create table [Authors]

(

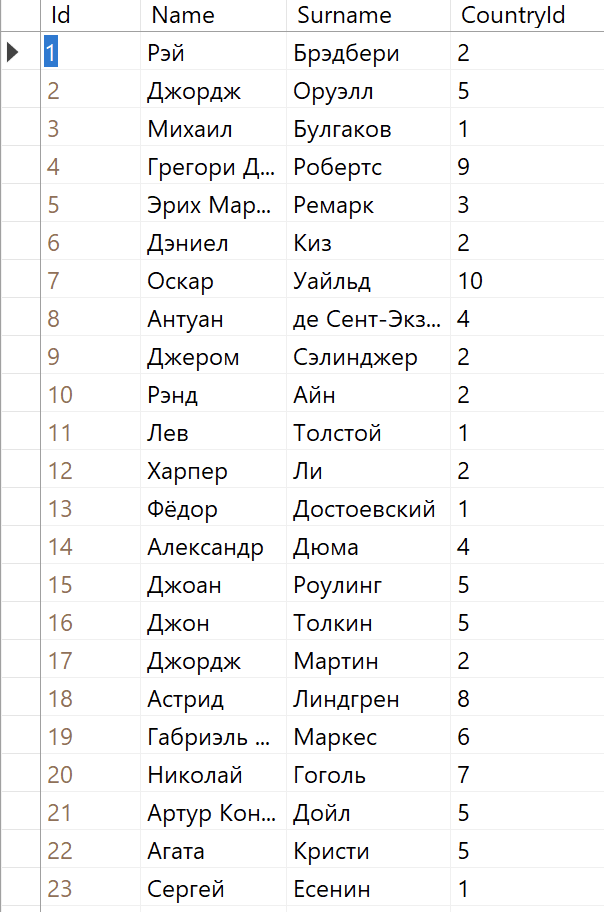
[Id] int not null identity (1,1) primary key,

[Name] nvarchar(max) not null check ([Name] <> N''),

[Surname] nvarchar(max) not null check ([Surname] <> N''),

[CountryId] int not null foreign key references Countries(Id)

)



create table [Books]

(

[Id] int not null identity (1,1) primary key,

[Name] nvarchar(max) not null check ([Name] <> N''),

[Pages] int not null check ([Pages] > 0),

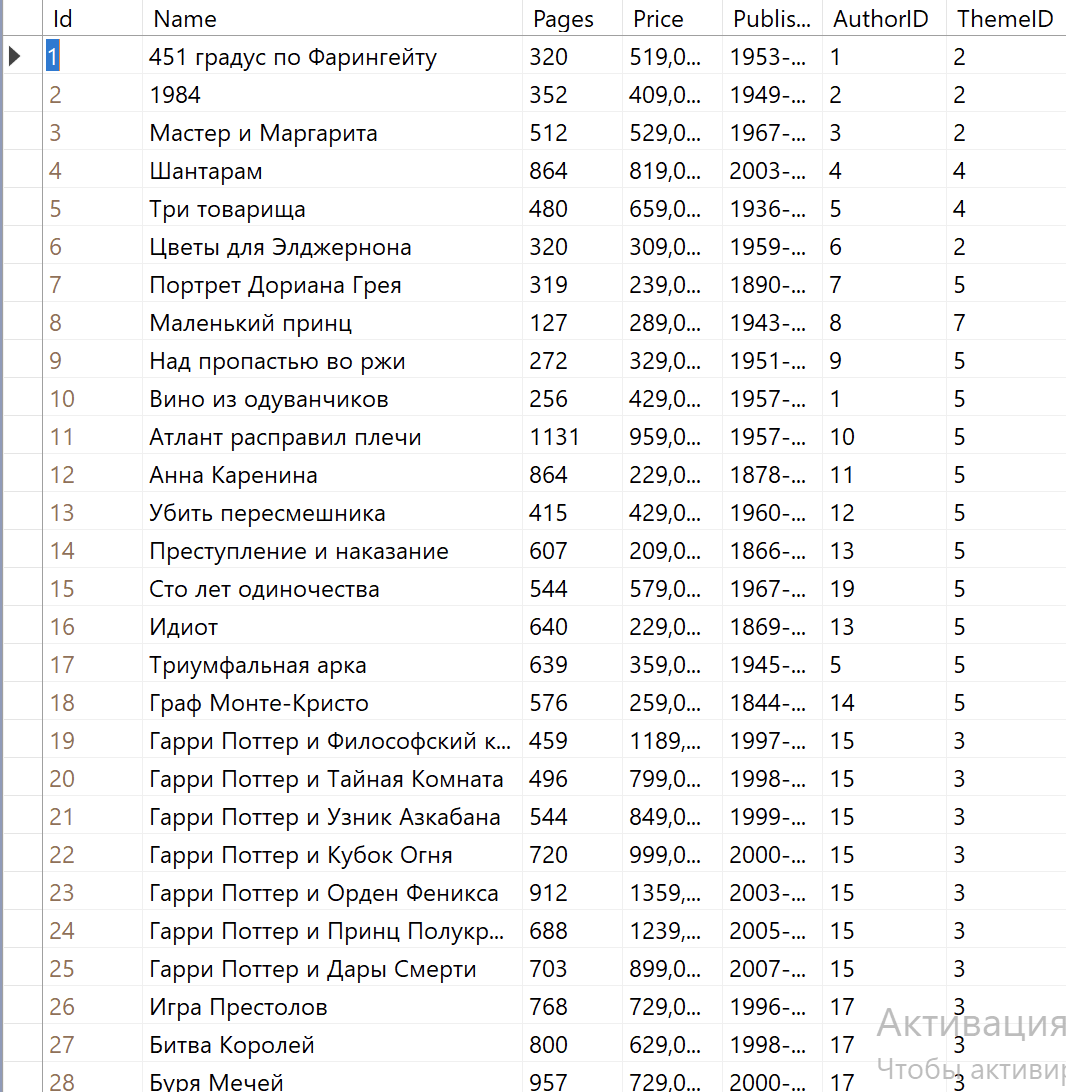
[Price] money not null check ([Price] > 0.0),

[PublishDate] date not null check ([PublishDate] < convert(date, getdate())),

[AuthorID] int not null foreign key references Authors(Id),

[ThemeID] int not null foreign key references Themes(Id)

)



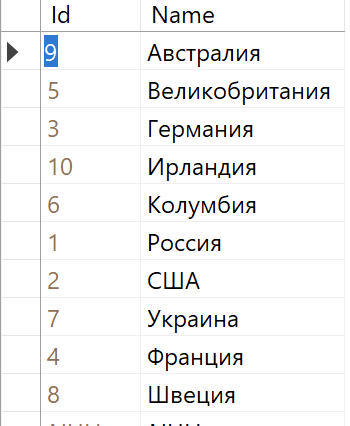
create table [Countries]

(

[Id] int not null identity (1,1) primary key,

[Name] nvarchar(50) not null unique check ([Name] <> N'')

)



create table [Sales]

(

[Id] int not null identity (1,1) primary key,

[Price] money not null check ([Price] >= 0.0),

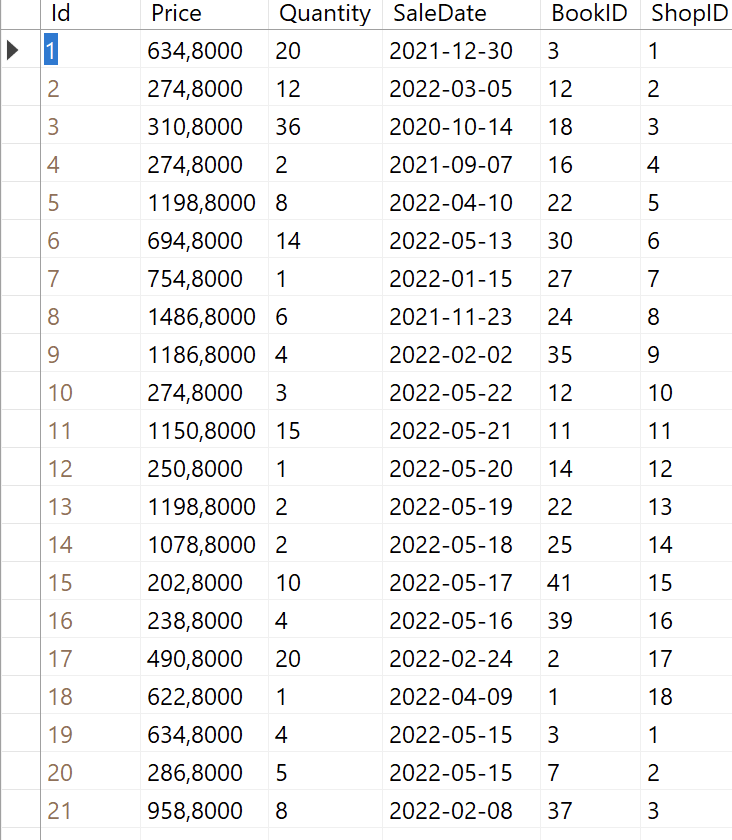
[Quantity] int not null check ([Quantity] > 0.0),

[SaleDate] date not null check ([SaleDate] <= convert(date, getdate())) default convert(date, getdate()),

[BookID] int not null foreign key references Books(Id),

[ShopID] int not null foreign key references Shops(Id)

)



create table [Shops]

(

[Id] int not null identity (1,1) primary key,

[Name] nvarchar(max) not null check ([Name] <> N''),

[CountryID] int not null foreign key references Countries(Id)

)



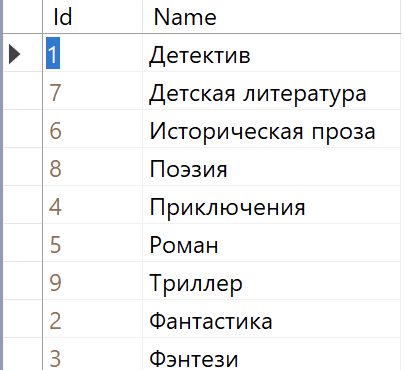
create table [Themes]

(

[Id] int not null identity (1,1) primary key,

[Name] nvarchar(100) not null unique check ([Name] <> N'')

)

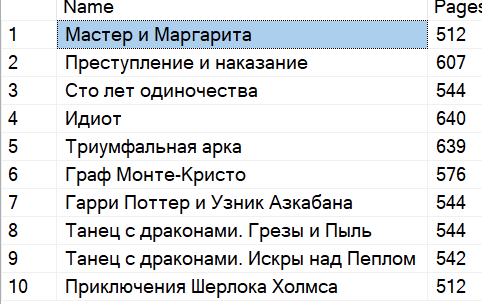


1. Показать все книги, количество страниц в которых больше 500, но меньше 650.

select Books.Name, Pages

from Books

where Books.Pages between 501 and 650

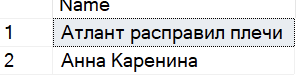


1. Показать все книги, в которых первая буква названия либо «А», либо «З».

select Books.Name

from Books

where Books.Name like '[А,З]%'



1. Показать все книги жанра «Детектив», количество проданных книг более 30 экземпляров**. (Роман, более 10 экземпляров)**

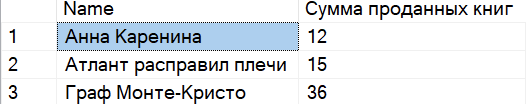
select Books.Name, sum(Sales.Quantity) as [Сумма проданных книг]

from Sales join Books on Sales.BookID = Books.Id

join Themes on Books.ThemeID = Themes.Id

where Themes.Name = 'Роман' and Sales.Quantity > 10

group by Books.Name

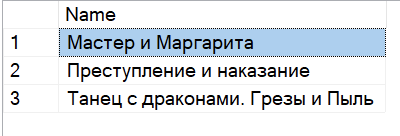


1. Показать все книги, в названии которых есть слово «Microsoft», но нет слова «Windows». **(есть союз «и» и нет слова «Гарри»)**

select Books.Name

from Books

where Books.Name like '% и %' and Books.Name not like 'Гарри%'



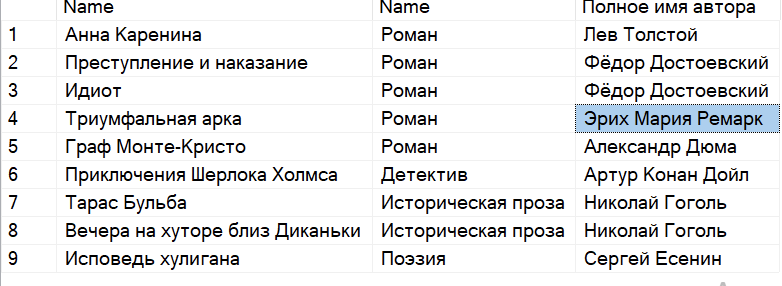
1. Показать все книги (название, тематика, полное имя автора в одной ячейке), цена одной страницы которых меньше 65 копеек.

select Books.Name, Themes.Name, Authors.Name + ' ' + Authors.Surname as [Полное имя автора]

from Authors join Books on Authors.Id = Books.AuthorID

join Themes on Themes.Id = Books.ThemeID

where Books.Price / Books.Pages < 0.65

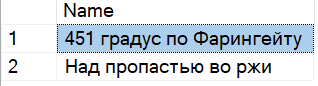


1. Показать все книги, название которых состоит из 4 слов.

select Books.Name

from Books

where len(Books.Name) - len(replace(Books.Name, ' ', '')) = 3



1. Показать информацию о продажах в следующем виде:
   * ▷  Название книги, но, чтобы оно не содержало букву «А». **(у)**
   * ▷  Тематика, но, чтобы не «Программирование». **(роман)**
   * ▷  Автор, но, чтобы не «Герберт Шилдт». **(Михаил Булгаков)**
   * ▷  Цена, но, чтобы в диапазоне от 10 до 20 гривен. **(600-1000 р)**
   * ▷  Количество продаж, но не менее 8 книг. **(не менее 4)**
   * ▷  Название магазина, который продал книгу, но он не должен быть в Украине или России.

select Books.Name as [Название книги], Themes.Name as [Жанр], Authors.Name + ' ' + Authors.Surname as [Автор], Sales.Price as [Цена], Sales.Quantity as [Количество], Shops.Name as [Название магазина]

from Books join Sales on Books.Id = Sales.BookID

join Themes on Books.ThemeID = Themes.Id

join Authors on Books.AuthorID = Authors.Id

join Shops on Shops.Id = Sales.ShopID

join Countries on Countries.Id = Shops.CountryID

where Books.Name not like '%у%' and Themes.Name <> 'Роман'

and Authors.Name + ' ' + Authors.Surname <> 'Михаил Булгаков'

and Sales.Price between 600 and 1000

and Sales.Quantity >=4

and not Countries.Name in ('Россия', 'Украина')



1. Показать следующую информацию в два столбца (числа в правом столбце приведены в качестве примера):  
   ▷ Количество авторов: 14  
   ▷ Количество книг: 47

▷ Средняя цена продажи: 85.43 грн.

▷ Среднее количество страниц: 650.6.

select 'Количество авторов' as [Величина], count(Authors.Id) as [Количество]

from Authors

union all

select 'Количество книг', count(Books.Id) as [Количество книг]

From Books

union all

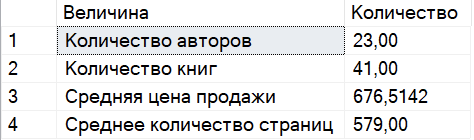
select 'Средняя цена продажи', avg(Sales.Price)

from Sales

union all

select 'Среднее количество страниц', avg(Books.Pages)

from Books

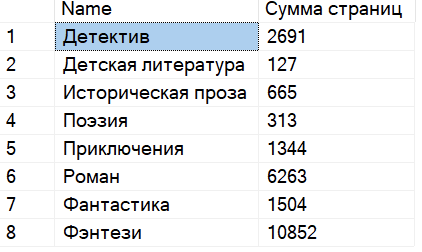


1. Показать тематики книг и сумму страниц всех книг по каждой из них.

select Themes.Name, sum(Pages) as [Сумма страниц]

from Themes join Books on Books.ThemeID = Themes.Id

group by Themes.Name



1. Показать количество всех книг и сумму страниц этих книг по каждому из авторов.

select Authors.Name + ' ' + Authors.Surname as [Имя автора],count(Books.Id) as [Количество книг], sum(Pages) as [Сумма страниц]

from Authors join Books on Books.AuthorID = Authors.Id

group by Authors.Name + ' ' + Authors.Surname



1. Показать книгу тематики «Программирование» с наибольшим количеством страниц. **(Детектив)**

select Books.Name

from Books

where Books.Pages = (select max(Books.Pages)

from Books join Themes on Books.ThemeID = Themes.Id

where Themes.Name = 'Детектив')



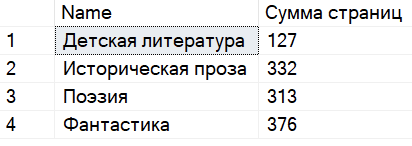
1. Показать среднее количество страниц по каждой тематике, которое не превышает 400.

select Themes.Name, avg(Pages) as [Сумма страниц]

from Themes join Books on Books.ThemeID = Themes.Id

group by Themes.Name

having avg(Pages) < 400



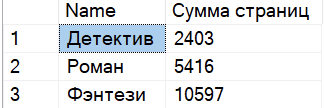
1. Показать сумму страниц по каждой тематике, учитывая только книги с количеством страниц более 400, и чтобы тематики были «Программирование», «Администрирование» и «Дизайн». **(Детектив, Фэнтези, Роман)**

select Themes.Name, sum(Pages) as [Сумма страниц]

from Themes join Books on Books.ThemeID = Themes.Id

where Books.Pages > 400 and Themes.Name in ('Детектив','Фэнтези','Роман')

group by Themes.Name



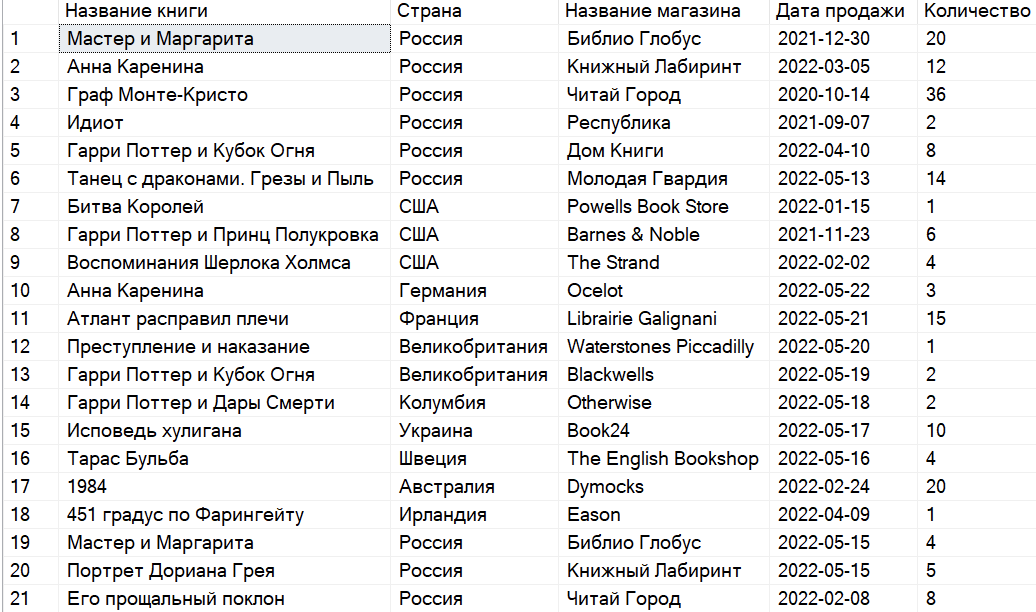
1. Показать информацию о работе магазинов: что, где, кем, когда и в каком количестве было продано.

select Books.Name as [Название книги], Countries.Name as [Страна], Shops.Name as [Название магазина], Sales.SaleDate as [Дата продажи], Sales.Quantity as [Количество]

from Books join Sales on Books.Id = Sales.BookID

join Shops on Sales.ShopID = Shops.Id

join Countries on Countries.Id = Shops.CountryID



1. Показать самый прибыльный магазин.

select top 1 Shops.Name as [Название магазина], sum(Sales.Quantity \* Sales.Price) as [Выручка]

from Sales join Shops on Sales.ShopID = Shops.Id

group by Shops.Name

order by sum(Sales.Quantity \* Sales.Price) desc

