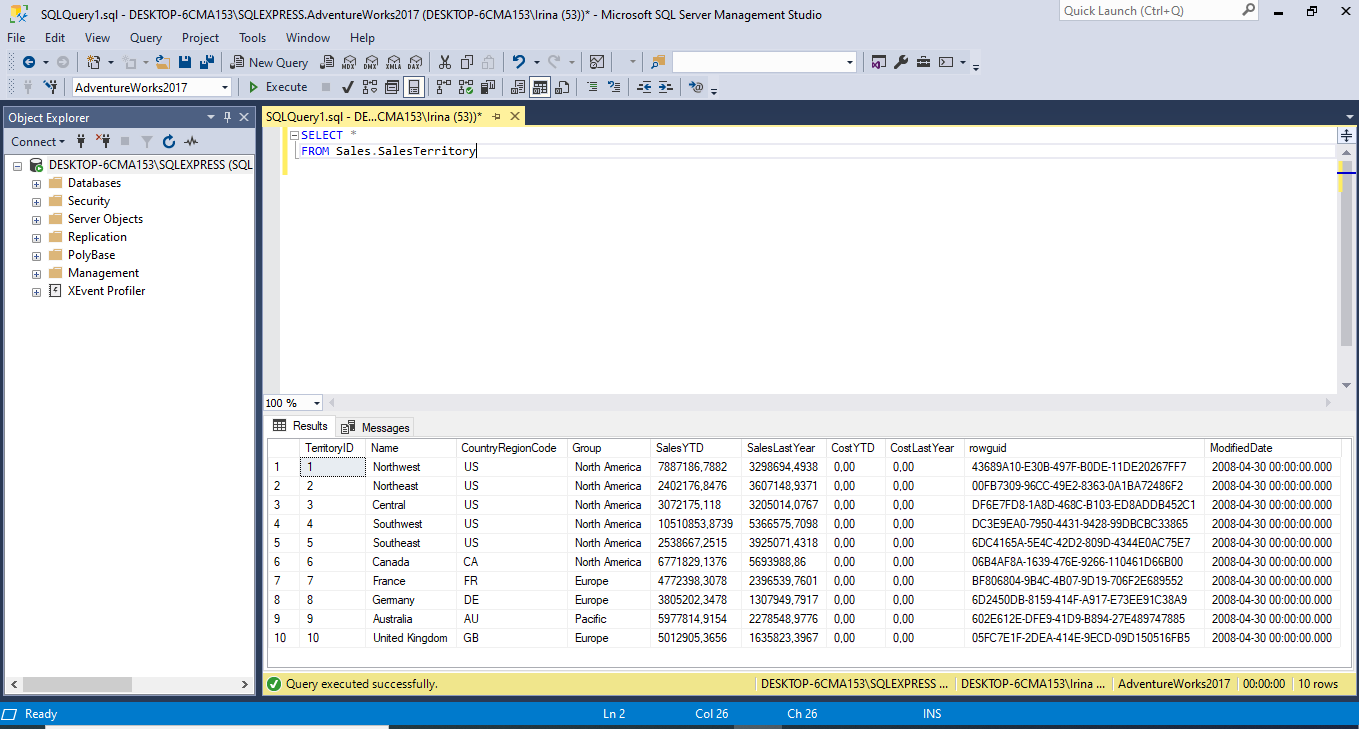
**Домашняя работа 2**

Основы языка SQL

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

SELECT \*

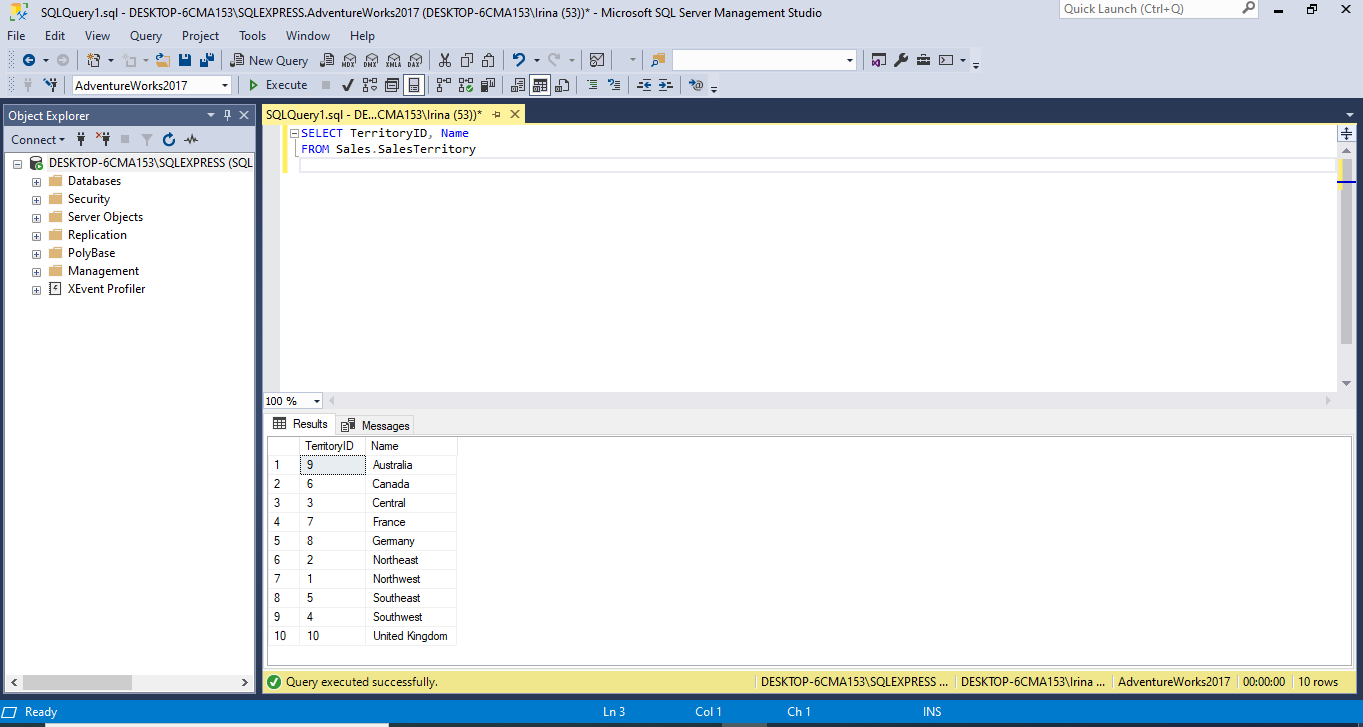
FROM Sales.SalesTerritory



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

SELECT TerritoryID, Name

FROM Sales.SalesTerritory

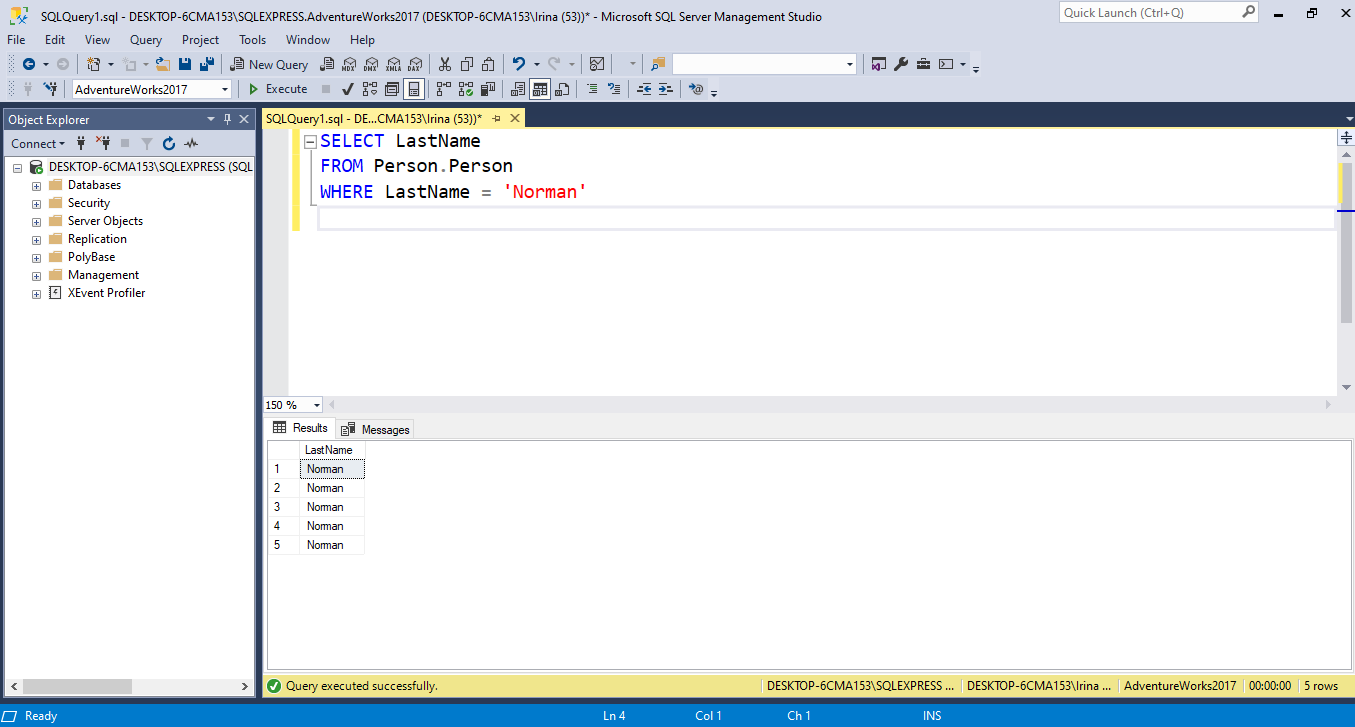


1. Найдите все данные, которые существует для людей из Person.Person с LastName = ‘Norman’.

SELECT LastName

FROM Person.Person

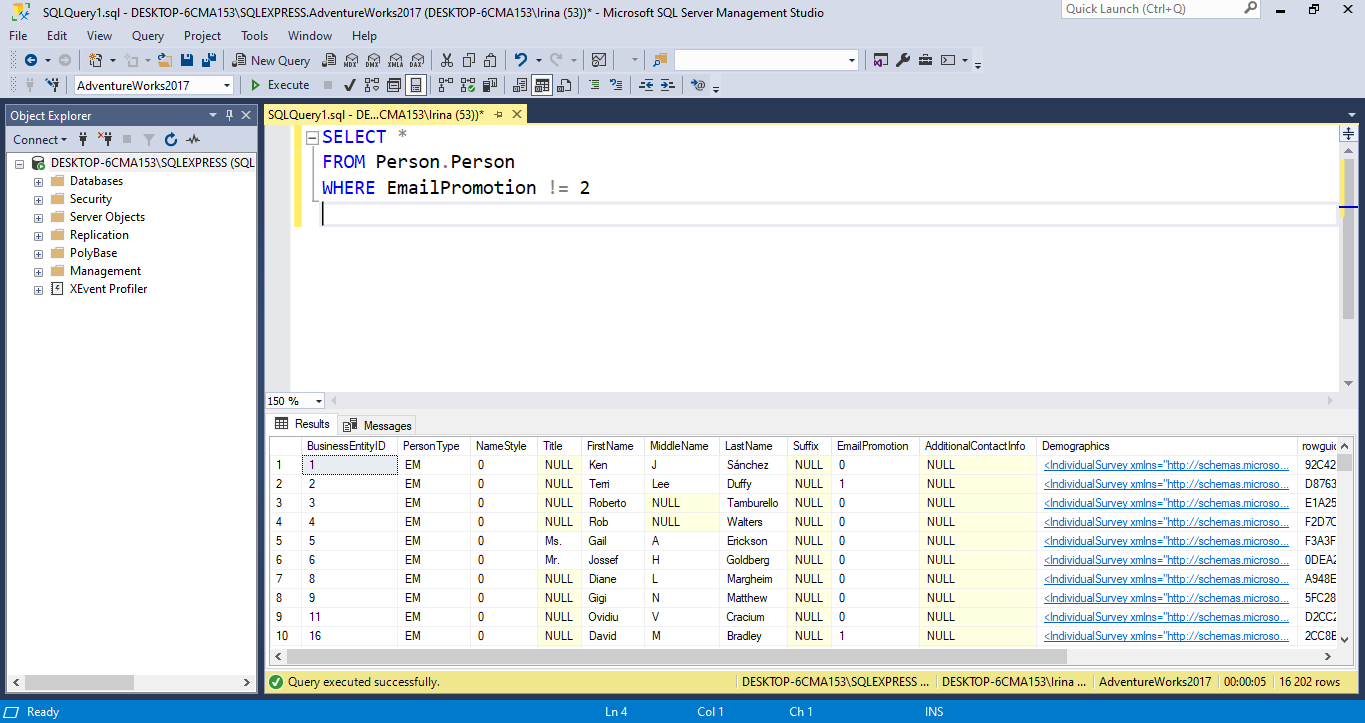
WHERE LastName = 'Norman'



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

FROM Person.Person

WHERE EmailPromotion != 2

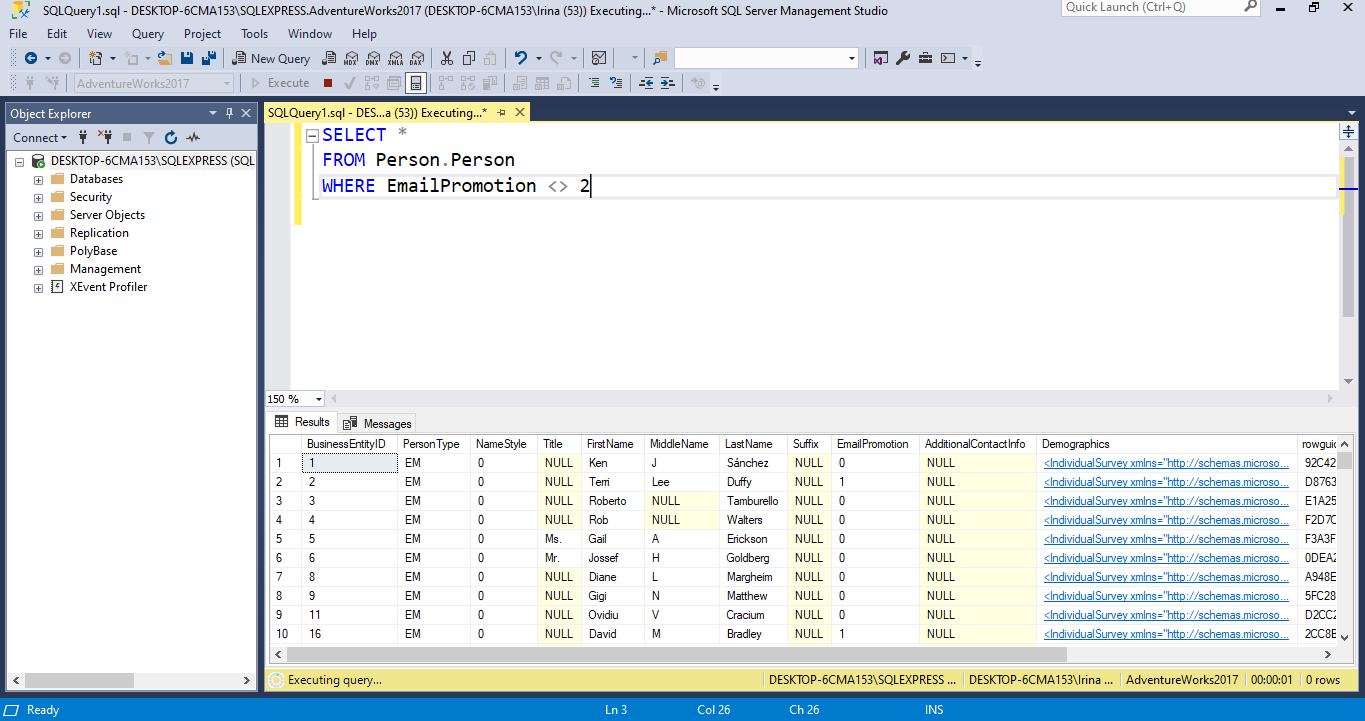


Или

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) - <https://www.w3schools.com/sql/sql_between.asp>

Оператор SQL BETWEEN – это логический оператор, который определяет диапазон для проверки. Другими словами, с помощью BETWEEN мы можем проверить, входит ли значение в определённый диапазон. Значениями могут быть числа, текст или даты. Начальные и конечные значения включены.

Упражнение 1

С помощью оператора выберите все записи, в которых значение столбца находится в диапазоне от 10 до 20. BETWEENPrice.SELECT \*

FROM Products

WHERE Price BETWEEN 10 AND 20;

Упражнение 2

С помощью оператора выберите все записи, в которых значение столбца НЕ находится в диапазоне от 10 до 20. BETWEENPrice.

SELECT \*

FROM Products

WHERE Price NOT BETWEEN 10 AND 20;

Упражнение 3

С помощью оператора выберите все записи, где значение столбца находится в алфавитном порядке между 'Geitost' и 'Pavlova'. BETWEENProductName.

SELECT \*

FROM Products

WHERE ProductName

BETWEEN 'Geitost' AND 'Pavlova';

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

Оператор SQL IN – это логический оператор, который проверяет, соответствует ли выражение какому-либо значению в списке значений.

Упражнение 1:

С помощью оператора выберите все записи, где находится "Норвегия" или "Франция". INCountry.

SELECT \*

FROM Customers

WHERE Country IN ('Norway', 'France');

Упражнение 2:

Используйте оператор, чтобы выбрать все записи, где НЕ "Норвегия" и НЕ "Франция". INCountry.

SELECT \*

FROM Customers

WHERE Country NOT IN ('Norway', 'France');

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

Оператор AND (И) – это логический оператор, который объединяет два или более логических выражения и возвращает true (истина), только если оба выражения имеют значение true. Оператор AND возвращает false (ложь), если одно из двух выражений оценивается как false.

Операторы AND и OR используются для фильтрации записей, основанных на более чем одном условии:

Оператор AND отображает запись, если все условия разделены и имеют значение true.

Оператор OR отображает запись, если какое-либо из условий разделено или имеет значение true.

Оператор NOT отображает запись, если условие не соответствует действительности.

Упражнение 1:

Выберите все записи, где столбец имеет значение 'Berlin', а столбец имеет значение 12209. City PostalCode.

SELECT \*

FROM Customers

WHERE City = 'Berlin' AND PostalCode = 12209;

Упражнение 2:

Выберите все записи, в столбце которых есть значение "Berlin" или "London".City

SELECT

\* FROM Customers

WHERE City = 'Berlin' OR City = 'London';

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

Оператор SQL LIKE – это стандартный оператор для сравнения значения столбца с другим значением столбца или строкой в кавычках.

Существует два подстановочных знака, часто используемых в сочетании с оператором LIKE

Знак процента (%) представляет ноль, один или несколько символов.

Знак подчеркивания (\_) представляет собой один символ.

Упражнение 1:

Выберите все записи, где значение столбца начинается с буквы "a". City.

SELECT \*

FROM Customers

WHERE City LIKE 'a%';

Упражнение 2:

Выберите все записи, где значение столбца заканчивается буквой "a". City.

SELECT \*

FROM Customers

WHERE City LIKE '%a';

Упражнение 3:

Выберите все записи, где значение столбца содержит букву "a". City.

SELECT \*

FROM Customers

WHERE City LIKE '%a%';

Упражнение 4:

Выберите все записи, где значение столбца начинается с буквы "a" и заканчивается буквой "b". City.

SELECT \* FROM Customers

WHERE City LIKE 'a%b';

Упражнение 5:

Выберите все записи, где значение столбца НЕ начинается с буквы "a". City.

SELECT \* FROM Customers

WHERE City NOT LIKE 'a%';

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

Оператор SQL ORDER BY – необязательное предложение операторов SELECT и UNION, который означает что операторы SELECT, UNION возвращают набор строк, отсортированных по значениям одного или более столбцов. Его можно применять как к числовым столбцам, так и к строковым. В последнем случае, сортировка будет происходить по алфавиту. Использование предложения ORDER BY является единственным способом отсортировать результирующий набор строк.

По умолчанию ключевое слово сортирует записи в порядке возрастания. Чтобы отсортировать записи в порядке убывания, используйте атрибут DESC (ORDER BY DESC).

Упражнение 1:

Выберите все записи из таблицы, отсортируйте результат в алфавитном порядке по столбцу. Customers City

SELECT \*

FROM Customers

ORDER BY City;

Упражнение 2:

Выберите все записи из таблицы, отсортируйте результат в алфавитном порядке по столбцу. Customers City

SELECT \*

FROM Customers

ORDER BY City DESC;

Упражнение 3:

Выберите все записи из таблицы, отсортировать результат в алфавитном порядке, сначала по столбцу, затем, по столбцу. Customers CountryCity

SELECT \*

FROM Customers

ORDER BY Country, City;

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

Оператор SQL TOP - возвращает верхнее число строк в результирующем наборе на основе top\_value. Например, TOP (10) вернет первые 10 строк из полного набора результатов.

Пример 1.

Выбирает первые три записи из таблицы "Customers", где страна — «Германия».

SELECT TOP 3 \*

FROM Customers

WHERE Country='Germany';

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

Оператор SQL DISTINCT - отброс строк дубликатов.

Упражнение 1:

Выберите все различные значения из столбца таблицы. Country Customers

SELECT DISTINCT Country FROM Customers;

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

Функция MIN () возвращает наименьшее значение выбранного столбца. Функция MAX () возвращает наибольшее значение выбранного столбца.

Упражнение:

Используйте функцию для выбора записи с наименьшим значением столбца.MINPrice

SELECT MIN(Price)

FROM Products;

Упражнение:

Используйте функцию SQL для выбора записи с наибольшим значением столбца.Price

SELECT MAX(Price)

FROM Products;

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

Функция возвращает количество строк, соответствующих заданному условию.COUNT()

Функция возвращает среднее значение числового столбца. AVG()

Функция возвращает общую сумму числового столбца. SUM()

Следует различать два случая применения агрегатных функций. Первый: агрегатные функции используются сами по себе и возвращают одно результирующее значение. Второй: агрегатные функции используются с оператором SQL GROUP BY, то есть с группировкой по полям (столбцам) для получения результирующих значений в каждой группе.

Пример 1

Находит количество продуктов.

SELECT COUNT(ProductID)

FROM Products;

Заметка: Значения NULL не учитываются.

Пример 2

Находит среднюю цену всех продуктов.

SELECT AVG(Price)

FROM Products;

Заметка: Значения NULL игнорируются.

Пример 3

Находит сумму полей "Количество" в таблице "OrderDetails":

SELECT SUM(Quantity)

FROM OrderDetails;

Заметка: Значения NULL игнорируются.

Упражнение 1:

Используйте правильную функцию, чтобы вернуть количество записей, для которых задано значение. Price18

SELECT COUNT(\*)

FROM Products

WHERE Price = 18;

Упражнение 2:

Используйте функцию SQL для расчета средней цены всех продуктов.

SELECT AVG(Price)

FROM Products;

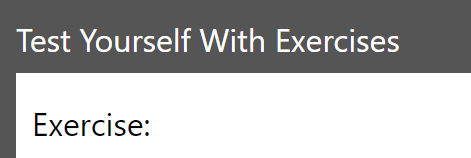
Упражнение 3:

Используйте функцию SQL для вычисления суммы всех значений столбцов в таблице.PriceProducts

SELECT SUM(Price)

FROM Products;

Выполните после каждого раздела задания:



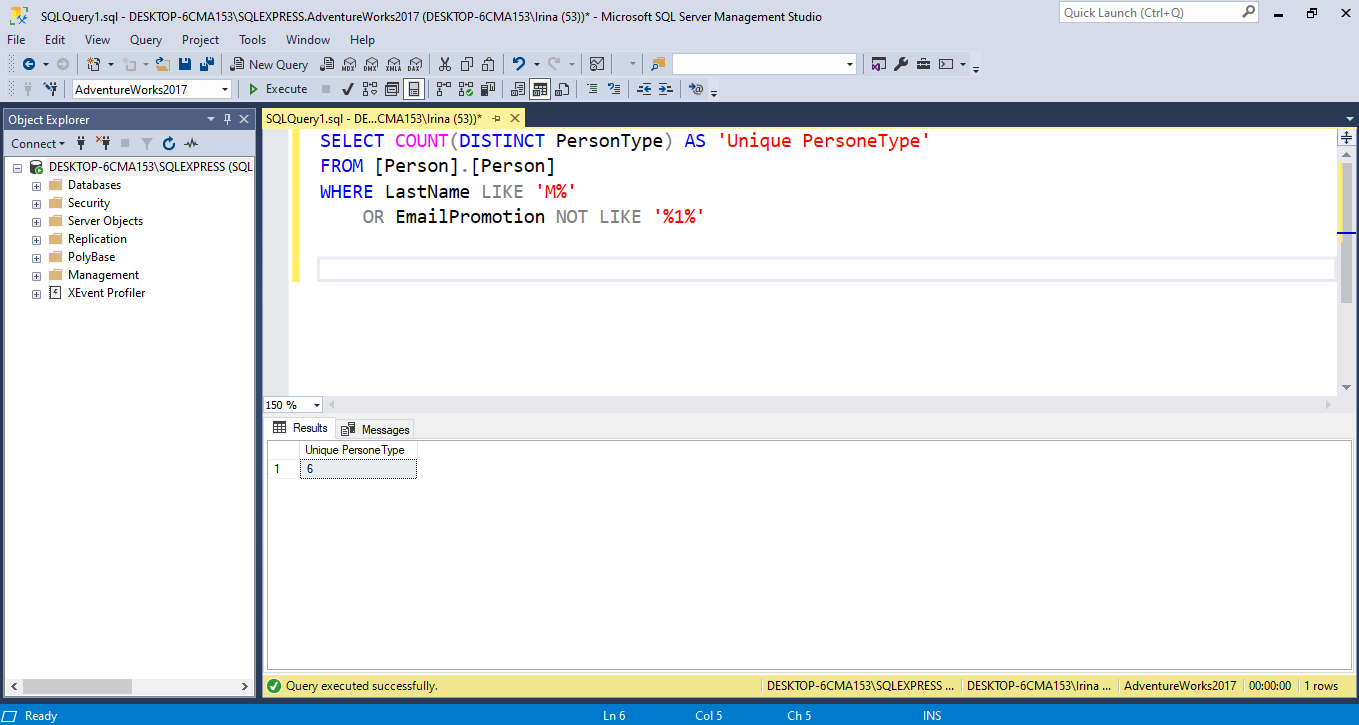
1. На [официальном сайте Microsoft](https://docs.microsoft.com/ru-ru/sql/t-sql/functions/aggregate-functions-transact-sql?view=sql-server-ver15) (https://docs.microsoft.com/ru-ru/sql/t-sql/functions/aggregate-functions-transact-sql?view=sql-server-ver15) ещё раз просмотрите синтаксис SUM, AVG, COUNT, MIN, MAX и примеры для каждой функции. Какие ещё агрегатные функции существуют в языке T-SQL? Приведите несколько примеров. APPROX\_COUNT\_DISTINCT , CHECKSUM\_AGG, GROUPING
2. Решите на базе данных AdventureWorks2017 следующие задачи:
3. 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 'Unique PersoneType'

FROM [Person].[Person]

WHERE LastName LIKE 'M%'

OR EmailPromotion NOT LIKE '%1%'



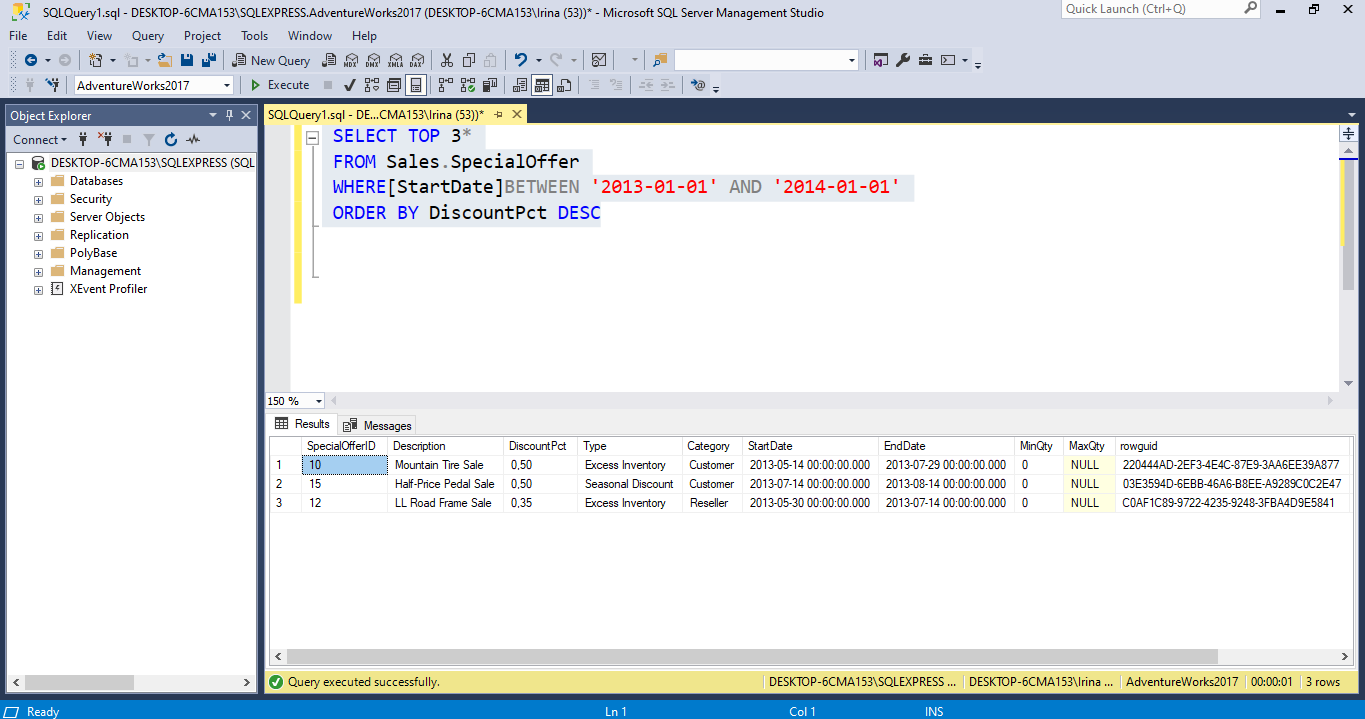
1. Вывести первых 3 специальных предложений из Sales.SpecialOffer с наибольшими DiscountPct, которые начинали действовать с 2013-01-01 по 2014-01-01.

SELECT TOP 3\*

FROM Sales.SpecialOffer

WHERE[StartDate]BETWEEN '2013-01-01' AND '2014-01-01'

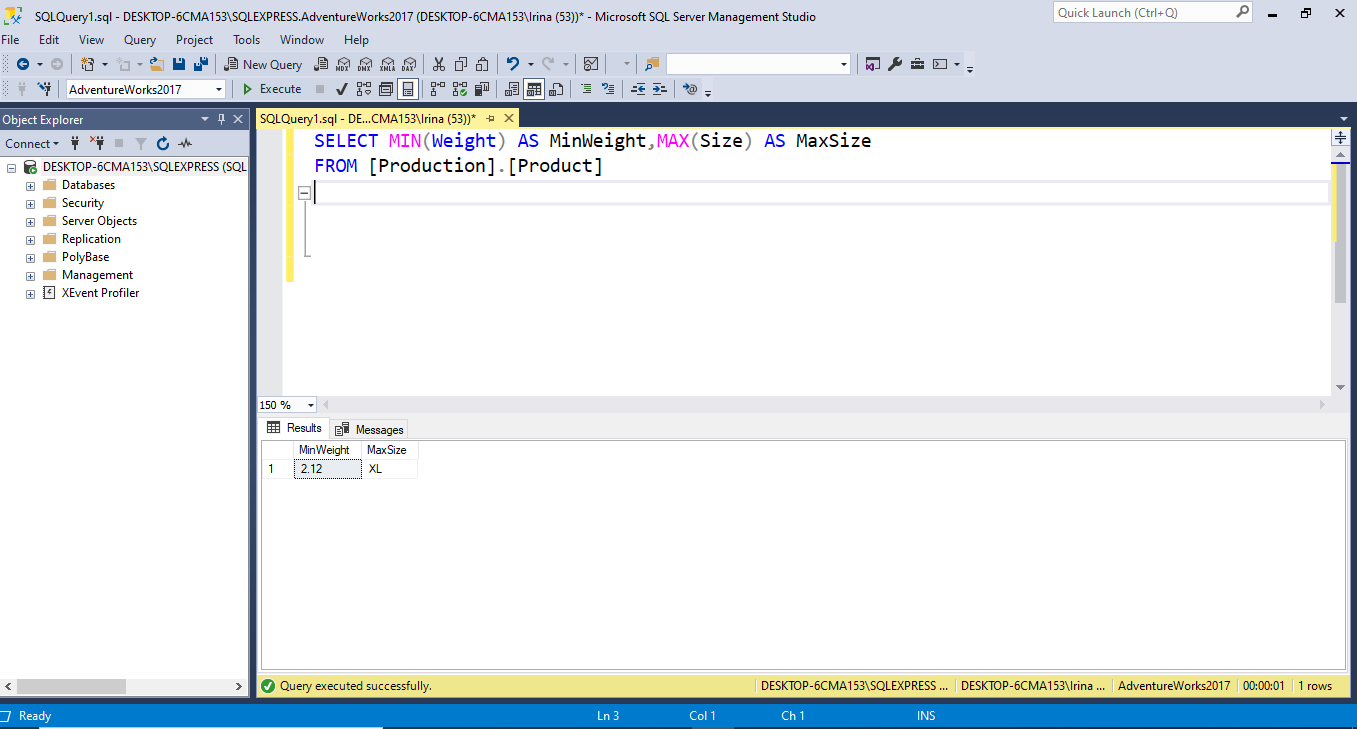
ORDER BY DiscountPct DESC



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

SELECT MIN(Weight) AS MinWeight,MAX(Size) AS MaxSize

FROM [Production].[Product]

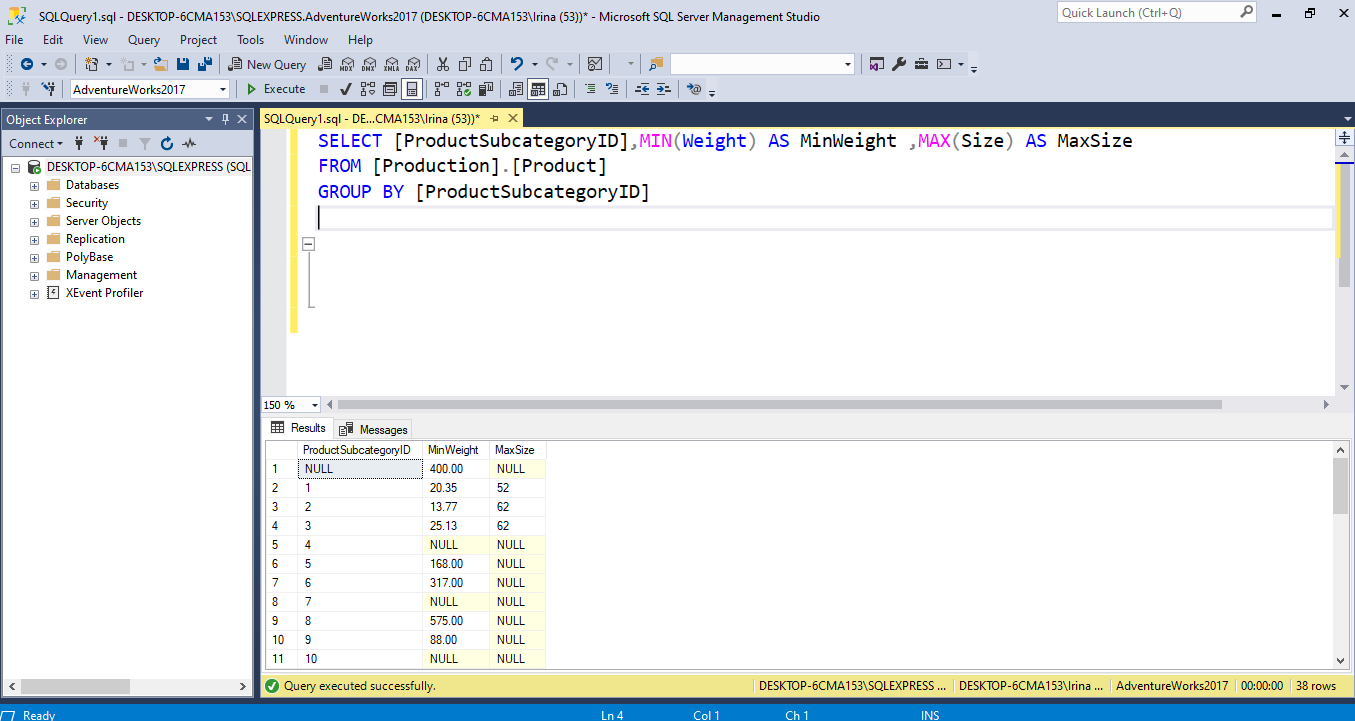


1. Найти самый минимальный вес и самый максимальный размер продукта для каждой подкатегории ProductSubcategoryID из Production.Product.

SELECT [ProductSubcategoryID],MIN(Weight) AS MinWeight ,MAX(Size) AS MaxSize

FROM [Production].[Product]

GROUP BY [ProductSubcategoryID]



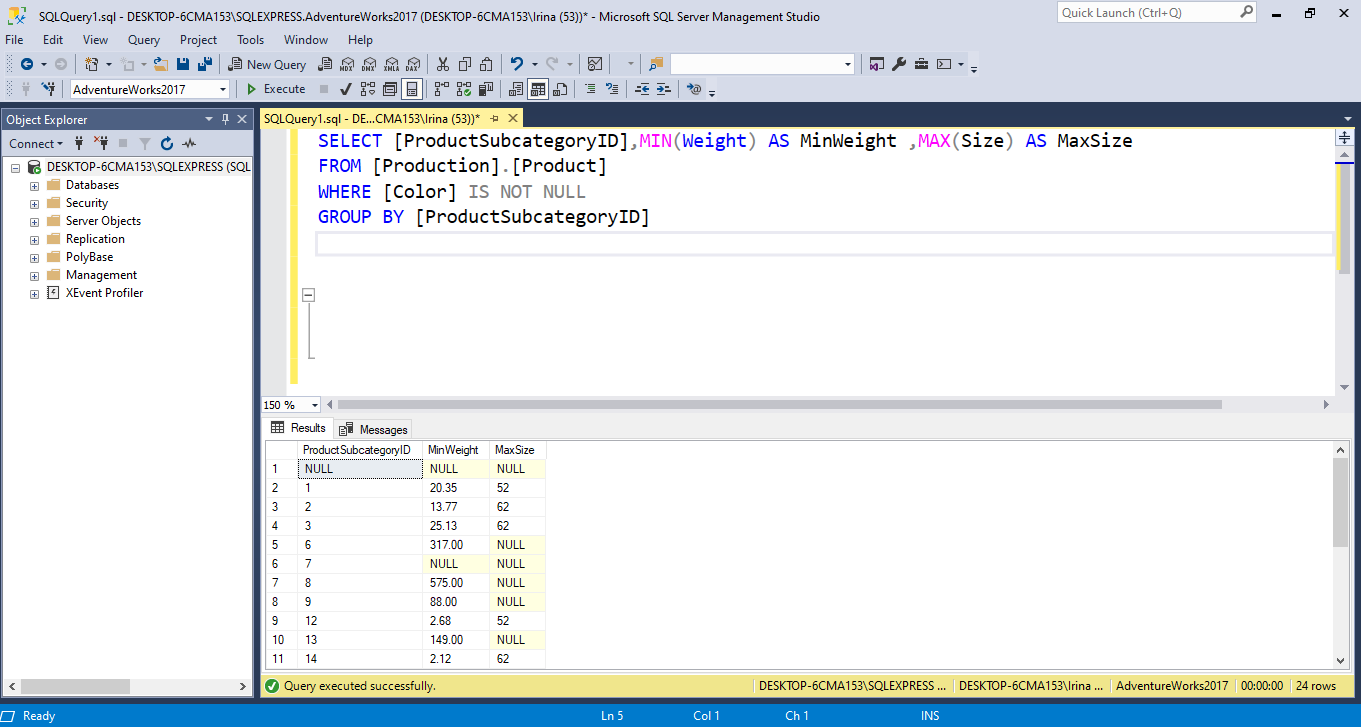
1. Найти самый минимальный вес и самый максимальный размер продукта для каждой подкатегории ProductSubcategoryID из Production.Product, где цвет продукта определен(Color).

SELECT [ProductSubcategoryID],MIN(Weight) AS MinWeight ,MAX(Size) AS MaxSize

FROM [Production].[Product]

WHERE [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.

*Справочно. Задание 2 тоже сохранила в свой репозиторий в папку Lesson2 для себя, чтобы потом не забыть ☺*