**Домашняя работа 2**

Основы языка SQL

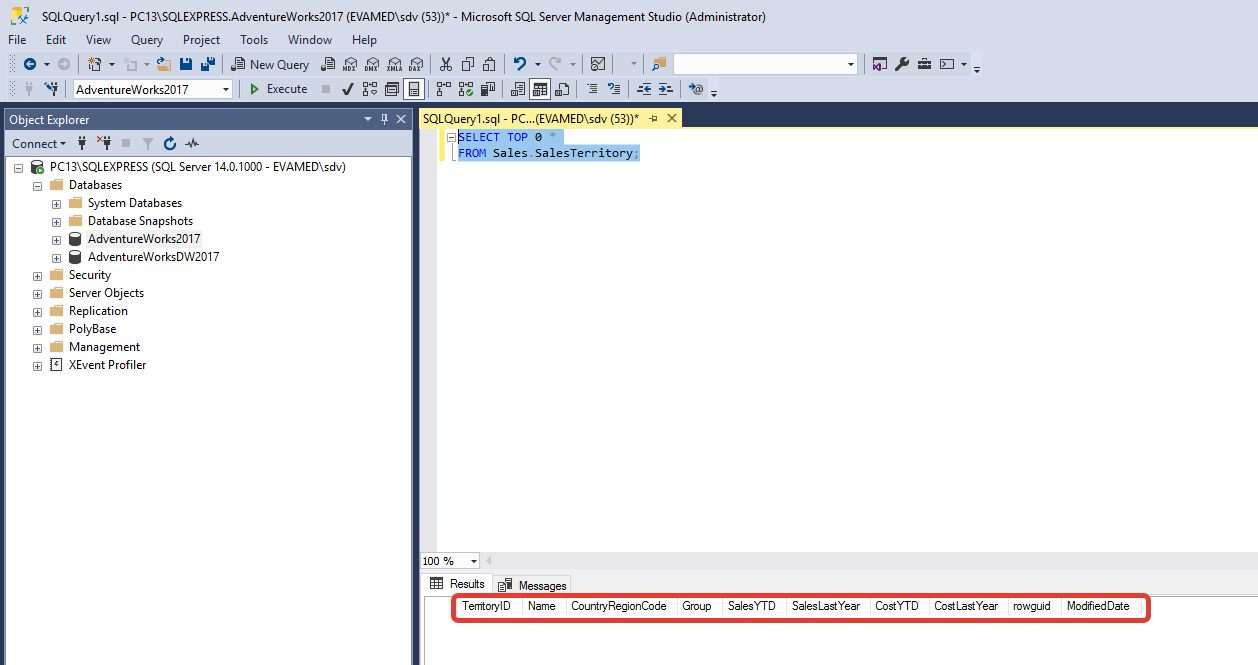
1. Решите на базе данных AdventureWorks2017 следующие задачи:
2. Извлечь все столбцы из таблицы Sales.SalesTerritory.

Чтобы извлечь названия всех столбцов, без строк:

SELECT TOP 0 \*

FROM Sales.SalesTerritory;

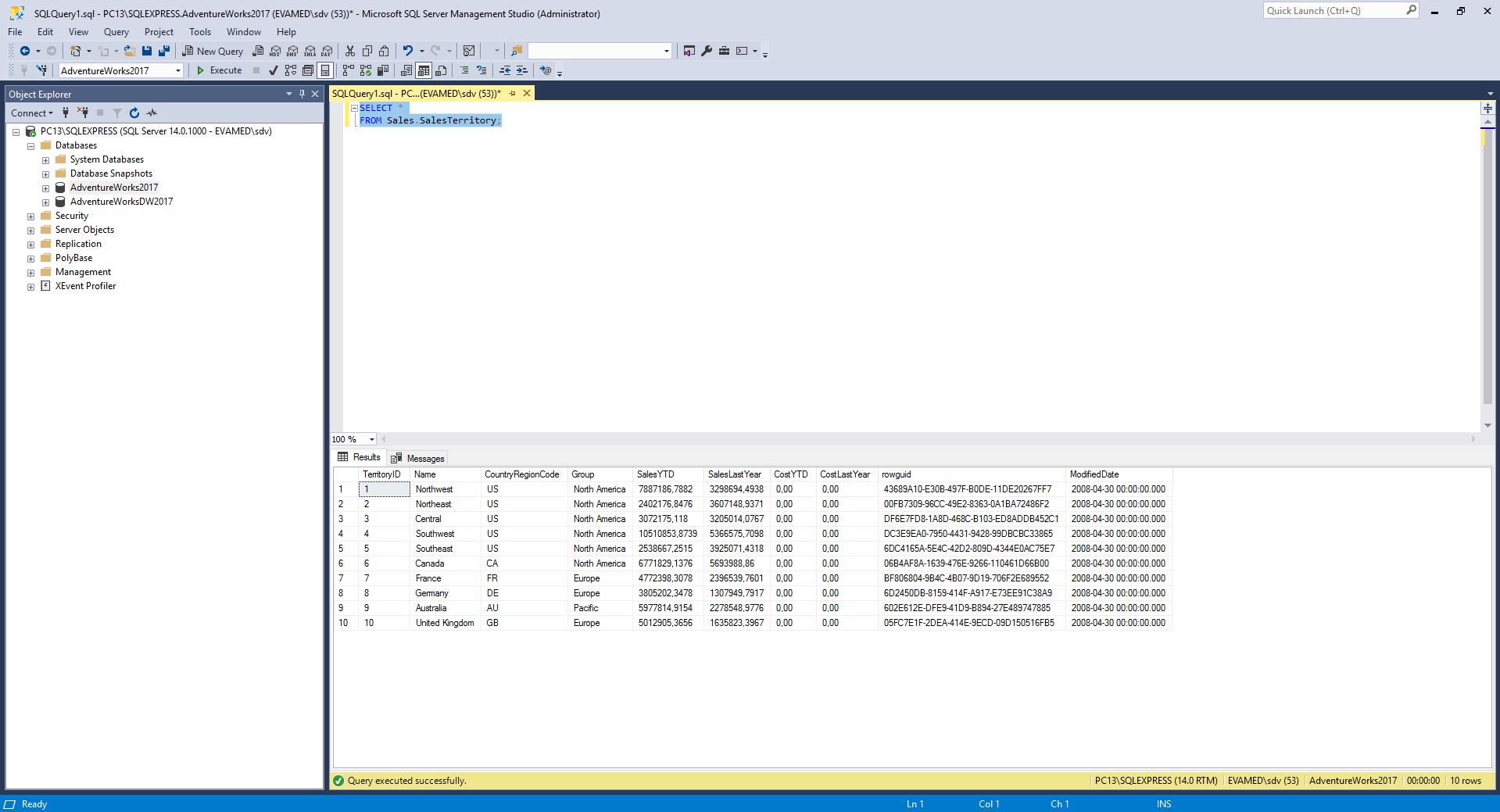
TerritoryID Name CountryRegionCode Group SalesYTD SalesLastYear CostYTD CostLastYear rowguid ModifiedDate



Чтобы извлечь все данные из всех столбцов:

SELECT \*

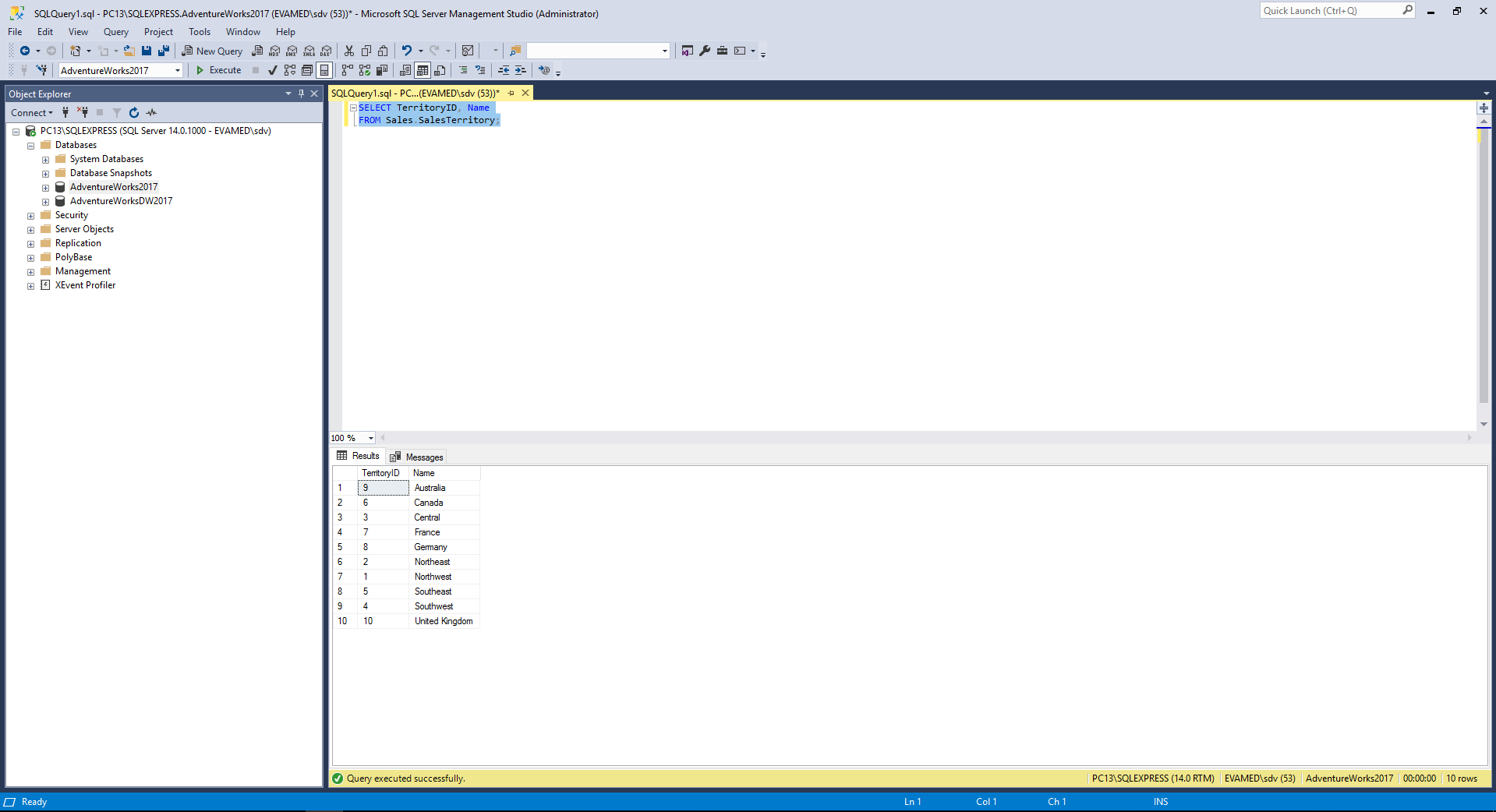
FROM Sales.SalesTerritory;



1. Извлечь столбцы TerritoryID и Name из таблицы Sales.SalesTerritory.

SELECT TerritoryID, Name

FROM Sales.SalesTerritory;

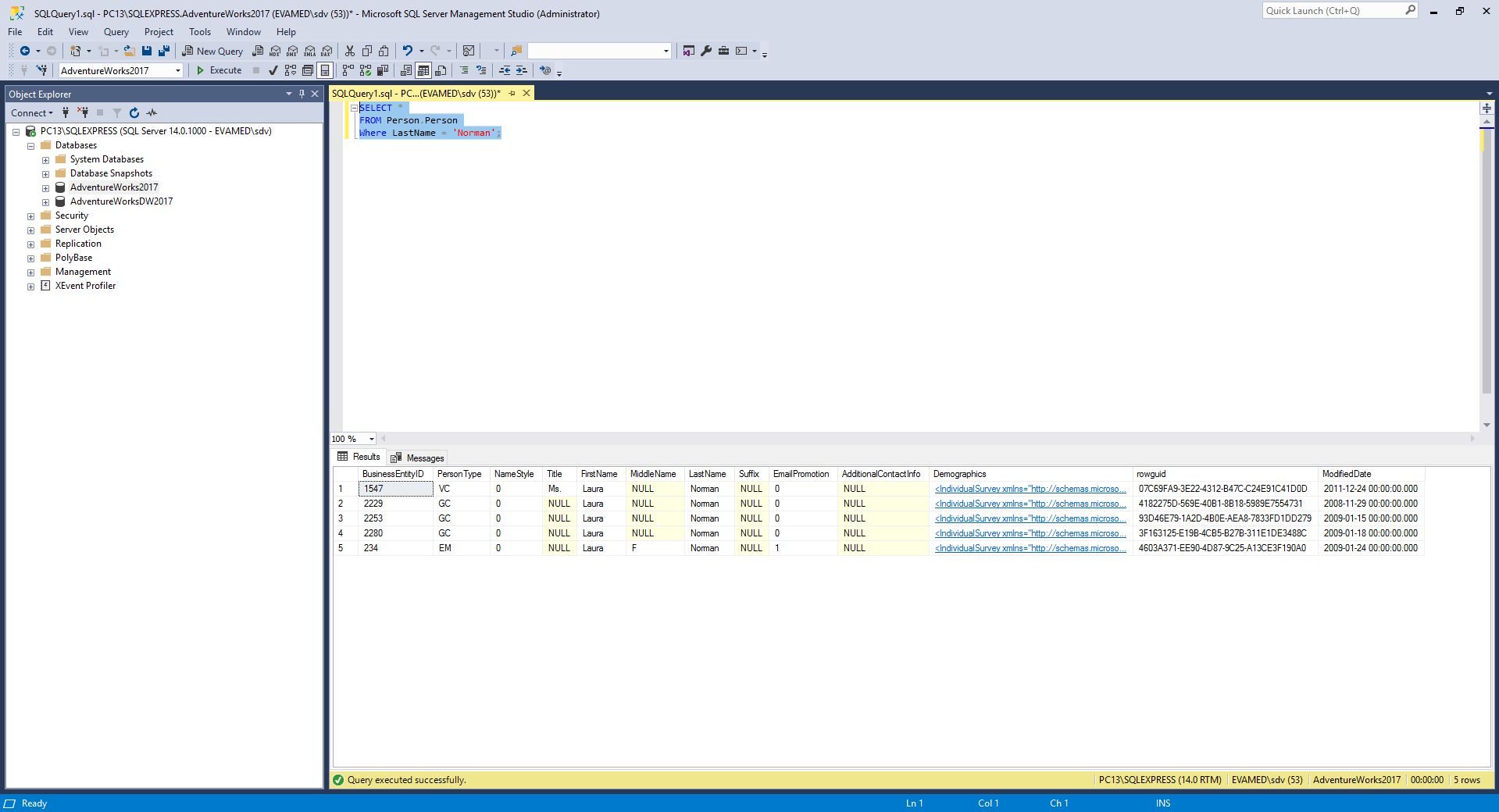


1. Найдите все данные, которые существует для людей из Person.Person с LastName = ‘Norman’.

SELECT \*

FROM Person.Person

Where LastName = 'Norman';

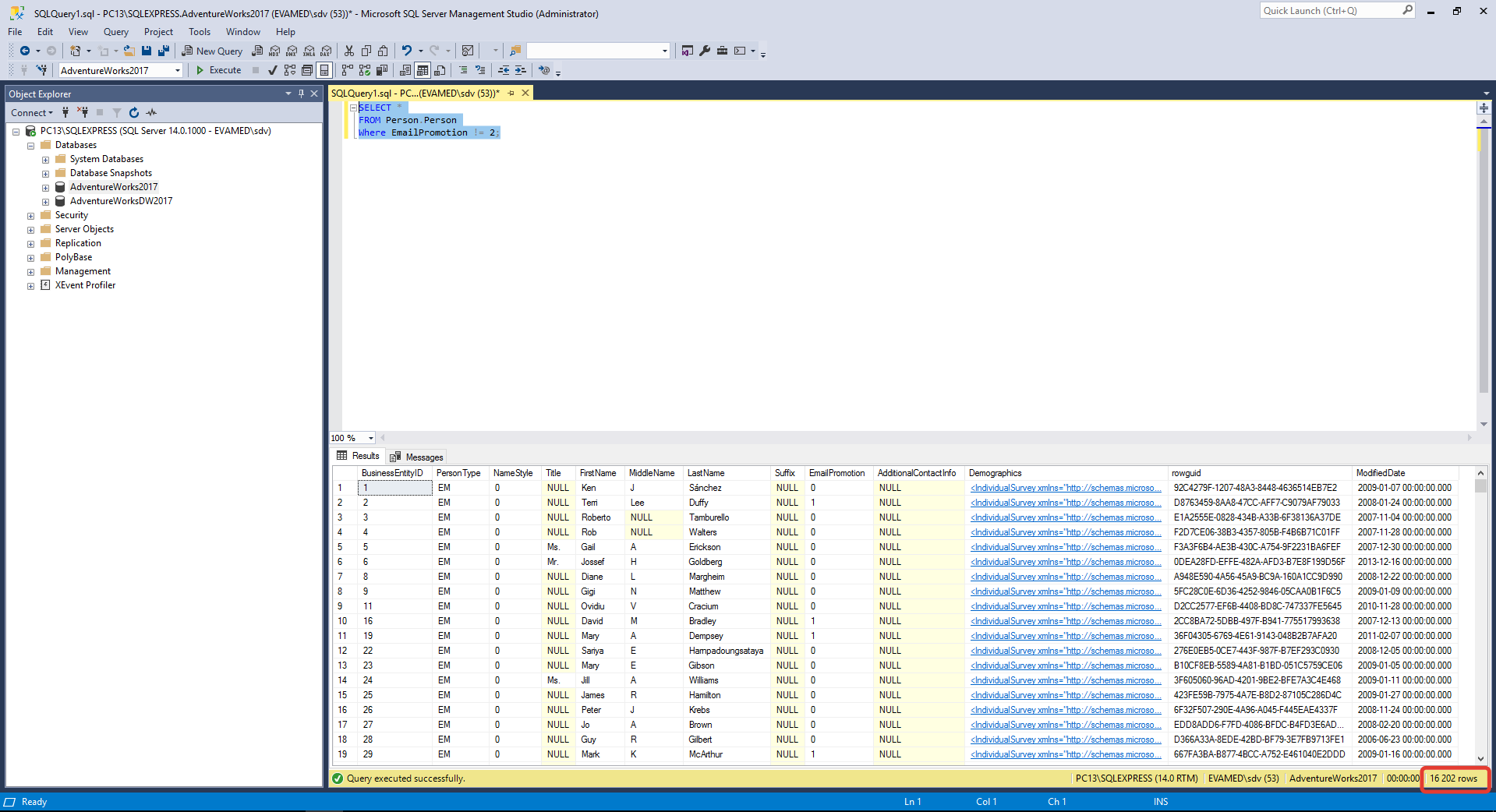


1. Найдите все строки из Person.Person, где EmailPromotion не равен 2.

SELECT \*

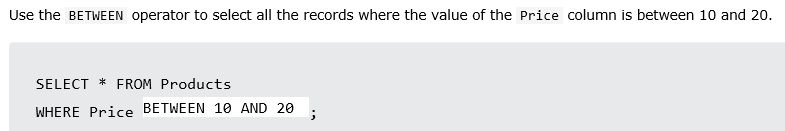
FROM Person.Person

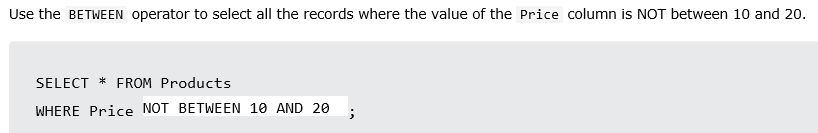
Where EmailPromotion != 2;

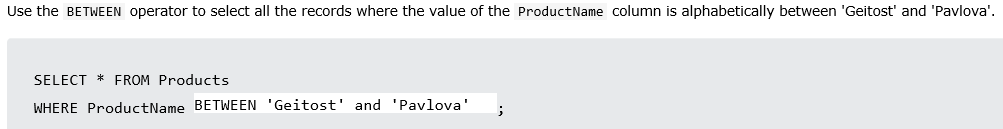


1. Повторите изученный материал. Обратите внимание, что на ссылках синтаксис указан для всех СУБД, нам надо только для Sql Server.

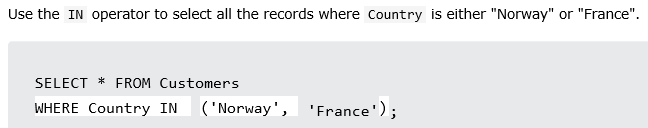
* [BETWEEN](https://www.w3schools.com/sql/sql_between.asp)

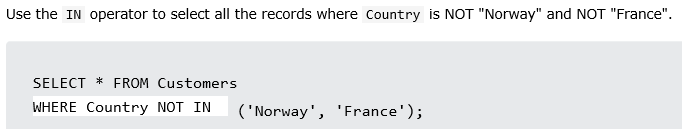




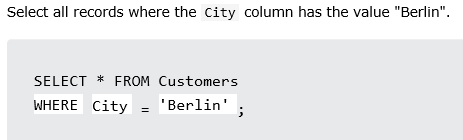


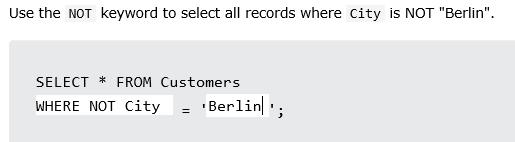
* [IN](https://www.w3schools.com/sql/sql_in.asp)

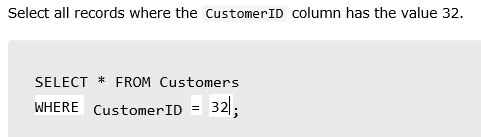


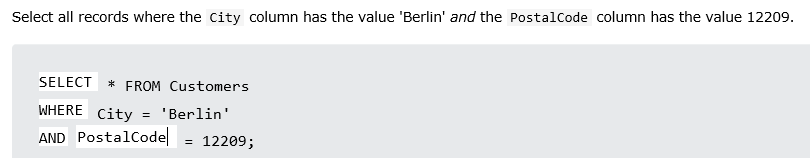


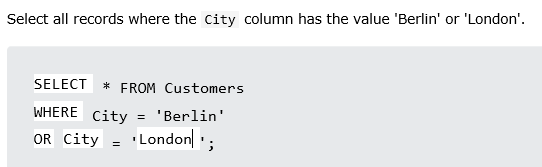
* [AND, OR, NOT](https://www.w3schools.com/sql/sql_and_or.asp)



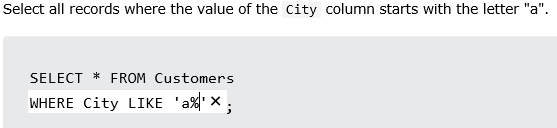


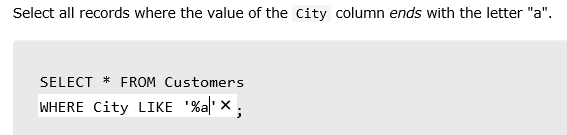


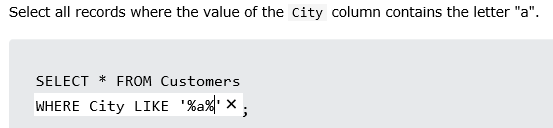


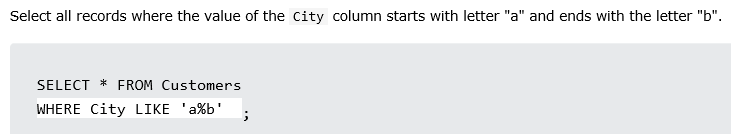


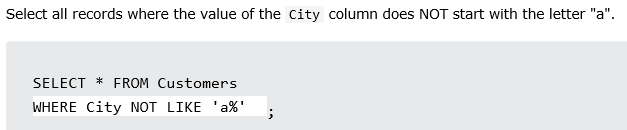
* [LIKE](https://www.w3schools.com/sql/sql_like.asp%20%20)



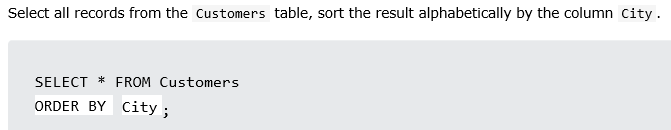


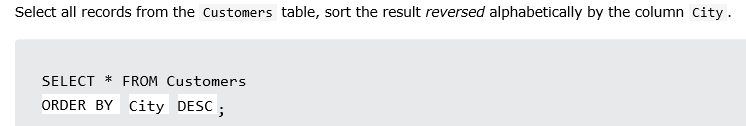






* [ORDER BY](https://www.w3schools.com/sql/sql_orderby.asp)



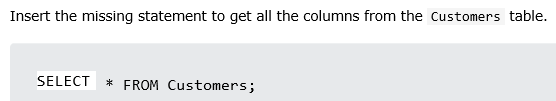


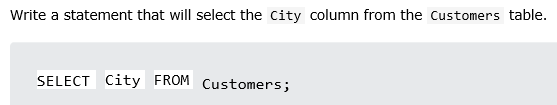
* [TOP](https://www.w3schools.com/sql/sql_top.asp%20)

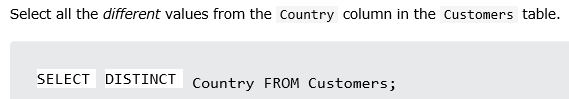




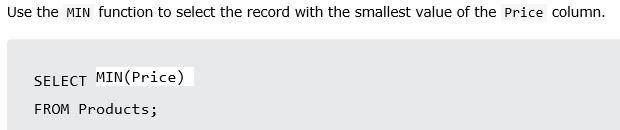
* [DISTINCT](https://www.w3schools.com/sql/sql_distinct.asp)

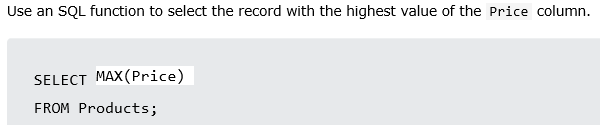




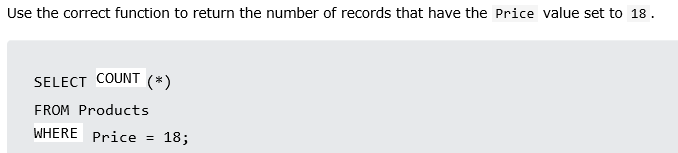


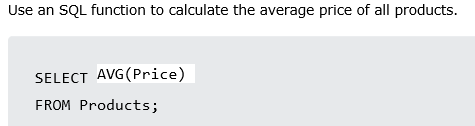
* [MIN, MAX](https://www.w3schools.com/sql/sql_min_max.asp)

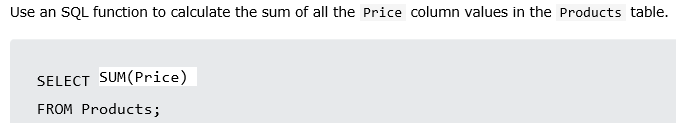




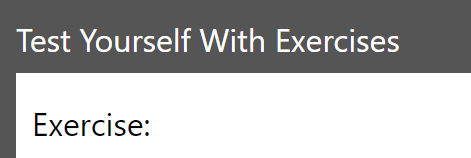
* [COUNT, AVG, SUM](https://www.w3schools.com/sql/sql_count_avg_sum.asp)







Выполните после каждого раздела задания:



1. На [официальном сайте Microsoft](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/aggregate-functions-transact-sql?view=sql-server-ver15) ещё раз просмотрите синтаксис SUM, AVG, COUNT, MIN, MAX и примеры для каждой функции. Какие ещё агрегатные функции существуют в языке T-SQL? Приведите несколько примеров.

Transact-SQL предоставляет следующие агрегатные функции.

* [APPROX\_COUNT\_DISTINCT](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/approx-count-distinct-transact-sql?view=sql-server-ver15)

Функция APPROX\_COUNT\_DISTINCT( expression ) вычисляет выражение для каждой строки в группе и возвращает приблизительное количество уникальных значений, не равных NULL, в группе. Эта функция предназначена для агрегирования в больших наборах данных, для которых скорость реагирования намного важнее абсолютной точности.

SELECT APPROX\_COUNT\_DISTINCT(O\_OrderKey) AS Approx\_Distinct\_OrderKey

FROM dbo.Orders;

Approx\_Distinct\_OrderKey

------------------------

15164704

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

Эта функция возвращает среднее арифметическое группы значений. Значения NULL она не учитывает.

* [CHECKSUM\_AGG](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/checksum-agg-transact-sql?view=sql-server-ver15)
* [COUNT](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/count-transact-sql?view=sql-server-ver15)
* [COUNT\_BIG](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/count-big-transact-sql?view=sql-server-ver15)
* [GROUPING](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/grouping-transact-sql?view=sql-server-ver15)
* [GROUPING\_ID](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/grouping-id-transact-sql?view=sql-server-ver15)
* [MAX](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/max-transact-sql?view=sql-server-ver15)
* [MIN](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/min-transact-sql?view=sql-server-ver15)
* [STDEV](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/stdev-transact-sql?view=sql-server-ver15)
* [STDEVP](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/stdevp-transact-sql?view=sql-server-ver15)
* [STRING\_AGG](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/string-agg-transact-sql?view=sql-server-ver15)
* [SUM](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/sum-transact-sql?view=sql-server-ver15)
* [VAR](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/var-transact-sql?view=sql-server-ver15)
* [VARP](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/varp-transact-sql?view=sql-server-ver15)

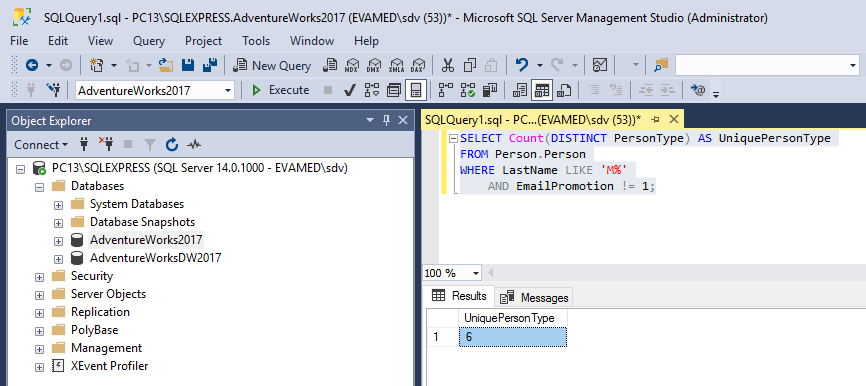
1. Решите на базе данных AdventureWorks2017 следующие задачи:
2. Сколько уникальных PersonType существует для людей из Person.Person с LastName начинающиеся с буквы М или **не** содержащий 1 в EmailPromotion.

SELECT Count(DISTINCT PersonType) AS UniquePersonType

FROM Person.Person

WHERE LastName LIKE 'M%'

AND EmailPromotion != 1;



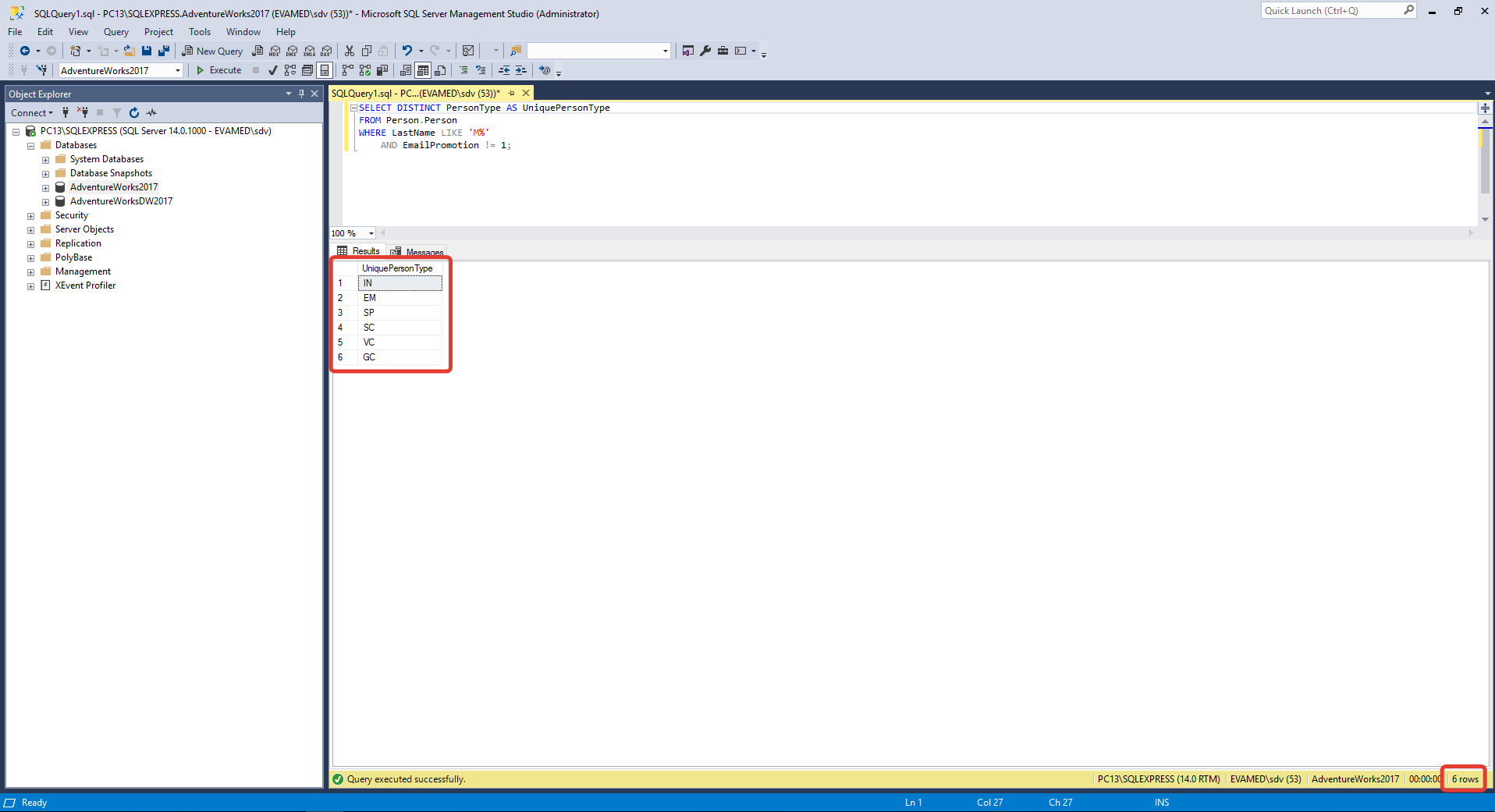
Для проверки:

SELECT DISTINCT PersonType AS UniquePersonType

FROM Person.Person

WHERE LastName LIKE 'M%'

AND EmailPromotion != 1;



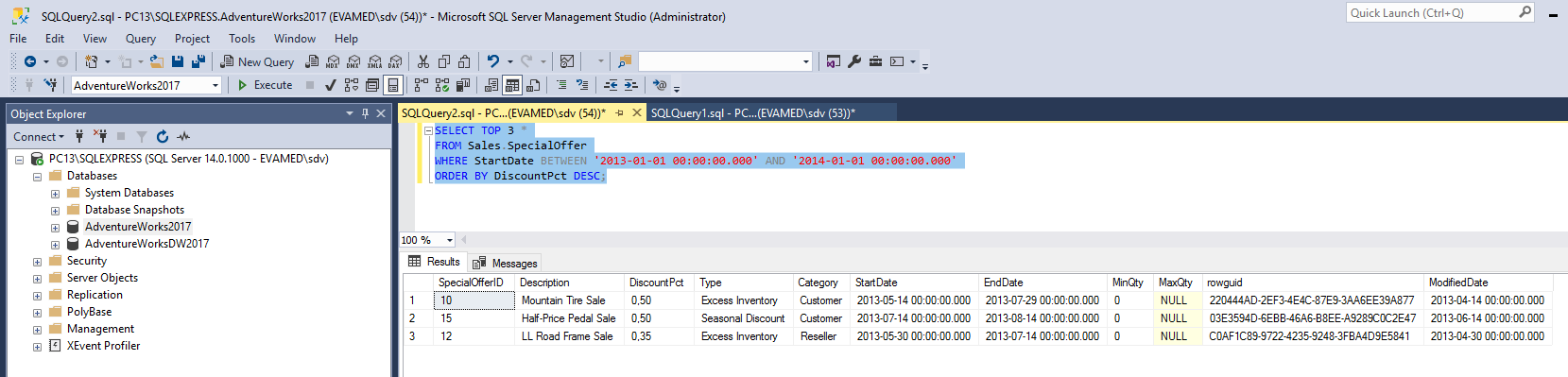
1. Вывести первых 3 специальных предложений из Sales.SpecialOffer с наибольшими DiscountPct, которые начинали действовать с 2013-01-01 по 2014-01-01.

SELECT TOP 3 \*

FROM Sales.SpecialOffer

WHERE StartDate BETWEEN '2013-01-01 00:00:00.000' AND '2014-01-01 00:00:00.000'

ORDER BY DiscountPct DESC;

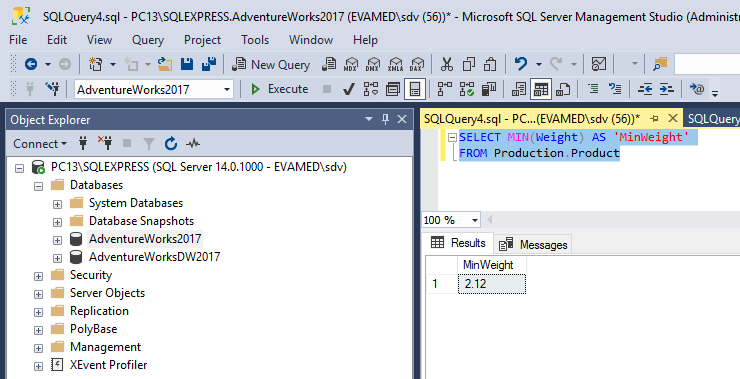


1. Найти самый минимальный вес и самый максимальный размер продукта из Production.Product.

Минимальный вес:

SELECT MIN(Weight) AS 'MinWeight'

FROM Production.Product



SELECT ProductID, Name, Weight AS 'MinWeight'

FROM Production.Product

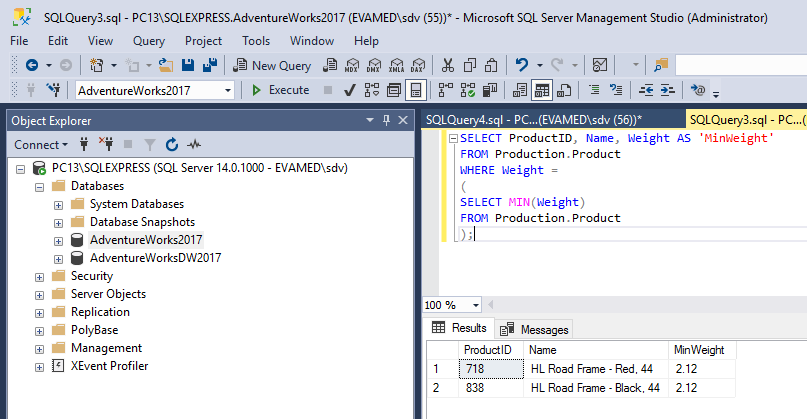
WHERE Weight =

(

SELECT MIN(Weight)

FROM Production.Product

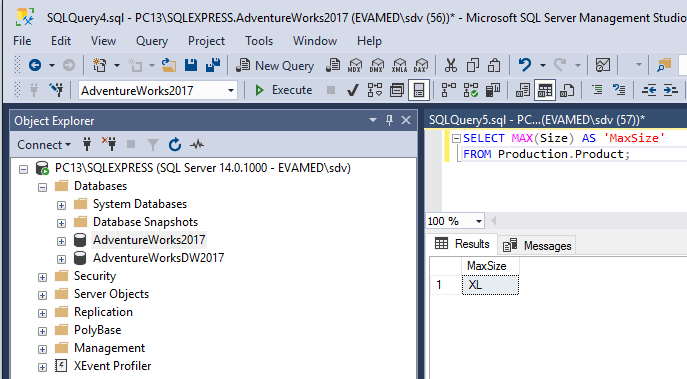
);



Максимальный размер:

SELECT MAX(Size) AS 'MaxSize'

FROM Production.Product;



SELECT ProductID, Name, Weight AS 'MaxSize'

FROM Production.Product

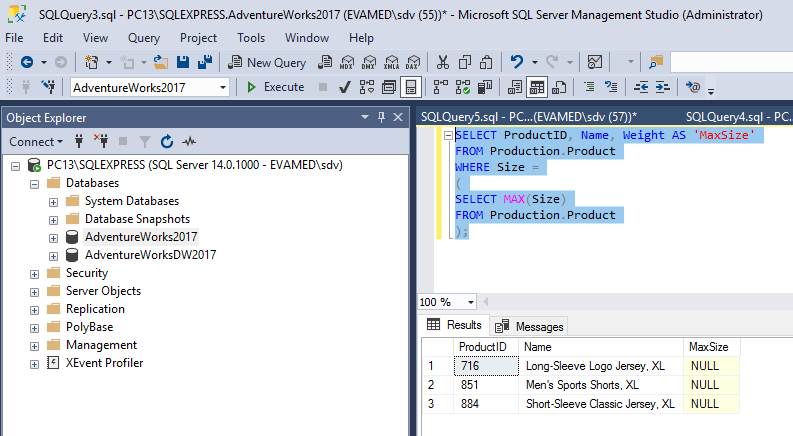
WHERE Size =

(

SELECT MAX(Size)

FROM Production.Product

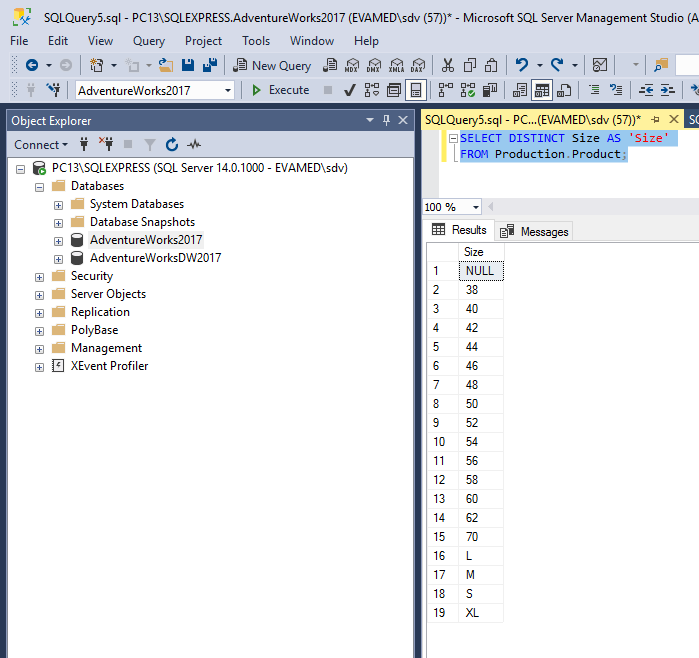
);



Какие есть варианты размеров. Для проверки:

SELECT DISTINCT Size AS 'Size'

FROM Production.Product;



1. Найти самый минимальный вес и самый максимальный размер продукта для каждой подкатегории ProductSubcategoryID из Production.Product.

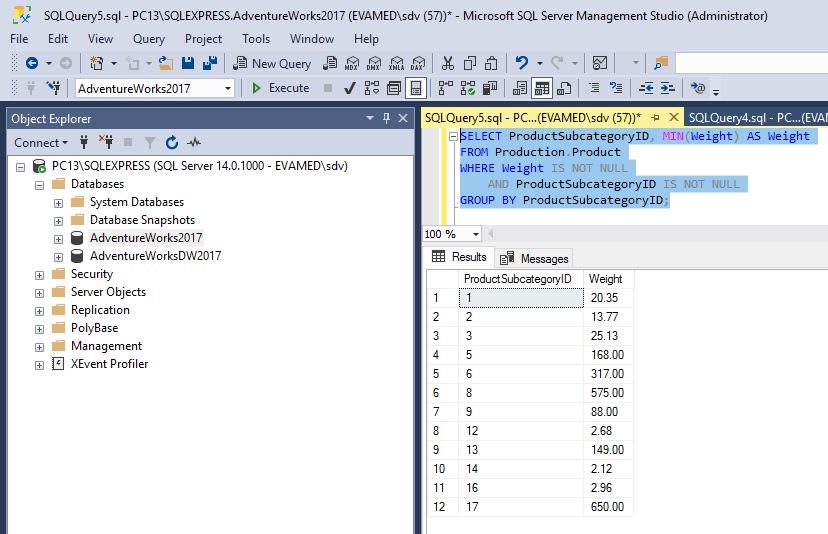
SELECT ProductSubcategoryID, MIN(Weight) AS Weight

FROM Production.Product

WHERE Weight IS NOT NULL

AND ProductSubcategoryID IS NOT NULL

GROUP BY ProductSubcategoryID;



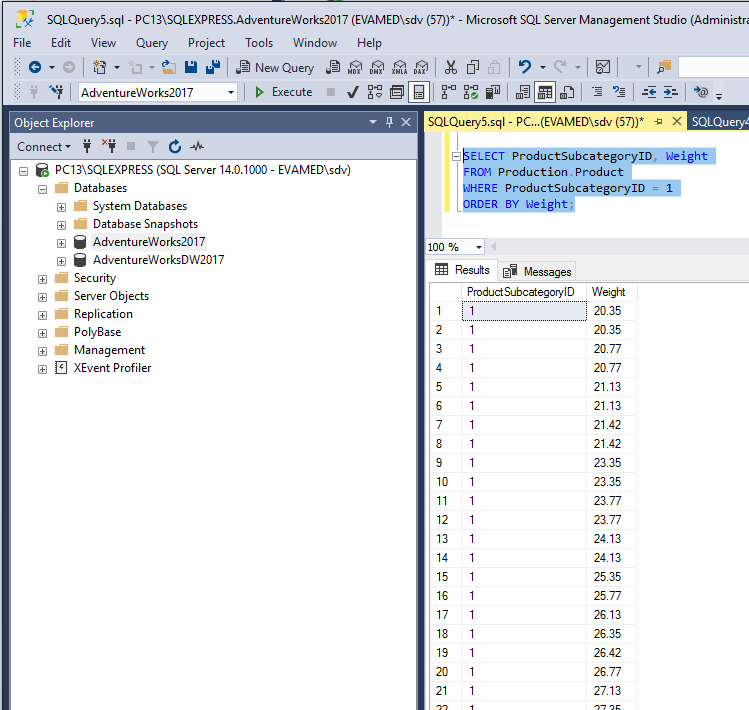
Для проверки:

SELECT ProductSubcategoryID, Weight

FROM Production.Product

WHERE ProductSubcategoryID = 1

ORDER BY Weight;



С цифровыми значениями результаты правильные. С буквенными размерами результаты неправильные.

SELECT ProductSubcategoryID, MAX(Size) AS MaxSize

FROM Production.Product

WHERE Size IS NOT NULL

AND ProductSubcategoryID IS NOT NULL

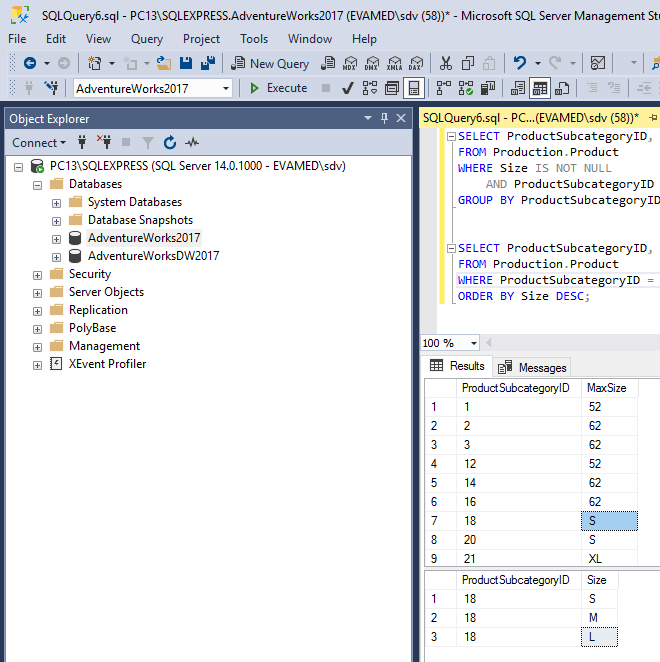
GROUP BY ProductSubcategoryID;

SELECT ProductSubcategoryID, Size

FROM Production.Product

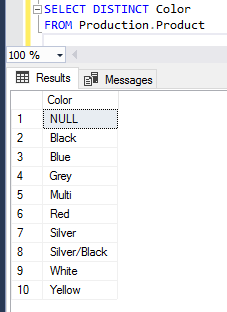
WHERE ProductSubcategoryID = 18

ORDER BY Size DESC;



1. Найти самый минимальный вес и самый максимальный размер продукта для каждой подкатегории ProductSubcategoryID из Production.Product, где цвет продукта определен(Color).

Какие варианты цветов есть:



Для буквенных размеров не подходит.

SELECT ProductSubcategoryID, MAX(Size) AS MaxSize

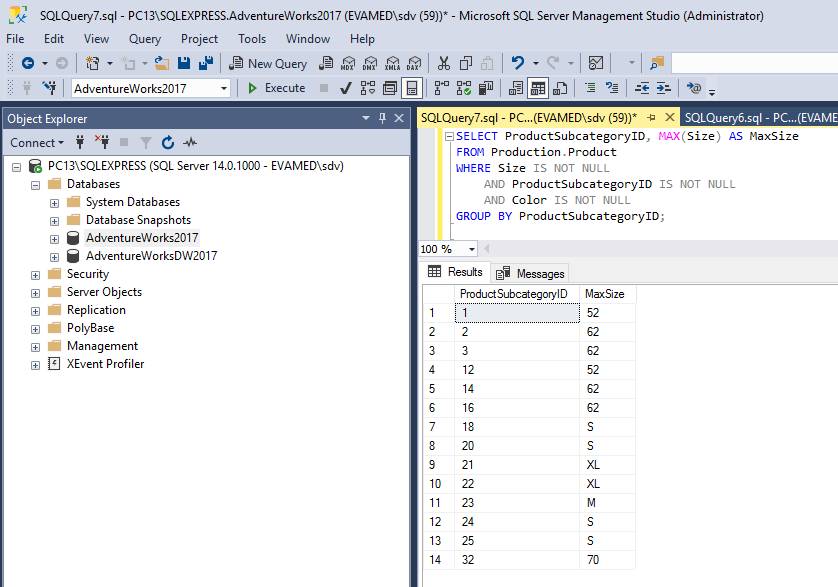
FROM Production.Product

WHERE Size IS NOT NULL

AND ProductSubcategoryID IS NOT NULL

AND Color IS NOT NULL

GROUP BY ProductSubcategoryID;



SELECT ProductSubcategoryID, MIN(Weight) AS 'MaxWeight'

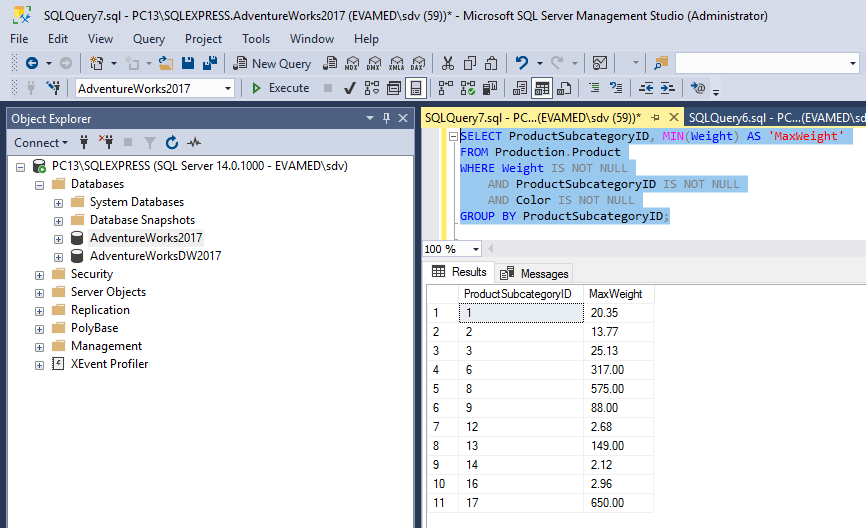
FROM Production.Product

WHERE Weight IS NOT NULL

AND ProductSubcategoryID IS NOT NULL

AND Color IS NOT NULL

GROUP BY ProductSubcategoryID;



1. Создайте doc файл (отчет), туда поместите ответы для заданий 1, 3, 4. Файл сохраните в свой репозиторий в папку Lesson2.