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

[**Cube, rollup and Others** 2](#_Toc51759293)

[A. 2](#_Toc51759294)

[Without0: 2](#_Toc51759295)

[With0 3](#_Toc51759296)

[B. 4](#_Toc51759297)

[C 7](#_Toc51759298)

[D 8](#_Toc51759299)

[E 9](#_Toc51759300)

[**Window functions** 9](#_Toc51759301)

[1. Create reports about ranking for sales persons 9](#_Toc51759302)

[A. 9](#_Toc51759303)

[C. 10](#_Toc51759304)

[2. Create reports about customer base 10](#_Toc51759305)

[A. 10](#_Toc51759306)

[B. 11](#_Toc51759307)

[C. 11](#_Toc51759308)

# **Cube, rollup and Others**



Report about income from sales by product, client and sales person. Please mind discounts. Also mind that for some combinations of values in these dimensions there are no sales at all, so create two versions of queries with and without zero values.

### Without0:

create view [dbo].[report\_tmp] as

 select [SalesLT].[Customer].CustomerID,FirstName, LastName, LineTotal, Name, SalesPerson, [SalesLT].[Product].ProductID from [SalesLT].[Customer]

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

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

 join [SalesLT].[Product] on [SalesLT].[Product].ProductID = [SalesLT].[SalesOrderDetail].ProductID;

GO

create view Report

as

select CustomerID, SalesPerson, ProductID,

SUM(LineTotal) as TotalSum, "Price Range" =

  CASE

        WHEN GROUPING (CustomerID) = 0 and GROUPING (SalesPerson) = 1 and GROUPING (ProductID) = 1 THEN 'Result by customer'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (SalesPerson) = 0 and GROUPING (ProductID) = 1 THEN 'Result by sales person'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (SalesPerson) = 1 and GROUPING (ProductID) = 0 THEN 'Result by product'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (SalesPerson) = 0 and GROUPING (ProductID) = 1 THEN 'Result by customer and sales person'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (SalesPerson) = 1 and GROUPING (ProductID) = 0 THEN 'Result by customer and product'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (SalesPerson) = 0 and GROUPING (ProductID) = 0 THEN 'Result by sales person and product'

       WHEN GROUPING (CustomerID) = 0 and GROUPING (SalesPerson) = 0 and GROUPING (ProductID) = 0 THEN 'Result by customer, sales person, product'

   END

from report\_tmp

group by cube (CustomerID, SalesPerson, ProductID);

GO

select \* from Report

UNION ALL

select null, null, null, count(\*), 'amount of rows' from Report;

GO

### With0

create view [dbo].[report\_tmp\_with0]

as

select [SalesLT].[Customer].CustomerID, [SalesLT].[Product].ProductID, [SalesLT].[Customer].SalesPerson, FirstName, LastName, LineTotal, Name

from [SalesLT].[SalesOrderHeader]

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

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

full outer join [SalesLT].[Product] on [SalesLT].[Product].ProductID = [SalesLT].[SalesOrderDetail].ProductID;

GO

create view Report\_with\_0

as

select CustomerID, SalesPerson, ProductID,

SUM(LineTotal) as TotalSum, "Price Range" =

  CASE

        WHEN GROUPING (CustomerID) = 0 and GROUPING (SalesPerson) = 1 and GROUPING (ProductID) = 1 THEN 'Result by customer'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (SalesPerson) = 0 and GROUPING (ProductID) = 1 THEN 'Result by sales person'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (SalesPerson) = 1 and GROUPING (ProductID) = 0 THEN 'Result by product'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (SalesPerson) = 0 and GROUPING (ProductID) = 1 THEN 'Result by customer and sales person'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (SalesPerson) = 1 and GROUPING (ProductID) = 0 THEN 'Result by customer and product'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (SalesPerson) = 0 and GROUPING (ProductID) = 0 THEN 'Result by sales person and product'

       WHEN GROUPING (CustomerID) = 0 and GROUPING (SalesPerson) = 0 and GROUPING (ProductID) = 0 THEN 'Result by customer, sales person, product'

   END

from report\_tmp\_with0

group by cube (CustomerID, SalesPerson, ProductID);

GO

select \* from Report\_with\_0

UNION ALL

select null, null, null, count(\*), 'amount of rows' from Report\_with\_0;

GO



Report about income from sales by product, client and country (region) for billing, shipping and client residency as they can be different. Is it case according our data? But you should generalize in any case. Please mind discounts. You should include in that report only data that supported by sales (so no zero entries except discounted price is zero)

create view report\_b\_tmp as

select [SalesLT].[Customer].CustomerID, [SalesLT].[SalesOrderHeader].SalesOrderID, ShipToAddressID, BillToAddressID, [SalesLT].[CustomerAddress].AddressID as ResidenceID, SalesPerson, [SalesLT].[Product].ProductID, ISNULL(LineTotal,0) as TotalSum

from [SalesLT].[SalesOrderHeader]

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

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

join [SalesLT].[Product] on [SalesLT].[SalesOrderDetail].ProductID = [SalesLT].[Product].ProductID

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

create view Report\_B as

select report\_b\_tmp.CustomerID, ProductID,Ship.CountryRegion as ShipAddress,

Bill.CountryRegion as BillAddress, Residence.CountryRegion as ResidenceAddress,

SUM(TotalSum) as IncomeSum,

 "Price Range" =

  CASE

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by residence'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Result by bill address'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by bill address and residence'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Result by ship address'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by residence and ship address'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Result by ship and bill address'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by residence, ship and bill address'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Result by product'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by product and residence'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Result by product and bill address'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by residence, bill address, product'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Result by product, ship address'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by product, ship address, residence'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Result by product, ship and bill address'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by residence, ship and bill address, product'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Result by customer'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by residence and customer'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Result by customer and bill address'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by customer, bill address and residence'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Result by ship address and customer'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by customer, ship address, residence'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Result by customer, ship and bill address'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by customer, ship and bill address, residence'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Result by customer and product'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by residence, customer and product'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Result by customer, product, bill address'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by residence, bill address, customer, product'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Result by customer, product, ship address'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by customer, product, ship address, residence'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Result by customer, product, bill and ship address'

        WHEN GROUPING (CustomerID) = 1 and GROUPING (ProductID) = 1 and GROUPING(Ship.CountryRegion) = 1 and GROUPING(Bill.CountryRegion) = 1 AND GROUPING(Residence.CountryRegion) = 1  THEN 'Result by all categories'

        WHEN GROUPING (CustomerID) = 0 and GROUPING (ProductID) = 0 and GROUPING(Ship.CountryRegion) = 0 and GROUPING(Bill.CountryRegion) = 0 AND GROUPING(Residence.CountryRegion) = 0  THEN 'Usual data'

   END

from

report\_b\_tmp,

[SalesLT].[Address] Ship,

[SalesLT].[Address] Bill,

[SalesLT].[Address] Residence

where

Ship.AddressID = report\_b\_tmp.ShipToAddressID

and Bill.AddressID = report\_b\_tmp.BillToAddressID

and Residence.AddressID = report\_b\_tmp.ResidenceID

group by cube (CustomerID, ProductID, Ship.CountryRegion, Bill.CountryRegion, Residence.CountryRegion);

select \* from Report\_B

union all

select null, null, null, null,null, count(\*), 'amount of rows' from Report\_B;

## C

Report about income from sales and provided discounts by location in form of hierarchy city>state/province>country/region. In that report you can rely on unique geographical names, but in general it is not the case. Think about how to solve that task in case that there is a possibility of existence of multiple cities in the same province with the same name. For big cities someone would need more detailed report that can include city districts, solve this task for extra points.

create view report\_tmp\_c as

select [SalesLT].[Customer].CustomerID, [SalesLT].[SalesOrderHeader].SalesOrderID, ShipToAddressID, [SalesLT].[Product].ProductID, OrderQty \* UnitPrice \* UnitPriceDiscount as Discount, ISNULL(LineTotal,0) as TotalSum

from [SalesLT].[SalesOrderHeader]

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

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

join [SalesLT].[Product] on [SalesLT].[SalesOrderDetail].ProductID = [SalesLT].[Product].ProductID

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

create view Report\_C as

select [SalesLT].[Address].CountryRegion,[SalesLT].[Address].City, [SalesLT].[Address].StateProvince,  SUM(TotalSum) as IncomeSum, SUM(Discount) as TotalDiscount

from report\_tmp\_c

join [SalesLT].[Address] on [SalesLT].[Address].AddressID = report\_tmp\_c.ShipToAddressID

group by

grouping sets (([SalesLT].[Address].City, [SalesLT].[Address].StateProvince, [SalesLT].[Address].CountryRegion), ( [SalesLT].[Address].StateProvince, [SalesLT].[Address].CountryRegion), (  [SalesLT].[Address].CountryRegion));

select \* from Report\_C

union all

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

## D

Report about income from sales and provided discounts by product and hierarchy of product categories (high level category-> next level category->...->low level category->product). Please mind that some products can be outside (any) category or be only partially categorized (be not in low level of hierarchy). You can rely on you data to solve to solve this task (especially on that how much subcategories in the current data set), but try to think how to solve this task in general (with arbitrary category tree).

create view report\_d\_tmp as

select [SalesLT].[SalesOrderDetail].SalesOrderId, [SalesLT].[Product].ProductID, [SalesLT].[Product].Name, [SalesLT].[Product].ProductCategoryID as Category, SelfCategory.ParentProductCategoryID as HigherLevelCategory,

 LineTotal, OrderQty \* UnitPrice \* UnitPriceDiscount as Discount

from [SalesLT].[SalesOrderHeader]

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

join [SalesLT].[Product] on [SalesLT].[SalesOrderDetail].ProductID = [SalesLT].[Product].ProductID

join [SalesLT].[ProductCategory] as SelfCategory on [SalesLT].[Product].ProductCategoryID = SelfCategory.ProductCategoryID

left join [SalesLT].[ProductCategory] as ParentCategory on SelfCategory.ParentProductCategoryID = ParentCategory.ProductCategoryID;

GO

create view Report\_D as

select Name, Category, HigherLevelCategory, SUM(LineTotal) as TotalSum, SUM(Discount) as TotalDiscount

from report\_d\_tmp

group by

rollup (Category, HigherLevelCategory, Name);

GO

select \* from Report\_D

union all

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

GO

E

Create integral report on number of product sales by product, client, sales person and hierarchy of regions.

select [SalesLT].[Customer].CustomerID, ProductID, SalesPerson, City, StateProvince, CountryRegion, SUM(OrderQty) as Amount

from [SalesLT].[SalesOrderHeader]

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

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

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

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

group by

rollup([SalesLT].[Customer].CustomerID, ProductID, SalesPerson, City, StateProvince, CountryRegion);

# **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,

    (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,City, AmountOfClients desc;