1. Написать представление, в котором необходимо вывести перечень магазинов с указанием их места расположения. При этом название страны следует вывести на английском языке и в сокращенном виде (например, United States – US)

create view View\_5\_1 (Country,ShopName)

as

select 'Country' = case c.CountryName

when 'Украина' then 'Ukraine -UA'

when 'Грузия' then 'Georgia - GR'

when 'Россия' then 'Russia - RU'

when 'Беларусь' then 'Belarus - BE'

when 'Казахстан' then 'Kazakhstan - KZ'

end,

sh.ShopName

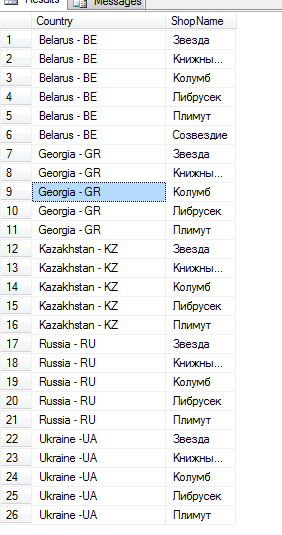
from [Shop]sh join [Country] c on sh.CountryId=c.Id

go

select \* from view\_5\_1

group by Country,ShopName

go



2. Написать запрос, который изменяет данные в таблице Books следующим образом: если книги были изданы после 2008 года, тогда их тираж увеличить на 1000 екзмпляров, иначе тираж увеличить на 100 ед. Примечание! Воспользоваться инструкцией CASE.

declare @value int;

select @value = case

when YEAR(b.DateOfPubl)>2017 then 1000

when YEAR(b.DateOfPubl)<=2017 then 100

end

from [Books]b

update books

set Сirculation = Сirculation+@value

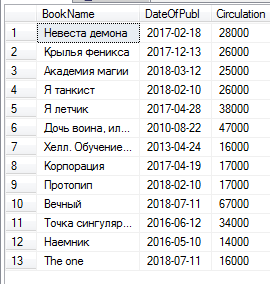
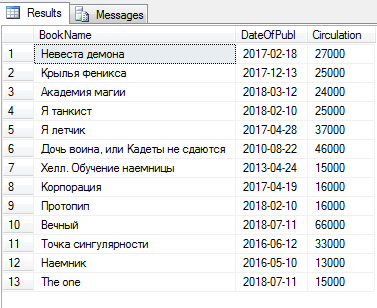
go

select b.BookName, b.DateOfPubl,b.Сirculation

from [Books]b

go

До изменений После изменений



3. Написать виртуальное представление, которое выводит общее количество продаж и дату последней реализации для каждого магазина.

WITH Virtual(ShopName, SaleCount, DateOfSal) AS

(select sh.ShopName, sum(s.SaleCount),max (s.DateOfSal)

from [Shop] sh join [Sales]s on s.Id=sh.SalesId

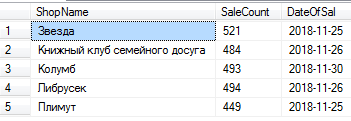
Group by sh.ShopName

)

select \*

from Virtual

go



4. Создать хранимую процедуру, которая выводит на экран список магазинов, которые продали хотя бы одну книгу Вашего издательства. Указать также месторасположение (страну) магазина.

create procedure Less5\_task\_4

as

select sh.ShopName, s.SaleCount, c.CountryName

From [Sales]s join [Shop]sh on s.Id=sh.SalesId

join [Country] c on sh.CountryId=c.Id

where s.SaleCount>0

go

exec Less5\_task\_4;



5. Написать процедуру, позволяющую просмотреть все книги определенного автора, при этом его имя передается при вызове

create procedure Less5\_task\_5\_1

@authorName nvarchar(100) ,

@authorSurName nvarchar(100)

as

select a.AuthorSurName+ ' ' + a.AuthorName as 'Author', b.BookName as 'Book'

from [Author] a join [Books] b on a.Id=b.AuthorId

where a.AuthorName=@authorName and a.AuthorSurName=@authorSurName

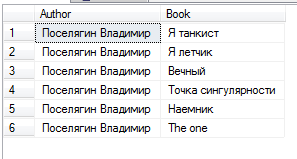
go

declare @authorName nvarchar(100) , @authorSurName nvarchar(100)

set @authorName = 'Владимир'

set @authorSurName = 'Поселягин'

exec Less5\_task\_5\_1 @authorName, @authorSurName;



6. Создать хранимую процедуру, которая возвращает максимальное из двух чисел.

create proc maxAB

@a int,

@b int

as

declare @max int

set @max=@a

if @max<@b

set @max=@b

return @max

go

declare @res int

exec @res=maxAB 5,25

select 'max from 5 and 25 is ', @res

go



7. Написать процедуру, которая выводит на экран книги и цены по указанной тематике. При этом необходимо указывать направление сортировки: 0 – по цене, по росту, 1 – по убыванию, любое другое – без сортировки.

create view View\_7 (JanreName,BookName, Price)

as

select j.JanreName, b.BookName, b.Price

from [Books] b join [Janre] j on b.JanreId=j.Id

go

create proc BooksByJanre

@janreName nvarchar(100) ,

@sort int

as

if @sort=0

begin

select \* from view\_7

where JanreName=@janreName

order by Price

end

else

if @sort=1

begin

select \* from view\_7

where JanreName=@janreName

order by Price desc

end

else

begin

select \* from view\_7

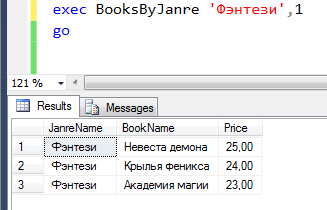
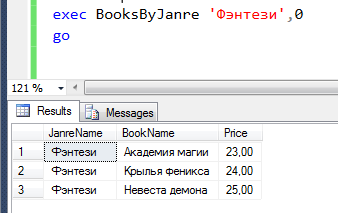
where JanreName=@janreName

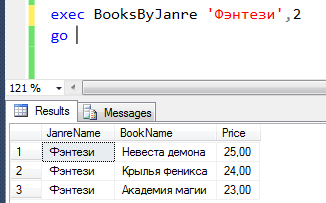
end

go

exec BooksByJanre 'Фэнтези',0

go





8. Написать процедуру, которая возвращает полное имя автора, книг которого больше всех было издано.

create proc MaxResAuthor

as

select tmp.Author

from (select top 1 a.AuthorSurName+' '+a.AuthorName+' '+a.AuthorSecName as 'Author', max(b.Сirculation) as 'Circulation'

from [Books] b join [Author]a on b.AuthorId=a.Id

group by a.AuthorSurName+' '+a.AuthorName+' '+a.AuthorSecName

order by 2 desc) as tmp

go

exec MaxResAuthor

go



9. Написать процедуру для расчета факториала числа.

create proc Factorial

@a int

as

declare @p int, @i int

set @p=1

set @i=1

while @i<=@a

begin

set @p\*=@i;

set @i+=1;

end

return @p

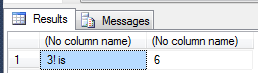
go

declare @res int

exec @res=Factorial 3

select '3! is ', @res

go



10.Написать хранимую процедуру, которая позволяет увеличить дату издательства каждой книги, которая соответствует шаблону на 2 года. Шаблон передается в качестве параметра в процедуру.

Выполнение задания было через временную таблицу, дабы не напороть боков в основной таблице

Select \*

into #TMP\_10

from [Books]

go

create proc UpdateTMP\_10

@tamplatBookName nvarchar(100)

as

UPDATE [#TMP\_10]

SET DateOfPubl=DATEADD(year, 2, DateOfPubl)

WHERE BookName LIKE @tamplatBookName+'%'

go

exec UpdateTMP\_10 'Я'

go

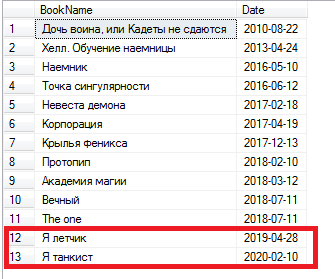
select tmp.BookName, tmp.DateOfPubl as 'Date'

from [#tmp\_10] tmp

order by 2

go

До Стало



11.Написать хранимую процедуру с параметрами, определяющими диапазон дат выпуска книг. Процедура позволяет обновить данные о тираже выпуска книг  
по следующим условиям:

• Если дата выпуска книги находится в определенном диапазоне, тогда тираж нужно увеличить в два раза, а цену за единицу увеличить на 20%;  
• Если дата выпуска книги не входит в диапазон, тогда тираж оставить без изменений.Предусмотреть вывод на экран соответствующих сообщений об ошибке, если передаваемые даты одинаковые, или конечная дата промежутка меньше начала, или же начальная больше текущей даты.

Выполнение задания было через временную таблицу, дабы не напороть боков в основной таблице

Select \*

into #TMP\_11

from [Books]

go

create proc UpdateTMP\_11

@min date,

@max date

as

UPDATE [#TMP\_11]

SET Сirculation =Сirculation \* 2, Price=Price\*1.2

WHERE DateOfPubl between @min and @max

go

exec UpdateTMP\_11 '2016-02-15','2018-03-16'

go

select tmp.BookName, tmp.DateOfPubl, tmp.Price, tmp.Сirculation

from [#tmp\_11] tmp

order by 2

go

Было Стало

