Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский ядерный университет «МИФИ»

Институт интеллектуальных кибернетических систем

Кафедра №42 «Криптология и кибербезопасность»

Отчёт о лабораторной работе №12 (Модуль 5)

по дисциплине "Наука о данных"

Выполнила студентка группы Б17-505:

Худоярова А.М.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Москва – 2020

Оглавление

[**Window functions** 2](#_Toc52146197)

[1. Create reports about ranking for sales persons 2](#_Toc52146198)

[A. 2](#_Toc52146199)

[C. 3](#_Toc52146200)

[2. Create reports about customer base 3](#_Toc52146201)

[A. 3](#_Toc52146202)

[B. 3](#_Toc52146203)

[C. 4](#_Toc52146204)

# **Window functions**

1. Create reports about ranking for sales persons

Rank your sales persons by number of clients, report should include rank, sales person id and client number in descending order.

create view report\_2\_1A as

select SalesPerson, Count(CustomerID) as AmountOfClients,

    Rank() over (order by Count(CustomerID)  desc) as RankSalesPerson

from [SalesLT].[Customer]

group by SalesPerson

GO

select \* from report\_2\_1A

union all

select 'Amount of rows', null, count(\*) from report\_2\_1A

Rank your sales persons by number of sales, your report should include all sales persons with id, dense rank and number of sales in descending order.

create view Report\_2\_1B as

select SalesPerson, count(SalesOrderID) as AmountOfSales,

 DENSE\_RANK () over (order by count(SalesOrderID) desc) DenseRank

from [SalesLT].[Customer]

left join [SalesLT].[SalesOrderHeader] on [SalesLT].[SalesOrderHeader].CustomerID = [SalesLT].[Customer].CustomerID

group by SalesPerson;

GO

select \* from Report\_2\_1B

union all

select 'Amount of rows', count(\*), null from Report\_2\_1B;



Rank your sales person by income from sales, your report should include all sales persons with id, rank and income in descending order.

create view Report\_2\_1C as

select SalesPerson, ISNULL(SUM(LineTotal),0) as IncomeSum,

rank() over (order by ISNULL(SUM(LineTotal),0) desc) as RankIncSales

from [SalesLT].[Customer]

left join [SalesLT].[SalesOrderHeader] on [SalesLT].[SalesOrderHeader].CustomerID = [SalesLT].[Customer].CustomerID

left join [SalesLT].[SalesOrderDetail] on [SalesLT].[SalesOrderDetail].SalesOrderID = [SalesLT].[SalesOrderHeader].SalesOrderID

group by SalesPerson;

GO

select \* from Report\_2\_1C

union all

select 'Amount of rows',count(\*), null from Report\_2\_1C

1. Create reports about customer base

Rank regions / states in the country by number of customers (use main office address), your report should include country, state or region, number of customers and percent rank ordered by country (alphabetically) and number of clients (descending). In case of equality in client numbers order region or states alphabetically.

select CountryRegion, StateProvince,

count([SalesLT].[Customer].CustomerID) as AmountOfClients,

percent\_rank ()

OVER (partition by CountryRegion order by count([SalesLT].[Customer].CustomerID) ) as PercentRank

from [SalesLT].[Customer]

join [SalesLT].[CustomerAddress] on [SalesLT].[Customer].CustomerID = [SalesLT].[CustomerAddress].CustomerID

and [SalesLT].[CustomerAddress].AddressType = 'Main Office'

join [SalesLT].[Address] on [SalesLT].[Address].AddressID = [SalesLT].[CustomerAddress].AddressID

group by CountryRegion, StateProvince

order by CountryRegion, AmountOfClients , StateProvince;



Include in previous report customers without information about address. Use dense rank instead of percent rank in that report.

select CountryRegion, StateProvince,

count([SalesLT].[Customer].CustomerID) as AmountOfClients,

dense\_rank ()

OVER (partition by CountryRegion order by count([SalesLT].[Customer].CustomerID) ) as DenseRank

from [SalesLT].[Customer]

left join [SalesLT].[CustomerAddress] on [SalesLT].[Customer].CustomerID = [SalesLT].[CustomerAddress].CustomerID

and [SalesLT].[CustomerAddress].AddressType = 'Main Office'

left join [SalesLT].[Address] on [SalesLT].[Address].AddressID = [SalesLT].[CustomerAddress].AddressID

group by CountryRegion, StateProvince

order by CountryRegion, AmountOfClients , StateProvince;



Rank cities in the country by number of customers (use main office address), your report should include country, state or region, city,  number of clients, rank (use plane rank here) and difference in number of client with previous position in by country ranking (for first position should be null). Order your report by country name (alphabetically), number of clients (descending) and city name (alphabetically).

select CountryRegion, StateProvince, City,

    count([SalesLT].[Customer].CustomerID) as AmountOfClients,

    abs((count([SalesLT].[Customer].CustomerID) - lag(count([SalesLT].[Customer].CustomerID)) over (partition by CountryRegion order by count(Customer.CustomerID) desc, City))) as Difference,

    rank() over (partition by CountryRegion order by count([SalesLT].[Customer].CustomerID) desc, City) as RankAmount

from [SalesLT].[Customer]

    join [SalesLT].[CustomerAddress] on [SalesLT].[Customer].CustomerID = [SalesLT].[CustomerAddress].CustomerID

    and [SalesLT].[CustomerAddress].AddressType = 'Main Office'

    join [SalesLT].[Address] on [SalesLT].[Address].AddressID = [SalesLT].[CustomerAddress].AddressID

group by CountryRegion, StateProvince, City

order by CountryRegion, AmountOfClients desc, City, RankAmount;