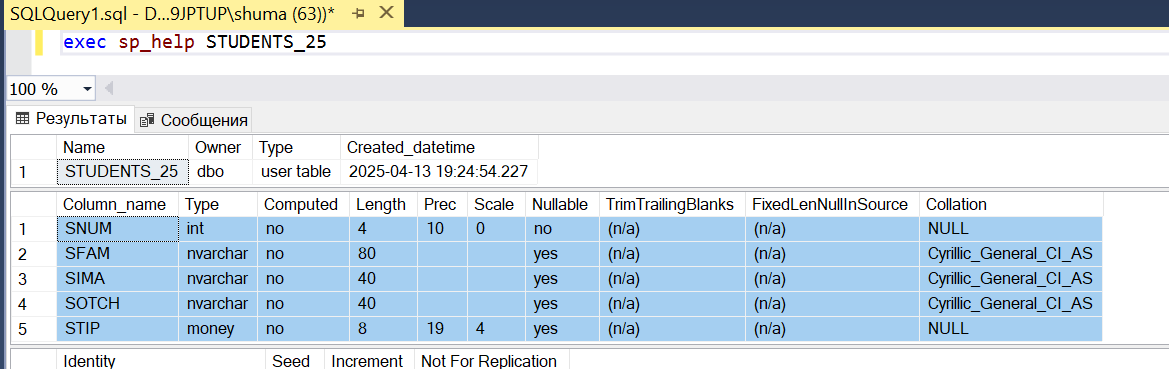
**Лабораторная работа № 9.**

**Операции с БД средствами языка манипулирования данными –**

**DML - SQL**

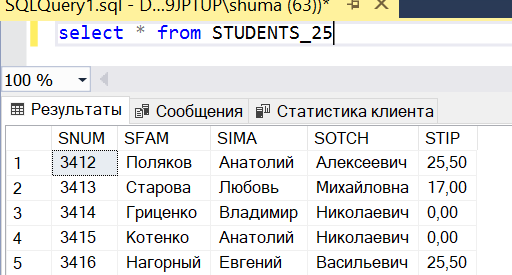
⇒ **1**. В таблицу **STUDENTS\_N** (N – Ваш номер по журналу) вставьте записи о 10- ти новых студентах (по своему усмотрению) со значением стипендии - 0.

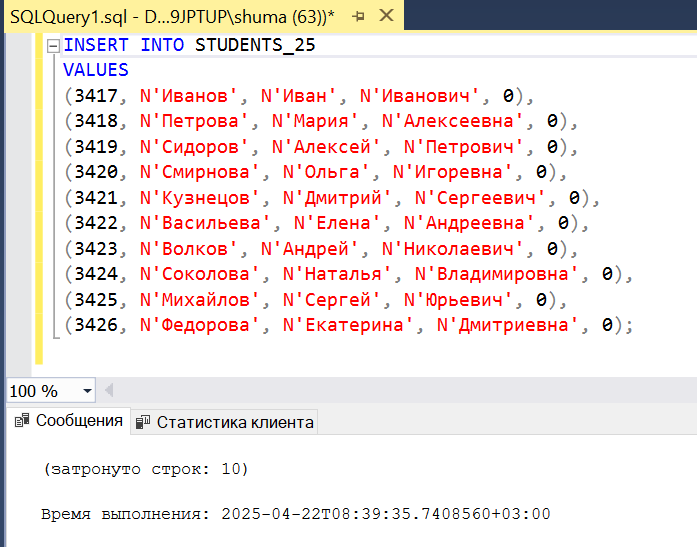
Вспомним, какие поля есть у нас в таблице



И внедрим в неё 10 случайных студентов

было



Выполнил  


INSERT INTO STUDENTS\_25

VALUES

(3417, N'Иванов', N'Иван', N'Иванович', 0),

(3418, N'Петрова', N'Мария', N'Алексеевна', 0),

(3419, N'Сидоров', N'Алексей', N'Петрович', 0),

(3420, N'Смирнова', N'Ольга', N'Игоревна', 0),

(3421, N'Кузнецов', N'Дмитрий', N'Сергеевич', 0),

(3422, N'Васильева', N'Елена', N'Андреевна', 0),

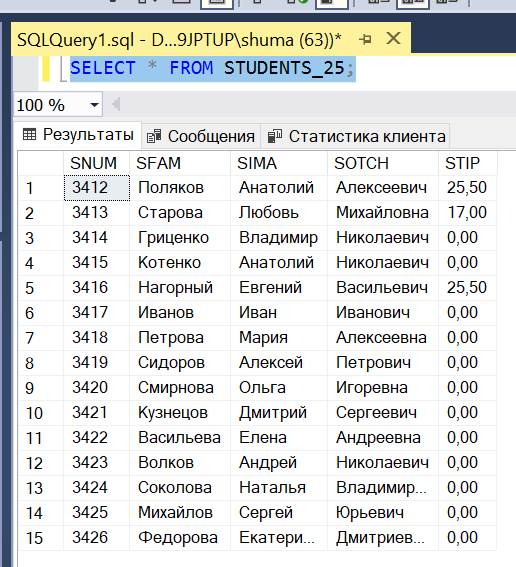
(3423, N'Волков', N'Андрей', N'Николаевич', 0),

(3424, N'Соколова', N'Наталья', N'Владимировна', 0),

(3425, N'Михайлов', N'Сергей', N'Юрьевич', 0),

(3426, N'Федорова', N'Екатерина', N'Дмитриевна', 0);

Итого



⇒ **2**. В таблице **PREDMET\_N** удалите ранее созданное в ЛР-8 ограничение на

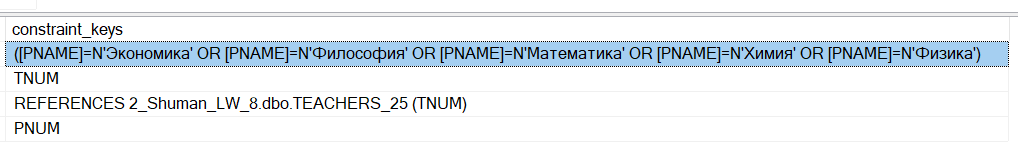
названия предметов и вставьте новые записи о предметах «Базы данных», «Языки программирования», «Проектирование информационных систем» с

количеством часов, равных предметам «Физика», «Химия», «Философия»

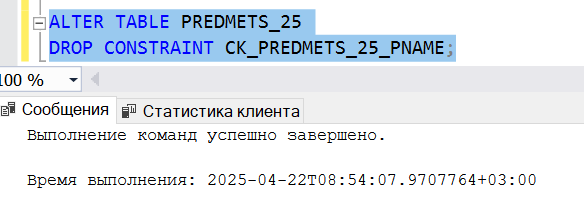
соответственно. В поле **TNUM** внесите значения NULL

удалим ранее созданные ограничения на названия предметов

было



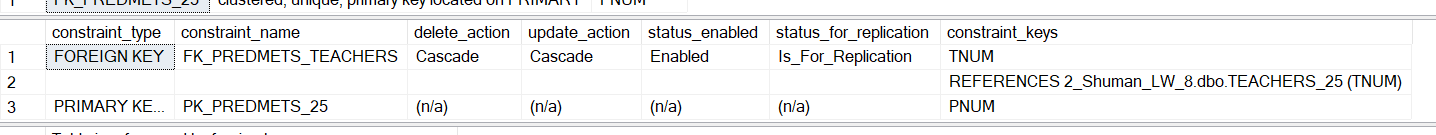
Выполнили



ALTER TABLE PREDMETS\_25

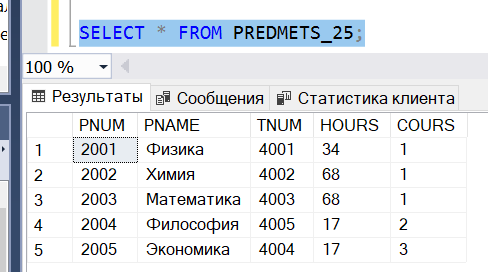
DROP CONSTRAINT CK\_PREDMETS\_25\_PNAME;

Итого



Далее вставим новые записи

Какие были

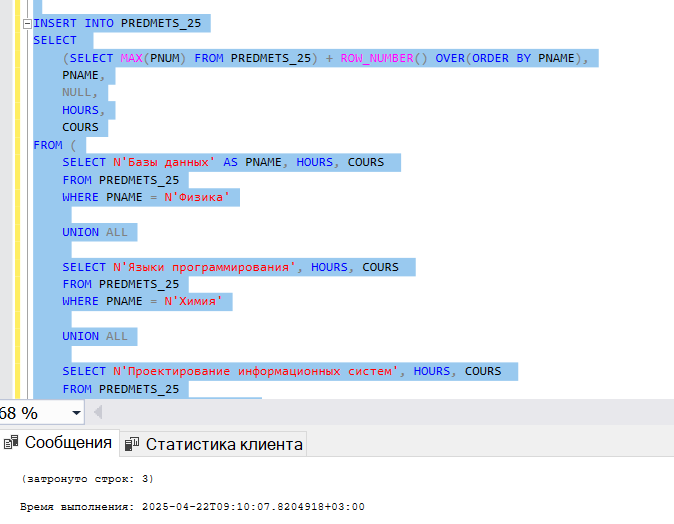


Вставим новые записи о предметах «Базы данных», «Языки программирования», «Проектирование информационных систем» с

количеством часов, равных предметам «Физика», «Химия», «Философия»

соответственно. В поле **TNUM** внесите значения NULL

Вставляем



INSERT INTO PREDMETS\_25

SELECT

(SELECT MAX(PNUM) FROM PREDMETS\_25) + ROW\_NUMBER() OVER(ORDER BY PNAME),

PNAME,

NULL,

HOURS,

COURS

FROM (

SELECT N'Базы данных' AS PNAME, HOURS, COURS

FROM PREDMETS\_25

WHERE PNAME = N'Физика'

UNION ALL

SELECT N'Языки программирования', HOURS, COURS

FROM PREDMETS\_25

WHERE PNAME = N'Химия'

UNION ALL

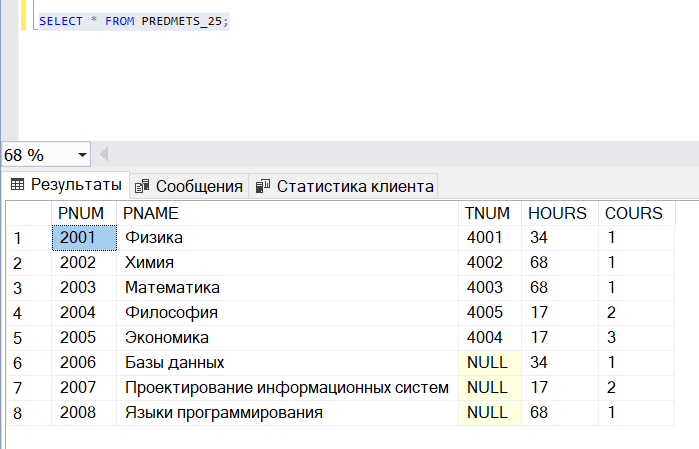
SELECT N'Проектирование информационных систем', HOURS, COURS

FROM PREDMETS\_25

WHERE PNAME = N'Философия'

) AS temp;

Проверяем



Разберём запрос

Чего я хотел добиться

Вставить все одной командой

Не прописывать PNUM (PK) каждый раз

Технически – я вставляю из таблицы, которая представляет собой объединение из трёх таблиц по одному элементу в каждой.

(SELECT MAX(PNUM) FROM PREDMETS\_25) + ROW\_NUMBER() OVER(ORDER BY PNAME) – эта строчка помогает мне не записывать каждый раз PNUM в ручную

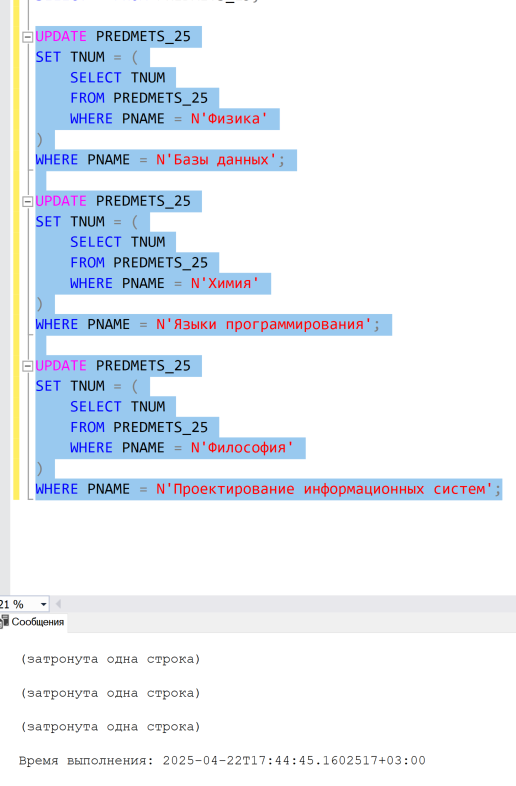
⇒ **3**. В таблицу **PREDMET\_N** внесите изменения в поле **TNUM** для вставленных

на предыдущем шаге записей, соответствующие тому, что преподаватель,

который ведет предмет «Физика» будет вести и предмет «Базы данных», кто вел

«Химия» - «Языки программирования» и соответственно «Философия» -

«Проектирование информационных систем»



UPDATE PREDMETS\_25

SET TNUM = (

SELECT TNUM

FROM PREDMETS\_25

WHERE PNAME = N'Физика'

)

WHERE PNAME = N'Базы данных';

UPDATE PREDMETS\_25

SET TNUM = (

SELECT TNUM

FROM PREDMETS\_25

WHERE PNAME = N'Химия'

)

WHERE PNAME = N'Языки программирования';

UPDATE PREDMETS\_25

SET TNUM = (

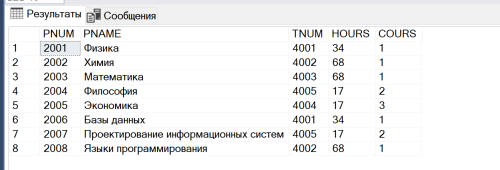
SELECT TNUM

FROM PREDMETS\_25

WHERE PNAME = N'Философия'

)

WHERE PNAME = N'Проектирование информационных систем';

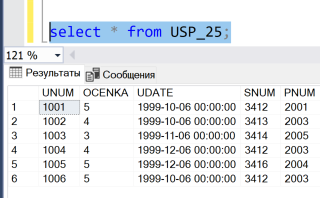


⇒ **4**. Внесите новые записи в таблицу **USP\_N** об оценках по всем новым предметам

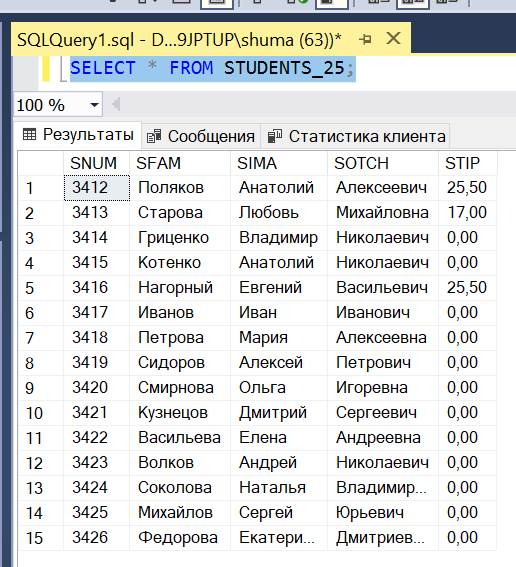
для всех студентов, имеющих стипендию 0 (оценки по собственному

усмотрению)

как выглядела таблица до

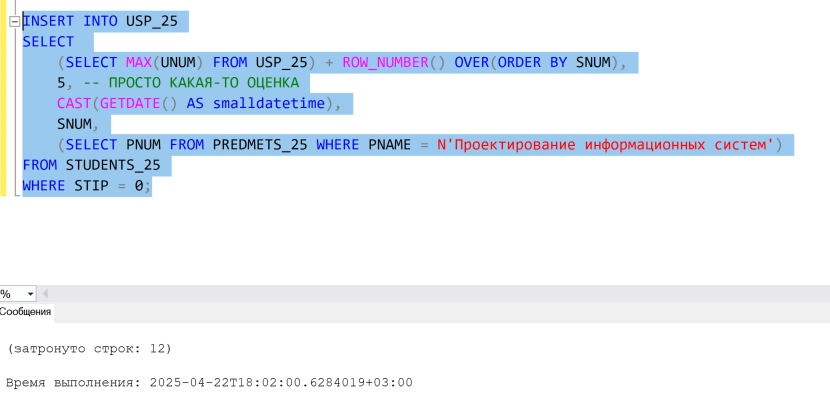


Вспомним, какие у нас были студенты



Результат





INSERT INTO USP\_25

SELECT

(SELECT MAX(UNUM) FROM USP\_25) + ROW\_NUMBER() OVER(ORDER BY SNUM),

4, -- ПРОСТО КАКАЯ-ТО ОЦЕНКА

CAST(GETDATE() AS smalldatetime),

SNUM,

(SELECT PNUM FROM PREDMETS\_25 WHERE PNAME = N'Базы данных')

FROM STUDENTS\_25

WHERE STIP = 0;

INSERT INTO USP\_25

SELECT

(SELECT MAX(UNUM) FROM USP\_25) + ROW\_NUMBER() OVER(ORDER BY SNUM),

3, -- ПРОСТО КАКАЯ-ТО ОЦЕНКА

CAST(GETDATE() AS smalldatetime),

SNUM,

(SELECT PNUM FROM PREDMETS\_25 WHERE PNAME = N'Языки программирования')

FROM STUDENTS\_25

WHERE STIP = 0;

INSERT INTO USP\_25

SELECT

(SELECT MAX(UNUM) FROM USP\_25) + ROW\_NUMBER() OVER(ORDER BY SNUM),

5, -- ПРОСТО КАКАЯ-ТО ОЦЕНКА

CAST(GETDATE() AS smalldatetime),

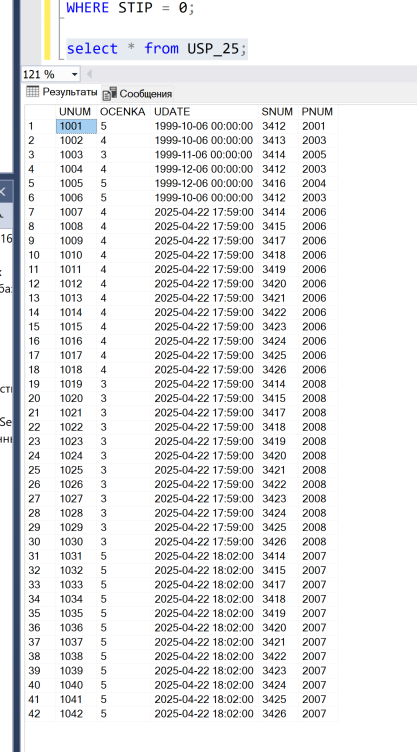
SNUM,

(SELECT PNUM FROM PREDMETS\_25 WHERE PNAME = N'Проектирование информационных систем')

FROM STUDENTS\_25

WHERE STIP = 0;

Результат

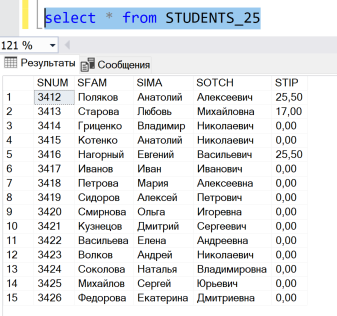


⇒ **5**. Измените размер стипендии до 30 всем студентам, имеющим оценки «5» по более чем двум предметам (если таких нет – скорректируйте таблицы любым

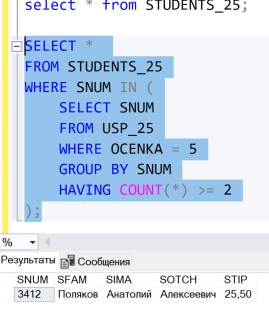
способом – через дополнительный запрос или через интерфейс Management

Studio)

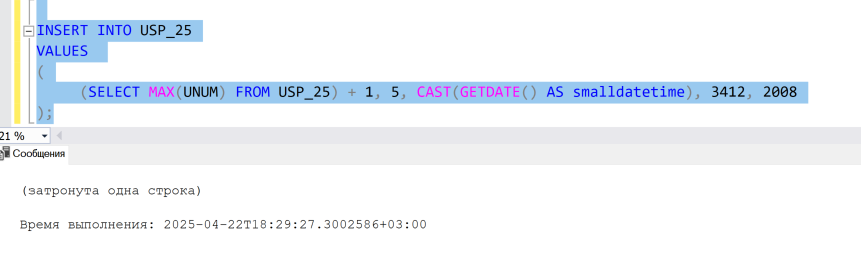
Как было



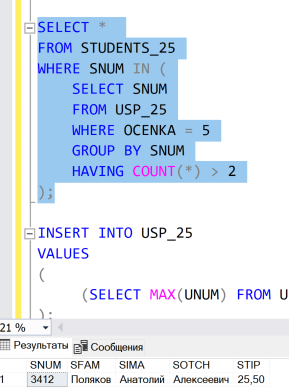
Предварительный запрос показал, что такого у меня нет  
поэтому – выявим ближайшего нам нужного



И теперь – добавим студенту ещё одну оценку



Теперь



SELECT \*

FROM STUDENTS\_25

WHERE SNUM IN (

SELECT SNUM

FROM USP\_25

WHERE OCENKA = 5

GROUP BY SNUM

HAVING COUNT(\*) > 2

);

INSERT INTO USP\_25

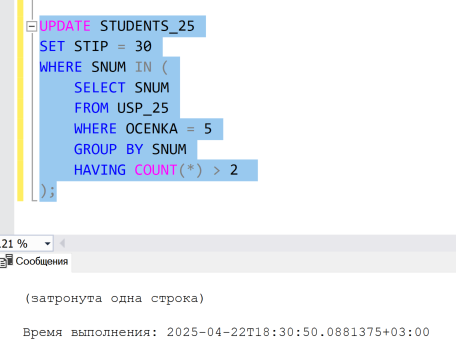
VALUES

(

(SELECT MAX(UNUM) FROM USP\_25) + 1, 5, CAST(GETDATE() AS smalldatetime), 3412, 2008

);

Теперь можно выполнять запрос



UPDATE STUDENTS\_25

SET STIP = 30

WHERE SNUM IN (

SELECT SNUM

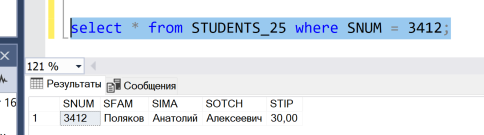
FROM USP\_25

WHERE OCENKA = 5

GROUP BY SNUM

HAVING COUNT(\*) > 2

);



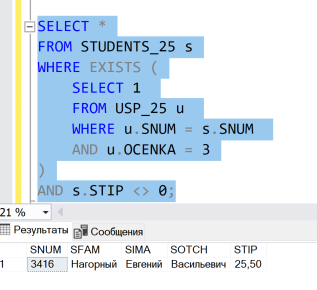
⇒ **6**. Измените размер стипендии до 5 всем студентам, имеющим хотя бы одну

оценку «3» по предметам, при этом если у него стипендия не была равна «0»

проверим, есть ли у меня такие студенты

Нету, зн надо добавлять

Номера тех, у кого есть оценка 3



SELECT \*

FROM STUDENTS\_25 s

WHERE EXISTS (

SELECT 1

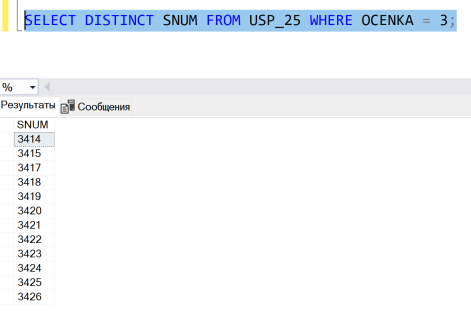
FROM USP\_25 u

WHERE u.SNUM = s.SNUM

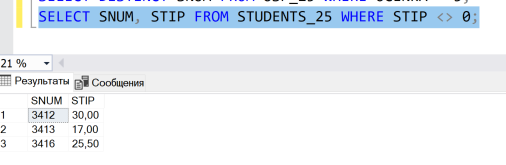
AND u.OCENKA = 3

)

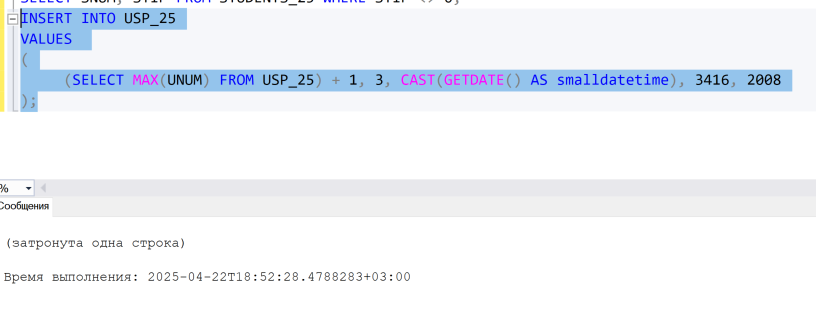
AND s.STIP <> 0;



Студенты, у кого не нулевая стипендия



Добавим 3416-ому оценку 3 по 2008



INSERT INTO USP\_25

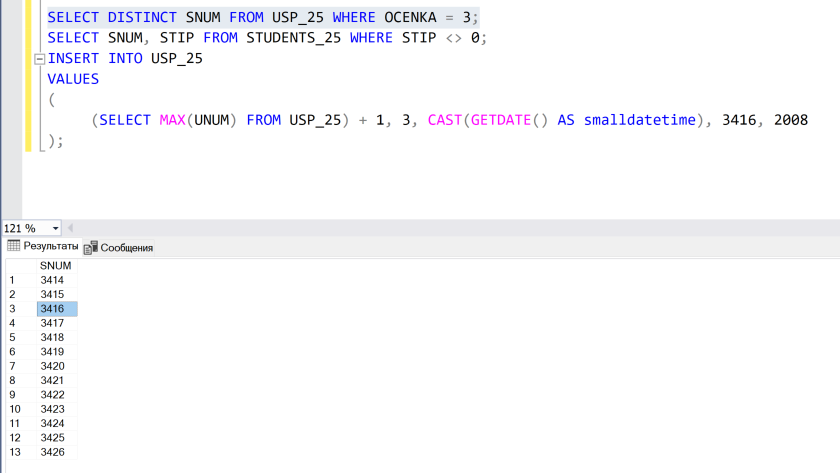
VALUES

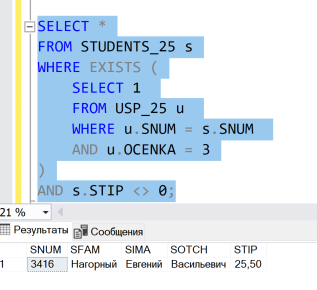
(

(SELECT MAX(UNUM) FROM USP\_25) + 1, 3, CAST(GETDATE() AS smalldatetime), 3416, 2008

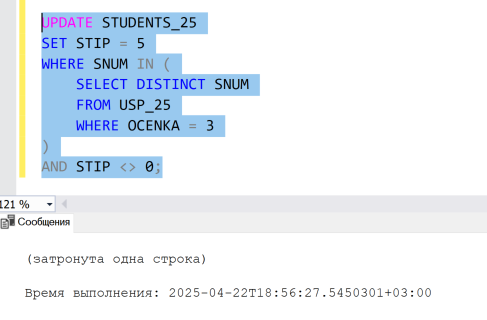
);

Проверяем





Выполняем команду



UPDATE STUDENTS\_25

SET STIP = 5

WHERE SNUM IN (

SELECT DISTINCT SNUM

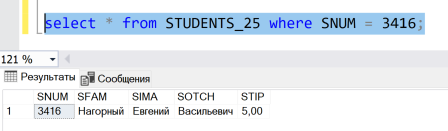
FROM USP\_25

WHERE OCENKA = 3

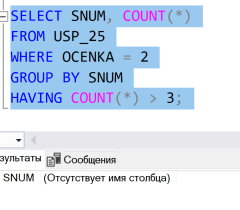
)

AND STIP <> 0;

В результате имеем



⇒ **7**. Удалите записи из таблицы **STUDENTS\_N** о студентах, имеющих более 3-х оценок (2) по любым предметам. – Обязательно сделайте запрос на выборку всех строк таблицы и скриншот поместите в отчет предварительно перед удалением и также после удаления.



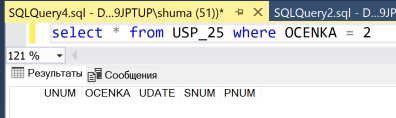
SELECT SNUM, COUNT(\*)

FROM USP\_25

WHERE OCENKA = 2

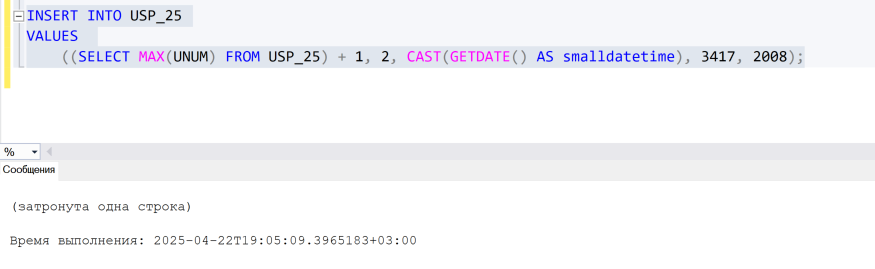
GROUP BY SNUM

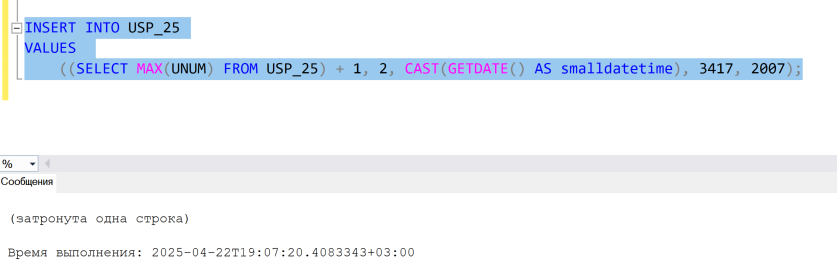
HAVING COUNT(\*) > 3;

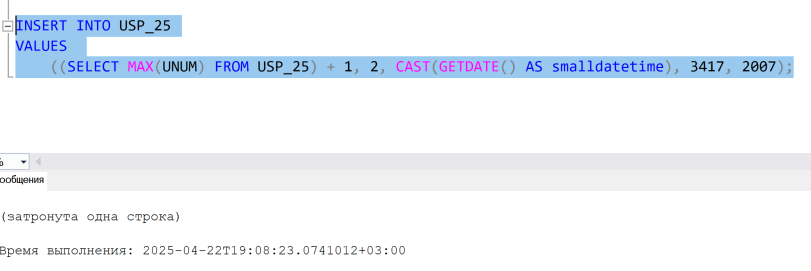


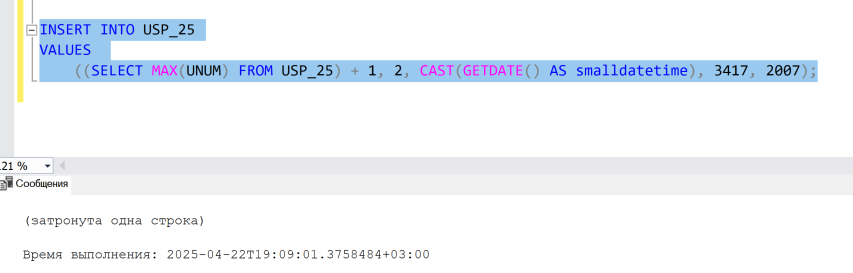
Нужно добавить такие оценки

Пусть это будет Студент 3417

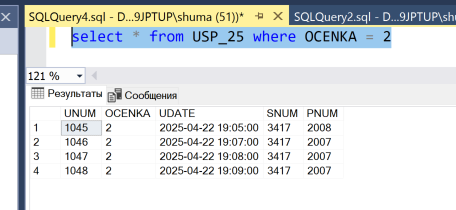


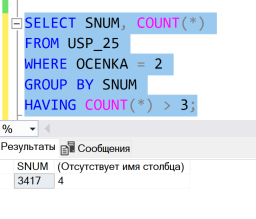




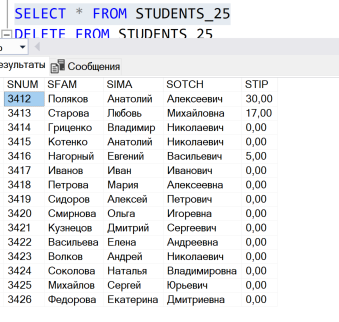


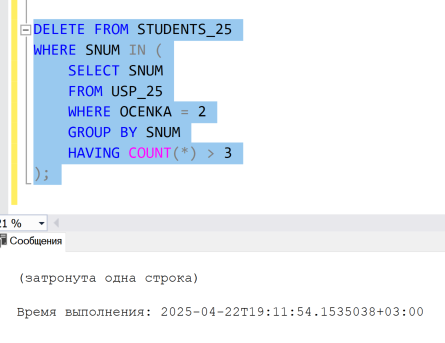
Итого





БЫЛО





DELETE FROM STUDENTS\_25

WHERE SNUM IN (

SELECT SNUM

FROM USP\_25

WHERE OCENKA = 2

GROUP BY SNUM

HAVING COUNT(\*) > 3

);

СТАЛО

