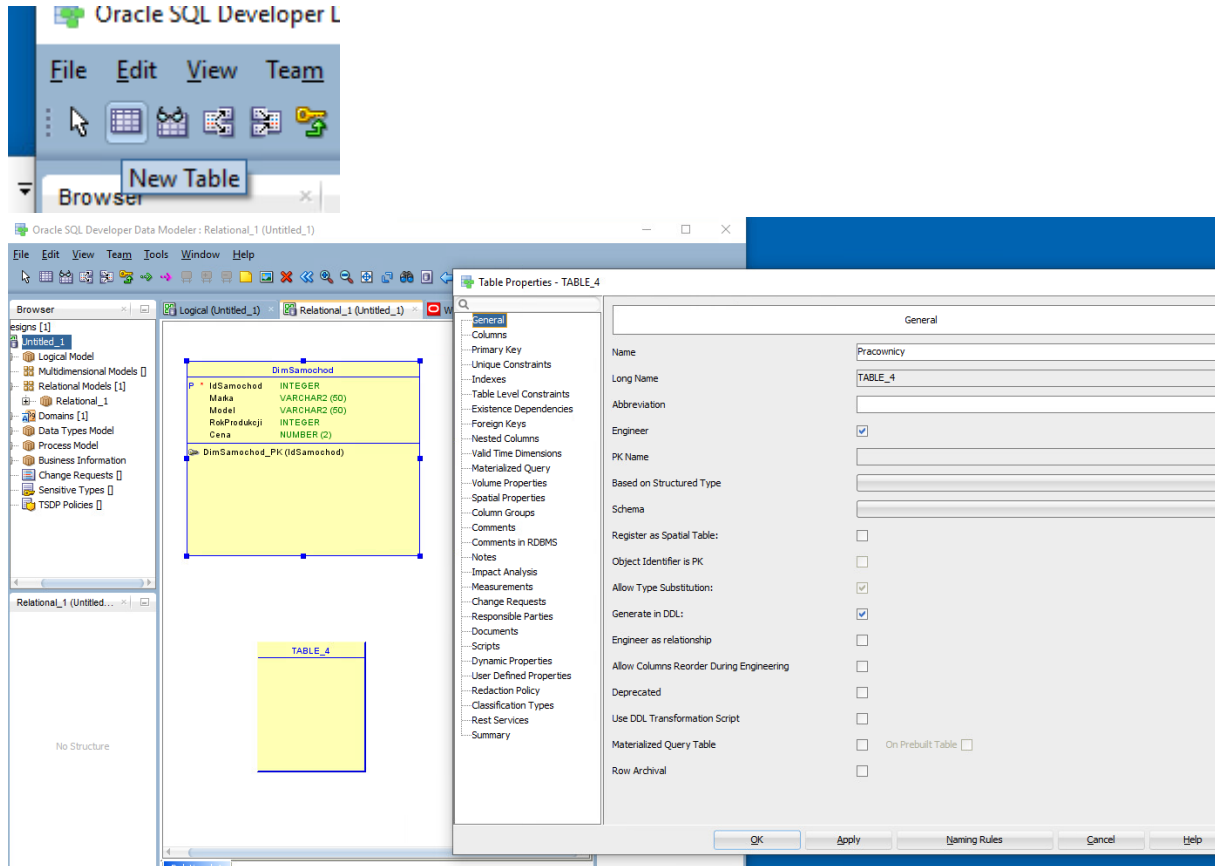
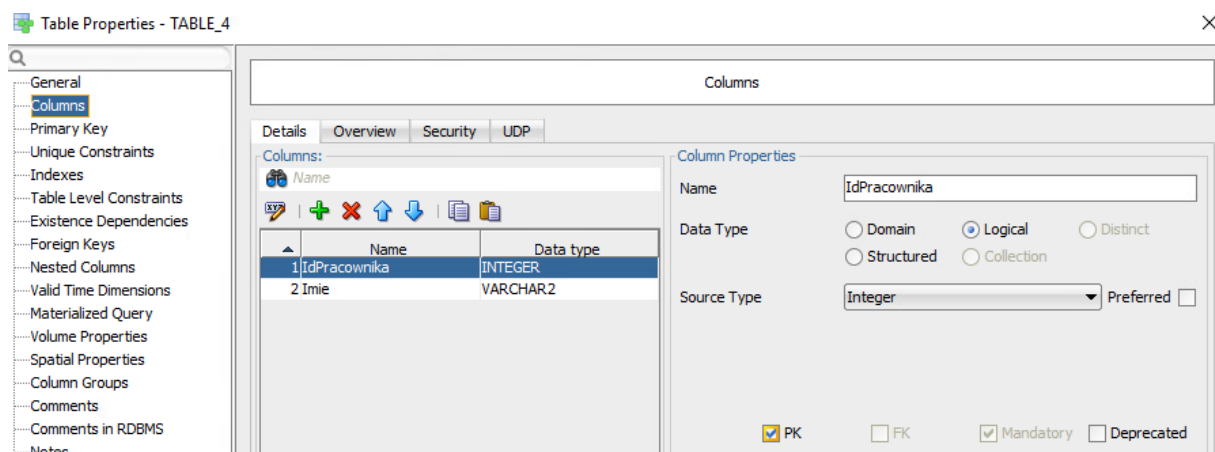


Drugi projekt z baz danych

1. W aplikacji DataModeler dodajemy nowa tabelę, dodajemy jej nazwę.



2. Dodajemy w zakładce Columns nowe kolumny, określamy typ danych oraz w przypadku Primary Key zaznaczamy okienko PK.



3. Dla tabeli wymiarów dodajemy przedrostek Dim.

Table Properties - TABLE_4

General

Name: DimPracownicy

Long Name: DimPracownicy

Abbreviation:

Engineer: ☒

PK Name:

Based on Structured Type:

Schema:

4. W przypadku tabeli faktów dodajemy przedrostek Fact oraz dodajemy kolumny odpowiednie dla PK z innych tabel.

Table Properties - TABLE_7

Columns

Details Overview Security UDP

Columns:

	Name	Data type
1	IdZamowienia	INTEGER
2	IdCzasu	INTEGER
3	IdSamochodu	INTEGER
4	IdKlienta	INTEGER
5	IdPracownika	INTEGER
6	Column_6	UNKNOWN

Column Properties

Name: IdSalonu

Data Type: ☐ Domain ☒ Logical ☐ Distinct

☐ Structured ☐ Collection

Source Type: Integer Preferred ☐

☐ PK ☐ FK ☐ Mandatory ☐ Deprecated

Comments in RDBMS Comments Notes

5. Następnie dodajemy Foreign Keys łącząc odpowiednie kolumny ze sobą z odpowiednich tabel wymiarów.

Table Properties - FactZamowienie

Table Properties - FactZamowienie

Foreign Keys

Name	Referenced Table	Delete Rule	Generate
1 FactZamowienie_DimCzas_FK	DimCzas	NO ACTION	<input checked="" type="checkbox"/>
2 FactZamowienie__FK		NO ACTION	<input checked="" type="checkbox"/>

Referenced Table: DimCzas

Referenced Constraint: DimCzas_PK

☒ Mandatory ☒ Transferable ☐ In Arc

Referenced Column	Column	Mandatory
IdCzasu	IdCzasu	<input type="checkbox"/>
	IdZamowienia	
	IdCzasu	
	IdSamochodu	
	IdKlienta	
	IdPracownika	
	IdSalonu	

Comments Notes

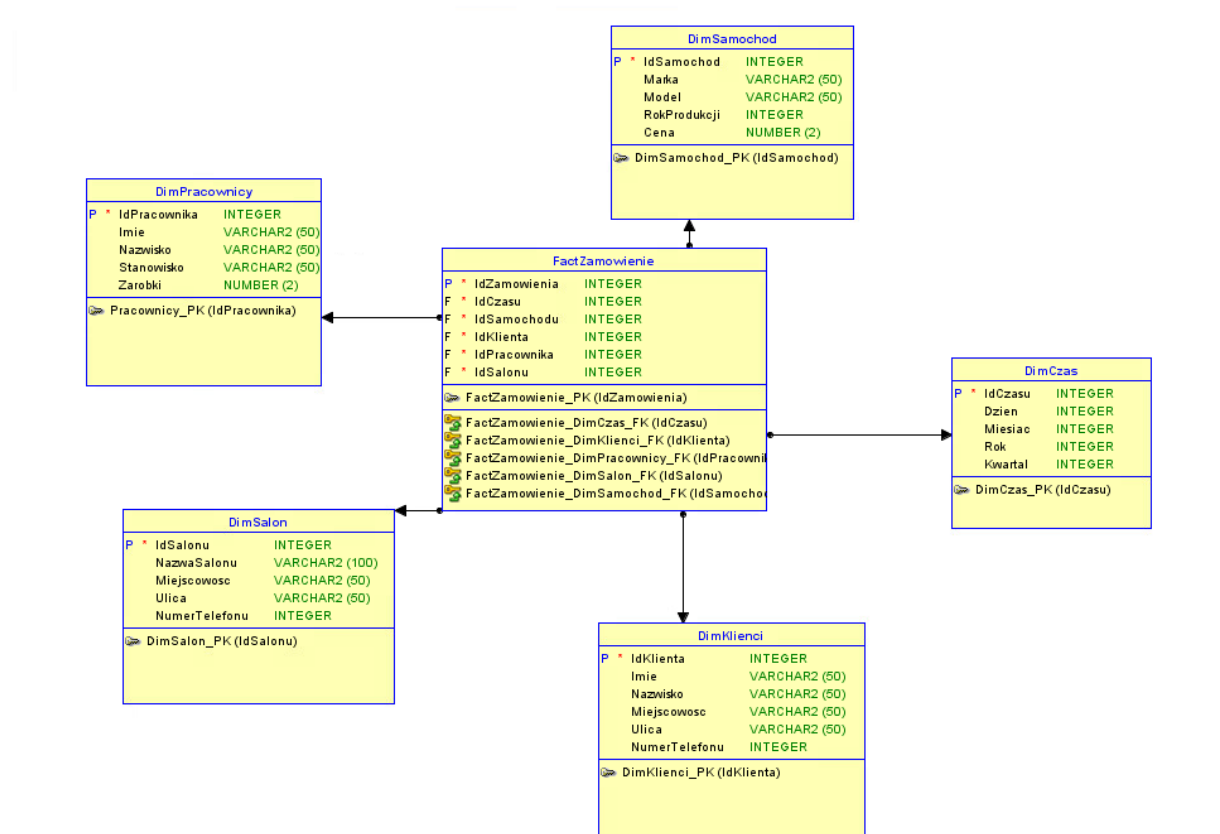
6. Zaznaczamy dla bezpieczeństwa danych Delete Rule: Cascade.

Table Properties - FactZamowienie

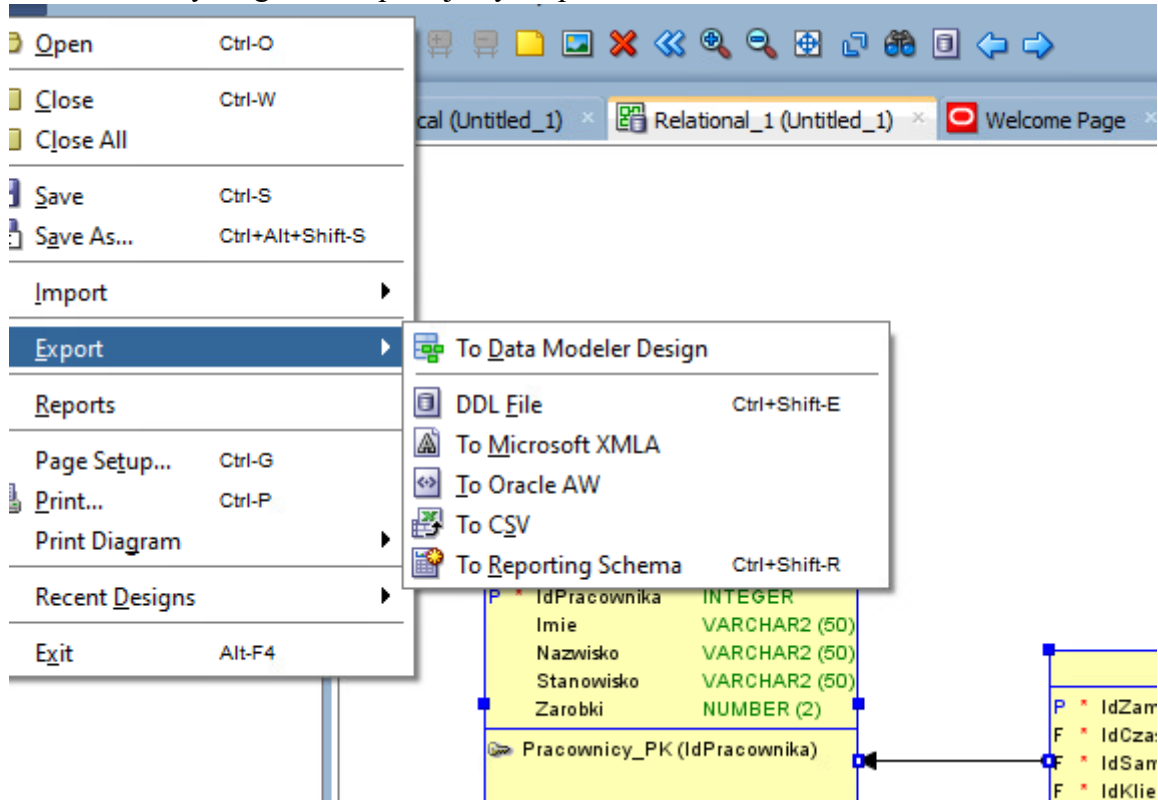
Foreign Keys

Name	Referenced Table	Delete Rule	Generate
1 FactZamowienie_DimCzas_FK	DimCzas	CASCADE	<input checked="" type="checkbox"/>
2 FactZamowienie_DimKlienci_FK	DimKlienci	CASCADE	<input checked="" type="checkbox"/>
3 FactZamowienie_DimPracownicy_FK	DimPracownicy	CASCADE	<input checked="" type="checkbox"/>
4 FactZamowienie_DimSalon_FK	DimSalon	CASCADE	<input checked="" type="checkbox"/>
5 FactZamowienie_DimSamochod_FK	DimSamochod	CASCADE	<input checked="" type="checkbox"/>

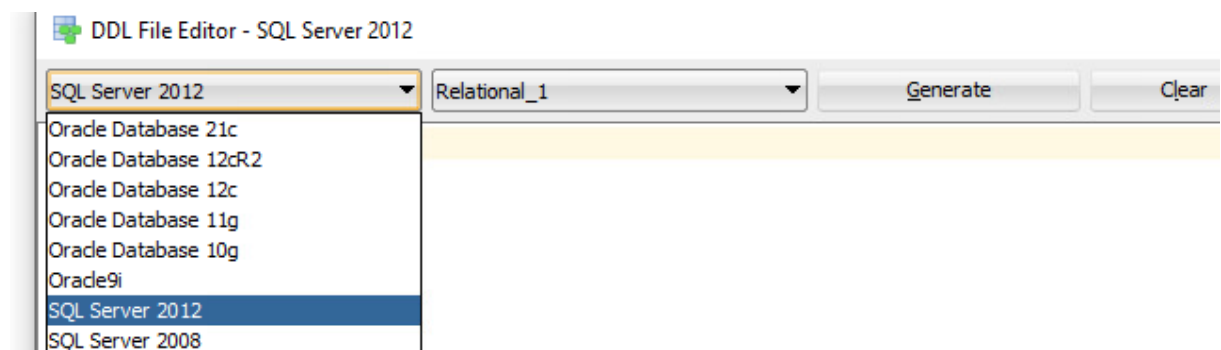
7. Mamy tak stworzy diagram gwiazdy, zaznaczamy automatyczny layout, aby dobrze zwizualizować zależności.



8. Gotowy diagram, eksportujemy do pliku ddl.



9. Wybieramy odpowiedni język, czyli Server 2012



10. Generujemy nasz kod SQL.

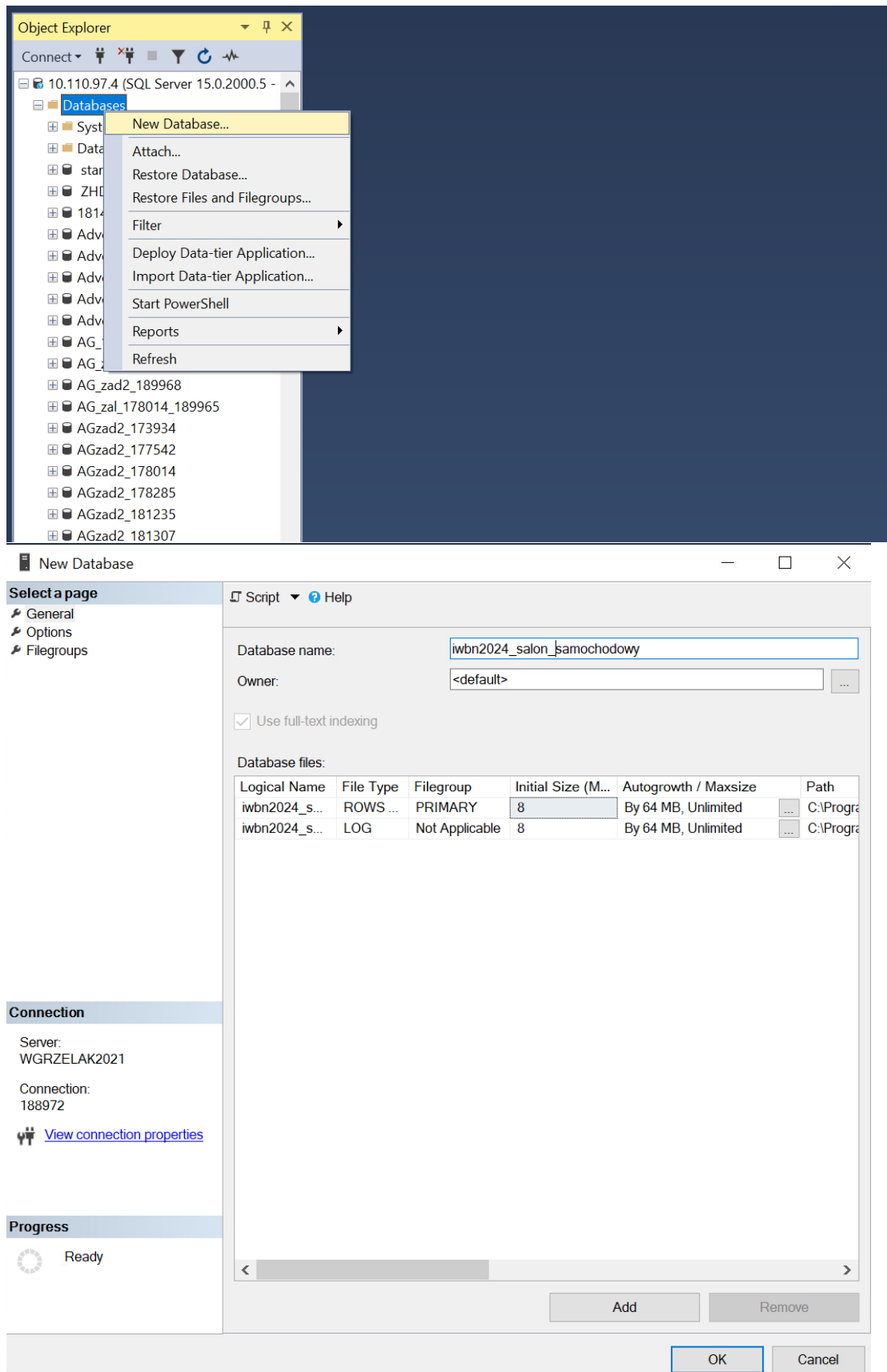
DDL File Editor - SQL Server 2012

SQL Server 2012	Relational_1	Generate	Clear
-----------------	--------------	----------	-------

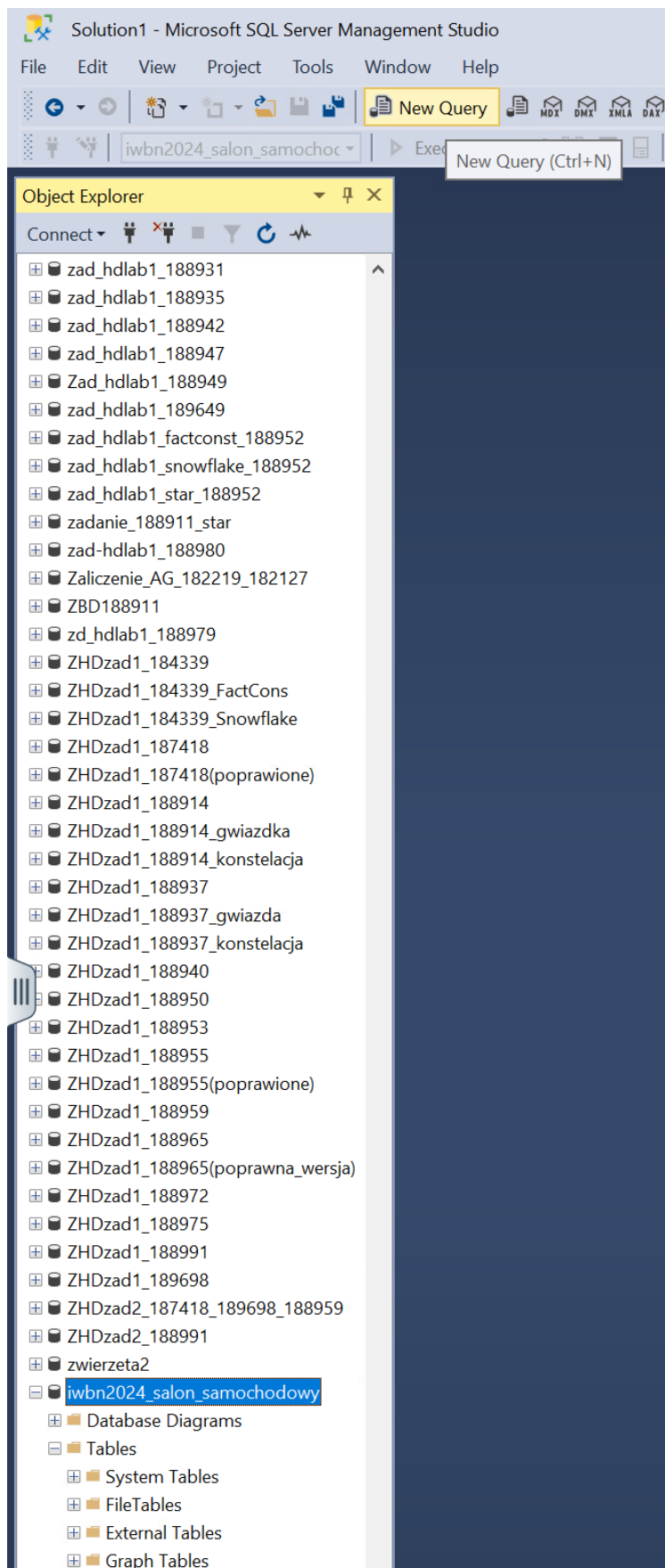
```
1  -- Generated by Oracle SQL Developer Data Modeler 21.2.0.183.1957
2  --   at:      2024-01-29 21:12:22 CET
3  --   site:    SQL Server 2012
4  --   type:    SQL Server 2012
5
6
7
8  CREATE TABLE DimCzas
9  (
10     IdCzasu INTEGER NOT NULL ,
11     Dzień INTEGER ,
12     Miesiac INTEGER ,
13     Rok INTEGER ,
14     Kwartal INTEGER
15 )
16 GO
17
18 ALTER TABLE DimCzas ADD CONSTRAINT DimCzas_PK PRIMARY KEY CLUSTERED (IdCzasu)
19     WITH (
20         ALLOW_PAGE_LOCKS = ON ,
21         ALLOW_ROW_LOCKS = ON )
22 GO
23
24 CREATE TABLE DimKlienci
25 (
26     IdKlienta INTEGER NOT NULL ,
27     Imie VARCHAR (50) ,
28     Nazwisko VARCHAR (50) ,
29     Miejscowosc VARCHAR (50) ,
```

Save	Find	Close
------	------	-------

11. Tworzymy i nazywamy nowa hurtownię danych na serwerze w SQL Server Management Studio



12. W naszej nowej hurtowni danych tworzymy tabele



SQLQuery3.sql - 10.110.97.4.iwbn2024_salon_samochodowy (188972 (54))* - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

Execute

Object Explorer

Connect

- zad_hdlab1_188931
- zad_hdlab1_188935
- zad_hdlab1_188942
- zad_hdlab1_188947
- Zad_hdlab1_188949
- zad_hdlab1_189649
- zad_hdlab1_factconst_188952
- zad_hdlab1_snowflake_188952
- zad_hdlab1_star_188952
- zadanie_188911_star
- zad_hdlab1_188980
- Zaliczenie_AG_182219_182127
- ZBD188911
- zd_hdlab1_188979
- ZHDzad1_184339
- ZHDzad1_184339_FactCons
- ZHDzad1_184339_Snowflake
- ZHDzad1_187418
- ZHDzad1_187418(poprawione)
- ZHDzad1_188914
- ZHDzad1_188914_gwiazdka
- ZHDzad1_188914_konstelacja
- ZHDzad1_188937
- ZHDzad1_188937_gwiazda
- ZHDzad1_188937_konstelacja
- ZHDzad1_188940
- ZHDzad1_188950
- ZHDzad1_188953
- ZHDzad1_188955

SQL Query Editor

```

UNCONSTRAIN FactZamowienie_DimPracownicy_FK FOREIGN KEY
(
    IdPracownika
)
REFERENCES DimPracownicy
(
    IdPracownika
)
ON DELETE CASCADE
ON UPDATE NO ACTION
GO

ALTER TABLE FactZamowienie
ADD CONSTRAINT FactZamowienie_DimSalon_FK FOREIGN KEY
(
    IdSalonu
)
REFERENCES DimSalon
(
    IdSalonu
)
ON DELETE CASCADE
ON UPDATE NO ACTION
GO

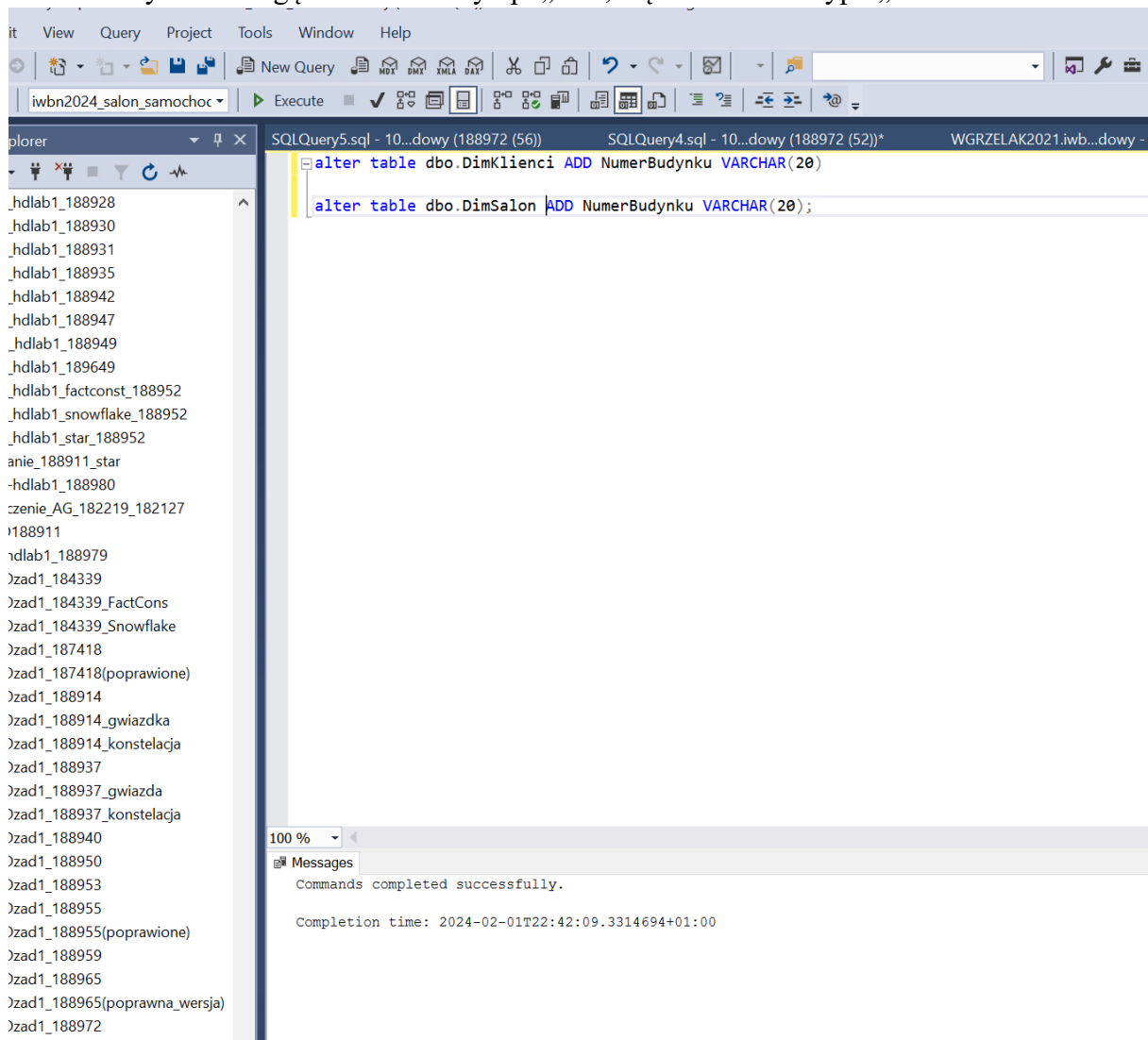
ALTER TABLE FactZamowienie
ADD CONSTRAINT FactZamowienie_DimSamochod_FK FOREIGN KEY
(
    IdSamochodu
)
REFERENCES DimSamochod
(
    IdSamochod
)

```

iwbn2024_salon_samochodowy

- Database Diagrams
- Tables
 - System Tables
 - FileTables
 - External Tables
 - Graph Tables
 - dbo.DimCzas
 - dbo.DimKlienci
 - dbo.DimPracownicy
 - dbo.DimSalon
 - dbo.DimSamochod
 - dbo.FactZamowienie

13. Przez pomyłkę w naszym kodzie zapomnieliśmy o kolumnie „Numer budynku”. Dodajemy ją do tabel Klienci i Salon. Robimy to ze względu atomizacji danych, ponieważ zapomnieliśmy, że kolumna „Ulica” zawiera jej nazwę oraz numer, także musieliśmy ją rozbić na dwie kolumny, stąd dodanie dodatkowej kolumny „Numer budynku”. Typ danych ustawiliśmy „VARCHAR”, ponieważ niektóre numery budynków mogą zawierać litery np. „1A”, stąd odrzucenie typu „INTIGER”.



14. Uzupełniamy nasze tabele danymi

The screenshot displays the SQL Server Enterprise Manager interface. The left pane shows the 'Object Explorer' with a tree view of the database 'webn2024_salon_samochodowy'. The right pane shows a 'SQL Query' window with the following SQL script:

```
INSERT INTO DimKlienci (IdKlienta, Imie, Nazwisko, Miejscowosc, Ulica, NumerBudynku, NumerTelefonu) VALUES
(1, 'Aleksandra', 'Nowak', 'Kraków', 'Długa', '5', '500600700'),
(2, 'Bartosz', 'Kowalski', 'Wrocław', 'Krótka', '56A', '600700000'),
(3, 'Cezary', 'Wiśniewski', 'Poznań', 'Wielka', '165', '700000900'),
(4, 'Dorota', 'Zielińska', 'Gdańsk', 'Wąska', '4B', '800900100'),
(5, 'Ewa', 'Majewska', 'Warszawa', 'Zakręt', '66', '900100200');

INSERT INTO DimPracownicy (IdPracownika, Imie, Nazwisko, Stanowisko, Zarobki) VALUES
(1, 'Marek', 'Jankowski', 'Menadżer', 8000.00),
(2, 'Iwona', 'Szweczyk', 'Księgowa', 5000.00),
(3, 'Łukasz', 'Ostrowski', 'Handlowiec', 6000.00),
(4, 'Katarzyna', 'Kaczmarek', 'Specjalista IT', 7000.00),
(5, 'Piotr', 'Zawadzki', 'Dyrektor', 10000.00);

INSERT INTO DimSalon (IdSalonu, NazwaSalonu, Miejscowosc, Ulica, NumerBudynku, NumerTelefonu) VALUES
(1, 'Autolux', 'Kraków', 'Główna', '333', '111222333'),
(2, 'MotoCentrum', 'Warszawa', 'Ogrodowa', '123', '222333444'),
(3, 'SamochodyPro', 'Wrocław', 'Polna', '14C', '333444555'),
(4, '4Koła', 'Poznań', 'Leśna', '142', '444555666'),
(5, 'AutoLoc', 'Gdańsk', 'Brzegowa', '22', '555666777');

INSERT INTO DimSamochod (IdSamochod, Marka, Model, RokProdukcji, Cena) VALUES
(1, 'Skoda', 'Octavia', 2018, 75000.00),
(2, 'Volkswagen', 'Golf', 2019, 85000.00),
(3, 'Toyota', 'Corolla', 2017, 70000.00),
(4, 'Fiat', 'Punto', 2015, 40000.00),
(5, 'Opel', 'Astra', 2020, 90000.00);

INSERT INTO FactZamowienie (IdZamowienia, IdCzasu, IdSamochodu, IdKlienta, IdPracownika, IdSalonu) VALUES
(1, 4, 2, 3, 4, 5),
(2, 3, 1, 2, 3, 4);
```

The 'Messages' pane at the bottom shows the execution results:

- (5 rows affected)
- (5 rows affected)
- (5 rows affected)
- (5 rows affected)
- (5 rows affected)
- (5 rows affected)

Completion time: 2024-02-01T22:54:08.262527+01:00

Wypełnianie tabeli DimCzas:

- Kod `INSERT INTO DimCzas` dodaje pięć rekordów do tabeli DimCzas.
- Każdy rekord zawiera informacje o różnych okresach czasu, takie jak dzień, miesiąc, rok i kwartał.

INSERT INTO DimCzas (IdCzasu, Dzień, Miesiąc, Rok, Kwartał) VALUES

(1, 1, 10, 2021, 4),

(2, 10, 6, 2022, 2),

(3, 6, 5, 2023, 2),

(4, 21, 12, 2022, 4),

(5, 17, 1, 2024, 1);

Wypełnianie tabeli DimKlienci:

- Kod `INSERT INTO DimKlienci` dodaje pięć rekordów do tabeli DimKlienci.
- Każdy rekord zawiera informacje o różnych klientach, takie jak imię, nazwisko, miejscowość, ulica, numer budynku i numer telefonu.

INSERT INTO DimKlienci (IdKlienta, Imie, Nazwisko, Miejscowosc, Ulica, NumerBudynku, NumerTelefonu) VALUES

(1, 'Aleksandra', 'Nowak', 'Kraków', 'Długa', '5', '500600700'),
(2, 'Bartosz', 'Kowalski', 'Wrocław', 'Krótka', '56A', '600700800'),
(3, 'Cezary', 'Wiśniewski', 'Poznań', 'Wielka', '165', '700800900'),
(4, 'Dorota', 'Zielińska', 'Gdańsk', 'Wąska', '4B', '800900100'),
(5, 'Ewa', 'Majewska', 'Warszawa', 'Zakręt', '66', '900100200');

Wypełnianie tabeli DimPracownicy:

- Kod `INSERT INTO DimPracownicy` dodaje pięć rekordów do tabeli DimPracownicy.
- Każdy rekord zawiera informacje o różnych pracownikach, takie jak imię, nazwisko, stanowisko i zarobki.

INSERT INTO DimPracownicy (IdPracownika, Imie, Nazwisko, Stanowisko, Zarobki) VALUES

(1, 'Marek', 'Jankowski', 'Menadżer', 8000.00),
(2, 'Iwona', 'Szewczyk', 'Księgowa', 5000.00),
(3, 'Łukasz', 'Ostrowski', 'Handlowiec', 6000.00),
(4, 'Katarzyna', 'Kaczmarek', 'Specjalista IT', 7000.00),
(5, 'Piotr', 'Zawadzki', 'Dyrektor', 10000.00);

Wypełnianie tabeli DimSalon:

- Kod `INSERT INTO DimSalon` dodaje pięć rekordów do tabeli DimSalon.
- Każdy rekord zawiera informacje o różnych salonach, takie jak nazwa salonu, miejscowość, ulica, numer budynku i numer telefonu.

INSERT INTO DimSalon (IdSalonu, NazwaSalonu, Miejscowosc, Ulica, NumerBudynku, NumerTelefonu) VALUES

(1, 'AutoLux', 'Kraków', 'Główna', '33J', '111222333'),
(2, 'MotoCentrum', 'Warszawa', 'Ogrodowa', '123', '222333444'),
(3, 'SamochodyPro', 'Wrocław', 'Pólna', '14C', '333444555'),
(4, '4Koła', 'Poznań', 'Leśna', '142', '444555666'),
(5, 'AutoMoc', 'Gdańsk', 'Brzegowa', '22', '555666777');

Wypełnianie tabeli DimSamochod:

- Kod `INSERT INTO DimSamochod` dodaje pięć rekordów do tabeli DimSamochod.
- Każdy rekord zawiera informacje o różnych samochodach, takie jak marka, model, rok produkcji i cena.

```
INSERT INTO DimSamochod (IdSamochod, Marka, Model, RokProdukcji, Cena) VALUES
```

```
(1, 'Skoda', 'Octavia', 2018, 75000.00),
```

```
(2, 'Volkswagen', 'Golf', 2019, 85000.00),
```

```
(3, 'Toyota', 'Corolla', 2017, 70000.00),
```

```
(4, 'Fiat', 'Punto', 2015, 40000.00),
```

```
(5, 'Opel', 'Astra', 2020, 90000.00);
```

Wypełnianie tabeli FactZamowienie:

- Kod `INSERT INTO FactZamowienie` dodaje pięć rekordów do tabeli FactZamowienie.
- Każdy rekord zawiera informacje o zamówieniach, takie jak id zamówienia, id czasu, id samochodu, id klienta, id pracownika i id salonu.

```
INSERT INTO FactZamowienie (IdZamowienia, IdCzasu, IdSamochodu, IdKlienta, IdPracownika, IdSalonu) VALUES
```

```
(1, 4, 2, 3, 4, 5),
```

```
(2, 3, 1, 2, 3, 4),
```

```
(3, 2, 3, 5, 2, 1),
```

```
(4, 1, 5, 4, 5, 3),
```

```
(5, 5, 4, 1, 1, 2);
```

15. Opis kodu tworzącego tabele.

Tabela DimCzas:

Kolumny:

- 'IdCzasu' (INTEGER) - identyfikator czasu, klucz główny.
- 'Dzien' (INTEGER) - dzień w miesiącu.
- 'Miesiac' (INTEGER) - numer miesiąca.
- 'Rok' (INTEGER) - rok.
- 'Kwartal' (INTEGER) - numer kwartału.

Tabela DimKlienci:

Kolumny:

- 'IdKlienta' (INTEGER) - identyfikator klienta, klucz główny.
- 'Imie' (VARCHAR(50)) - imię klienta.
- 'Nazwisko' (VARCHAR(50)) - nazwisko klienta.
- 'Miejscowosc' (VARCHAR(50)) - miejscowość klienta.
- 'Ulica' (VARCHAR(50)) - ulica klienta.
- 'NumerBudynku' (VARCHAR(20)) - numer budynku klienta.
- 'NumerTelefonu' (INTEGER) - numer telefonu klienta.

Tabela DimPracownicy:

Kolumny:

- 'IdPracownika' (INTEGER) - identyfikator pracownika, klucz główny.
- 'Imie' (VARCHAR(50)) - imię pracownika.
- 'Nazwisko' (VARCHAR(50)) - nazwisko pracownika.
- 'Stanowisko' (VARCHAR(50)) - stanowisko pracownika.
- 'Zarobki' (DECIMAL(10,2)) - zarobki pracownika.

Tabela DimSalon:

Kolumny:

- 'IdSalonu' (INTEGER) - identyfikator salonu, klucz główny.
- 'NazwaSalonu' (VARCHAR(100)) - nazwa salonu.
- 'Miejscowosc' (VARCHAR(50)) - miejscowość salonu.
- 'Ulica' (VARCHAR(50)) - ulica salonu.
- 'NumerBudynku' (VARCHAR(20)) - numer budynku salonu.
- 'NumerTelefonu' (INTEGER) - numer telefonu salonu.

Tabela DimSamochod:

Kolumny:

- 'IdSamochod' (INTEGER) - identyfikator samochodu, klucz główny.
- 'Marka' (VARCHAR(50)) - marka samochodu.
- 'Model' (VARCHAR(50)) - model samochodu.
- 'RokProdukcji' (INTEGER) - rok produkcji samochodu.
- 'Cena' (DECIMAL(10,2)) - cena samochodu.

Tabela FactZamowienie:

Kolumny:

- 'IdZamowienia' (INTEGER) - identyfikator zamówienia, klucz główny.
- 'IdCzasu' (INTEGER) - identyfikator czasu, klucz obcy powiązany z tabelą DimCzas.
- 'IdSamochodu' (INTEGER) - identyfikator samochodu, klucz obcy powiązany z tabelą DimSamochod.
- 'IdKlienta' (INTEGER) - identyfikator klienta, klucz obcy powiązany z tabelą DimKlienci.
- 'IdPracownika' (INTEGER) - identyfikator pracownika, klucz obcy powiązany z tabelą DimPracownicy.
- 'IdSalonu' (INTEGER) - identyfikator salonu, klucz obcy powiązany z tabelą DimSalon.

Opis typów danych użytych przy tworzeniu tabel:

1. INTEGER:

- Liczba całkowita, używana dla identyfikatorów, numerów oraz roków.

2. VARCHAR(n):

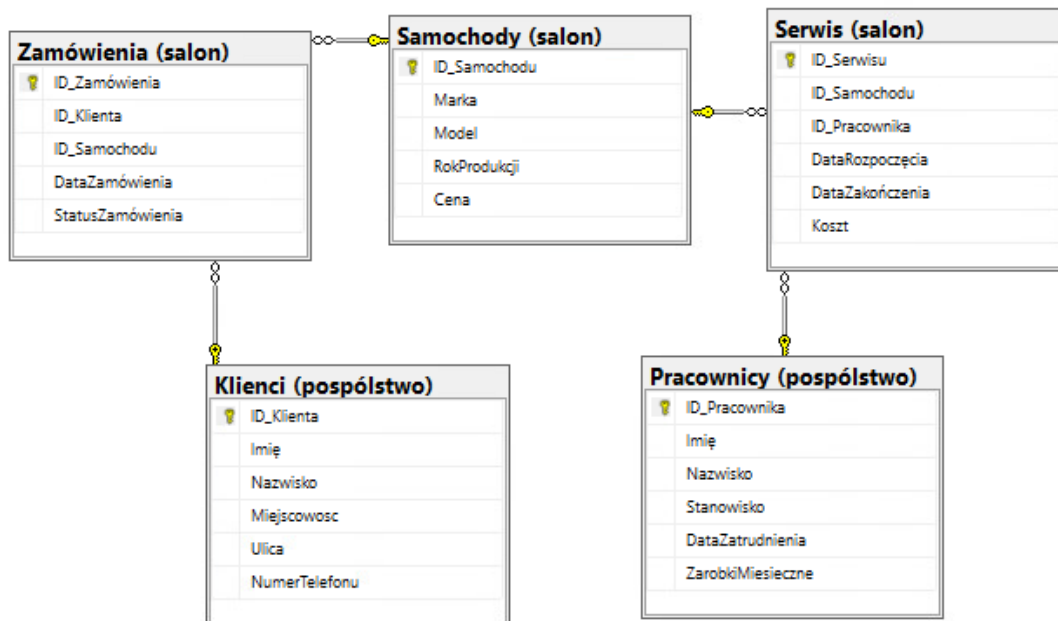
- Ciąg znaków o zmiennej długości, gdzie "n" określa maksymalną długość tekstu.

3. DECIMAL(p, s):

- Liczba zmiennoprzecinkowa o precyzji "p" miejsc, z czego "s" miejsc po przecinku.

16. Schematy z cz1 i cz2:

Schemat z części pierwszej projektu zaliczeniowego:

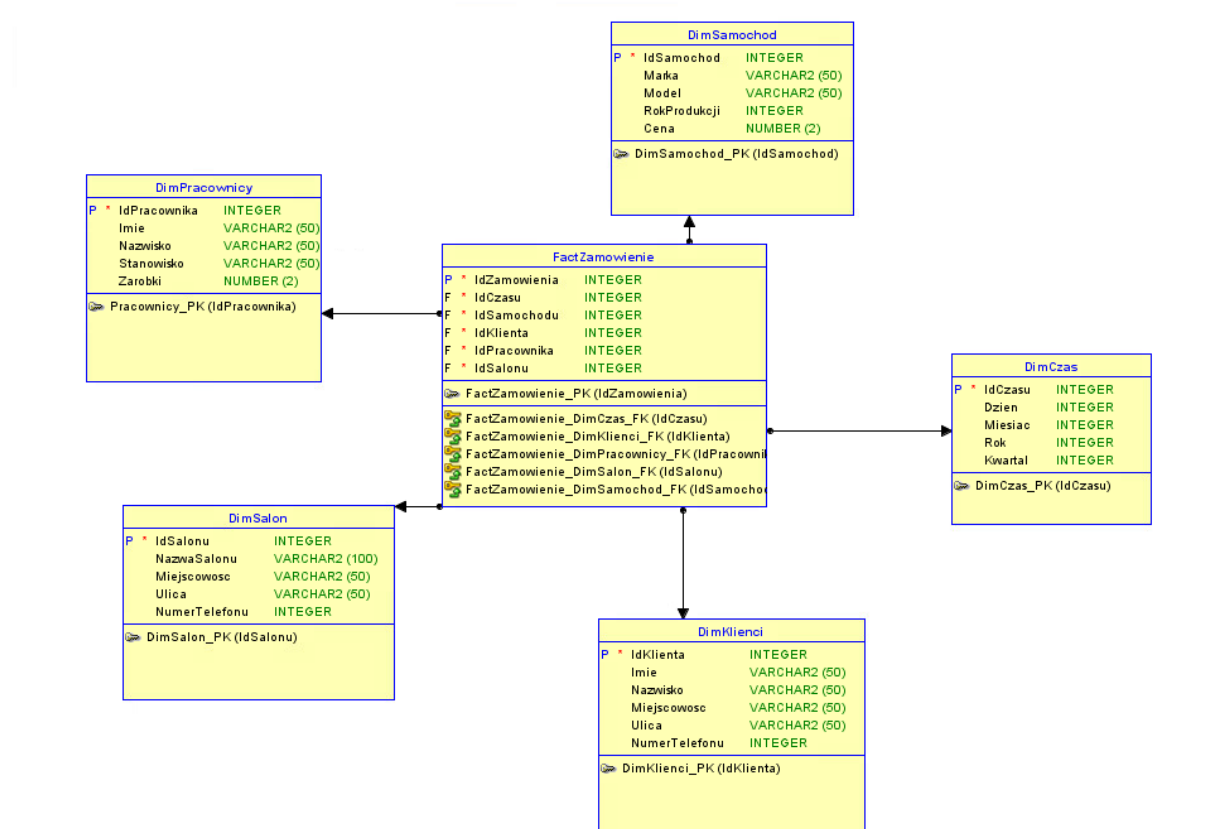


Baza danych reprezentuje wirtualny salon samochodowy zorganizowany w kilka tabel. Pierwsza tabela, "Samochody", przechowuje informacje o samochodach, takie jak marka, model, rok produkcji oraz cena. Kolejna tabela, "Klienci", zawiera dane klientów, obejmujące imię, nazwisko, adres, oraz numer telefonu. Tabela "Zamówienia" utrzymuje informacje o złożonych zamówieniach, w tym klientach, zamówionych samochodach, dacie zamówienia i statusie realizacji.

Dodatkowo, tabela "Pracownicy" skupia się na zatrudnionych osobach, prezentując ich imię, nazwisko, stanowisko, datę zatrudnienia oraz zarobki miesięczne. Natomiast tabela "Serwis" obejmuje informacje dotyczące przeprowadzonych serwisów, takie jak samochód poddany serwisowi, pracownik odpowiedzialny za serwis, daty rozpoczęcia i zakończenia, oraz koszt serwisu.

W relacjach między tabelami, zamówienia są powiązane z klientami i samochodami poprzez klucze obce, a tabela serwisów jest powiązana zarówno z samochodami, jak i pracownikami. To kompleksowe ułożenie tabel umożliwia śledzenie klientów, zamówień, serwisów oraz pracowników w ramach jednego spójnego systemu.

Schemat z drugiej części projektu zaliczeniowego:



Hurtownia danych oparta na schemacie gwiazdy obejmuje pięć tabel wymiarów oraz jedną tabelę faktów. Oto opis poszczególnych elementów hurtowni:

1. Tabela "DimCzas":
 - Przechowuje informacje o czasie, takie jak dzień, miesiąc, rok i kwartał.
2. Tabela "DimKlienci":
 - Zawiera dane dotyczące klientów, takie jak imię, nazwisko, miejscowość, ulica, numer telefonu oraz numer budynku.
3. Tabela "DimPracownicy":
 - Utrzymuje informacje o pracownikach, takie jak imię, nazwisko, stanowisko oraz zarobki.
4. Tabela "DimSalon":
 - Przechowuje dane o salonach, w tym nazwę salonu, miejscowość, ulicę, numer telefonu oraz numer budynku.
5. Tabela "DimSamochod":
 - Zawiera informacje o samochodach, takie jak marka, model, rok produkcji oraz cena.

6. Tabela "FactZamowienie":

- Stanowi centralną tabelę faktów, związującą klucze obce z tabel wymiarów. Przechowuje informacje o zamówieniach, w tym klucze obce do tabel "DimCzas", "DimSamochod", "DimKlienci", "DimPracownicy" i "DimSalon".

Relacje między tabelami:

- Tabela "FactZamowienie" jest połączona z tabelami wymiarów poprzez klucze obce. Klucze obce wskazują na rekordy w tabelach wymiarów, co umożliwia analizę danych w kontekście czasu, klientów, pracowników, salonów oraz samochodów.

Ta hurtownia danych umożliwia kompleksową analizę zamówień, uwzględniając różne aspekty, takie jak czas, klienci, pracownicy, salony oraz samochody.

17. Porównanie obu projektów.

Kod Tworzący Bazę Danych:

Plusy:

- Prosty i intuicyjny do zrozumienia dla osób początkujących.
- Skoncentrowany na jednostkowej funkcji - przechowywaniu danych o samochodach, klientach, pracownikach i zamówieniach.
- Wysokiej czytelności operacje tworzenia, wstawiania danych oraz definiowania kluczy obcych.

Minusy:

- Brak wyraźnego podziału między danymi operacyjnymi, a danymi analitycznymi.
- Brak optymalizacji dla analizy danych z perspektywy hurtowni danych.
- Brak hierarchii danych wymiarów i faktów, co może utrudnić analizę danych w kontekście biznesowym.

Kod Tworzący Hurtownię Danych:

Plusy:

- Wykorzystuje model gwiazdy, co ułatwia analizę danych z różnych perspektyw biznesowych.
- Jasno określone tabele wymiarów i faktów, co ułatwia zrozumienie struktury hurtowni danych.
- Zastosowanie kluczy obcych i indeksów dla optymalizacji analizy danych.

Minusy:

- Bardziej złożony w porównaniu do kodu bazy danych, co może być trudne dla osób początkujących.
- Wymaga zaawansowanej wiedzy w obszarze hurtowni danych.
- Dodatkowe tabele wymiarów mogą prowadzić do większej ilości złączeń podczas zapytań, co może wpływać na wydajność.

Porównanie Schematów:

Struktura bazy danych:

- Prosta, płaska struktura zorientowana na przechowywanie operacyjnych danych o samochodach, klientach, pracownikach i zamówieniach.

Zastosowanie:

- Idealna do codziennego zarządzania operacjami salonu samochodowego.
- Przydatna dla transakcyjnych operacji, ale mniej efektywna dla analizy danych biznesowych.

Struktura hurtowni danych:

- Hierarchiczna struktura gwiazdy, w której tabele wymiarów reprezentują kontekst biznesowy, a tabela faktów przechowuje kluczowe dane do analizy.

Zastosowanie:

- Idealna do analizy danych biznesowych, generowania raportów i identyfikacji trendów.
- Lepsza efektywność w analizie z perspektywy różnych wymiarów, takich jak czas, klienci, samochody i pracownicy.

Podsumowanie:

Zarówno kod tworzący bazę danych, jak i kod tworzący hurtownię danych mają swoje miejsce w procesie zarządzania danymi. Baza danych jest skierowana na operacyjną stronę biznesu, podczas gdy hurtownia danych oferuje bardziej zaawansowane możliwości analizy i raportowania. Wybór zależy od konkretnych potrzeb przedsiębiorstwa i priorytetów związanych z operacjami lub analizą danych biznesowych.