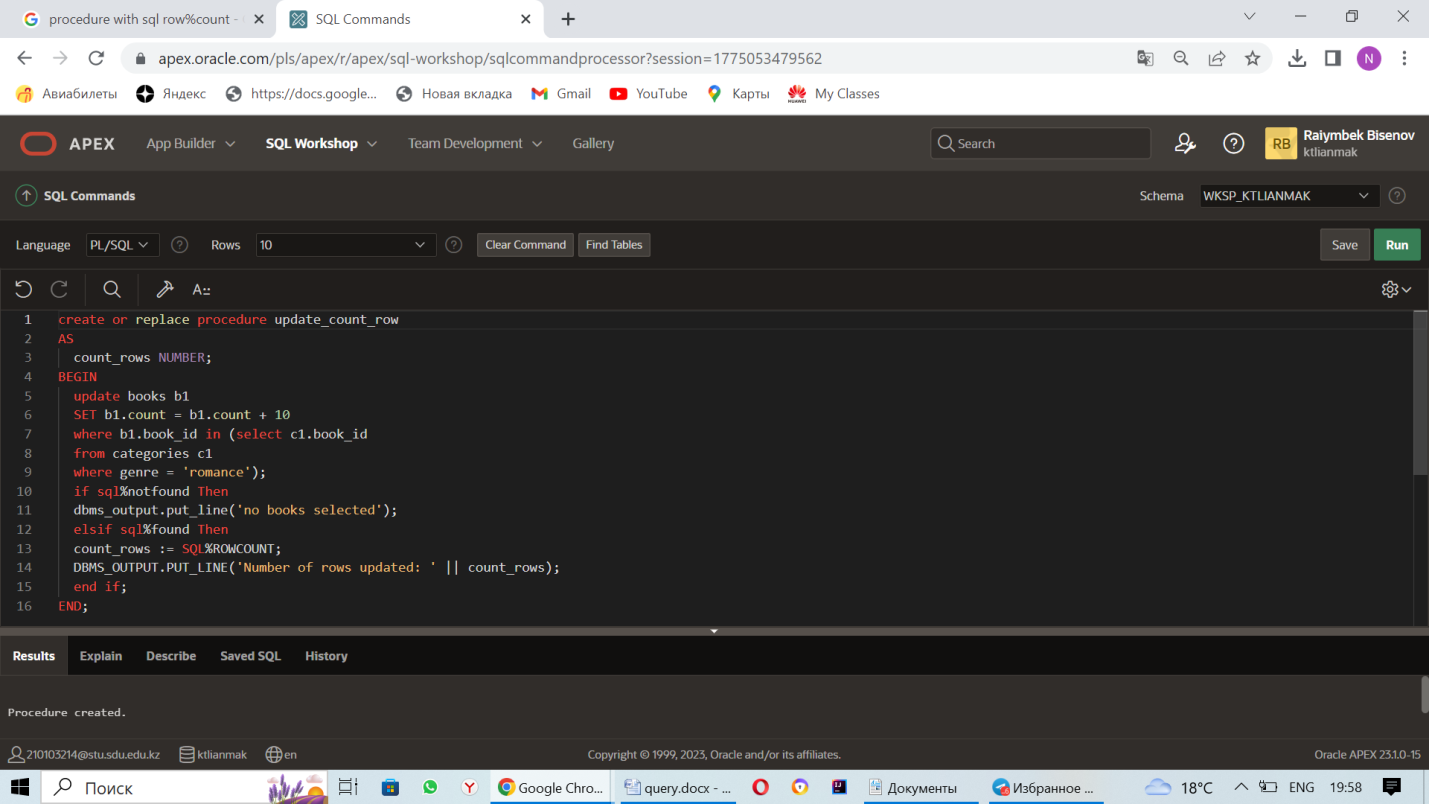
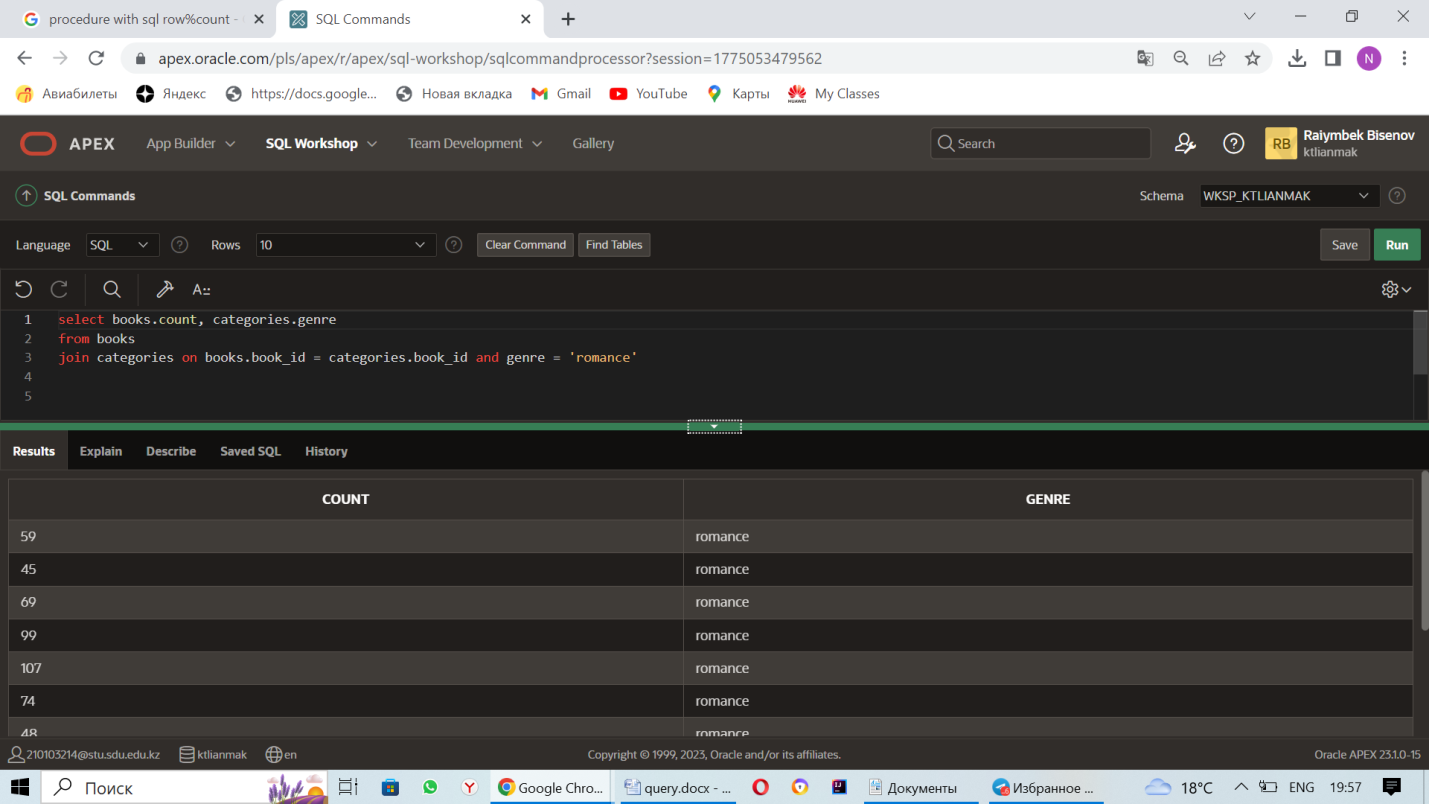
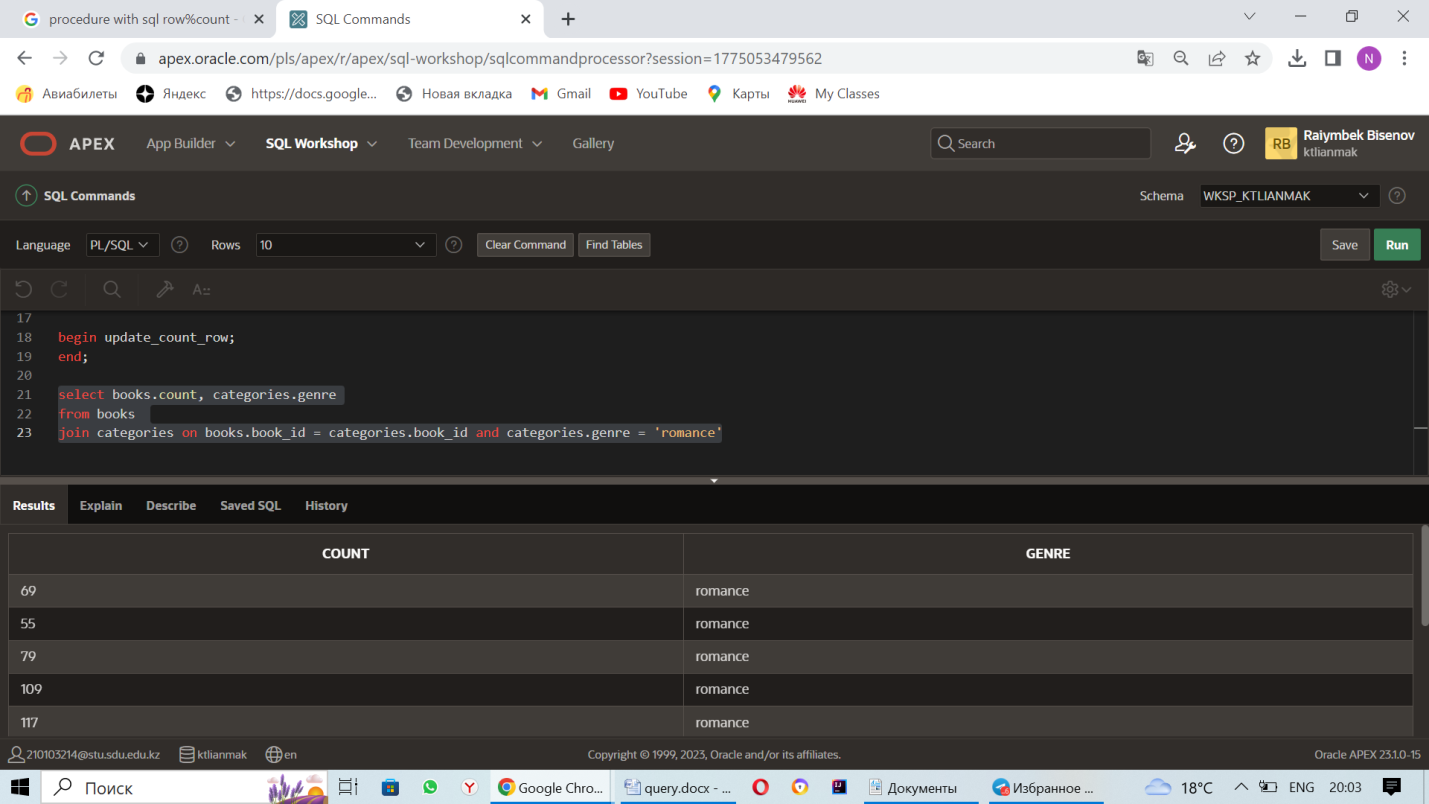
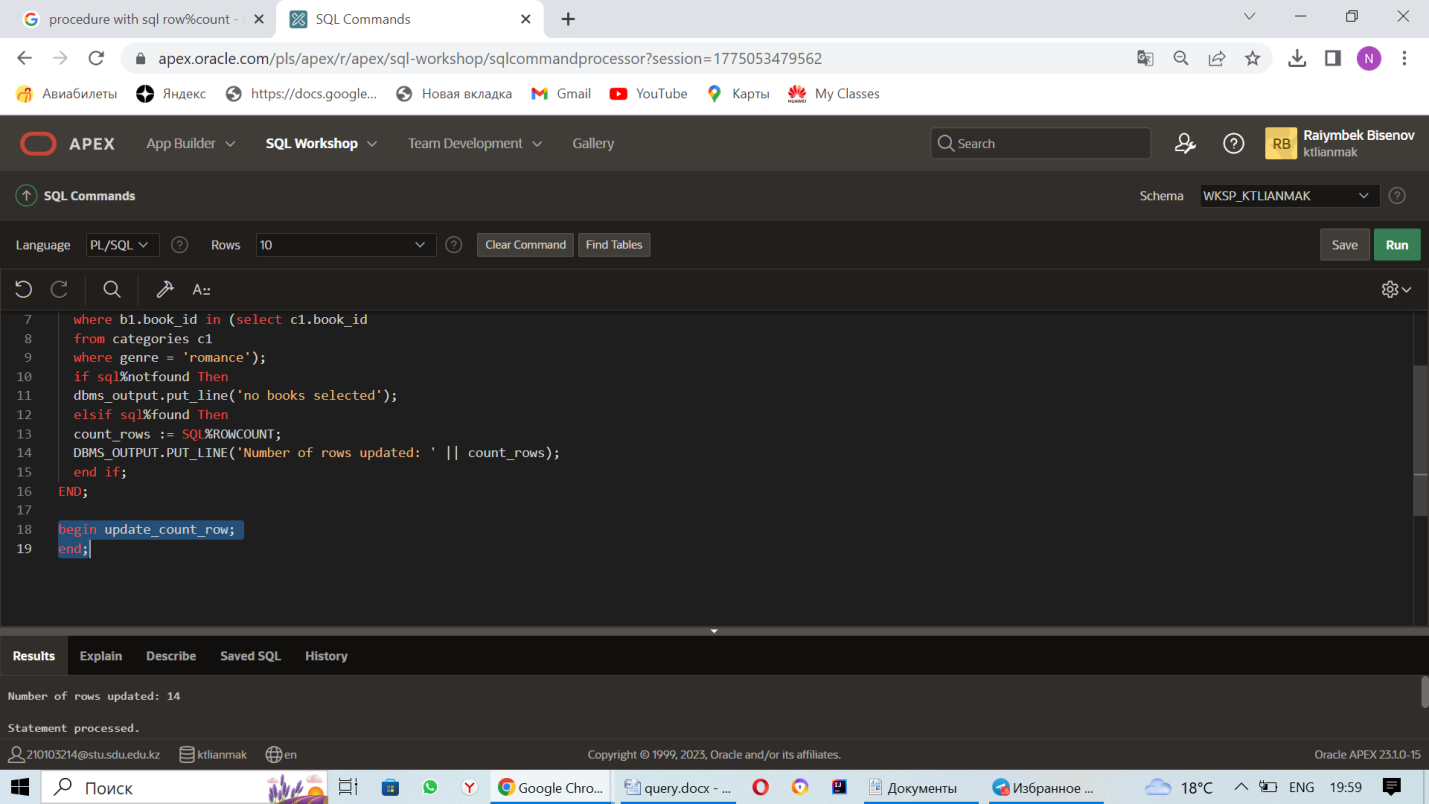
1.In honor of Valentine 's Day, the bookstore received 10 new books in the genre of "romance”





1. Procedure which uses SQL%ROWCOUNT to determine the number of rows affected

create or replace procedure update\_count\_row

AS

count\_rows NUMBER;

BEGIN

update books b1

SET b1.count = b1.count + 10

where b1.book\_id in (select c1.book\_id

from categories c1

where genre = 'romance');

if sql%notfound Then

dbms\_output.put\_line('no books selected');

elsif sql%found Then

count\_rows := SQL%ROWCOUNT;

DBMS\_OUTPUT.PUT\_LINE('Number of rows updated: ' || count\_rows);

end if;

END;

begin update\_count\_row;

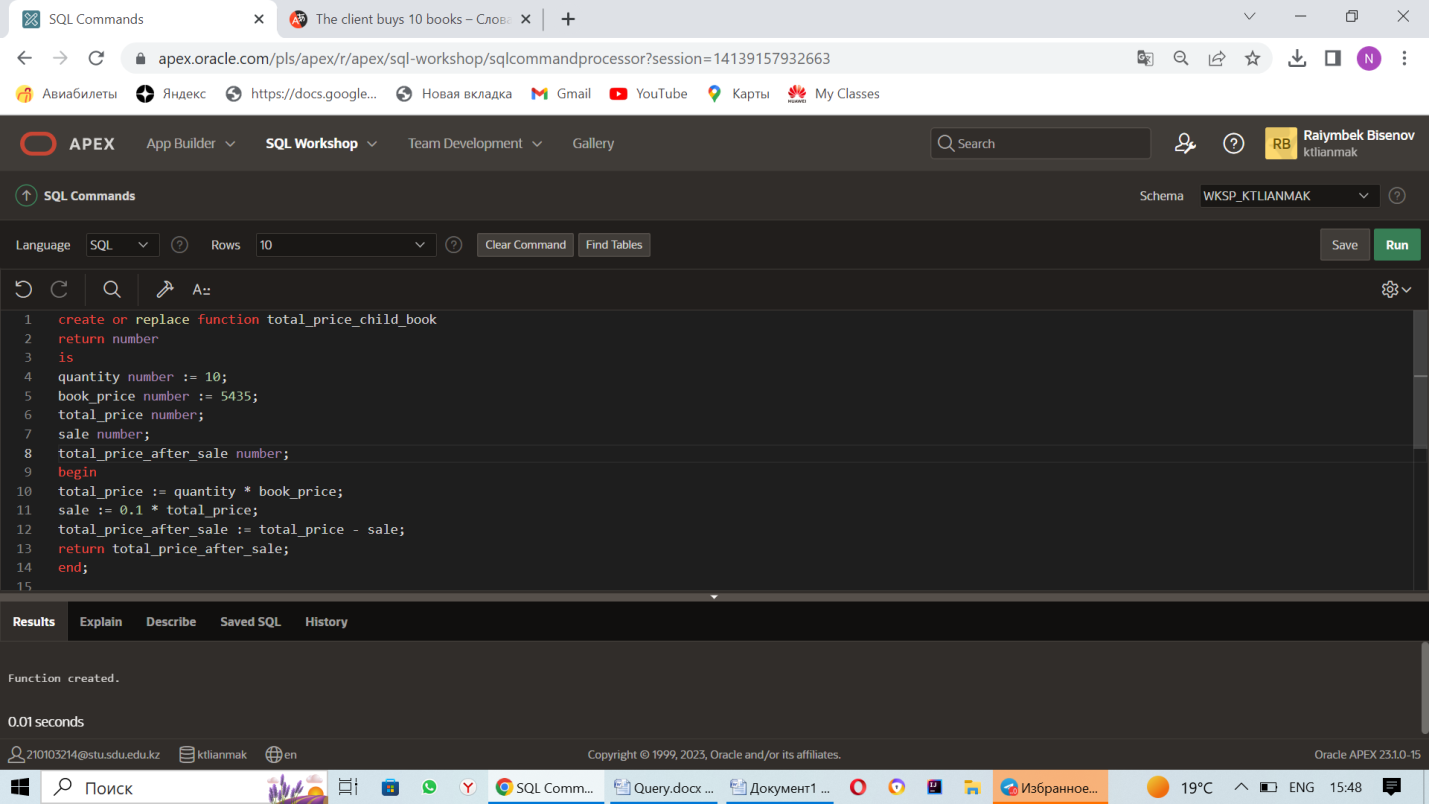
end;

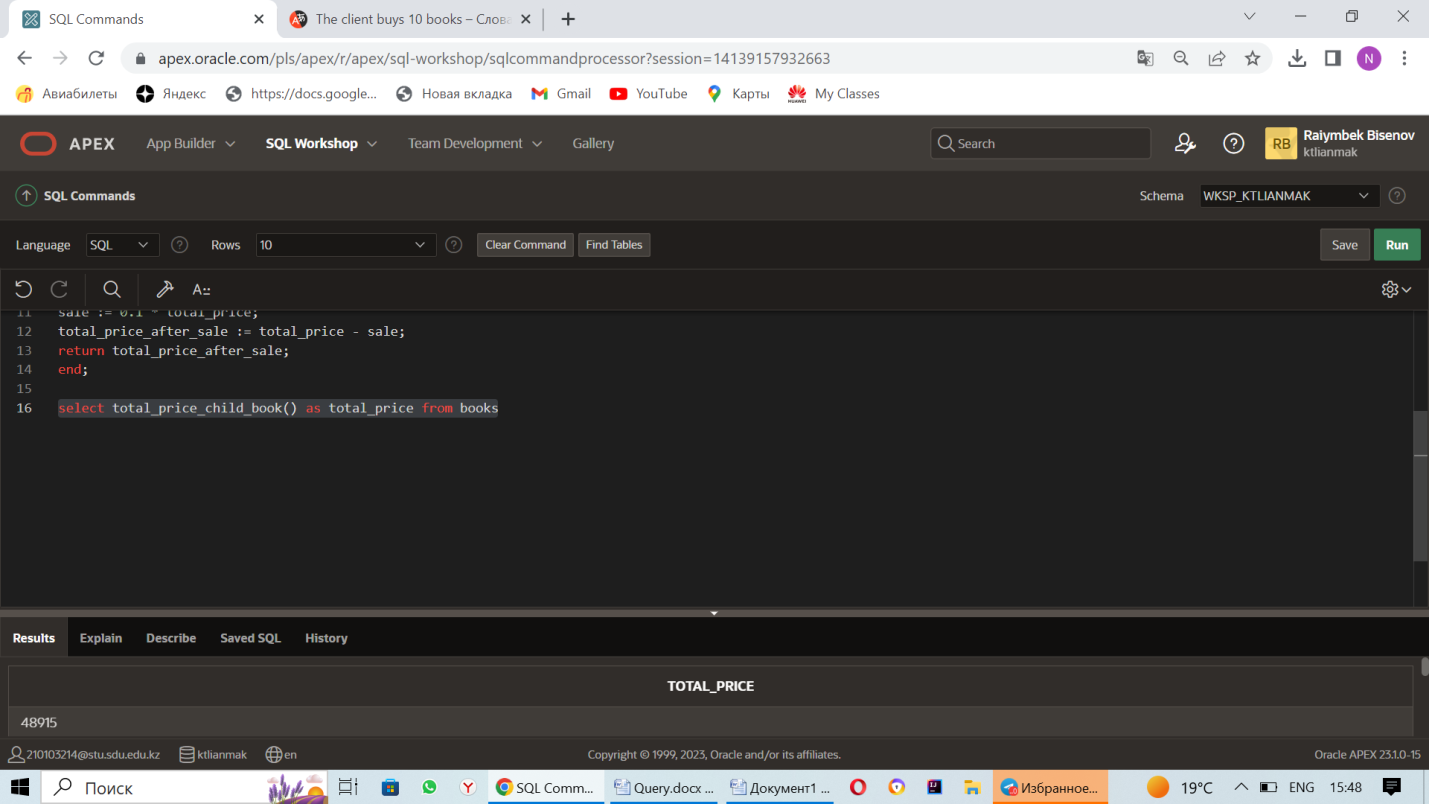
select books.count, categories.genre

from books

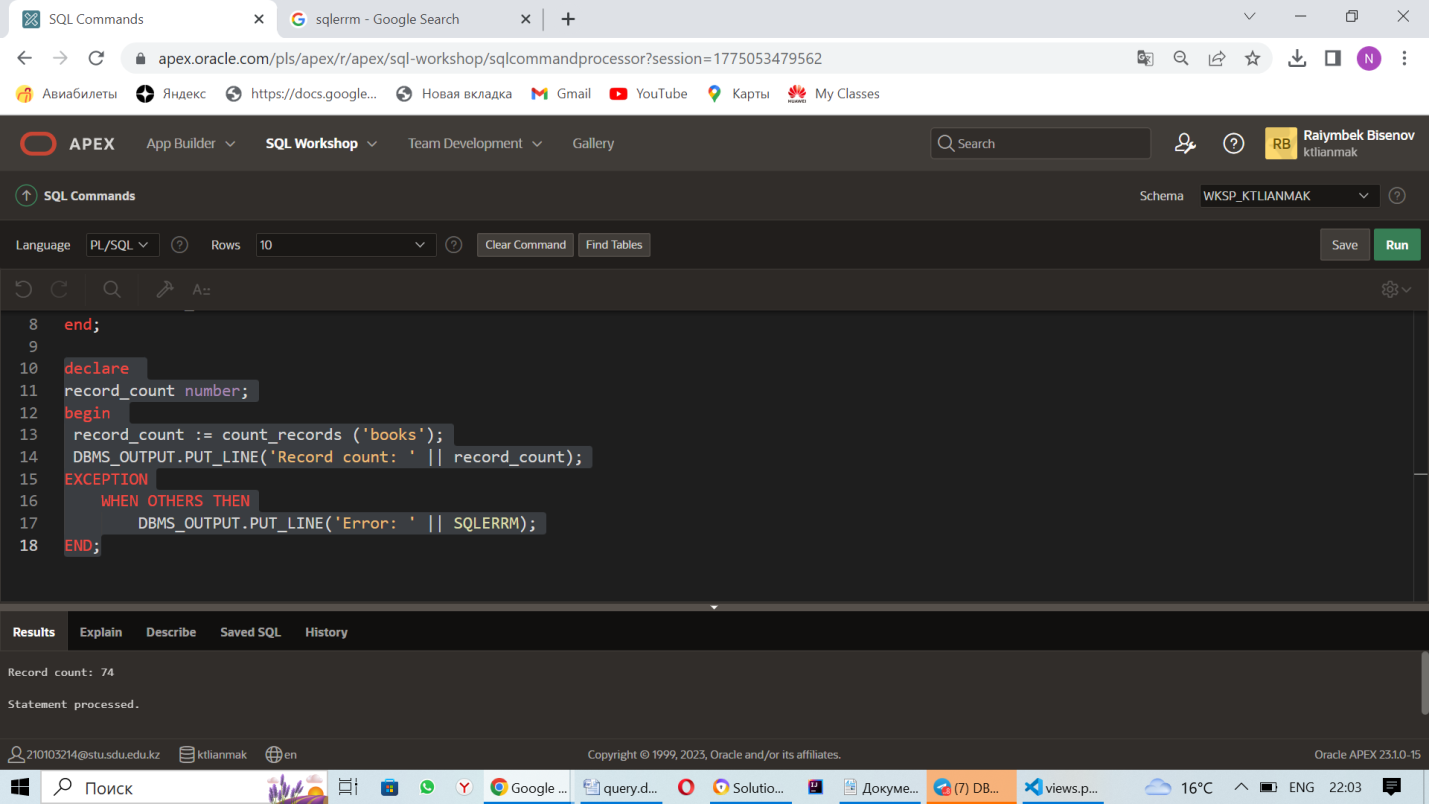
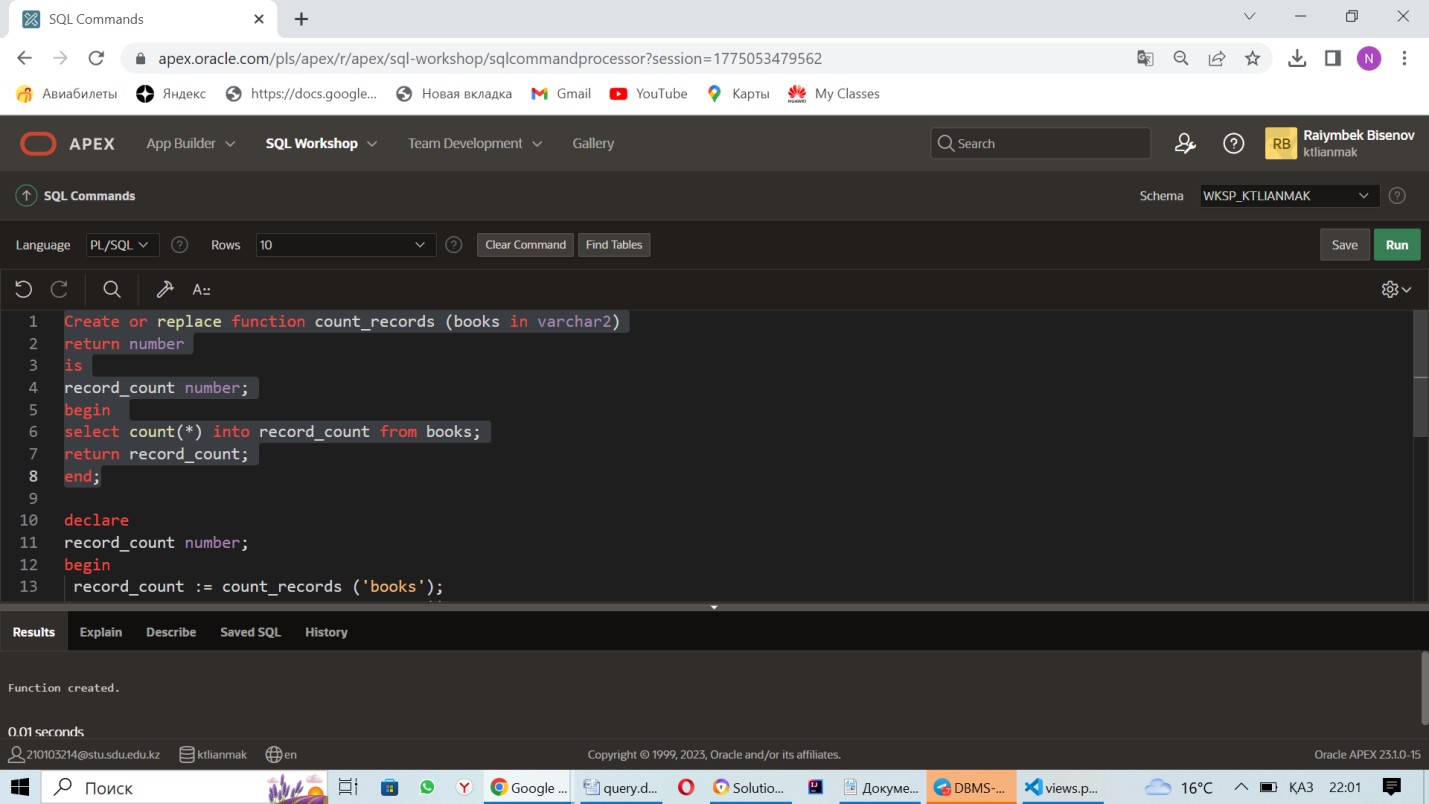
join categories on books.book\_id = categories.book\_id and categories.genre = 'romance'

2. The client buys 10 books from the 'reading child' sections. And this client will get a 10% discount for the purchase of 10 books. You need to calculate the total amount.





2. The function shows the number of records in the book table



2. Function which counts the number of records

Create or replace function count\_records (books in varchar2)

return number

is

record\_count number;

begin

select count(\*) into record\_count from books;

return record\_count;

end;

declare

record\_count number;

begin

record\_count := count\_records ('books');

DBMS\_OUTPUT.PUT\_LINE('Record count: ' || record\_count);

EXCEPTION

WHEN OTHERS THEN

DBMS\_OUTPUT.PUT\_LINE('Error: ' || SQLERRM);

END;