**1. Функции и процедуры для выполнения CRUD-операций (общая база данных).**

1.1 create or replace procedure pr\_add\_ сustomers(new\_id integer,name varchar(50))

language plpgsql as $$

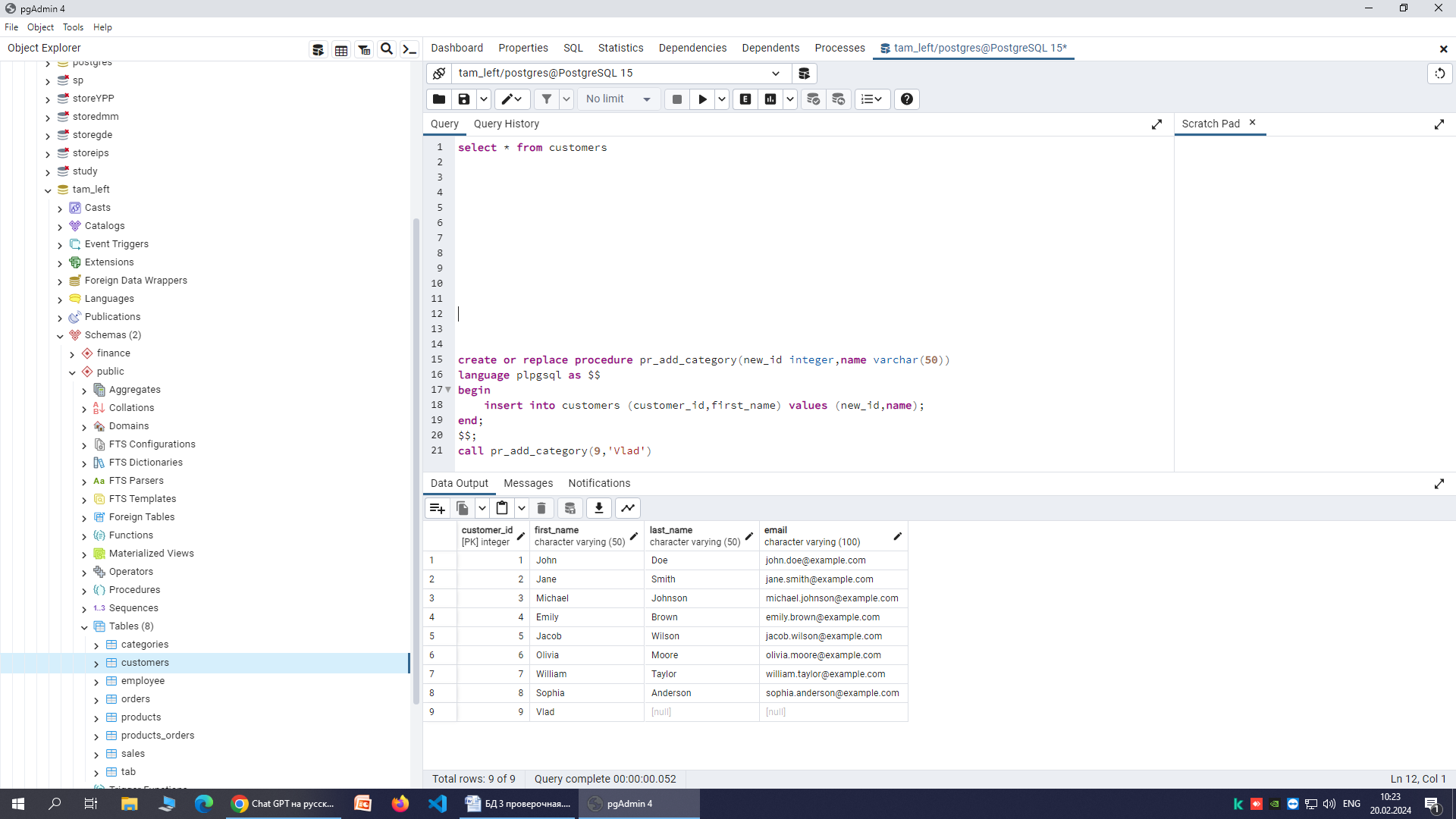
begin

insert into сustomers (customer\_id,first\_name) values (new\_id,name);

end;

$$;

call pr\_add\_category(9,'Vlad')



create or replace procedure pr\_update\_сustomers(cur\_id integer, name\_new varchar(50))

language plpgsql as $$

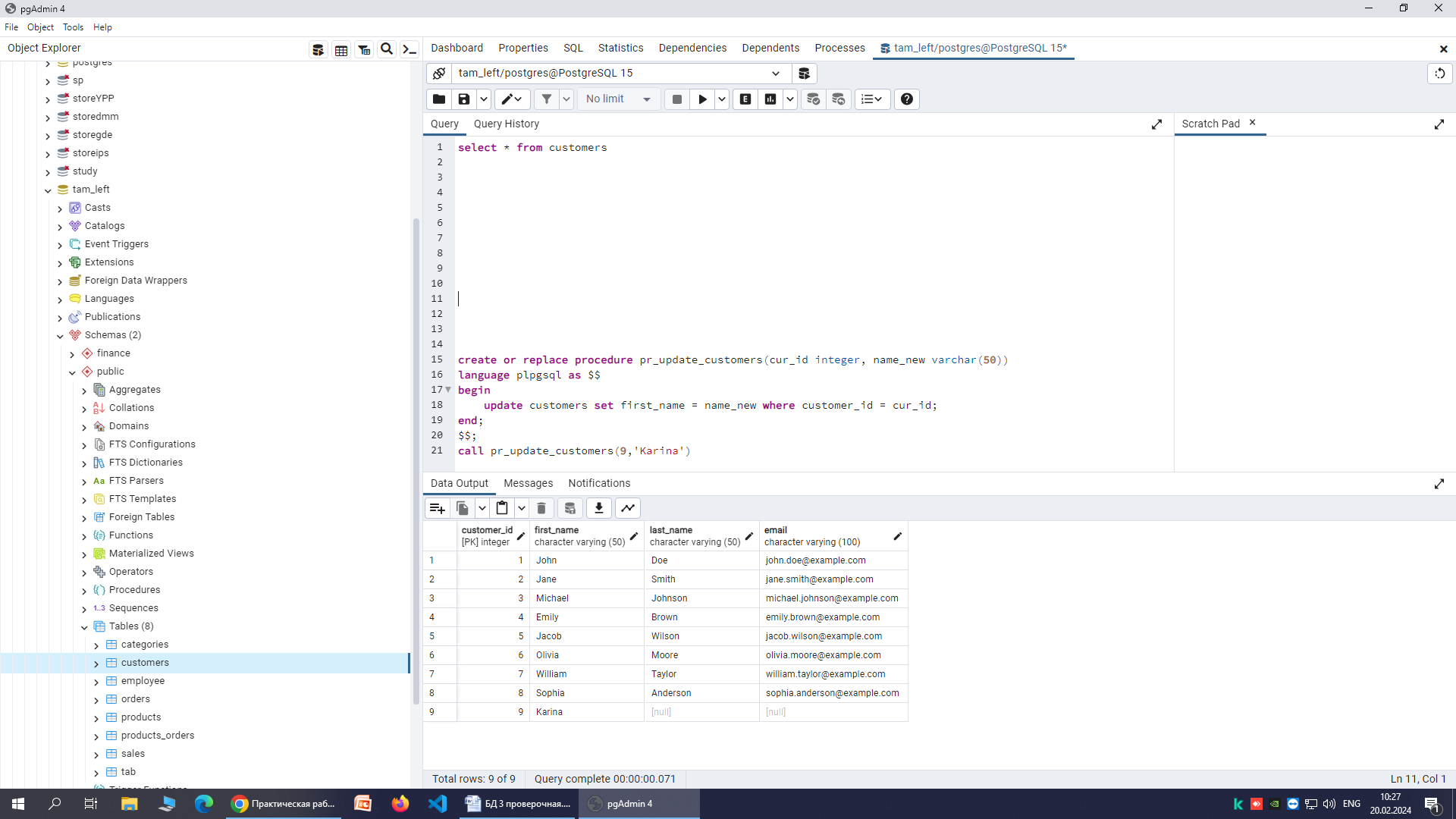
begin

update сustomers set first\_name = name\_new where customer\_id = cur\_id;

end;

$$;

call pr\_update\_сustomers(9,'Karina')



create or replace procedure pr\_delete\_сustomers(cur\_id integer)

language plpgsql as $$

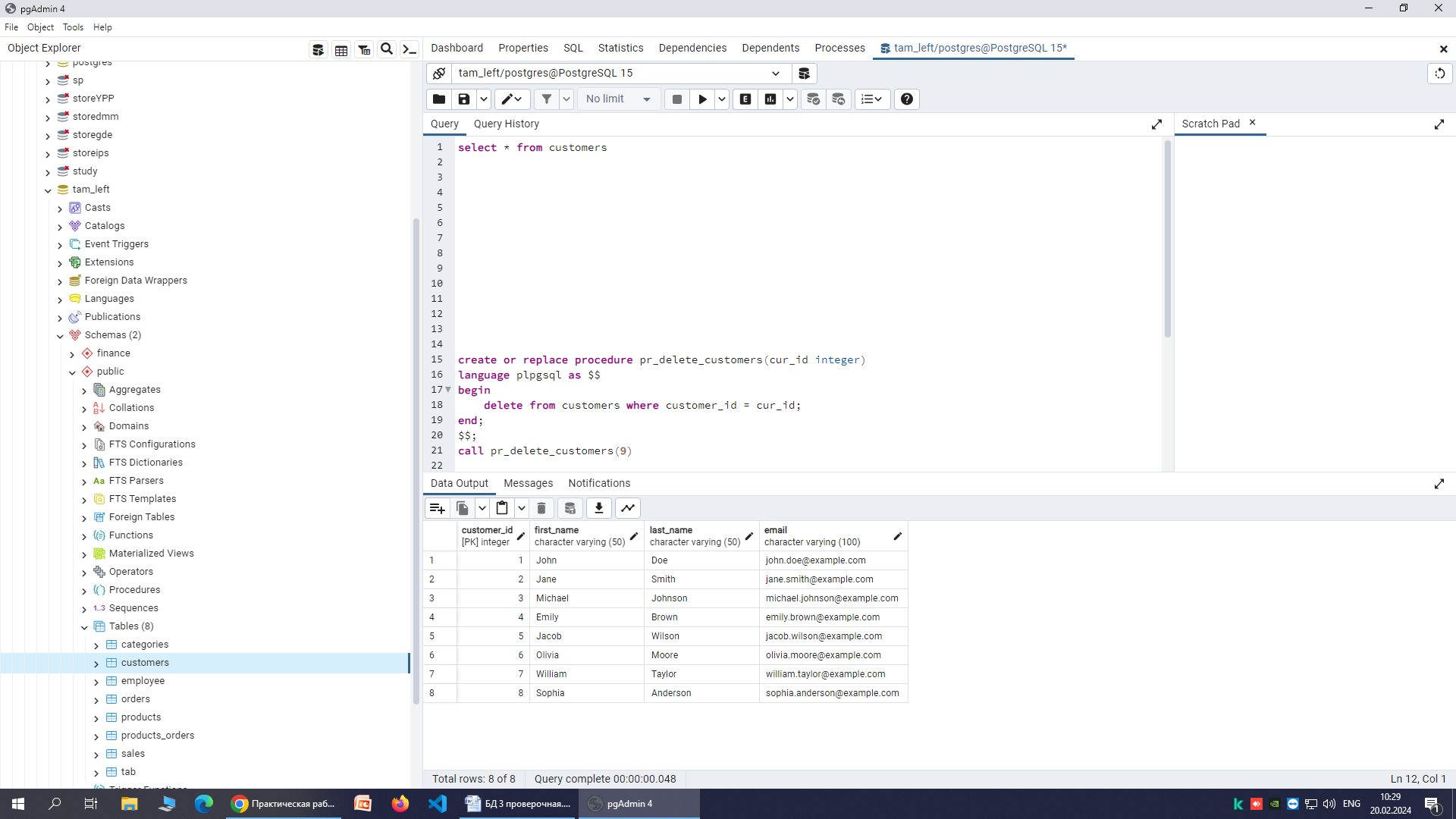
begin

delete from сustomers where customer\_id = cur\_id;

end;

$$;

call pr\_delete\_сustomers(9)



1.2 create or replace function f\_add\_products(id integer,name varchar(50))

returns integer as $$

begin

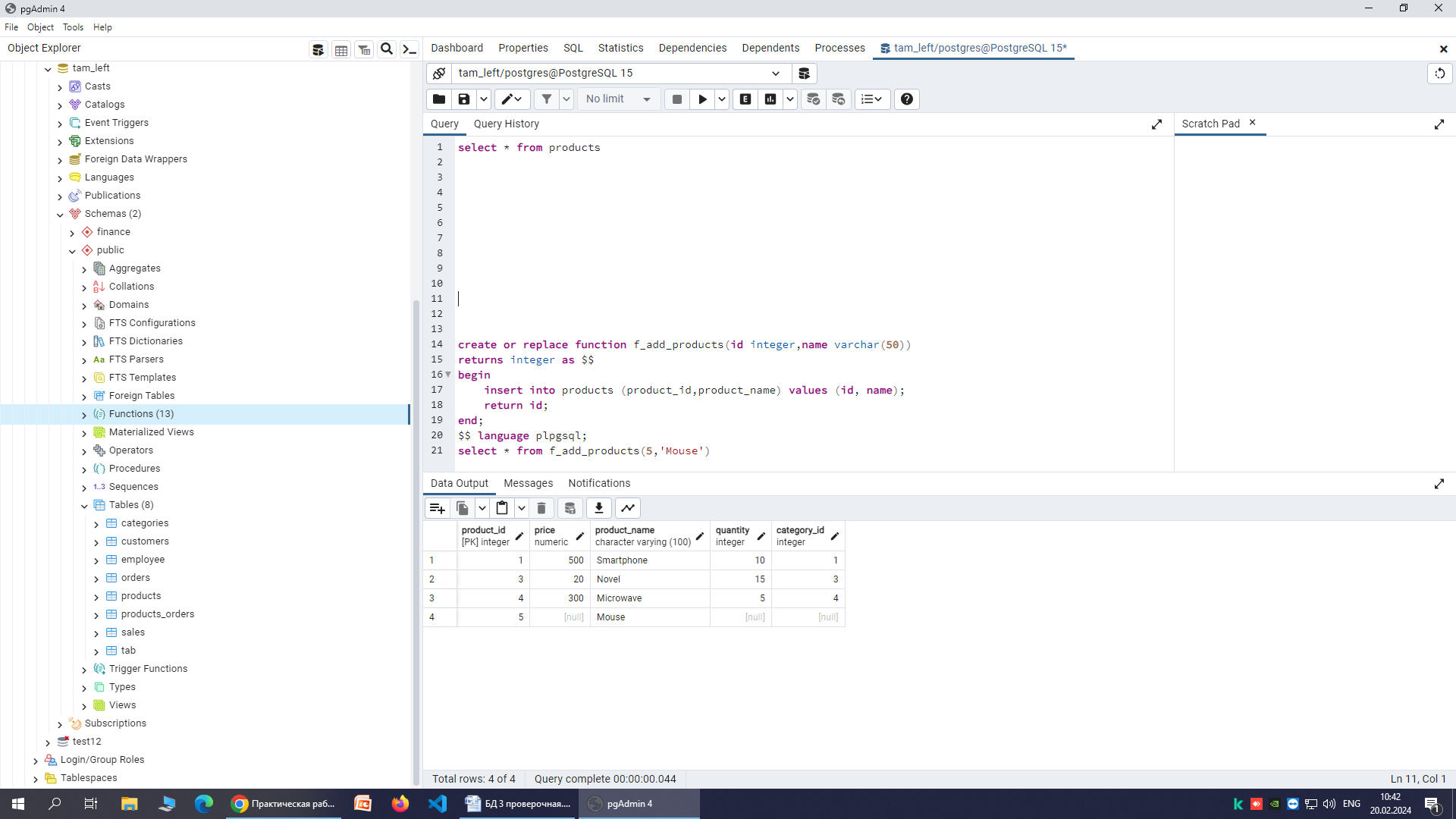
insert into products (product\_id,product\_name) values (id, name);

return id;

end;

$$ language plpgsql;

select \* from f\_add\_products(5,'Mouse')



create or replace function f\_update\_products(cur\_id integer, name\_new varchar(50))

returns integer as $$

begin

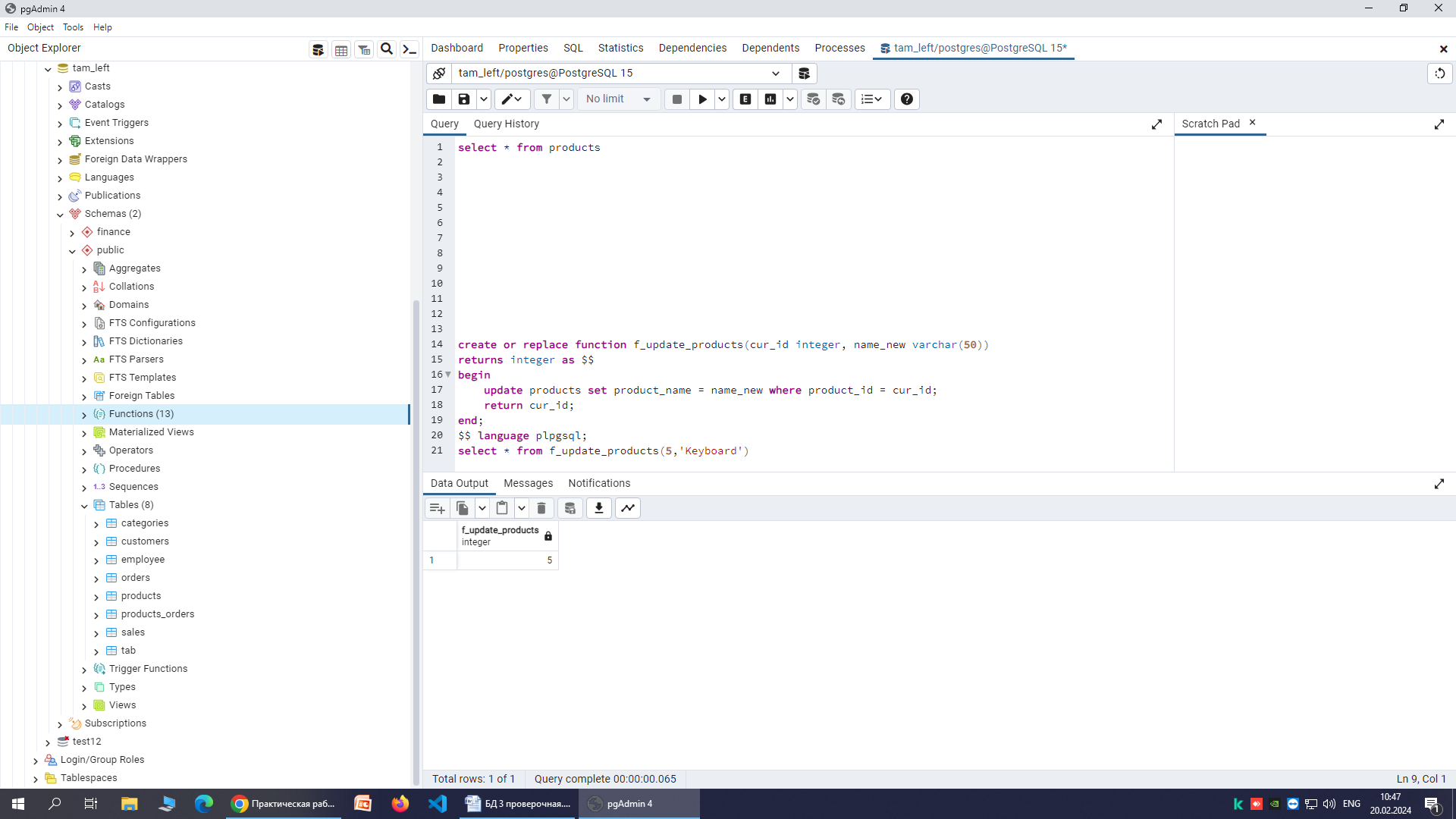
update products set product\_name = name\_new where product\_id = cur\_id;

return product\_id;

end;

$$ language plpgsql;

select \* from f\_update\_products(5,'Keyboard')



create or replace function f\_delete\_products(cur\_id integer)

returns integer as $$

begin

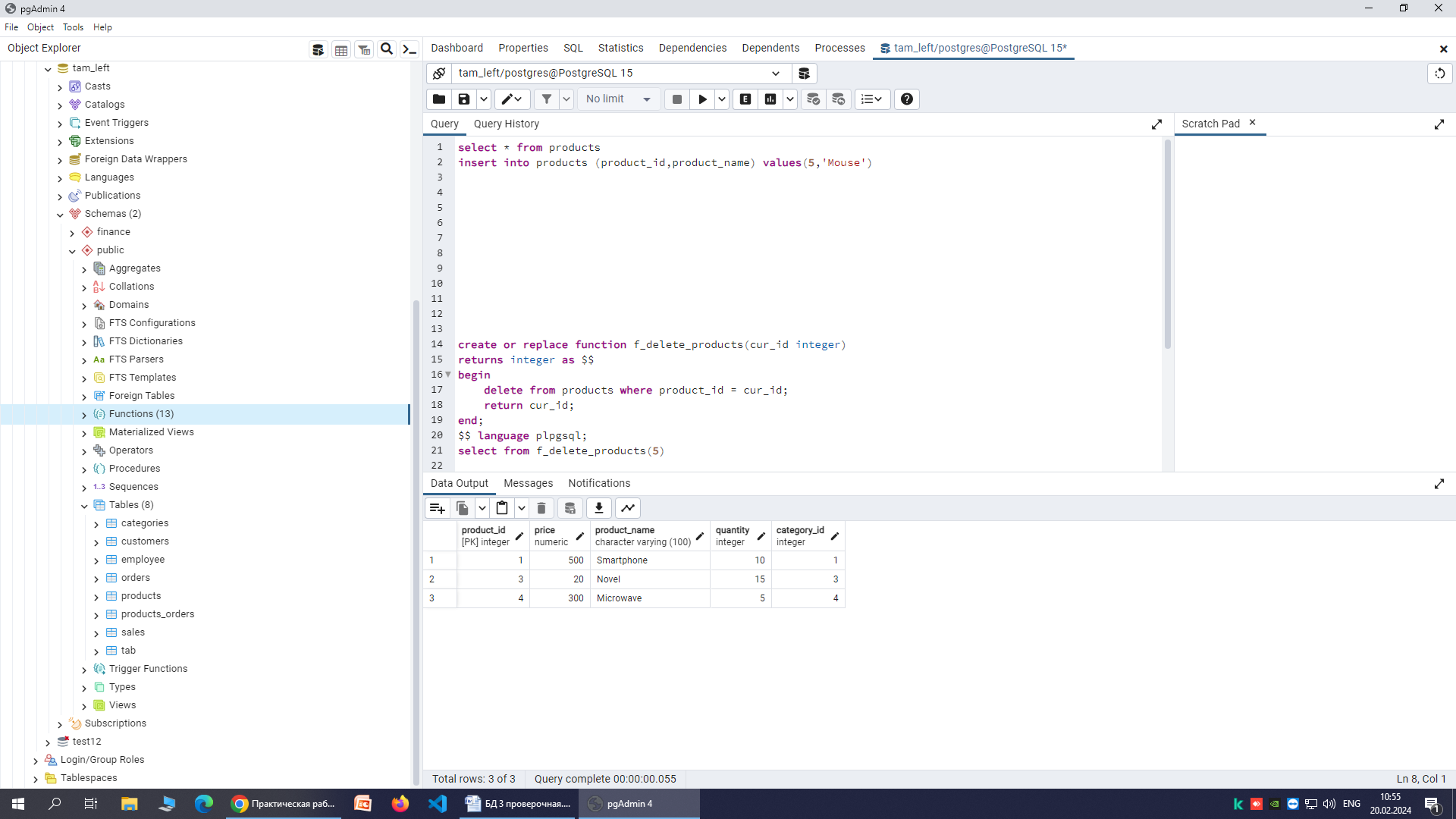
delete from products where product\_id = cur\_id;

return cur\_id;

end;

$$ language plpgsql;

select from f\_delete\_products(5)



create or replace function f\_select\_category(cur\_id integer)

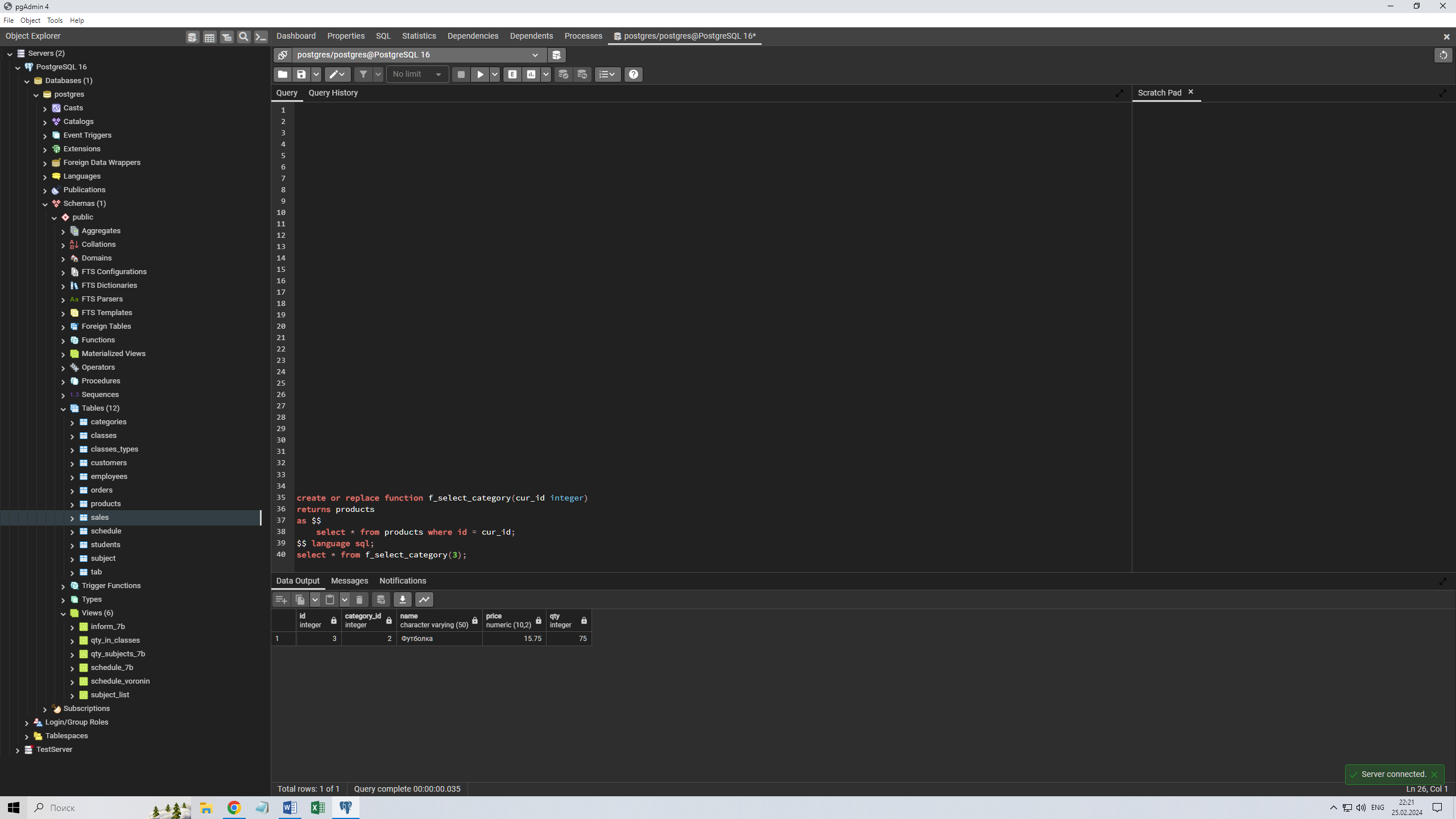
returns products

as $$

select \* from products where id = cur\_id;

$$ language sql;

select \* from f\_select\_category(3);



**2. Одиночные курсоры**

2.1 create or replace function get\_cursor (curs1 refcursor)

returns refcursor as $$

begin

open $1 for select \* from categories;

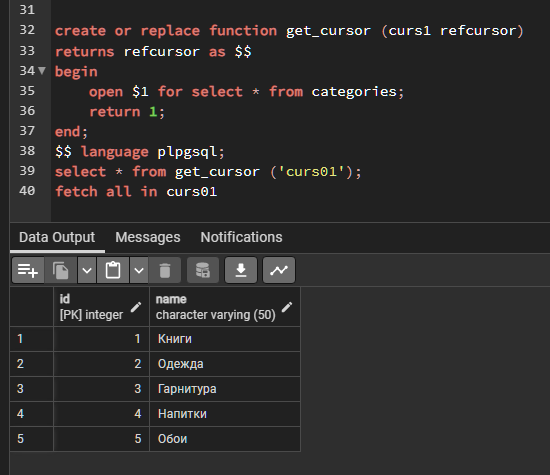
return $1;

end;

$$ language plpgsql;

select \* from get\_cursor ('curs01');

fetch all in curs01



2.2 create or replace function difcursor()

returns varchar(50) as $$

declare

curs1 cursor for select name from products where category\_id=3;

nm varchar(50);

begin

open curs1;

RAISE NOTICE 'Товары с категорией 3:';

loop

fetch curs1 into nm;

EXIT WHEN NOT FOUND;

RAISE NOTICE '%', nm;

return next nm;

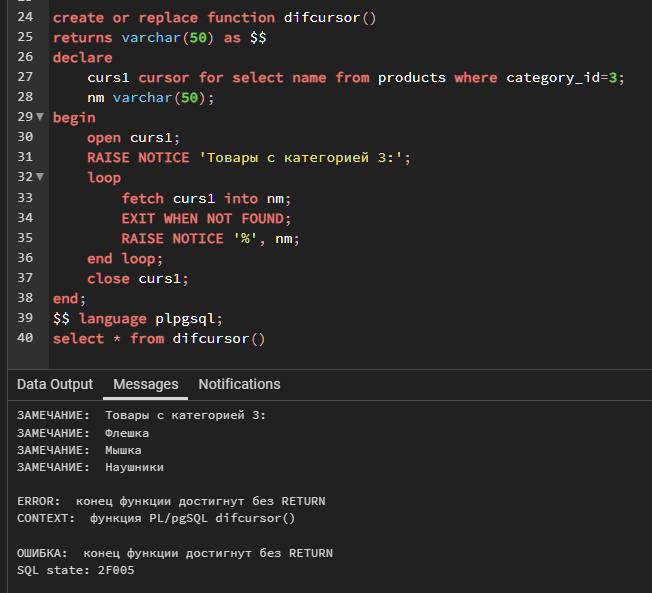
end loop;

close curs1;

end;

$$ language plpgsql;

select \* from difcursor()



2.3 create or replace function get\_cursor3()

returns numeric(15,2) as $$

declare

curs1 cursor for select dis\_amount from orders where datee='2023-10-01';

pay numeric(15,2):=0;

summ numeric(15,2):=0;

begin

open curs1;

loop

fetch curs1 into pay;

EXIT WHEN NOT FOUND;

summ:=summ+pay;

end loop;

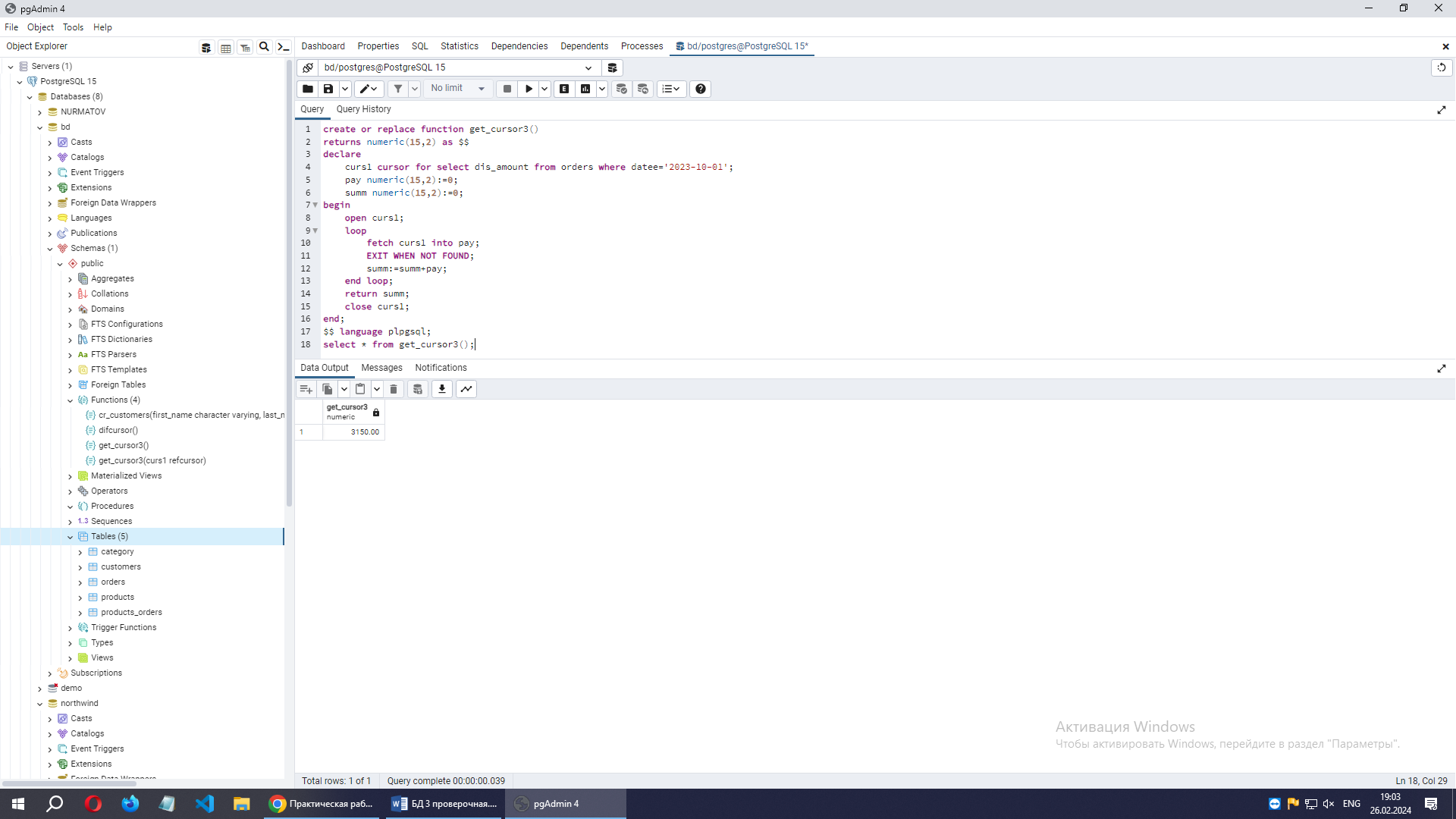
return summ;

close curs1;

end;

$$ language plpgsql;

select \* from get\_cursor3();



2.4

create or replace procedure get\_cursor\_p(out refcursor, out refcursor)

as $$

begin

open $1 for SELECT \* FROM clients WHERE phone LIKE '\_951%';

open $2 for SELECT \* FROM clients WHERE phone LIKE '\_982%';

end;

$$ language plpgsql;

do $$

declare

a refcursor:='a';

b refcursor:='b';

tmpa record;

tmpb record;

begin

call get\_cursor\_p(a,b);

fetch from a into tmpa;

raise notice 'Номера на 951:';

while (found) loop

raise notice '%', tmpa;

fetch from a into tmpa;

end loop;

fetch from b into tmpb;

raise notice 'Номера на 982:';

while (found) loop

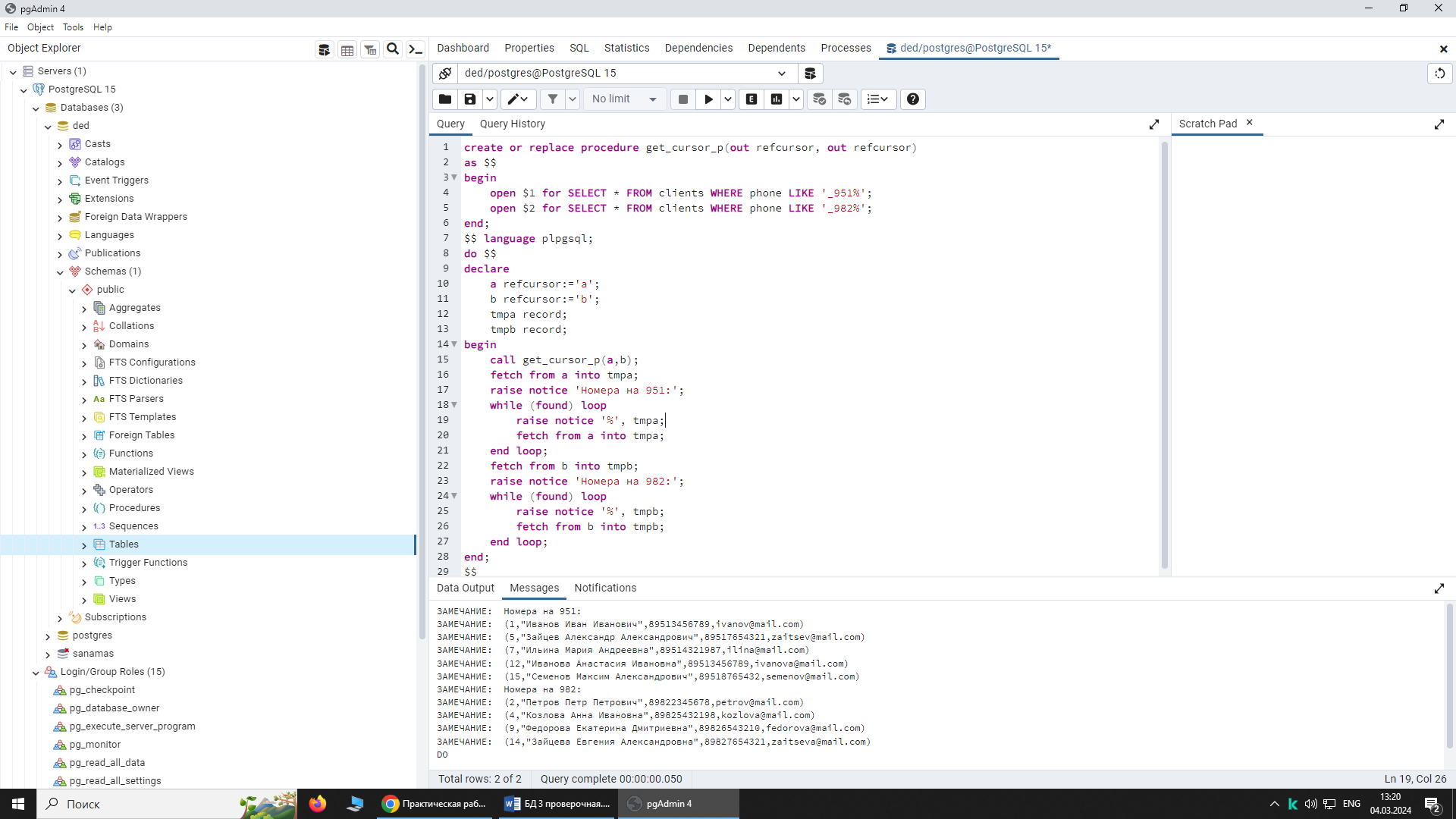
raise notice '%', tmpb;

fetch from b into tmpb;

end loop;

end;

$$



2.5 create or replace procedure get\_cursor\_2\_5()

as $$

declare

curs1 cursor for SELECT \* FROM clients WHERE phone LIKE '\_91%' for update;

tmpa record;

begin

raise notice 'Номера на 917:';

open curs1;

loop

fetch from curs1 into tmpa;

EXIT WHEN NOT FOUND;

if (tmpa.phone like '\_917%') then

UPDATE clients SET phone = CONCAT(SUBSTRING(phone, 1,1),'982', SUBSTRING(phone,5)) WHERE CURRENT OF curs1;

raise notice '%', tmpa.id;

else delete from clients where CURRENT OF curs1;

end if;

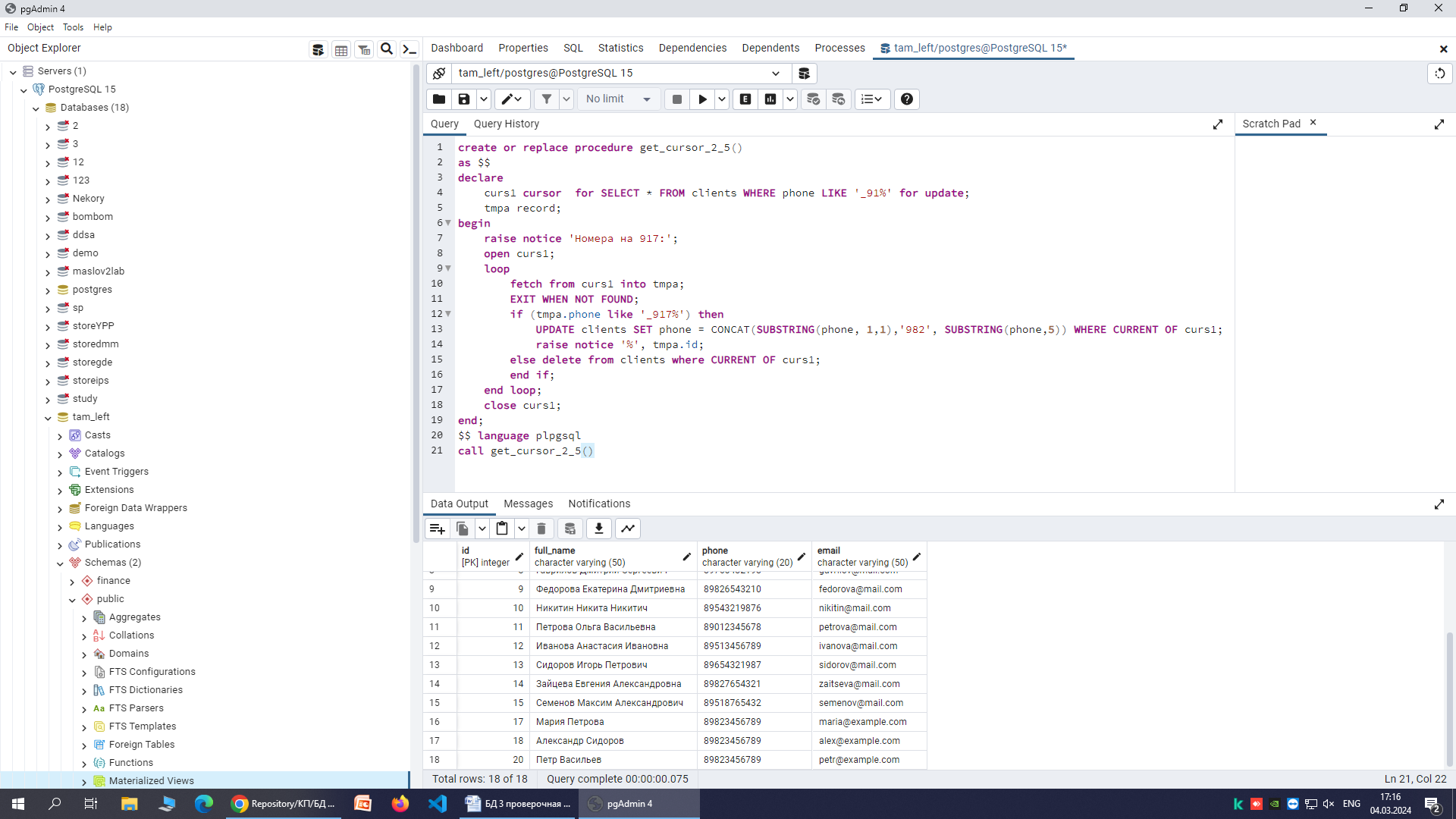
end loop;

close curs1;

end;

$$ language plpgsql

call get\_cursor\_2\_5()



2.6 и 2.7

begin;

select \* from curs();

fetch from qwerty ;

fetch next in qwerty;

fetch ABSOLUTE -1 in qwerty;

fetch PRIOR in qwerty;

fetch ABSOLUTE 10 in qwerty;

move ABSOLUTE 6 in qwerty;

fetch BACKWARD in qwerty;

fetch RELATIVE 2 in qwerty;

move BACKWARD 1 in qwerty;

fetch BACKWARD 3 in qwerty;

move forward 3 in qwerty;

fetch forward 4 in qwerty;

end;

CREATE TABLE сustomers (

customer\_ID INT PRIMARY KEY,

first\_Name VARCHAR(50),

last\_Name VARCHAR(50),

email VARCHAR(100)

)

INSERT INTO сustomers VALUES

(1, 'John', 'Doe', 'john.doe@example.com'),

(2, 'Jane', 'Smith', 'jane.smith@example.com'),

(3, 'Michael', 'Johnson', 'michael.johnson@example.com'),

(4, 'Emily', 'Brown', 'emily.brown@example.com'),

(5, 'Jacob', 'Wilson', 'jacob.wilson@example.com'),

(6, 'Olivia', 'Moore', 'olivia.moore@example.com'),

(7, 'William', 'Taylor', 'william.taylor@example.com'),

(8, 'Sophia', 'Anderson', 'sophia.anderson@example.com');

CREATE TABLE clients (

id INT PRIMARY KEY,

full\_name VARCHAR(50),

phone VARCHAR(20),

email VARCHAR(50)

);

delete from clients

INSERT INTO clients VALUES

(1, 'Иванов Иван Иванович', '89513456789', 'ivanov@mail.com'),

(2, 'Петров Петр Петрович', '89822345678', 'petrov@mail.com'),

(3, 'Сидорова Ольга Петровна', '89654321987', 'sidorova@mail.com'),

(4, 'Козлова Анна Ивановна', '89825432198', 'kozlova@mail.com'),

(5, 'Зайцев Александр Александрович', '89517654321', 'zaitsev@mail.com'),

(6, 'Семенова Елена Владимировна', '89098765432', 'semenova@mail.com'),

(7, 'Ильина Мария Андреевна', '89514321987', 'ilina@mail.com'),

(8, 'Гаврилов Дмитрий Сергеевич', '89765432198', 'gavrilov@mail.com'),

(9, 'Федорова Екатерина Дмитриевна', '89826543210', 'fedorova@mail.com'),

(10, 'Никитин Никита Никитич', '89543219876', 'nikitin@mail.com'),

(11, 'Петрова Ольга Васильевна', '89012345678', 'petrova@mail.com'),

(12, 'Иванова Анастасия Ивановна', '89513456789', 'ivanova@mail.com'),

(13, 'Сидоров Игорь Петрович', '89654321987', 'sidorov@mail.com'),

(14, 'Зайцева Евгения Александровна', '89827654321', 'zaitseva@mail.com'),

(15, 'Семенов Максим Александрович', '89518765432', 'semenov@mail.com');

INSERT INTO clients (ID, full\_name, phone, email) VALUES (16, 'Иван Иванов', '89134556789', 'ivan@example.com');

INSERT INTO clients (ID, full\_name, phone, email) VALUES (17, 'Мария Петрова', '89173456789', 'maria@example.com');

INSERT INTO clients (ID, full\_name, phone, email) VALUES (18, 'Александр Сидоров', '89173456789', 'alex@example.com');

INSERT INTO clients (ID, full\_name, phone, email) VALUES (19, 'Екатерина Николаева', '89151345789', 'ekaterina@example.com');

INSERT INTO clients (ID, full\_name, phone, email) VALUES (20, 'Петр Васильев', '89173456789', 'petr@example.com');

CREATE TABLE employees (

id INT PRIMARY KEY,

surname VARCHAR(50),

name VARCHAR(50),

patronymic VARCHAR(50),

phone VARCHAR(15),

mail VARCHAR(100)

);

INSERT INTO employees VALUES

(1, 'Иванов', 'Иван', 'Иванович', '+79123456789', 'ivanov@example.com'),

(2, 'Петров', 'Петр', 'Петрович', '+79234567890', 'petrov@example.com'),

(3, 'Сидоров', 'Сидор', 'Сидорович', '+79345678901', 'sidorov@example.com'),

(4, 'Козлов', 'Александр', 'Александрович', '+79456789012', 'kozlov@example.com'),

(5, 'Николаев', 'Николай', 'Николаевич', '+79567890123', 'nikolaev@example.com'),

(6, 'Андреева', 'Мария', 'Ивановна', '+79678901234', 'andreeva@example.com'),

(7, 'Смирнова', 'Елена', 'Александровна', '+79789012345', 'smirnova@example.com'),

(8, 'Васильев', 'Василий', 'Дмитриевич', '+79890123456', 'vasiliev@example.com'),

(9, 'Павлов', 'Павел', 'Сергеевич', '+79901234567', 'pavlov@example.com'),

(10, 'Кузнецов', 'Игорь', 'Игоревич', '+79912345678', 'kuznetsov@example.com'),

(11, 'Михайлов', 'Михаил', 'Михайлович', '+79923456789', 'mikhailov@example.com'),

(12, 'Федоров', 'Алексей', 'Алексеевич', '+79934567890', 'fedorov@example.com'),

(13, 'Сергеев', 'Сергей', 'Сергеевич', '+79945678901', 'sergeev@example.com'),

(14, 'Александрова', 'Ольга', 'Ивановна', '+79956789012', 'alexandrova@example.com'),

(15, 'Яковлев', 'Яков', 'Яковлевич', '+79967890123', 'yakovlev@example.com'),

(16, 'Максимова', 'Анна', 'Игоревна', '+79978901234', 'maximova@example.com'),

(17, 'Антонов', 'Антон', 'Антонович', '+79989012345', 'antonov@example.com'),

(18, 'Григорьева', 'Елизавета', 'Петровна', '+79990123456', 'grigoryeva@example.com'),

(19, 'Денисов', 'Денис', 'Денисович', '+79901234567', 'denisov@example.com'),

(20, 'Филиппова', 'Екатерина', 'Игоревна', '+79912345678', 'filippova@example.com');