**MAJOR ASSIGNMENT-SQL[MAJOR]**

1.Create a table “Station” to store information About weather observation stations:

ID,Number,Primarykey,CITYCHAR(20),STATE,CHAR(2),LAT\_NNumber,LONG\_W

Number.

ANS) CREATE TABLE Station (

ID NUMBER NOT NULL PRIMARY KEY,

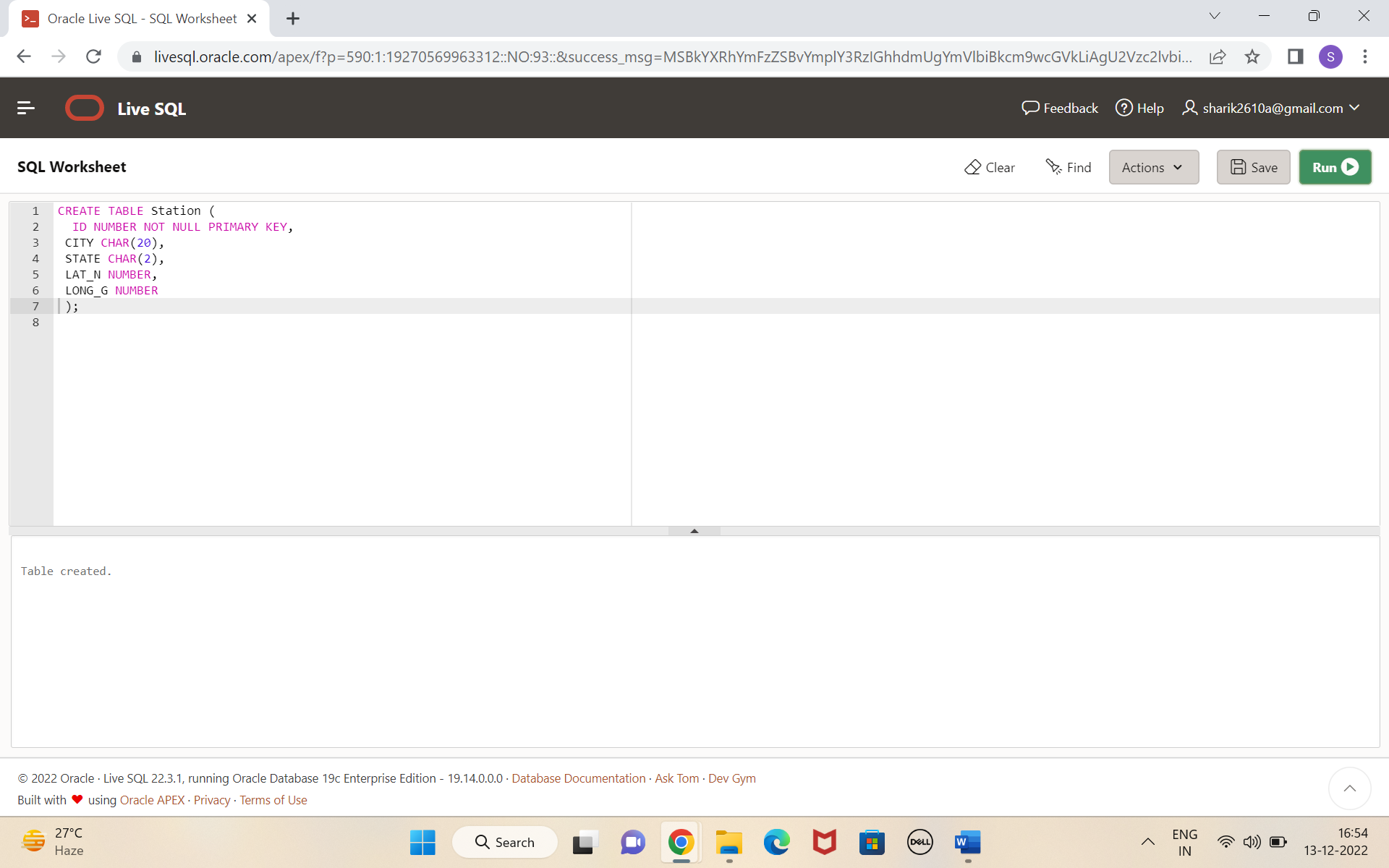
CITY CHAR (20),

STATE CHAR (2),

LAT\_N NUMBER,

LONG\_G NUMBER

);

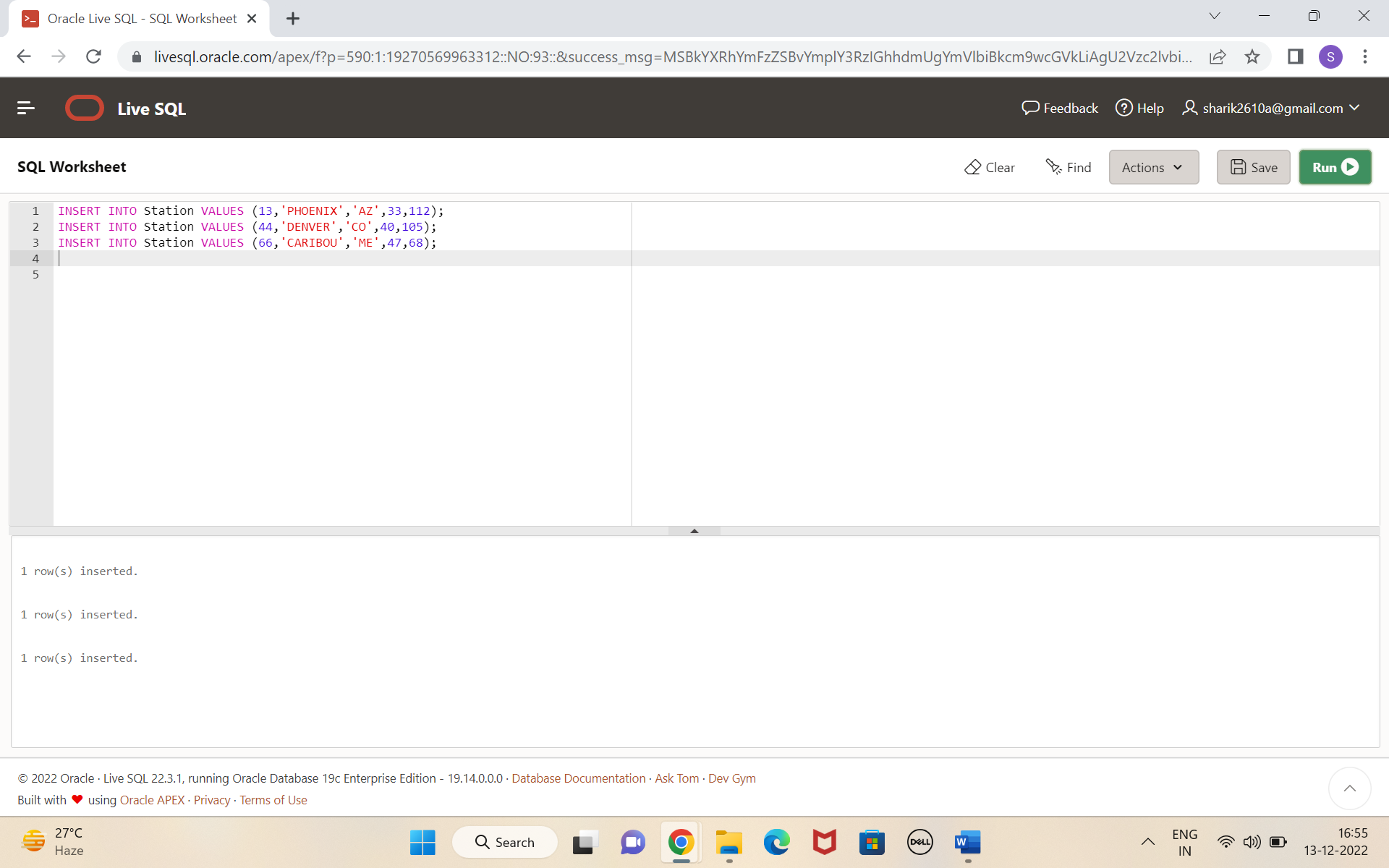


2. Insert the following records in to the table

ANS) INSERT INTO Station VALUES (13,'PHOENIX','AZ',33,112);

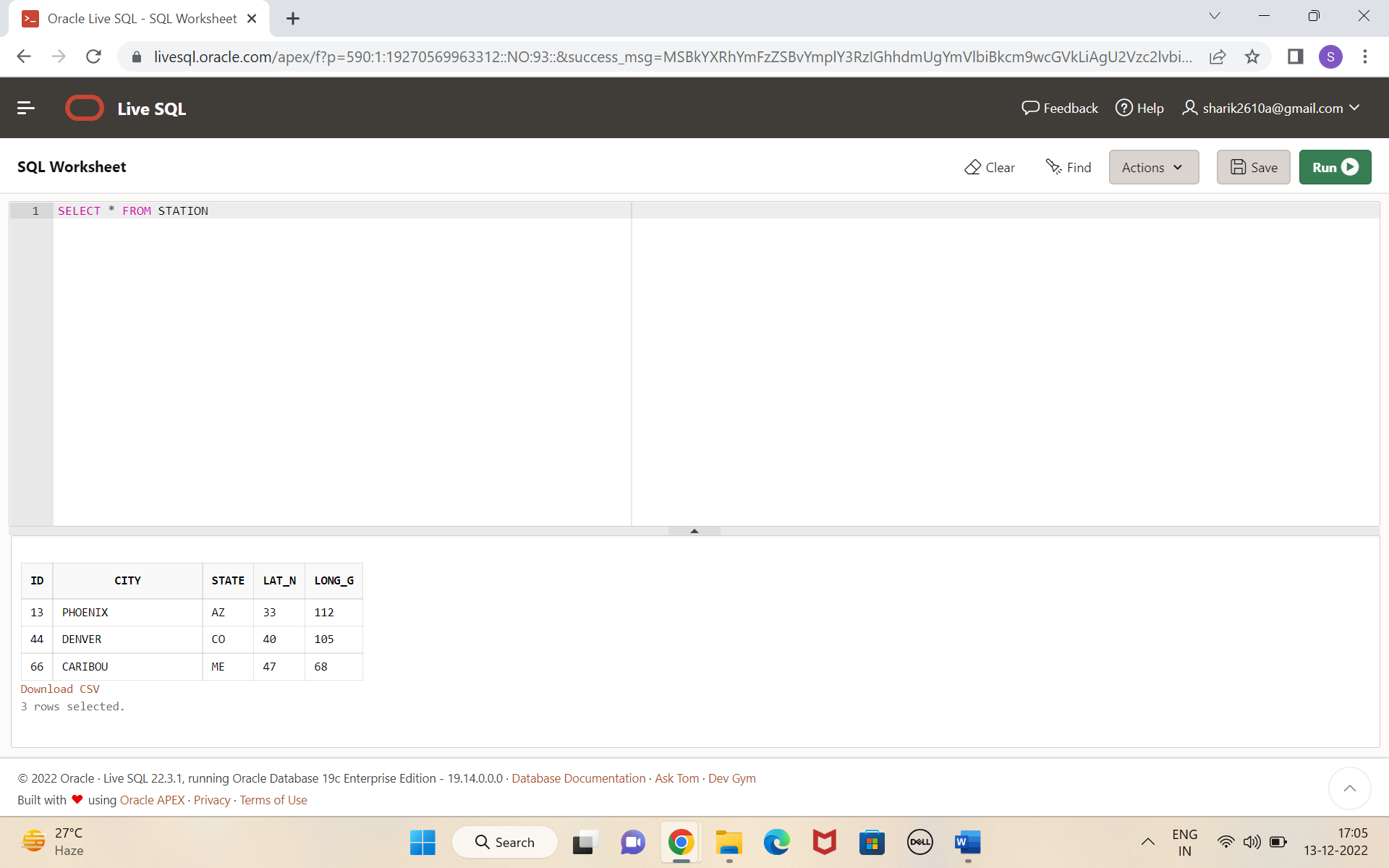
INSERT INTO Station VALUES (44,'DENVER','CO',40,105);

INSERT INTO Station VALUES (66,'CARIBOU','ME',47,68);



3. Execute a query to look at table STATION in undefined order

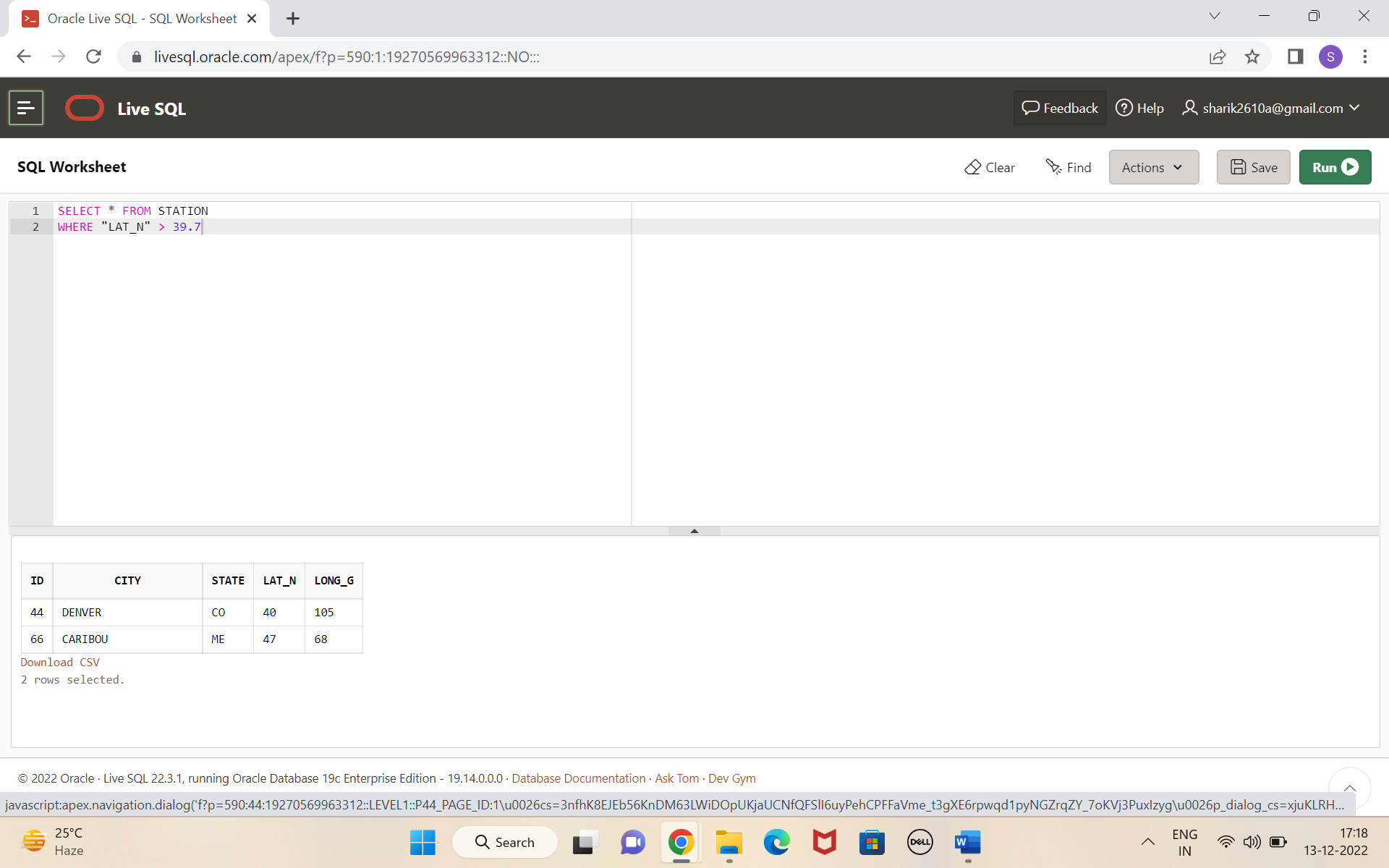
ANS) SELECT \* FROM STATION



4. Execute a query to select Northern stations (Northern latitude > 39.7).

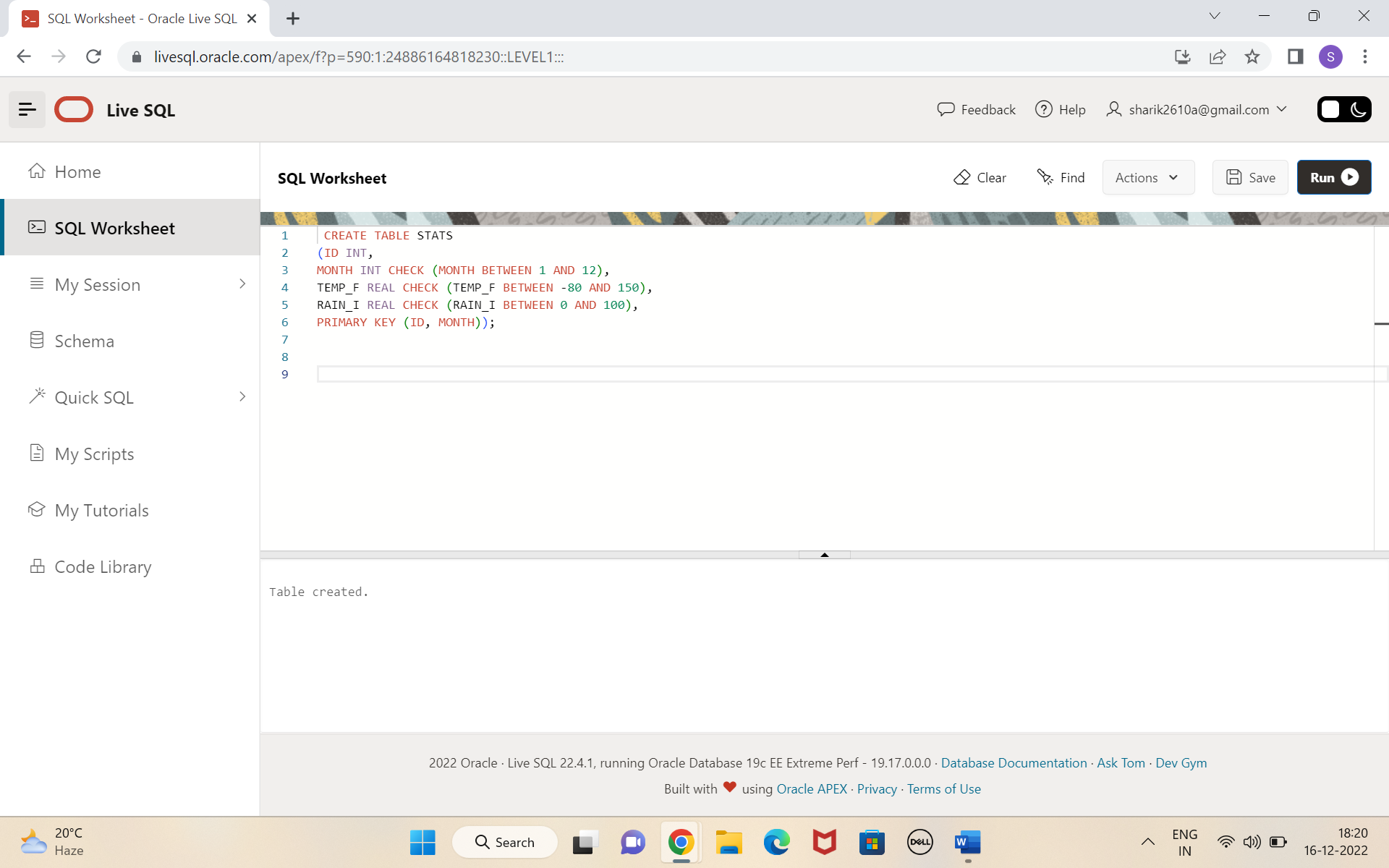
ANS) SELECT \* FROM STATION

WHERE "LAT\_N" > 39.7



5. Create another table ,‘STATS’, to store normalized temperature and precipitation data

ANS) CREATE TABLE STATS  
(ID INT,  
MONTH INT CHECK (MONTH BETWEEN 1 AND 12),  
TEMP\_F REAL CHECK (TEMP\_F BETWEEN -80 AND 150),  
RAIN\_I REAL CHECK (RAIN\_I BETWEEN 0 AND 100),  
PRIMARY KEY (ID, MONTH));



6. Populate the table STATS with some statistics for January and July:

ANS) INSERT INTO STATS (ID, MONTH, TEMP\_F, RAIN\_I) VALUES

(13,1,57.4,.31);

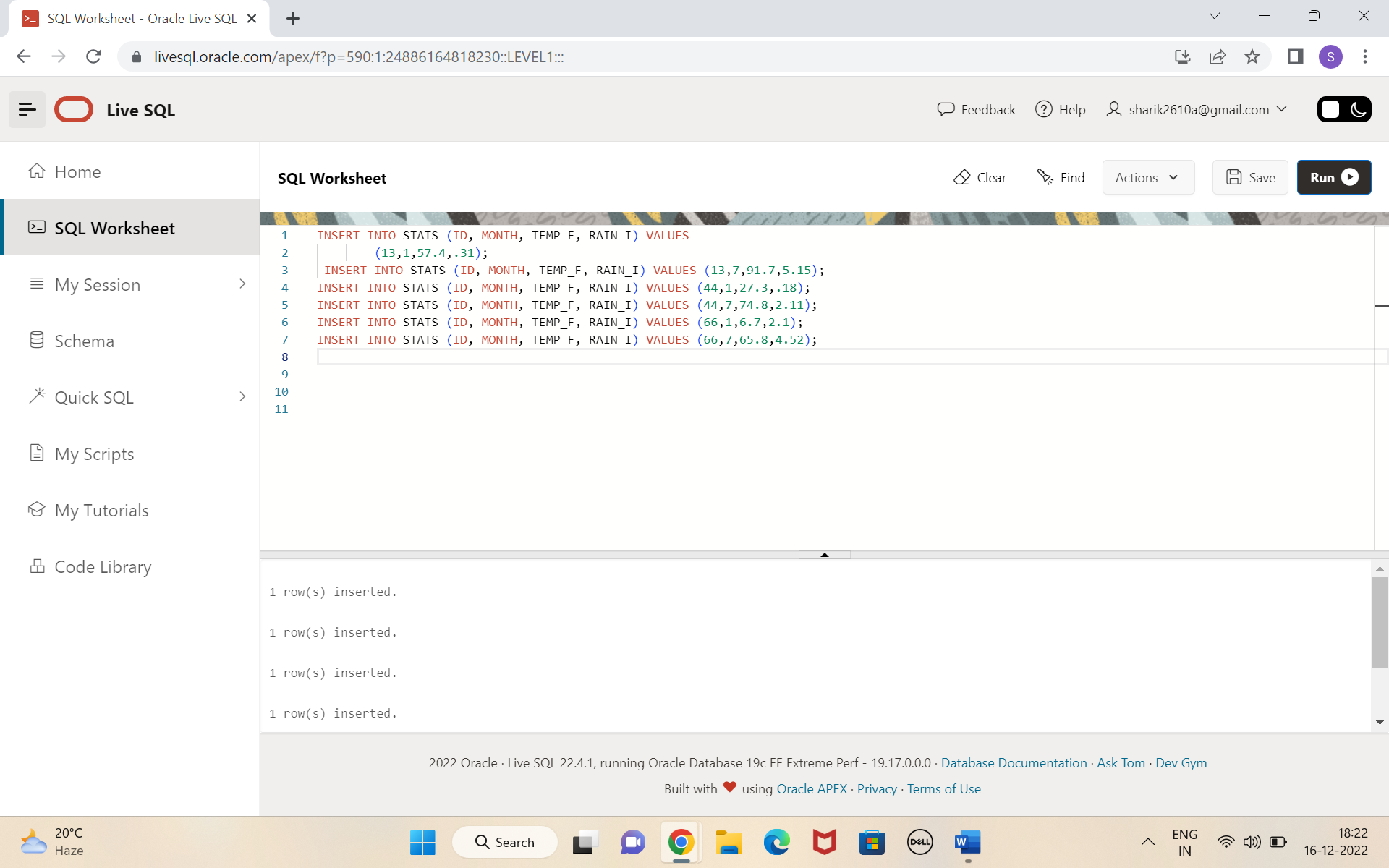
INSERT INTO STATS (ID, MONTH, TEMP\_F, RAIN\_I) VALUES (13,7,91.7,5.15);

INSERT INTO STATS (ID, MONTH, TEMP\_F, RAIN\_I) VALUES (44,1,27.3,.18);

INSERT INTO STATS (ID, MONTH, TEMP\_F, RAIN\_I) VALUES (44,7,74.8,2.11);

INSERT INTO STATS (ID, MONTH, TEMP\_F, RAIN\_I) VALUES (66,1,6.7,2.1);

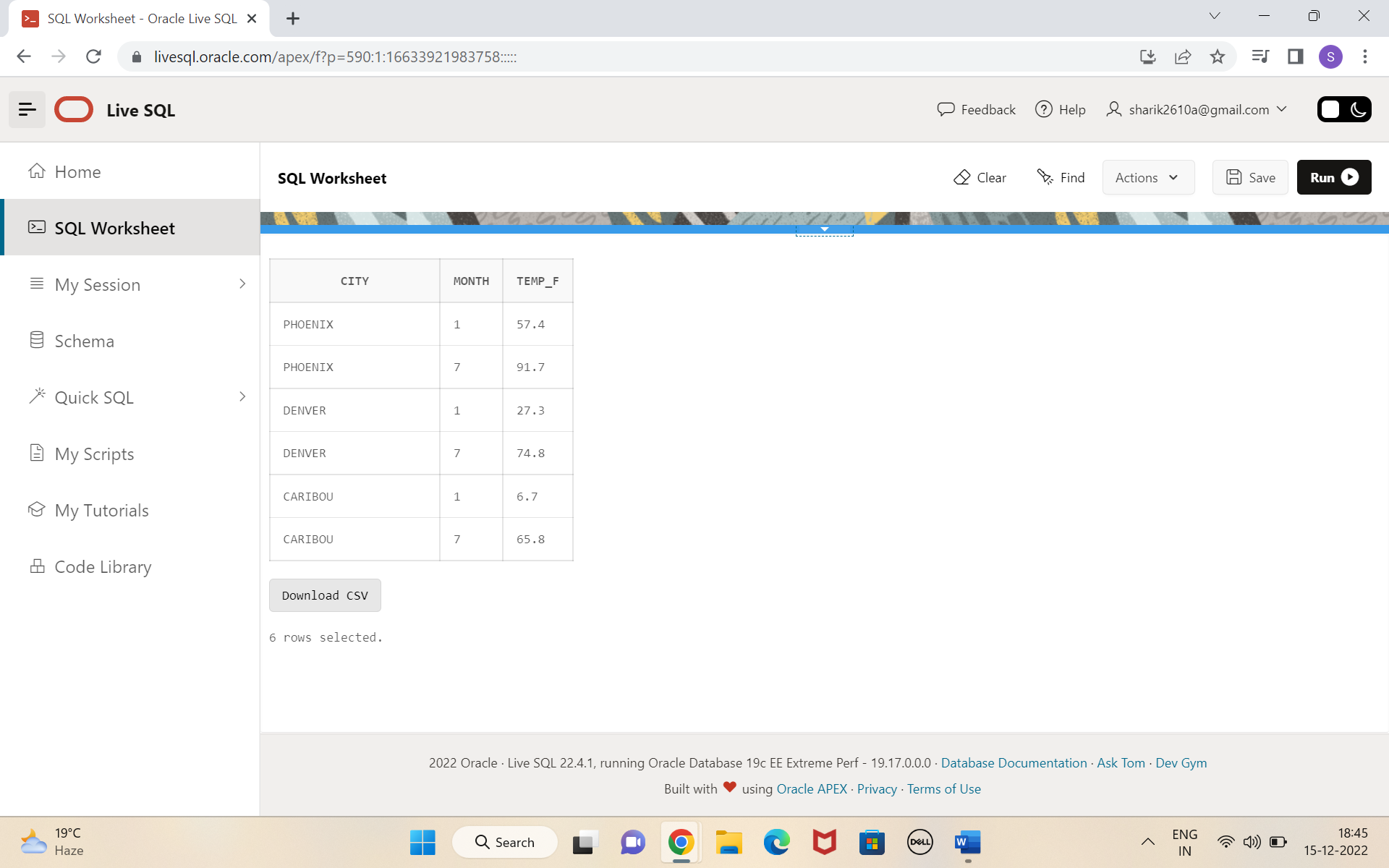
INSERT INTO STATS (ID, MONTH, TEMP\_F, RAIN\_I) VALUES (66,7,65.8,4.52);



7) Execute a query to display temperature stats (from STATS table) for each city (from Station table).

ANS) SELECT CITY, MONTH, TEMP\_F FROM (SELECT \* FROM Station D

INNER JOIN STATS I ON D. ID=I.ID);



8. Execute a query to look at the table STATS, ordered by month and greatest

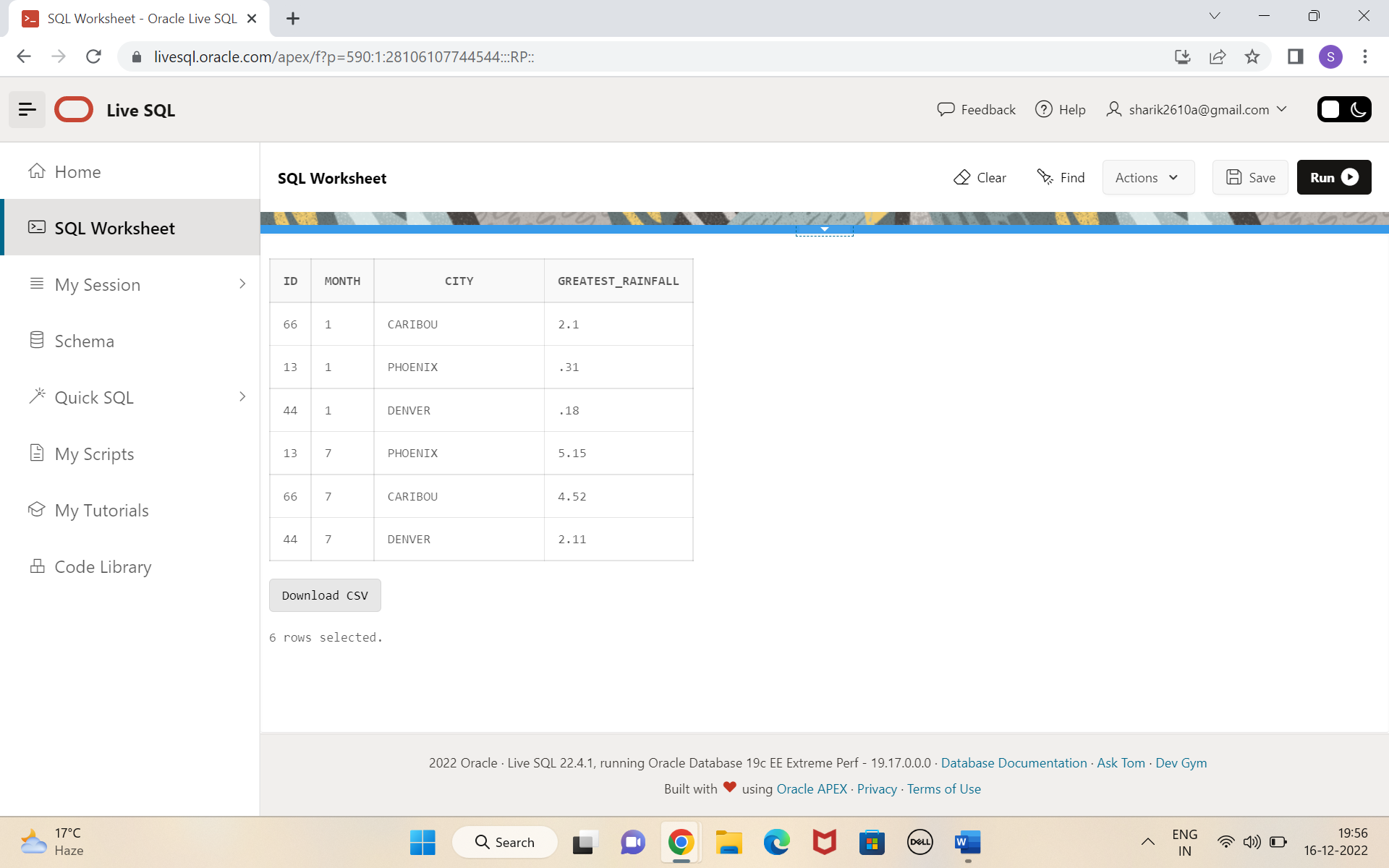
rainfall, with columns rearranged. It should also show the corresponding cities.

ANS) SELECT I.ID, I.MONTH, D.CITY, (I. RAIN\_I) as greatest\_rainfall FROM

STATS I

INNER JOIN Station D ON I.ID = D.ID

order by MONTH, RAIN\_I DESC;



9. Execute a query to look at Temperatures for July from table STATS , lowest

temperatures first, picking up city name and latitude.

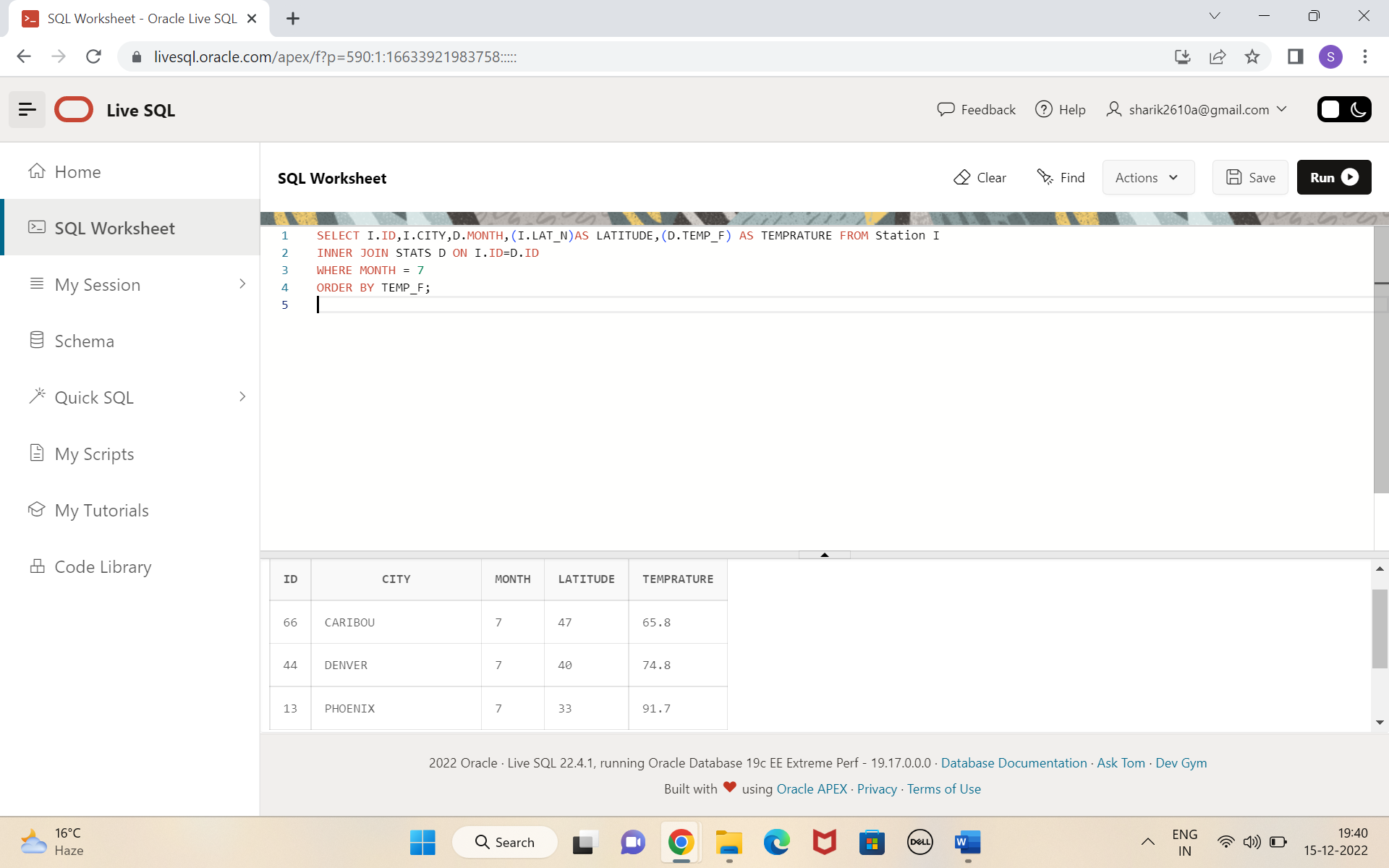
ANS) SELECT I.ID, I. CITY, D. MONTH, (I.LAT\_N) AS LATITUDE, (D. TEMP\_F) AS

TEMPRATURE FROM Station I

INNER JOIN STATS D ON I.ID= D.ID

WHERE MONTH = 7

ORDER BY TEMP\_F;



10. Execute a query to show MAX and MIN temperatures as well as average rainfall for each city.

ANS ) select count(\*),city ,max( temp\_ f)as max \_ temperature ,

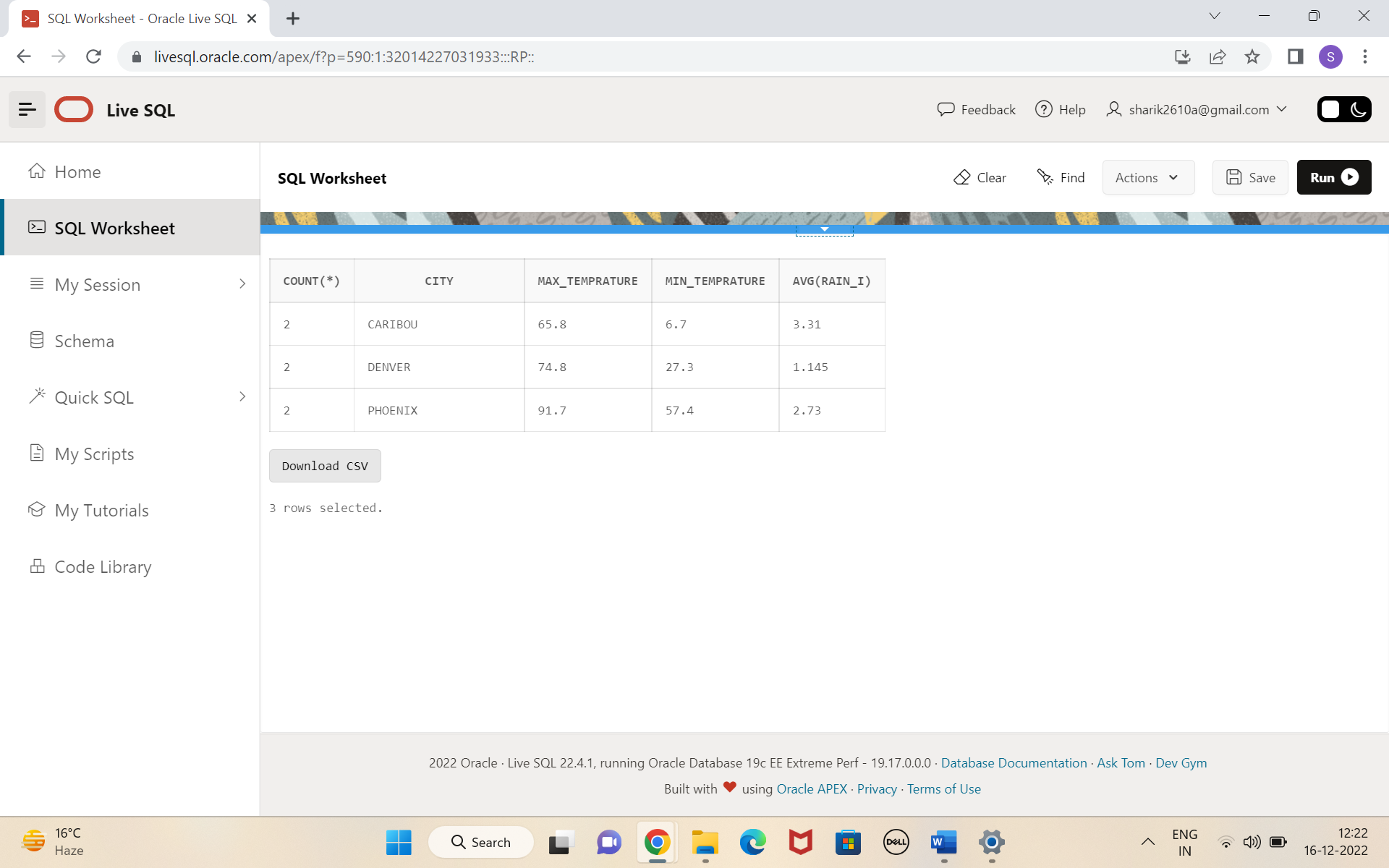
Min ( temp \_ f) as min \_ temperature ,

Avg (rain \_I)

from stats d

inner join Station i on d .id=i.id

group by city;



11.Execute a query to display each city’s monthly temperature in Celcius and rain fall

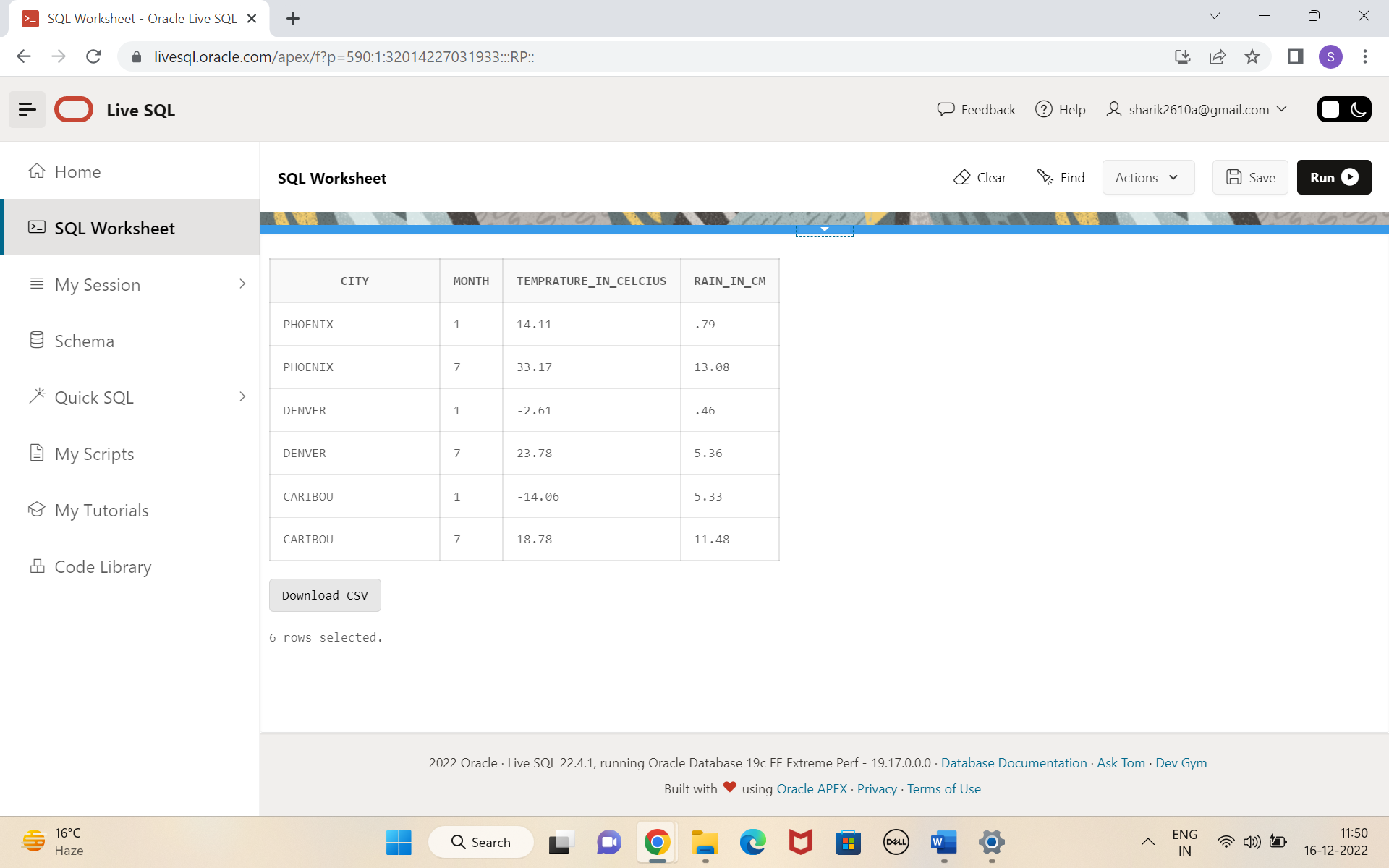
in Centimeter .

ANS) SELECT CITY, MONTH, ROUND((TEMP\_F-32) \* (5/9),2) AS

Temprature\_In\_Celcius, ROUND((RAIN\_I\*2.54),2) as Rain\_in\_cm

From (select \* from STATS D

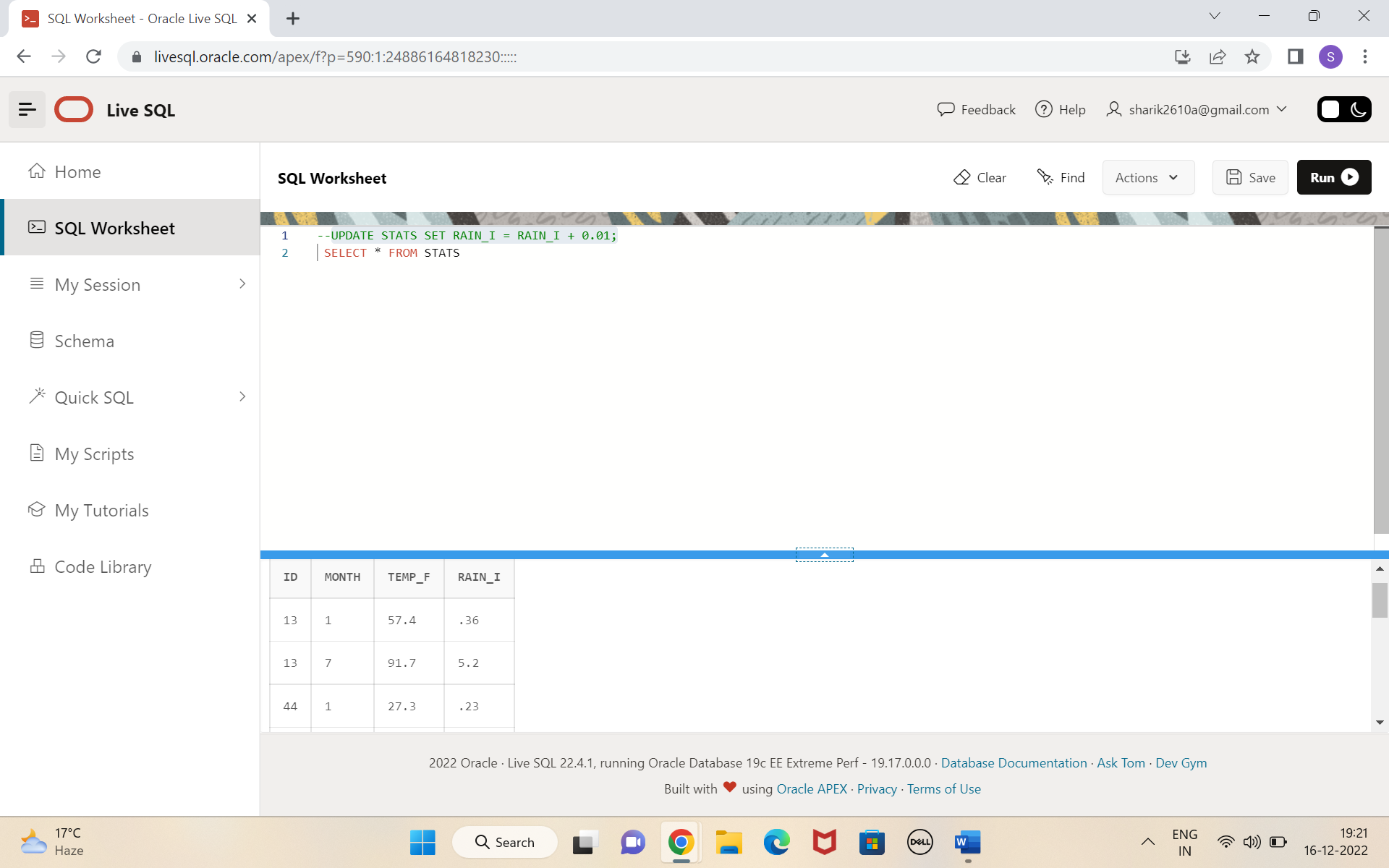
INNER JOIN Station I ON D.ID =I.ID);



12. Update all rows of table STATS to compensate for faulty rain gauges known to

read 0.01 inches low.

ANS) UPDATE STATS SET RAIN\_I = RAIN\_I + 0.01;



13. Update Denver's July temperature reading as 74.9.

ANS) **UPDATE STATS SET TEMP\_F = 74.9**

**WHERE ID = 44**

**AND MONTH = 7;**

