



**МИНОБРНАУКИ РОССИИ**  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
**МИРЭА – Российский технологический университет**  
**РТУ МИРЭА**  
**Колледж приборостроения и кибербезопасности**

**Практическая работа №8**  
по учебной дисциплине МДК.11.01  
**Технология разработки и защиты баз данных**  
специальность 09.02.07 Информационные системы и программирование  
**Создание хранимых процедур (функций)**

выполнил  
студент группы ПКС-31  
Лопатин Л.В.  
преподаватель  
Понеделко Е. В.

Москва

2025

**Цель работы:** Формирование навыков работы в среде СУБД PostgreSQL по созданию хранимых процедур (функций).

### Ход работы.

**Задание 1.** Рассмотрел примеры и адаптировал код.

**Пример 1.** Создать хранимую процедуру, которая получает в качестве входного параметра количество баллов и на основании шкалы оценок вычисляет полученную оценку. Результат возвращается через выходной параметр. Результат представлен на рисунке 1.

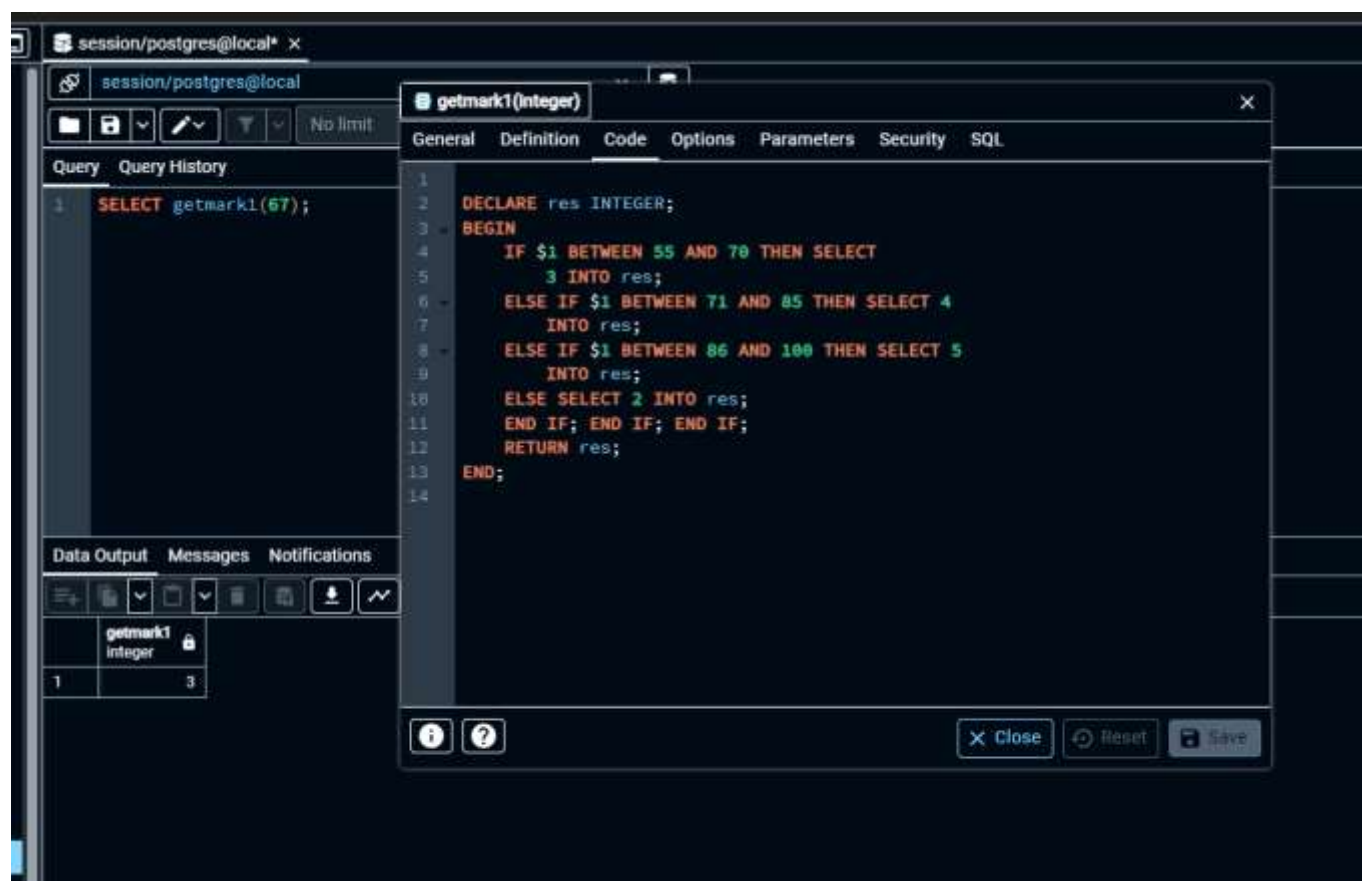


Рис. 1 Пример 1

**Пример 2.** Чтобы при смене правил вычисления оценок не нужно было бы менять процедуру, создали справочную таблицу для хранения всех оценок и их диапазонов Marks. Второй вариант функции получения оценки по набранным баллам будет обращаться к этой таблице за информацией. Этот вариант оформлен в виде функции с одним параметром, хранящим набранные

баллы, и возвращающую найденную оценку или 2 в случае, когда набранным баллам ничего в таблице не соответствует. Результат представлен на рисунке 2 и 3.

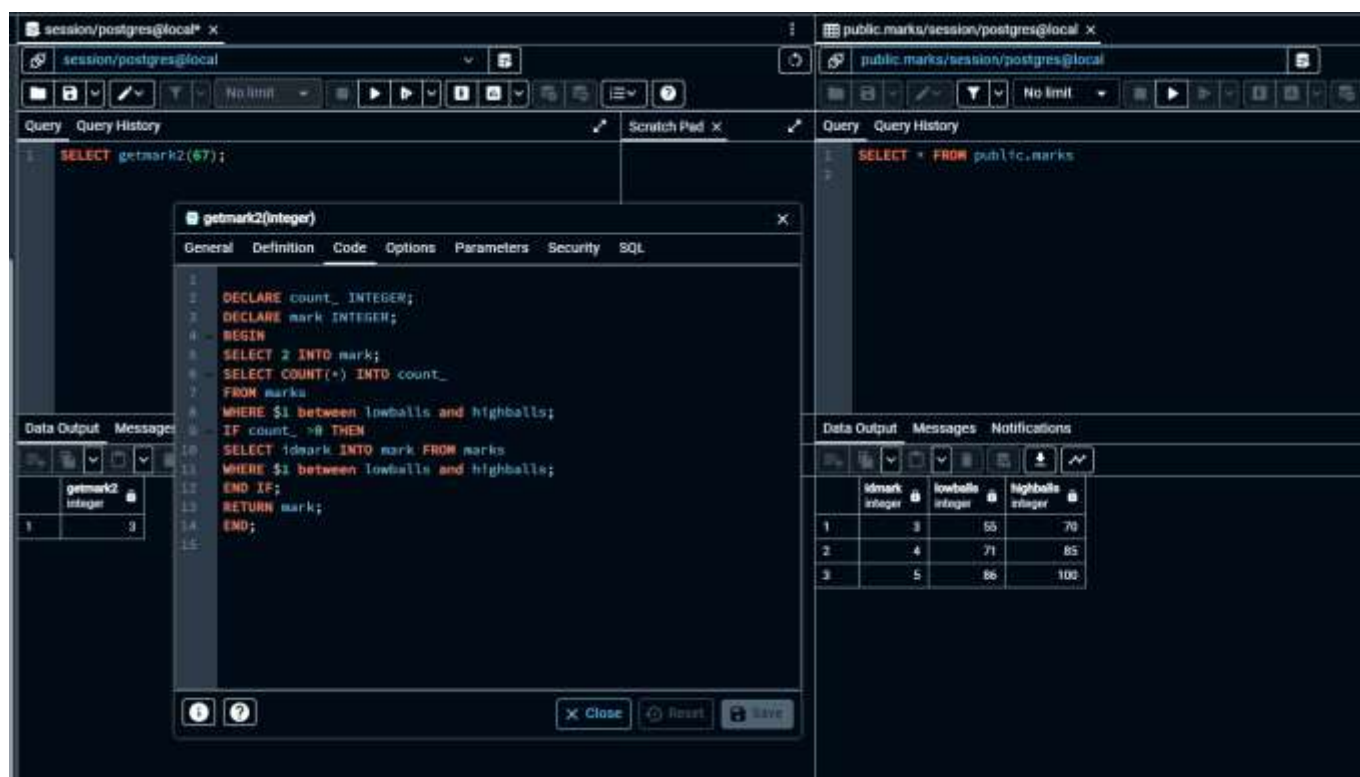


Рис. 2 Пример 2 Вариант 1

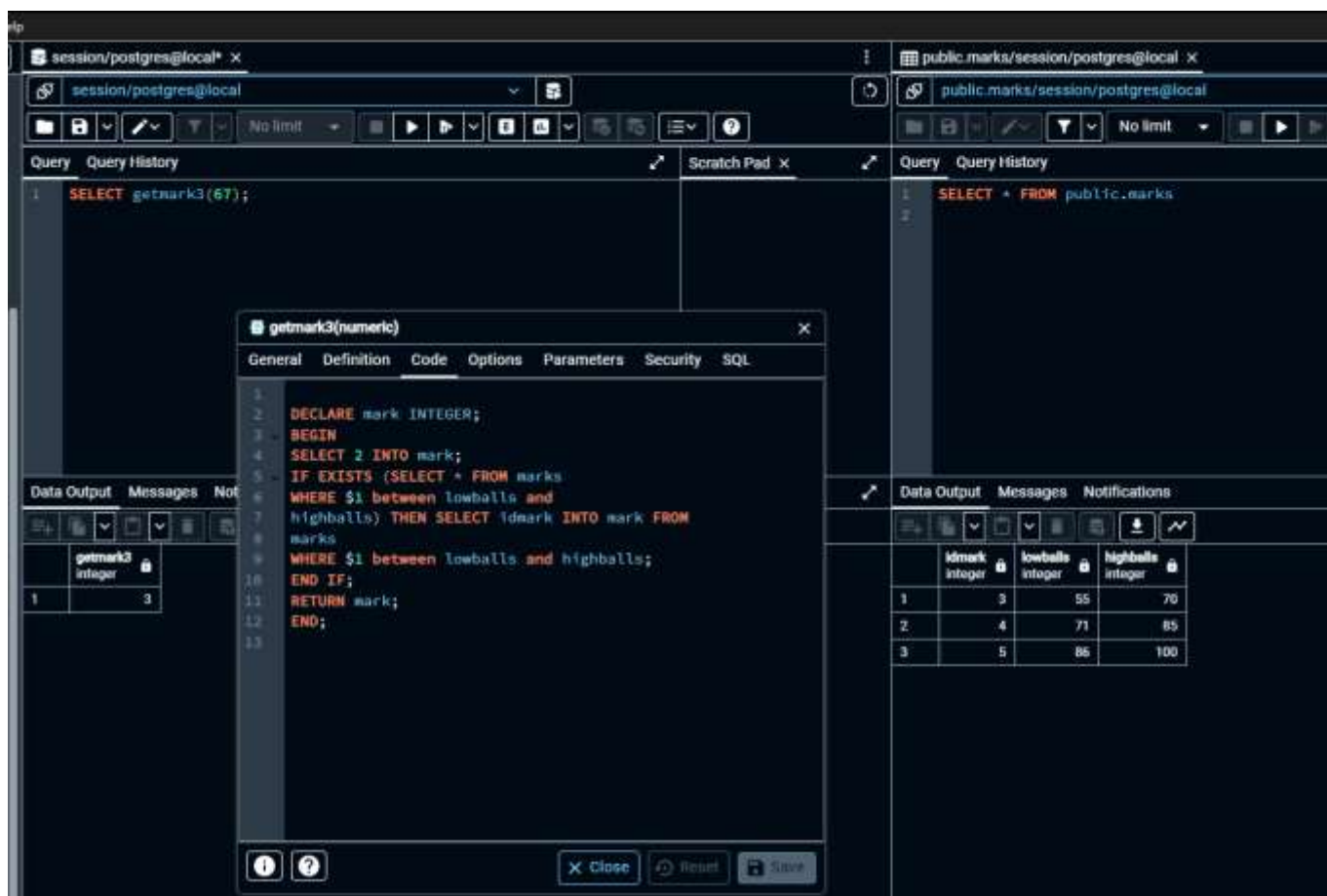


Рис. 3 Пример 2 Вариант 2

**Задание 2.** Проанализировал БД «Сессия» и создал 2 процедуры.

**Процедура 1.** Пользовательская потребность: Необходимо создать список стипендиатов. Для этого создал таблицу *stipendia*, хранящую *id* студентов, среднюю оценку и статус стипендии (обычная или повышенная) (см. рисунок 4). Создал функцию для обновления таблицы (см. рисунок 5). Результат выполнения представлен на рисунке 6. Для проверки выполнения функции записал временного студента (см. рисунок 6), добавил оценки за экзамены (см. рисунок 7), выполнил функцию (см. рисунок 8).

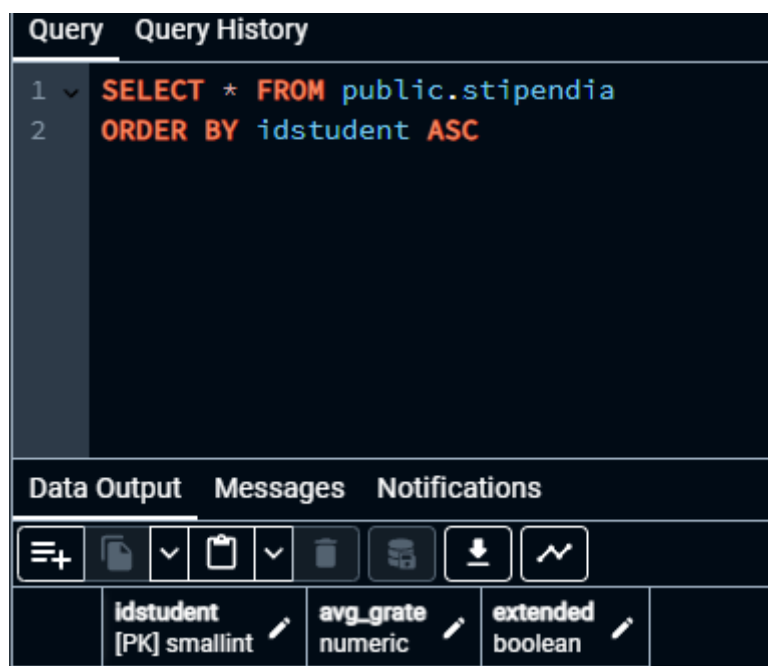


Рис. 4 Таблица стипендиатов

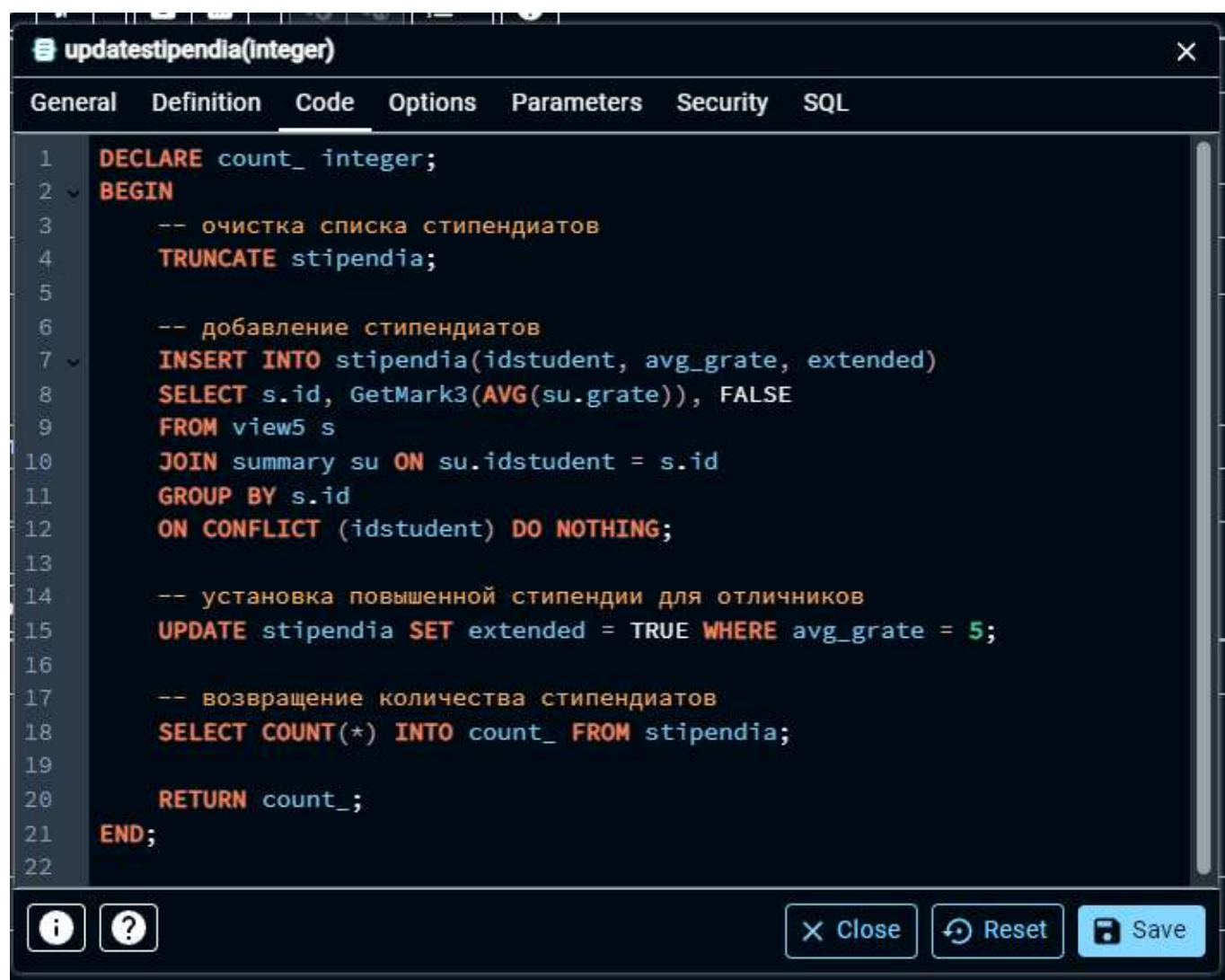


Рис. 5 Функция для обновления таблицы стипендиатов

The screenshot displays a PostgreSQL client interface with two tabs. The first tab, 'session/postgres@local', shows a query window with the SQL statement: `SELECT updatestipendia(1);`. Below the query window, the 'Data Output' tab is active, showing a single row of results for the function call. The second tab, 'public.stipendia/session/postgres@local', shows a query window with the SQL statement: `SELECT * FROM public.stipendia ORDER BY idstudent ASC`. Below the query window, the 'Data Output' tab is active, showing a table with four rows of data.

**Query 1 Results:**

	updatestipendia integer
1	4

Total rows: 1 of 1    Query complete 00:00:00.255    Ln 1, Col 1

**Query 2 Results:**

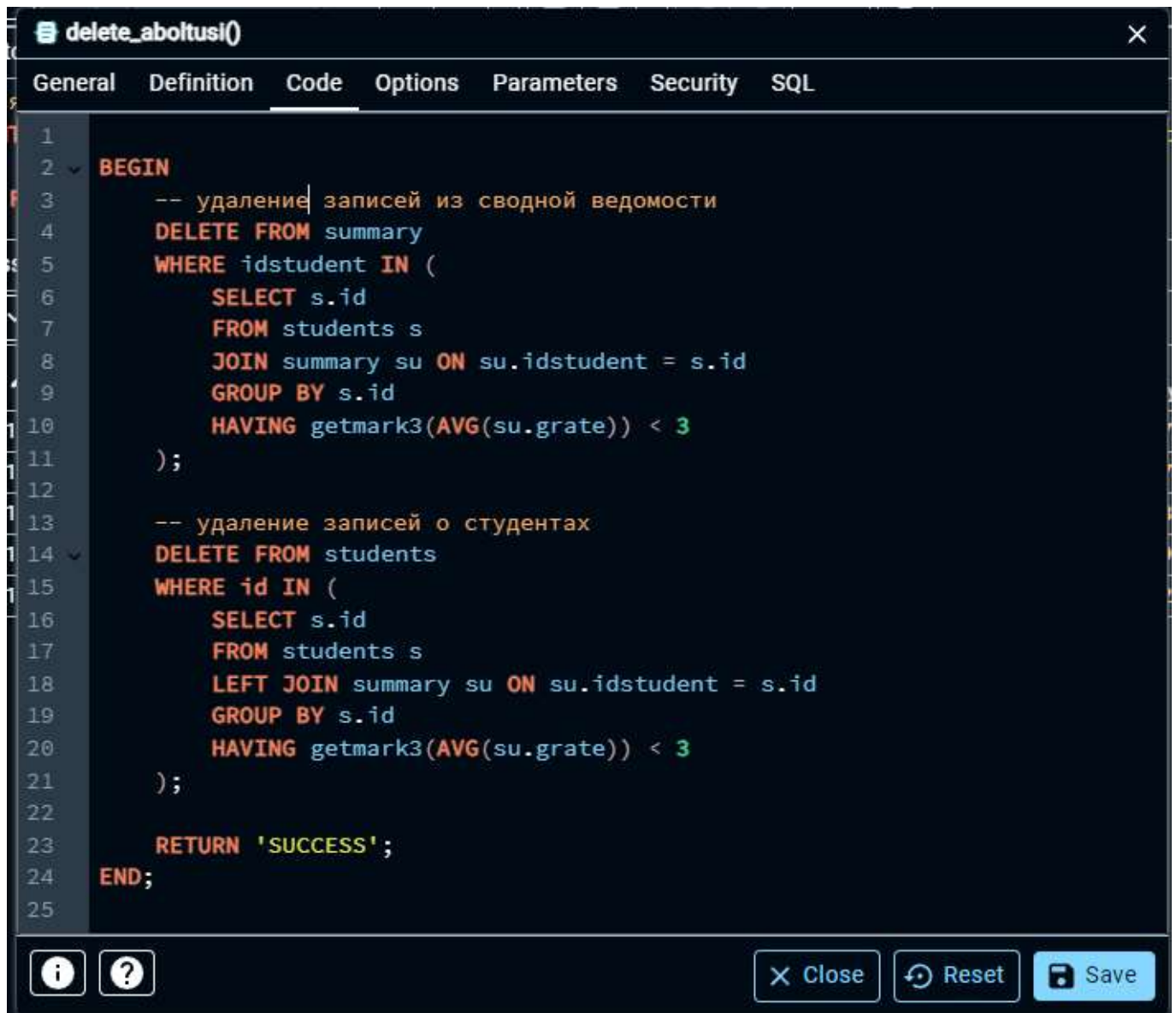
	idstudent [PK] smallint	avg_grate numeric	extended boolean
1	10	4	false
2	11	5	true
3	12	5	true
4	13	5	true

Рис. 6 Результат выполнения функции 1

**Процедура 2.** Пользовательская потребность: Необходимо очистить записи о студентах с плохой успеваемостью(средняя оценка меньше 3). Создал функцию для очистки необходимых записей из таблиц students(студенты) и



summary(сводная ведомость)(см. рисунок 7). Для проверки выполнения функции записал временного студента(см. рисунок 8), добавил оценки за экзамены(см. рисунок 9), выполнил функцию(см. рисунок 10). Результат представлен на рисунке 11.



```
1
2 BEGIN
3     -- удаление записей из сводной ведомости
4     DELETE FROM summary
5     WHERE idstudent IN (
6         SELECT s.id
7         FROM students s
8         JOIN summary su ON su.idstudent = s.id
9         GROUP BY s.id
10        HAVING getmark3(AVG(su.grate)) < 3
11    );
12
13    -- удаление записей о студентах
14    DELETE FROM students
15    WHERE id IN (
16        SELECT s.id
17        FROM students s
18        LEFT JOIN summary su ON su.idstudent = s.id
19        GROUP BY s.id
20        HAVING getmark3(AVG(su.grate)) < 3
21    );
22
23    RETURN 'SUCCESS';
24 END;
25
```

Рис. 7 Функция для очистки записей о студентах с плохой успеваемостью

session/postgres@local\* x session/postgres... x

session/postgres@local

Query Query History

```

1 -- добавляем временного студента
2 INSERT INTO students VALUES (14, 'NAME', 'SURNAME', 'PARENTNAME', '3', '876-32-45-123', 'addr', '2000-01-02');
3
4 SELECT * FROM students;
5

```

Data Output Messages Notifications

SQL

	id [PK] smallint	name character varying (50)	surname character varying (50)	parentname character varying (50)	groupnum character varying (15)	phonenum character varying (15)	address text	dateofbirth date
1	10	Игорь	Семенов	Алексеевич	1	899-66-77-778	ул. Лесная, д. 12	2002-02-15
2	11	Марина	Ковалева	Дмитриевна	2	911-88-77-776	ул. Зеленая, д. 8	2001-12-30
3	12	Петр	Иванов	Александрович	1	892-34-56-789	ул. Морская, д. 5	2000-03-14
4	13	Алина	Федорова	Сергеевна	2	893-45-67-890	ул. Лесная, д. 8	2001-04-25
5	14	NAME	SURNAME	PARENTNAME	3	876-32-45-123	addr	2000-01-02

Рис. 8 Добавление временного студента



session/postgres@local\* × session/postgres... ×

session/postgres@local

Query Query History

```

6 -- добавляем записи об экзаменах и оценках
7 INSERT INTO summary VALUES
8 ('2000-01-01',11,14,100,101),
9 ('2000-01-01',11,14,100,102),
10 ('2000-01-01',11,14,100,103),
11 ('2000-01-01',11,14,100,104),
12 ('2000-01-01',11,14,100,105),
13 ('2000-01-01',11,14,100,106);
14
15 SELECT * FROM summary;

```

Data Output Messages Notifications

	dateofexam date	grate smallint	ldstudent smallint	ldplan smallint	id [PK] smallint
1	2025-06-10	80	10	1	1
2	2025-06-15	100	11	2	2
3	2025-06-10	80	12	3	3
4	2025-06-15	100	13	4	4
5	2023-12-01	100	12	99	99
6	2023-12-01	100	13	100	100
7	2000-01-01	11	14	100	101
8	2000-01-01	11	14	100	102
9	2000-01-01	11	14	100	103
10	2000-01-01	11	14	100	104
11	2000-01-01	11	14	100	105
12	2000-01-01	11	14	100	106

Рис. 9 Добавление временных записей об успеваемости

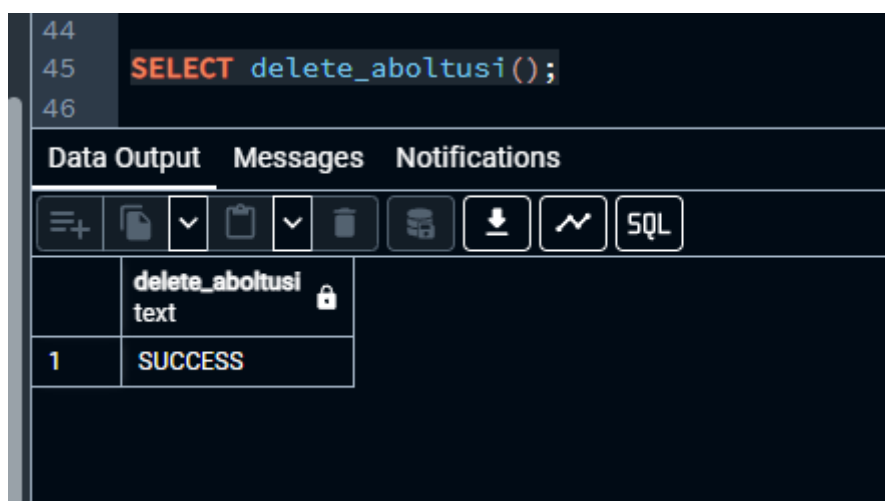


Рис. 10 Выполнение функции

Data Output Messages Notifications								
	id [PK] smallint	name character varying (50)	surname character varying (50)	parentname character varying (50)	groupnum character varying (15)	phonenum character varying (15)	address text	dateofbirth date
1	10	Игорь	Семенов	Александрович	1	899-66-77-778	ул. Лесная, д. 12	2002-02-15
2	11	Марина	Ковалева	Дмитриевна	2	911-88-77-776	ул. Зеленая, д. 8	2001-12-30
3	12	Петр	Иванов	Александрович	1	892-34-56-789	ул. Морская, д. 5	2000-03-14
4	13	Алина	Федорова	Сергеевна	2	893-45-67-890	ул. Лесная, д. 8	2001-04-25

Рис. 11 Результат выполнения

Вывод: в ходе выполнения практической работы сформировал навыки работы в среде СУБД PostgreSQL по созданию функций и процедур.