

POLITECHNIKA ŚWIĘTOKRZYSKA

WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI I INFORMATYKI

PROJEKT BAZY DANYCH 2	
TEMAT: SIEĆ SALONÓW KOMÓRKOWYCH	
ZESPÓŁ: Cioć Karol, Szemraj Filip	GRUPA: 2ID11B

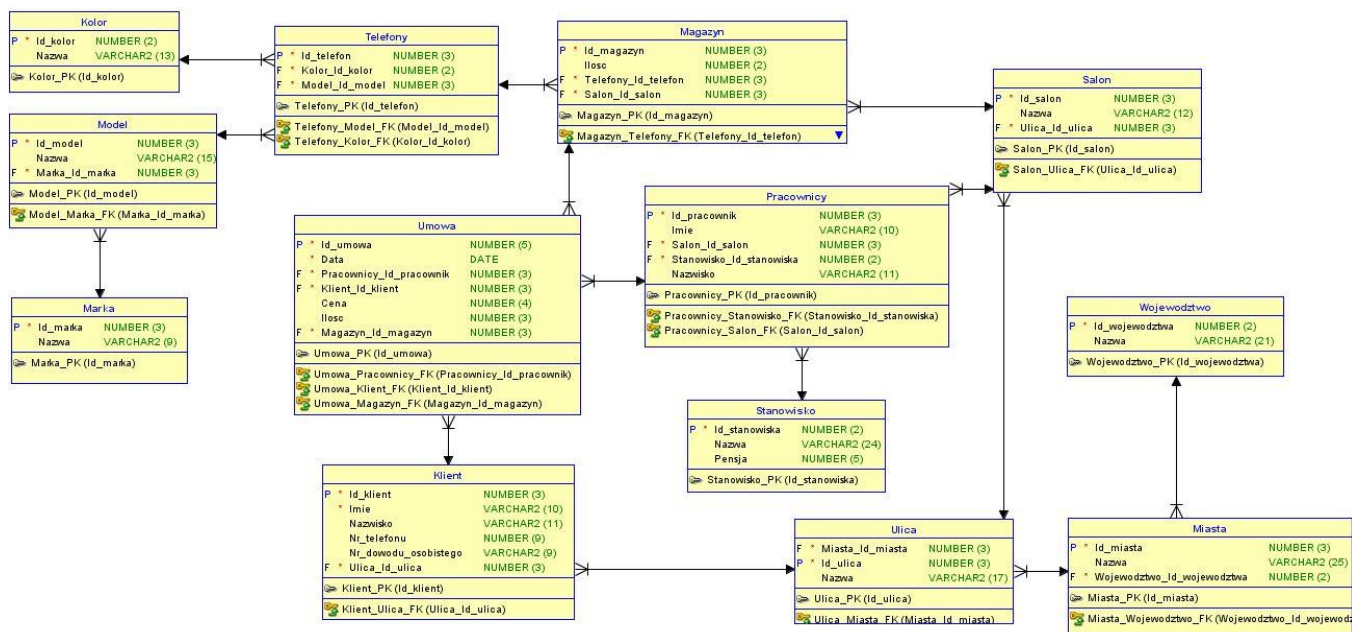
1.Opis problemu oraz założenia projektowe.....	2
2.Schemat ERD bazy danych.....	3
3.Opis tabel.....	4
4.Opis procesu transformacji bazy do hurtowni.	5
5.Schemat ERD hurtowni danych.	6
6.Opis procesu ładowania i transformacji danych z bazy do hurtowni.....	6
7.Porównanie wykonania zapytań między bazą danych a hurtownią danych.....	9
ROLLUP.....	9
BAZA DANYCH: Zestawienie ilości umów w danym mieście i salonach.	9
HURTOWNIA DANYCH	9
BAZA DANYCH: Zestawienie ilości umów w danych salonach dla pracowników.	10
HURTOWNIA DANYCH	11
BAZA DANYCH: Ilość zawartych umów dla danych modeli telefonów.....	12
HURTOWNIA DANYCH	12
CUBE.....	14
BAZA DANYCH: Liczba klientów dla kombinacji marek telefonów, modeli, kolorów.	14
HURTOWNIA DANYCH	14
BAZA DANYCH: Średnia cena umów dla kombinacji marek telefonów i stanowisk pracowników.	15
HURTOWNIA DANYCH	15
BAZA DANYCH: Średnia wartość umów w danym województwie.	16
HURTOWNIA DANYCH	17
Partycje obliczeniowe	18
BAZA DANYCH: Średnia wydanych pieniędzy w danym salonie.....	18
HURTOWNIA DANYCH	18
BAZA DANYCH: Średnia wartość umowy w danych wojewodztwach	19
HURTOWNIA DANYCH	19
BAZA DANYCH: Sprzedaż danego modelu.	20
HURTOWNIA DANYCH	21
Okna.....	22
BAZA DANYCH: Roczna liczba zawarcia umów w miastach.....	22
HURTOWNIA DANYCH	22
BAZA DANYCH: Ilość zawartych umów w danym salonie w każdym roku.	23
HURTOWNIA DANYCH	23
BAZA DANYCH: Miesięczna różnica średniej wartości umowy w danych miastach.....	24

HURTOWNIA DANYCH	25
Rankingi.....	26
BAZA DANYCH: Ranking ilości zawartych umów przez pracowników.	27
HURTOWNIA DANYCH	27
BAZA DANYCH: Ranking ilości umów w danych salonach.	28
HURTOWNIA DANYCH	28
BAZA DANYCH: Ranking zawartych umów przez klientów.	29
HURTOWNIA DANYCH	30
8.Uwagi i wnioski.	30

1. Opis problemu oraz założenia projektowe.

Tematem naszego projektu jest sieć salonów komórkowych. Na początku stworzyliśmy schemat bazy danych, a następnie musieliśmy go przekształcić na hurtownię danych. W naszej bazie danych klient może zawierać wiele umów, na której może być wiele telefonów, sam klient jest opisywany za pomocą imienia, nazwiska, numeru telefonu oraz numeru dowodu osobistego. Pracownik może pracować tylko w jednym salonie komórkowym oraz może mieć tylko jedno stanowisko, które definiuje wartość jego pensji. Telefony w naszej bazie są opisywane za pomocą marki, modelu oraz koloru. Każdy salon posiada magazyn, w którym znajdują się określone telefony oraz ich ilość. Dany telefon może być w wielu salonach oraz salon może mieć wiele danych telefonów. Klient oraz salony posiadają dane adresowe w postaci ulicy, miasta oraz województwa.

2. Schemat ERD bazy danych.



1 Schemat ERD bazy danych.

3. Opis tabel.

- Tabela Marka zawiera klucz główny **Id_marka** NUMBER(3) o typie numerycznym oraz pole znakowe VARCHAR2(15) **Nazwa**, które przechowuje nazwę danej marki.
- Tabela Model zawiera klucz główny **Id_model** o typie numerycznym NUMBER(3), pole znakowe VARCHAR2(15) **Nazwa**, które przechowuje nazwę danego modelu oraz klucz obcy **Marka_Id_marka** o typie numerycznym NUMBER(3), który jest kluczem głównym w tabeli Marka.
- Tabela Kolor zawiera klucz główny **Id_kolor** o typie numerycznym NUMBER(2) oraz pole znakowe VARCHAR2(20) **Nazwa**, które przechowuje nazwę danego koloru.
- Tabela Telefony zawiera klucz główny **Id_telefon** o typie numerycznym NUMBER(3) oraz dwa klucze obce **Kolor_Id_kolor** NUMBER(2) i **Model_Id_model** NUMBER(3).
- Tabela Magazyn zawiera klucz główny **Id_magazyn** o typie numerycznym NUMBER(3), pole **Ilosc** typu numerycznego NUMBER(3), które przechowuje dane na temat ilości telefonów oraz dwa klucze obce **Telefony_Id_telefon** NUMBER(3) oraz **Salon_Id_salon** NUMBER(3).
- Tabela Salon zawiera klucz główny **Id_salon** o typie numerycznym NUMBER(3), pole znakowe **Nazwa** VARCHAR2(12), które przechowuje nazwę danego salonu oraz klucz obcy **Ulica_Id_ulica** NUMBER(3), który pozwala sprawdzić dane adresowe salonu.
- Tabela Ulica zawiera klucz główny **Id_ulica** o typie numerycznym NUMBER(3), pole znakowe **Nazwa** VARCHAR2(17), które przechowuje nazwę danej ulicy oraz klucz obcy **Miasta_Id_miasta** NUMBER(2).

- Tabela Miasta zawiera klucz główny **Id_miasta** o typie numerycznych NUMBER(2), pole znakowe **Nazwa** VARCHAR2(25), które przechowuje nazwę danego miasta oraz klucz obcy **Wojewodztwo_Id_wojewodztwa** NUMBER(2).
- Tabela Wojewodztwo zawiera klucz główny **Id_wojewodztwa** o typie numerycznym NUMBER(2) oraz pole znakowe **Nazwa** VARCHAR2(21), które przechowuje nazwę danego województwa.
- Tabela Klient zawiera klucz główny **Id_klient** o typie numerycznym NUMBER(3), pole znakowe **Imie** VARCHAR2(10), które przechowuje imię klienta, pole znakowe **Nazwisko** VARCHAR2(11), które przechowuje nazwisko klienta, pole numeryczne **Nr_telefonu** NUMBER(9), które przechowuje numer telefonu klienta, pole znakowe **Nr_dowodu_osobistego** VARCHAR2(9), które przechowuje numer dowodu osobistego klienta oraz klucz obcy **Ulica_Id_ulica** NUMBER(3), który pozwala zidentyfikować dane adresowe klienta.
- Tabela Pracownicy zawiera klucz główny **Id_pracownik** o typie numerycznym NUMBER(3), pole znakowe **Imie** VARCHAR2(10), które przechowuje imię pracownika, pole znakowe **Nazwisko** VARCHAR2(11), które przechowuje nazwisko pracownika, oraz dwa klucze główne **Salon_Id_Salon** NUMBER(3), pozwala nam zidentyfikować w którym salonie dany pracownik pracuje, **Stanowisko_Id_stanowisko** NUMBER(2), pozwala zidentyfikować nazwę stanowiska oraz pensję danego pracownika.
- Tabela Stanowisko zawiera klucz główny **Id_stanowiska** o typie numerycznym NUMBER(2), pole znakowe **Nazwa** VARCHAR2(24), które przechowuje nazwę danego stanowiska oraz pole numeryczne **Pensja** NUMBER(5), które przechowuje wartość pensji przypisaną do danego stanowiska.

4. Opis procesu transformacji bazy do hurtowni.

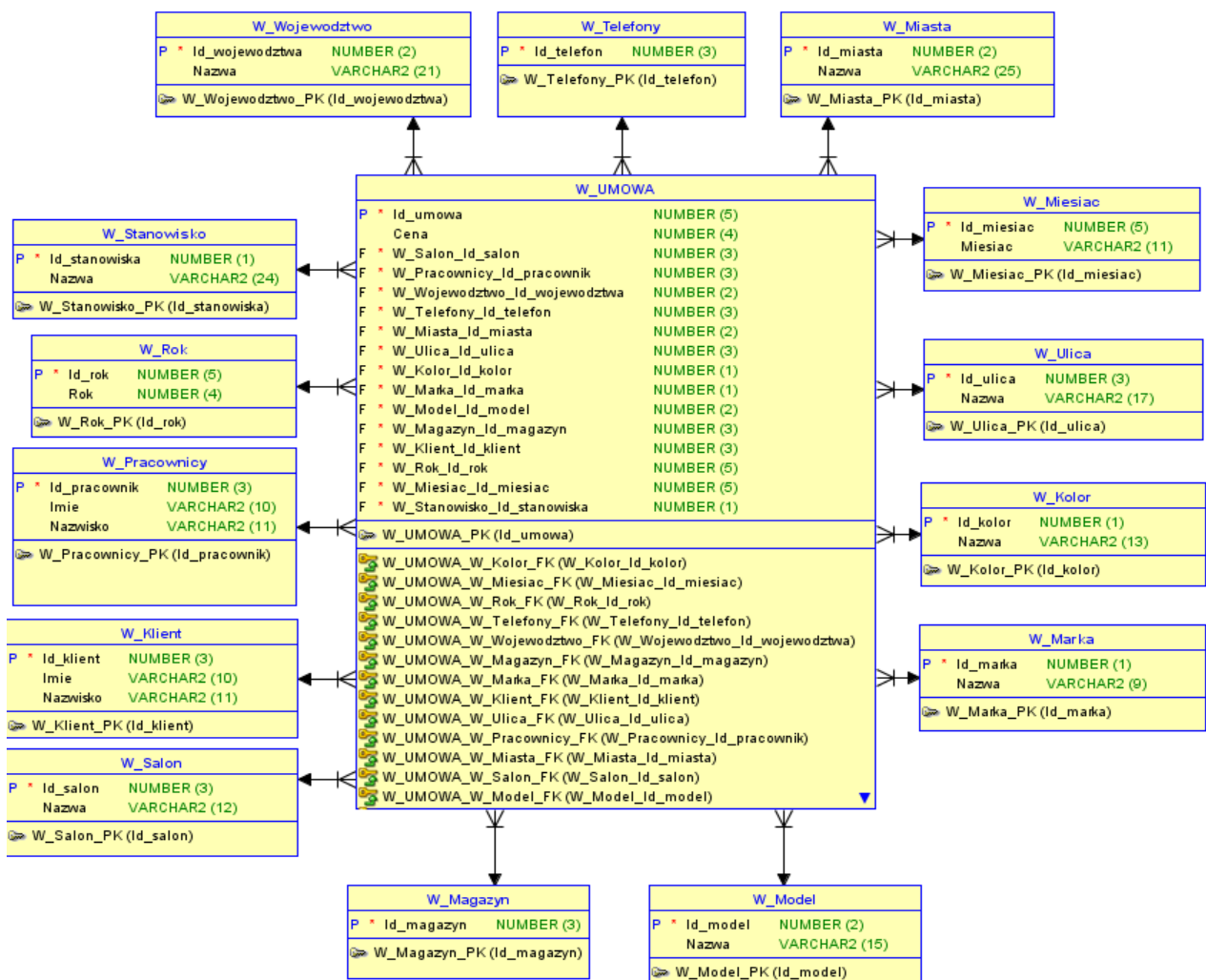
Hurtownia danych posiada informacje na temat zawieranych umów w salonach komórkowych. Tabelą faktów stała się tabela **W_UMOWA**, która zawiera klucz główny **Id_umowa**, pole numeryczne **Cena**, które przechowuje dane na temat ceny towaru na danej umowie oraz klucze obce do wszystkich tabel wymiarów, które są opisane poniżej. Tabele w hurtowni danych zawierają przedrostek **W_**, który oznacza warehouse czyli hurtownia.

- Tabela **Wojewodztwo** została przekształcona na tabelę wymiaru **W_Wojewodztwo**, która zawiera takie same pola jak w bazie danych.
- Tabela **Miasta** została przekształcona na tabelę wymiaru **W_Miasta**, która zawiera klucz główny **Id_miasta** oraz pole **VARCHAR2(25) Nazwa**, które przechowuje nazwę danego miasta.
- Tabela **Ulica** została przekształcona na tabelę wymiaru **W_Ulica**, która zawiera klucz główny **Id_ulica** oraz pole **VARCHAR2(16) nazwa**, które przechowuje nazwę danej ulicy.
- Tabela **Klient** została przekształcona na tabelę wymiaru **W_Klient**, która zawiera klucz główny **Id_klient**, pole **VARCHAR2(10) Imie** przechowujące imię klienta oraz pole **VARCHAR2(11) Nazwisko** przechowujące nazwisko klienta, reszta pól została usunięta ponieważ byłaby bezużyteczna.
- Tabela **Salon** została przekształcona na tabelę wymiaru **W_Salon**, która zawiera klucz główny **Id_salon** oraz pole **VARCHAR2(12) Nazwa**, które przechowuje nazwę salonu.

- Tabela **Pracownicy** została przekształcona na tabelę wymiaru **W_Pracownicy**, która zawiera klucz główny **Id_pracownik**, pole **VARCHAR2(10) Imie** przechowujące imię pracownika, pole **VARCHAR2(11) Nazwisko** przechowujące nazwisko pracownika.

- Tabela **Stanowisko** została przekształcona na tabelę wymiaru **W_Stanowisko**, która zawiera klucz główny **Id_stanowiska** oraz pole **VARCHAR2(24) Nazwa** przechowujące nazwę stanowiska.
- Tabela **Marka** została przekształcona na tabelę wymiaru **W_Marka**, która zawiera takie same pole jak tabela w bazie danych.
- Tabela **Model** została przekształcona na tabelę wymiaru **W_Model**, która zawiera klucz główny **Id_model** oraz pole **VARCHAR2(15) Nazwa** przechowujące nazwę modelu.
- Tabela **Kolor** została przekształcona na tabelę wymiaru **W_Kolor**, która zawiera takie same pole jak tabela w bazie danych.
- Tabela **Telefony** została przekształcona na tabelę wymiaru **W_Telefony**, która zawiera klucz główny **Id_telefon**.
- Tabela **Magazyn** została przekształcona na tabelę wymiaru **W_Magazyn**, która zawiera klucz główny **Id_magazyn**.

5. Schemat ERD hurtowni danych.



2 Schemat ERD hurtowni danych.

6. Opis procesu ładowania i transformacji danych z bazy do hurtowni.

W celu załadowania danych z bazy do hurtowni stworzyliśmy procedury, które będą wybierały dane pola z tabel znajdujących się w bazie danych, które będą przeniesione do tabel znajdujących się w hurtowni danych. Procedury są uruchamiane poleceniem `execute nazwa_procedury`.

```
create or replace procedure Load_Magazyn as
begin insert into W_Magazyn select Id_magazyn from Magazyn; end;
/
create or replace procedure Load_Kolor as
begin insert into W_Kolor select Id_kolor, Nazwa from Kolor; end;
/
create or replace procedure Load_Ulica as
begin insert into W_Ulica select Id_ulica, Nazwa from Ulica; end;
/
create or replace procedure Load_Miasta as
begin insert into W_Miasta select Id_miasta, Nazwa from Miasta; end;
/
create or replace procedure Load_Telefony as
begin insert into W_Telefony select Id_telefon from Telefony; end;
/
create or replace procedure Load_Wojewodztwo as
begin insert into W_Wojewodztwo select Id_wojewodztwa, Nazwa from
Wojewodztwo; end;
/
create or replace procedure Load_Pracownicy as
begin insert into W_Pracownicy select Id_pracownik, Imie, Nazwisko from
Pracownicy; end;
/
create or replace procedure Load_Stanowisko as
begin insert into W_Stanowisko select s.id_stanowiska, s.nazwa FROM
stanowisko s; end;
/
create or replace procedure Load_Salon as
begin insert into W_Salon select Id_salon, Nazwa from Salon; end;
/
create or replace procedure Load_Klient as
begin insert into W_Klient select Id_klient, Imie, Nazwisko from Klient;
end;
/
create or replace procedure Load_Model as
begin insert into W_Model select Id_model, Nazwa from Model; end;
/
create or replace procedure Load_Marka as
begin insert into W_Marka select Id_marka, Nazwa from Marka; end;
/
create or replace procedure Load_Rok as
begin insert into W_Rok select id_umowa, EXTRACT(YEAR FROM data) from umowa
; end;
/
create or replace procedure Load_Miesiac as
begin insert into W_Miesiac select id_umowa,
CASE EXTRACT(MONTH FROM data)
```

```

WHEN 1 THEN 'Styczen'
WHEN 2 THEN 'Luty'
WHEN 3 THEN 'Marzec'
WHEN 4 THEN 'Kwiecien'
WHEN 5 THEN 'Maj'
WHEN 6 THEN 'Czerwiec'
WHEN 7 THEN 'Lipiec'
WHEN 8 THEN 'Sierpien'
WHEN 9 THEN 'Wrzesien'
WHEN 10 THEN 'Pazdziernik'
WHEN 11 THEN 'Listopad'
WHEN 12 THEN 'Grudzien'
END FROM umowa; end;
/
create or replace procedure Load_Umowa as
begin insert into W_UMOWA
Select
umowa.id_umowa,umowa.cena,salon.id_salon,umowa.pracownicy_id_pracownik,
wojewodztwo.id_wojewodztwa,telefony.id_telefon,miasta.id_miasta,
ulica.id_ulica,kolor.id_kolor,marka.id_marka,model.id_model,
umowa.magazyn_id_magazyn,umowa.klient_id_klient,umowa.id_umowa,
umowa.id_umowa,stanowisko.id_stanowiska
FROM
    umowa,salon,pracownicy,wojewodztwo,telefony,miasta,ulica,kolor,
    marka,model,magazyn,stanowisko
    WHERE wojewodztwo.id_wojewodztwa = miasta.wojewodztwo_id_wojewodztwa
    AND miasta.id_miasta = ulica.miasta_id_miasta
    AND ulica.id_ulica = salon.ulica_id_ulica
    AND salon.id_salon = pracownicy.salon_id_salon
    AND stanowisko.id_stanowiska = pracownicy.stanowisko_id_stanowiska
    AND pracownicy.id_pracownik = umowa.pracownicy_id_pracownik
    AND magazyn.id_magazyn = umowa.magazyn_id_magazyn
    AND marka.id_marka = model.marka_id_marka
    AND model.id_model = telefony.model_id_model
    AND kolor.id_kolor = telefony.kolor_id_kolor
    AND telefony.id_telefon = magazyn.telefony_id_telefon;
end;
/

```

```

execute Load_Magazyn;
execute Load_Kolor;
execute Load_Ulica;
execute Load_Miasta;
execute Load_Telefony;
execute Load_Wojewodztwo;
execute Load_Pracownicy;
execute Load_Stanowisko;
execute Load_Salon;
execute Load_Klient;
execute Load_Model;
execute Load_Marka;
execute Load_Rok;
execute Load_Miesiac;
execute Load_UMOWA;

```

7. Porównanie wykonania zapytań między bazą danych a hurtownią danych.

ROLLUP

BAZA DANYCH: Zestawienie ilości umów w danym mieście i salonach.

```

SELECT CASE WHEN SALON = 'RAZEM' THEN 'RAZEM' ELSE (SELECT salon.nazwa FROM
salon WHERE id_salon = SALON) END AS NAZWA_SALONU,
CASE WHEN miasto = 'RAZEM' THEN 'RAZEM' ELSE (SELECT miasta.nazwa FROM
miasta WHERE id_miasta=miasto) END AS NAZWA_MIASTA,
SALON, ILOSC
FROM(
    SELECT NVL(TO_CHAR(miasta.id_miasta),'RAZEM') as miasto,
    NVL(TO_CHAR(salon.id_salon),'RAZEM') AS SALON,
    COUNT(umowa.id_umowa) AS ILOSC
    FROM miasta, ulica, umowa, salon, pracownicy
    WHERE miasta.id_miasta = ulica.miasta_id_miasta
    AND ulica.id_ulica = salon.ulica_id_ulica
    AND salon.id_salon = pracownicy.salon_id_salon
    AND umowa.pracownicy_id_pracownik = pracownicy.id_pracownik
    GROUP BY ROLLUP(miasta.id_miasta,salon.id_salon)
) ORDER BY ILOSC ASC;

```

HURTOWNIA DANYCH

```

SELECT CASE WHEN SALON = 'Wszystkie salony' THEN 'Wszystkie salony' ELSE
(SELECT w_salon.nazwa FROM w_salon WHERE id_salon = SALON)
END AS NAZWA_SALONU,
CASE WHEN MIASTO = 'Wszystkie miasta' THEN 'Wszystkie miasta' ELSE (SELECT
w_miasta.nazwa FROM w_miasta WHERE id_miasta = MIASTO)
END AS NAZWA_MIASTA, ILOSC
FROM(
SELECT NVL(TO_CHAR(w_miasta.id_miasta),'Wszystkie miasta') AS MIASTO,
NVL(TO_CHAR(w_salon.id_salon),'Wszystkie salony') AS SALON,
COUNT(w_umowa.id_umowa) ILOSC
FROM w_miasta, w_ulica, w_umowa, w_salon
WHERE w_miasta.id_miasta = w_umowa.w_miasta_id_miasta
AND w_ulica.id_ulica = w_umowa.w_ulica_id_ulica
AND w_salon.id_salon = w_umowa.w_salon_id_salon
GROUP BY ROLLUP(w_miasta.id_miasta,w_salon.id_salon)
) ORDER BY ILOSC ASC;

```

NAZWA_SALONU	NAZWA_MIASTA	SALON	ILOSC
1 'Play'	'Zielona Gora'	89	16
2 'Plus'	'Warszawa'	49	16
3 'Play'	'Poznan'	77	18
4 'Play'	'Opole'	85	21
5 'T-Mobile'	'Bialystok'	83	21
6 'Play'	'Krakow'	58	21
7 'Plus'	'Katowice'	14	33
8 'Orange'	'Gdansk'	5	38
9 'Plus'	'Warszawa'	97	39
10 'Play'	'Zielona Gora'	15	44
11 'Plus'	'Gdansk'	88	45
12 'T-Mobile'	'Opole'	33	46
13 'Orange'	'Opole'	74	48
14 'Play'	'Opole'	2	54
15 'Nju-mobile'	'Ostrowiec Swietokrzyski'	93	58
16 'Plus'	'Rzeszow'	98	63
17 'Nju-mobile'	'Bialystok'	25	64

3 Wynik pierwszego zapytania - ROLL UP.

Numer wykonania	Baza danych	Hurtownia danych
1	0,033	0,019
2	0,017	0,004
3	0,008	0,005
Średnia	0,0193	0,0093

BAZA DANYCH: Zestawienie ilości umów w danych salonach dla pracowników.

```
SELECT CASE WHEN ID_SAL IS NULL THEN 'Wszystkie_salony'
ELSE (SELECT nazwa FROM salon WHERE id_salon = ID_SAL) END AS NAZWA_SALONU,
CASE WHEN ID_SAL IS NULL THEN 0
ELSE (SELECT id_salon FROM salon WHERE id_salon = ID_SAL) END AS ID_SALONU,
(SELECT pracownicy.imie FROM pracownicy WHERE pracownicy.id_pracownik =
ID_PRAC) as IMIE,
(SELECT pracownicy.nazwisko FROM pracownicy WHERE pracownicy.id_pracownik =
ID_PRAC) as NAZWISKO, ILOSC
FROM (
    SELECT salon.id_salon AS ID_SAL, pracownicy.id_pracownik AS ID_PRAC,
    COUNT(umowa.id_umowa) AS ILOSC
    FROM pracownicy, umowa, salon
    WHERE salon.id_salon = pracownicy.salon_id_salon
    AND pracownicy.id_pracownik = umowa.pracownicy_id_pracownik
    GROUP BY ROLLUP(salon.id_salon, pracownicy.id_pracownik)
) ORDER BY ILOSC ASC;
```

HURTOWNIA DANYCH

```
SELECT CASE WHEN ID_SAL IS NULL THEN 'Wszystkie_salony'
ELSE (SELECT nazwa FROM w_salon WHERE id_salon = ID_SAL) END AS
NAZWA_SALONU,
CASE WHEN ID_SAL IS NULL THEN 0
ELSE (SELECT id_salon FROM w_salon WHERE id_salon = ID_SAL) END AS
ID_SALONU,
(SELECT w_pracownicy.imie FROM w_pracownicy WHERE w_pracownicy.id_pracownik
= ID_PRAC) as IMIE,
(SELECT w_pracownicy.nazwisko FROM w_pracownicy WHERE
w_pracownicy.id_pracownik = ID_PRAC) as NAZWISKO, ILOSC
FROM (
    SELECT w_salon.id_salon AS ID_SAL, w_pracownicy.id_pracownik AS
ID_PRAC, COUNT(w_umowa.id_umowa) AS ILOSC
    FROM w_pracownicy, w_umowa, w_salon
    WHERE w_salon.id_salon = w_umowa.w_salon_id_salon
    AND w_pracownicy.id_pracownik = w_umowa.w_pracownicy_id_pracownik
    GROUP BY ROLLUP(w_salon.id_salon, w_pracownicy.id_pracownik)
) ORDER BY ILOSC ASC;
```

	NAZWA_SALONU	ID_SALONU	IMIE	NAZWISKO	ILOSC
1	'Plus'	21	'Joachim'	'Kaczmarek'	8
2	'Play'	80	'Michał'	'Wojcik'	10
3	'Plus'	17	'Grzegorz'	'Kaczmarek'	10
4	'Play'	92	'Karol'	'Zalecki'	10
5	'Play'	57	'Lukasz'	'Adamczyk'	10
6	'Play'	3	'Denis'	'Wozniak'	11
7	'Plus'	98	'Grzegorz'	'Dudek'	12
8	'Play'	2	'Jarosław'	'Pazurek'	12
9	'Orange'	5	'Jarosław'	'Wojcik'	12
10	'Plus'	10	'Konrad'	'Pazurek'	12
11	'Nju-mobile'	69	'Jarosław'	'Dudek'	13
12	'Plus'	72	'Michał'	'Pazurek'	13
13	'Play'	67	'Grzegorz'	'Krawczyk'	13
14	'Nju-mobile'	99	'Mateusz'	'Krawczyk'	13
15	'Play'	39	'Piotr'	'Banabak'	13
16	'Play'	44	'Bartosz'	'Wojcik'	13
17	'Orange'	82	'Bartosz'	'Wozniak'	13
18	'Plus'	95	'Gabriel'	'Pazurek'	13

4 Wynik drugiego zapytania - ROLL UP.

Numer wykonania	Baza danych	Hurtownia danych
1	0,044	0,053
2	0,041	0,016
3	0,041	0,037
Średnia	0,042	0,0353

BAZA DANYCH: Ilość zawartych umów dla danych modeli telefonów.

```

SELECT CASE WHEN MARKA = 'Wszystkie marki' THEN 'Wszystkie marki'
ELSE (SELECT marka.nazwa FROM marka WHERE id_marka = MARKA)END AS
NAZWA_MARKI,
CASE WHEN MODEL = 'Wszystkie modele' THEN 'Wszystkie modele'
ELSE (SELECT model.nazwa FROM model WHERE id_model = MODEL)END AS
NAZWA_MODELU,
CASE WHEN KOLOR = 'Wszystkie kolory' THEN 'Wszystkie kolory'
ELSE (SELECT kolor.nazwa FROM kolor WHERE id_kolor = KOLOR)END AS
NAZWA_KOLORU,ILOSC
FROM (
    SELECT NVL(TO_CHAR(marka.id_marka),'Wszystkie marki') as MARKA,
    NVL(TO_CHAR(model.id_model),'Wszystkie modele') AS MODEL,
    NVL(TO_CHAR(kolor.id_kolor),'Wszystkie kolory') AS KOLOR,
    COUNT(umowa.id_umowa) AS ILOSC
    FROM model, telefony, magazyn, umowa, marka, kolor
    WHERE kolor.id_kolor = telefony.kolor_id_kolor
    AND marka.id_marka = model.marka_id_marka
    AND model.id_model = telefony.model_id_model
    AND telefony.id_telefon = magazyn.telefony_id_telefon
    AND umowa.magazyn_id_magazyn = magazyn.id_magazyn
    GROUP BY ROLLUP(marka.id_marka,model.id_model,kolor.id_kolor)
)ORDER BY ILOSC ASC;

```

HURTOWNIA DANYCH

```

SELECT CASE WHEN MARKA = 'Wszystkie marki' THEN 'Wszystkie marki'
ELSE (
SELECT w_marka.nazwa FROM w_marka WHERE id_marka = MARKA) END AS
NAZWA_MARKI,
CASE WHEN MODEL = 'Wszystkie modele' THEN 'Wszystkie modele' ELSE (
SELECT w_model.nazwa FROM w_model WHERE id_model = MODEL) END AS
NAZWA_MODELU,
CASE WHEN KOLOR = 'Wszystkie kolory' THEN 'Wszystkie kolory'
ELSE (
SELECT w_kolor.nazwa FROM w_kolor WHERE id_kolor = KOLOR) END AS
NAZWA_KOLORU, ILOSC
FROM (
SELECT NVL(TO_CHAR(w_marka.id_marka), 'Wszystkie marki') as MARKA,
NVL(TO_CHAR(w_model.id_model), 'Wszystkie modele') AS MODEL,
NVL(TO_CHAR(w_kolor.id_kolor), 'Wszystkie kolory') AS KOLOR,
COUNT(w_umowa.id_umowa) AS ILOSC
FROM w_model, w_umowa, w_marka, w_kolor
WHERE w_marka.id_marka = w_umowa.w_marka_id_marka
AND w_model.id_model = w_umowa.w_model_id_model
AND w_kolor.id_kolor = w_umowa.w_kolor_id_kolor
GROUP BY ROLLUP(w_marka.id_marka, w_model.id_model, w_kolor.id_kolor)
);

```

	NAZWA_MARKI	NAZWA_MODELU	NAZWA_KOLORU	ILOSC
1	'Samsung'	'S22'	'Czarny'	88
2	'Samsung'	'S22'	'Srebrny'	89
3	'Huawei'	'Nova 10 SE'	'Wszystkie kolory'	91
4	'iPhone'	'13 Max'	'Wszystkie kolory'	91
5	'iPhone'	'13 Max'	'Czarny'	91
6	'Huawei'	'Nova 10 SE'	'Gwiazdzisty'	91
7	'Samsung'	'S20'	'Srebrny'	92
8	'Sony'	'Xperia Pro IX'	'Wszystkie kolory'	95
9	'Samsung'	'S23'	'Wszystkie kolory'	95
10	'Sony'	'Xperia Pro IX'	'Teczowy'	95
11	'Samsung'	'S23'	'Srebrny'	95
12	'Xiaomi'	'A1'	'Fioletowy'	98
13	'Huawei'	'Nova 11i'	'Gwiazdzisty'	98
14	'Huawei'	'Nova 10 Pro'	'Gwiazdzisty'	98
15	'Xiaomi'	'A1'	'Wszystkie kolory'	98
16	'Huawei'	'Nova 10 Pro'	'Wszystkie kolory'	98
17	'Sony'	'Xperia VII'	'Czerwony'	99

5 Wynik trzeciego zapytania - ROLL UP.

Numer wykonania	Baza danych	Hurtownia danych
1	0,023	0,013
2	0,011	0,01
3	0,008	0,013
Średnia	0,014	0,012

CUBE

BAZA DANYCH: Liczba klientów dla kombinacji marek telefonów, modeli, kolorów.

```
SELECT CASE WHEN MARKA = 'Wszystkie marki' THEN 'Wszystkie marki' ELSE (
SELECT nazwa FROM marka WHERE id_marka = MARKA) END AS NAZWA_MARKI,
CASE WHEN MODEL = 'Wszystkie modele' THEN 'Wszystkie modele' ELSE (
SELECT nazwa FROM model WHERE id_model = MODEL) END AS NAZWA_MODELU,
CASE WHEN KOLOR = 'Wszystkie kolory' THEN 'Wszystkie kolory' ELSE (
SELECT nazwa FROM kolor WHERE id_kolor = KOLOR) END AS NAZWA_KOLORU,
LICZBA_KLIENTOW
FROM (
SELECT NVL(TO_CHAR(marka.id_marka), 'Wszystkie marki') AS MARKA,
NVL(TO_CHAR(model.id_model), 'Wszystkie modele') AS MODEL,
NVL(TO_CHAR(kolor.id_kolor), 'Wszystkie kolory') AS KOLOR,
COUNT(klient.id_klient) AS LICZBA_KLIENTOW
FROM marka, model, telefony, magazyn, umowa, klient, kolor
WHERE marka.id_marka = model.marka_id_marka
AND model.id_model = telefony.model_id_model
AND telefony.id_telefon = magazyn.telefony_id_telefon
AND magazyn.id_magazyn = umowa.magazyn_id_magazyn
AND umowa.klient_id_klient = klient.id_klient
AND telefony.kolor_id_kolor = kolor.id_kolor
GROUP BY CUBE(marka.id_marka, model.id_model, kolor.id_kolor)
);
```

HURTOWNIA DANYCH

```
SELECT CASE WHEN MARKA = 'Wszystkie marki' THEN 'Wszystkie marki' ELSE (
SELECT nazwa FROM w_marka WHERE id_marka = MARKA) END AS NAZWA_MARKI,
CASE WHEN MODEL = 'Wszystkie modele' THEN 'Wszystkie modele' ELSE (
SELECT nazwa FROM w_model WHERE id_model = MODEL) END AS NAZWA_MODELU,
CASE WHEN KOLOR = 'Wszystkie kolory' THEN 'Wszystkie kolory' ELSE (
SELECT nazwa FROM w_kolor WHERE id_kolor = KOLOR) END AS NAZWA_KOLORU,
LICZBA_KLIENTOW
FROM (
SELECT NVL(TO_CHAR(w_marka.id_marka), 'Wszystkie marki') AS MARKA,
NVL(TO_CHAR(w_model.id_model), 'Wszystkie modele') AS MODEL,
NVL(TO_CHAR(w_kolor.id_kolor), 'Wszystkie kolory') AS KOLOR,
COUNT(w_klient.id_klient) AS LICZBA_KLIENTOW
FROM w_marka, w_model, w_umowa, w_klient, w_kolor
WHERE w_marka.id_marka = w_umowa.w_marka_id_marka
AND w_model.id_model = w_umowa.w_model_id_model
AND w_kolor.id_kolor = w_umowa.w_kolor_id_kolor
AND w_klient.id_klient = w_umowa.w_klient_id_klient
GROUP BY CUBE(w_marka.id_marka, w_model.id_model, w_kolor.id_kolor)
);
```


NAZWA_MARKI	NAZWA_MODELU	NAZWA_KOLORU	LICZBA_KLIENTOW
1 Wszystkie marki	Wszystkie modele	Wszystkie kolory	10800
2 Wszystkie marki	Wszystkie modele	'Czarny'	1904
3 Wszystkie marki	Wszystkie modele	'Srebrny'	1517
4 Wszystkie marki	Wszystkie modele	'Gwiazdzisty'	1575
5 Wszystkie marki	Wszystkie modele	'Teczowy'	2136
6 Wszystkie marki	Wszystkie modele	'Czerwony'	1948
7 Wszystkie marki	Wszystkie modele	'Fioletowy'	1720
8 Wszystkie marki	'11 Pro'	Wszystkie kolory	121
9 Wszystkie marki	'11 Pro'	'Teczowy'	121
10 Wszystkie marki	'12 Mini'	Wszystkie kolory	330
11 Wszystkie marki	'12 Mini'	'Czarny'	105
12 Wszystkie marki	'12 Mini'	'Srebrny'	117
13 Wszystkie marki	'12 Mini'	'Fioletowy'	108
14 Wszystkie marki	'12 Pro'	Wszystkie kolory	207
15 Wszystkie marki	'12 Pro'	'Gwiazdzisty'	105
16 Wszystkie marki	'12 Pro'	'Teczowy'	102
17 Wszystkie marki	'12 Max'	Wszystkie kolory	306

6 Wynik pierwszego zapytania - CUBE.

Numer wykonania	Baza danych	Hurtownia danych
1	0,021	0,019
2	0,013	0,023
3	0,027	0,014
Średnia	0,0203	0,0186

BAZA DANYCH: Średnia cena umów dla kombinacji marek telefonów i stanowisk pracowników.

```

SELECT CASE WHEN MARKA = 'Wszystkie marki' THEN 'Wszystkie marki' ELSE (
SELECT nazwa FROM marka WHERE id_marka = MARKA) END AS NAZWA_MARKI,
CASE WHEN STANOWISKO = 'Średnia_cena_umowy' THEN 'Średnia_cena_umowy' ELSE (
SELECT nazwa FROM stanowisko WHERE id_stanowiska = STANOWISKO) END AS
NAZWA_STANOWISKA, SREDNIA_CENA_UMOW
FROM (
    SELECT NVL(TO_CHAR(marka.id_marka), 'Wszystkie marki') AS MARKA,
    NVL(TO_CHAR(stanowisko.id_stanowiska), 'Średnia_cena_umowy') AS
    STANOWISKO, ROUND(AVG(umowa.cena),2) AS SREDNIA_CENA_UMOW
    FROM marka, stanowisko, umowa, magazyn, pracownicy, telefony, model
    WHERE marka.id_marka = model.marka_id_marka
    AND model.id_model = telefony.model_id_model
    AND telefony.id_telefon = magazyn.Telefony_Id_telefon
    AND magazyn.id_magazyn = umowa.magazyn_id_magazyn
    AND umowa.pracownicy_id_pracownik = pracownicy.id_pracownik
    AND pracownicy.stanowisko_id_stanowiska = stanowisko.id_stanowiska
    GROUP BY CUBE(marka.id_marka, stanowisko.id_stanowiska)
);

```

HURTOWNIA DANYCH

```

SELECT CASE WHEN MARKA = 'Wszystkie marki' THEN 'Wszystkie marki' ELSE (
SELECT nazwa FROM w_marka WHERE id_marka = MARKA) END AS NAZWA_MARKI,
CASE WHEN STANOWISKO = 'Średnia_cena_umowy' THEN 'Średnia_cena_umowy' ELSE (
SELECT nazwa FROM w_stanowisko WHERE id_stanowiska = STANOWISKO) END AS
NAZWA_STANOWISKA, SREDNIA_CENA_UMOW
FROM (
    SELECT NVL(TO_CHAR(w_marka.id_marka), 'Wszystkie marki') AS MARKA,
    NVL(TO_CHAR(w_stanowisko.id_stanowiska), 'Średnia_cena_umowy') AS
    STANOWISKO,
    ROUND(AVG(w_umowa.cena),2) AS SREDNIA_CENA_UMOW
    FROM w_marka, w_umowa, w_pracownicy, w_stanowisko
    WHERE w_marka.id_marka = w_umowa.w_marka_id_marka
    AND w_pracownicy.id_pracownik = w_umowa.w_pracownicy_id_pracownik
    AND w_stanowisko.id_stanowiska = w_umowa.w_stanowisko_id_stanowiska
    GROUP BY CUBE(w_marka.id_marka, w_stanowisko.id_stanowiska)
);

```

NAZWA_MARKI	NAZWA_STANOWISKA	SREDNIA_CENA_UMOW
1 Wszystkie marki	Średnia_cena_umowy	2967,21
2 Wszystkie marki	'Doradca'	2998,88
3 Wszystkie marki	'Sprzedawca'	2947,08
4 Wszystkie marki	'Specjalista marketingu'	2964,74
5 Wszystkie marki	'Prezes'	2958,06
6 Wszystkie marki	'Konsultant'	2969,7
7 'iPhone'	Średnia_cena_umowy	2959,43
8 'iPhone'	'Doradca'	2935,62
9 'iPhone'	'Sprzedawca'	2840,55
10 'iPhone'	'Specjalista marketingu'	2948,56
11 'iPhone'	'Prezes'	3024,38
12 'iPhone'	'Konsultant'	3042,39
13 'Samsung'	Średnia_cena_umowy	2952,29
14 'Samsung'	'Doradca'	2942,8
15 'Samsung'	'Sprzedawca'	2991,81
16 'Samsung'	'Specjalista marketingu'	2934,61
17 'Samsung'	'Prezes'	2882,84

7 Wynik drugiego zapytania - CUBE.

Numer wykonania	Baza danych	Hurtownia danych
1	0,02	0,012
2	0,01	0,006
3	0,008	0,006
Średnia	0,0126	0,008

BAZA DANYCH: Średnia wartość umów w danym województwie.

```
SELECT CASE WHEN wojewodztwo = 'Wszystkie wojewodztwa' THEN 'Wszystkie
wojewodztwa' ELSE (
SELECT nazwa FROM wojewodztwo WHERE id_wojewodztwa = wojewodztwo) END AS
NAZWA_WOJEWODZTWA, rok,srednia_cena_umowy
FROM (
SELECT NVL(TO_CHAR(wojewodztwo.id_wojewodztwa),'Wszystkie
wojewodztwa') AS wojewodztwo,
NVL(TO_CHAR(EXTRACT(YEAR FROM umowa.data)),'Kazdy rok') AS rok,
ROUND(AVG(umowa.cena),2) AS srednia_cena_umowy
FROM umowa, magazyn, salon, ulica, miasta, wojewodztwo, pracownicy
WHERE umowa.magazyn_id_magazyn = magazyn.id_magazyn
AND salon.id_salon = pracownicy.salon_id_salon
AND pracownicy.id_pracownik = umowa.pracownicy_id_pracownik
AND salon.ulica_id_ulica = ulica.id_ulica
AND ulica.miasta_id_miasta = miasta.id_miasta
AND miasta.wojewodztwo_id_wojewodztwa = wojewodztwo.id_wojewodztwa
GROUP BY CUBE (wojewodztwo.id_wojewodztwa,EXTRACT(YEAR FROM
umowa.data))
);
```

HURTOWNIA DANYCH

```
SELECT CASE WHEN wojewodztwo = 'Wszystkie wojewodztwa' THEN 'Wszystkie
wojewodztwa' ELSE (
SELECT nazwa FROM wojewodztwo WHERE id_wojewodztwa = wojewodztwo) END AS
NAZWA_WOJEWODZTWA, rok,srednia_cena_umowy
FROM (
SELECT NVL(TO_CHAR(w_wojewodztwo.id_wojewodztwa),'Wszystkie
wojewodztwa') AS wojewodztwo,
NVL(TO_CHAR(w_rok.rok),'Kazdy rok') AS rok,
ROUND(AVG(w_umowa.cena),2) AS srednia_cena_umowy
FROM w_umowa,w_wojewodztwo,w_rok
WHERE w_umowa.w_wojewodztwo_id_wojewodztwa =
w_wojewodztwo.id_wojewodztwa AND w_rok.id_rok = w_umowa.w_rok_id_rok
GROUP BY CUBE (w_wojewodztwo.id_wojewodztwa, w_rok.rok)
)ORDER BY wojewodztwo, rok;
```

	NAZWA_WOJEWODZTWA	ROK	SREDNIA_CENA_UMOWY
1	Wszystkie wojewodztwa	Kazdy rok	2967,21
2	Wszystkie wojewodztwa	2010	2991,88
3	Wszystkie wojewodztwa	2011	2943,02
4	Wszystkie wojewodztwa	2012	3031,25
5	Wszystkie wojewodztwa	2013	2917,72
6	Wszystkie wojewodztwa	2014	2966,39
7	Wszystkie wojewodztwa	2015	2968,1
8	Wszystkie wojewodztwa	2016	2860,74
9	Wszystkie wojewodztwa	2017	3056,48
10	Wszystkie wojewodztwa	2018	2975,46
11	Wszystkie wojewodztwa	2019	2954,5
12	Wszystkie wojewodztwa	2020	2982,63
13	Wszystkie wojewodztwa	2021	2914,22
14	Wszystkie wojewodztwa	2022	3018,97
15	'Swietokrzyskie'	Kazdy rok	2913,06
16	'Swietokrzyskie'	2010	2922,86
17	'Swietokrzyskie'	2011	2706,63
18	'Swietokrzyskie'	2012	3101,03

8 Wynik trzeciego zapytania - CUBE.

Numer wykonania	Baza danych	Hurtownia danych
1	0,026	0,015
2	0,021	0,013
3	0,014	0,024
Średnia	0,0203	0,0173

Partycje obliczeniowe

BAZA DANYCH: Średnia wydanych pieniędzy w danym salonie.

```
SELECT DISTINCT salon.nazwa AS NAZWA_SALONU,
ulica.nazwa AS ULICA, miasta.nazwa AS MIASTO,
ROUND(AVG(umowa.cena) over (PARTITION BY salon.id_salon),2) AS
SREDNIA_WYDANYCH_PIENIEDZY
FROM salon, umowa, magazyn, ulica, miasta, pracownicy
WHERE salon.id_salon = pracownicy.salon_id_salon
AND pracownicy.id_pracownik = umowa.pracownicy_id_pracownik
AND magazyn.id_magazyn = umowa.magazyn_id_magazyn
AND ulica.id_ulica = salon.ulica_id_ulica
AND ulica.miasta_id_miasta = miasta.id_miasta
ORDER BY srednia_wydanych_pieniedzy DESC;
```

HURTOWNIA DANYCH

```
SELECT DISTINCT w_salon.nazwa AS NAZWA_SALONU,
w_ulica.nazwa AS ULICA, w_miasta.nazwa AS MIASTO,
ROUND(AVG(w_umowa.cena) over (PARTITION BY w_salon.id_salon),2)
AS SREDNIA_WYDANYCH_PIENIEDZY
FROM w_salon, w_umowa, w_ulica, w_miasta
WHERE w_salon.id_salon = w_umowa.w_salon_id_salon
AND w_ulica.id_ulica = w_umowa.w_ulica_id_ulica
AND w_miasta.id_miasta = w_umowa.w_miasta_id_miasta
ORDER BY srednia_wydanych_pieniedzy DESC;
```

	NAZWA_SALONU	ULICA	MIASTO	SREDNIA_WYDANYCH_PIENIEDZY
1	'Play'	'Sloneczna'	'Krakow'	3420,1
2	'T-Mobile'	'Dobra'	'Bialystok'	3237,57
3	'Play'	'Parkowa'	'Zielona Gora'	3192,88
4	'Play'	'Sierpowa'	'Lodz'	3188,85
5	'Plus'	'Przeskok'	'Krakow'	3180,38
6	'Nju-mobile'	'Orzechowa'	'Poznan'	3139,19
7	'Orange'	'Mieszka I'	'Bialystok'	3128,18
8	'Nju-mobile'	'Draski'	'Krakow'	3126,18
9	'Plus'	'Przeskok'	'Bialystok'	3124,3
10	'Orange'	'Barki'	'Warszawa'	3104,92
11	'T-Mobile'	'Barki'	'Lublin'	3098,27
12	'Play'	'Barbary'	'Lublin'	3089,88
13	'Play'	'Graniczna'	'Opole'	3085,59
14	'Orange'	'Chopina'	'Rzeszow'	3084,46
15	'Play'	'Gimnazjalna'	'Opole'	3081,57
16	'Nju-mobile'	'Klikowska'	'Lodz'	3079,08
17	'Plus'	'Poniatowskiego'	'Katowice'	3073,24
18	'T-Mobile'	'Linowa'	'Bialystok'	3067,75

9 Wynik pierwszego zapytania - partycja obliczeniowa.

Numer wykonania	Baza danych	Hurtownia danych
1	0,032	0,023
2	0,021	0,024
3	0,021	0,021
Średnia	0,0246	0,0226

BAZA DANYCH: Średnia wartość umowy w danych wojewodztwach

```

SELECT DISTINCT wojewodztwo.nazwa AS NAZWA_WOJEWODZTWA,
ROUND(AVG(umowa.cena) OVER (PARTITION BY wojewodztwo.id_wojewodztwa),2)
AS SREDNIA_CENA_UMOWY
FROM miasta, ulica, pracownicy, salon, umowa, wojewodztwo
WHERE wojewodztwo.id_wojewodztwa = miasta.wojewodztwo_id_wojewodztwa
AND miasta.id_miasta = ulica.miasta_id_miasta
AND ulica.id_ulica = salon.ulica_id_ulica
AND salon.id_salon = pracownicy.salon_id_salon
AND pracownicy.id_pracownik = umowa.pracownicy_id_pracownik
ORDER BY SREDNIA_CENA_UMOWY DESC;

```

HURTOWNIA DANYCH

```
SELECT DISTINCT w_wojewodztwo.nazwa AS NAZWA_WOJEWODZTWA,
ROUND(AVG(w_umowa.cena) OVER (PARTITION BY w_wojewodztwo.id_wojewodztwa),2)
AS SREDNIA_CENA_UMOWY
FROM w_umowa,w_wojewodztwo
WHERE w_wojewodztwo.id_wojewodztwa = w_umowa.w_wojewodztwo_id_wojewodztwa
ORDER BY SREDNIA_CENA_UMOWY DESC;
```

	NAZWA_WOJEWODZTWA	SREDNIA_CENA_UMOWY
1	'Dolnoslaskie'	3058,02
2	'Mazowieckie'	3030,08
3	'Podkarpacie'	3017,18
4	'Wielkopolskie'	3008,88
5	'Lubelskie'	3001,1
6	'Kujawsko-pomorskie'	2994,41
7	'Zachodniopomorskie'	2984,99
8	'Lubuskie'	2979,32
9	'Lodzkie'	2971,78
10	'Opolskie'	2956,31
11	'Podlaskie'	2955,22
12	'Malopolskie'	2950,97
13	'Warminsko-mazurskie'	2934,35
14	'Slaskie'	2920,89
15	'Swietokrzyskie'	2913,06
16	'Pomorskie'	2876,62

10 Wynik drugiego zapytania - partycja obliczeniowa.

Numer wykonania	Baza danych	Hurtownia danych
1	0,044	0,019
2	0,041	0,016
3	0,017	0,013
Średnia	0,034	0,016

BAZA DANYCH: Sprzedaż danego modelu.

```
SELECT DISTINCT marka.nazwa AS NAZWA_MARKI,model.nazwa AS NAZWA_MODELU,
kolor.nazwa AS KOLOR,
COUNT(umowa.id_umowa) OVER (PARTITION BY telefony.id_telefon)
AS SPRZEDAZ_MODELU
FROM model, telefony,magazyn,umowa,marka, kolor
WHERE marka.id_marka = model.marka_id_marka
AND kolor.id_kolor = telefony.kolor_id_kolor
AND model.id_model = telefony.model_id_model
AND magazyn.telefony_id_telefon = telefony.id_telefon
AND umowa.magazyn_id_magazyn = magazyn.id_magazyn
ORDER BY marka.nazwa, NAZWA_MODELU DESC;
```

HURTOWNIA DANYCH

```
SELECT DISTINCT w_marka.nazwa AS NAZWA_MARKI,w_model.nazwa AS NAZWA_MODELU,
w_kolor.nazwa AS KOLOR,
COUNT(w_umowa.id_umowa) OVER (PARTITION BY w_telefony.id_telefon)
AS SPRZEDAZ_MODELU
FROM w_model, w_telefony,w_umowa,w_marka, w_kolor
WHERE w_model.id_model = w_umowa.w_model_id_model
AND w_telefony.id_telefon = w_umowa.w_telefony_id_telefon
AND w_marka.id_marka = w_umowa.w_marka_id_marka
AND w_kolor.id_kolor = w_umowa.w_kolor_id_kolor
ORDER BY w_marka.nazwa, NAZWA_MODELU DESC;
```

NAZWA_MARKI	NAZWA_MODELU	KOLOR	SPRZEDAZ_MODELU
1 'Huawei'	'P60 Pro'	'Czerwony'	104
2 'Huawei'	'P60'	'Gwiazdzisty'	102
3 'Huawei'	'P50 Pocket'	'Gwiazdzisty'	211
4 'Huawei'	'Nova Y90'	'Czarny'	101
5 'Huawei'	'Nova Y90'	'Teczowy'	229
6 'Huawei'	'Nova Y70'	'Teczowy'	130
7 'Huawei'	'Nova Y70'	'Czerwony'	103
8 'Huawei'	'Nova Y61'	'Czarny'	103
9 'Huawei'	'Nova Y61'	'Fioletowy'	112
10 'Huawei'	'Nova 8i'	'Srebrny'	106
11 'Huawei'	'Nova 8i'	'Teczowy'	122
12 'Huawei'	'Nova 11i'	'Gwiazdzisty'	98
13 'Huawei'	'Nova 11i'	'Czarny'	103
14 'Huawei'	'Nova 11i'	'Czerwony'	110
15 'Huawei'	'Nova 10 SE'	'Gwiazdzisty'	91
16 'Huawei'	'Nova 10 Pro'	'Gwiazdzisty'	98
17 'Huawei'	'Mate X3'	'Gwiazdzisty'	103
18 'Huawei'	'Mate X3'	'Fioletowy'	109

11 Wynik trzeciego zapytania - partycja obliczeniowa.

Numer wykonania	Baza danych	Hurtownia danych
1	0,036	0,02
2	0,017	0,018
3	0,019	0,016
Średnia	0,024	0,018

Okna

BAZA DANYCH: Roczna liczba zawarcia umów w miastach.

```
SELECT DISTINCT miasta.nazwa AS MIASTO,
EXTRACT (YEAR FROM umowa.data) AS ROK,
COUNT(umowa.id_umowa) over (PARTITION BY miasta.id_miasta,
EXTRACT(YEAR FROM umowa.data) ORDER BY EXTRACT(YEAR FROM umowa.data) RANGE
BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS ilosc_umow
FROM umowa, miasta, ulica, salon, pracownicy
WHERE miasta.id_miasta = ulica.miasta_id_miasta
AND ulica.id_ulica = salon.ulica_id_ulica
AND salon.id_salon = pracownicy.salon_id_salon
AND pracownicy.id_pracownik = umowa.pracownicy_id_pracownik
ORDER BY MIASTO, ROK DESC;
```

HURTOWNIA DANYCH

```
SELECT DISTINCT w_miasta.nazwa AS MIASTO,w_rok.rok AS ROK,
COUNT(w_umowa.id_umowa) OVER (PARTITION BY w_miasta.id_miasta, w_rok.rok
ORDER BY w_rok.rok RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW)
AS ilosc_umow
FROM w_umowa, w_miasta, w_rok
WHERE w_miasta.id_miasta = w_umowa.w_miasta_id_miasta
AND w_rok.id_rok = w_umowa.w_rok_id_rok
ORDER BY MIASTO, ROK DESC;
```

	MIASTO	ROK	ILOSC_UMOW
1	'Bialystok'	2022	121
2	'Bialystok'	2021	148
3	'Bialystok'	2020	139
4	'Bialystok'	2019	132
5	'Bialystok'	2018	128
6	'Bialystok'	2017	127
7	'Bialystok'	2016	134
8	'Bialystok'	2015	131
9	'Bialystok'	2014	129
10	'Bialystok'	2013	132
11	'Bialystok'	2012	126
12	'Bialystok'	2011	136
13	'Bialystok'	2010	125
14	'Bydgoszcz'	2022	41
15	'Bydgoszcz'	2021	50
16	'Bydgoszcz'	2020	38
17	'Bydgoszcz'	2019	26
18	'Bydgoszcz'	2018	27

12 Wynik pierwszego zapytania - okna.

Numer wykonania	Baza danych	Hurtownia danych
1	0,034	0,031
2	0,019	0,023
3	0,019	0,015
Średnia	0,024	0,023

BAZA DANYCH: Ilość zawartych umów w danym salonie w każdym roku.

```

SELECT DISTINCT salon.nazwa AS NAZWA_SALONU, salon.id_salon,
EXTRACT(YEAR FROM umowa.data) AS ROK,
COUNT(umowa.id_umowa) OVER (PARTITION BY salon.id_salon, EXTRACT (YEAR FROM
umowa.data) ORDER BY EXTRACT(YEAR FROM umowa.data)
RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS ilosc_umow
FROM pracownicy, salon, umowa
WHERE salon.id_salon = pracownicy.salon_id_salon
AND pracownicy.id_pracownik = umowa.pracownicy_id_pracownik
ORDER BY ROK, ilosc_umow DESC;

```

HURTOWNIA DANYCH

```

SELECT DISTINCT w_salon.nazwa AS NAZWA_SALONU, w_salon.id_salon,
w_rok.rok AS ROK,
COUNT(w_umowa.id_umowa) OVER (PARTITION BY w_salon.id_salon, w_rok.rok
ORDER BY w_rok.rok RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW)
AS ilosc_umow
FROM w_pracownicy, w_salon, w_umowa, w_rok
WHERE w_salon.id_salon = w_umowa.w_salon_id_salon
AND w_rok.id_rok = w_umowa.w_rok_id_rok
AND w_pracownicy.id_pracownik = w_umowa.w_pracownicy_id_pracownik
ORDER BY ROK, ilosc_umow DESC;

```

	NAZWA_SALONU	ID_SALON	ROK	ILOSC_UMOW
1	'Play'	60	2010	23
2	'T-Mobile'	22	2010	21
3	'Play'	24	2010	20
4	'Plus'	12	2010	20
5	'Orange'	82	2010	19
6	'Nju-mobile'	36	2010	17
7	'Orange'	38	2010	17
8	'T-Mobile'	29	2010	17
9	'Nju-mobile'	99	2010	15
10	'Plus'	31	2010	15
11	'T-Mobile'	51	2010	15
12	'Plus'	4	2010	14
13	'Plus'	17	2010	14
14	'Play'	100	2010	13
15	'Orange'	63	2010	12
16	'Play'	39	2010	12
17	'Play'	53	2010	12
18	'Play'	73	2010	12

13 Wynik drugiego zapytania - okna.

Numer wykonania	Baza danych	Hurtownia danych
1	0,076	0,075
2	0,043	0,045
3	0,064	0,043
Średnia	0,061	0,0543

BAZA DANYCH: Miesięczna różnica średniej wartości umowy w danych miastach.

```
SELECT DISTINCT miasta.nazwa AS MIASTO, miesiac, srednia AS SREDNIA_CENA,
ROUND(NVL(srednia-LAG(srednia,1)OVER(PARTITION BY miasta.id_miasta ORDER BY
miesiac DESC),0),2) AS roznica_sredniej
FROM (
    SELECT miasta.id_miasta id_miasto,
    ROUND(AVG(umowa.cena),2) as srednia,
    CASE EXTRACT(MONTH FROM umowa.data) WHEN 1 THEN 'Styczeń' WHEN 2 THEN
    'Luty' WHEN 3 THEN 'Marzec' WHEN 4 THEN 'Kwiecień' WHEN 5 THEN 'Maj'
    WHEN 6 THEN 'Czerwiec' WHEN 7 THEN 'Lipiec' WHEN 8 THEN 'Sierpień'
    WHEN 9 THEN 'Wrzesień' WHEN 10 THEN 'Październik' WHEN 11 THEN
    'Listopad' WHEN 12 THEN 'Grudzień' END AS miesiac FROM miasta, umowa,
    ulica, salon, pracownicy WHERE miasta.id_miasta =
    ulica.miasta_id_miasta
    AND ulica.id_ulica = salon.ulica_id_ulica
    AND salon.id_salon = pracownicy.salon_id_salon
    AND pracownicy.id_pracownik = umowa.pracownicy_id_pracownik
    GROUP BY (miasta.id_miasta,EXTRACT(MONTH FROM umowa.data))
),miasta
WHERE miasta.id_miasta = id_miasto;
```

HURTOWNIA DANYCH

```
SELECT DISTINCT w_miasta.nazwa AS MIASTO, miesiac AS MIESIAC,
srednia AS SREDNIA_CENA,
ROUND(NVL(srednia - LAG(srednia,1) OVER (PARTITION BY w_miasta.id_miasta
ORDER BY miesiac DESC),0),2) AS roznica_sredniej
FROM (
    SELECT w_miasta.id_miasta id_miasto,
    ROUND(AVG(w_umowa.cena),2) AS srednia, w_miesiac.miesiac AS miesiac
    FROM w_miasta, w_miesiac, w_umowa
    WHERE w_miasta.id_miasta = w_umowa.w_miasta_id_miasta
    AND w_miesiac.id_miesiac = w_umowa.w_miesiac_id_miesiac
    GROUP BY (w_miasta.id_miasta, w_miesiac.miesiac)
),w_miasta
WHERE w_miasta.id_miasta = id_miasto;
```

MIASTO	MIESIAC	SREDNIA_CENA	ROZNICA_SREDNIEJ
1 'Ostrowiec Swietokrzyski'	Styczeń	2871,17	0
2 'Ostrowiec Swietokrzyski'	Sierpień	2932,1	60,93
3 'Ostrowiec Swietokrzyski'	Marzec	2719,09	-213,01
4 'Ostrowiec Swietokrzyski'	Maj	3056,44	337,35
5 'Ostrowiec Swietokrzyski'	Luty	2927,87	-128,57
6 'Ostrowiec Swietokrzyski'	Lipiec	2942,11	14,24
7 'Ostrowiec Swietokrzyski'	Kwiecień	2822,15	-119,96
8 'Ostrowiec Swietokrzyski'	Czerwiec	3025,12	202,97
9 'Wroclaw'	Styczeń	2758,73	0
10 'Wroclaw'	Sierpień	3067,99	309,26
11 'Wroclaw'	Marzec	3060,18	-7,81
12 'Wroclaw'	Maj	2947,16	-113,02
13 'Wroclaw'	Luty	3187,93	240,77
14 'Wroclaw'	Lipiec	3354,41	166,48
15 'Wroclaw'	Kwiecień	3083,23	-271,18
16 'Wroclaw'	Czerwiec	2954,72	-128,51
17 'Bydgoszcz'	Styczeń	2969,16	0

14 Wynik trzeciego zapytania - okna.

Numer wykonania	Baza danych	Hurtownia danych
1	0,042	0,023
2	0,032	0,028
3	0,011	0,029
Średnia	0,0283	0,0266

Rankingi

BAZA DANYCH: Ranking ilości zawartych umów przez pracowników.

```
SELECT pracownicy.imie AS IMIE, pracownicy.nazwisko AS NAZWISKO,
salon.nazwa AS SALON, ILOSC_UMOW, RANKING
FROM (
    SELECT pracownicy.id_pracownik id_prac, salon.id_salon id_sal,
    COUNT(umowa.id_umowa) as ILOSC_UMOW,
    DENSE_RANK() OVER (ORDER BY COUNT(umowa.id_umowa) DESC) AS RANKING
    FROM umowa, pracownicy, salon
    WHERE salon.id_salon = pracownicy.salon_id_salon
    AND pracownicy.id_pracownik = umowa.pracownicy_id_pracownik
    GROUP BY (pracownicy.id_pracownik, salon.id_salon)
), pracownicy, salon
WHERE salon.id_salon = id_sal
AND pracownicy.id_pracownik = id_prac;
```

HURTOWNIA DANYCH

```
SELECT w_pracownicy.imie AS IMIE, w_pracownicy.nazwisko AS NAZWISKO,
w_salon.nazwa AS SALON, ILOSC_UMOW, RANKING
FROM (
    SELECT w_pracownicy.id_pracownik id_prac, w_salon.id_salon id_sal,
    COUNT(w_umowa.id_umowa) as ILOSC_UMOW,
    DENSE_RANK() OVER (ORDER BY COUNT(w_umowa.id_umowa) DESC) AS RANKING
    FROM w_umowa, w_pracownicy, w_salon
    WHERE w_pracownicy.id_pracownik = w_umowa.w_pracownicy_id_pracownik
    AND w_salon.id_salon = w_umowa.w_salon_id_salon
    GROUP BY (w_pracownicy.id_pracownik, w_salon.id_salon)
), w_pracownicy, w_salon
WHERE w_salon.id_salon = id_sal
AND w_pracownicy.id_pracownik = id_prac;
```

IMIE	NAZWISKO	SALON	ILOSC_UMOW	RANKING
1 'Pawel'	'Kowalczyk'	'Plus'	41	1
2 'Jaroslaw'	'Pazurek'	'Nju-mobile'	37	2
3 'Filip'	'Nowak'	'T-Mobile'	35	3
4 'Jaroslaw'	'Krawczyk'	'T-Mobile'	35	3
5 'Joachim'	'Wojcik'	'Play'	34	4
6 'Konrad'	'Adamczyk'	'Play'	34	4
7 'Grzegorz'	'Zalecki'	'Orange'	34	4
8 'Pawel'	'Adamczyk'	'Play'	34	4
9 'Bartosz'	'Dudek'	'Play'	34	4
10 'Karol'	'Kaczmarek'	'Plus'	33	5
11 'Lukasz'	'Kaczmarek'	'Nju-mobile'	33	5
12 'Joachim'	'Zalecki'	'Orange'	33	5
13 'Pawel'	'Kaczmarek'	'Plus'	32	6
14 'Karol'	'Pazurek'	'Play'	32	6
15 'Filip'	'Adamczyk'	'Play'	32	6
16 'Bartosz'	'Kaczmarek'	'Play'	32	6
17 'Bartosz'	'Zalecki'	'T-Mobile'	32	6
18 'Michal'	'Krawczyk'	'Plus'	31	7

15 Wynik pierwszego zapytania - rankingi.

Numer wykonania	Baza danych	Hurtownia danych
1	0,042	0,022
2	0,034	0,033
3	0,031	0,023
Średnia	0,0356	0,026

BAZA DANYCH: Ranking ilości umów w danych salonach.

```

SELECT salon.nazwa AS NAZWA_SALONU, miasta.nazwa AS MIASTO,
ulica.nazwa AS ULICA, ILOSC_UMOW, RANKING
FROM (
    SELECT salon.id_salon id_sal, miasta.id_miasta id_miasto,
    ulica.id_ulica id_ul,
    COUNT(umowa.id_umowa) as ILOSC_UMOW,
    DENSE_RANK() OVER (ORDER BY COUNT(umowa.id_umowa) DESC) AS RANKING
    FROM salon, miasta, umowa, ulica, magazyn, pracownicy
    WHERE miasta.id_miasta = ulica.miasta_id_miasta
    AND ulica.id_ulica = salon.ulica_id_ulica
    AND salon.id_salon = pracownicy.salon_id_salon
    AND pracownicy.id_pracownik = umowa.pracownicy_id_pracownik
    AND magazyn.id_magazyn = umowa.magazyn_id_magazyn
    GROUP BY (salon.id_salon, miasta.id_miasta, ulica.id_ulica)
),salon, miasta,ulica
WHERE salon.id_salon = id_sal
AND miasta.id_miasta = id_miasto
AND ulica.id_ulica = id_ul;

```

HURTOWNIA DANYCH

```

SELECT w_salon.nazwa AS NAZWA_SALONU, w_miasta.nazwa AS MIASTO,
w_ulica.nazwa, ILOSC, RANKING
FROM (
    SELECT w_salon.id_salon id_sal, w_miasta.id_miasta
    id_miasto, w_ulica.id_ulica id_ul,
    COUNT(w_umowa.id_umowa) AS ILOSC,
    DENSE_RANK() OVER(ORDER BY COUNT(w_umowa.id_umowa) DESC) AS RANKING
    FROM w_salon, w_umowa, w_miasta, w_ulica
    WHERE w_salon.id_salon = w_umowa.w_salon_id_salon
    AND w_miasta.id_miasta = w_umowa.w_miasta_id_miasta
    AND w_ulica.id_ulica = w_umowa.w_ulica_id_ulica
    GROUP BY (w_salon.id_salon, w_miasta.id_miasta, w_ulica.id_ulica)
), w_salon, w_miasta, w_ulica
WHERE w_salon.id_salon = id_sal
AND w_miasta.id_miasta = id_miasto
AND w_ulica.id_ulica = id_ul;

```

	NAZWA_SALONU	MIASTO	ULICA	ILOSC_UMOW	RANKING
1	'Play'	'Wroclaw'	'Szkolna'	315	1
2	'Plus'	'Krakow'	'Sloneczna'	244	2
3	'Play'	'Rzeszow'	'Chopina'	207	3
4	'T-Mobile'	'Bydgoszcz'	'Obszar Kolei'	194	4
5	'Plus'	'Lublin'	'Barki'	194	4
6	'Plus'	'Bialystok'	'Przeskok'	193	5
7	'Play'	'Opole'	'Rieczna'	187	6
8	'Nju-mobile'	'Katowice'	'Rynek'	178	7
9	'Play'	'Opole'	'Ciosa'	178	7
10	'Play'	'Gdansk'	'Stawowa'	175	8
11	'Orange'	'Bialystok'	'Podrygu'	172	9
12	'T-Mobile'	'Bialystok'	'Liniowa'	170	10
13	'Play'	'Krakow'	'Mariacka'	169	11
14	'Play'	'Poznan'	'Jasna'	164	12
15	'Orange'	'Rzeszow'	'Chopina'	163	13
16	'Nju-mobile'	'Bialystok'	'Liniowa'	160	14
17	'Nju-mobile'	'Opole'	'Klonowa'	159	15
18	'Nju-mobile'	'Krakow'	'Konia'	157	16

16 Wynik drugiego zapytania - rankingi.

Numer wykonania	Baza danych	Hurtownia danych
1	0,019	0,013
2	0,023	0,012
3	0,022	0,018
Średnia	0,0213	0,0143

BAZA DANYCH: Ranking zawartych umów przez klientów.

```
SELECT klient.imie AS IMIE, klient.nazwisko AS NAZWISKO, ILOSC_UMOW,
RANKING
FROM (
    SELECT klient.id_klient id_kl,
    COUNT(umowa.id_umowa) AS ILOSC_UMOW,
    DENSE_RANK() OVER(ORDER BY COUNT(umowa.id_umowa)DESC) AS RANKING
    FROM klient, umowa
    WHERE umowa.klient_id_klient = klient.id_klient
    GROUP BY (klient.id_klient)
),klient
WHERE klient.id_klient = id_kl;
```

HURTOWNIA DANYCH

```
SELECT w_klient.imie AS IMIE, w_klient.nazwisko AS NAZWISKO,ILOSC_UMOW,
RANKING
FROM (
    SELECT w_klient.id_klient id_kl,
    COUNT(w_umowa.id_umowa) AS ILOSC_UMOW,
    DENSE_RANK() OVER(ORDER BY COUNT(w_umowa.id_umowa)DESC) AS RANKING
    FROM w_umowa, w_klient
    WHERE w_umowa.w_klient_id_klient = w_klient.id_klient
    GROUP BY w_klient.id_klient
),w_klient
WHERE w_klient.id_klient = id_kl;
```

IMIE	NAZWISKO	ILOSC_UMOW	RANKING
1 'Karol'	'Labryga'	48	1
2 'Filip'	'Wozniak'	34	2
3 'Jaroslaw'	'Labryga'	33	3
4 'Konrad'	'Zalecki'	33	3
5 'Denis'	'Pazurek'	32	4
6 'Lukasz'	'Mazur'	32	4
7 'Michal'	'Zalecki'	31	5
8 'Michal'	'Mazur'	31	5
9 'Piotr'	'Nowak'	31	5
10 'Piotr'	'Wozniak'	31	5
11 'Bartosz'	'Labryga'	31	5
12 'Denis'	'Banabak'	31	5
13 'Grzegorz'	'Adamczyk'	30	6
14 'Lukasz'	'Nowak'	30	6
15 'Denis'	'Adamczyk'	30	6
16 'Mateusz'	'Labryga'	30	6
17 'Joachim'	'Pazurek'	30	6

17 Wynik trzeciego zapytania - rankingi.

Numer wykonania	Baza danych	Hurtownia danych
1	0,023	0,02
2	0,012	0,013
3	0,013	0,012
Średnia	0,016	0,015

8. Uwagi i wnioski.

Podczas tworzenia projektu zysaliśmy nową wiedzę na temat projektowania baz danych oraz hurtowni danych. Dowiedzieliśmy się, że w przypadku, kiedy baza danych ma być przetransformowana na hurtownię danych nie należy używać w zapytaniach złączeń JOIN, ponieważ przy ogromnych ilościach danych w hurtowni czas wykonania takich zapytań byłby bardzo duży. Dowiedzieliśmy się również, że w celu spójności danych w bazie należy tworzyć słowniki dla obiektów, które mogą być słownikowane.

Nasza baza danych została zasilona danymi z plików z rozszerzeniem .csv, do których dane wygenerowaliśmy za pomocą programu napisanego w języku c++. Do każdego takiego pliku został utworzony plik kontrolny z rozszerzeniem .ctl, w którym zdefiniowane są polecenia, które dodają dane z pliku csv do danej tabeli w bazie danych. Samo ładowanie danych przeprowadziliśmy za pomocą SQLloader. Wszystkie polecenia ładujące dane do bazy poprzez plik kontrolny zostały umieszczone w pliku wsadowym .bat.

Następnie napisaliśmy zapytania do bazy danych, po czym zaprojektowaliśmy schemat hurtowni danych w układzie gwieżdztym. Kolejnym krokiem była transformacja danych z bazy danych do hurtowni. Zrealizowaliśmy ten etap tworząc procedury, które wybierają dane pola z tabel znajdujących się w bazie danych i wstawiają je do tabel znajdujących się w hurtowni.

Na koniec napisaliśmy zapytania do hurtowni danych oraz porównaliśmy czasy wykonania danych zapytań pomiędzy bazą danych a hurtownią. Z porównań wynikło to, że zapytania dla hurtowni danych wykonują się szybciej, jest to spowodowane tym, że hurtownia ma układ gwiazdy i wykonuje mniejszą ilość porównań.