Бюджетное учреждение высшего образования Ханты-Мансийского автономного округа – Югры

«СУРГУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ

КАФЕДРА АВТОМАТИЗИРОВАННЫХ СИСТЕМ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ

ОТЧЕТ

ЛАБОРАТОРНАЯ РАБОТА №2

ПО ДИСЦИПЛИНЕ: «СОВРЕМЕННЫЕ ПРОМЫШЛЕННЫЕ СУБД»

Выполнил: студент группы №606-11

Ахметова Э. Р.

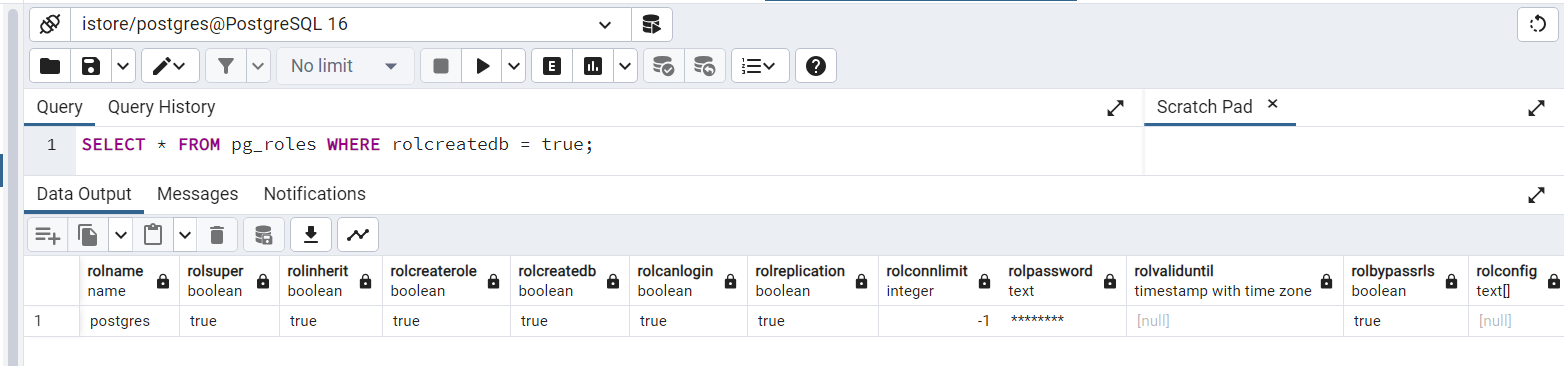
Проверил: ст. преподаватель

Гавриленко А. В.

Сургут 2024

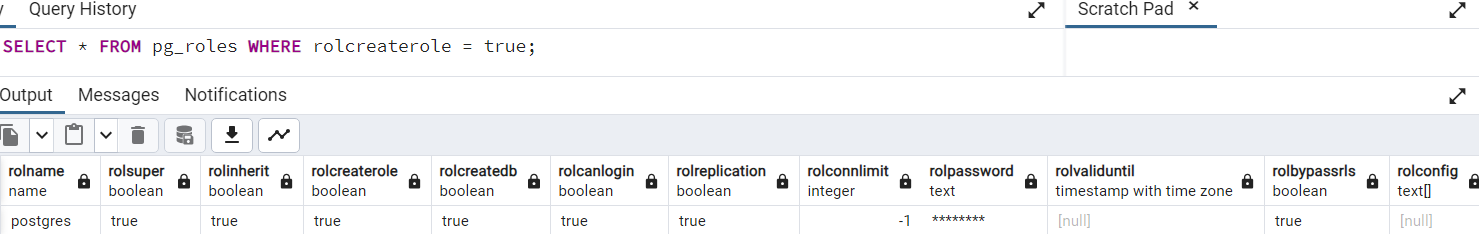
1. Определите всех пользователей, которые могут создавать базы данных (rolcreatedb).

SELECT \* FROM pg\_roles WHERE rolcreatedb = true;



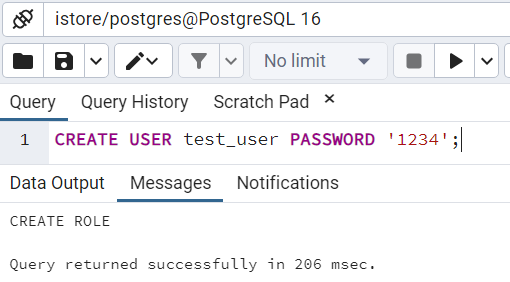
1. Определите всех пользователей, которые могут создавать роли (rolcreaterole).

SELECT \* FROM pg\_roles WHERE rolcreaterole = true;

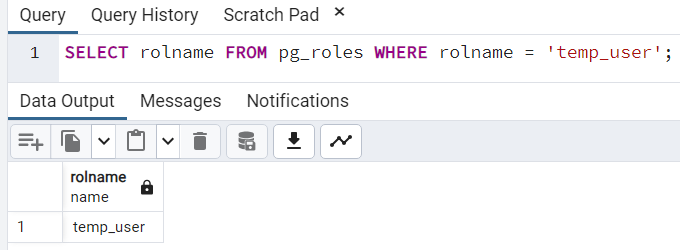


1. Создайте пользователя для входа с подключением к вашей базе данных (например, пользователя с именем connectDB). Докажите правильность выполненных действий.

CREATE USER test\_user PASSWORD '1234';

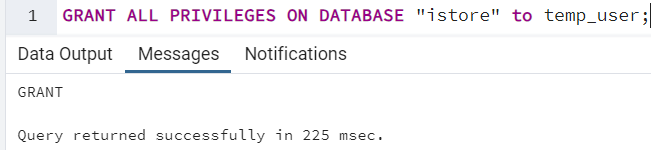


Доказательство:



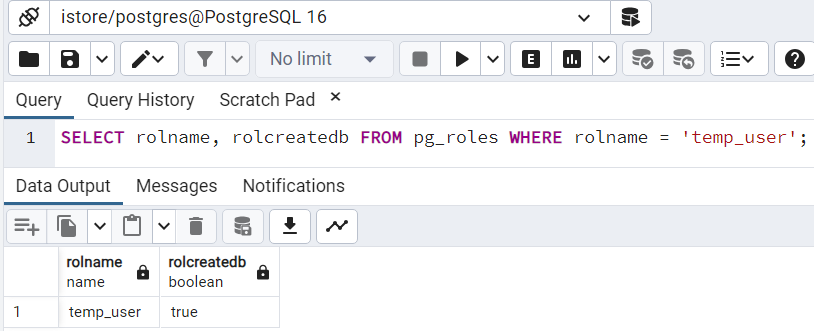
Созданному пользователю присвойте права на создание и изменение баз данных (права: createDB). Докажите правильность выполненных действий.

GRANT ALL PRIVILEGES ON DATABASE "istore" to temp\_user;

****

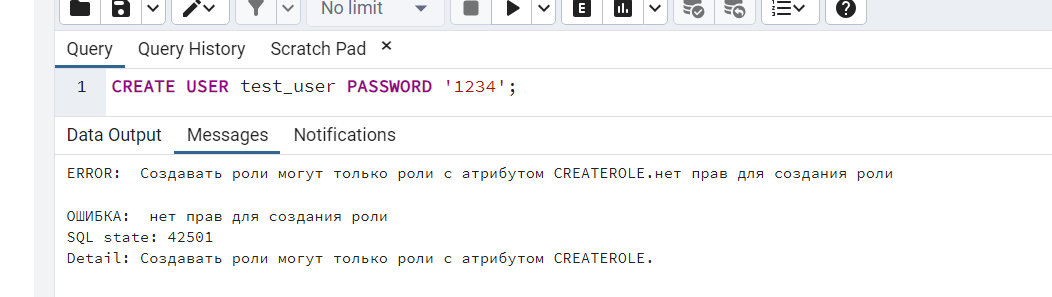
ALTER ROLE temp\_user CREATEDB;

Доказательство:



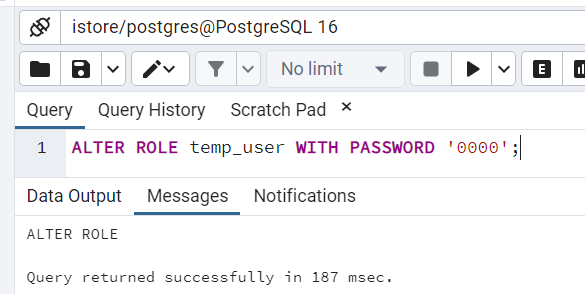
Подключитесь к серверу, используя созданную учетную запись (пользователь: connectDB), и создайте еще одну учетную запись пользователя для входа, результат объясните.

CREATE USER test\_user PASSWORD '1234';



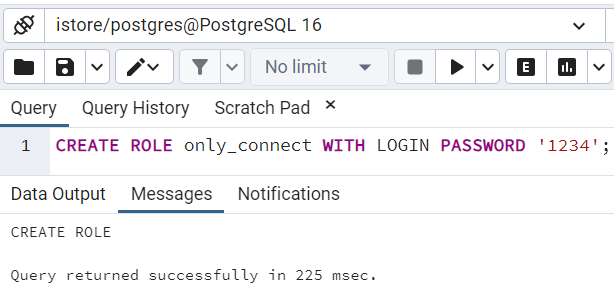
1. Для созданного пользователя (пользователь: connectDB) измените пароль.

ALTER ROLE temp\_user WITH PASSWORD '0000';



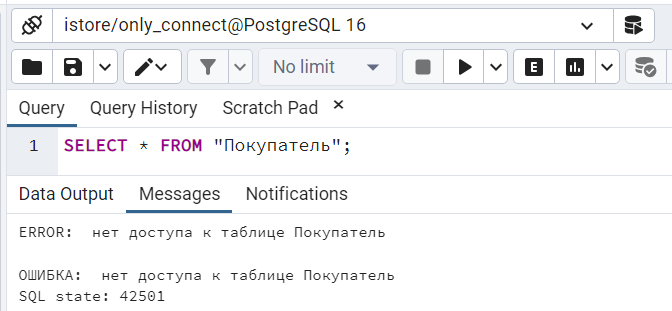
1. Создайте пользователя (например, пользователя с именем only\_connect) и присвойте ему права на подключение к серверу через запрос sql в окне запросника. Команду объяснить.

CREATE ROLE only\_connect WITH LOGIN PASSWORD '1234';



1. Подключитесь к серверу от лица пользователя (пользователь: only\_connect) и попробуйте выполнить запрос на просмотр данных в таблице. Результат объяснить. Замечание. Для проверки правильности выполненных действий можно выполнить произвольный запрос к этой базе данных, например, отображающий содержимое таблицы: SELECT \* FROM your\_table;

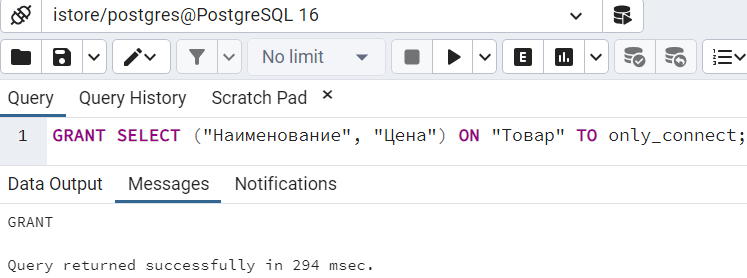
SELECT \* FROM "Покупатель";



Это происходит потому, что по умолчанию новый пользователь не имеет прав на чтение данных из таблиц.

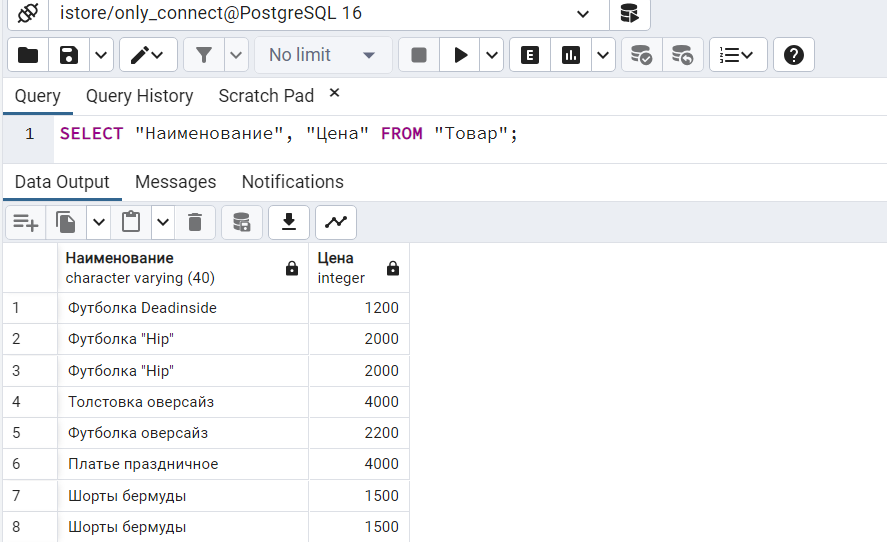
1. Добавьте таблицу (или возьмите свою таблицу) с тремя колонками: FirstColumn, SecondColumn, ThirdColumn. Типы не имеют значение. Добавьте несколько записей. Работая от лица админа, даём пользователю only\_connect право просматривать две колонки из таблицы (вашей таблицы).

GRANT SELECT ("Наименование", "Цена") ON "Товар" TO only\_connect;



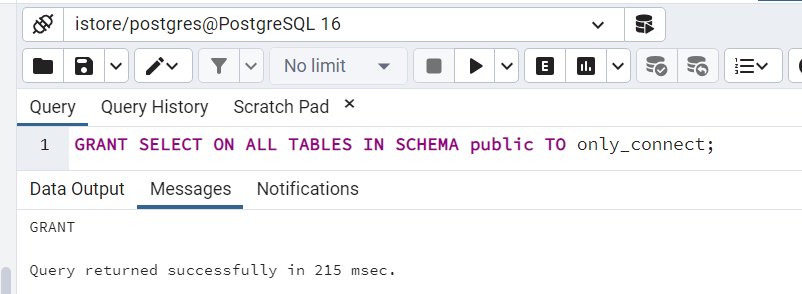
Проверить, получается ли читать данные в этих двух колонках/читать данные всех колонок таблицы. Результат объяснить.

SELECT "Наименование", "Цена" FROM "Товар";



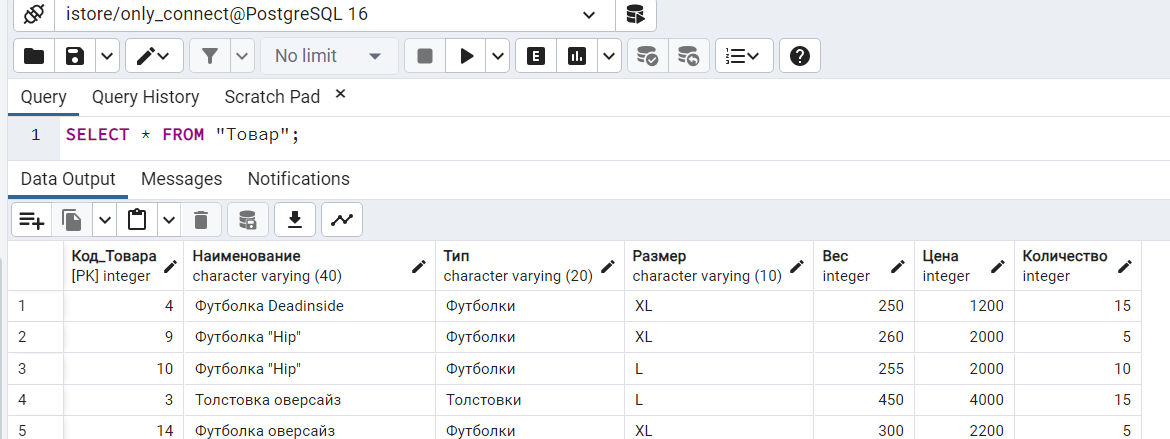
1. Выдайте пользователю only\_connect права на чтение данных во всей бд. Повторите запрос sql на выдачу данных к другой таблице. Результат объяснить.

GRANT SELECT ON ALL TABLES IN SCHEMA public TO only\_connect;



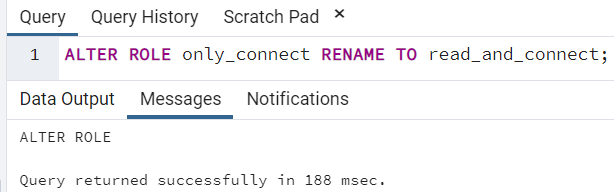
Проверка:

SELECT \* FROM "Товар";



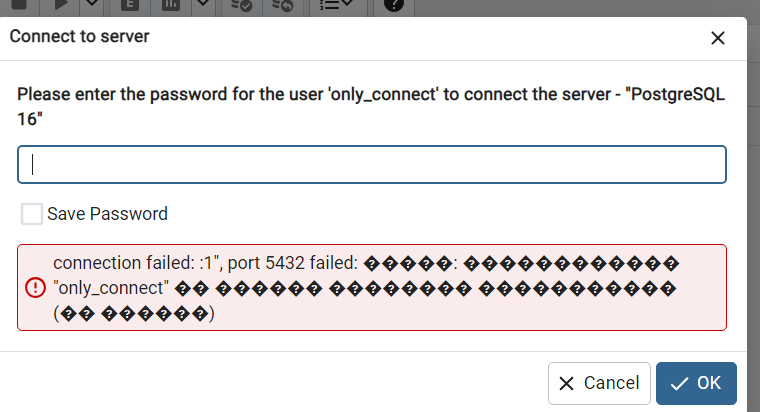
1. Переименуйте пользователя (пользователь: only\_connect) в пользователя (например: read\_and\_connect).

ALTER ROLE only\_connect RENAME TO read\_and\_connect;



1. Попробуйте снова подключиться к серверу. Результат объяснить.

Повторное подключение недоступно поскольку пользователя с этим именем больше не существует.



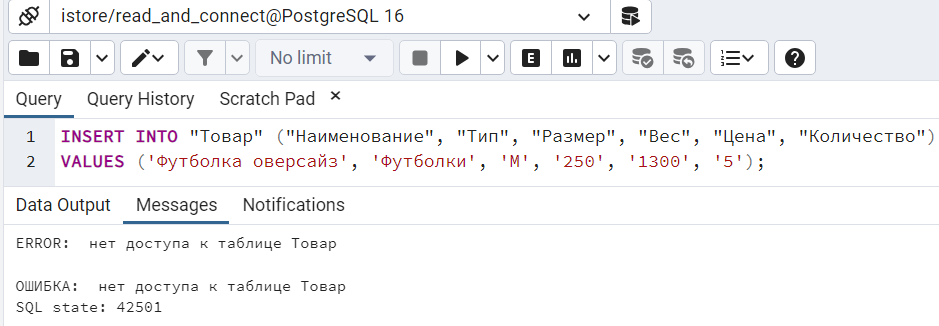
1. Настройте новое подключение к серверу для пользователя read\_and\_connect.

Подключение настроено через интерфейс.



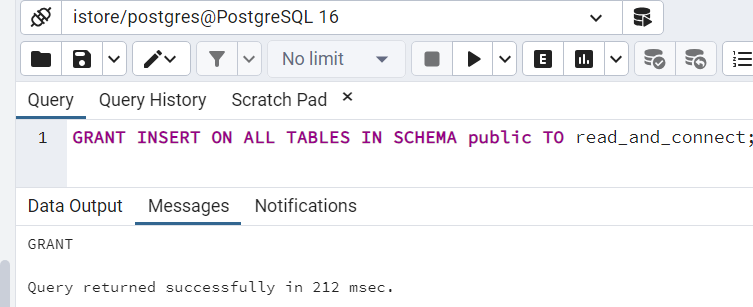
1. От пользователя read\_and\_connect сделайте запрос (или через графический интерфейс) на добавление новых данных в сущность. Результат объяснить. Замечание. Для проверки правильности выполненных действий можно выполнить произвольный запрос к этой базе данных: INSERT INTO your\_table (column) VALUES (value);

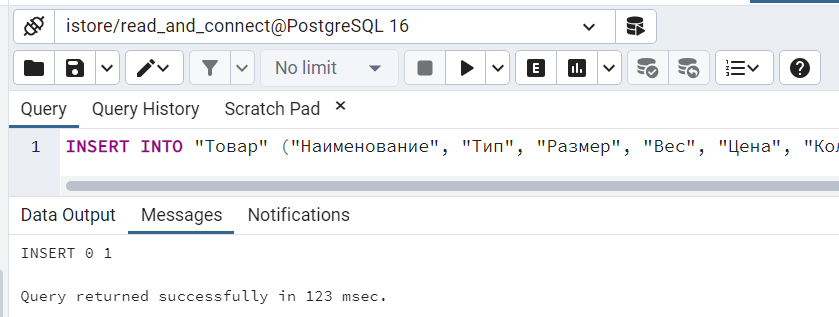
INSERT INTO "Товар" ("Наименование", "Тип", "Размер", "Вес", "Цена", "Количество") VALUES ('Футболка оверсайз', 'Футболки', 'M', '250', '1300', '5');



1. Добавьте права на вставку (INSERT) данных пользователю (пользователь: read\_and\_connect). Повторите запрос из 12 пункта.

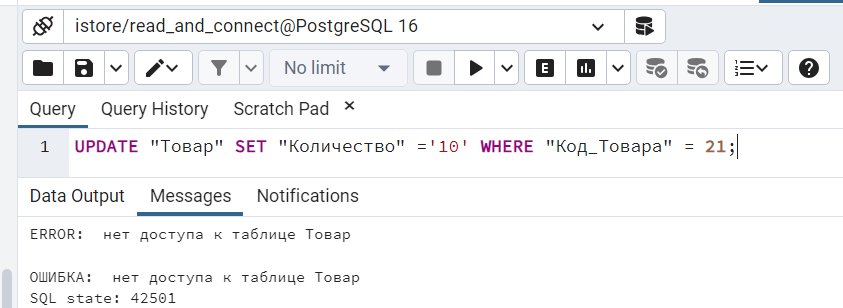
GRANT INSERT ON ALL TABLES IN SCHEMA public TO read\_and\_connect;





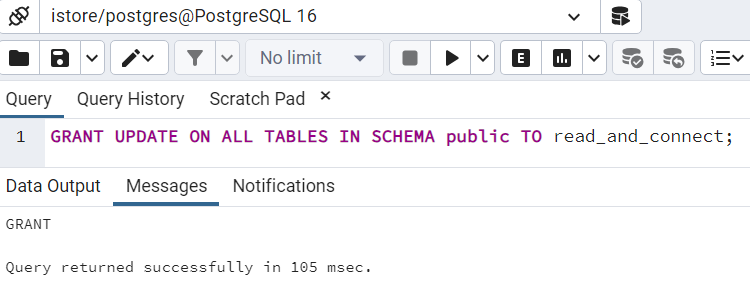
1. Обновите вставленные данные (UPDATE) в сущности из пункта 13. Объясните результат. Замечание. Для проверки правильности выполненных действий можно выполнить произвольный запрос к этой базе данных: UPDATE your\_table SET column = 'value WHERE column\_2 = value\_2;

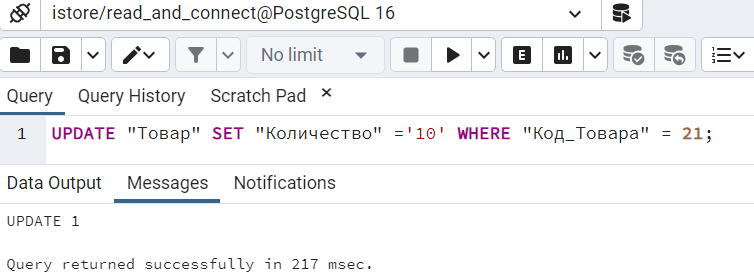
UPDATE "Товар" SET "Количество" ='10' WHERE "Код\_Товара" = 21;



1. Добавьте права на обновление (UPDATE) пользователю read\_and\_connect. Повторите пункт 14.

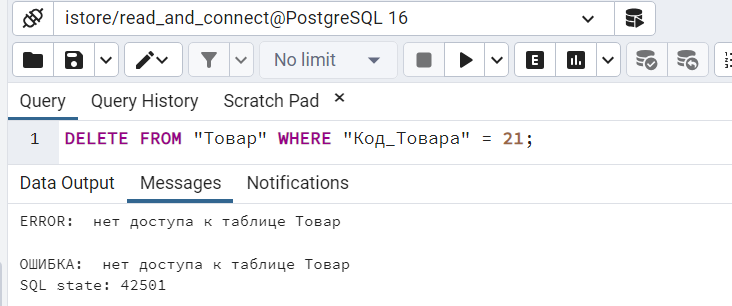
GRANT UPDATE ON ALL TABLES IN SCHEMA public TO read\_and\_connect;





1. Удалите вставленные данные (DELETE) в сущность из пункта 15. Объясните результат. Замечание. Для проверки правильности выполненных действий можно выполнить произвольный запрос к этой базе данных: DELETE FROM your\_table WHERE column = value;

DELETE FROM "Товар" WHERE "Код\_Товара" = 21;



17. Добавьте права на удаление (DELETE) пользователю read\_and\_connect. Повторите пункт 16.

GRANT DELETE ON ALL TABLES IN SCHEMA public TO read\_and\_connect;

