Домашняя работа 4

Углубленный уровень SQL

3. При каких значениях оконные функции Row Number, Rank и Dense Rank вернут одинаковый результат?

Не придумала таких(

4. a) Проверьте, есть ли здесь UnitMeasureCode, начинающиеся на букву ‘Т’.

select UnitMeasureCode from [Production].[UnitMeasure];



Сколько всего различных кодов здесь есть?

select distinct UnitMeasureCode from [Production].[UnitMeasure];

или

select UnitMeasureCode from [Production].[UnitMeasure]

group by UnitMeasureCode;



Вставьте следующий набор данных в таблицу:

TT1, Test 1, 9 сентября 2020

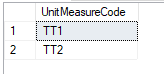
TT2, Test 2, getdate()

Проверьте теперь, есть ли здесь UnitMeasureCode, начинающиеся на букву ‘Т’.

Insert into Production.UnitMeasure (UnitMeasureCode, [Name], [ModifiedDate])

values ('TT1', 'Test 1',convert(datetime,'9 september 2020', 106)),

('TT2', 'Test 2', getdate());



b) Теперь загрузите вставленный набор в новую, не существующую таблицу Production.UnitMeasureTest. Догрузите сюда информацию из Production.UnitMeasure по UnitMeasureCode = ‘CAN’. Посмотрите результат в отсортированном виде по коду.

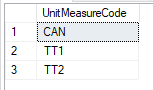
select [UnitMeasureCode] into Production.UnitMeasureTest

from [Production].[UnitMeasure]

where UnitMeasureCode like 'T%' or UnitMeasureCode='CAN';

select \* from Production.UnitMeasureTest

order by UnitMeasureCode;



c) Измените UnitMeasureCode для всего набора из Production.UnitMeasureTest на ‘TTT’.

update Production.UnitMeasureTest

set UnitMeasureCode='TTT';

d) Удалите все строки из Production.UnitMeasureTest.

delete from Production.UnitMeasureTest;

e) Найдите информацию из Sales.SalesOrderDetail по заказам 43659,43664. С помощью оконных функций MAX, MIN, AVG найдем агрегаты по LineTotal для каждого SalesOrderID.

select SalesOrderID, LineTotal '(LineTotal)',

max(LineTotal) over(partition by SalesOrderID) maksim,

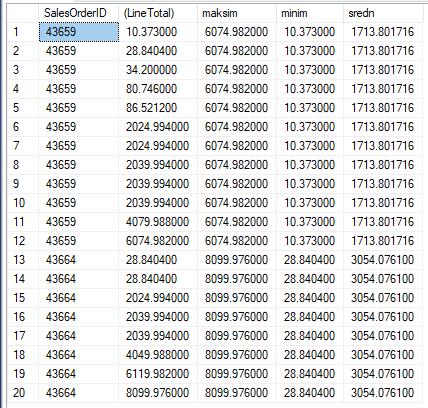
min (LineTotal) over(partition by SalesOrderID) minim,

AVG (LineTotal) over(partition by SalesOrderID) sredn

from Sales.SalesOrderDetail

where [SalesOrderID] in ('43659','43664')

order by SalesOrderID, LineTotal;



f) Создайте рейтинг cреди продавцов на основе годовых продаж SalesYTD, используя ранжирующую оконную функцию.

Добавьте поле Login, состоящий из 3 первых букв фамилии в верхнем регистре + ‘login’ + TerritoryGroup (Null заменить на пустое значение). Кто возглавляет рейтинг?

select Sales.SalesPerson.BusinessEntityID,

concat(UPPER(LEFT(LastName,3)),'login',isnull([Group],'')) Login,

dense\_rank () over (order by Sales.SalesPerson.SalesYTD desc) ratingCURR,

dense\_rank () over (order by Sales.SalesPerson.[SalesLastYear] desc) ratingPREV

from Sales.SalesPerson

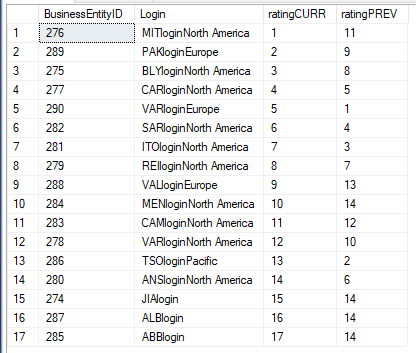
join [Person].[Person]

on [Person].[Person].BusinessEntityID=[Sales].[SalesPerson].BusinessEntityID

left join [Sales].[SalesTerritory]

on [Sales].[SalesTerritory].[TerritoryID]=[Sales].[SalesPerson].[TerritoryID]

order by ratingCURR;



А кто возглавлял рейтинг в прошлом году (SalesLastYear)?

order by ratingCURR ==> order by ratingPREV

g) Найдите первый будний день месяца (FROM не используем). Нужен стандартный код на все времена.

set datefirst 1;

select datename(dw,case when

datepart(dw,datefromparts(year(getdate()),month(getdate()),1)) in (6,7)

then '2018-01-01'

else datefromparts(year(getdate()),month(getdate()),1)

end)

as firstWD1;

--или

set datefirst 1;

select datename(dw,case when

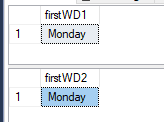
datepart(dw,DATEADD(DAY,1-DAY(getdate()),getdate())) > 5

then '2018-01-01'

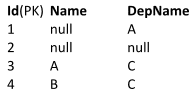
else DATEADD(DAY,1-DAY(getdate()),getdate())

end)

as firstWD2;



5. По просьбам с прошлого занятия, давайте еще раз остановимся и отточим понимание функции count. Найдите значения count(1), count(name), count(id), count(\*) для следующей таблицы:



count(1)=4;

count(name)=2;

count(id)=4;

count(\*)=4;