Lab report #2 Sadovskaya Veronika

GitHub: https://github.com/sdveronika/DataMola22

Task 1 - CREATE Daily Reports Layouts

Day	Profit	Day	Orders	Dish	Amount
01.01.2022	10255	01.01.2022	150	Pasta	120
02.01.2022	12000	02.01.2022	167	Pizza	100
03.01.2022	7375	03.01.2022	118	Chebupelli	136
04.01.2022	13055	04.01.2022	179	Soup	51
05.01.2022	10955	05.01.2022	186	Greek salad	200
06.01.2022	5255	06.01.2022	95	Total dishes per day	607
07.01.2022	11155	07.01.2022	163		
08.01.2022	9055	08.01.2022	134		
Total profit per month		Total orders per month			
		*not delivery			

Category	Amount dishes	Date	Average check of the order
Hot	5000	01.01.2022	36
Appetizer	359	02.01.2022	50
Total dishes per day	5359	03.01.2022	26
		04.01.2022	69
		05.01.2022	45
		06.01.2022	25
		07.01.2022	51
		08.01.2022	49
		Average check for month	43,875

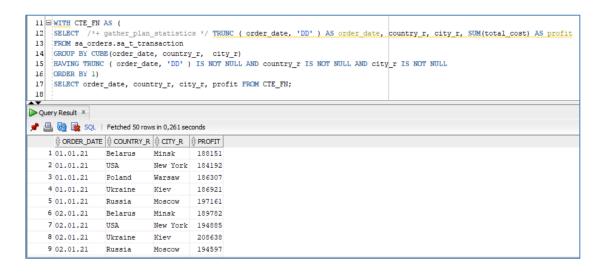
Task 2 - CREATE Monthly Reports Layouts

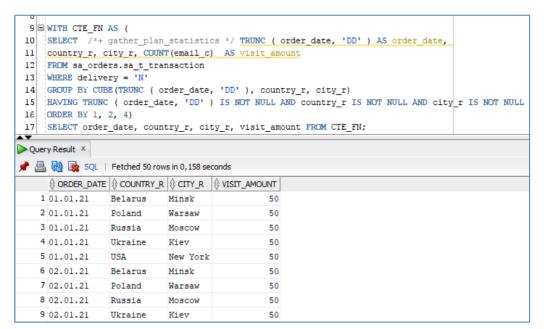
Day	Profit	Day	Orders	Dish	Amount
01.01.2022	1025500	01.01.2022	15000	Pasta	12000
01.02.2022	1200000	01.02.2022	16700	Pizza	10000
01.03.2022	737500	01.03.2022	11800	Chebupelli	13600
01.04.2022	1305500	01.04.2022	17900	Soup	5100
01.05.2022	1095500	01.05.2022	18600	Greek salad	20000
01.06.2022	525500	01.06.2022	9500	Total dishes per month	60700
01.07.2022	1115500	01.07.2022	16300		
01.08.2022	905500	01.08.2022	13400		
Total profit per year		Total orders per year			
		*not delivery			

Category	Amount dishes	Date	Average check of the order
Hot	500000	01.01.2022	3600
Appetizer	35900	01.02.2022	5000
Total dishes per month	535900	01.03.2022	2600
		01.04.2022	6900
		01.05.2022	4500
		01.06.2022	2500
		01.07.2022	5100
		01.08.2022	4900
		Average check for year	4387,5

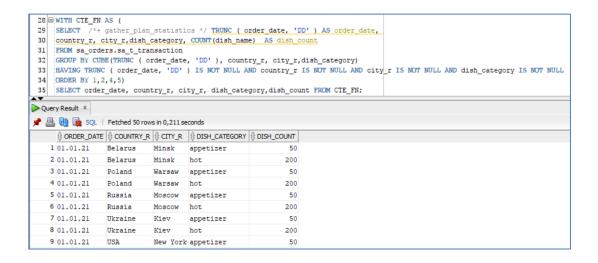
Note: tasks 1 and 2 present the simplest examples of results, to which you can add grouping by country, region, city, delivery, or order payment method.

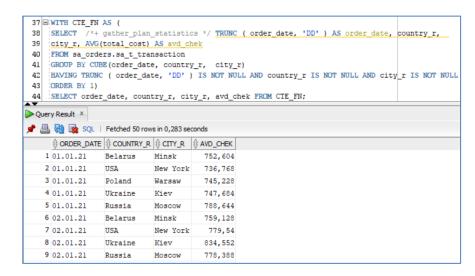
Task 3 - CREATE Test AdHoc SQL - Daily Reports (CUBE)



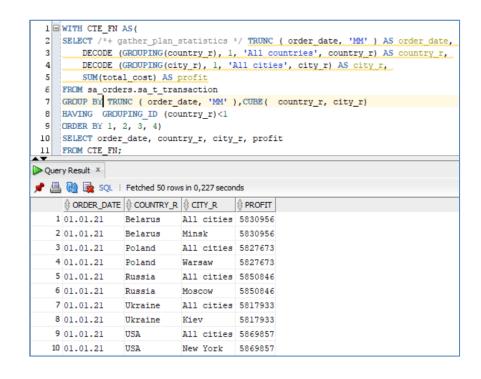


16 🖃	WITH CTE FN	λς /							
	_		etatieti	CS */ TDIINC	/ order date	IDDI) AS order date			
	SELECT /*+ gather_plan_statistics */ TRUNC (order_date, 'DD') AS order_date, country r, city r, dish name, COUNT(dish name) AS dish count FROM sa orders.sa t transaction								
	GROUP BY CUBE(TRUNC (order_date, 'DD'), country_r, city_r, dish_name) HAVING TRUNC (order_date, 'DD') IS NOT NULL AND country r IS NOT NULL AND city r IS NOT NULL AND dish name IS NOT NULL								
	ORDER BY 1, 2, 4, 5)								
_	ry Result ×								
📌 🖺	🙀 🅦 SQL	Fetched 50 row	s in 0,291 se	econds					
	♦ ORDER_DATE								
1	01.01.21	Belarus	Minsk	chebupelli	50				
2	01.01.21	Belarus	Minsk	greek salad	50				
3	01.01.21	Belarus	Minsk	pasta	50				
4	01.01.21	Belarus	Minsk	pizza	50				
5	01.01.21	Belarus	Minsk	soup	50				
6	01.01.21	Poland	Warsaw	chebupelli	50				
7	01.01.21	Poland	Warsaw	greek salad	50				
8	01.01.21	Poland	Warsaw	pasta	50				
	01.01.21	Poland	Warsaw	pizza	50				

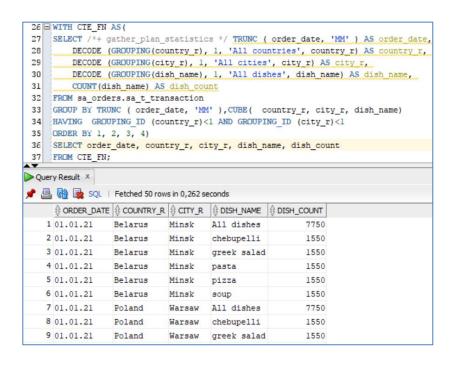


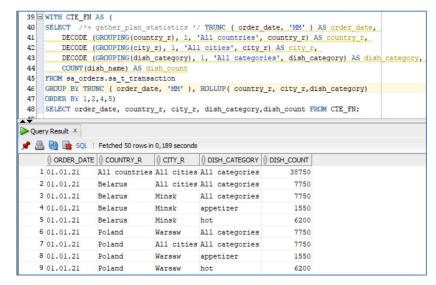


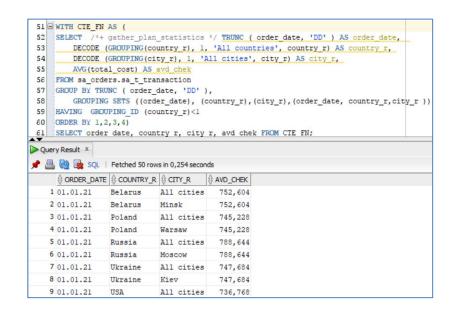
Task 4 - CREATE Test AdHoc SQL - Monthly Reports (ROLLUP & GROUPING SETS)



```
13 WITH CTE FN AS (
 14 SELECT /*+ gather_plan_statistics */ TRUNC ( order_date, 'MM' ) AS order_date,
 15
     DECODE (GROUPING(country r), 1, 'All countries', country r) AS country r,
 16
        DECODE (GROUPING(city r), 1, 'All cities', city r) AS city r,
 17
        COUNT (email c) AS visit amount
 18
    FROM sa_orders.sa_t_transaction
 19
    WHERE delivery = 'N'
 20
    GROUP BY TRUNC ( order_date, 'MM' ), CUBE( country_r, city_r)
 21
     HAVING GROUPING_ID (country_r)<1
 22
    ORDER BY 1, 2, 3, 4)
    SELECT order_date, country_r, city_r, visit_amount
24 FROM CTE FN;
Query Result X
📌 🚇 🙀 🗽 SQL | Fetched 50 rows in 0,135 seconds
    ♦ ORDER_DATE ♦ COUNTRY_R ♦ CITY_R ♦ VISIT_AMOUNT
   1 01.01.21 Belarus All cities 1550
   2 01.01.21
                Belarus
                          Minsk
   3 01.01.21
                Poland
                          All cities
                                     1550
                                            1550
    4 01.01.21
                Poland
                          Warsaw
   5 01.01.21 Russia All cities
                                          1550
   6 01.01.21 Russia
                          Moscow
   7 01.01.21
                Ukraine
                          All cities
                Ukraine
   8 01.01.21
                          Kiev
                                            1550
                         All cities 1550
   9 01.01.21 USA
```







Task 5 - CREATE Test AdHoc SQL - ROLLUP by Time

