRIMARY KEY,
22     nazwa VARCHAR(255) NOT NULL
23 );
24
25 CREATE TABLE Produkt (
26     id INT PRIMARY KEY,
27     nazwa VARCHAR(255) NOT NULL UNIQUE

```

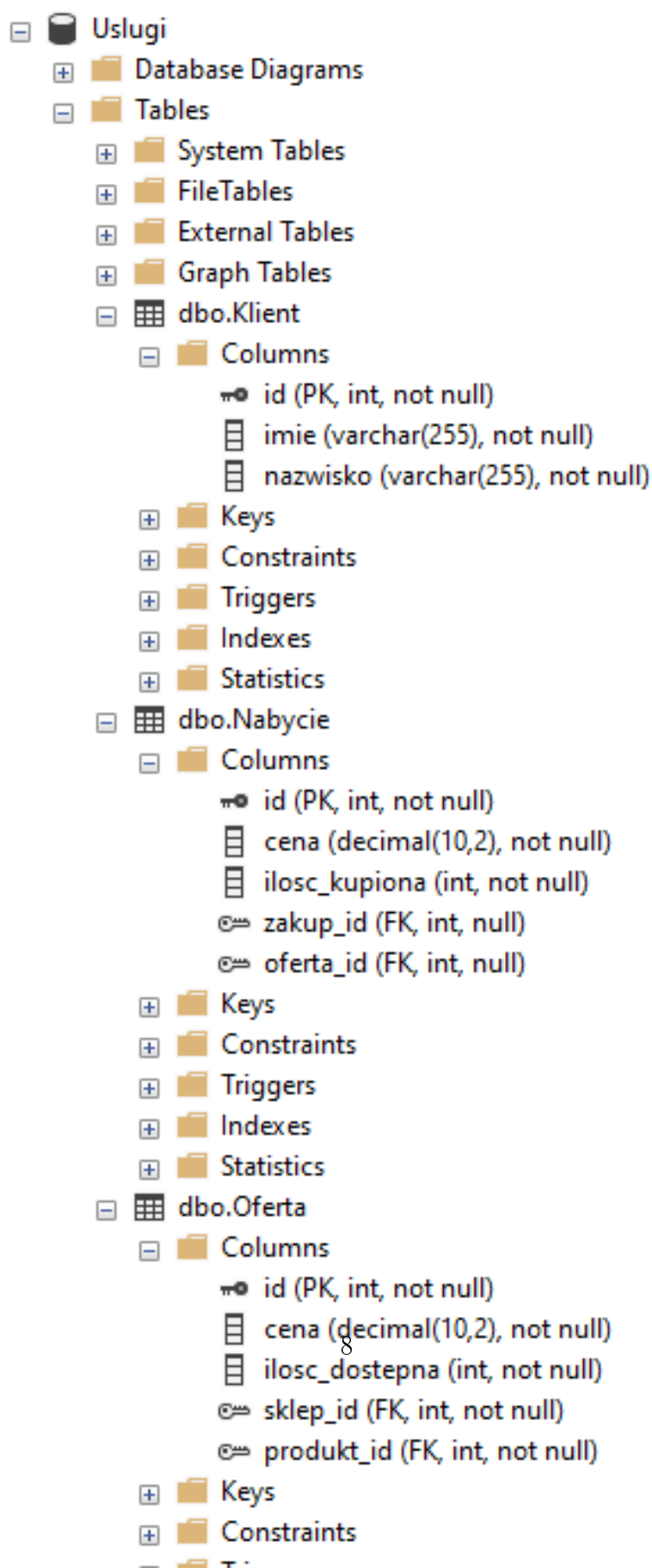
```

28 );
29
30 CREATE TABLE Oferta (
31     id INT PRIMARY KEY,
32     cena DECIMAL(10,2) NOT NULL CHECK (cena >= 0),
33     ilosc_dostepna INT NOT NULL CHECK (ilosc_dostepna >= 0),
34     sklep_id INT NOT NULL,
35     produkt_id INT NOT NULL,
36     FOREIGN KEY (sklep_id) REFERENCES Sklep(id) ON DELETE CASCADE,
37     FOREIGN KEY (produkt_id) REFERENCES Produkt(id) ON DELETE CASCADE
38 );
39
40 CREATE TABLE Zakup (
41     id INT PRIMARY KEY,
42     data DATE NOT NULL,
43     czas TIME NOT NULL,
44     klient_id INT,
45     sklep_id INT NOT NULL,
46     FOREIGN KEY (klient_id) REFERENCES Klient(id) ON DELETE SET NULL,
47     FOREIGN KEY (sklep_id) REFERENCES Sklep(id) ON DELETE CASCADE,
48     CONSTRAINT unique_zakup UNIQUE (data, czas, klient_id, sklep_id)
49 );
50
51 CREATE TABLE Nabycie (
52     id INT PRIMARY KEY,
53     cena DECIMAL(10,2) NOT NULL CHECK (cena >= 0),
54     ilosc_kupiona INT NOT NULL CHECK (ilosc_kupiona > 0),
55     zakup_id INT,
56     oferta_id INT,
57     FOREIGN KEY (zakup_id) REFERENCES Zakup(id) ON DELETE SET NULL,
58     FOREIGN KEY (oferta_id) REFERENCES Oferta(id)
59 );

```



## 1.4 Inicjalizacja bazy danych w systemie MS SQL





## 1.5 Testy działania bazy danych

```
Messages
(4 rows affected)
(3 rows affected)
(5 rows affected)
(3 rows affected)
(3 rows affected)
(3 rows affected)
(3 rows affected)
(3 rows affected)
(3 rows affected)
(5 rows affected)
(3 rows affected)
(4 rows affected)
Msg 847, Level 16, State 5, Line 43
The INSERT statement conflicted with the CHECK constraint "CK_Oferta_cena_94CD0497". The conflict occurred in database "Uslugi", table "dbo.Oferta", column 'cena'.
The statement has been terminated.
Msg 847, Level 16, State 5, Line 45
The INSERT statement conflicted with the CHECK constraint "CK_Oferta_ilosc_do_07C12930". The conflict occurred in database "Uslugi", table "dbo.Oferta", column 'ilosc_dostepna'.
The statement has been terminated.
Msg 847, Level 16, State 5, Line 47
The INSERT statement conflicted with the CHECK constraint "CK_Magazyn_ilosc_k_128E7A35". The conflict occurred in database "Uslugi", table "dbo.Magazyn", column 'ilosc_kupiona'.
The statement has been terminated.
Msg 847, Level 16, State 5, Line 49
The INSERT statement conflicted with the CHECK constraint "CK_Magazyn_cena_114A936A". The conflict occurred in database "Uslugi", table "dbo.Magazyn", column 'cena'.
The statement has been terminated.
Msg 2627, Level 14, State 1, Line 51
Violation of UNIQUE KEY constraint 'unique_zakup'. Cannot insert duplicate key in object 'dbo.zakup'. The duplicate key value is (2025-05-07, 10:50:00.0000000, 1, 1).
The statement has been terminated.
Completion time: 2025-05-07T20:29:31.894861+01:00
```

Rysunek 4: Widok konsoli po wykonaniu poniższego kodu

```
1
2 INSERT INTO Klient (id, imie, nazwisko) VALUES
3 (1, 'Jan', 'Kowalski'),
4 (2, 'Anna', 'Nowak'),
5 (3, 'Piotr', 'Wójcik');
6
7 INSERT INTO Sklep (id, nazwa) VALUES
8 (1, 'Biedronka'),
9 (2, 'Biedronka'),
10 (3, 'BIEDRONKA_pl._GRUNWALDZKI');
11
12 INSERT INTO Produkt (id, nazwa) VALUES
13 (1, 'Masło'),
14 (2, 'Bułki'),
15 (3, 'Wędliny');
16
17 INSERT INTO Oferta (id, cena, ilosc_dostepna, sklep_id, produkt_id) VALUES
18 (1, 25.99, 10, 1, 1),
19 (2, 45.50, 5, 1, 2),
20 (3, 15.75, 15, 2, 1),
21 (4, 35.30, 7, 2, 3),
22 (5, 99.99, 0, 3, 2);
23
24 INSERT INTO Zakup (id, data, czas, klient_id, sklep_id) VALUES
```

```

25 (1, '2025-03-07', '10:30:00', 1, 1),
26 (2, '2025-03-07', '11:00:00', 2, 2),
27 (3, '2025-03-07', '12:00:00', 3, 3);
28
29 INSERT INTO Nabycie (id, cena, ilosc_kupiona, zakup_id, oferta_id) VALUES
30 (1, 25.99, 3, 1, 1),
31 (2, 45.50, 2, 2, 2),
32 (3, 15.75, 5, 3, 3),
33 (4, 35.30, 1, 3, 4);
34
35 -- Invalid operations
36
37 INSERT INTO Oferta (id, cena, ilosc_dostepna, sklep_id, produkt_id) VALUES (6, -
38
39 INSERT INTO Oferta (id, cena, ilosc_dostepna, sklep_id, produkt_id) VALUES (7, 2
40
41 INSERT INTO Nabycie (id, cena, ilosc_kupiona, zakup_id, oferta_id) VALUES (5, 25
42
43 INSERT INTO Nabycie (id, cena, ilosc_kupiona, zakup_id, oferta_id) VALUES (5, -2
44
45 INSERT INTO Zakup (id, data, czas, klient_id, sklep_id) VALUES (4, '2025-03-07',

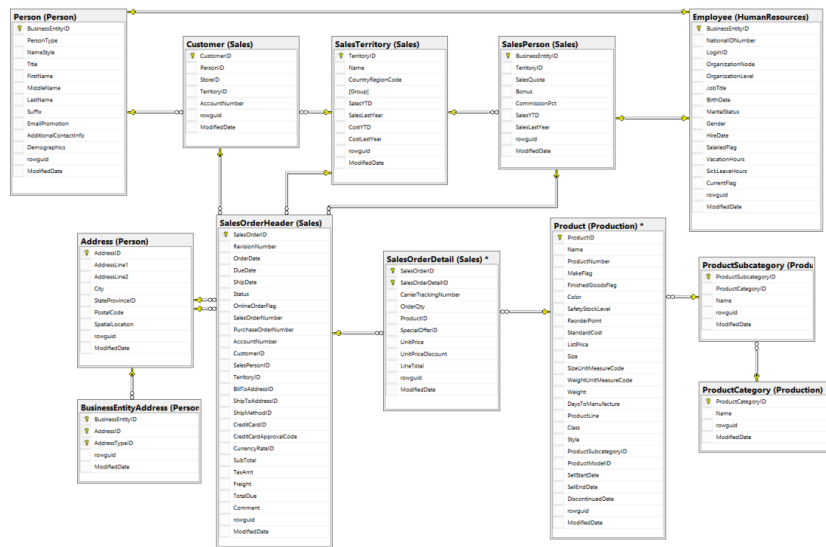
```

Wszystkie operacje zadziałały, oprócz tych, które naruszały ograniczenia. Błędy w konsoli jasno opisują, które konkretnie ograniczenie zostało złamane. Tymi kwerendami sprawdzone zostały wszystkie ograniczenia.

## 2 Zadanie 2

### 2.1 Opis zadania

Do wykonania zadania użyłem bazy danych "AdventureWorks2014" (<https://github.com/Microsoft/sql-server-samples/releases/download/adventureworks/AdventureWorks2014.bak>)



Rysunek 5: Wykorzystane w rozwiązaniu tabele

Kwerendy do wykonania:

1. Ile jest produktów w bazie? Ile kategorii i podkategorii?
2. Wypisz produkty, które nie mają zdefiniowanego koloru.
3. Podaj roczną kwotę transakcji (SalesOrderHeader.TotalDue) w poszczególnych latach.
4. Ilu jest klientów, a ilu sprzedawców w sklepie? Ilu w poszczególnych regionach?
5. Ile było wykonanych transakcji w poszczególnych latach?
6. Podaj produkty, które nie zostały kupione przez żadnego klienta. Zestawienie pogrupuj według kategorii i podkategorii.
7. Oblicz minimalną i maksymalną kwotę rabatu udzielonego na produkty w poszczególnych podkategoriach.
8. Podaj produkty, których cena jest wyższa od średniej ceny produktów w sklepie.
9. Ile średnio produktów w każdej kategorii sprzedaje się w poszczególnych miesiącach?
10. Ile średnio czasu klient czeka na dostawę zamówionych produktów? Przygotuj zestawienie w zależności od kodu regionu (SalesTerritory.CountryRegionCode).

## 2.2 Kwerendy

SELECT COUNT(\*) AS "Liczba produktów" FROM Production.Product;

SELECT COUNT(\*) AS "Liczba kategorii" FROM Production.ProductCategory;

SELECT COUNT(\*) AS "Liczba podkategorii" FROM Production.ProductSubcategory;

100 %

ResultsMessages

Liczba produktów

1504

Liczba kategorii

14

Liczba podkategorii

137

Rysunek 6: Wynik kwerendy 1

SELECT \* FROM Production.Product WHERE Color IS NULL

100 %

ResultsMessages

ProductID	Name	ProductNumber	MakeFlag	FreshnessFlag	Color	SafetyStockLevel	ReorderPoint	StandardCost	ListPrice	Size	SizeUnitMeasureCode	WeightUnitMeasureCode	Weight	DaysToManufacture	ProductLine	Class	Style	Product
1	Adjustable Floor	AB-0301	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
2	Bearing Ball	BA-0327	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
3	BB Ball Bearing	BE-0349	1	0	NULL	800	600	0.00	0.00	NULL	NULL	NULL	NULL	1	NULL	NULL	NULL	NULL
4	Hexnut Ball Bearing	BE-0360	0	0	NULL	800	600	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
5	Blade	BL-2036	1	0	NULL	800	600	0.00	0.00	NULL	NULL	NULL	NULL	1	NULL	NULL	NULL	NULL
6	Crown Race	CR-0801	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
7	Chain Sprocket	CS-0812	1	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	1	NULL	NULL	NULL	NULL
8	Decal 1	DC-0732	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
9	Decal 2	DC-0824	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
10	Down Tube	DT-0377	1	0	NULL	800	600	0.00	0.00	NULL	NULL	NULL	NULL	1	NULL	NULL	NULL	NULL
11	Mountain End Cap	EC-0052	1	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	1	NULL	NULL	NULL	NULL
12	Road End Cap	EC-0098	1	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	1	NULL	NULL	NULL	NULL
13	Roasting End Cap	EC-1209	1	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	1	NULL	NULL	NULL	NULL
14	Fork End	FE-3760	1	0	NULL	800	600	0.00	0.00	NULL	NULL	NULL	NULL	1	NULL	NULL	NULL	NULL
15	Rat Washer 1	FW-1000	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
16	Rat Washer 6	FW-1200	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
17	Rat Washer 2	FW-1400	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
18	Rat Washer 3	FW-1400	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
19	Rat Washer 4	FW-3000	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
20	Rat Washer 3	FW-5100	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
21	Rat Washer 5	FW-5000	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
22	Rat Washer 5	FW-7160	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
23	Rat Washer 7	FW-5100	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
24	Fork Cover	FC-2054	1	0	NULL	800	600	0.00	0.00	NULL	NULL	NULL	NULL	1	NULL	NULL	NULL	NULL
25	Guide Pulley	GP-0802	0	0	NULL	800	600	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
26	LL Grip Tape	GT-0020	0	0	NULL	800	600	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
27	HL Grip Tape	GT-1209	0	0	NULL	800	600	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
28	HL Grip Tape	GT-2909	0	0	NULL	800	600	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
29	Thin-Jaw Hex Nut 9	HJ-1213	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
30	Thin-Jaw Hex Nut 10	HJ-1200	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
31	Thin-Jaw Hex Nut 1	HJ-1420	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
32	Thin-Jaw Hex Nut 2	HJ-1420	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
33	Thin-Jaw Hex Nut 15	HJ-3410	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
34	Thin-Jaw Hex Nut 16	HJ-3416	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
35	Thin-Jaw Hex Nut 5	HJ-3416	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
36	Thin-Jaw Hex Nut 6	HJ-3824	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
37	Thin-Jaw Hex Nut 3	HJ-5161	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
38	Thin-Jaw Hex Nut 4	HJ-5162	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
39	Thin-Jaw Hex Nut 13	HJ-5811	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL
40	Thin-Jaw Hex Nut 14	HJ-5813	0	0	NULL	1000	750	0.00	0.00	NULL	NULL	NULL	NULL	0	NULL	NULL	NULL	NULL

Query executed successfully.

DESKTOP-20LE4BP (15.0 RTM) | DESKTOP-20LE4BP\sa... | AdventureWorks2014 | 00:00:00 | 248 rows

Rysunek 7: Wynik kwerendy 2

12

```

SELECT SUM(TotalDue) AS "Roczna kwota transakcji", YEAR(OrderDate) AS "Rok"
FROM Sales.SalesOrderHeader
GROUP BY YEAR(OrderDate)
ORDER BY YEAR(OrderDate)

```

	Roczna kwota transakcji	Rok
1	14155699,525	2011
2	37675700,312	2012
3	48965887,9632	2013
4	22419498,3157	2014

Rysunek 8: Wynik kwerendy 3

```

-- Ilu jest klientów, a ilu sprzedawców w sklepie? Ilu w poszczególnych regionach?

SELECT COUNT(*) AS "Liczba klientów"
FROM Sales.Customer;

SELECT COUNT(*) AS "Liczba sprzedawców"
FROM Sales.SalesPerson;

SELECT COUNT(*) AS "Liczba klientów", SalesTerritory.Name AS "Terytorium"
FROM Sales.Customer
JOIN Sales.SalesTerritory ON Customer.TerritoryID = SalesTerritory.TerritoryID
GROUP BY SalesTerritory.Name
ORDER BY SalesTerritory.Name;

SELECT COUNT(*) AS "Liczba sprzedawców", SalesTerritory.Name AS "Terytorium"
FROM Sales.SalesPerson
JOIN Sales.SalesTerritory ON SalesPerson.TerritoryID = SalesTerritory.TerritoryID

```

	Liczba klientów
1	19820

	Liczba sprzedawców
1	17

	Liczba klientów	Terytorium
1	3665	Australia
2	1791	Canada
3	132	Central
4	1884	France
5	1852	Germany
6	113	Northeast
7	3520	Northwest
8	176	Southeast
9	4696	Southwest
10	1991	United K...

	Liczba sprzedawców	Terytorium
1	1	Australia
2	2	Canada
3	1	Central
4	1	France
5	1	Germany
6	1	Northeast
7	3	Northwest
8	1	Southeast
9	2	Southwest
10	1	United K...

Rysunek 9: Wynik kwerendy 4

<pre> SELECT COUNT(*) AS "Roczna liczba transakcji", YEAR(OrderDate) AS "Rok" FROM Sales.SalesOrderHeader GROUP BY YEAR(OrderDate) ORDER BY YEAR(OrderDate) </pre>		
100 %		
Results Messages		
	Roczna liczba transakcji	Rok
1	1607	2011
2	3915	2012
3	14182	2013
4	11761	2014

Rysunek 10: Wynik kwerendy 5

<pre> -- Podaj produkty, które nie zostały kupione przez żadnego klienta. Zestawienie pogrupuj według kategorii i podkategorii. SELECT ProductCategory.Name AS "Kategoria", ProductSubcategory.Name AS "Podkategoria", Product.Name AS "Nazwa produktu", ProductNumber AS "Numer produktu" FROM Production.Product LEFT JOIN Production.ProductSubcategory ON ProductSubcategory.ProductSubcategoryID = Product.ProductSubcategoryID LEFT JOIN Sales.SalesOrderDetail ON SalesOrderDetail.ProductID = Product.ProductID WHERE SalesOrderDetail.SalesOrderDetailID IS NULL GROUP BY ProductCategory.Name, ProductSubcategory.Name, Product.Name ORDER BY ProductCategory.Name DESC, ProductSubcategory.Name, Product.Name </pre>			
100 %			
Results Messages			
Kategoria	Podkategoria	Nazwa produktu	Numer produktu
Components	Button Brackets	ML Button Bracket	BB-0107
Components	Forks	ML Fork	FK-0138
Components	Handlebars	ML Road Handlebars	HB-R720
Components	Mountain Frames	HL Mountain Frame - Black	FR-M048-46
Components	Mountain Frames	HL Mountain Frame - Silver	FR-M048-44
Components	Road Frames	HL Road Frame - Black	FR-R026-52
Components	Road Frames	HL Road Frame - Black	FR-R026-58
Components	Road Frames	HL Road Frame - Black	FR-R026-62
Components	Road Frames	HL Road Frame - Red	FR-R029-52
Components	Road Frames	HL Road Frame - Red	FR-R029-56
Components	Road Frames	HL Road Frame - Red	FR-R029-58
Components	Road Frames	LL Road Frame - Black	FR-R038-48
Components	Road Frames	LL Road Frame - Black	FR-R038-62
Components	Road Frames	LL Road Frame - Red	FR-R039-58
Components	Road Frames	ML Road Frame - Red	FR-R129-44
Components	Road Frames	ML Road Frame - Red	FR-R129-58
Components	Road Frames	ML Road Frame - Red	FR-R129-60
Components	Road Frames	ML Road Frame-W - Yellow	FR-R129-42
Components	Saddles	ML Road Saddle	SE-R005
Components	Touring Frames	LL Touring Frame - Yellow	FR-T071-54
Components	Wheels	LL Road Front Wheel	FW-F023
Components	Wheels	Touring Front Wheel	FW-T005
Components	Wheels	Touring Rear Wheel	RW-T005
Clothing	Shirts	Men's Sports Shirt - XL	SH-M007-X
Accessories	Lights	Headlights - Dual Beam	LT-H002
Accessories	Lights	Headlights - Windproof	LT-H003
Accessories	Lights	Tailights - Battery Powered	LT-T006
Accessories	Panniers	Touring Panniers, Large	PA-T005
Accessories	Pumps	Mountain Pump	PU-M004
NULL	NULL	Adjustable Race	AR-A001
NULL	NULL	BB Ball Bearing	BE-2349
NULL	NULL	Bearing Ball	BA-B027
NULL	NULL	Brake	BL-0016

Rysunek 11: Wynik kwerendy 6

```
-- Oblicz minimalną i maksymalną kwotę rabatu udzielonego na produkty w poszczególnych podkategoriach.
SELECT ProductSubcategory Name AS "Podkategoria", MIN(SalesOrderDetail.UnitPriceDiscount) AS "Minimalny rabat", MAX(SalesOrderDetail.UnitPriceDiscount) AS "Maksymalny rabat"
FROM Production.Product
JOIN Production.ProductSubcategory ON ProductSubcategory.ProductSubcategoryID = Product.ProductSubcategoryID
JOIN SalesOrderDetail ON SalesOrderDetail.ProductID = Product.ProductID
GROUP BY ProductSubcategory Name
ORDER BY MAX(SalesOrderDetail.UnitPriceDiscount) DESC
```

	Podkategoria	Minimalny rabat	Maksymalny rabat
1	Mountain Bikes	0.00	0.40
2	Road Bikes	0.00	0.30
3	Young Bikes	0.00	0.20
4	Gloves	0.00	0.15
5	Shirts	0.00	0.15
6	Helmet	0.00	0.15
7	Socks	0.00	0.10
8	Tights	0.00	0.10
9	Wets	0.00	0.10
10	Bike Racks	0.00	0.10
11	Bellows and Caps	0.00	0.10
12	Choppers	0.00	0.10
13	Caps	0.00	0.10
14	Jersey	0.00	0.10
15	Young Frames	0.00	0.05
16	Wheels	0.00	0.05
17	Bell-Shims	0.00	0.05
18	Headsets	0.00	0.05
19	Road Frames	0.00	0.05
20	Handlebars	0.00	0.05
21	Hydration Packs	0.00	0.05
22	Locks	0.00	0.02
23	Pumps	0.00	0.02
24	Tires and Tubes	0.00	0.02
25	Baskets	0.00	0.02
26	Chairs	0.00	0.02
27	Cranksets	0.00	0.02
28	Cranksets	0.00	0.02
29	Forks	0.00	0.02
30	Saddles	0.00	0.02
31	Mountain Frames	0.00	0.02
32	Pedals	0.00	0.02
33	Bottom Brackets	0.00	0.00
34	Bike Stands	0.00	0.00

Query executed successfully. DESKTOP-2GLE4BP (11.0 RTM) DESKTOP-2GLE4BP\janta... AdventureWorks2014 00:00:00 31 rows

Rysunek 12: Wynik kwerendy 7

```
-- Podać produkty, których cena jest wyższa od średniej ceny produktów w sklepie.
SELECT Name, ListPrice
FROM Production.Product
WHERE ListPrice > (
    SELECT AVG(ListPrice)
    FROM Production.Product
)
ORDER BY ListPrice
```

	Name	ListPrice
1	Road-750 Black, 58	539.99
2	Mountain-500 Black, 40	539.99
3	Mountain-500 Black, 42	539.99
4	Mountain-500 Black, 44	539.99
5	Mountain-500 Black, 46	539.99
6	Mountain-500 Black, 52	539.99
7	Road-750 Black, 44	539.99
8	Road-750 Black, 40	539.99
9	Road-750 Black, 52	539.99
10	Mountain-500 Silver, 40	564.99
11	Mountain-500 Silver, 42	564.99
12	Mountain-500 Silver, 44	564.99
13	Mountain-500 Silver, 46	564.99
14	Mountain-500 Silver, 52	564.99
15	ML Road Frame - Road, 44	594.83
16	ML Road Frame - Road, 40	594.83
17	ML Road Frame - Road, 52	594.83
18	ML Road Frame - Road, 58	594.83
19	ML Road Frame - Road, 63	594.83
20	ML Road Frame W - Yellow, 38	594.83
21	ML Road Frame W - Yellow, 40	594.83
22	ML Road Frame W - Yellow, 42	594.83
23	ML Road Frame W - Yellow, 44	594.83
24	ML Road Frame W - Yellow, 48	594.83
25	Touring-3000 Blue, 44	742.35
26	Touring-3000 Blue, 50	742.35
27	Touring-3000 Blue, 54	742.35
28	Touring-3000 Blue, 58	742.35
29	Touring-3000 Blue, 62	742.35
30	Touring-3000 Yellow, 44	742.35
31	Touring-3000 Yellow, 50	742.35
32	Touring-3000 Yellow, 54	742.35
33	Touring-3000 Yellow, 58	742.35

Query executed successfully. DESKTOP-2GLE4BP (11.0 RTM) DESKTOP-2GLE4BP\janta... AdventureWorks2014 00:00:00 33 rows

Rysunek 13: Wynik kwerendy 8

```
-- Ile średnio produktów w każdej kategorii sprzedaje się w poszczególnych miesiącach?
```

```
WITH MonthlySales AS (
    SELECT Production.ProductCategory.Name AS Category, MONTH(Sales.SalesOrderHeader.OrderDate) AS Month, SUM(Sales.SalesOrderDetail.OrderQty) AS SoldUnitsCount
    FROM Sales.SalesOrderDetail
    JOIN Sales.SalesOrderHeader ON Sales.SalesOrderDetail.SalesOrderID = Sales.SalesOrderHeader.SalesOrderID
    JOIN Production.Product ON Sales.SalesOrderDetail.ProductID = Production.Product.ProductID
    JOIN Production.ProductSubcategory ON Production.Product.ProductID = Production.ProductSubcategory.ProductSubcategoryID
    JOIN Production.ProductCategory ON Production.ProductSubcategory.ProductSubcategoryID = Production.ProductCategory.ProductCategoryID
    GROUP BY Production.ProductCategory.Name, MONTH(Sales.SalesOrderHeader.OrderDate)
)
SELECT Category AS "Kategoria", Month AS "Miesiąc", AVG(SoldUnitsCount) AS "Średnia sprzedaż"
FROM MonthlySales
GROUP BY Category, Month
ORDER BY Category, Month
```

Kategoria	Miesiąc	Średnia sprzedaż
Accessories	1	4302
Accessories	2	3412
Accessories	3	7178
Accessories	4	4190
Accessories	5	7005
Accessories	6	5776
Accessories	7	3803
Accessories	8	4761
Accessories	9	4902
Accessories	10	5503
Accessories	11	4176
Accessories	12	4144
Bikes	1	5463
Bikes	2	3903
Bikes	3	11305
Bikes	4	4762
Bikes	5	10175
Bikes	6	8076
Bikes	7	4402
Bikes	8	6553
Bikes	9	7210
Bikes	10	9542
Bikes	11	4740
Bikes	12	7128
Clothing	1	4462
Clothing	2	2003
Clothing	3	3409
Clothing	4	3324
Clothing	5	4762

Rysunek 14: Wynik kwerendy 9

```
-- Ile średnio czasu klient czeka na dostawę zamówionych produktów? Przygotuj zestawienie w zależności od kodu regionu (SalesTerritory.CountryRegionCode).
```

```
SELECT AVG(CAST(OrderDate - ShipDate AS FLOAT)) AS "Średni czas dostawy w dniach", SalesTerritory.CountryRegionCode AS "Kod regionu"
FROM Sales.SalesOrderHeader
JOIN Sales.SalesTerritory ON SalesTerritory.TerritoryID = SalesOrderHeader.TerritoryID
GROUP BY SalesTerritory.CountryRegionCode
ORDER BY SalesTerritory.CountryRegionCode
```

Kod regionu	Średni czas dostawy w dniach
AU	7
CA	7
DE	7,00343118558267
FR	7
GB	7
US	7

Rysunek 15: Wynik kwerendy 10

## 3 Wnioski

### 3.1 Wnioski z zadania 1

Zadanie okazało się znacznie bardziej skomplikowane, niż mogło by się wydawać na pierwszy rzut oka. Im więcej czasu spędzi się nad takim z pozoru prostym diagramem, tym więcej można dodać reguł do uściślenia zasad. Bez jasnych wytycznych klienta można też było przyjąć zupełnie inne założenia wobec bazy danych. Aktualna wersja bazy danych jest dość abstrakcyjna -



nie jest jasne, czy dotyczy tylko sklepów internetowych, czy stacjonarnych, czy obu naraz. Moim zamysłem było nie ograniczać za bardzo możliwości, tak więc aktualną tabelę można by wykorzystać do obu rodzajów sklepów.

W przykładowych danych, którymi zasiałem bazę, widać już pewne usprawnienie - prawdopodobnie w przyszłości można by wyciągnąć kategorię sklepu do osobnej tabeli, aby, na przykład, mieć pewność, że wszystkie sklepy konkretnej marki będą łatwo dostępne. Wtedy każdy z tych sklepów byłby np. kategorii "Biedronka". Można by przekształcić też kategorię w kierunku opisu innych właściwości sklepu, np. czy jest internetowy, czy stacjonarny.

Podsumowując, liczba reguł znacznie wzrosła, ale bez jasnych wytycznych klienta warto ograniczyć się w ich ilości, żeby zezwalać na więcej możliwości w przyszłości. Oczywiście, jedyną pewną rzeczą w oprogramowaniu jest ciągła zmiana, ale nowoczesne narzędzia bazodanowe są w stanie bardzo wspomóc migracje do nowych wytycznych. Tak więc nie trzeba przesadnie "bać" się różnych pomysłów w początkowej fazie projektowania, byleby je dobrze i ciągle konsultować z klientem.

### 3.2 Wnioski z zadania 2

Pewne jest to, że te 10 kwerend to było zdecydowanie za mało, by dowiedzieć się wszystkiego o stanie omawianej firmy. To też za mało, żeby próbować zrozumieć czynniki wpływające na taką, a nie inną, sytuację. Ale mimo tego, można i tak wywnioskować ciekawe informacje.

Oferta sklepu wydaje się dość różnorodna i logicznie podzielona na kategorie i podkategorie. Sklep z każdym rokiem wydaje się zarabiać coraz więcej (mogłoby się wydawać, że w roku 2014 jest spadek, ale zapisy w bazie danych kończą się w połowie roku).

Liczba klientów jest dość spora, zwłaszcza, jak na liczbę sprzedawców. Korelacja między liczbą klientów a liczbą sprzedawców w danym regionie wydaje się istnieć (więcej klientów -> więcej sprzedawców). Wyjątkiem jest Kanada - w innych regionach jest tylko 1 sprzedawca dla tylu klientów. Rozkład klientów na pewno nie jest równomiernie podzielony na regiony - region Southeast i Northeast wydają się radzić sobie z nimi najgorzej (być może z dalszych badań udałoby się odkryć przyczynę).

Liczba transakcji zdecydowanie wzrasta z każdym rokiem. Zarobki wzrastają jednak wolniej, co oznacza, że średnia wartość transakcji spadła w późniejszych latach.

Aż połowa produktów nigdy nie została sprzedana. Być może oferta jest zbyt różnorodna, być może zostały wprowadzone bardzo niedawno - bez dalszych badań nie wiadomo.

Sklep nie zawsze udziela rabatów. Maksymalne wartości rabatów wydają się umiarkowane. Przydałoby się poznać średnią wartość rabatu, bo maksimum i minimum daje nam za mało informacji.

Próg średniej ceny przekracza ponad  $1/4$  produktów. Wydaje się to być standardową sytuacją.

Zauważalny jest wzrost sprzedaży w miesiącach letnich, co było do przewidzenia dla sklepu rowerowego.

Średni czas oczekiwania klienta w zależności od regionu jest podejrzany. W aż 5 z 6 regionów średni czas wynosi dokładnie 7 dni.

Podsumowując, udało się odkryć część informacji dotyczących sklepu. Bez dalszych badań można jedynie przypuszczać, co mogło być powodem takich, a nie innych, wyników.