**1. Общие функции**

1.2 CREATE OR REPLACE FUNCTION random (a int, b int)

RETURNS int AS $$

DECLARE

res int;

BEGIN

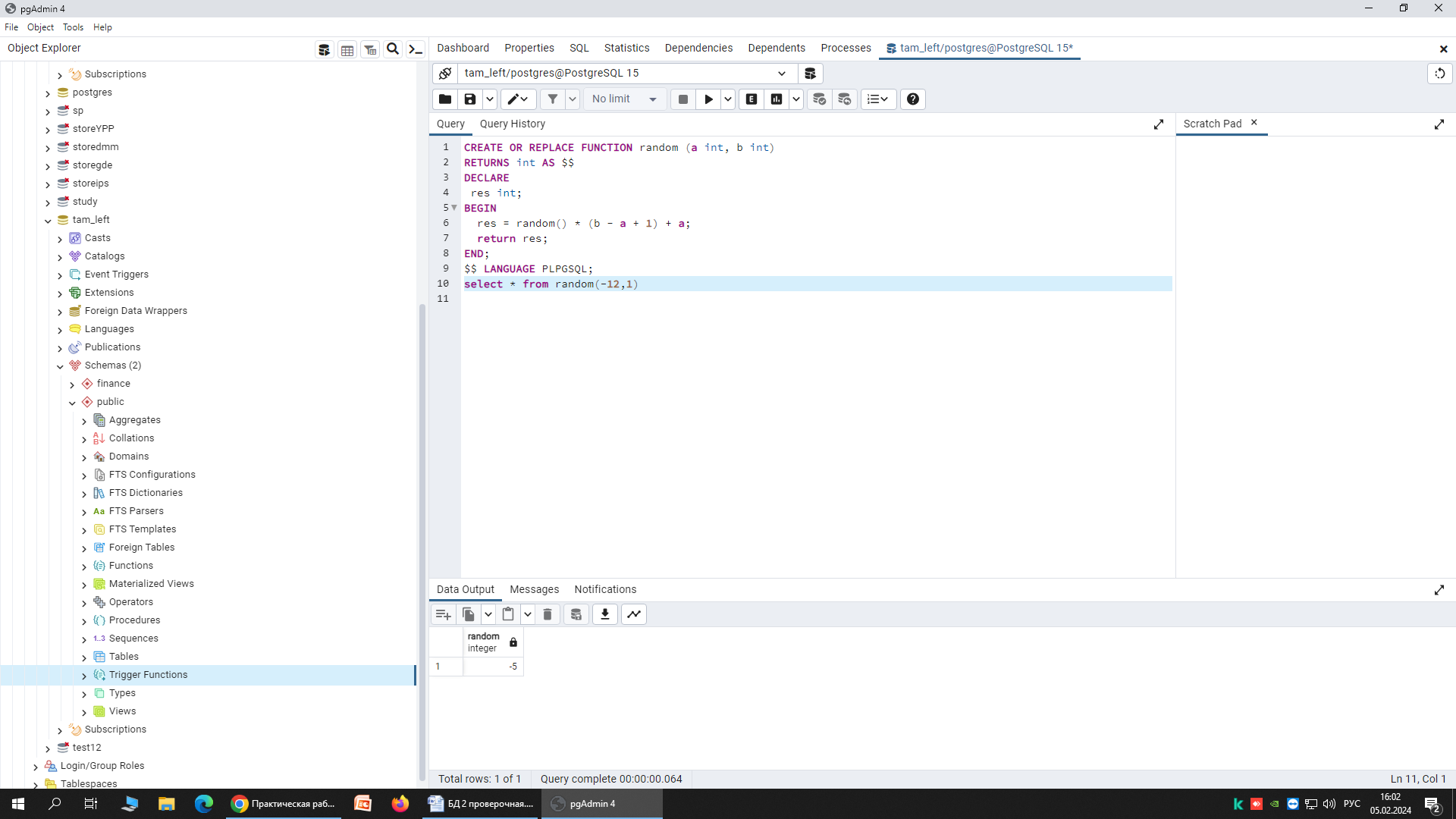
res = random() \* (b - a + 1) + a;

return res;

END;

$$ LANGUAGE PLPGSQL;

select \* from random(-12,1)



1.2 CREATE OR REPLACE FUNCTION random\_double (a int, b int, p int)

RETURNS decimal AS $$

DECLARE

num decimal;

res decimal;

BEGIN

num = random() \* (b - a)+a;

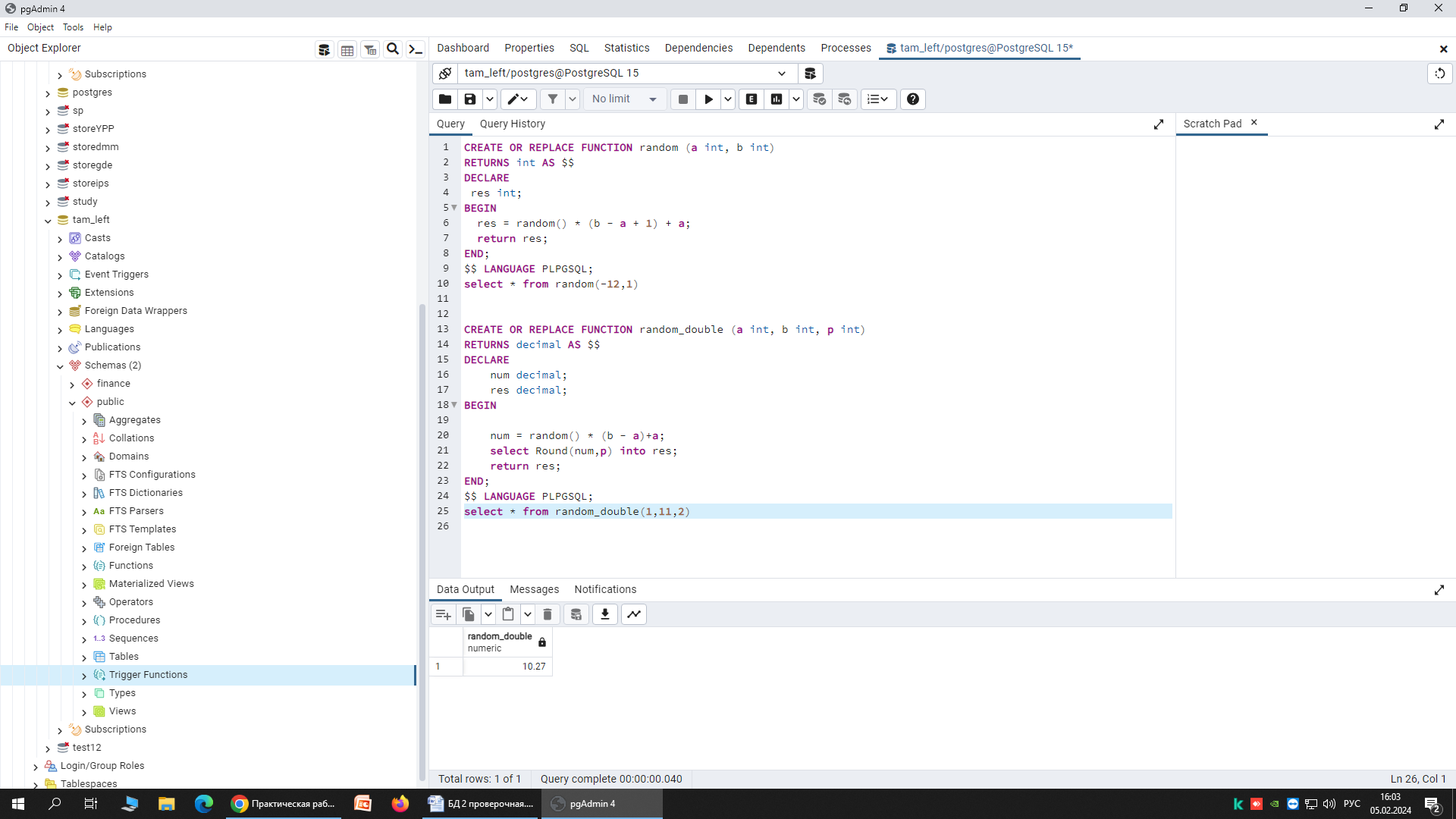
select Round(num,p) into res;

return res;

END;

$$ LANGUAGE PLPGSQL;

select \* from random\_double(1,11,2)



**2. Функции, работающие с массивами**

2.1 CREATE OR REPLACE FUNCTION mult\_arr(arr numeric[], k int)

RETURNS decimal[] AS $$

DECLARE

x int;

BEGIN

FOREACH x IN ARRAY arr

LOOP

arr[x]:=arr[x]\*k;

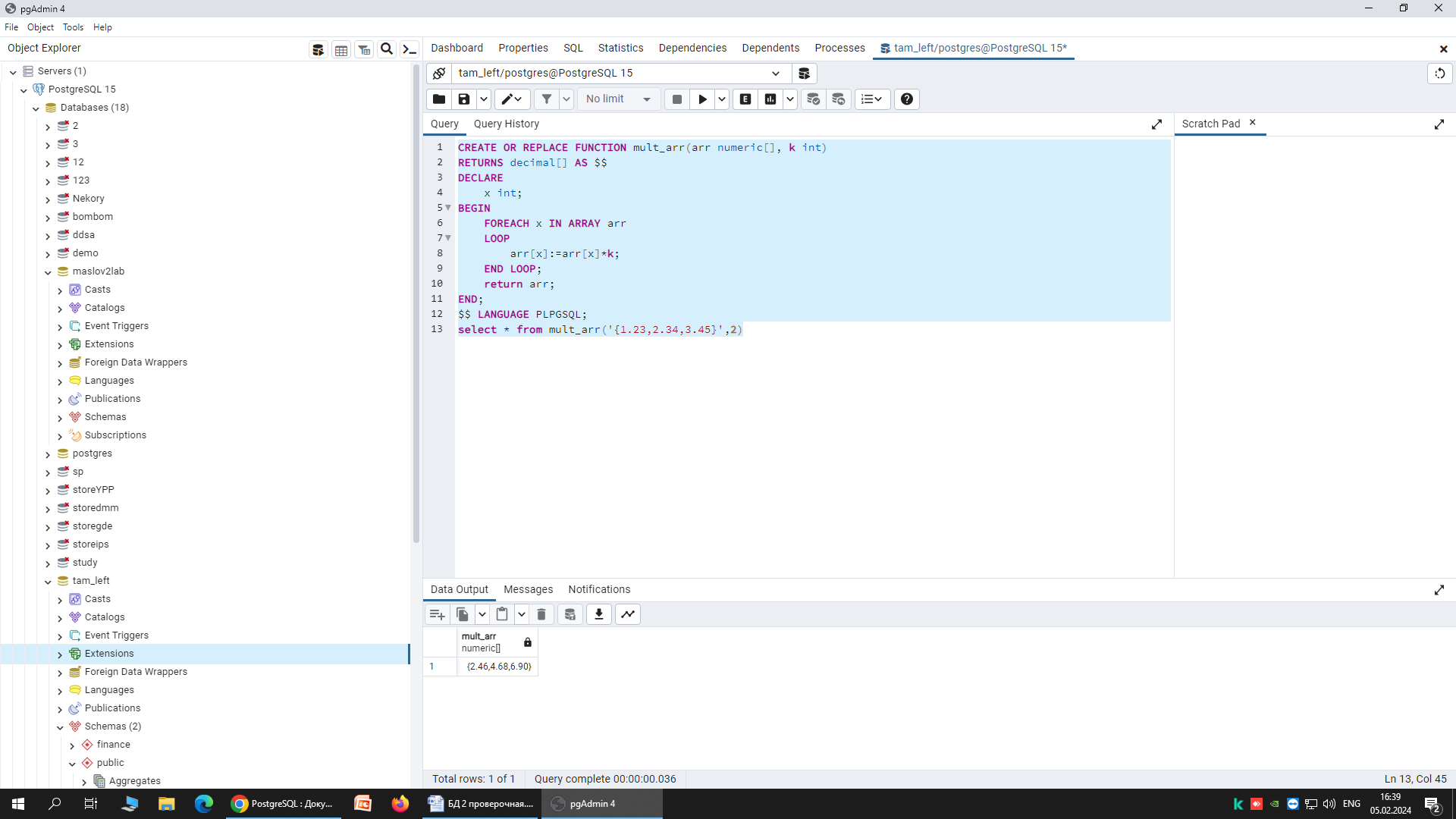
END LOOP;

return arr;

END;

$$ LANGUAGE PLPGSQL;

select \* from mult\_arr('{1.23,2.34,3.45}',2)



2.2 create or replace function get\_arr(s text, n numeric, i integer)

returns text[] as $$

declare

arr text[];

begin

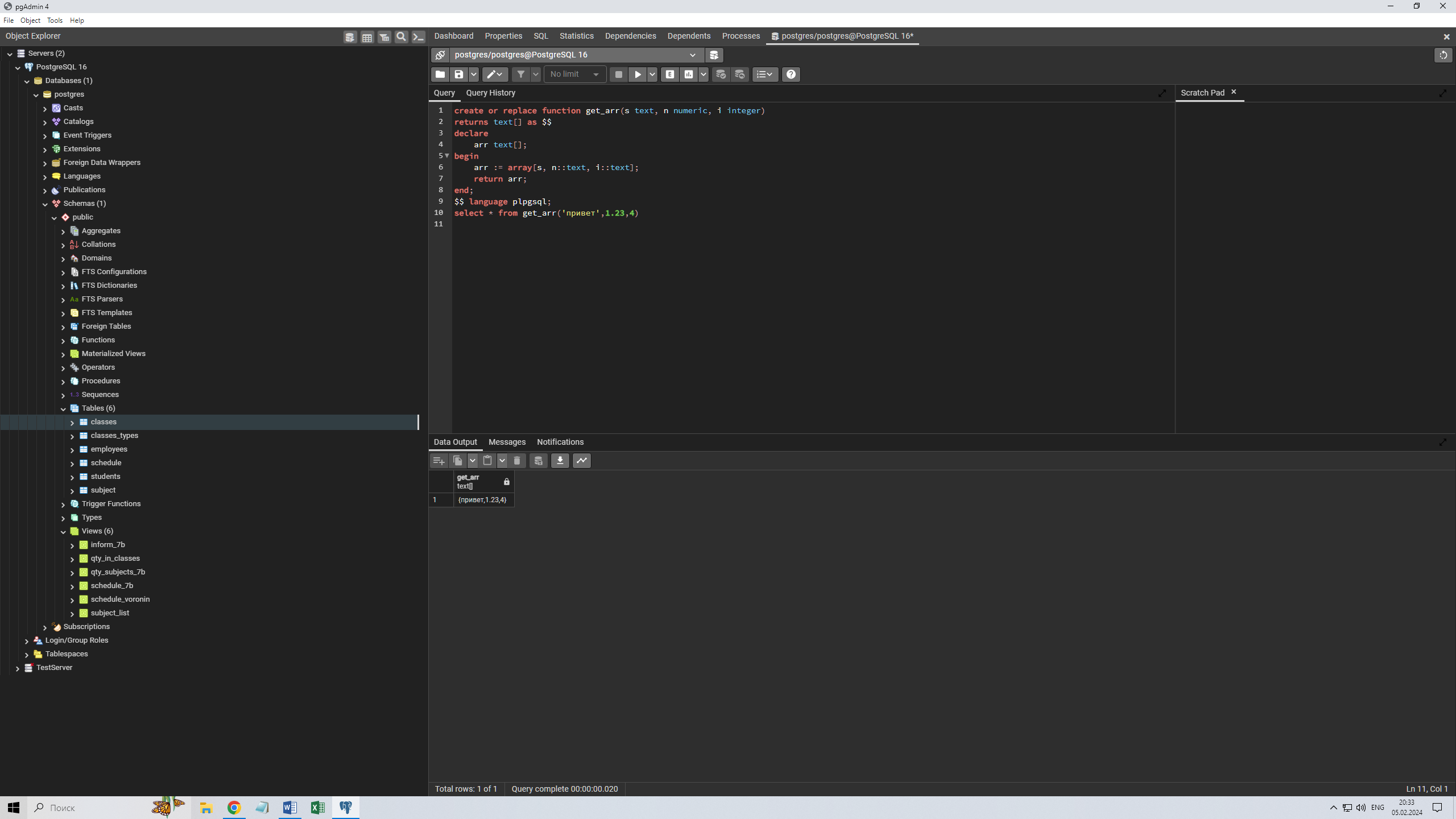
arr := array[s, n::text, i::text];

return arr;

end;

$$ language plpgsql;

select \* from get\_arr('привет',1.23,4)



**3. Функции с произвольным числом аргументов**

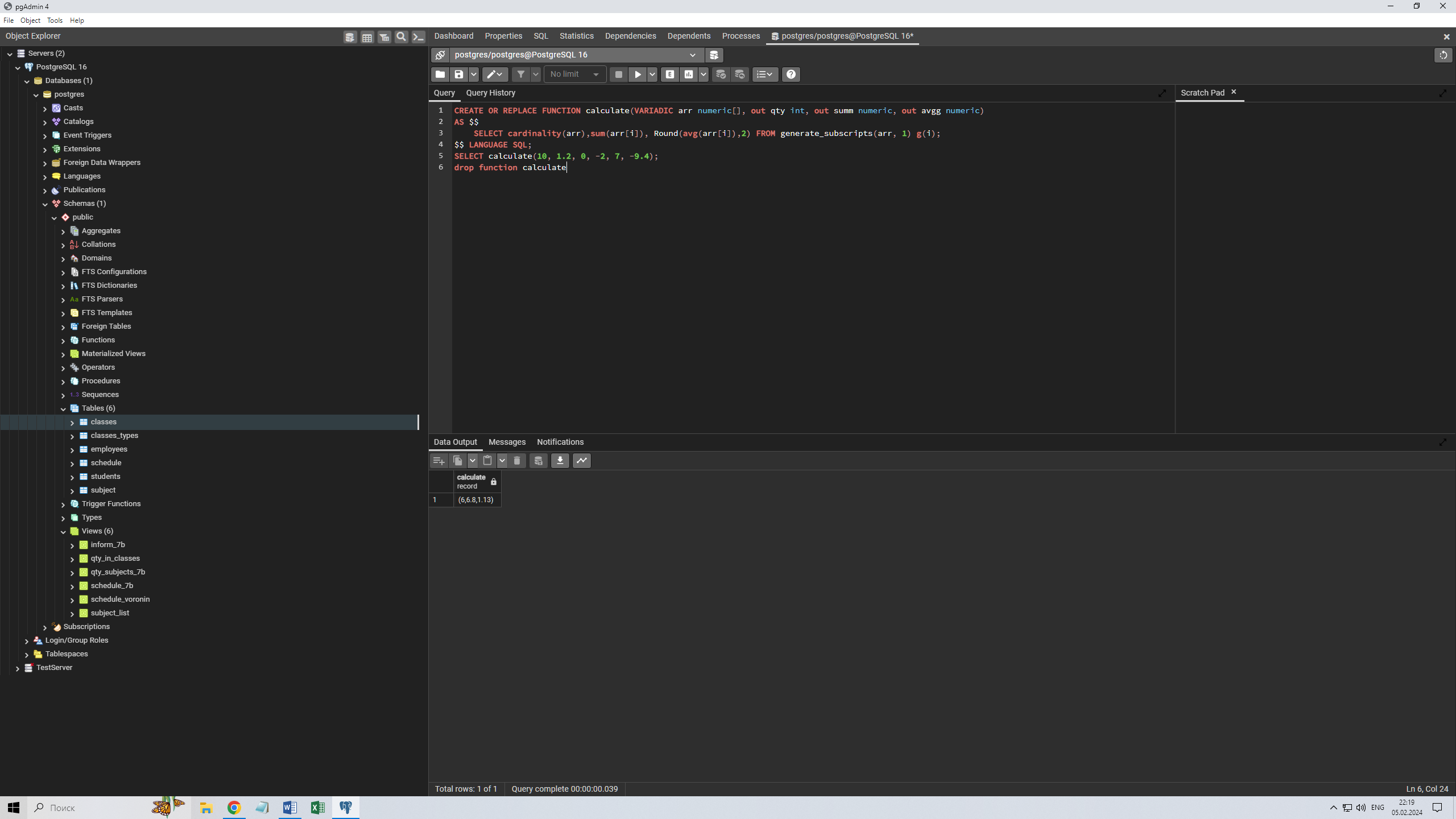
3.1 CREATE OR REPLACE FUNCTION calculate(VARIADIC arr numeric[], out qty int, out summ numeric, out avgg numeric)

AS $$

SELECT cardinality(arr),sum(arr[i]), Round(avg(arr[i]),2) FROM generate\_subscripts(arr, 1) g(i);

$$ LANGUAGE SQL;

SELECT calculate(10, 1.2, 0, -2, 7, -9.4);



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