

Bazy danych

Sprawozdanie z projektu

Mateusz Grabowski

WCY21IY4S1

1. Treść i opis zadania

Projekt przedstawia bazę danych sklepu komputerowego. Przechowuje się w niej między innymi nazwy danych produktów, numery zamówień, dane o klientach sklepu, dane o pracownikach magazyn, zestawienie wszystkich magazynów z danych oddziałów firmy sklepu, zbiór ocen konsumenta. W tym celu utworzono 10 obiektów opisywanych co najmniej 5 atrybutami. Zdefiniowane obiekty to:

1. Produkt
2. Magazyn
3. Klient
4. Zamowienie
5. Feedback
6. Pracownik
7. Producent
8. Kupon
9. Oddzial
10. Usługi

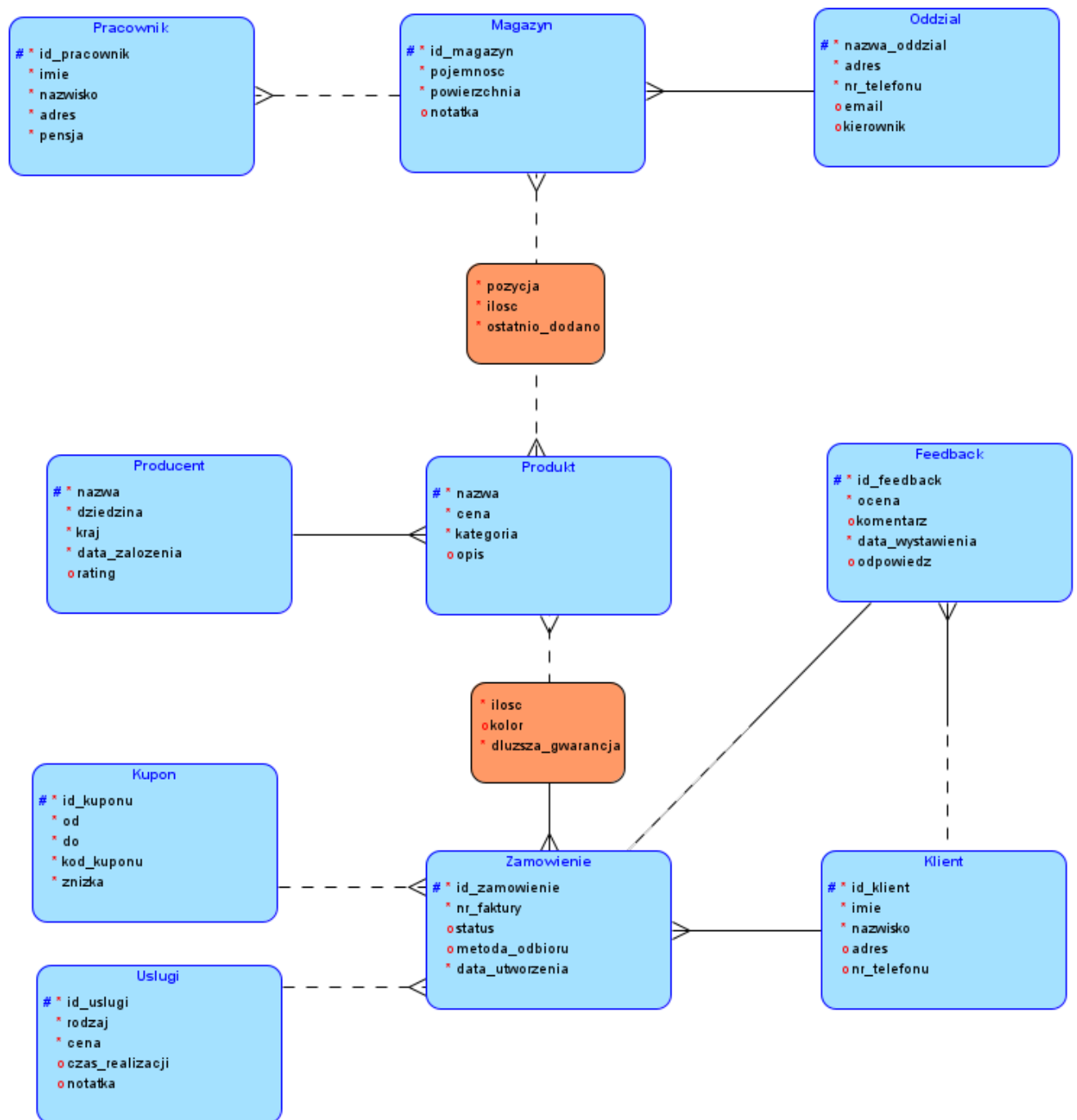
Przy tworzeniu tablic skorzystano z sekwencji.

Producent pełni funkcję słownika dla wszystkich producentów, którzy są jednym z atrybutów obiektu Produkt. Kupon pełni funkcję słownika dla wszystkich kuponów, które są również jednym z atrybutów obiektu Produkt. Pojedyncze zamówienie może wykorzystać 1 kupon. Oddział pełni funkcję słownika dla wszystkich nazw oddziałów; magazynów sklepu komputerowego.

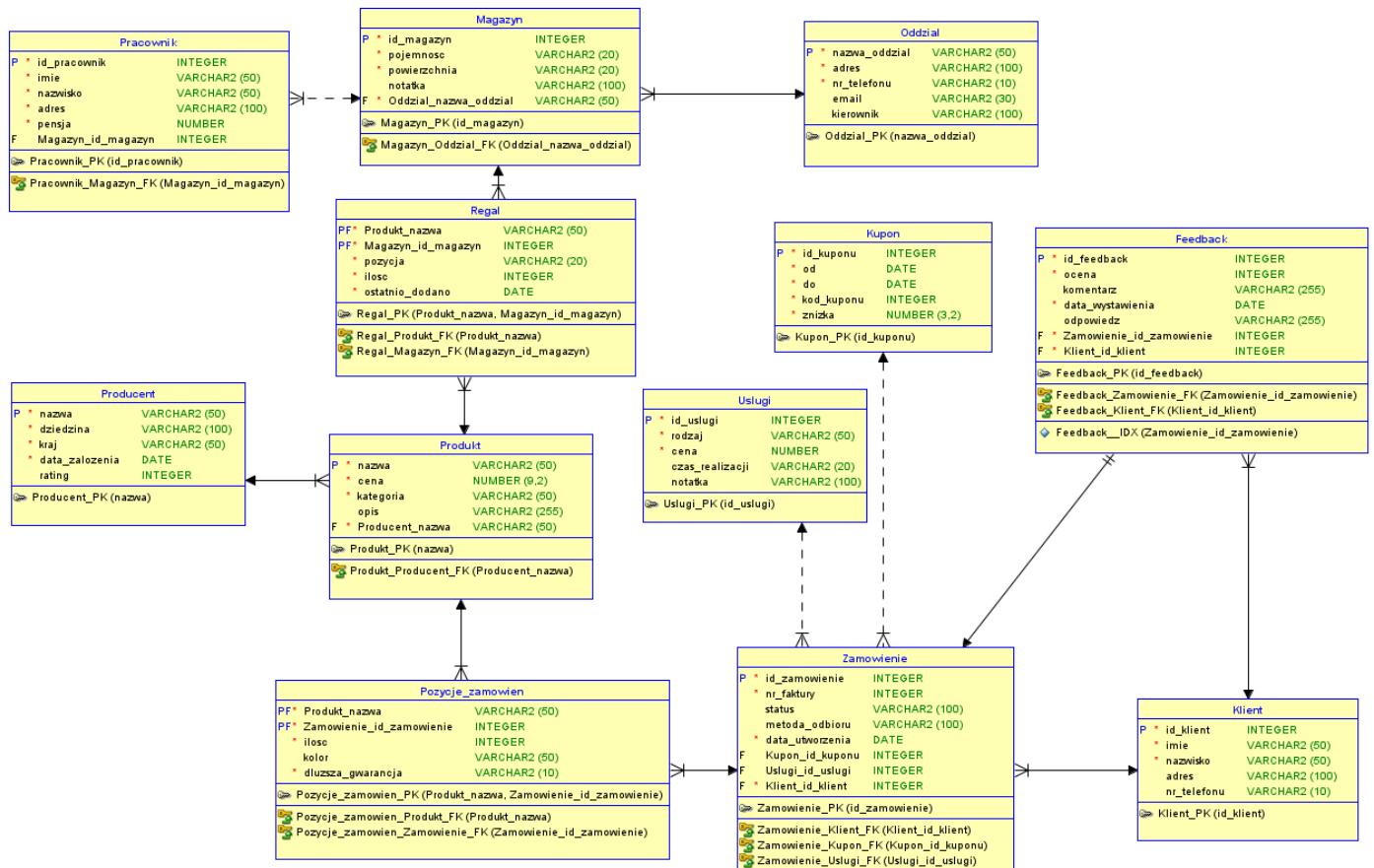
Założono, że produkty rozróżnia się poprzez ich nazwę, nazwa produktu w bazie danych Produkt występuje tylko raz. Obiekt Produkt oraz Magazyn są ze sobą w relacji wiele-do-wielu. Produkt o danej nazwie może znajdować się w wielu magazynach, ale również w magazynie może znajdować się wiele produktów o różnych nazwach. W relacji zastosowano intersekcję pełniącą rolę regału w konkretnym magazynie o nazwie Regal z dodatkowymi atrybutami: pozycja(np. Regal 2A), ilosc(danego produktu),ostatnie_zaksiegowanie. Obiekt Produkt oraz Zamowienie są ze sobą w relacji wiele-do-wielu. Zamówienie może zawierać wiele produktów, ale również produkt o danej nazwie może znajdować się w wielu zamówieniach. W relacji zastosowano intersekcję pełniącą rolę pozycji dla poszczególnych produktów w konkretnych zamówieniach o nazwie Pozycja_zamowienie z dodatkowymi atrybutami: ilosc, kolor oraz przedluzenie_gwarancji. Na każdej pozycji może znaleźć się jeden produkt o konkretnej nazwie i jego określona ilość, kolor i decyzja o przedłużeniu gwarancji.

Zamówienie może, ale nie musi zawierać kupon ze słownika kuponów. W zamówieniu również może, ale nie musi być uwzględniona konkretna usługa. Magazyn może mieć wielu pracowników oraz pracownik może pracować w swoim magazynie. Klient musi mieć od przynajmniej 1 do wielu zamówień, a zamówienie musi dotyczyć swojego klienta. Klient może wносить wiele uwag do zamówienia, a poszczególna uwagi musi pochodzić od jednego klienta. Istniejąca uwaga musi dotyczyć zamówienia, którego dotyczy, natomiast zamówienie może, ale nie musi, mieć uwagę.

2. Logiczny model danych bazy danych



3. Fizyczny model danych bazy danych



Tworzenie tabel:

```

CREATE TABLE feedback (
    id_feedback          INTEGER NOT NULL,
    ocena                INTEGER NOT NULL,
    komentarz            VARCHAR2(255),
    data_wystawienia     DATE NOT NULL,
    odpowiedz            VARCHAR2(255),
    zamowienie_id_zamowienie INTEGER NOT NULL,
    klient_id_klient     INTEGER NOT NULL
);
    
```

```

CREATE UNIQUE INDEX feedback_idx ON
    feedback (
        zamowienie_id_zamowienie
    ASC );
    
```

```

ALTER TABLE feedback ADD CONSTRAINT feedback_pk PRIMARY KEY ( id_feedback );
    
```

```

CREATE TABLE klient (
    id_klient    INTEGER NOT NULL,
    imie         VARCHAR2(50) NOT NULL,
    nazwisko     VARCHAR2(50) NOT NULL,
    adres        VARCHAR2(100),
    nr_telefonu  VARCHAR2(10)
);
    
```

```

ALTER TABLE klient ADD CONSTRAINT klient_pk PRIMARY KEY ( id_klient );
    
```

```

CREATE TABLE kupon (
    id_kuponu    INTEGER NOT NULL,
    od            DATE NOT NULL,
    
```

```

do          DATE NOT NULL,
kod_kuponu  INTEGER NOT NULL,
znizka      NUMBER(3, 2) NOT NULL
);

ALTER TABLE kupon ADD CONSTRAINT kupon_pk PRIMARY KEY ( id_kuponu );

CREATE TABLE magazyn (
    id_magazyn          INTEGER NOT NULL,
    pojemnosc           VARCHAR2(20) NOT NULL,
    powierzchnia        VARCHAR2(20) NOT NULL,
    notatka              VARCHAR2(100),
    oddzial_nazwa_oddzial VARCHAR2(50) NOT NULL
);

ALTER TABLE magazyn ADD CONSTRAINT magazyn_pk PRIMARY KEY ( id_magazyn );

CREATE TABLE oddzial (
    nazwa_oddzial VARCHAR2(50) NOT NULL,
    adres          VARCHAR2(100) NOT NULL,
    nr_telefonu    VARCHAR2(10) NOT NULL,
    email          VARCHAR2(30),
    kierownik      VARCHAR2(100)
);

ALTER TABLE oddzial ADD CONSTRAINT oddzial_pk PRIMARY KEY ( nazwa_oddzial );

CREATE TABLE pozycje_zamowien (
    produkt_nazwa          VARCHAR2(50) NOT NULL,
    zamowienie_id_zamowienie INTEGER NOT NULL,
    ilosc                   INTEGER NOT NULL,
    kolor                   VARCHAR2(50),
    dluzsza_gwarancja       VARCHAR2(10) NOT NULL
);

ALTER TABLE pozycje_zamowien ADD CONSTRAINT pozycje_zamowien_pk PRIMARY KEY (
    produkt_nazwa,
    zamowienie_id_zamowienie );

CREATE TABLE pracownik (
    id_pracownik          INTEGER NOT NULL,
    imie                   VARCHAR2(50) NOT NULL,
    nazwisko               VARCHAR2(50) NOT NULL,
    adres                  VARCHAR2(100) NOT NULL,
    pensja                 NUMBER NOT NULL,
    magazyn_id_magazyn     INTEGER
);

ALTER TABLE pracownik ADD CONSTRAINT pracownik_pk PRIMARY KEY ( id_pracownik );

CREATE TABLE producent (
    nazwa          VARCHAR2(50) NOT NULL,
    dziedzina      VARCHAR2(100) NOT NULL,
    kraj           VARCHAR2(50) NOT NULL,
    data_zalozenia DATE NOT NULL,
    rating         INTEGER
);

ALTER TABLE producent ADD CONSTRAINT producent_pk PRIMARY KEY ( nazwa );

CREATE TABLE produkt (
    nazwa          VARCHAR2(50) NOT NULL,
    cena           NUMBER(9, 2) NOT NULL,
    kategoria      VARCHAR2(50) NOT NULL,
    opis           VARCHAR2(255),
    producent_nazwa VARCHAR2(50) NOT NULL

```

```

);

ALTER TABLE produkt ADD CONSTRAINT produkt_pk PRIMARY KEY ( nazwa );

CREATE TABLE regal (
    produkt_nazwa      VARCHAR2(50) NOT NULL,
    magazyn_id_magazyn  INTEGER NOT NULL,
    pozycja             VARCHAR2(20) NOT NULL,
    ilosc               INTEGER NOT NULL,
    ostatnio_dodano     DATE NOT NULL
);

ALTER TABLE regal ADD CONSTRAINT regal_pk PRIMARY KEY ( produkt_nazwa,
                                                         magazyn_id_magazyn );

CREATE TABLE uslugi (
    id_uslugi           INTEGER NOT NULL,
    rodzaj              VARCHAR2(50) NOT NULL,
    cena                NUMBER NOT NULL,
    czas_realizacji     VARCHAR2(20),
    notatka              VARCHAR2(100)
);

ALTER TABLE uslugi ADD CONSTRAINT uslugi_pk PRIMARY KEY ( id_uslugi );

CREATE TABLE zamowienie (
    id_zamowienie       INTEGER NOT NULL,
    nr_faktury           INTEGER NOT NULL,
    status               VARCHAR2(100),
    metoda_odbioru       VARCHAR2(100),
    data_utworzenia      DATE NOT NULL,
    kupon_id_kuponu      INTEGER,
    uslugi_id_uslugi     INTEGER,
    klient_id_klient     INTEGER NOT NULL
);

ALTER TABLE zamowienie ADD CONSTRAINT zamowienie_pk PRIMARY KEY ( id_zamowienie );

ALTER TABLE feedback
    ADD CONSTRAINT feedback_klient_fk FOREIGN KEY ( klient_id_klient )
        REFERENCES klient ( id_klient );

ALTER TABLE feedback
    ADD CONSTRAINT feedback_zamowienie_fk FOREIGN KEY ( zamowienie_id_zamowienie )
        REFERENCES zamowienie ( id_zamowienie );

ALTER TABLE magazyn
    ADD CONSTRAINT magazyn_oddzial_fk FOREIGN KEY ( oddzial_nazwa_oddzial )
        REFERENCES oddzial ( nazwa_oddzial );

ALTER TABLE pozycje_zamowien
    ADD CONSTRAINT pozycje_zamowien_produk_t_fk FOREIGN KEY ( produkt_nazwa )
        REFERENCES produkt ( nazwa );

ALTER TABLE pozycje_zamowien
    ADD CONSTRAINT pozycje_zamowien_zamowienie_fk FOREIGN KEY (
zamowienie_id_zamowienie )
        REFERENCES zamowienie ( id_zamowienie );

ALTER TABLE pracownik
    ADD CONSTRAINT pracownik_magazyn_fk FOREIGN KEY ( magazyn_id_magazyn )
        REFERENCES magazyn ( id_magazyn );

ALTER TABLE produkt
    ADD CONSTRAINT produkt_producent_fk FOREIGN KEY ( producent_nazwa )
        REFERENCES producent ( nazwa );

```


[illegible]

cache 20;

```
insert into FEEDBACK values (seq_id_feedback.nextval, 10, 'bardzo szybko zrealizowano
zamowienie', '2022/07/12', 'Dziekujemy za pozytywna ocene', 1, 1);
insert into FEEDBACK values (seq_id_feedback.nextval, 6, 'dlugi czas oczekiwania',
'2022/09/24', 'Dziekujemy za pozytywna ocene', 2, 1);
insert into FEEDBACK values (seq_id_feedback.nextval, 10, 'wyrozumiala obsluga',
'2022/09/11', '', 3, 1);
insert into FEEDBACK values (seq_id_feedback.nextval, 8, 'wszystko dziala',
'2022/07/07', 'Dziekujemy za pozytywna ocene', 4, 12);
insert into FEEDBACK values (seq_id_feedback.nextval, 2, 'dlugi czas oczekiwania',
'2022/10/01', 'Przepraszamy za niedogodnosci', 5, 2);
insert into FEEDBACK values (seq_id_feedback.nextval, 4, '', '2022/10/28',
'Przepraszamy za niedogodnosci', 6, 3);
insert into FEEDBACK values (seq_id_feedback.nextval, 9, 'bardzo mila obsluga',
'2022/11/29', '', 7, 4);
insert into FEEDBACK values (seq_id_feedback.nextval, 7, '', '2021/08/20', '', 8, 4);
insert into FEEDBACK values (seq_id_feedback.nextval, 10, 'wyrozumiala obsluga',
'2022/08/06', '', 9, 5);
insert into FEEDBACK values (seq_id_feedback.nextval, 8, '', '2020/07/06', '', 10, 6);
insert into FEEDBACK values (seq_id_feedback.nextval, 10, 'wyrozumiale podejscie do
klienta', '2021/11/12', '', 11, 7);
insert into FEEDBACK values (seq_id_feedback.nextval, 9, 'szybko dotarla przesyłka',
'2022/05/12', '', 12, 8);
```

-- 6. PRODUCENT

```
insert into PRODUCENT values ('Firma A', 'karty graficzne', 'Polska', '2000/04/15', 9);
insert into PRODUCENT values ('Firma B', 'procesory', 'Niemcy', '1997/06/23', 10);
insert into PRODUCENT values ('Firma C', 'monitory', 'Stany Zjednoczone', '1989/01/20',
10);
insert into PRODUCENT values ('Firma D', 'klawiatury', 'Stany Zjednoczone',
'2007/06/18', '');
insert into PRODUCENT values ('Firma E', 'sluchawki', 'Stany Zjednoczone',
'2004/11/23', 9);
insert into PRODUCENT values ('Firma F', 'karty graficzne', 'Stany Zjednoczone',
'1996/03/13', 6);
insert into PRODUCENT values ('Firma G', 'komputery', 'Stany Zjednoczone',
'2005/07/02', '');
insert into PRODUCENT values ('Firma H', 'laptopy', 'Stany Zjednoczone', '2005/04/15',
6);
insert into PRODUCENT values ('Firma I', 'karty graficzne', 'Chiny', '2017/10/09', 5);
insert into PRODUCENT values ('Firma J', 'procesory', 'Stany Zjednoczone',
'2000/05/12', 8);
insert into PRODUCENT values ('Firma K', 'procesory', 'Francja', '2002/07/23', 4);
insert into PRODUCENT values ('Firma L', 'monitory', 'Francja', '2000/04/15', 3);
```

-- 7. PRODUKT

```
insert into PRODUKT values ('GTX 1080', 1799.99, 'karta graficzna', 'wypuszczona na
rynek w 2016 roku', 'Firma A');
insert into PRODUKT values ('RTX 2060', 3299.99, 'karta graficzna', 'wypuszczona na
rynek w 2020 roku', 'Firma A');
insert into PRODUKT values ('RTX 4090', 3899.79, 'karta graficzna', '', 'Firma A');
insert into PRODUKT values ('i5-8400', 684.99, 'procesor', 'wypuszczony na rynek w 2017
roku', 'Firma J');
insert into PRODUKT values ('RX 7900 XTX', 2799, 'karta graficzna', '', 'Firma F');
insert into PRODUKT values ('i5-13600KF', 1529, 'procesor', '', 'Firma J');
insert into PRODUKT values ('i7-13700K', 2169, 'procesor', '', 'Firma J');
insert into PRODUKT values ('i9-13900KF', 2849, 'procesor', '', 'Firma J');
insert into PRODUKT values ('Acer EK240YCb', 479, 'monitor', 'dobrej jakosci monitor',
'Firma C');
insert into PRODUKT values ('G4M3R HERO', 7800, 'komputer', '', 'Firma G');
insert into PRODUKT values ('Logitech G PRO X GAMING', 499, 'sluchawki', '', 'Firma
E');
insert into PRODUKT values ('HP X24ih Gaming', 479, 'monitor', '', 'Firma L');
```



```

insert into MAGAZYN values (seq_id_magazyn.nextval, '2000 m^3', '500 m^2', 'Magazyn nr
1','Sklep_Katowice');
insert into MAGAZYN values (seq_id_magazyn.nextval, '2550 m^3', '600 m^2', 'Magazyn nr
2','Sklep_Katowice');
insert into MAGAZYN values (seq_id_magazyn.nextval, '1800 m^3', '400 m^2', 'Magazyn nr
3','Sklep_Katowice');

insert into MAGAZYN values (seq_id_magazyn.nextval, '1550 m^3', '420 m^2', 'Magazyn nr
1','Sklep_Krakow');
insert into MAGAZYN values (seq_id_magazyn.nextval, '3000 m^3', '700 m^2', 'Magazyn nr
2','Sklep_Krakow');

insert into MAGAZYN values (seq_id_magazyn.nextval, '1800 m^3', '400 m^2',
'', 'Sklep_Gdansk');

insert into MAGAZYN values (seq_id_magazyn.nextval, '1800 m^3', '400 m^2',
'', 'Sklep_Szczecin');

insert into MAGAZYN values (seq_id_magazyn.nextval, '4000 m^3', '500 m^2',
'', 'Sklep_Lodz');

insert into MAGAZYN values (seq_id_magazyn.nextval, '5000 m^3', '1000 m^2',
'', 'Sklep_Wroclaw');

insert into MAGAZYN values (seq_id_magazyn.nextval, '2300 m^3', '700 m^2',
'', 'Sklep_Poznan');

insert into MAGAZYN values (seq_id_magazyn.nextval, '2700 m^3', '700 m^2',
'', 'Sklep_Gdynia');

insert into MAGAZYN values (seq_id_magazyn.nextval, '1800 m^3', '550 m^2',
'', 'Sklep_Bydgoszcz');

insert into MAGAZYN values (seq_id_magazyn.nextval, '1800 m^3', '400 m^2',
'', 'Sklep_Lublin');

insert into MAGAZYN values (seq_id_magazyn.nextval, '1800 m^3', '400 m^2',
'', 'Sklep_Rzeszow');

insert into MAGAZYN values (seq_id_magazyn.nextval, '1000 m^3', '380 m^2',
'', 'Sklep_Opole');

-- 11. REGAL
--katowice 1
insert into REGAL values ('GTX 1080', 1, '1A', 200, '2022/07/04');
insert into REGAL values ('RTX 2060', 1, '2A', 450, '2022/04/04');
insert into REGAL values ('RTX 4090', 1, '2B', 200, '2022/07/04');
-- katowice 2
insert into REGAL values ('i5-13600KF', 2, '1D', 376, '2022/03/11');
insert into REGAL values ('i7-13700K', 2, '3A', 874, '2021/03/28');
insert into REGAL values ('i9-13900KF', 2, '4F', 937, '2022/09/09');
-- katowice 3
insert into REGAL values ('Acer EK240YCbi', 3, '1A', 510, '2021/03/28');
insert into REGAL values ('HP X24ih Gaming', 3, '1C', 124, '2022/09/08');
-- krakow 1
insert into REGAL values ('GTX 1080', 4, '1A', 873, '2022/10/19');
insert into REGAL values ('RTX 2060', 4, '2D', 431, '2022/04/14');
insert into REGAL values ('i9-13900KF', 4, '1D', 785, '2022/06/30');
-- krakow 2
insert into REGAL values ('G4M3R HERO', 5, '1A', 430, '2022/05/29');

-- gdansk
insert into REGAL values ('G4M3R HERO', 6, '6D', 108, '2022/02/14');

-- szczecin
insert into REGAL values ('G4M3R HERO', 7, '1A', 1400, '2022/11/03');
insert into REGAL values ('Logitech G PRO X GAMING', 7, '1B', 2600, '2021/12/24');

```

[illegible]

```

insert into PRACOWNIK values (seq_id_pracownik.nextval, 'Alan', 'Kalinowski',
'ul.Osiedle Młodych 1/14, Gdynia', 4351, 11);

insert into PRACOWNIK values (seq_id_pracownik.nextval, 'Janusz', 'Szulc',
'ul.Krakowska 32, Bydgoszcz', 5293, 12);

insert into PRACOWNIK values (seq_id_pracownik.nextval, 'Olaf', 'Szymański',
'ul.Warszawska 45, Lublin', 4023, 13);

insert into PRACOWNIK values (seq_id_pracownik.nextval, 'Mateusz', 'Urbański',
'ul.Perłowa 19, Rzeszów', 3770, 14);

insert into PRACOWNIK values (seq_id_pracownik.nextval, 'Dawid', 'Pawlak', 'ul.Królowej
Jadwigi 3/20, Opole', 2609, 15);

```

Tworzenie zestawień:

```

-- Złączenie wewnętrzne 4 tabel z klauzulą HAVING
-- Zestawienie wypisuje klientów, którzy wystawili więcej niż jedno zamówienie
-- oraz średnia ocenę z tych zamówień

```

```

SELECT klient.id_klient, ROUND(AVG(feedback.ocena),2) as "Średnia ocena",
COUNT(zamowienie.id_zamowienie) as "Liczba zamówień"
FROM zamowienie
INNER JOIN klient
ON zamowienie.klient_id_klient = klient.id_klient
INNER JOIN feedback
ON zamowienie.id_zamowienie = feedback.zamowienie_id_zamowienie
INNER JOIN kupon
ON zamowienie.kupon_id_kuponu = kupon.id_kuponu
GROUP BY klient.id_klient
HAVING AVG(feedback.ocena) >= 4 AND COUNT(zamowienie.id_zamowienie) >= 2;

```

	ID_KLIENT	Średnia ocena	Liczba zamówień
1	1	8,67	3
2	4	8	2

```

-- Podzapytanie nieskorelowane z użyciem operatora IN
-- Wykaz wszystkich zamówień w których jeden z produktów ma kolor czerwony
SELECT *
FROM zamowienie
WHERE id_zamowienie IN (SELECT zamowienie_id_zamowienie
                        FROM pozycje_zamowien
                        WHERE kolor = 'czerwony');

```

	ID_ZAMOWIENIE	NR_FAKTURY	STATUS	METODA_ODBIORU	DATA_UTWORZENIA	KUPON_ID_KUPONU	USLUGI_ID_USLUGI	KLIENT_ID_KLIENT
1	1	3048	zrealizowano na miejscu		22/07/05	2	7	1
2	4	3928	zrealizowano na miejscu		22/07/02	8	6	12

```

-- Podzapytanie skorelowane z użyciem operatora ALL
-- Wykaz firm tworzących karty graficzne, które istnieją na rynku krócej niż wszystkie
firmy z pozostałych dziedzin
SELECT *
FROM producent x
WHERE x.dziedzina = 'karty graficzne' AND extract (year from sysdate) - extract (year
from x.data_zalozenia) < ALL
(select extract (year from sysdate) - extract (year from y.data_zalozenia)
FROM producent y
WHERE y.dziedzina != x.dziedzina);

```

	NAZWA	DZIEDZINA	KRAJ	DATA_ZALOZENIA	RATING
1	Firma I	karty graficzne	Chiny	17/10/09	5

```
--Operacja agregująca z klauzulą GROUP
-- Wykaz wszystkich srednich zarobków z wszystkich magazynow z kazdej filii firmy/miasta
SELECT magazyn.oddzial_nazwa_oddzial, magazyn_id_magazyn, ROUND(AVG(pensja),2) AS
średnia_pensja
FROM pracownik
INNER JOIN magazyn
ON magazyn.id_magazyn = pracownik.magazyn_id_magazyn
GROUP BY magazyn_id_magazyn, magazyn.oddzial_nazwa_oddzial
ORDER BY magazyn.oddzial_nazwa_oddzial;
```

	⚡ ODDZIAL_NAZWA_ODDZIAL	⚡ MAGAZYN_ID_MAGAZYN	⚡ ŚREDNIA_PENSJA
1	Sklep_Bydgoszcz	12	5293
2	Sklep_Gdansk	6	4655
3	Sklep_Gdynia	11	4351
4	Sklep_Katowice	1	3306,67
5	Sklep_Katowice	2	2964,5
6	Sklep_Katowice	3	4039
7	Sklep_Krakow	4	3200
8	Sklep_Krakow	5	4594
9	Sklep_Lodz	8	2770
10	Sklep_Lublin	13	4023
11	Sklep_Opole	15	2609
12	Sklep_Poznan	10	2514
13	Sklep_Rzeszow	14	3770
14	Sklep_Szczecin	7	2596
15	Sklep_Wroclaw	9	3424

```
-- Widoki
-- Tworzenie perspektywy na podstawie poprzedniego przykladu
create or replace view perspektywa as
SELECT magazyn.oddzial_nazwa_oddzial, magazyn_id_magazyn, ROUND(AVG(pensja),2) AS
średnia_pensja
FROM pracownik
INNER JOIN magazyn
ON magazyn.id_magazyn = pracownik.magazyn_id_magazyn
GROUP BY magazyn_id_magazyn, magazyn.oddzial_nazwa_oddzial
ORDER BY magazyn.oddzial_nazwa_oddzial;
```

```
select * from perspektywa;
```

	⚡ ODDZIAL_NAZWA_ODDZIAL	⚡ MAGAZYN_ID_MAGAZYN	⚡ ŚREDNIA_PENSJA
1	Sklep_Bydgoszcz	12	5293
2	Sklep_Gdansk	6	4655
3	Sklep_Gdynia	11	4351
4	Sklep_Katowice	1	3306,67
5	Sklep_Katowice	2	2964,5
6	Sklep_Katowice	3	4039
7	Sklep_Krakow	4	3200
8	Sklep_Krakow	5	4594
9	Sklep_Lodz	8	2770
10	Sklep_Lublin	13	4023
11	Sklep_Opole	15	2609
12	Sklep_Poznan	10	2514
13	Sklep_Rzeszow	14	3770
14	Sklep_Szczecin	7	2596
15	Sklep_Wroclaw	9	3424

-- Zmienne związane

```
select * from PRACOWNIK
where pracownik.nazwisko = :nazwisko
order by pracownik.nazwisko;
```

Dialog box titled "Enter Binds" showing a variable named "nazwisko" with a value of "Kalinowski". The dialog includes a "Name:" field, a "Value:" field, and a "NULL" checkbox. Below the dialog, a table with 7 columns is visible: ID_PRACOWNIK, IMIE, NAZWISKO, ADRES, PENSJA, and MAGAZYN_ID_MAGAZYN. The table contains 3 rows of data.

	ID_PRACOWNIK	IMIE	NAZWISKO	ADRES	PENSJA	MAGAZYN_ID_MAGAZYN
1	17	Andrzej	Kalinowski	ul.Koszykowa 2/13, Wroclaw	3424	9
2	18	Bolesław	Kalinowski	ul.Milenijna 20, Warszawa	2514	10
3	19	Alan	Kalinowski	ul.Osiedle Młodych 1/14, Gdynia	4351	11

Tworzenie aktualizacji(pojedynczych rekordów i grup rekordów):

```
--aktualizacja jednego rekordu
--1.
update MAGAZYN
-- dolaczenie notatki do wybranego magazynu
set notatka = 'Magazyn nr 1'
where id_magazyn = 15;
-- aktualizacja grupy rekordów
-- 2.
-- podwyżka dla Mateuszow
update PRACOWNIK
set pensja = 10000
where imie = 'Mateusz';

-- 3.
-- zmiana ceny dla procesorow
update PRODUKT
set cena = cena + 10000
where nazwa like 'i%';
-- 4.
-- zmiana czasu oczekiwania dla wszystkich uslug
update USLUGI
set czas_realizacji = '7 dni';
-- aktualizacja jednego rekordu
-- 5.
-- zmiana nr telefonu wybranej osoby
update KLIENT
set nr_telefonu = '111222333'
where imie = 'Jolanta' and nazwisko = 'Nowakowska';
```

Tworzenie usuwania(pojedynczych rekordów i grup rekordów):

```
-- usuniecie grupy rekordow
--1. usuniecie kart graficznych z pozycji zamowien o nazwie GTX 1080
delete POZYCJE_ZAMOWIEN
where produkt_nazwa = 'GTX 1080';
-- usuniecie jednego rekordu
--2. usuniecie jednego regalu
delete REGAL
where magazyn_id_magazyn = 15;
--usuniecie grupy rekordow
-- 3. usuniecie wszystkich kuponow ze zniżka inna niz 0,15 0,4 0,7 oraz 0,3
delete KUPON
where zniżka NOT IN (0.15, 0.4, 0.7, 0.3);
--usuniecie grupy rekordow
-- 4. usuniecie uslug wiekszych z id wiekszym niz 7
delete USLUGI
where id_uslugi > 7;
-- usuniecie grupy rekordow
-- 5. usuniecie 3 klientow
delete KLIENT
where id_klient between 9 and 11;
```