Задание 1  
  
create table students

(

id serial primary key ,

name text ,

total\_score integer

);

insert into students(id, name, total\_score) values (1, 'Мария Петрова', 96);

insert into students(id, name, total\_score) values (2, 'Александр Иванов', 79);

insert into students(id, name, total\_score) values (3, 'Марат Мухаметшин', 88);

insert into students(id, name, total\_score) values (4, 'Ильфат Якупов', 91);

insert into students(id, name, total\_score) values (5, 'Алина Мустафина', 93);

insert into students(id, name, total\_score) values (6, 'Айгуль Ибрагимова', 99);

create table activity\_scores

(

student\_id integer references students(id) ,

activity\_type text ,

score integer

);

create or replace function calculate\_scholarship() returns trigger

language plpgsql as $$

declare

total\_score integer := 0;

begin

select sum(score) into total\_score

from activity\_scores

where student\_id = new.student\_id;

if total\_score < 80 then

raise notice 'Стипендия = 0';

elsif total\_score >= 80 and total\_score < 90 then

raise notice 'Стипендия = 500';

else

raise notice 'Стипендия = 1000';

end if;

return new;

end;

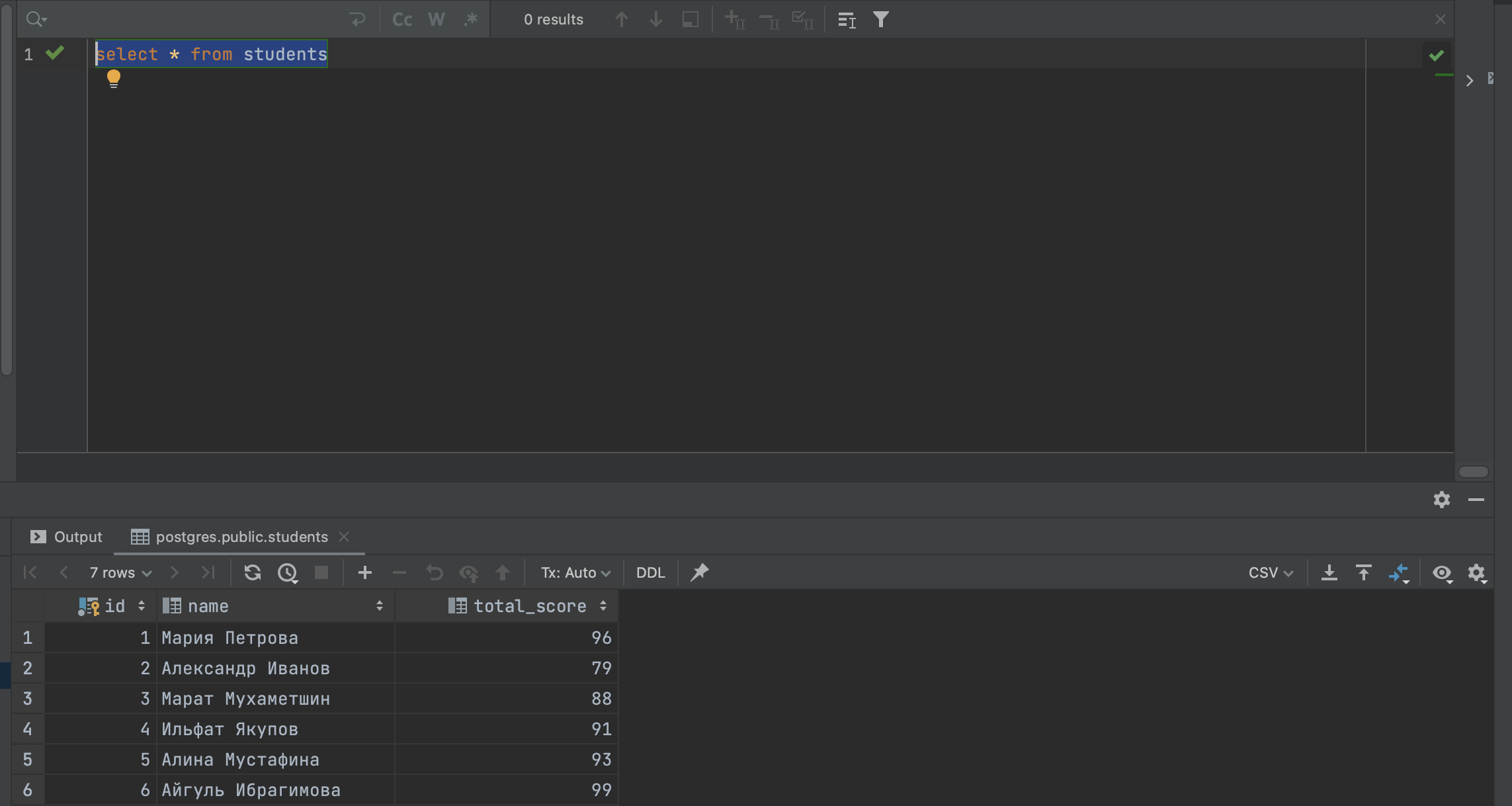
$$

create trigger update\_scholarship\_trigger

after insert on activity\_scores

for each row execute function calculate\_scholarship();

select \* from students



insert into activity\_scores(student\_id, activity\_type, score) values (1, 'Тест', 48)

insert into activity\_scores(student\_id, activity\_type, score) values (1, 'Экзамен', 50)

insert into activity\_scores(student\_id, activity\_type, score) values (2, 'Тест', 41)

insert into activity\_scores(student\_id, activity\_type, score) values (2, 'Экзамен', 40)

insert into activity\_scores(student\_id, activity\_type, score) values (3, 'Тест', 44)

insert into activity\_scores(student\_id, activity\_type, score) values (3, 'Экзамен', 40)

insert into activity\_scores(student\_id, activity\_type, score) values (4, 'Тест', 34)

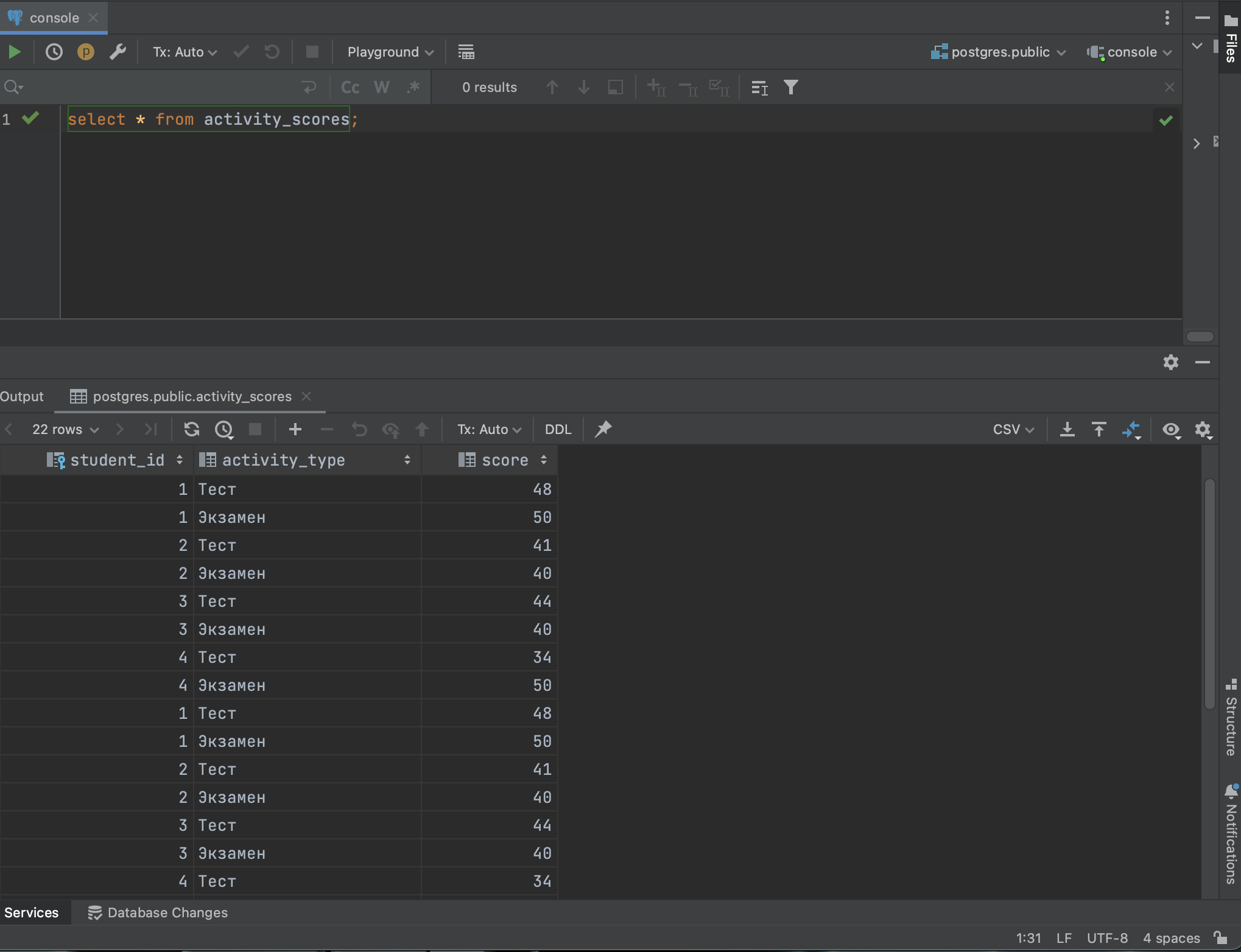
insert into activity\_scores(student\_id, activity\_type, score) values (4, 'Экзамен', 50)

insert into activity\_scores(student\_id, activity\_type, score) values (5, 'Тест', 44)

insert into activity\_scores(student\_id, activity\_type, score) values (5, 'Экзамен', 49)

insert into activity\_scores(student\_id, activity\_type, score) values (6, 'Тест', 49)

insert into activity\_scores(student\_id, activity\_type, score) values (6, 'Экзамен', 49)



postgres.public> insert into activity\_scores(student\_id, activity\_type, score) values (1, 'Тест', 48)

Стипендия = 1000

[2023-09-02 22:18:07] 1 row affected in 3 ms

postgres.public> insert into activity\_scores(student\_id, activity\_type, score) values (1, 'Экзамен', 50)

Стипендия = 1000

[2023-09-02 22:18:07] 1 row affected in 2 ms

postgres.public> insert into activity\_scores(student\_id, activity\_type, score) values (2, 'Тест', 41)

Стипендия = 1000

[2023-09-02 22:18:07] 1 row affected in 2 ms

postgres.public> insert into activity\_scores(student\_id, activity\_type, score) values (2, 'Экзамен', 40)

Стипендия = 1000

[2023-09-02 22:18:07] 1 row affected in 2 ms

postgres.public> insert into activity\_scores(student\_id, activity\_type, score) values (3, 'Тест', 44)

Стипендия = 1000

[2023-09-02 22:18:07] 1 row affected in 2 ms

postgres.public> insert into activity\_scores(student\_id, activity\_type, score) values (3, 'Экзамен', 40)

Стипендия = 1000

[2023-09-02 22:18:07] 1 row affected in 2 ms

postgres.public> insert into activity\_scores(student\_id, activity\_type, score) values (4, 'Тест', 34)

Стипендия = 1000

[2023-09-02 22:18:07] 1 row affected in 4 ms

postgres.public> insert into activity\_scores(student\_id, activity\_type, score) values (4, 'Экзамен', 50)

Стипендия = 1000

[2023-09-02 22:18:07] 1 row affected in 2 ms

postgres.public> insert into activity\_scores(student\_id, activity\_type, score) values (5, 'Тест', 44)

Стипендия = 0

[2023-09-02 22:18:07] 1 row affected in 2 ms

postgres.public> insert into activity\_scores(student\_id, activity\_type, score) values (5, 'Экзамен', 49)

Стипендия = 1000

[2023-09-02 22:18:07] 1 row affected in 3 ms

postgres.public> insert into activity\_scores(student\_id, activity\_type, score) values (6, 'Тест', 49)

Стипендия = 0

[2023-09-02 22:18:07] 1 row affected in 2 ms

postgres.public> insert into activity\_scores(student\_id, activity\_type, score) values (6, 'Экзамен', 49)

Стипендия = 1000

Задание 2

create or replace function update\_total\_score() returns trigger

language plpgsql as $$

declare

sum\_score integer := 0;

score\_ integer;

begin

for score\_ in

select score

from activity\_scores

where student\_id = new.student\_id

loop

sum\_score := sum\_score + score\_;

end loop;

update students

set total\_score = sum\_score

where id = new.student\_id;

return new;

end;

$$

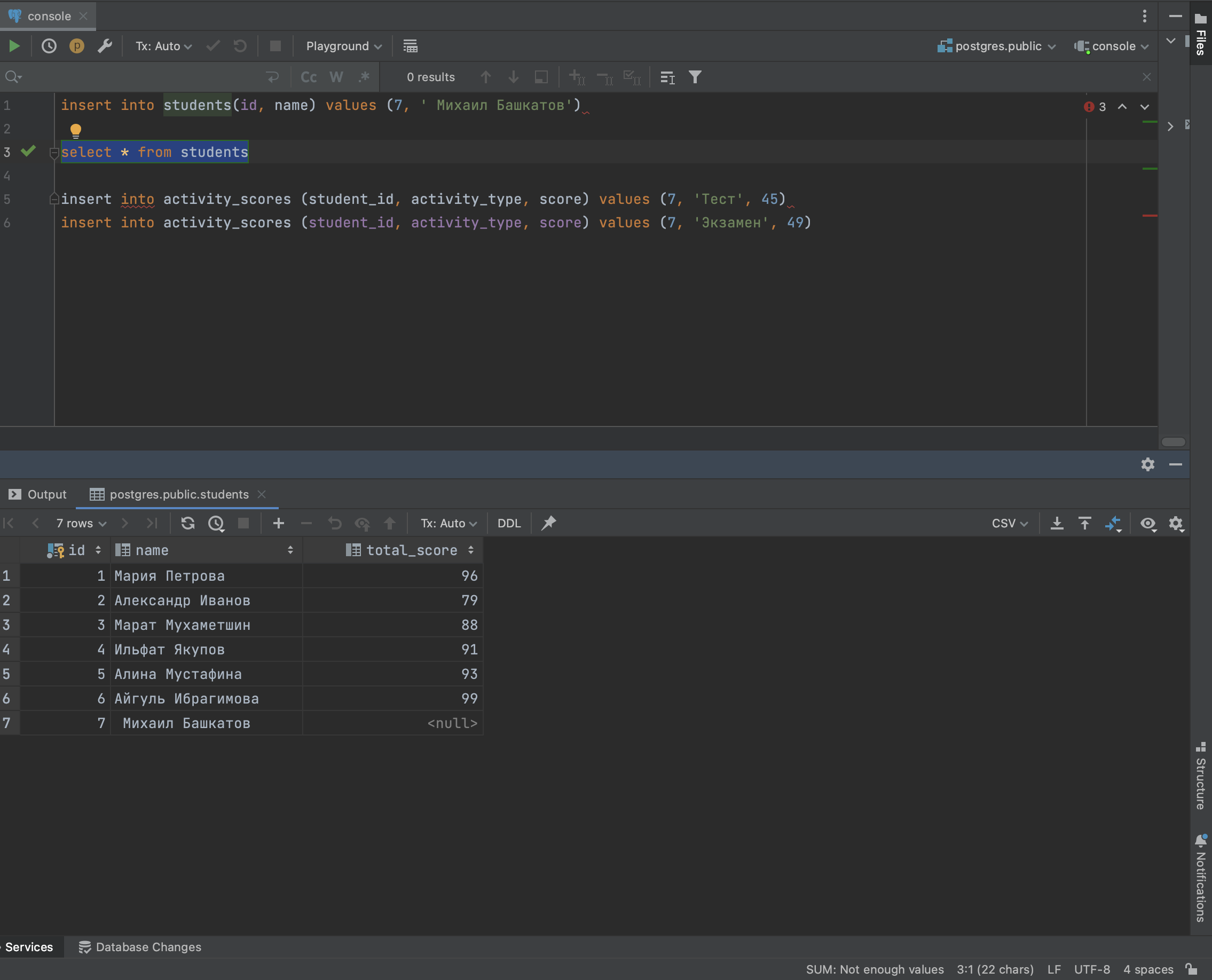
create trigger update\_total\_score\_trigger

after insert on activity\_scores

for each row execute function update\_total\_score();

insert into students(id, name) values (7, ' Михаил Башкатов')

select \* from students



insert into activity\_scores(student\_id, activity\_type, score) values (7, 'Тест', 45)

insert into activity\_scores(student\_id, activity\_type, score) values (7, 'Экзамен', 49)

