**LAB CYCLE 2**

1.Write a PL/SQL code to accept the text and reverse the given text. Check the text is palindrome or not.

PL/SQL

DECLARE

s VARCHAR2(10) := 'cat';

l VARCHAR2(20);

t VARCHAR2(10);

BEGIN

FOR i IN REVERSE 1..Length(s) LOOP

l := Substr(s, i, 1);

t := t ||''||l;

END LOOP;

IF t = s THEN

dbms\_output.Put\_line(t ||''||' is palindrome');

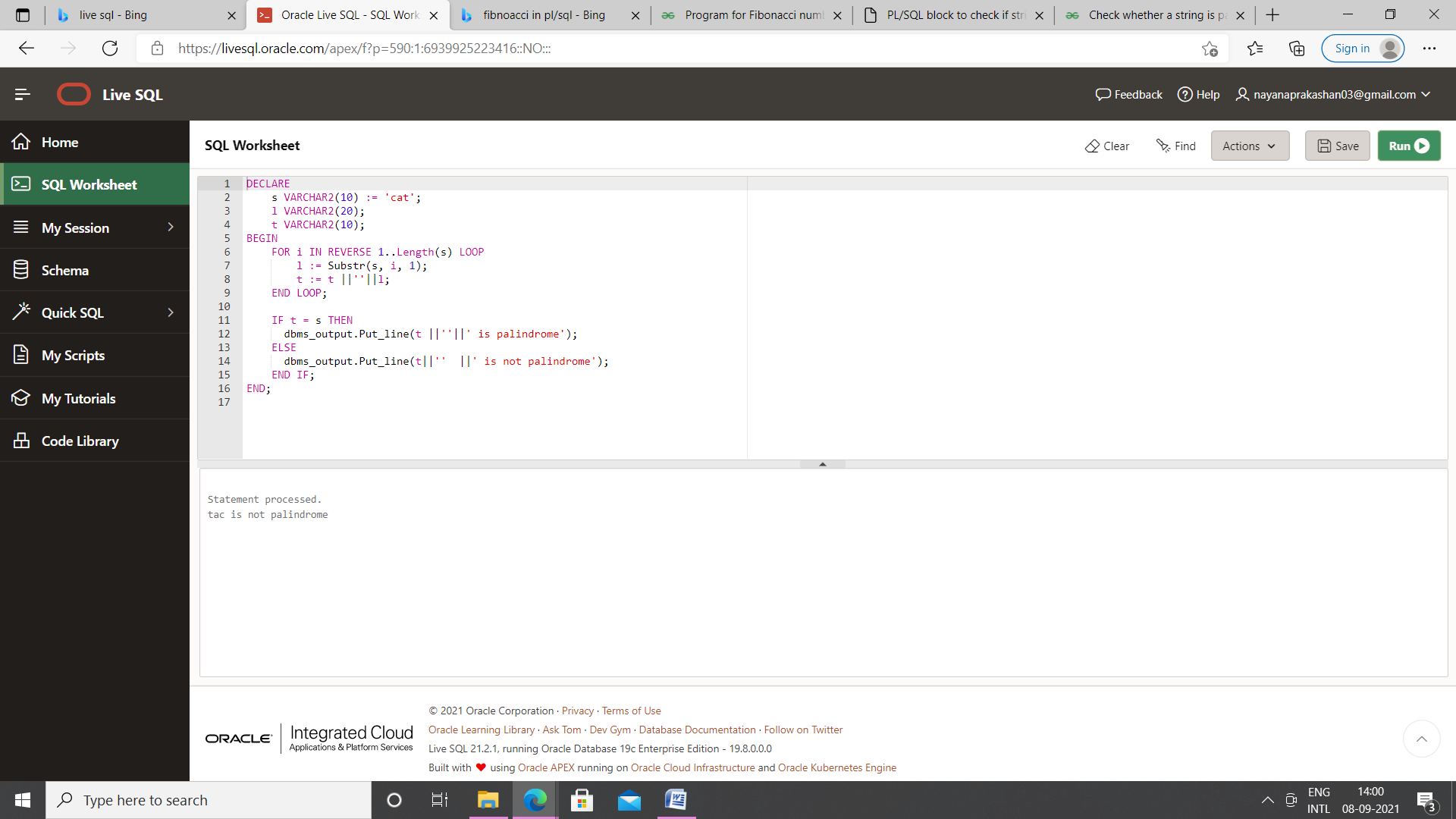
ELSE

dbms\_output.Put\_line(t||'' ||' is not palindrome');

END IF;

END;

OUTPUT



2. Write a program to read two numbers; If the first no > 2nd no, then swap the numbers; if the first number is an odd number, then find its cube; if first no < 2nd no then raise it to its power; if both the numbers are equal, then find its sqrt.

PL/SQL

DECLARE

a INTEGER:=12;

b INTEGER:=5;

temp INTEGER:=0;

c INTEGER;

cube INTEGER;

BEGIN

IF a > b THEN

temp:=a;

a:=b;

b:=temp;

DBMS\_OUTPUT.PUT\_LINE('After swapping the a value is '||a ||' and b

value is '||b);

IF MOD(b,2) !=0 THEN

cube:=a \* a \* a;

DBMS\_OUTPUT.PUT\_LINE('Cube is :'||cube);

ELSE

DBMS\_OUTPUT.PUT\_LINE('first number is even');

END IF;

ELSIF a < b THEN

c:=a \*\*b;

DBMS\_OUTPUT.PUT\_LINE('Power is :'||c); ELSIF a=b THEN

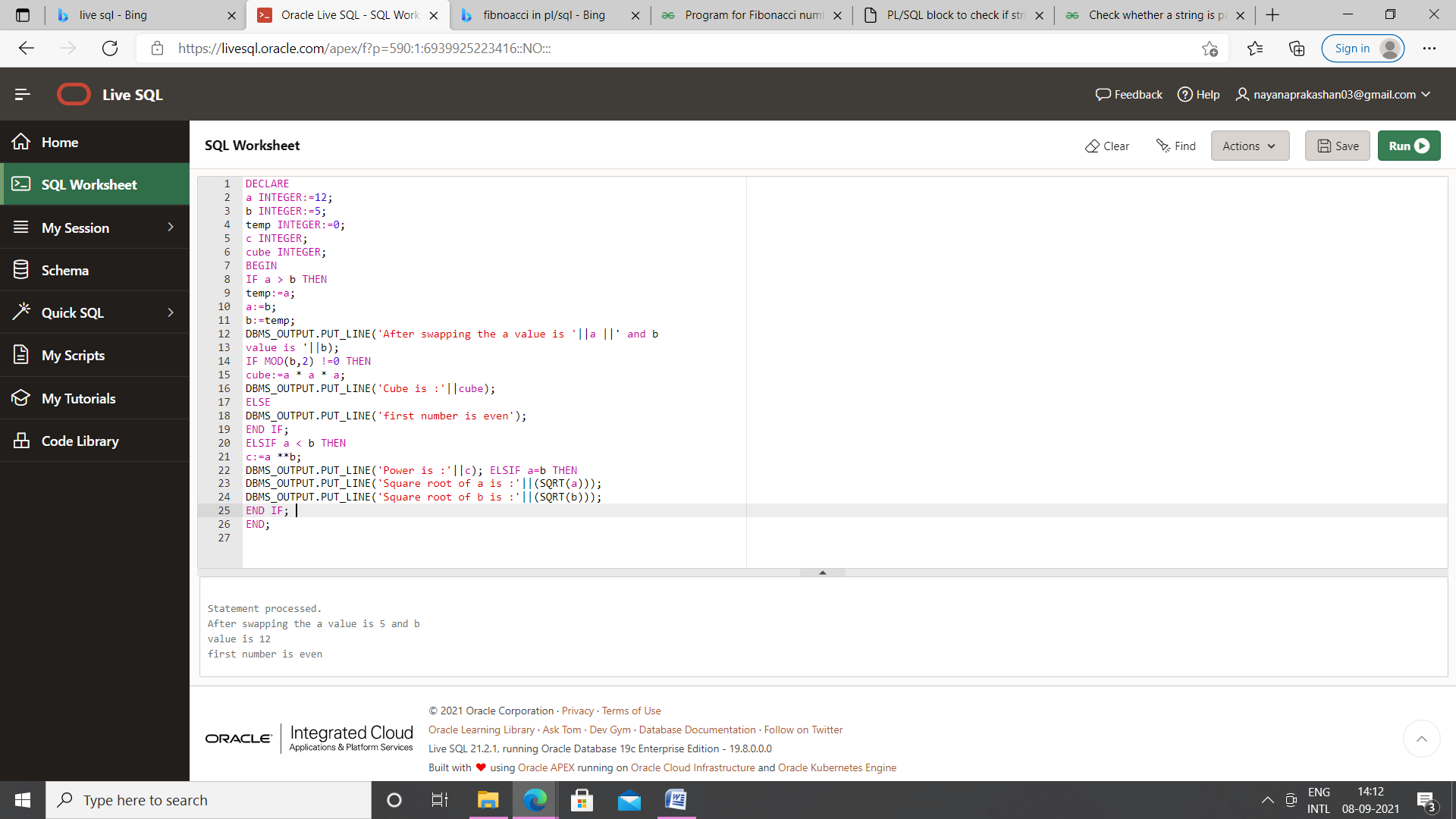
DBMS\_OUTPUT.PUT\_LINE('Square root of a is :'||(SQRT(a)));

DBMS\_OUTPUT.PUT\_LINE('Square root of b is :'||(SQRT(b)));

END IF;

END;

OUTPUT



3. Write a program to generate first 10 terms of the Fibonacci series

**PL/SQL**

declare

first number := 0;

second number := 1;

temp number;

n number := 5;

i number;

begin

    dbms\_output.put\_line('Series:');

    dbms\_output.put\_line(first);

    dbms\_output.put\_line(second);

    for i in 2..n

    loop

        temp:=first+second;

first := second;

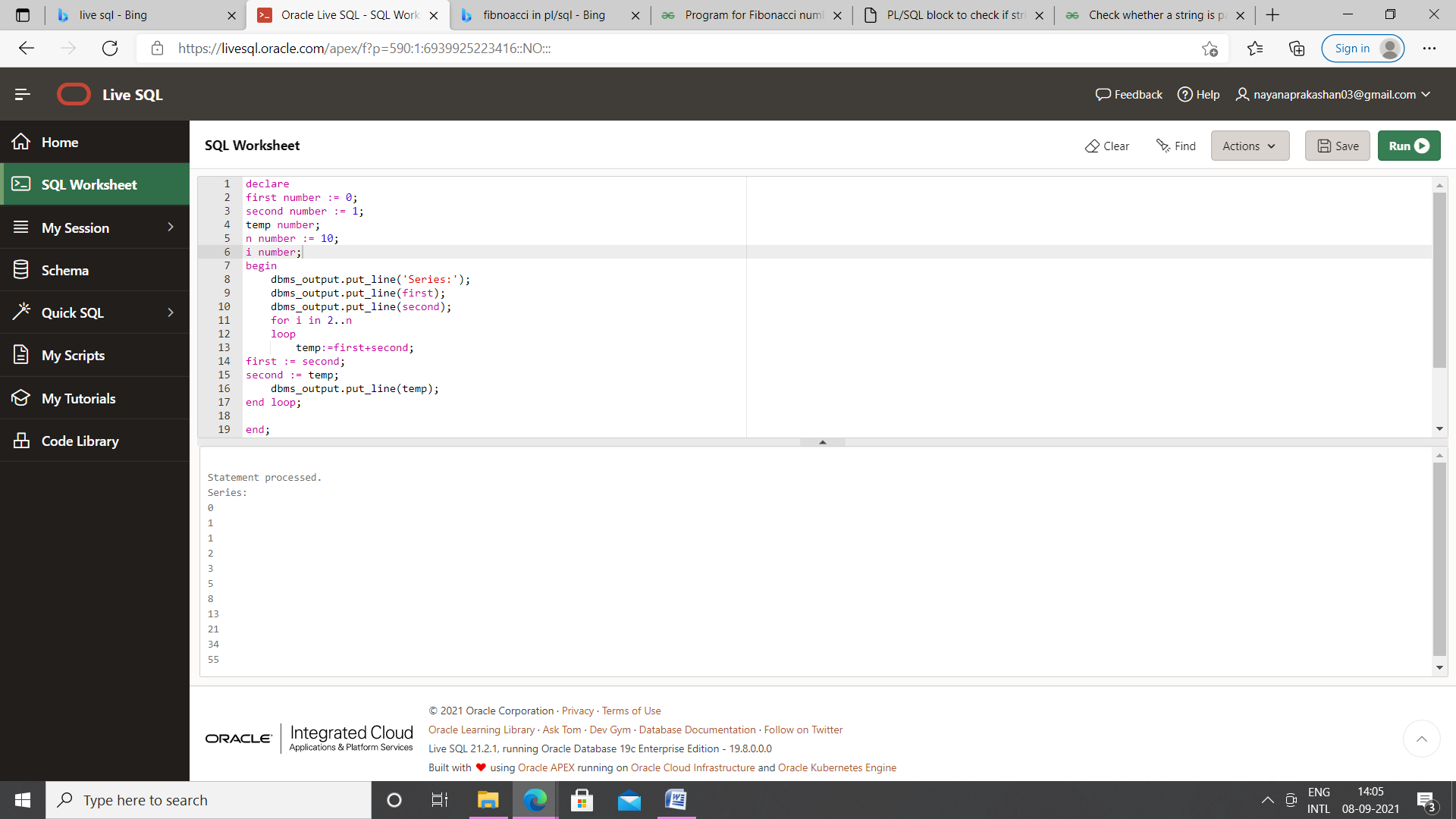
second := temp;

    dbms\_output.put\_line(temp);

end loop;

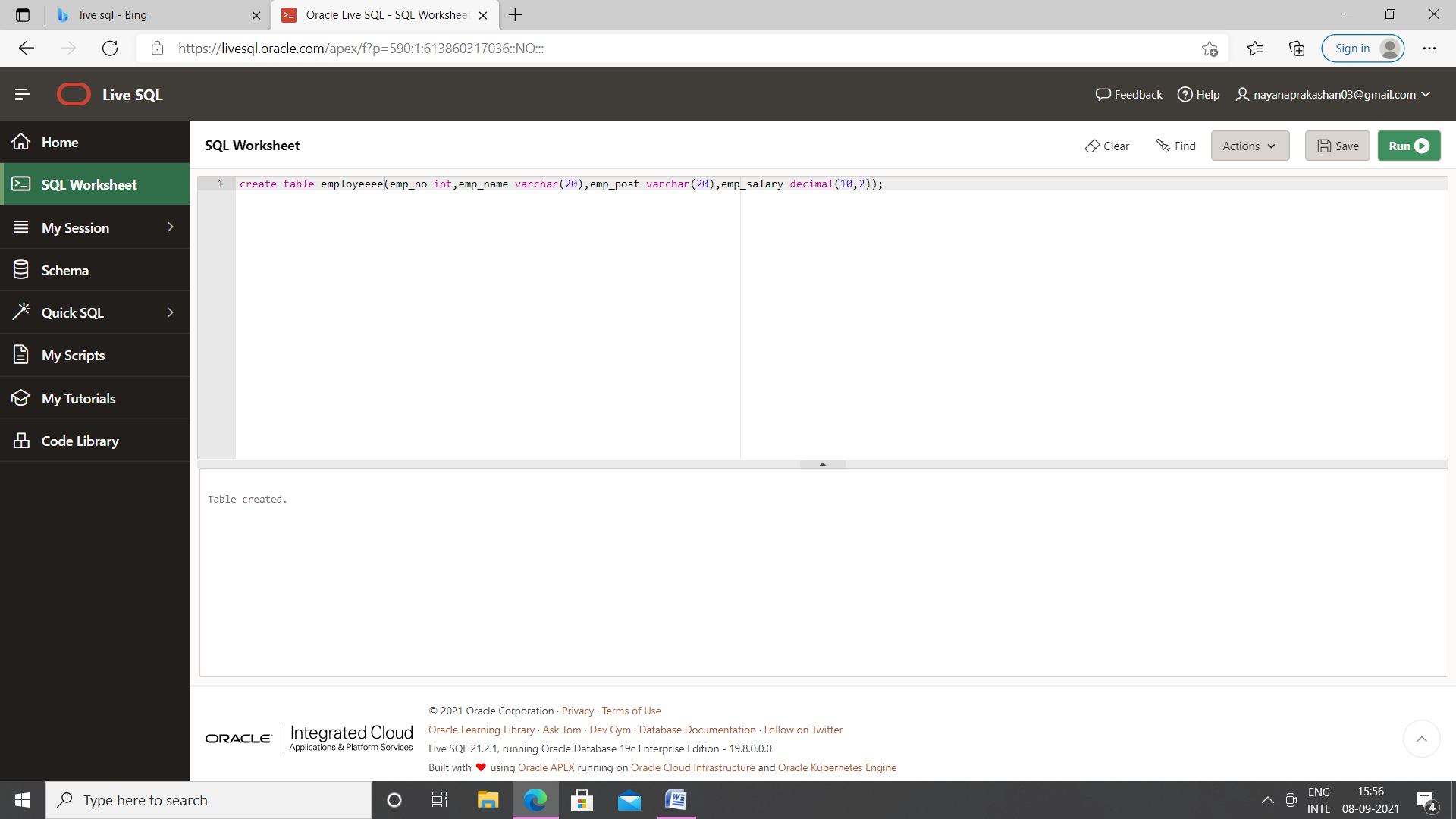
end;

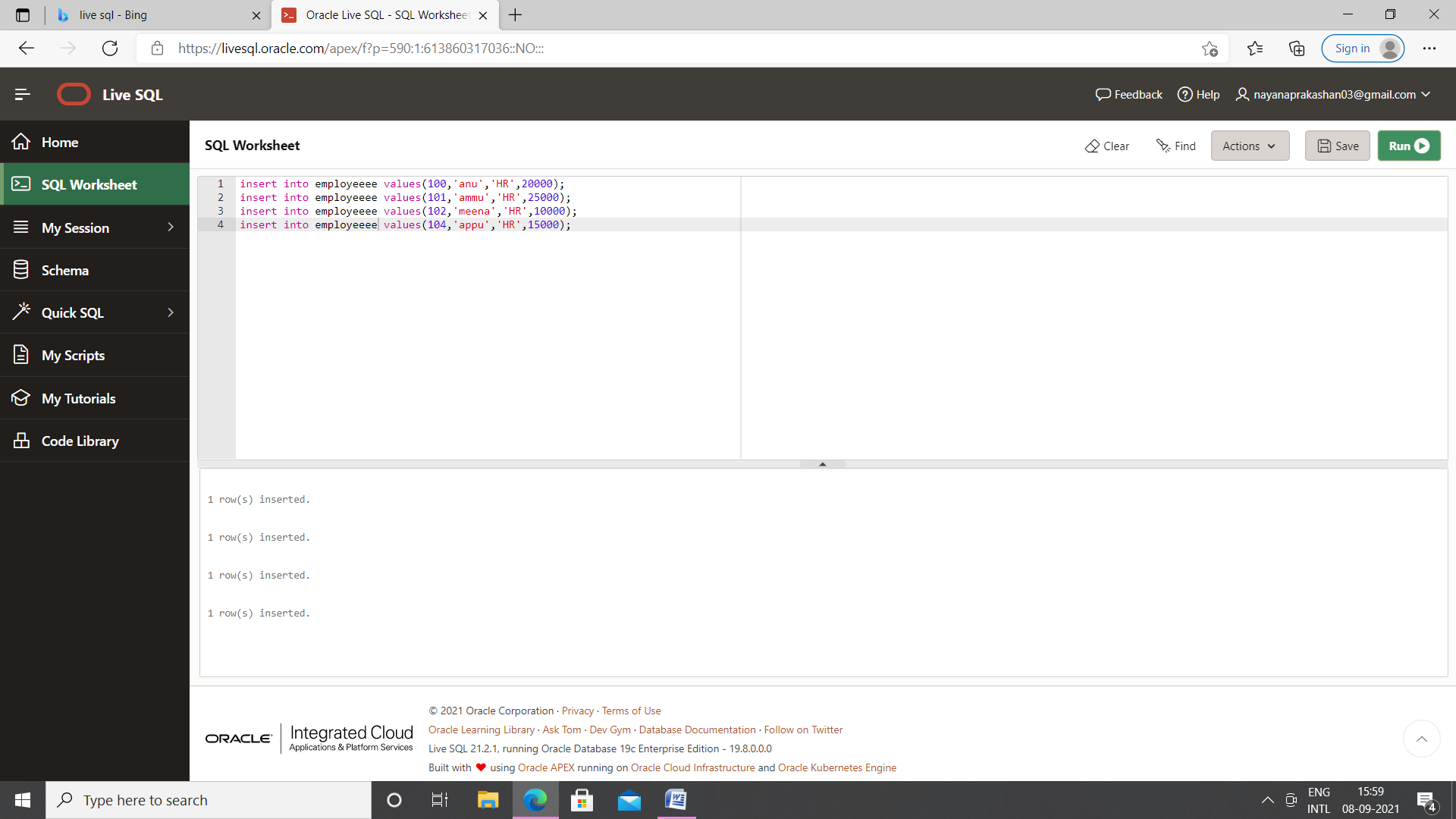
**OUTPUT**



4. Write a PL/SQL program to find the salary of an employee in the EMP table (Get the empno from the user). Find the employee drawing minimum salary. If the minimum salary is less than 7500, then give an increment of 15%. Also create an emp %rowtype record. Accept the empno from the user, and display all the information about the employee.

Table creation





**PL/SQL**

Declare

emno employee.emp\_no%type;

salary employee.emp\_salary%type;

emp\_rec employee%rowtype;

begin

emno:=104;

select emp\_salary into salary from employee where emp\_no=emno;

if salary<7500 then

update employee set emp\_salary=emp\_salary \* 15/100 whereemp\_no=emno;

else

dbms\_output.put\_line('No more increment');

end if;

select \* into emp\_rec from employee where emp\_no=emno;

dbms\_output.put\_line('Employee num: '||emp\_rec.emp\_no);

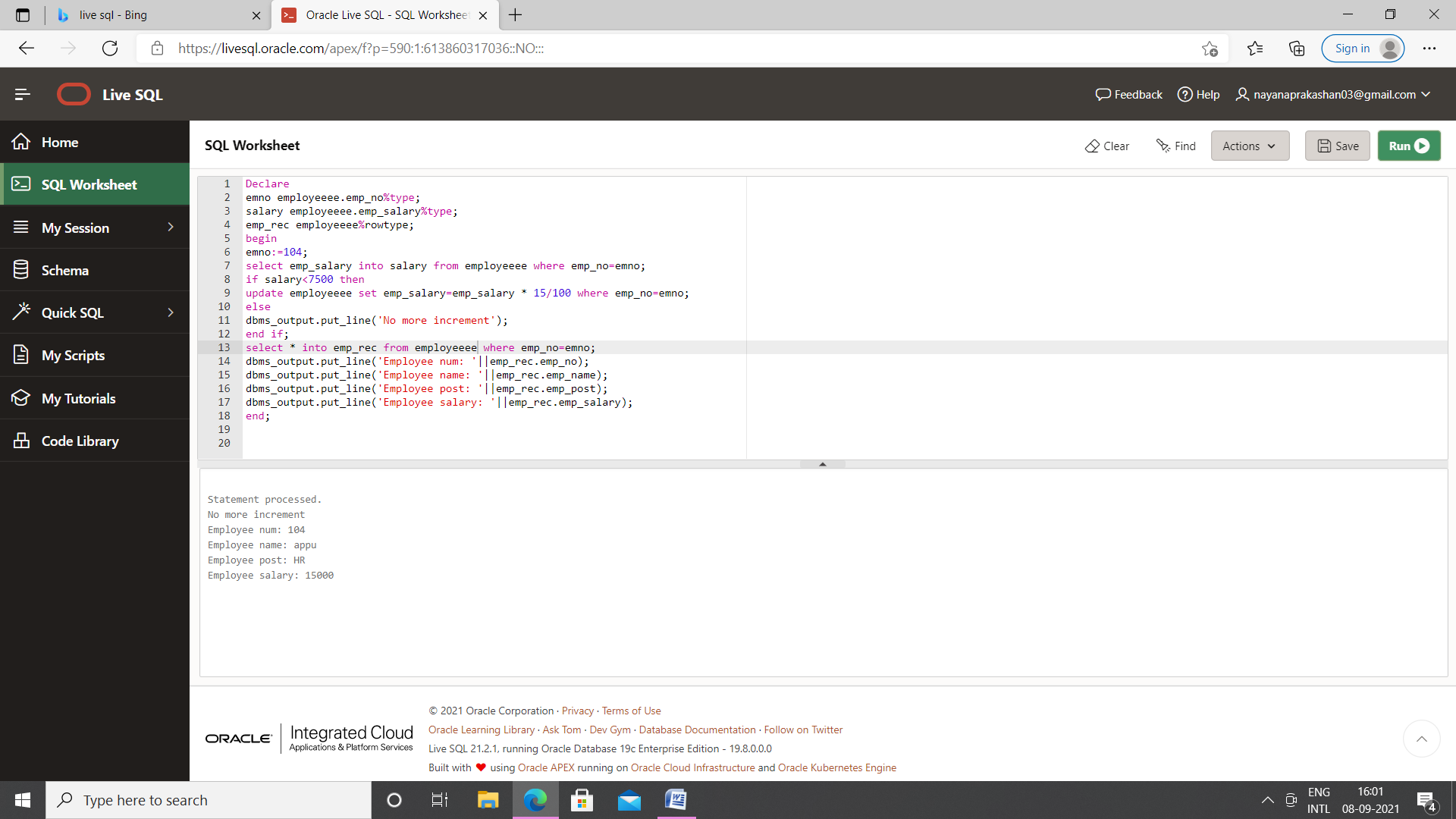
dbms\_output.put\_line('Employee name: '||emp\_rec.emp\_name);

dbms\_output.put\_line('Employee post: '||emp\_rec.emp\_post);

dbms\_output.put\_line('Employee salary: '||emp\_rec.emp\_salary);

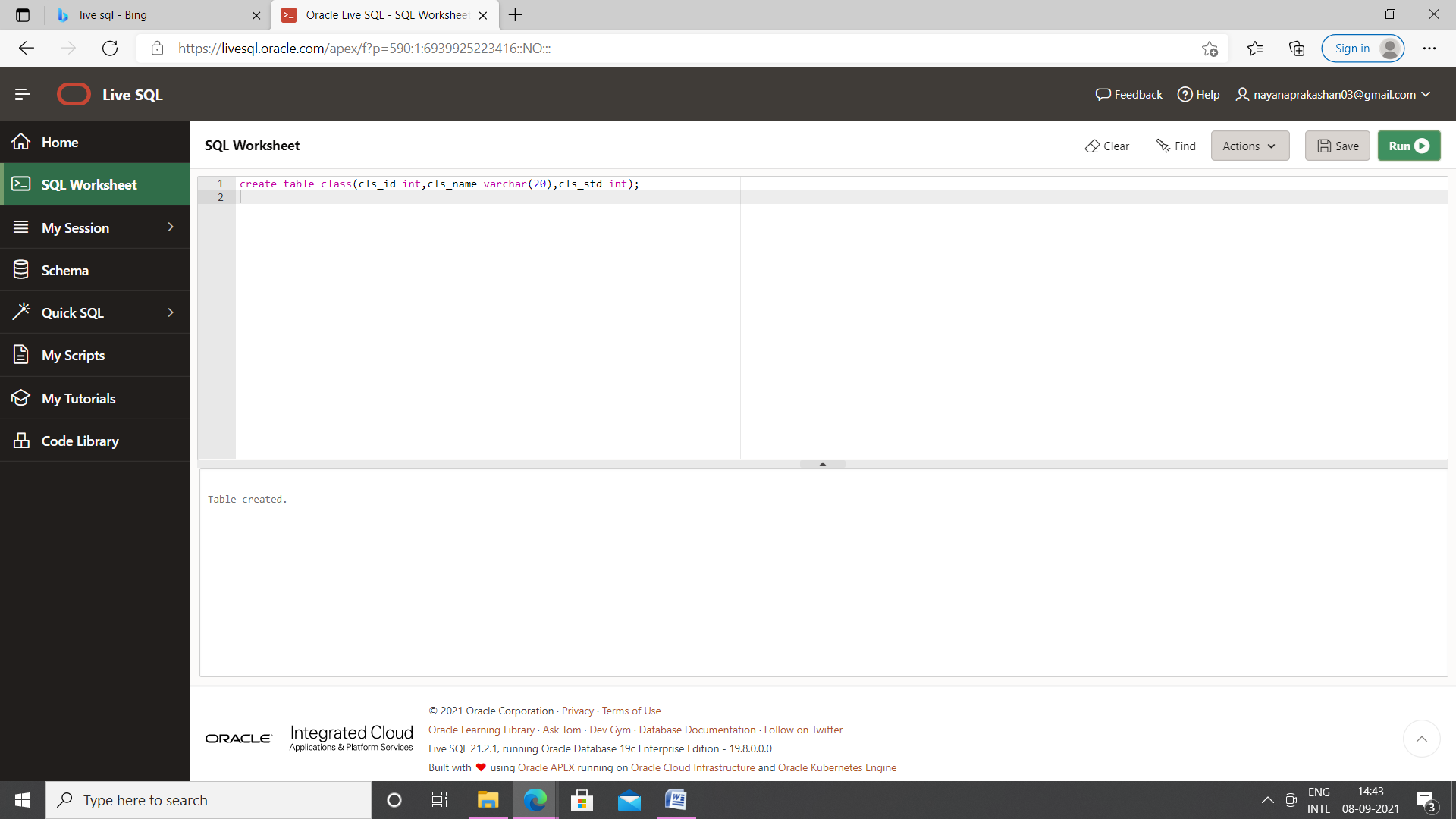
end;

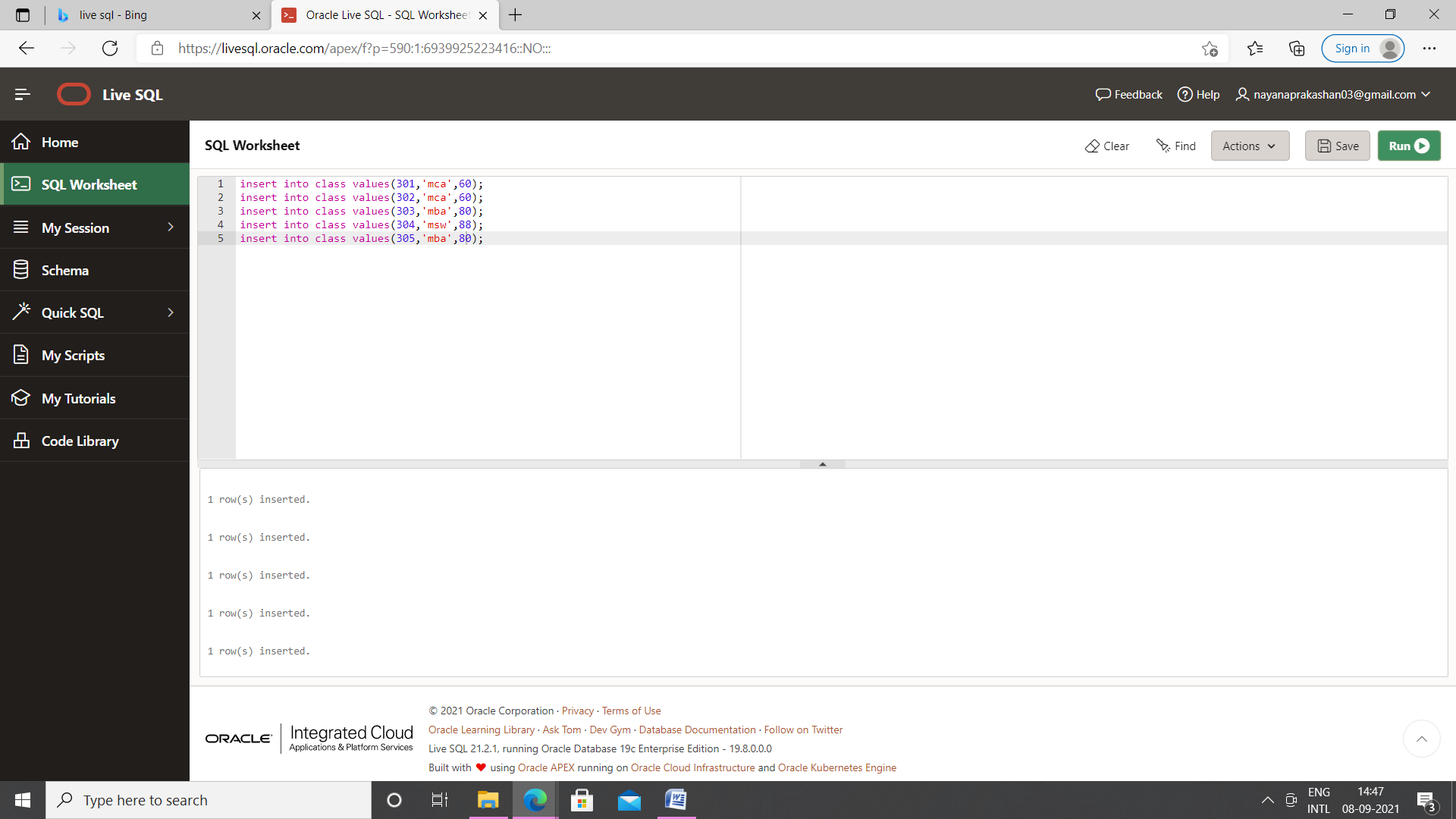
**OUTPUT**



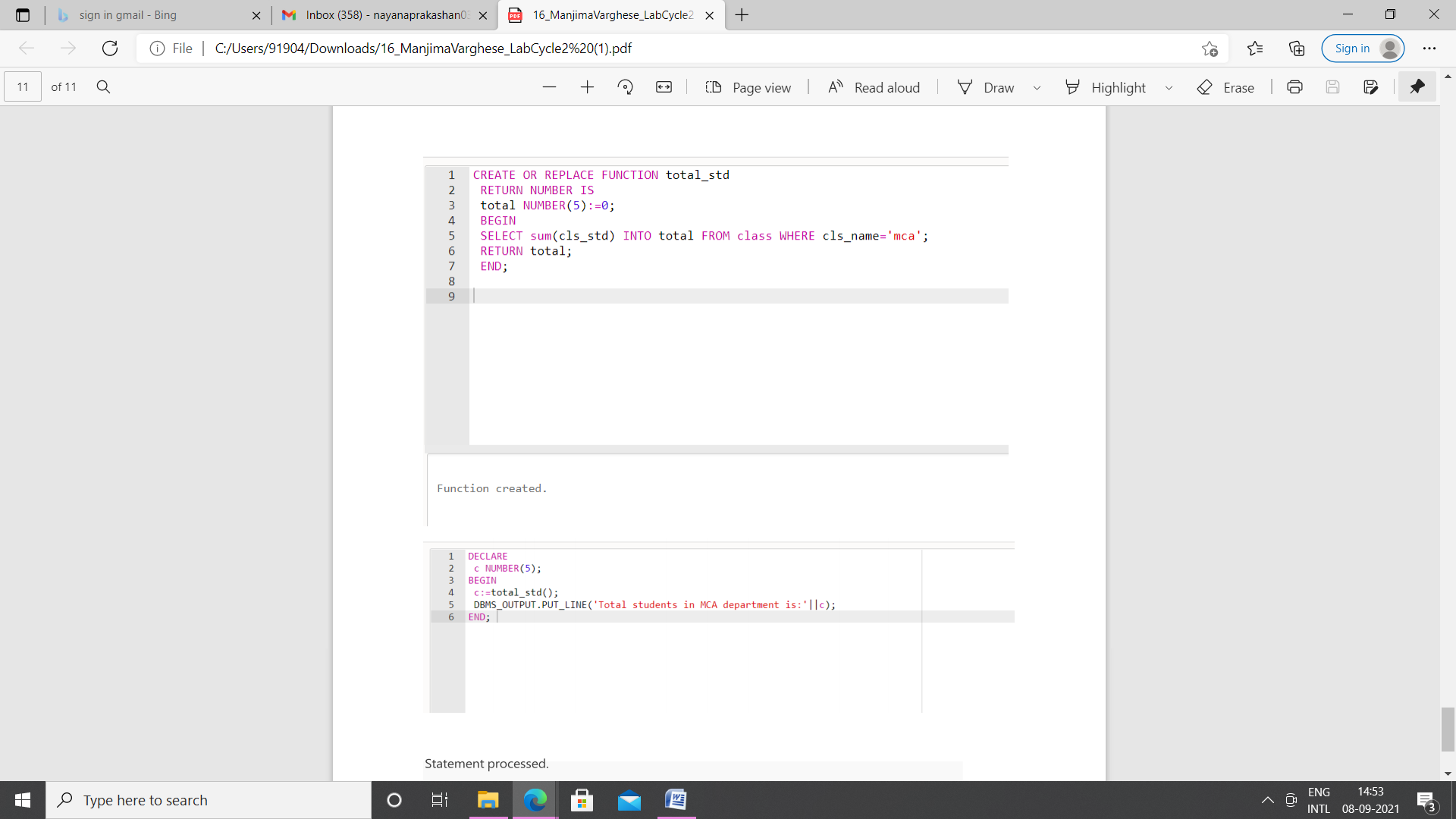
5. Write a PL/SQL function to find the total strength of students present in different classes of the MCA department using the table Class(ClassId, ClassName, Strength)

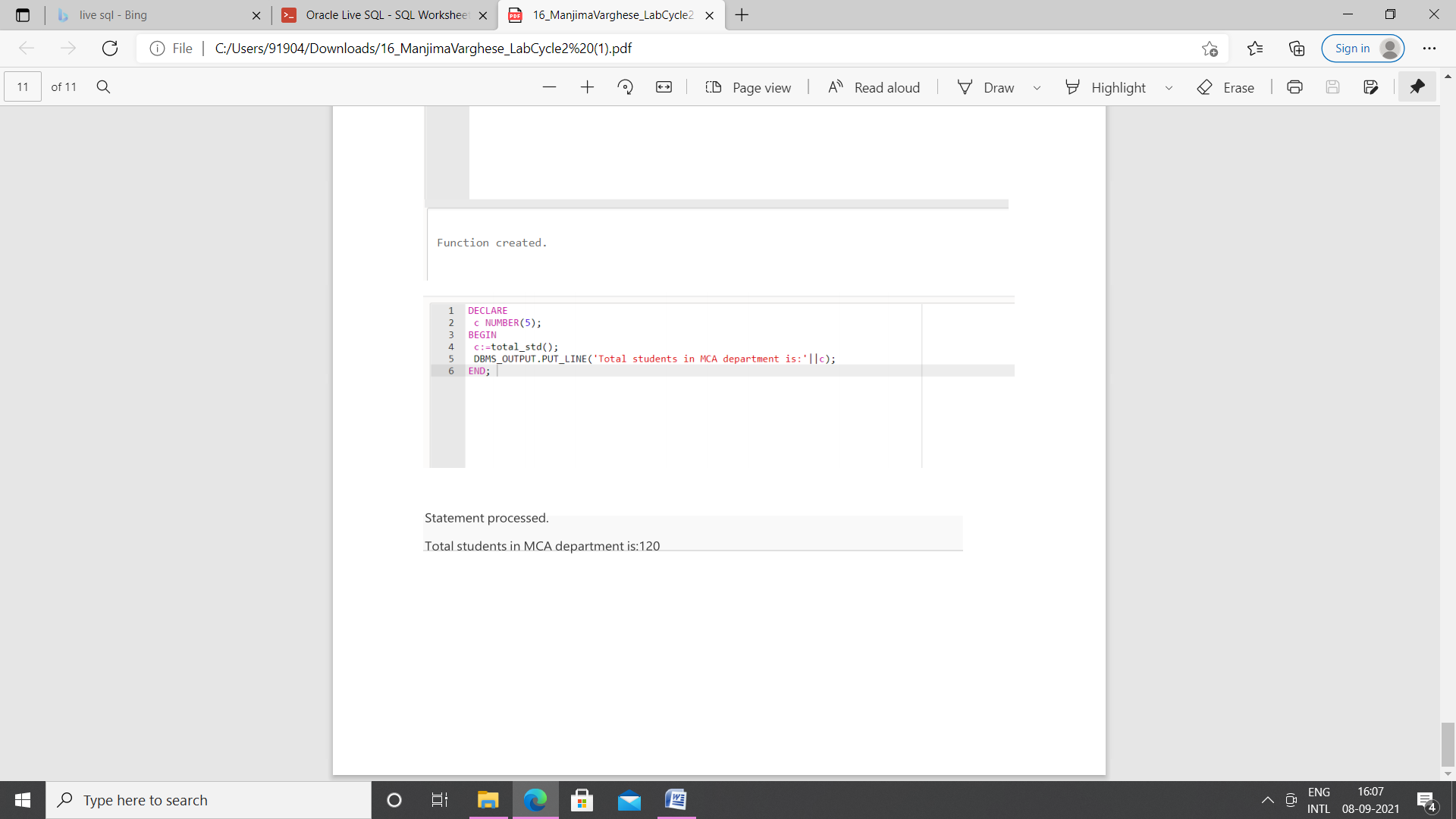
Table creation





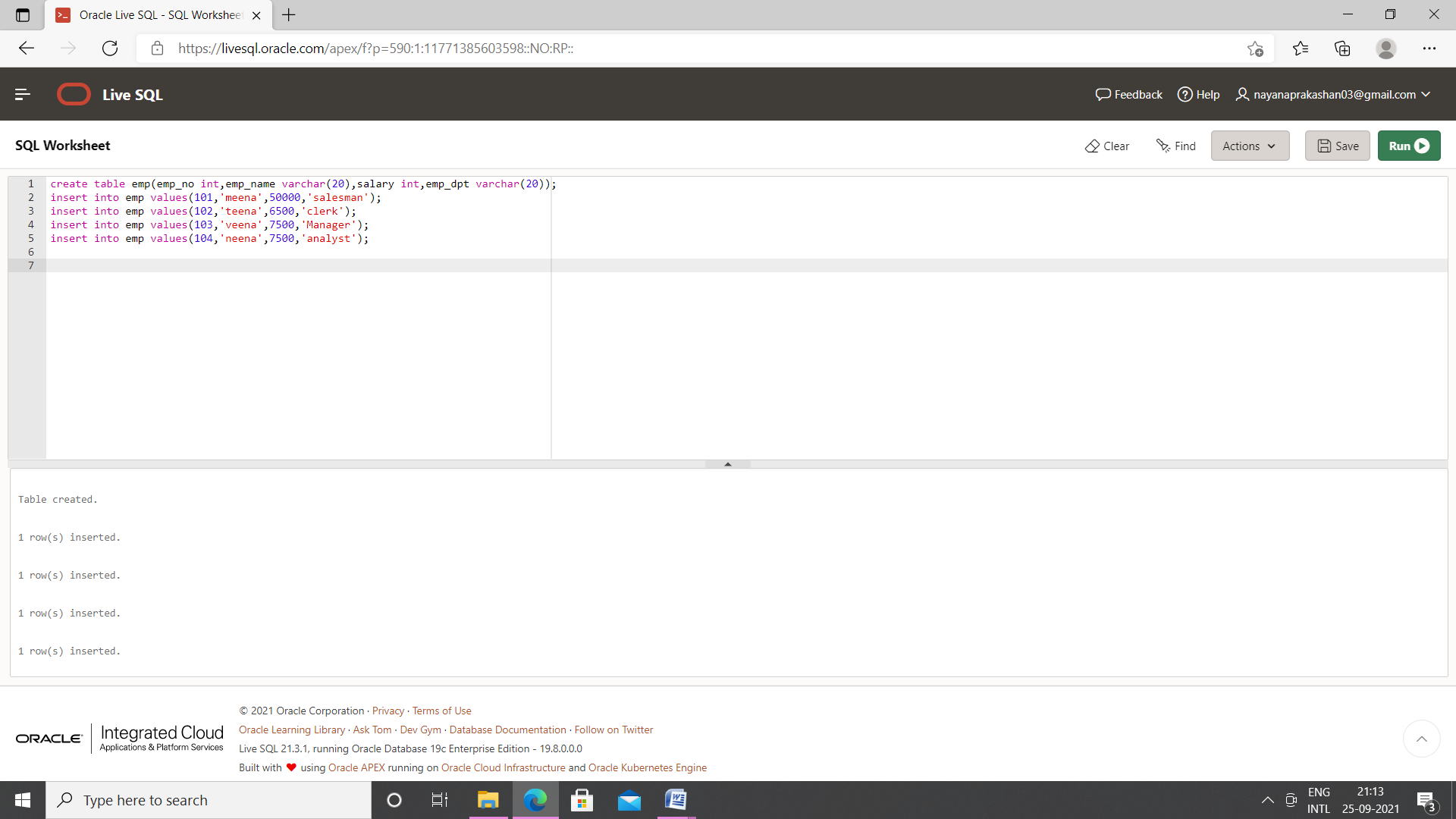
PL/SQL



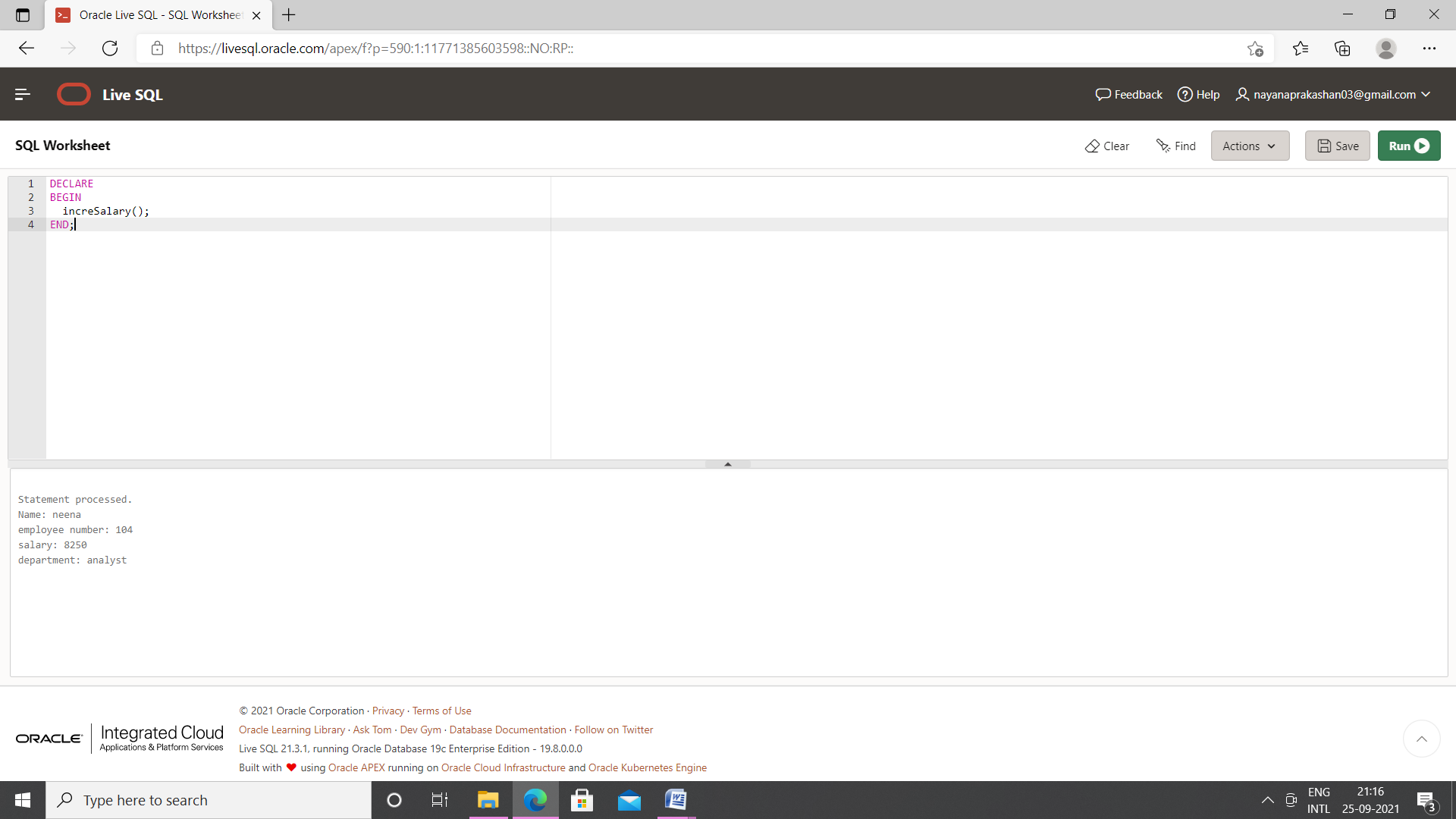


6.Write a PL/SQL **procedure** to increase the salary for the specified employee. Using empno in the employee table based on the following criteria: increase the salary by 5% for clerks, 7% for salesman, 10% for analyst and 20 % for manager. Activate using PL/SQL block.

Table creation

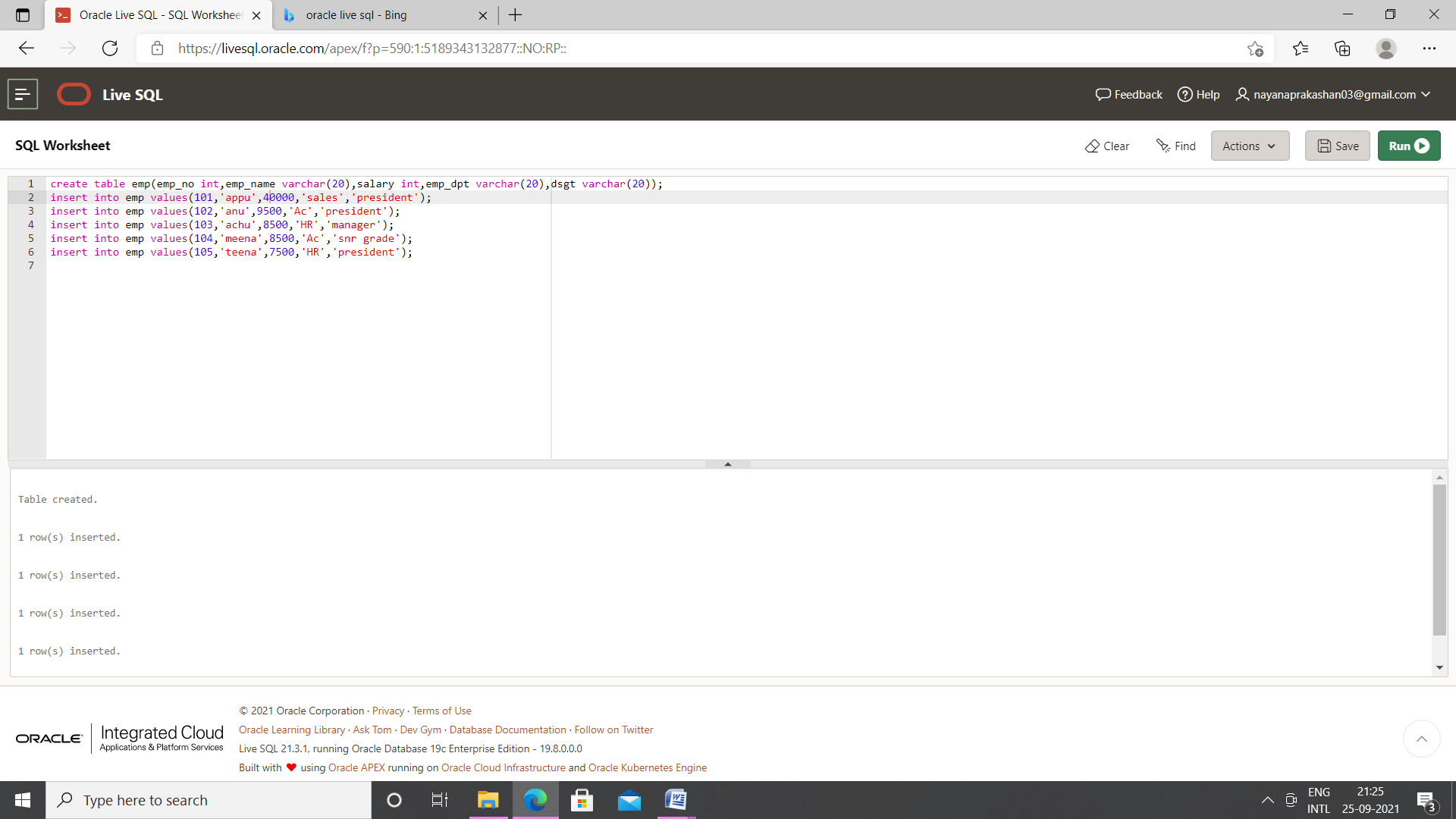


PL/SQL

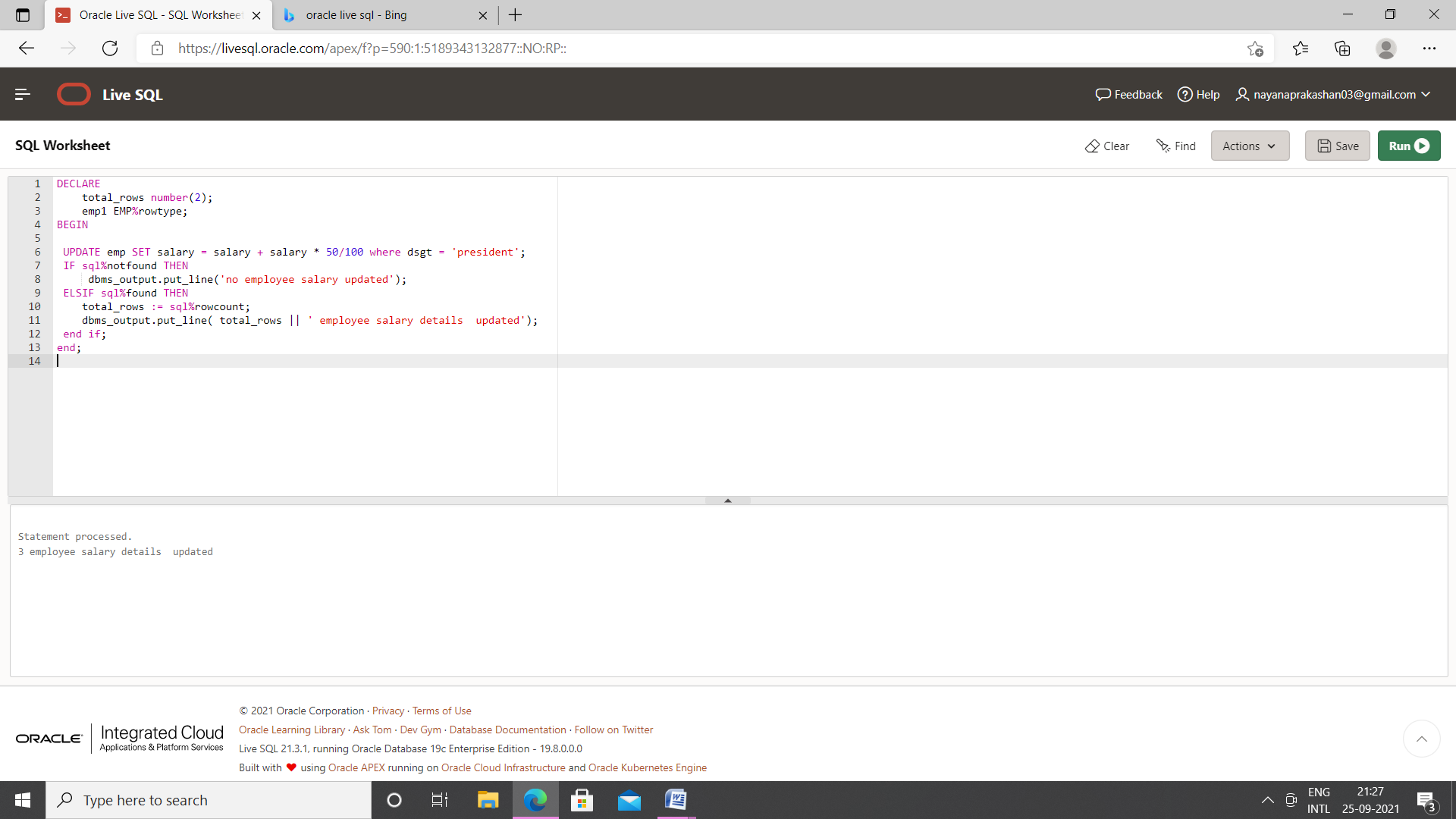


7.Create a **cursor** to modify the salary of ‘president’ belonging to all departments by 50%

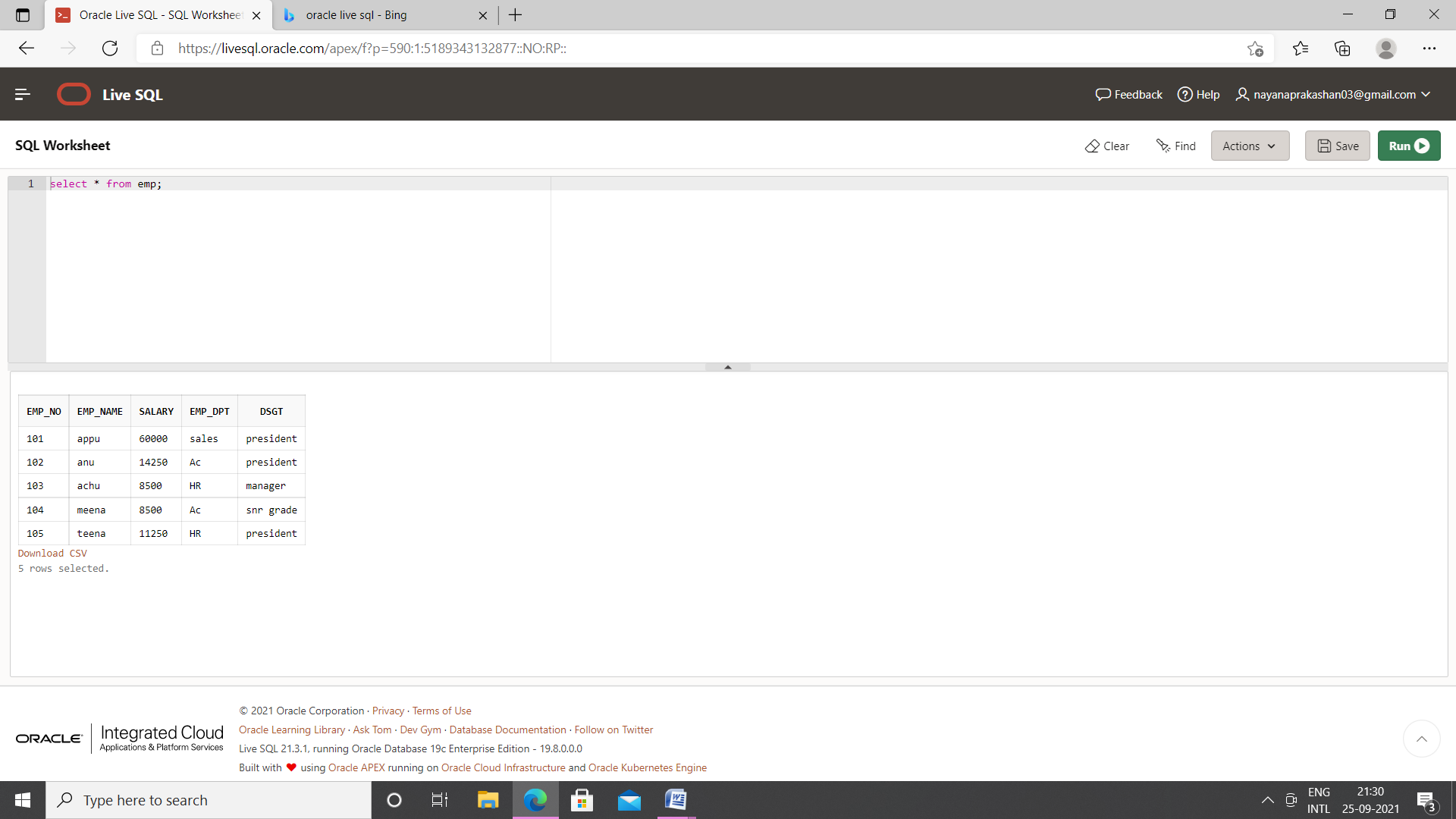
Table Creation



PL/SQL

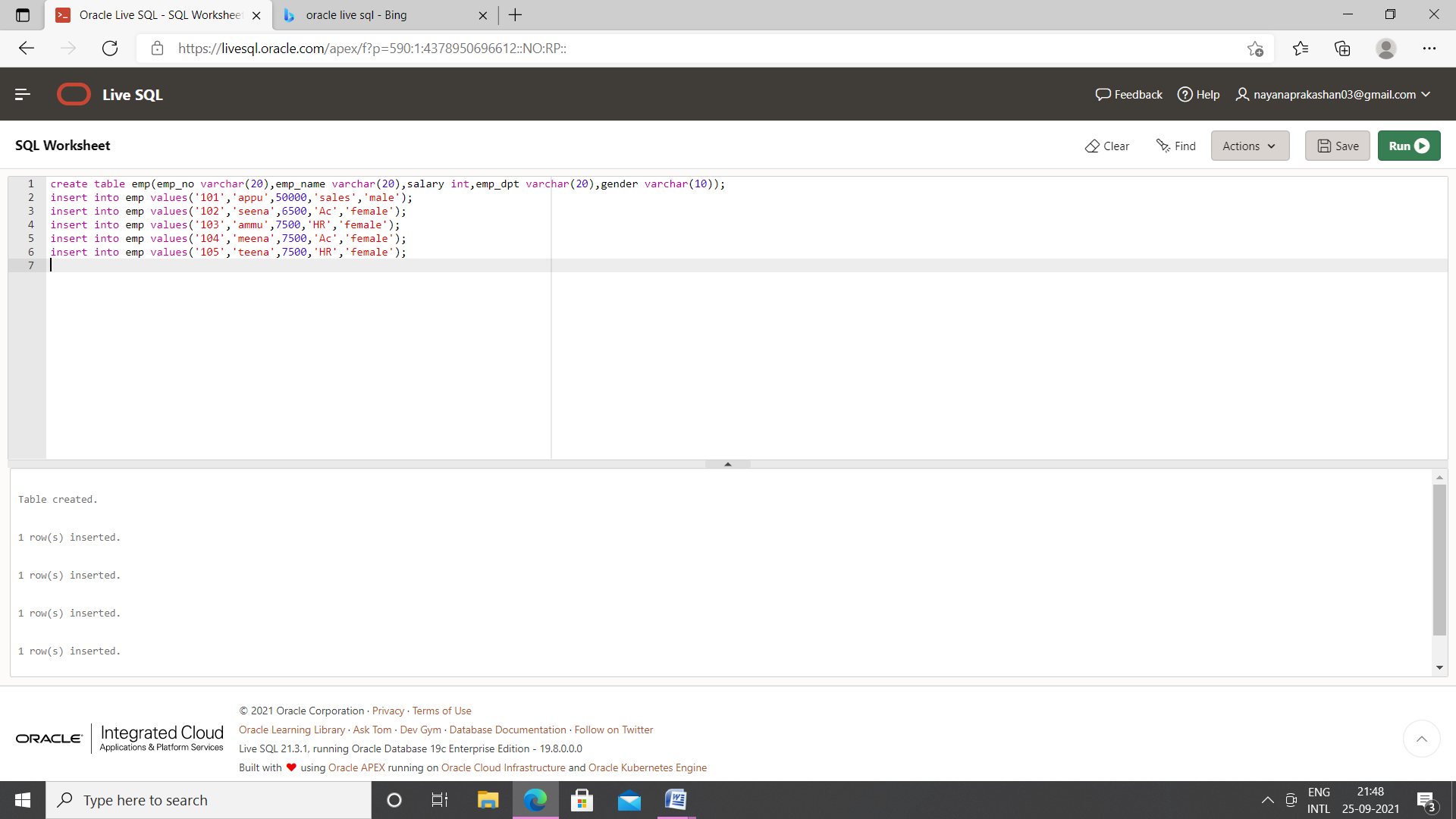


Output

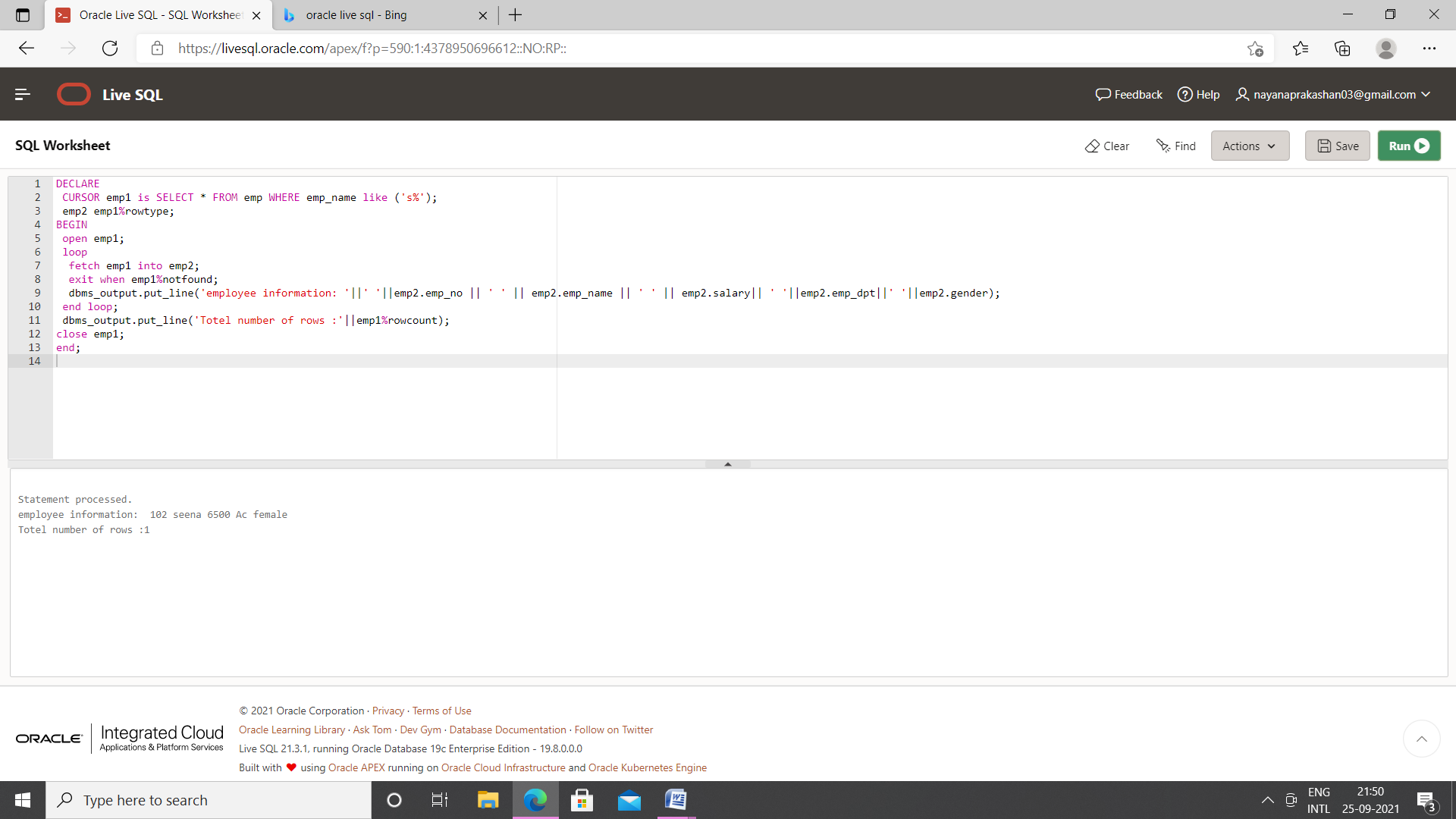


8.Write a **cursor** to display list of Male and Female employees whose name starts with S.

Table Creation

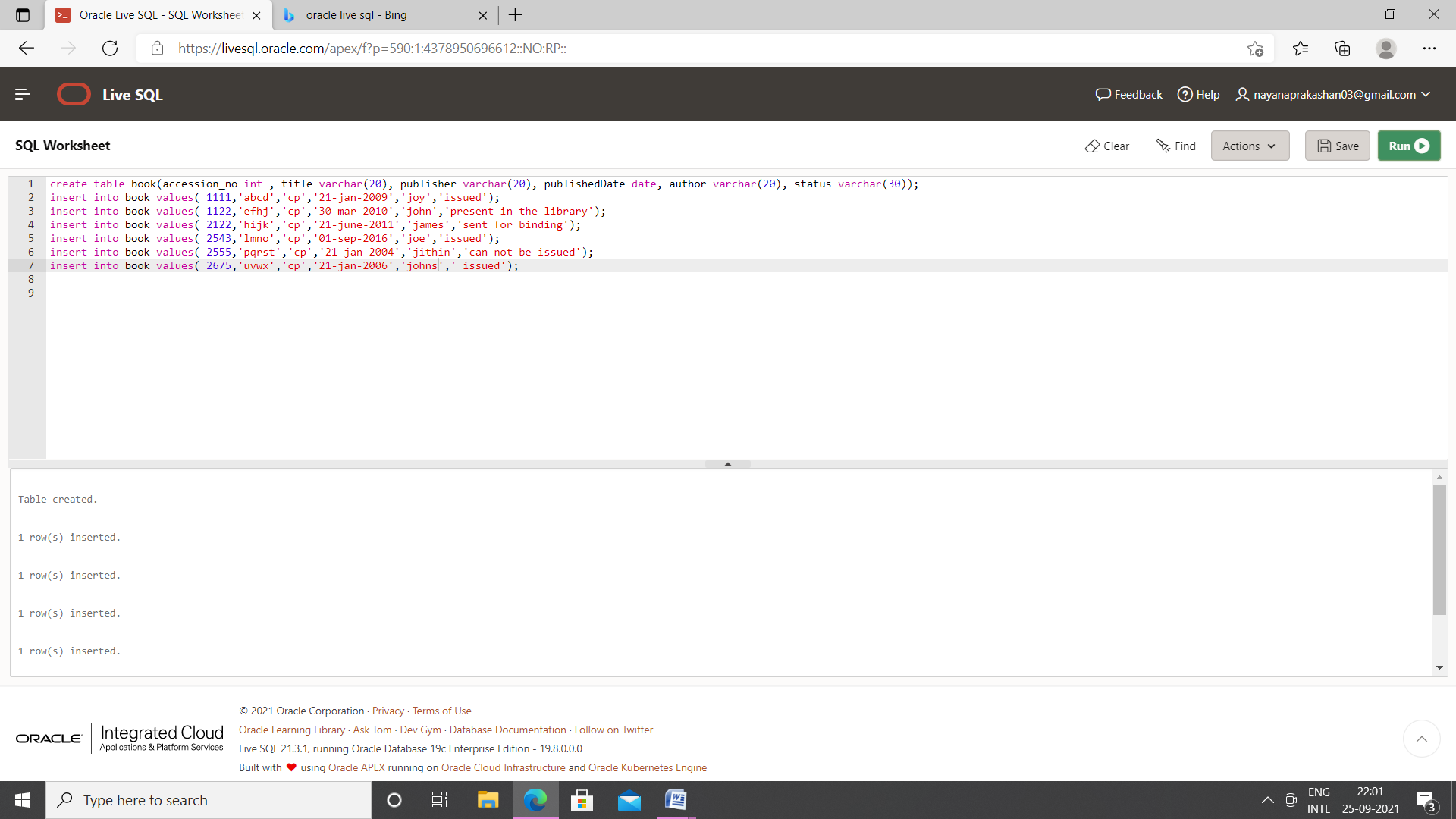


PL/SQL

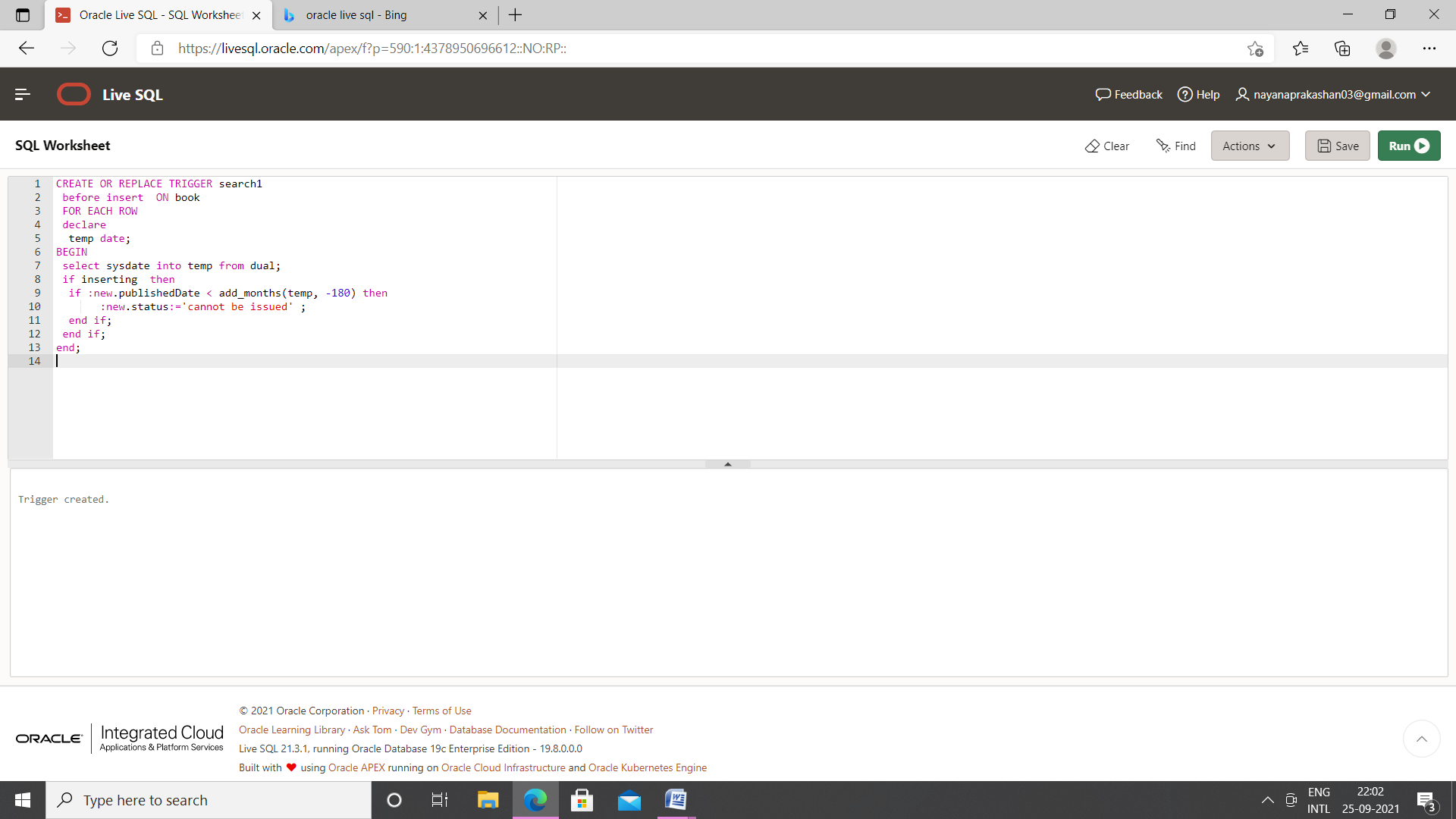


9.Create the following tables for Library Information System: Book : (accession-no, title, publisher, publishedDate, author, status). Status could be issued, present in the library, sent for binding, and cannot be issued. Write a **trigger** which sets the status of a book to "cannot be issued", if it is published 15 years back.

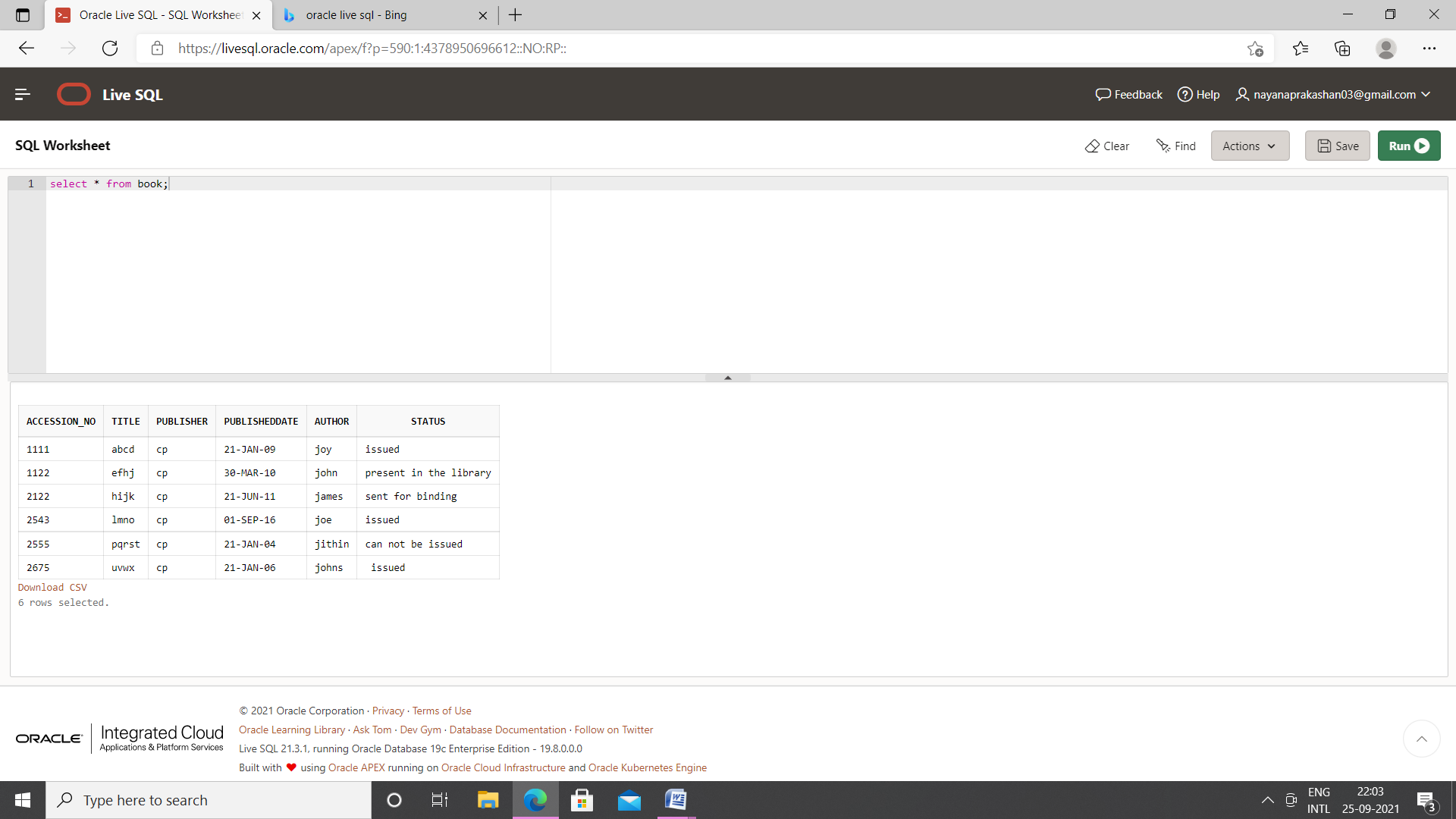
Table creation



PL/SQL

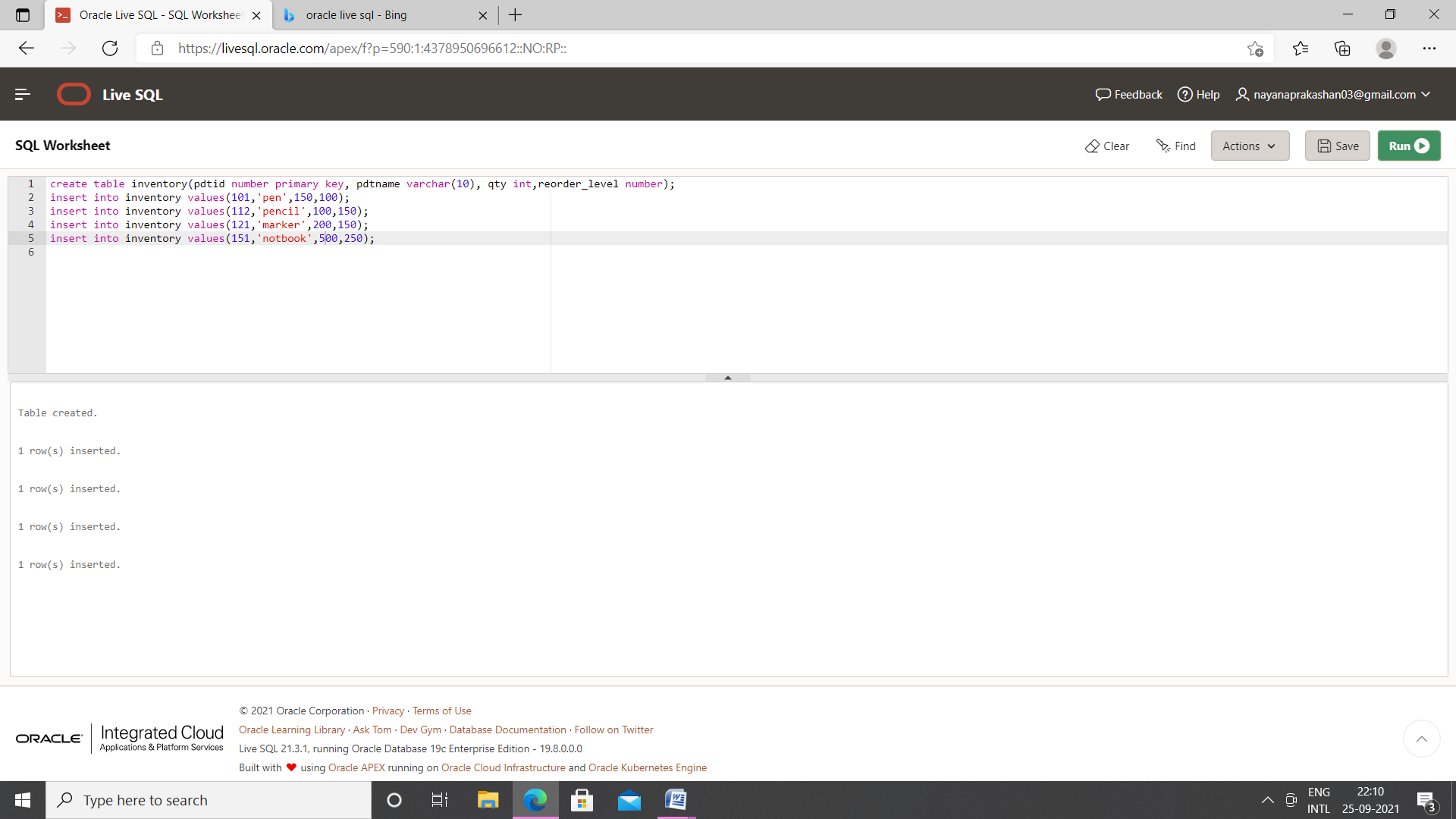


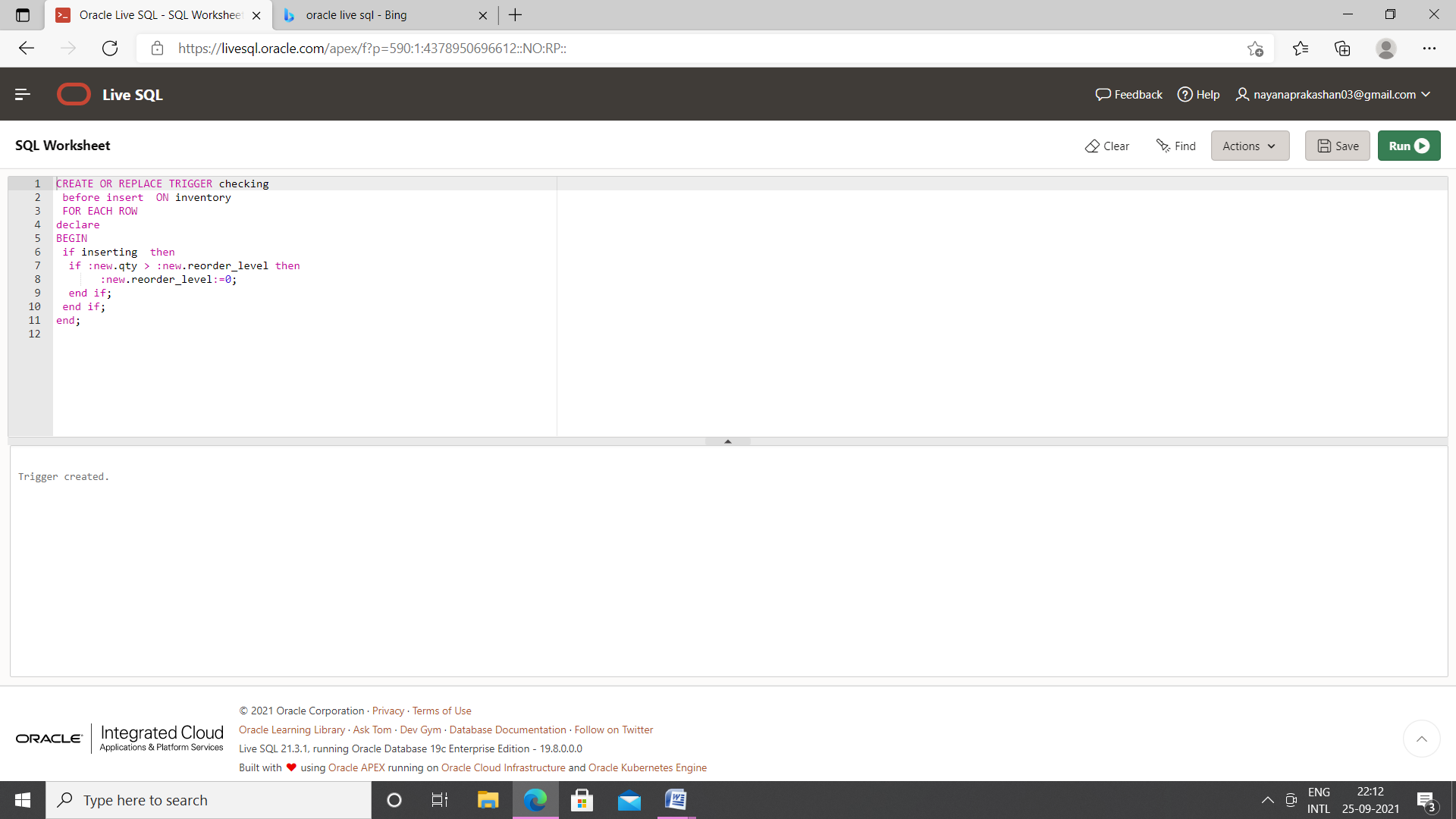
Output



10.Create a table Inventory with fields pdtid, pdtname, qty and reorder\_level. Create a **trigger** control on the table for checking whether qty<reorder\_level while inserting values.

Table creation





Output

