**Lab report #6**

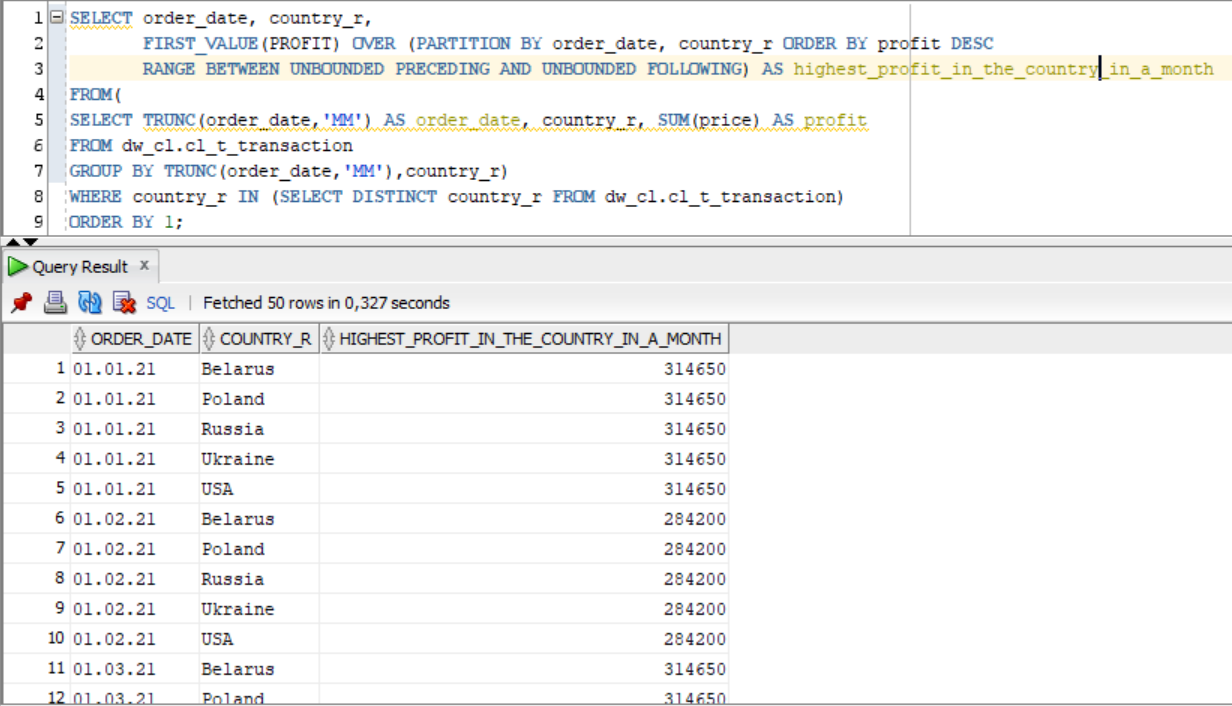
**Sadovskaya Veronika**

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

**Task 1 - Create Ad Hoc SQL FIRST\_VALUE, LAST\_VALUE**

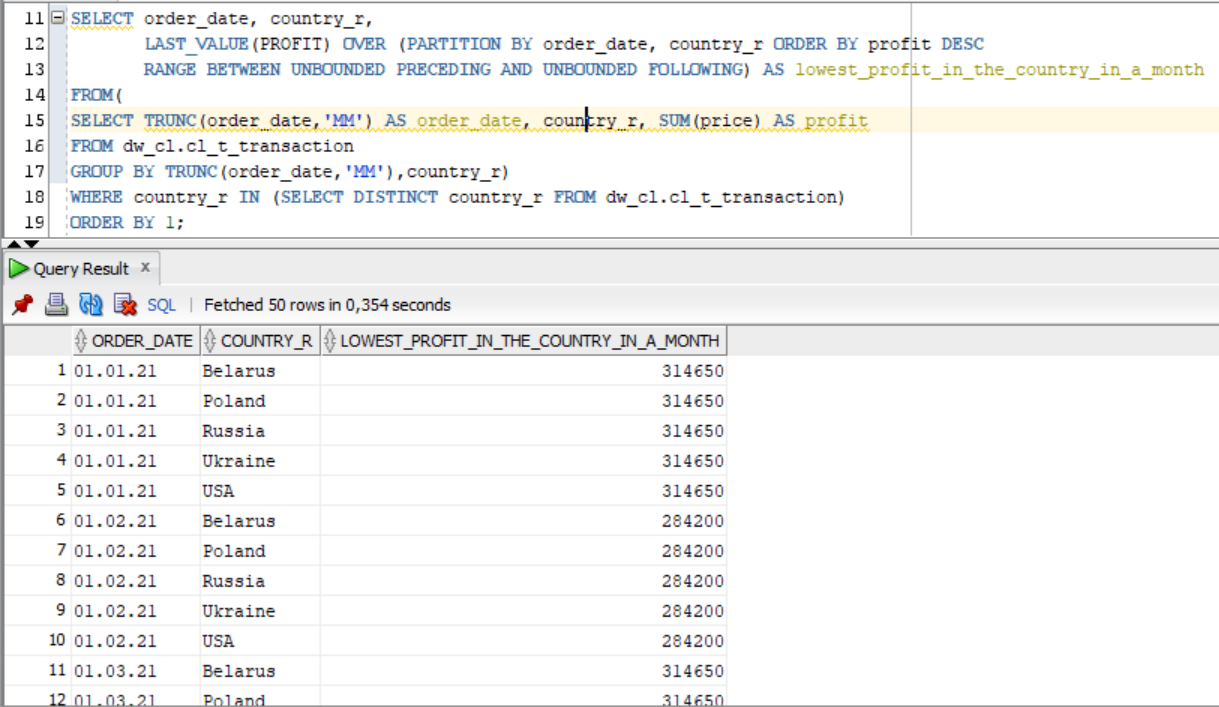
1. Create Ad Hoc SQL FIRST\_VALUE

For an example of how this function works, we used a query that displays the highest profit for each country for the month (the data turned out to be the same, since a cross join was used when generating orders):



1. Create Ad Hoc SQL LAST\_VALUE

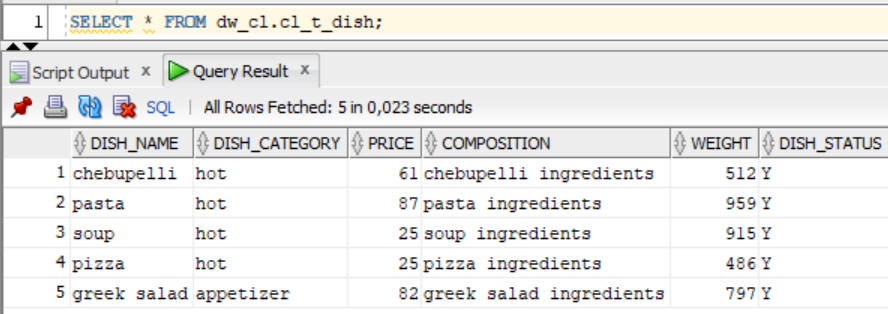
For an example of how this function works, we used a query that displays the lowest profit for each country for the month (the data turned out to be the same, since a cross join was used when generating orders):

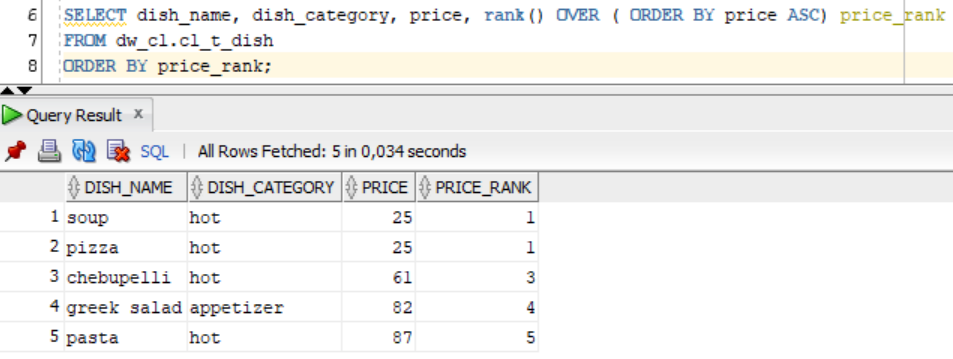


**Task 2 - Create Ad Hoc SQL RANK, DENSE\_RANK, ROWNUM**

1. Create Ad Hoc SQL RANK

Let's show an example of how the rank function works on a table containing information about dishes. Select from table dw\_cl.cl\_t\_dish:

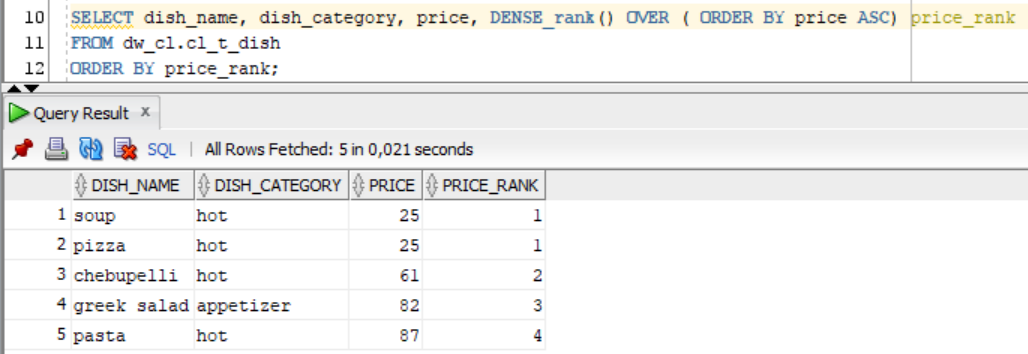




The Oracle/PLSQL RANK function returns the rank of a value in a group of values. It is very similar to the [DENSE\_RANK function](https://www.techonthenet.com/oracle/functions/dense_rank.php). However, the rank function can cause non-consecutive rankings if the tested values are the same. Whereas, the [DENSE\_RANK function](https://www.techonthenet.com/oracle/functions/dense_rank.php) will always result in consecutive rankings.

1. Create Ad Hoc SQL DENSE\_RANK

Execute the previous query, only use the dense\_rank function:

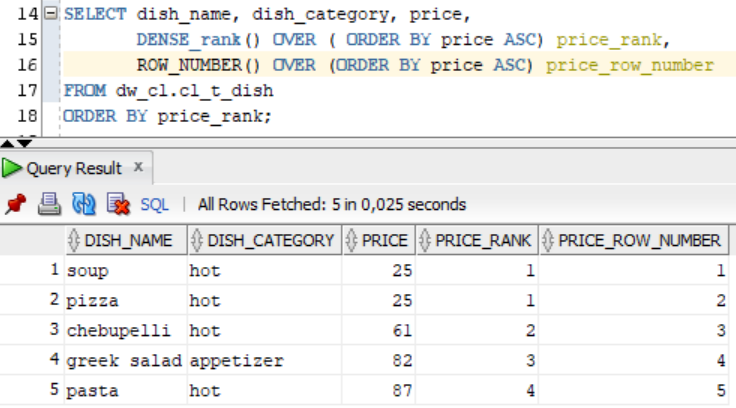


When using the rank function, the value 2 was missed, but when using dense\_rank, all values ​​were displayed sequentially, without missing values.

Unlike the [RANK()](https://www.oracletutorial.com/oracle-analytic-functions/oracle-rank/) function, the DENSE\_RANK() function returns rank values as consecutive integers. It does not skip rank in case of ties. Rows with the same values for the rank criteria will receive the same rank values.

1. Create Ad Hoc SQL ROWNUM

An example of the operation of the row\_number function will be shown in the previous query using the dense\_rank function, adding a field calculated using the row\_number function:

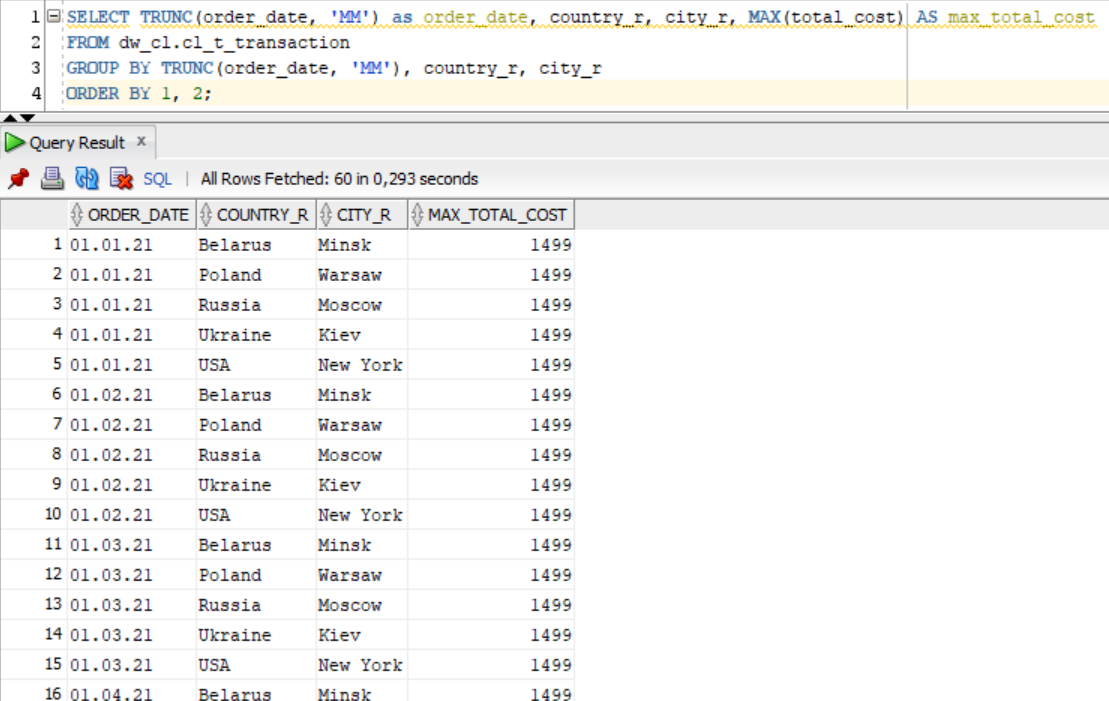


The row\_number function assigns a unique number for each row in the ordered result set. If the partitioning clause is specified, then each row is assigned a number unique within a data partition, based upon its position in the sort order in that partition.

**Task 3 - Create Ad Hoc SQL AGGREAGATE FUNCS**

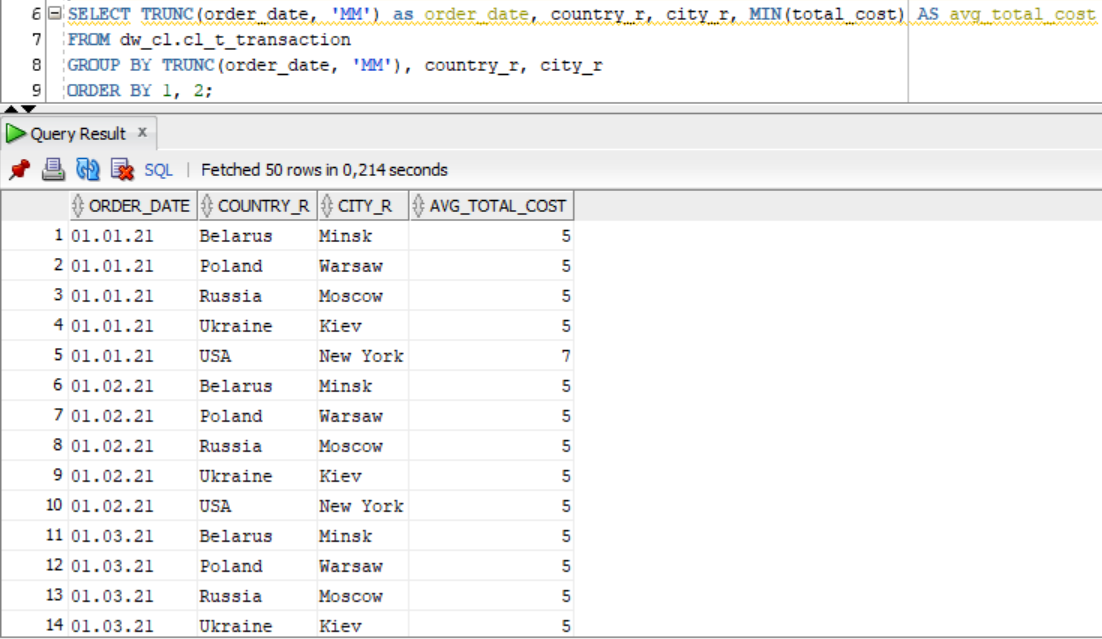
1. Create Ad Hoc SQL MAX()

An example of the operation of the MAX() function will be shown on a query that displays the maximum order amount in each city for a month:



1. Create Ad Hoc SQL MIN()

An example of the operation of the MIN() function will be shown on a query that displays the minimum order amount in each city for a month:



1. Create Ad Hoc SQL AVG()

An example of the operation of the AVG() function will be shown on a query that displays the average order amount in each city for a month:

