

SPRAWOZDANIE
PROJEKT ZALICZENIOWY
BAZY DANYCH

Bartosz Gawryszewski

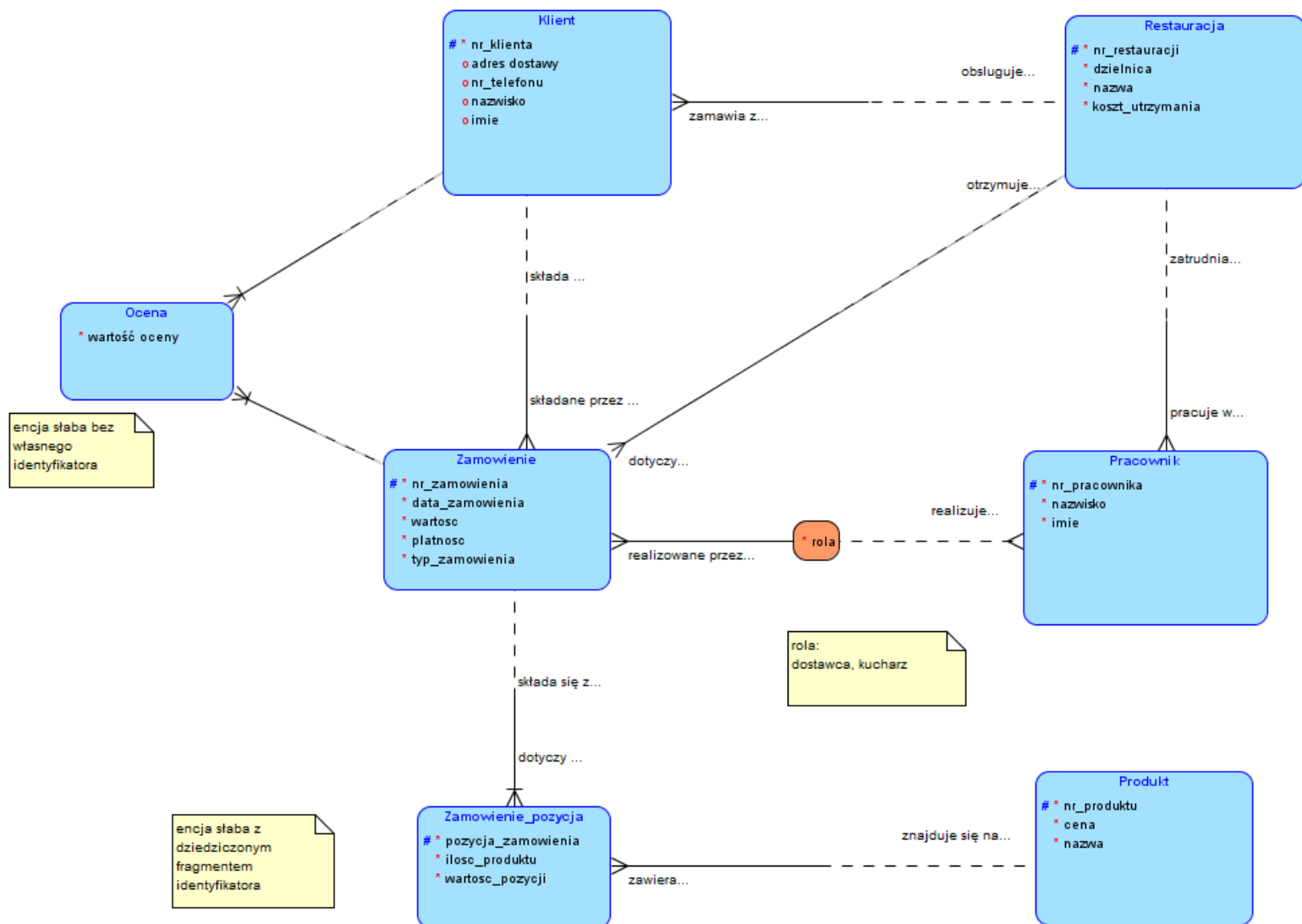
WCY20IY2S1

Prowadzący: mgr inż. Józef Woźniak

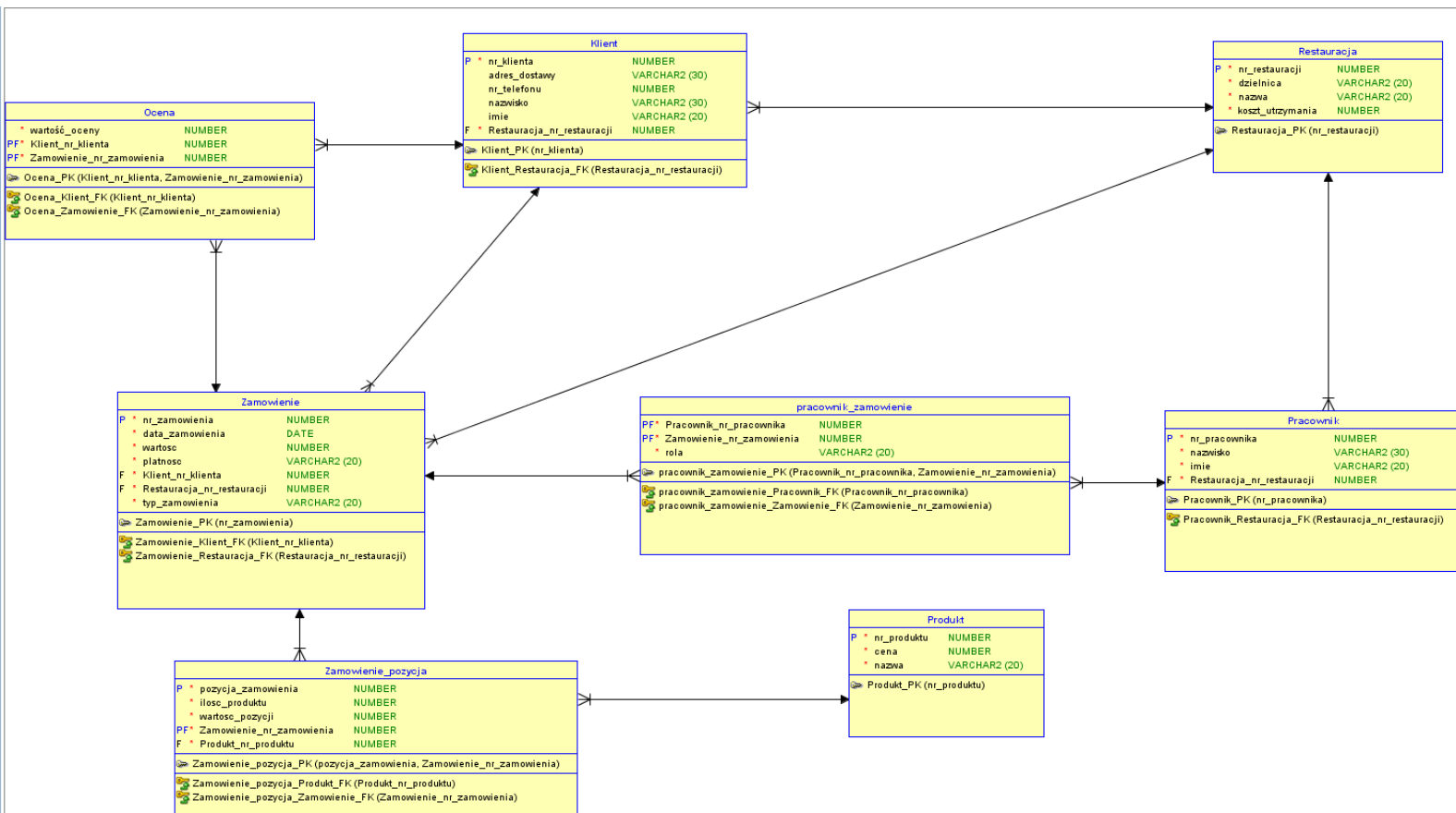
6.1 Opis projektowanej rzeczywistości

Model przedstawia potencjalne działanie sieci restauracji oferującej dostawę do klienta. Umożliwia on śledzenie przychodów w ciągu jednego dnia, ocenianie składanych zamówień przez klientów, monitorowanie ilości zrealizowanych zamówień przez każdego pracownika, oraz ilości sprzedaży każdego produktu z menu. Klient może złożyć zamówienie w samej restauracji, wtedy nie ma informacji na temat jego adresu, numeru telefonu, oraz nazwiska i imienia, lub też w dostawie, w której podaje wymienione wcześniej dane. Każdy klient rozróżniany jest dzięki unikatowemu numerowi klienta. Przy zamówieniu pracują 1 lub 2 osoby (na miejscu – sam kucharz, w dostawie – kucharz i dostawca). Każdy pracownik posiada przypisaną do swojego zamówienia rolę. Restauracja posiada określony dzienny koszt utrzymania. Za zamówienie można zapłacić kartą lub też gotówką. Każdy z klientów może ocenić złożone przez siebie zamówienie.

6. 2 Model logiczny:



Model relacyjny



6.3 Oprogramowanie tworzące bazę danych

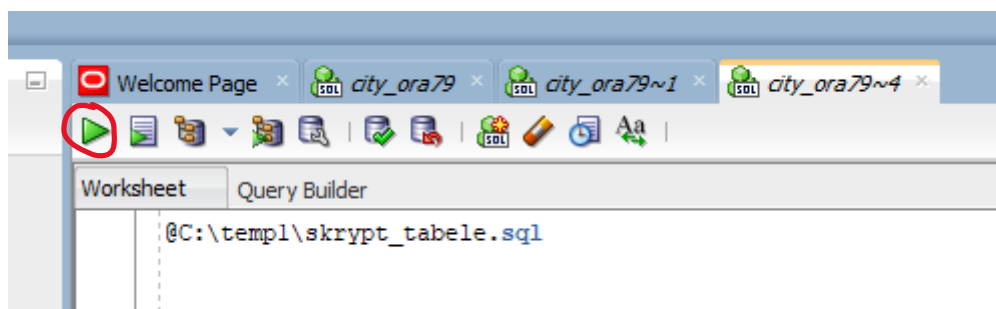
Baza danych zaprojektowana została przy pomocy oprogramowania Oracle SQL Developer Data Modeler. Sama baza umiejscowiona została na serwerze city, wykorzystywanym przez nas na laboratoriach. Wszelkie operacje na bazie wykonywane są w języku SQL przy pomocy oprogramowania Oracle SQL Developer.

6.4 Skrypty wdrożeniowe instalujące i deinstalujące zrealizowany projekt

Skryptem instalującym tabele oraz wprowadzającym do nich przygotowane przykładowe dane jest umieszczony w zadaniu plik **skrypt_tabele.sql**, przygotowane perspektywy – **perspektywy.sql**. Całość projektu deinstalować można za pomocą skryptu – **deinstalacja.sql**.

6.5 Instrukcja instalacji projektu i sprawdzenia jego poprawności

Aby zainstalować projekt na wybranym serwerze należy najpierw się z nim połączyć, następnie na swojej sesji podać lokalizację skryptu instalującego (np. @C:\temp\skrypt_tabele.sql).



Następnie nacisnąć trzeba przycisk **Run Statement** (zaznaczony na powyższym rysunku).

Wynik działania:

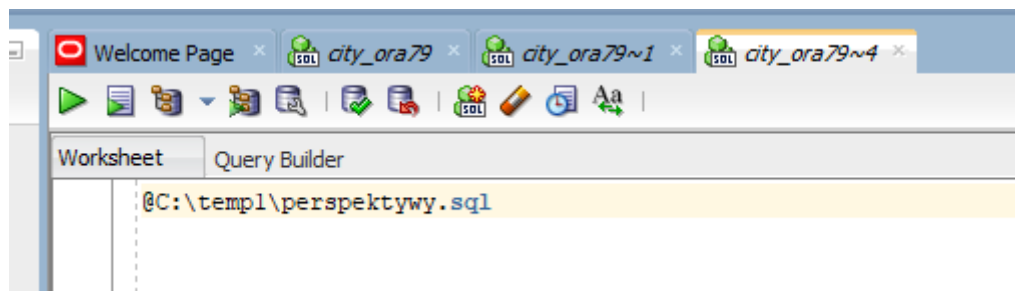
Task completed in 6,683 seconds

```
Table RESTAURACJA created.  
  
Table KLIENT created.  
  
Table PRACOWNIK created.  
  
Table PRODUKT created.  
  
Table ZAMOWIENIE created.  
  
Table ZAMOWIENIE_POZYCJA created.  
  
Table OCENA created.  
  
Table PRACOWNIK_ZAMOWIENIE created.
```

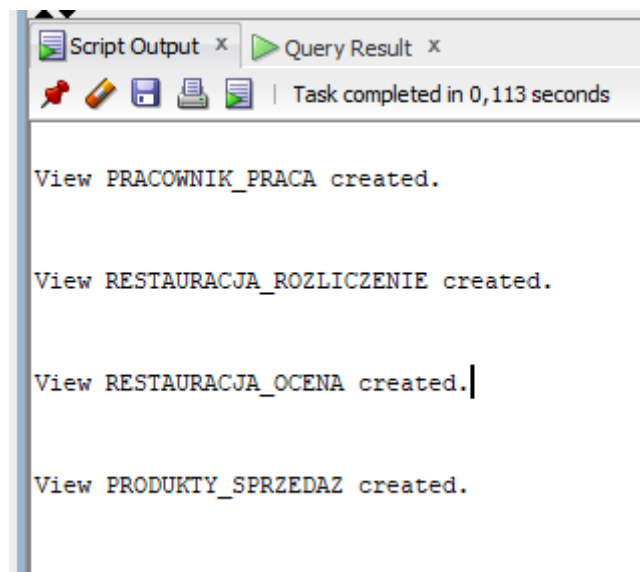
Oraz duża liczba powiadomień o dodaniu danych do tabel:

[illegible]

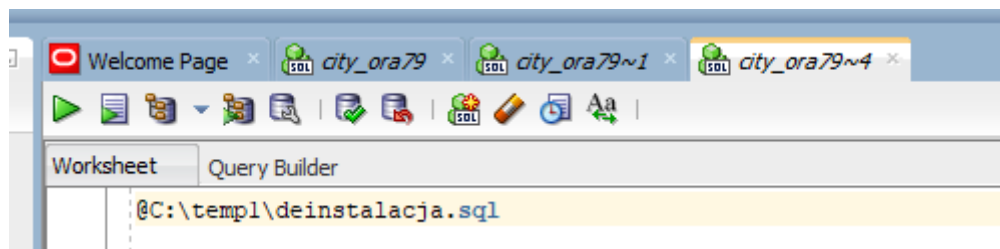
Podobnie z perspektywami:



Wynik działania:



Oraz deinstalacją:



Wynik działania:

```
Table KLIENT dropped.  
  
Table OCENA dropped.  
  
Table PRACOWNIK dropped.  
  
Table PRACOWNIK_ZAMOWIENIE dropped.  
  
Table PRODUKT dropped.  
  
Table RESTAURACJA dropped.  
  
Table ZAMOWIENIE dropped.  
  
Table ZAMOWIENIE_POZYCJA dropped.  
  
View PRACOWNIK_PRACA dropped.  
  
View PRODUKTY_SPRZEDAZ dropped.  
  
View RESTAURACJA_OCENA dropped.  
  
View RESTAURACJA_ROZLICZENIE dropped.
```


Opis przygotowanych perspektyw

a) restauracja_rozliczenie

Perspektywa obrót dzienny dla każdej restauracji (suma wartości jej zamówień minus dzienny koszt utrzymania) oraz ilość klientów.

```
--rozliczenie na koniec dnia
create or replace view restauracja_rozliczenie as
select nazwa, (sum(wartosc) - koszt_utrzymania) as Obrot_dzienny, count(k.nr_klienta) Klienci
from zamowienie z
join restauracja r on z.nr_restauracji = r.nr_restauracji
join klient k on z.nr_klienta = k.nr_klienta
group by nazwa, r.koszt_utrzymania;
```

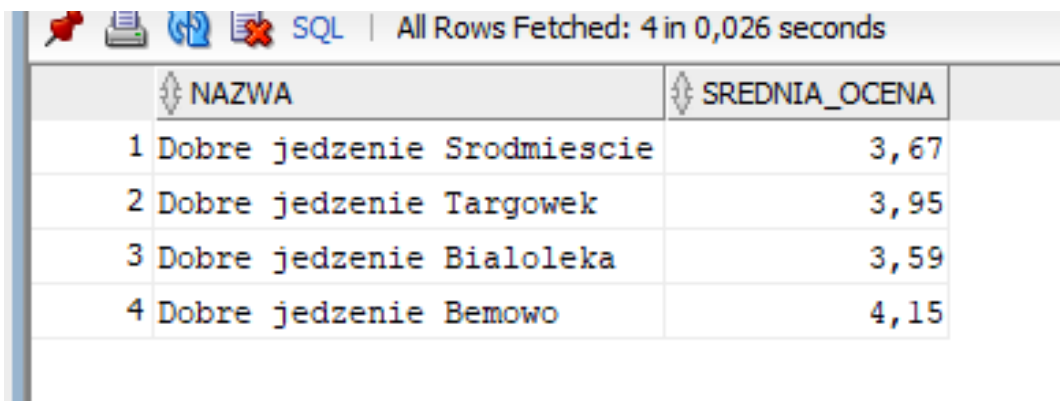
	NAZWA	OBROT_DZIENNY	KLIENCI
1	Dobre jedzenie Srodmiescie	329	21
2	Dobre jedzenie Bialoleka	367	17
3	Dobre jedzenie Targowek	732	19
4	Dobre jedzenie Bemowo	203	13

b) restauracja_ocena

Perspektywa przedstawia średnią ocenę zamówień jednego dnia

```
--srednia ocena na dzien
```

```
create or replace view restauracja_ocena as
select nazwa, round(avg(wartosc_oceny), 2) Srednia_ocena
from restauracja r
join zamowienie z on r.nr_restauracji = z.nr_restauracji
join ocena o on z.nr_zamowienia = o.nr_zamowienia
group by nazwa
;
```



SQL | All Rows Fetched: 4 in 0,026 seconds

	NAZWA	SREDNIA_OCENA
1	Dobre jedzenie Srodmiescie	3,67
2	Dobre jedzenie Targowek	3,95
3	Dobre jedzenie Bialoleka	3,59
4	Dobre jedzenie Bemowo	4,15

c) pracownik_praca

Perspektywa przedstawia ilość zrealizowanych zamówień przez pracownika oraz ich średnią wartość oraz ocenę.

```
create or replace view pracownik_praca as
select p.nr_restauracji, p.nr_pracownika, nazwisko, imie, count(pz.nr_zamowienia) Ilosc_zamowien,
round(avg(wartosc), 2) Srednia_wartosc_zamowienia, round(avg(wartosc_oceny), 2) Srednia_ocena
from pracownik p
join pracownik_zamowienie pz on p.nr_pracownika = pz.nr_pracownika
join zamowienie z on pz.nr_zamowienia = z.nr_zamowienia
join ocena o on z.nr_zamowienia = o.nr_zamowienia
group by p.nr_restauracji, p.nr_pracownika, nazwisko, imie
;
```

	NR_RESTAURACJI	NR_PRACOWNIKA	NAZWISKO	IMIE	ILOSC_ZAMOWIEN	SREDNIA_WARTOSC_ZAMOWIENIA	SREDNIA_OCENA
1	3	9	Szybki	Szymon	6	51	4,17
2	2	6	Zły	Zygmunt	11	50,09	3,82
3	3	3	Pracowity	Przemysław	7	47,29	4,57
4	3	13	Lukasz	Stefan	6	45,5	3,67
5	2	5	Leniwy	Larry	5	46,6	3,6
6	1	11	Szczesny	Wojciech	9	48,56	3,89
7	2	1	Koziol	Paweł	10	45,8	4
8	4	14	Czopur	Patryk	8	38,13	3,63
9	1	7	Pomocna	Paulina	7	57,29	4
10	4	4	Niezdarny	Natan	9	36,78	3,11
11	1	8	Poprawny	Piotr	3	69	4
12	1	2	Koziol	Piotr	11	59,73	4
13	4	10	Wolny	Wojciech	7	47,86	2,86
14	2	12	Wysocki	Maksymilian	4	59,25	4,5

d) produkty_sprzedaz

Perspektywa przedstawia sprzedaż każdego produktu z menu

```
create or replace view produkty_sprzedaz as
select nazwa, sum(ilosc_produktu) Sprzedana_ilosc
from produkt p
join zamowienie_pozycja zp on p.nr_produktu = zp.nr_produktu
group by nazwa;
```

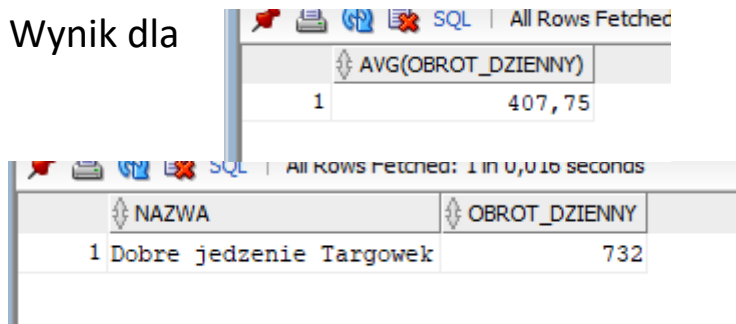
	NAZWA	SPRZEDANA_ILOSC
1	Kotlet schabowy	23
2	Coca-Cola	10
3	Zupa pomidorowa	20
4	Herbata	21
5	Pierogi z kapusta	17
6	Kluski leniwe	25
7	Ciasto czekoladowe	12
8	Zupa grzybowa	13
9	Woda	8
10	Rosol	26
11	Pierogi z pieczarkami	8
12	Kotlet mielony	13
13	Zupa ogorkowa	10
14	Pierogi ruskie	16
15	Sernik	20
16	Kurczak panierowany	31

Przykładowe operacje na modelu:

1.

```
--restauracje posiadajace wiekszy obrot dzienny niz srednia obrotow dziennych wszystkich restauracji  
select nazwa, obrot_dzienny from restauracja_rozliczenie  
where obrot_dzienny > (select avg(obrot_dzienny) from restauracja_rozliczenie);
```

Wynik dla



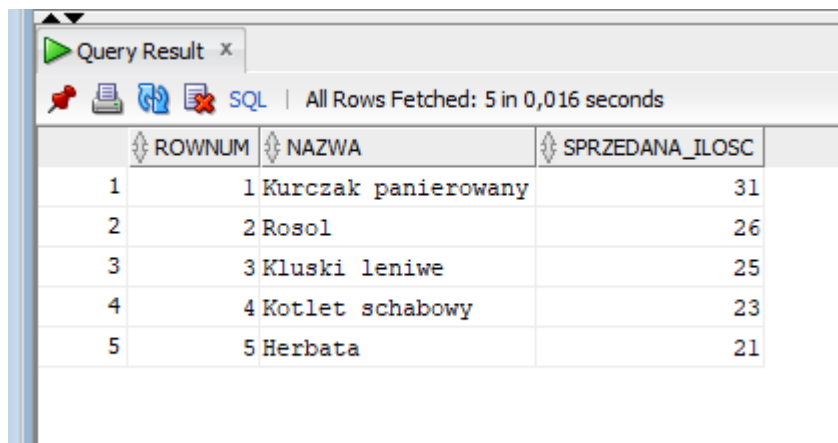
The screenshot shows two SQL query results. The first result is a single row with the average daily turnover (AVG(OBROT_DZIENNY)) of 407,75. The second result is a table with two columns: NAZWA (Name) and OBROT_DZIENNY (Daily Turnover). It lists one restaurant: 'Dobre jedzenie Targowek' with a turnover of 732.

AVG(OBROT_DZIENNY)
407,75

NAZWA	OBROT_DZIENNY
1 Dobre jedzenie Targowek	732

2.

```
--5 najczesciej zamawianych produktow  
select rownum, nazwa, sprzedana_ilosc  
from  
(select nazwa, sprzedana_ilosc from produkty_sprzedaz order by sprzedana_ilosc desc)  
where rownum <=5;
```



The screenshot shows a SQL query result with 5 rows. The columns are ROWNUM, NAZWA (Name), and SPRZEDANA_ILOSC (Sold Quantity). The products are listed in descending order of quantity sold.

ROWNUM	NAZWA	SPRZEDANA_ILOSC
1	1 Kurczak panierowany	31
2	2 Rosol	26
3	3 Kluski leniwe	25
4	4 Kotlet schabowy	23
5	5 Herbata	21

3.

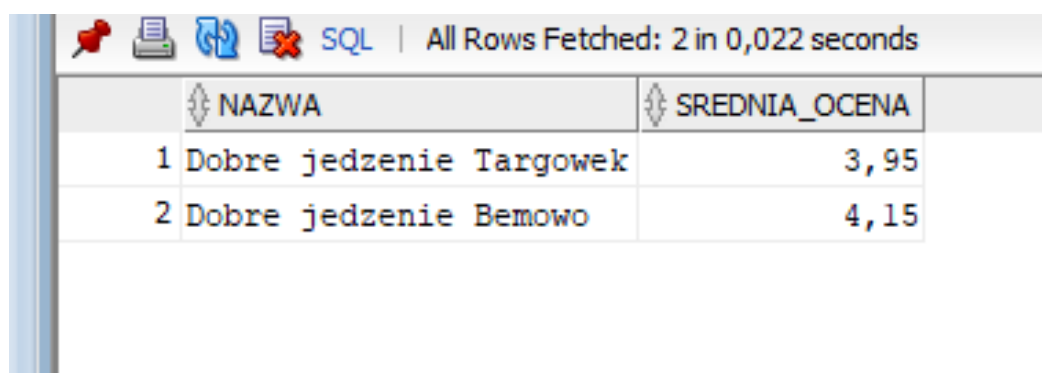
```
--pracownicy ktorzy wykonali wieksza ilosc zamowien niz srednia ilosc zamowien wykonana przez pracownika w ich restauracji
select nr_pracownika, nazwisko, imie, ilosc_zamowien
from pracownik_praca p
where ilosc_zamowien > (select avg(ilosc_zamowien) from pracownik_praca x where x.nr_restauracji = p.nr_restauracji)
order by nr_pracownika
;
```

SQL | All Rows Fetched: 6 in 0,045 seconds

	NR_PRACOWNIKA	NAZWISKO	IMIE	ILOSC_ZAMOWIEN
1	1	Kozioł	Paweł	10
2	2	Kozioł	Piotr	11
3	3	Pracowity	Przemysław	7
4	4	Niezdarny	Natan	9
5	6	Zły	Zygmunt	11
6	11	Szczesny	Wojciech	9

4.

```
--restauracje majace srednia ocene wieksza niz srednia ocen wszystkich restauracji  
select nazwa, round(avg(wartosc_oceny), 2) Srednia_ocena  
from restauracja r  
join zamowienie z on r.nr_restauracji = z.nr_restauracji  
join ocena o on z.nr_zamowienia = o.nr_zamowienia  
group by nazwa  
having avg(wartosc_oceny) > (select avg(srednia_ocena) from restauracja_ocena);
```



	NAZWA	SREDNIA_OCENA
1	Dobre jedzenie Targowek	3,95
2	Dobre jedzenie Bemowo	4,15