Assignment-SQL[Major]

1.Create a table "Station" to store information about weather observation stations:

SYNTAX:

CREATE TABLE STATIONS

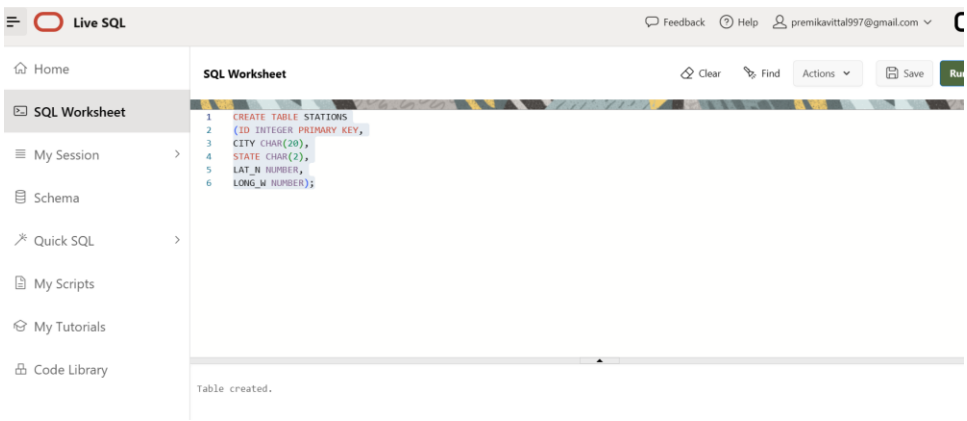
(ID INTEGER PRIMARY KEY,

CITY CHAR (20),

STATE CHAR (2),

LAT_N NUMBER,

LONG_W NUMBER);



2. Insert the following records into the table:

ID	CITY	STATE	LAT_N	LONG_W
13	PHOENIX	AZ	33	112
44	DENVER	CO	40	105
66	CARIBOU	ME	47	68

SYNTAX:

INSERT INTO STATION VALUES (13, 'Phoenix', 'AZ', 33, 112);

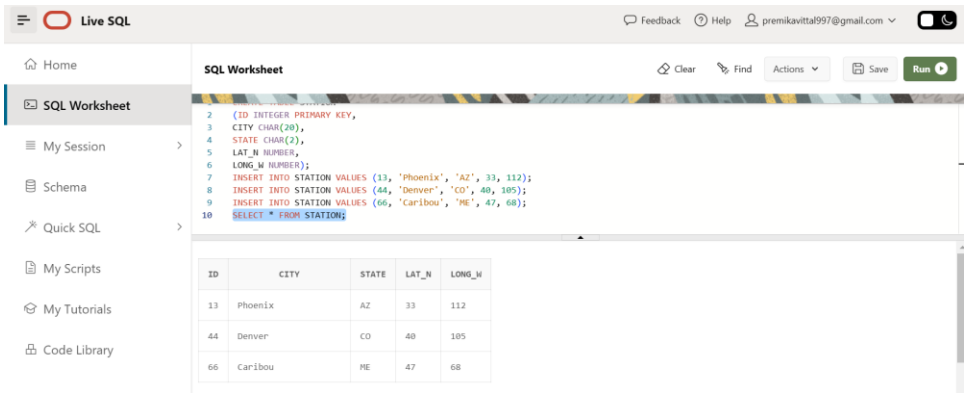
INSERT INTO STATION VALUES (44, 'Denver', 'CO', 40, 105);

INSERT INTO STATION VALUES (66, 'Caribou', 'ME', 47, 68);

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

SYNTAX:

SELECT * FROM STATION;

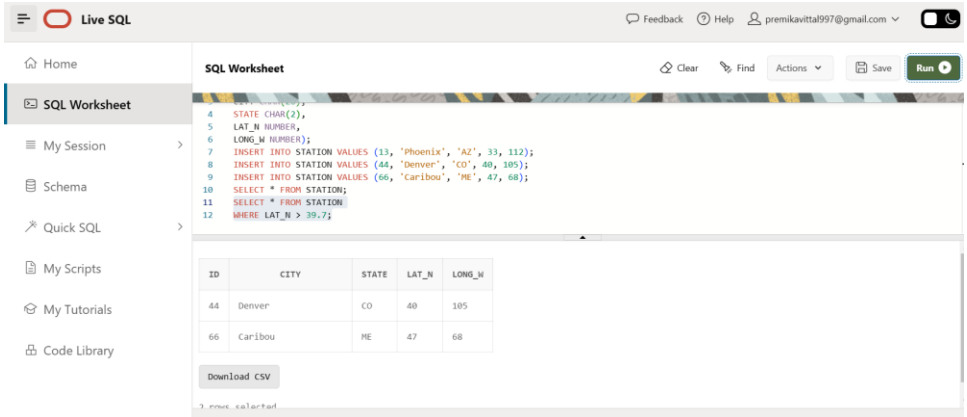


4.Execute a query to select Northern stations (Northern latitude>39.7)

STNTAX:

SELECT * FROM STATION

WHERE LAT_N > 39.7;



5.Create another table, 'STATS', to store normalized temperature and precipitation data:

Column	Data type	Remark
ID	Number	must match some STATION table ID(so name & location will be known).
MONTH	Number	Range between 1 and 12
TEMP_F	Number	in Fahrenheit degrees,Range between -80 and 150
RAIN_I	Number	in inches, Range between 0 and 100

There will be no Duplicate ID and MONTH combination.

SYNTAX:

CREATE TABLE STATS

(ID INTEGER REFERENCES STATION(ID),

MONTH INTEGER CHECK (MONTH BETWEEN 1 AND 12),

TEMP_F NUMBER CHECK (TEMP_F BETWEEN -80 AND 150),

RAIN_I NUMBER CHECK (RAIN_I BETWEEN 0 AND 100),

PRIMARY KEY (ID, MONTH));

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

ID	MONTH	TEMP_F	RAIN_I
13	1	57.4	.31
13	7	91.7	5.15
44	1	27.3	.18
44	7	74.8	2.11
66	1	6.7	2.1
66	7	65.8	4.52

SYNTAX:

INSERT INTO STATS VALUES (13, 1, 57.4, 0.31);

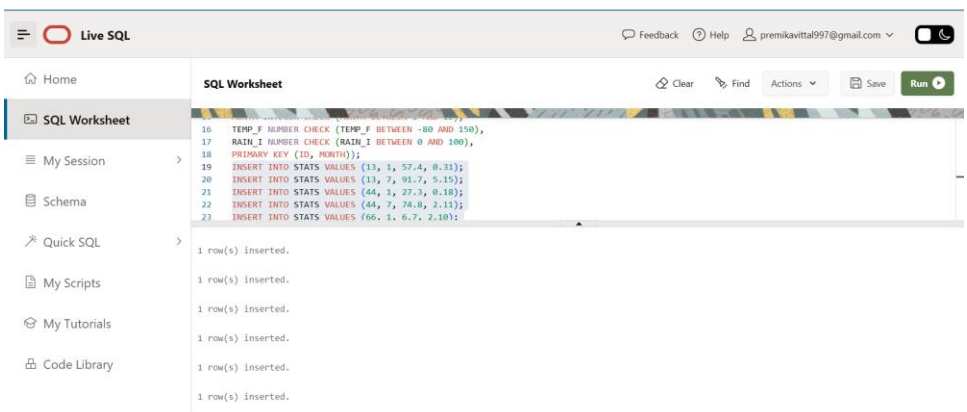
INSERT INTO STATS VALUES (13, 7, 91.7, 5.15);

INSERT INTO STATS VALUES (44, 1, 27.3, 0.18);

INSERT INTO STATS VALUES (44, 7, 74.8, 2.11);

INSERT INTO STATS VALUES (66, 1, 6.7, 2.10);

INSERT INTO STATS VALUES (66, 7, 65.8, 4.52);

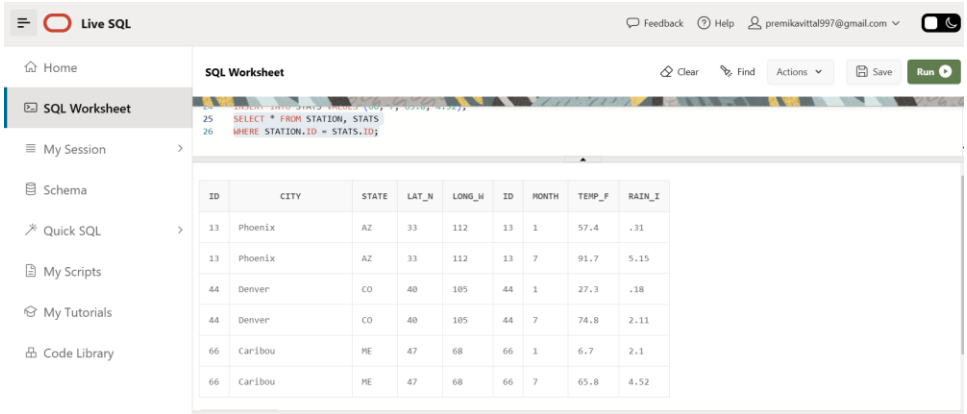


7.Execute a query to display temperature stats (fromSTATS table) for each city (from Station table).

SYNTAX:

SELECT * FROM STATION, STATS

WHERE STATION.ID = STATS.ID;



ID	CITY	STATE	LAT_N	LONG_W	ID	MONTH	TEMP_F	RAIN_I
13	Phoenix	AZ	33	112	13	1	57.4	.31
13	Phoenix	AZ	33	112	13	7	91.7	5.15
44	Denver	CO	40	105	44	1	27.3	.18
44	Denver	CO	40	105	44	7	74.8	2.11
66	Caribou	ME	47	68	66	1	6.7	2.1
66	Caribou	ME	47	68	66	7	65.8	4.52

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.

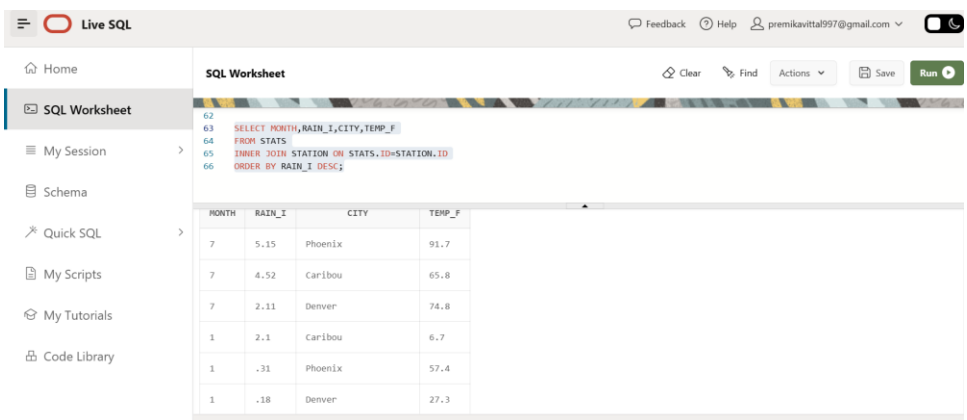
SYNTAX:

SELECT MONTH,RAIN_I,CITY,TEMP_F

FROM STATS

INNER JOIN STATION ON STATS.ID=STATION.ID

ORDER BY RAIN_I DESC;

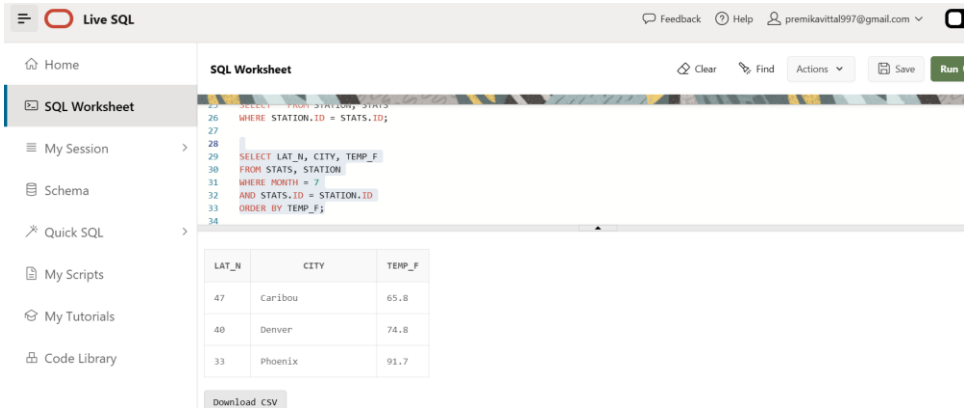


MONTH	RAIN_I	CITY	TEMP_F
7	5.15	Phoenix	91.7
7	4.52	Caribou	65.8
7	2.11	Denver	74.8
1	2.1	Caribou	6.7
1	.31	Phoenix	57.4
1	.18	Denver	27.3

9. Execute a query to look at temperatures for July from table STATS, lowest temperatures first , picking up city name and latitude

SYNTAX:

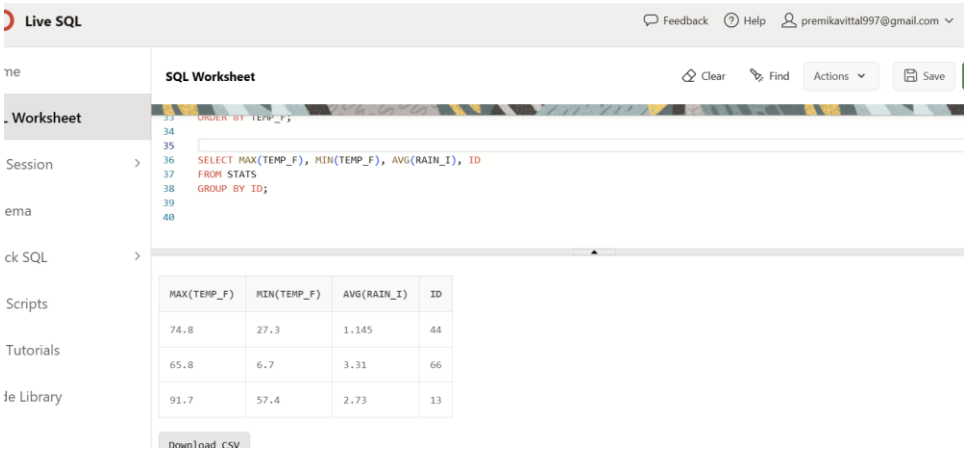
SELECT LAT_N, CITY, TEMP_F
FROM STATS, STATION
WHERE MONTH = 7
AND STATS.ID = STATION.ID
ORDER BY TEMP_F;



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

SYNTAX:

SELECT MAX(TEMP_F), MIN(TEMP_F), AVG(RAIN_I), ID
FROM STATS
GROUP BY ID;



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

SYNTAX:

CREATE VIEW METRIC_STATSS (CITY, MONTH, TEMP_C, RAIN_C) AS

SELECT CITY,

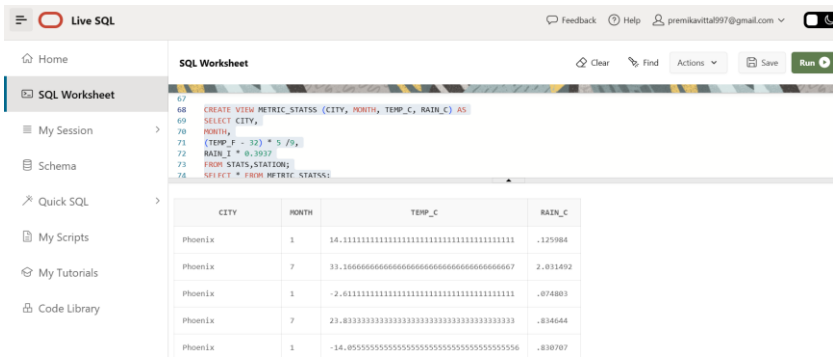
MONTH,

(TEMP_F - 32) * 5 /9,

RAIN_I * 0.3937

FROM STATS,STATION;

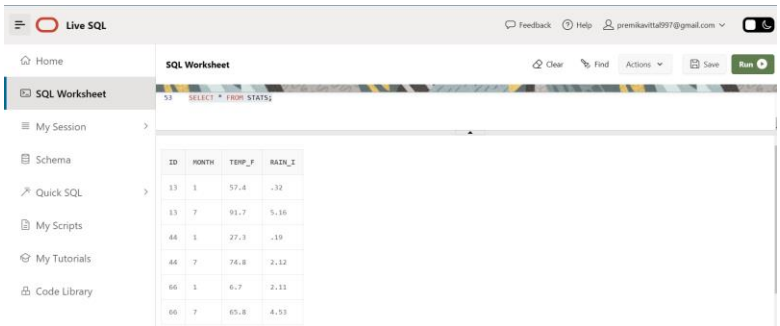
SELECT * FROM METRIC_STATSS;



12.Update all rows of table STATS to compensate for faulty rain gauges known to read 0.01 inches low.

SYNTAX:

UPDATE STATS SET RAIN_I = RAIN_I + 0.01;



13.Update Denver's July temperature reading as 74.9

SYNTAX:

UPDATE STATS SET TEMP_F = 74.9
WHERE ID = 44
AND MONTH = 7;