DAY 4

1. **Write a simple PL/SQL (Anonymous block) program to generate multiplication table for a given number**

DECLARE

i NUMBER:=2;

BEGIN

dbms\_output.put\_line('- - - - - - - - - -');

WHILE (i<=3) LOOP

dbms\_output.put\_line(i||' Multiplication table');

dbms\_output.put\_line('- - - - - - - - - -');

FOR j IN 1 .. 10 LOOP

dbms\_output.put\_line(i||'\*'||j||'='||(i\*j));

END LOOP;

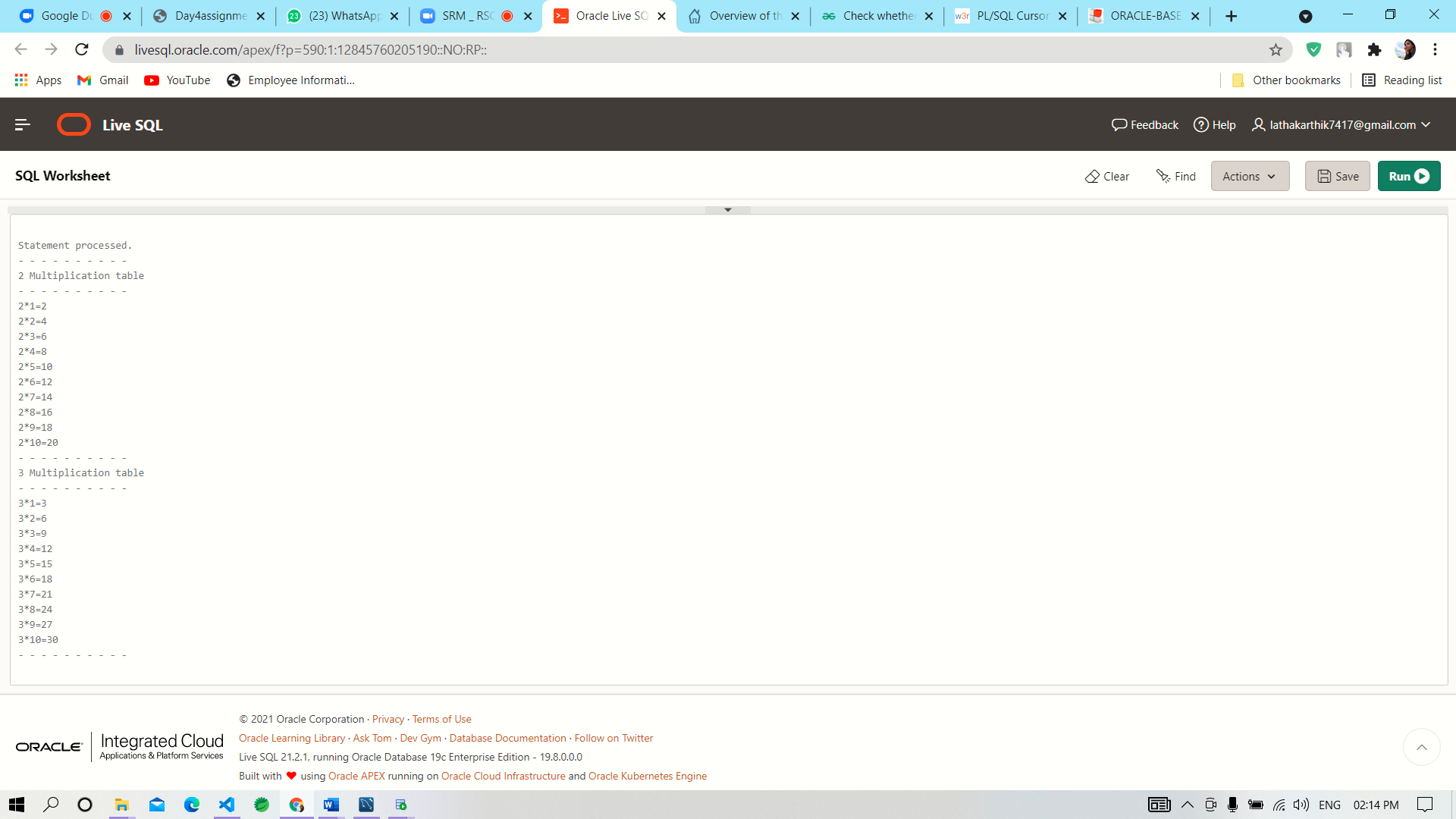
i:=i+1;

dbms\_output.put\_line('- - - - - - - - - -');

END LOOP;

END;

**OUTPUT:**



1. **Write a PL/SQL code to print first 50 whole numbers. Also insert the list in temp table which is created with only one column of number datatype**

create table temp(wholenumber number(2));

Begin

for loopcount in 1..50 loop

insert into temp values(loopcount);

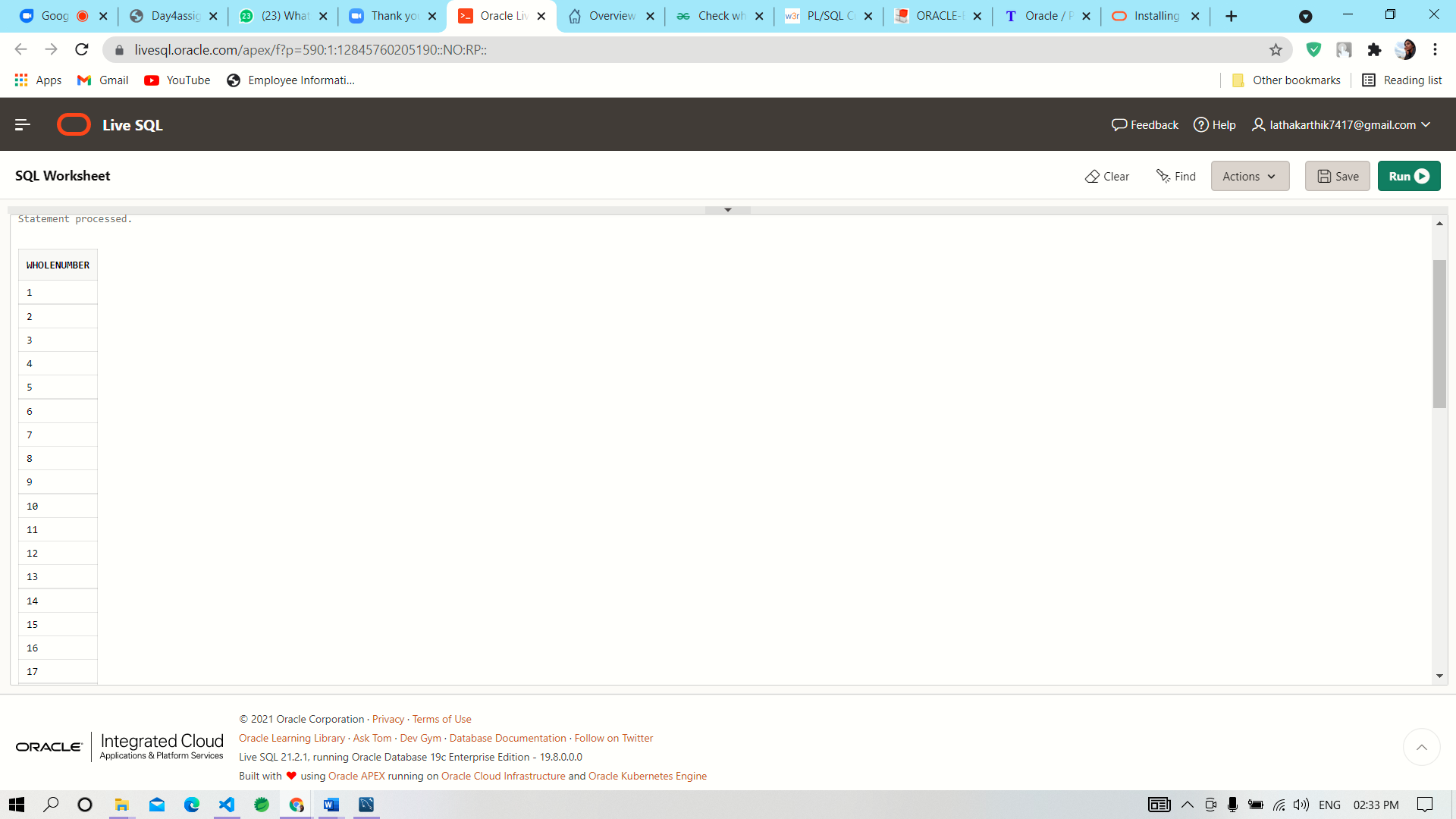
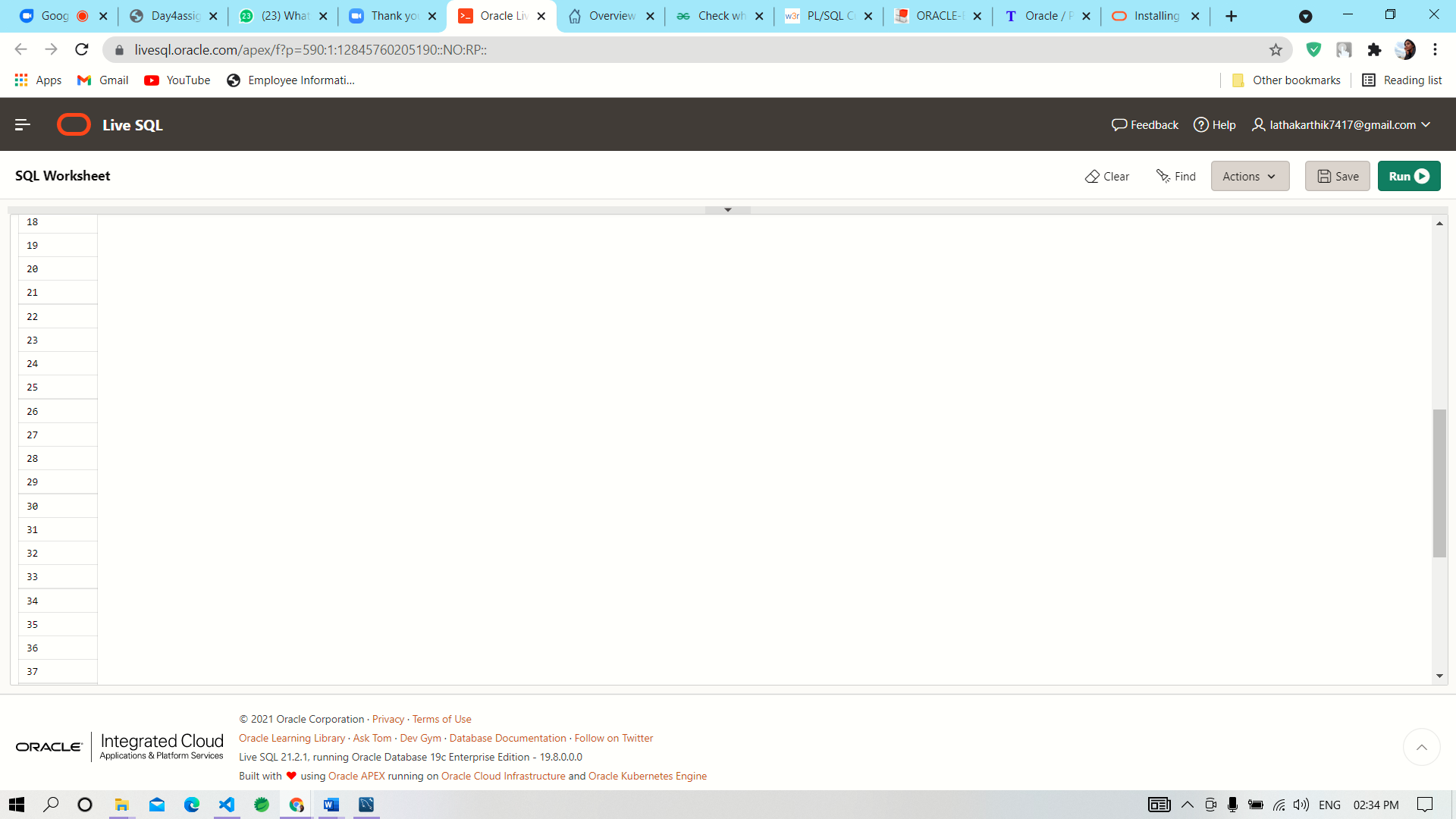
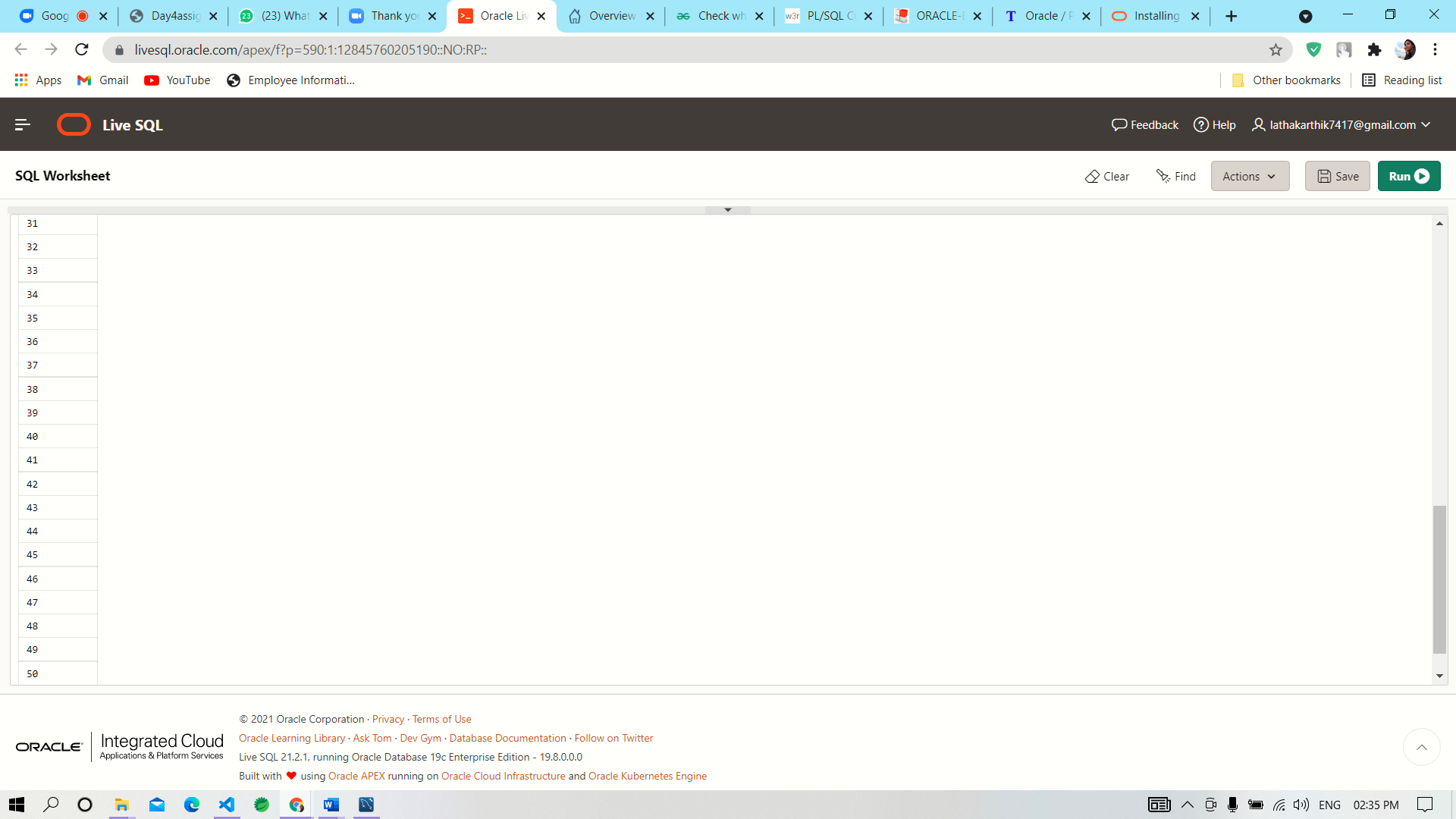
end loop;

end;

/

select \* from temp;

**OUTPUT:**

**3. Using cursor display the details of all those employees in employee table whose sum of sal and comm is more than 3000. (Create your employee table’s comm column with some NULL values).**

CREATE TABLE EMPL{

EMPID NUMBER(3) PRIMARY KEY,

EMPNAME VARCHAR(10),

EMAIL VARCHAR(20),

PHONENUMBER VARCHAR(10),

COMM DECIMAL(8,2),

SALARY DECIMAL(8,2)

};

INSERT INTO EMPL VALUES(101,”Babu”,[babu101@gmail.com](mailto:babu101@gmail.com),”9878987898”,1000,8000.00);

INSERT INTO EMPL VALUES (102,"Somu","somu102@gmail.com","9448534120 ",200,NULL);

INSERT INTO EMPL VALUES (103,"Ramu","ramu103@gmail.com","9448886634 ",400, 9000.00);

INSERT INTO EMPL VALUES (104,"Gobu","gobu104@gmail.com","9447783412 ",500,3000.00);

INSERT INTO EMPL VALUES (105,"Anbu","anbu105@gmail.com","9848341200 ",4500,9000.00);

INSERT INTO EMPL VALUES (106,"Rahu","rahuu106@gmail.com","9848341400 ",600,5000.00);

INSERT INTO EMPL VALUES (107,"Rahul","rahul107@gmail.com","9898341400",NULL,2000.00);

DESC EMPL;

SELECT \* FROM EMPL;

DECLARE

c\_empid empl.empid%TYPE :=&ceid;

c\_empname empl.empname%TYPE;

c\_comm empl.comm%TYPE;

c\_salary empl.salary%TYPE;

CURSOR empcursor IS

SELECT empid, empname, email, salary FROM EMPL where salary>20000;

BEGIN

OPEN empcursor;

LOOP

FETCH empcursor INTO c\_empid,c\_empname,c\_comm,c\_salary;

EXIT WHEN empcursor %NOTFOUND;

END LOOP;

END;

/

**------------------------------------------------------------------------------------------------------------------------------------**

4. Create a trigger for the employee table, which makes the entry in ENAME column in uppercase.

select \* from employees;

CREATE OR REPLACE TRIGGER emp\_trg

BEFORE INSERT OR UPDATE ON employees

FOR EACH ROW

DECLARE

dup\_flag INTEGER;

BEGIN

:NEW.empname := UPPER(:NEW.empname);

END;

**------------------------------------------------------------------------------------------------------------------------------------**

**5. Write a PL/SQL block to fire any two built in exceptions in ORACLE by assuming your own data in a table.**

DECLARE

temp emp%rowtype;

BEGIN

SELECT \* INTO temp FROM emp where empid=102;

EXCEPTION

WHEN no\_data\_found THEN

dbms\_output.put\_line('No data found!');

END;

**------------------------------------------------------------------------------------------------------------------------------------**

**6) Write a function in PL/SQL to check whether the given string is a palindrome or not. (Use any appropriate built-in functions available)**

DECLARE

s VARCHAR2(10) := 'MAM';

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

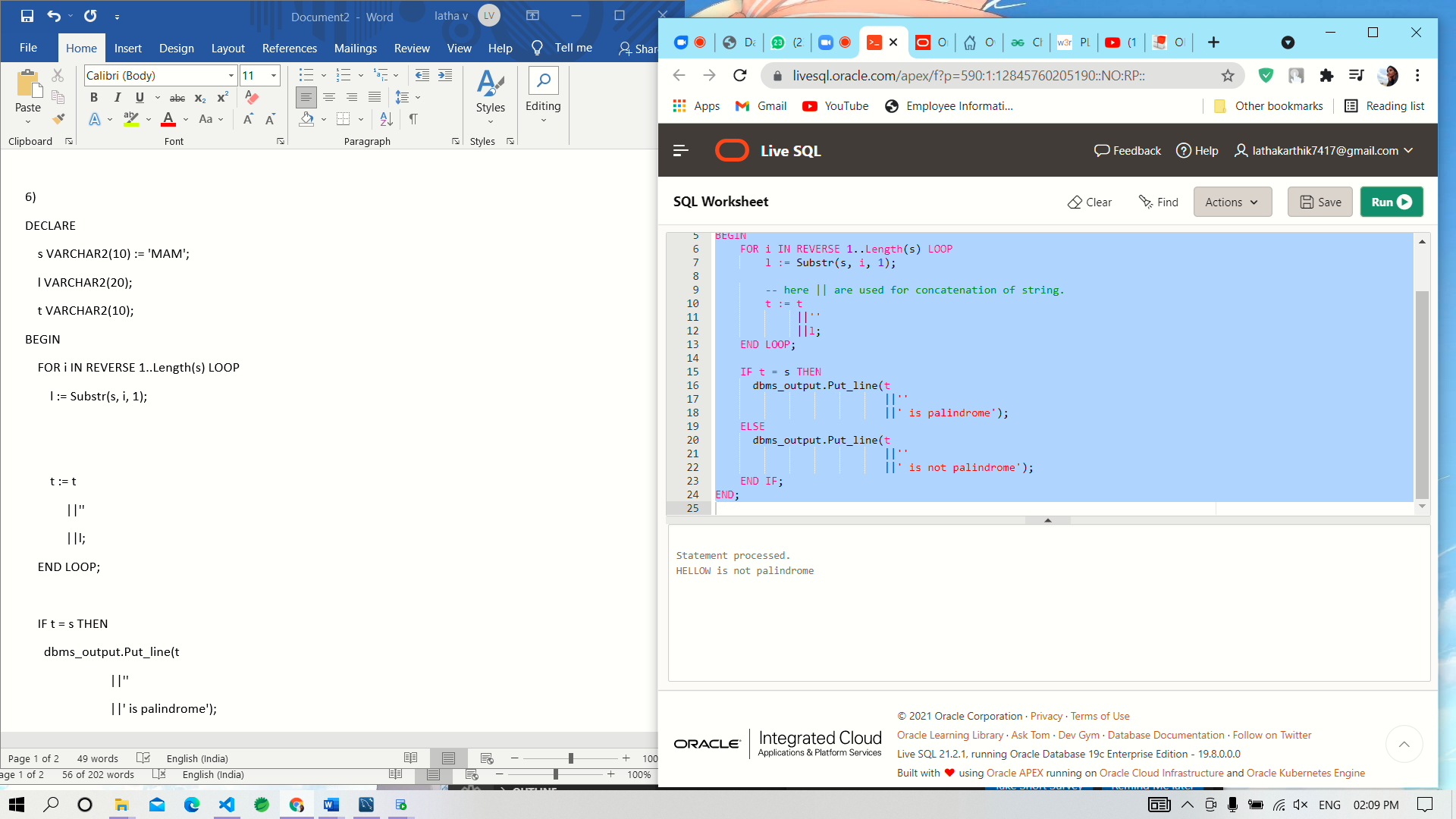
||''

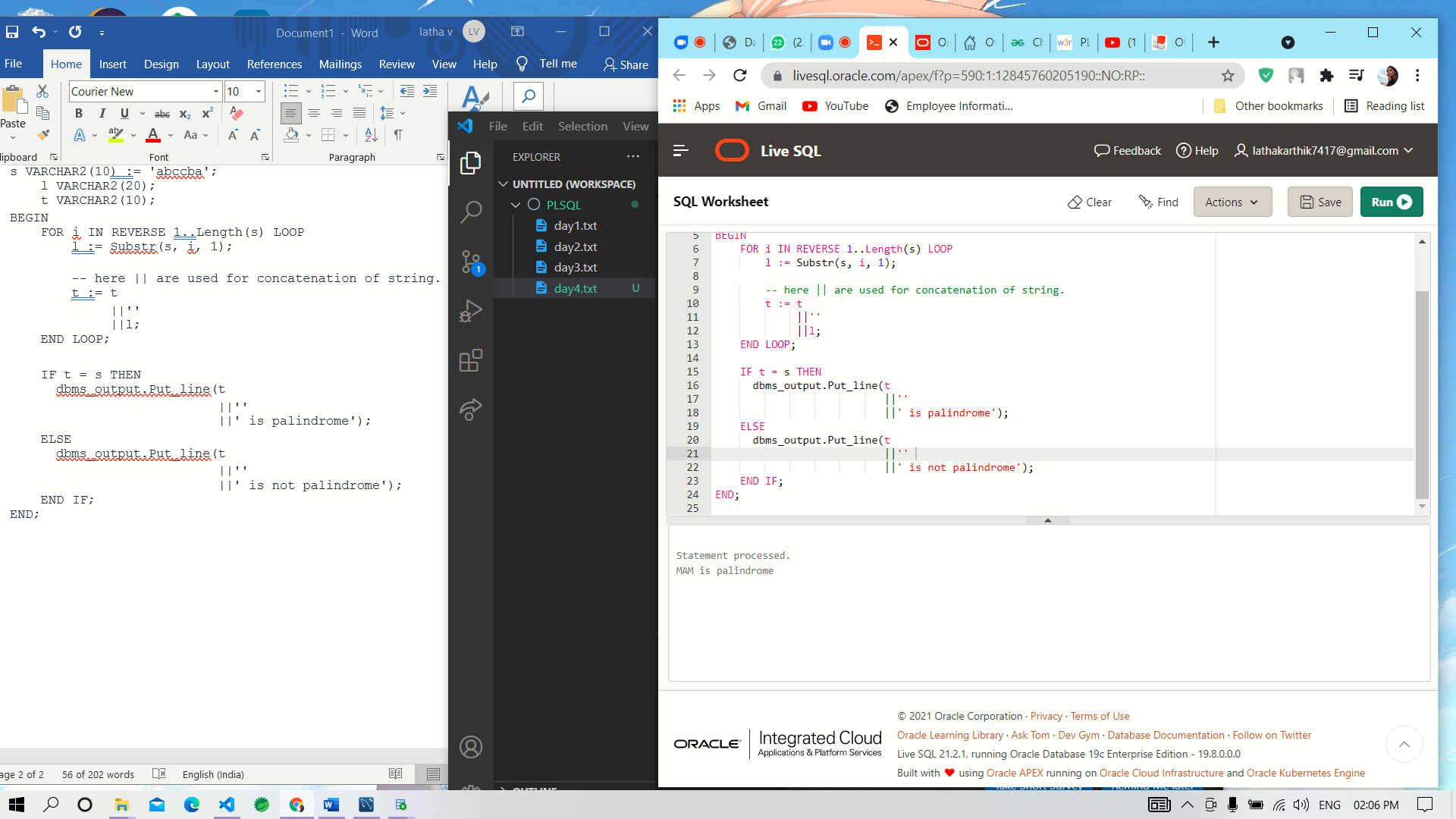
||' is not palindrome');

END IF;

END;

**OUTPUT:**

1. **sample for not a palindrome**
2. **Sample for a palindrome value**



**--------------------------------------------------------------------------------------------------------------------------------------**

**7. Write a PL/SQL block to fire any one user defined exception by assuming your own data in a table.**

excep EXCEPTION;

n number;

BEGIN

FOR n IN (SELECT \* FROM EMPL) LOOP

if n.salary=5000 THEN

RAISE excep;

END IF;

END LOOP;

EXCEPTION

WHEN excep THEN dbms\_output\_Put\_line(‘this is an custom exception’);

WHEN OTHERS THEN dbms\_output\_Put\_line(‘Error occurred’);

END;

**------------------------------------------------------------------------------------------------------------------------------------**

**8. Write a named procedure to delete records available in your table.**

CREATE OR REPLACE PROCEDURE empdel

IS

BEGIN

DELETE FROM EMPL WHERE EMPID=104;

COMMIT;

END;

**------------------------------------------------------------------------------------------------------------------------------------**