



Агрегатные функции

Создать БД	<code>CREATE DATABASE teststep;</code>
Изменить наименование БД	<code>ALTER DATABASE teststep MODIFY NAME = teststep2;</code>
Удалить БД	<code>DROP DATABASE teststep;</code>
Создать таблицу	<code>USE teststep CREATE TABLE students (-- PRIMARY KEY = UNIQUE + NOT NULL id INT PRIMARY KEY, fio NVARCHAR(50) NOT NULL, groupNumber NVARCHAR(10));</code>
Изменить наименование таблицы	<code>USE teststep EXEC sp_rename 'students', 'Students';</code>
Удалить таблицу	<code>USE teststep DROP TABLE students;</code>
Добавить поле в таблицу	<code>USE teststep ALTER TABLE students ADD letter NVARCHAR(5);</code>
Изменить тип поля в таблице	<code>USE teststep ALTER TABLE students ALTER COLUMN letter NVARCHAR(10);</code>
Удалить поле из таблицы	<code>USE teststep ALTER TABLE students DROP COLUMN letter;</code>
Вставить строку с информацией в таблицу	<code>USE teststep INSERT INTO students (id, fio, groupNumber) VALUES (1, N'Иванов Иван Иванович', N'EKO-11021'), (2, N'Петров Петр Петрович', N'EKO-11022'), (3, N'Сидоров Сидор Сидорович', N'EKO-11021');</code>
Выборка данных из таблицы	<code>USE teststep SELECT * FROM students;</code>
Обновить (редактировать) данные поля в таблице по условию	<code>USE teststep UPDATE students SET groupNumber = N'EKO-11023' WHERE id = 3;</code>
Удалить строку в таблице по условию	<code>USE teststep DELETE FROM students WHERE id = 3;</code>
Очистить данные таблицы	<code>USE teststep TRUNCATE TABLE students;</code>
Выборка данных из таблицы	<code>USE teststep SELECT * FROM students;</code>

SQL

Выборка данных из таблицы с псевдонимами	<pre>USE teststep SELECT id AS ИД, fio AS ФИО FROM students;</pre>
Выборка уникальных данных из таблицы	<pre>USE teststep SELECT DISTINCT * FROM students;</pre>
Выборка ограниченного количества записей (строк) из таблицы	<pre>USE teststep SELECT TOP 2 * FROM students;</pre>
Выборка данных из таблицы по условию	<pre>USE teststep SELECT * FROM students WHERE id = 1;</pre>
Выборка данных из таблицы по составному условию	<pre>USE teststep SELECT * FROM students WHERE id > 1 AND id < 3; -- AND OR NOT = != > < <= >=</pre>
Выборка данных из таблицы по условию вхождения в множество	<pre>USE teststep SELECT * FROM students WHERE id IN (1); -- IN NOT IN</pre>
Выборка данных из таблицы по условию вхождения в диапазон	<pre>USE teststep SELECT * FROM students WHERE id BETWEEN 1 AND 3;</pre>
Выборка данных из таблицы по условию вхождения в фильтр	<pre>USE teststep SELECT * FROM students WHERE fio LIKE N'%Иван%'; -- % _</pre>
Выборка данных из таблицы по условию вхождения в фильтр регулярного выражения	<pre>USE teststep SELECT * FROM students WHERE id LIKE '[1-2]'; -- [abc] [!abc] [начало-конец]</pre>
Выборка данных из таблицы с сортировкой	<pre>USE teststep SELECT * FROM students ORDER BY id DESC; -- ASC DESC</pre>

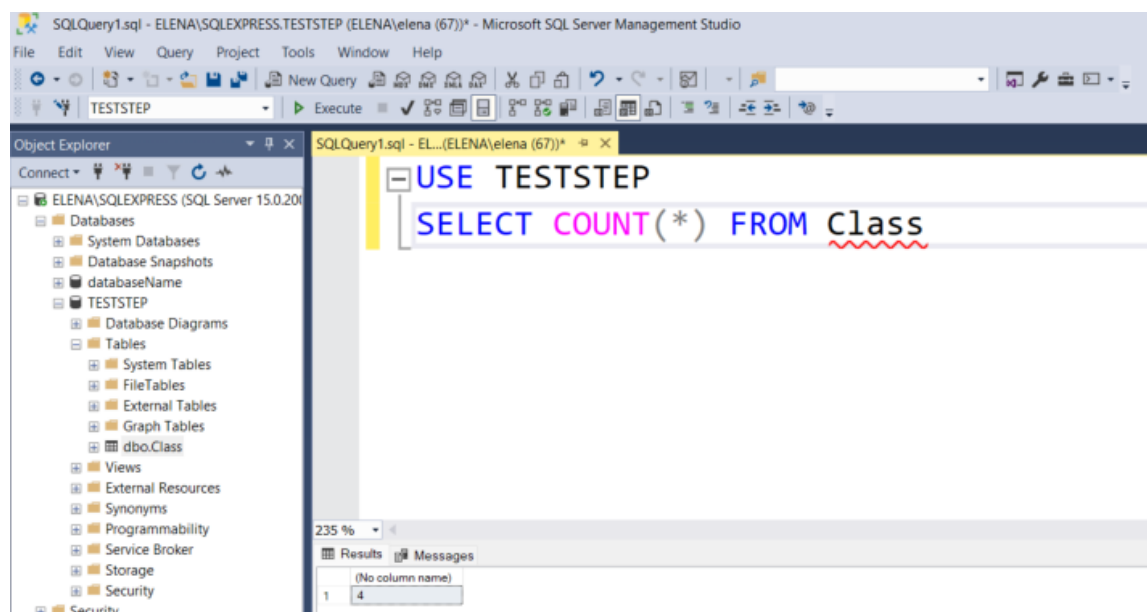
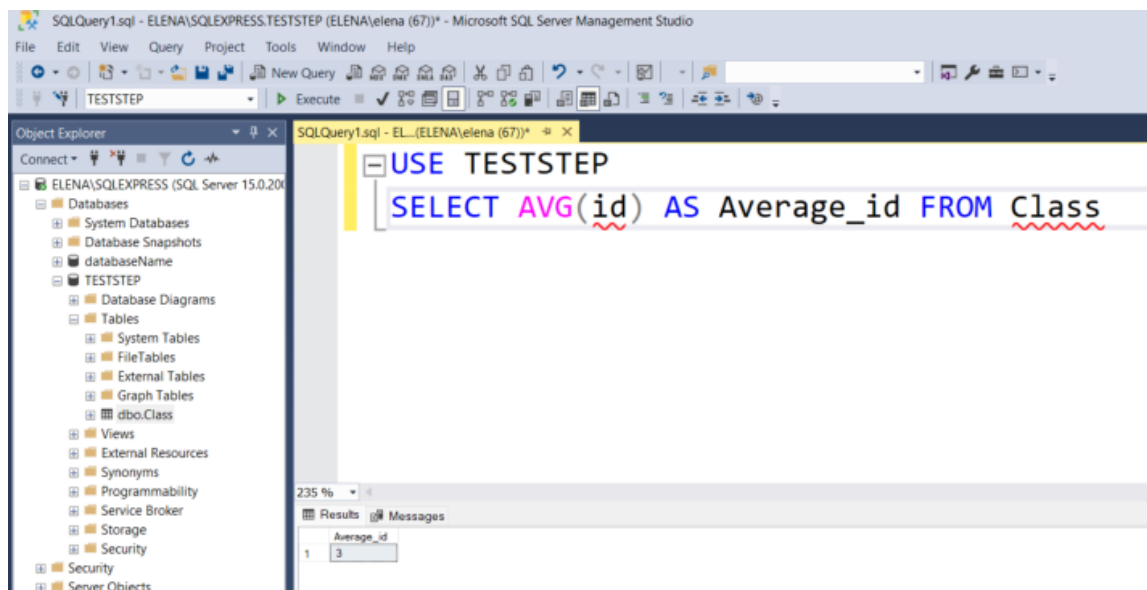
SQL

Внутри запроса **SELECT** должны указываться агрегатные функции (**SUM**, **AVG**, **COUNT**, **MAX**, **MIN**). Обязательно должны указываться столбцы, к которым они будут применяться.

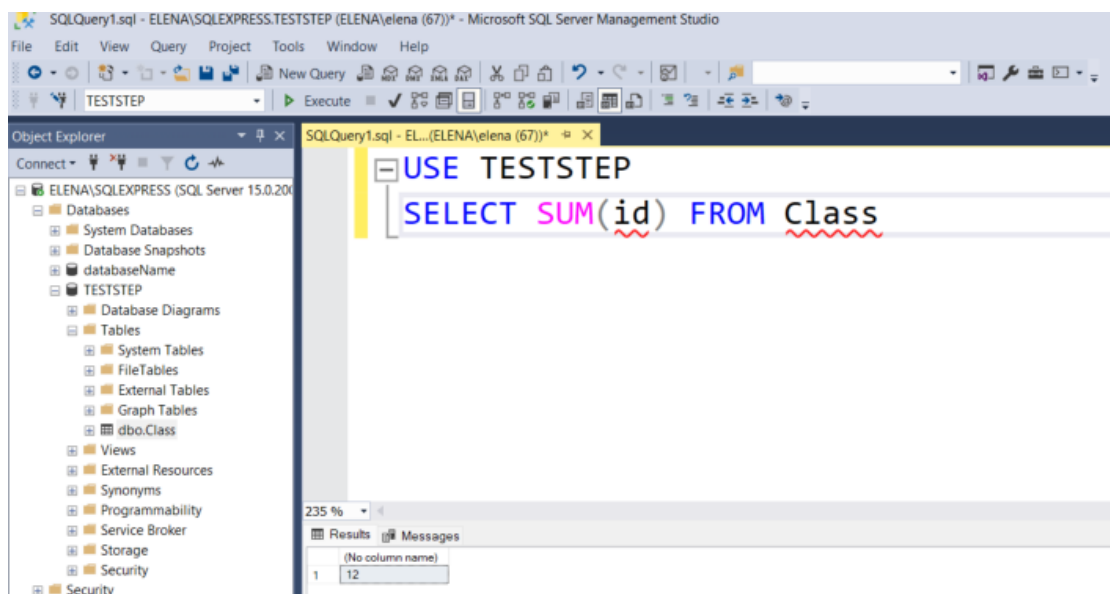
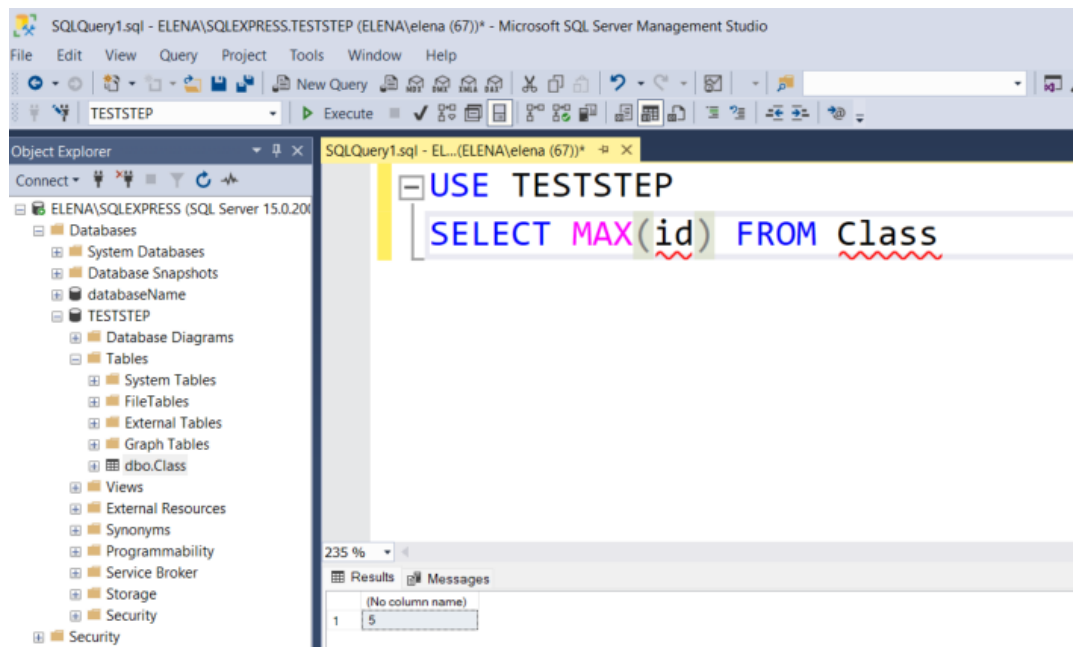
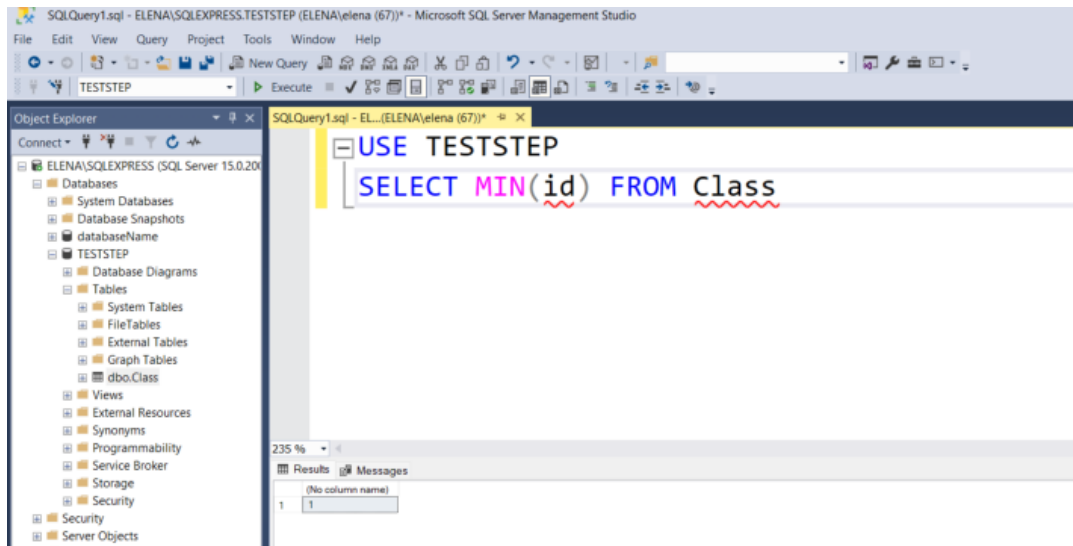
- **MIN()/ MAX()** – для возврата **минимального** либо **максимального** значения указанного столбца;
- **COUNT()** – для возврата **числа** строк;
- **AVG()** – для возврата **среднего значения** указанного столбца;
- **SUM()** – для возврата **суммы** всех полей столбца, у которых имеются числовые значения.

Выборка данных из таблицы с использованием агрегатных функций

```
USE teststep
SELECT AVG(id)
FROM students;
-- AVG COUNT SUM MIN MAX
```



SQL



SQL

Для того чтобы определить месяц есть стандартная функция языка **T-SQL** — **MONTH()**, которая позволяет получить номер месяца из переданной даты. Для того чтобы получить год и день, как часть даты, вы можете воспользоваться стандартными функциями **YEAR()** и **DAY()** соответственно.

Выборка данных из таблицы с использованием агрегатных функций для работы с числами

```
USE teststep
SELECT POWER(id, 2), fio
FROM students;
-- ROUND FLOOR TRUNCATE POWER SQRT
-- RAND
```

Для того чтобы определить количество символов (длину строки), есть функция **LEN()**, которая принимает строковые данные. Эта функция является встроенной в **MS SQL Server** и не входит в стандарт языка **T-SQL**, то есть ее выполнение обеспечивает сама **СУБД**.

Выборка данных из таблицы с использованием агрегатных функций для работы со строками

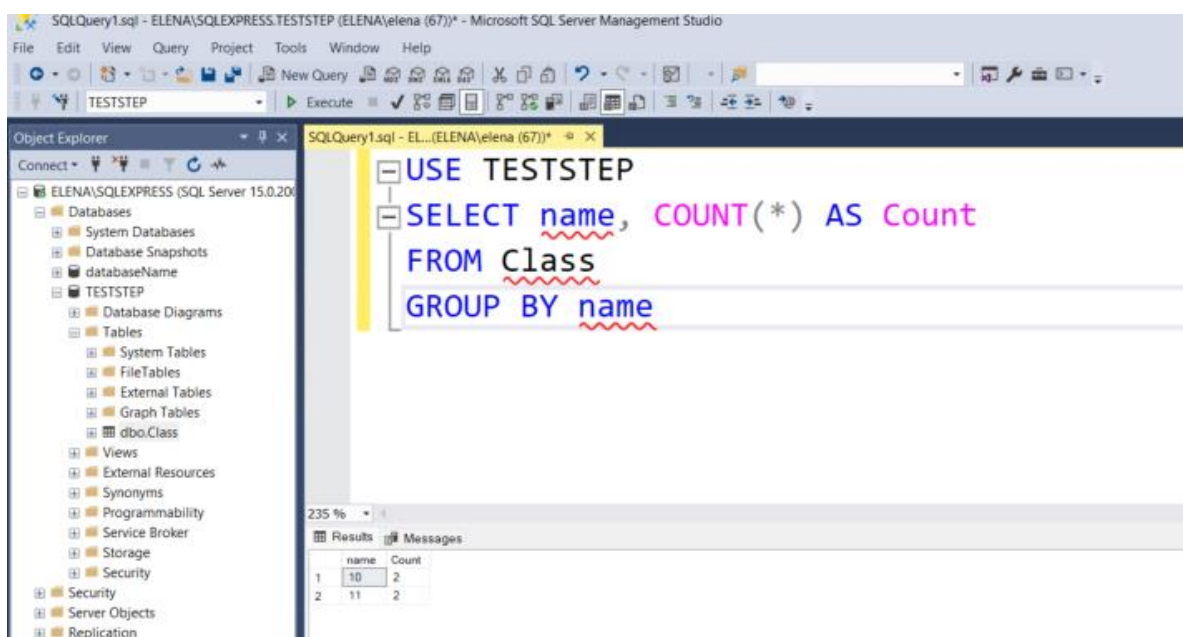
```
USE teststep
SELECT SUBSTRING(fio, 1, 7)
FROM students;
-- CONCAT LENGTH TRIM SUBSTRING
-- REPLACE LOWER UPPER
```

Агрегатные функции могут работать со всеми записями таблицы разом, а могут и с отдельными группами. Чтобы эти группы сформировать, используйте оператор **GROUP BY**. Элемент **GROUP BY** не обязателен для запроса. Он используется в тех случаях, когда необходимо задать агрегацию по определенному столбцу (к примеру, когда необходимо узнать, сколько клиентов проживает в каждом районе города). При применении **GROUP BY** необходимо, чтобы выполнялись условия:

- Список столбцов (по которым производится группировка) внутри запроса **SELECT** должен соответствовать списку столбцов в **GROUP BY**.
- Должна использоваться агрегатная функция

Выборка данных из таблицы с использованием агрегатных функций (обязательно) и группировки

```
USE teststep
SELECT COUNT(id) AS количество, fio
FROM students
GROUP BY fio;
```



SQL

Полученные группы тоже можно отфильтровывать: для этого предназначена конструкция **HAVING**. Where нельзя использовать с агрегатными функциями, поэтому для группировки есть свое условие, которое может работать с агрегатными функциями создать дополнительное условие фильтрации для группы.

Выборка данных из таблицы с использованием агрегатных функций (обязательно) и группировки с условием	<pre>USE teststep SELECT COUNT(id) AS количество, fio FROM students GROUP BY fio HAVING COUNT(id) < 2;</pre>
--	---



Агрегатные функции

Выборка данных из таблицы с использованием агрегатных функций	<pre>USE teststep SELECT AVG(id) FROM students; -- AVG COUNT SUM MIN MAX</pre>
Выборка данных из таблицы с использованием агрегатных функций для работы с числами	<pre>USE teststep SELECT POWER(id, 2), fio FROM students; -- ROUND FLOOR TRUNCATE POWER SQRT RAND</pre>
Выборка данных из таблицы с использованием агрегатных функций для работы со строками	<pre>USE teststep SELECT SUBSTRING(fio, 1, 7) FROM students; -- CONCAT LENGTH TRIM SUBSTRING REPLACE LOWER UPPER</pre>
Выборка данных из таблицы с использованием агрегатных функций (обязательно) и группировки	<pre>USE teststep SELECT COUNT(id) AS количество, fio FROM students GROUP BY fio;</pre>
Выборка данных из таблицы с использованием агрегатных функций (обязательно) и группировки с условием	<pre>USE teststep SELECT COUNT(id) AS количество, fio FROM students GROUP BY fio HAVING COUNT(id) < 2;</pre>



Практическая работа

*** Запросы необходимо сохранить для проверки**

1. Вывести количество палат, вместимость которых больше 10.
2. Вывести названия корпусов и количество палат в каждом из них.
3. Вывести названия отделений и количество палат в каждом из них.
4. Вывести названия отделений и суммарную надбавку врачей в каждом из них.
5. Вывести названия отделений, в которых проводят обследования 5 и более врачей.
6. Вывести количество врачей и их суммарную зарплату (сумма ставки и надбавки).
7. Вывести среднюю зарплату (сумма ставки и надбавки) врачей.
8. Вывести названия палат с минимальной вместительностью.
9. Вывести в каких из корпусов 1, 6, 7 и 8, суммарное количество мест в палатах превышает 100. При этом учитывать только палаты с количеством мест больше 10.



Домашняя работа

*** Прислать запросы для проверки**

1. Вывести количество преподавателей кафедры "Software Development".
2. Вывести количество лекций, которые читает преподаватель "Dave McQueen".
3. Вывести количество занятий, проводимых в аудитории "D201".
4. Вывести названия аудиторий и количество лекций, проводимых в них.
5. Вывести количество студентов, посещающих лекции преподавателя "Jack Underhill".
6. Вывести среднюю ставку преподавателей факультета "Computer Science".
7. Вывести минимальное и максимальное количество студентов среди всех групп.
8. Вывести средний фонд финансирования кафедр.
9. Вывести полные имена преподавателей и количество читаемых ими дисциплин.
10. Вывести количество лекций в каждый день недели.