Домашняя работа 3

Продвинутый уровень SQL

2. a) Вывести список цен в виде текстового комментария, основанного на диапазоне цен для продукта. Вывести имя продукта и новое поле PriceRange.

X(a. StandardCost равен 0 или не определен – ‘Not for sale’

Quiery > Results to text

select name, string\_agg(isnull(StandardCost,'Not for sale'),'; ') PriceRange

from Production.Product

where StandardCost is null or StandardCost=0

group by name;

)X

select name,

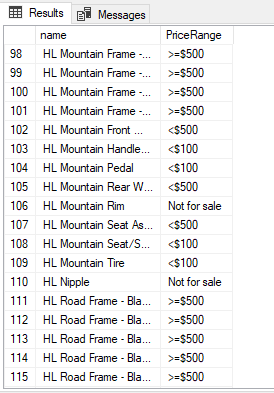
iif(StandardCost is null or StandardCost=0, 'Not for sale',

iif(StandardCost>0 and StandardCost<100, '<$100',

iif(StandardCost>=100 and StandardCost<500, '<$500', '>=$500'))) PriceRange

from Production.Product

order by name;

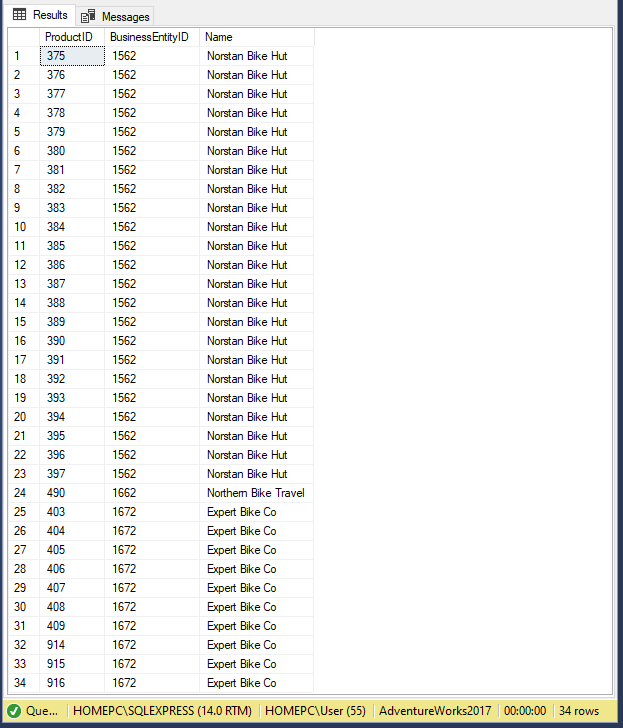
b) Найти ProductID, BusinessEntityID и имя поставщика продукции из Purchasing. ProductVendor и Purchasing.Vendor, где StandardPrice больше $10. Также в имени вендора должна присутствовать (вне зависимости от положения) буква X или имя должно начинаться на букву N.

select ProductID, v.BusinessEntityID, Name

from Purchasing.ProductVendor pv join Purchasing.Vendor v

on pv.BusinessEntityID=v.BusinessEntityID

where StandardPrice>10 and (Name like '%x%' or Name like 'N%');



c) Найти имена всех вендоров, продукция которых не продавалась за всё время.

select Name

from Purchasing.ProductVendor pv

right join Purchasing.Vendor v

on pv.BusinessEntityID=v.BusinessEntityID

where pv.BusinessEntityID is null;



3. a) Найдите производителей, препараты которых не продавались в 2019 году.

SELECT [Фирмы-производители].[Название] Firm FROM [Фирмы-производители]

JOIN [Фармпрепараты] ON [id фирмы]=[Фирмы-производители].id

LEFT JOIN [Продажи] ON [id фармпрепарата]=[Фармпрепараты].id

LEFT JOIN [Кассовые чеки] Kass ON [Кассовые чеки].id=[id чека]

GROUP BY [Фирмы-производители].[Название]

HAVING COUNT(SELECT 1 FROM [Кассовые чеки]

WHERE YEAR([Дата])=2019 AND id=Kass.id)<1;

b) Выведите увеличенную в 2 раза цену препаратов типа А.

SELECT 2\*[Цена] Dabl FROM [Фармпрепараты]

JOIN [Типы фармпрепаратов] ON [Типы фармпрепаратов].id=[id типа]

WHERE [Тип]=’A’;

c) Найдите производителей и кол-во фармпрепаратов для кажого из них, где препарат не относится к типу А.

SELECT [Фирмы-производители].[Название], COUNT([Фармпрепараты].id)

FROM [Фирмы-производители]

JOIN [Фармпрепараты] ON [id фирмы]=[Фирмы-производители].id

JOIN [Типы фармпрепаратов] ON [Типы фармпрепаратов].id=[id типа]

WHERE NOT [Тип]=’A’

GROUP BY [Фирмы-производители].[Название];

d) Название аптеки и названия фармпрепаратов, которые в ней продавались по субботам.

SELECT [Аптечные пункты].[Название], [Фармпрепараты].[Название]

FROM [Аптечные пункты]

JOIN [Продажи] ON [id аптечного пункта]=[Аптечные пункты].id

JOIN [Фармпрепараты] ON [Фармпрепараты].id=[id фармпрепарата]

JOIN [Кассовые чеки] ON [Кассовые чеки].id=[id чека]

WHERE DATENAME(dw,[Дата])=’Saturday’;

4. Для того, чтобы изучить существующие функции (ссылки: 1, 2, 3, 4), придумайте и решите задачи, в решении которых использовались бы функции.

[DATEDIFF](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/datediff-transact-sql?view=sql-server-ver15)

[CONCAT\_WS](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/concat-ws-transact-sql?view=sql-server-ver15)

/\* Отобразить информацию о сотрудниках в два столбца:

дата приёма на работу;

строка, содержащая полное ФИО(разделённое пробелами), через разделитель ">>" должность и зарплату сотрудника \*/

select [HireDate],

concat\_ws(' >> ',concat\_ws(' ',[FirstName],[MiddleName],[LastName]), [JobTitle], [Rate])

'name, position, rate info'

from [HumanResources].[Employee] join [HumanResources].[EmployeePayHistory]

on [HumanResources].[Employee].[BusinessEntityID]=[HumanResources].[EmployeePayHistory].[BusinessEntityID]

join [Person].[Person] on [Person].[Person].[BusinessEntityID]=[HumanResources].[Employee].[BusinessEntityID]

order by [HireDate] desc