



## Введение в SQL

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>
Работать с указанной БД	<code>USE teststep;</code>
Создать таблицу	<code>USE teststep CREATE TABLE students (-- PRIMARY KEY = UNIQUE + NOT NULL id INT NOT NULL IDENTITY(1, 1) PRIMARY KEY, fio NVARCHAR(MAX) NOT NULL CHECK (fio &lt;&gt; N''), groupNumber NVARCHAR(10) UNIQUE, grant MONEY NOT NULL CHECK (grant &gt;= 0.0) DEFAULT (0.0), BirthDate DATE NOT NULL CHECK (BirthDate &lt;= getdate()), StartTime TIME NOT NULL CHECK (StartTime BETWEEN '08:00' AND '18:00') );</code>
Связать таблицы	<code>ALTER TABLE students ADD FOREIGN KEY (GroupId REFERENCES Groups(Id);</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'ЕКО- 11021'), (2, N'Петров Петр Петрович', N'ЕКО- 11022'),</code>

# SQL

	(3, N'Сидоров Сидор Сидорович', N'ЕКО-11021');
Выборка данных из таблицы	USE teststep SELECT * FROM students;
Выборка данных из таблицы с псевдонимами	USE teststep SELECT id AS ИД, fio AS ФИО FROM students;
Выборка уникальных данных из таблицы	USE teststep SELECT DISTINCT * FROM students;
Выборка ограниченного количества записей (строк) из таблицы	USE teststep SELECT TOP 2 * FROM students;
Выборка данных из таблицы по условию	USE teststep SELECT * FROM students WHERE id = 1;
Выборка данных из таблицы по составному условию	USE teststep SELECT * FROM students WHERE id > 1 AND id < 3; -- AND OR NOT = != > < <= >=
Выборка данных из таблицы по условию вхождения в множество	USE teststep SELECT * FROM students WHERE id IN (1); -- IN NOT IN
Выборка данных из таблицы по условию вхождения в диапазон	USE teststep SELECT * FROM students WHERE id BETWEEN 1 AND 3;
Выборка данных из таблицы по условию вхождения в фильтр	USE teststep SELECT * FROM students WHERE fio LIKE N'%Иван%'; -- % _
Выборка данных из таблицы по условию вхождения в фильтр регулярного выражения	USE teststep SELECT * FROM students WHERE id LIKE '[1-2]'; -- [abc] [!abc] [начало-конец]
Выборка данных из таблицы с сортировкой	USE teststep SELECT * FROM students ORDER BY id DESC; -- ASC DESC

# SQL

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) &lt; 2;</pre>
Обновить (редактировать) данные поля в таблице по условию	<pre>USE teststep UPDATE students SET groupNumber = 'EKO-11023' WHERE id = 3;</pre>
Удалить строку в таблице по условию	<pre>USE teststep DELETE FROM students WHERE id = 3;</pre>
Очистить данные таблицы	<pre>USE teststep TRUNCATE TABLE students;</pre>
Подзапросы	<pre>USE Hospital SELECT Name FROM Wards WHERE Places = (SELECT MIN(Places) FROM Wards);</pre>
Связать данные при помощи идентификаторов в WHERE	<pre>use Hospital select D.Surname, D.Salary+D.Premium as Salary from Doctors as D, Vacations as V where D.Id = V.DoctorId and MONTH(V.StartDate)!=10;</pre>
Связать данные при помощи UNION	<pre>SELECT FirstName + ' ' + LastName AS FullName, BirthDate FROM Students WHERE MONTH(BirthDate) &gt; 5 AND MONTH(BirthDate) &lt; 9</pre>

# SQL

	<pre>UNION SELECT FirstName + ' ' + LastName, BirthDate FROM Teachers WHERE MONTH(BirthDate) &gt; 5 AND MONTH(BirthDate) &lt; 9 ORDER BY BirthDate;</pre>
Связать данные при помощи INNER JOIN ON	<pre>SELECT FirstName, LastName, Name AS Subject, Assesment, GroupName FROM Groups AS G JOIN Students AS S ON G.Id = S.GroupId JOIN Achievements AS A ON S.Id = A.StudentId JOIN Subjects AS Sb ON Sb.Id = A.SubjectId WHERE GroupName LIKE '29%' ORDER BY LastName;</pre>
Связать данные при помощи LEFT OUTER JOIN ON	<pre>SELECT FirstName + ' ' + LastName AS FullName FROM Students AS S LEFT JOIN Achievements AS A ON S.Id = A.StudentId WHERE Assesment IS NULL;</pre>
Связать данные при помощи RIGHT OUTER JOIN ON	<pre>SELECT FirstName + ' ' + LastName AS FullName, Assesment FROM Achievements AS A RIGHT JOIN Students AS S ON S.Id = A.StudentId;</pre>
Связать данные при помощи FULL OUTER JOIN ON	<pre>SELECT FirstName, LastName, GroupName FROM Students AS S FULL JOIN Groups AS G ON G.Id = S.GroupId ORDER BY FirstName;</pre>
Связать данные при помощи комбинирования JOIN ON	<pre>SELECT [Name] AS [Subject], LastName, FirstName FROM TeachersSubjects AS TS RIGHT JOIN Subjects AS S ON S.Id = TS.SubjectId LEFT JOIN Teachers AS T ON T.Id = TS.TeacherId ORDER BY [Name];</pre>