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

CREATE TABLE weather.STATION

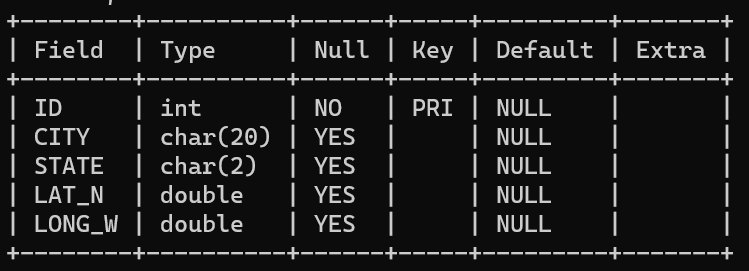
(ID INTEGER PRIMARY KEY,

CITY CHAR(20),

STATE CHAR(2),

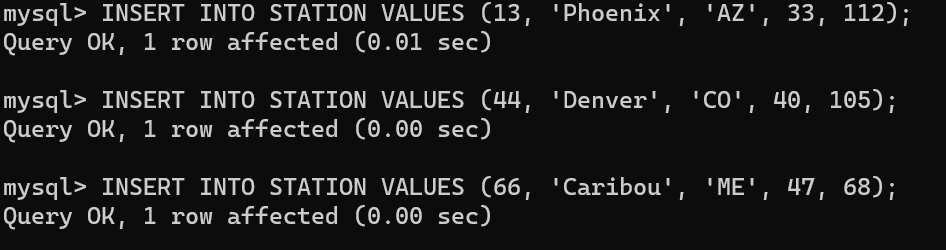
LAT\_N REAL,

LONG\_W REAL);



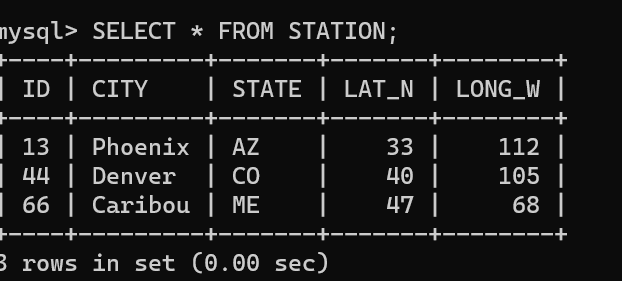
1. Insert the following records into the table:

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);



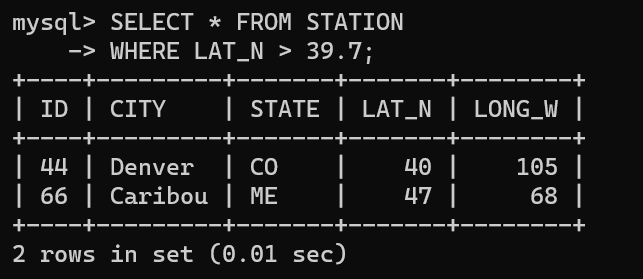
1. Execute a query to look at table STATION in undefined order.

SELECT \* FROM STATION;



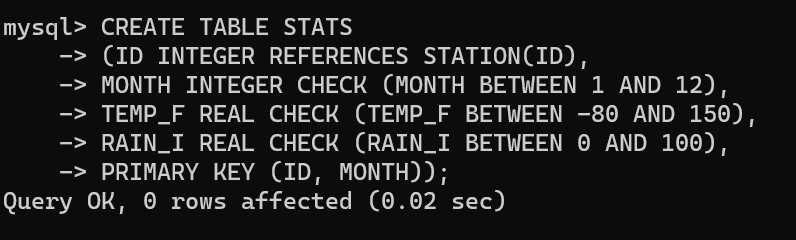
1. Execute a query to select Northern stations (Northern latitude > 39.7).

SELECT \* FROM STATION  
WHERE LAT\_N > 39.7;



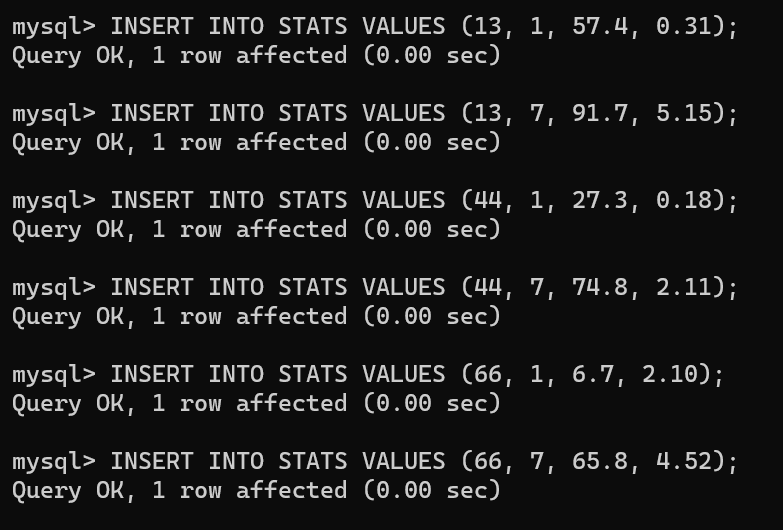
1. Create another table, ‘STATS’, to store normalized temperature and precipitation data:

CREATE TABLE STATS  
(ID INTEGER REFERENCES STATION(ID),  
MONTH INTEGER 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));



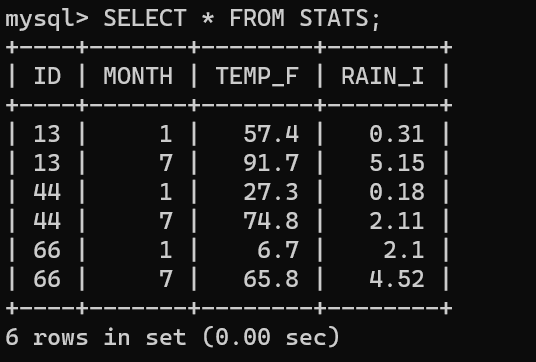
1. Populate the table STATS with some statistics for January and July:

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);



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

SELECT \* FROM STATS;



1. 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.

SELECT s.CITY, t.MONTH, t.RAIN\_I AS greatest\_rainfall

FROM Station s

JOIN STATS t ON s.ID = t.ID

ORDER BY t.MONTH, greatest\_rainfall DESC;



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

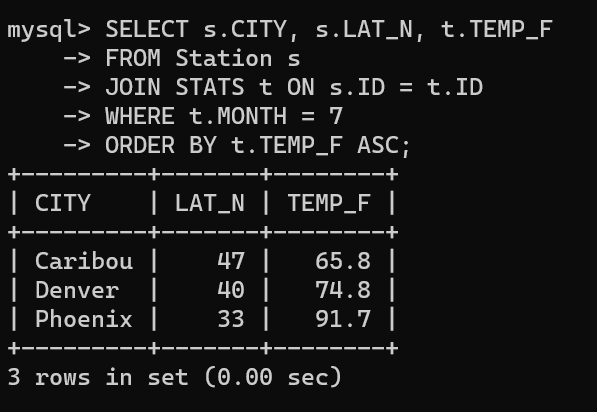
SELECT s.CITY, s.LAT\_N, t.TEMP\_F

FROM Station s

JOIN STATS t ON s.ID = t.ID

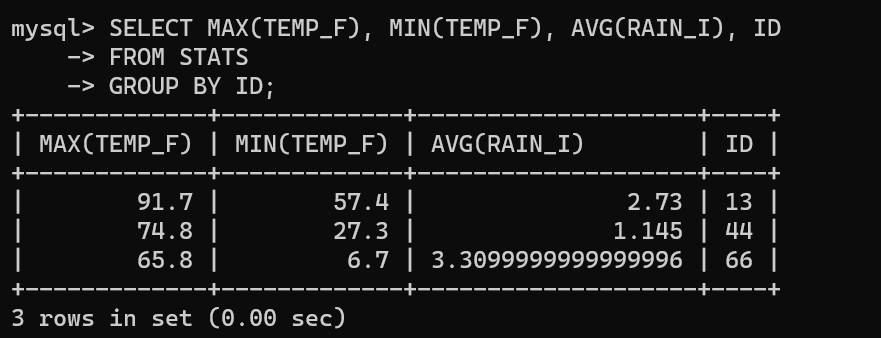
WHERE t.MONTH = 7

ORDER BY t.TEMP\_F ASC;



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

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 rainfall in Centimeter.

SELECT s.CITY,

t.MONTH,

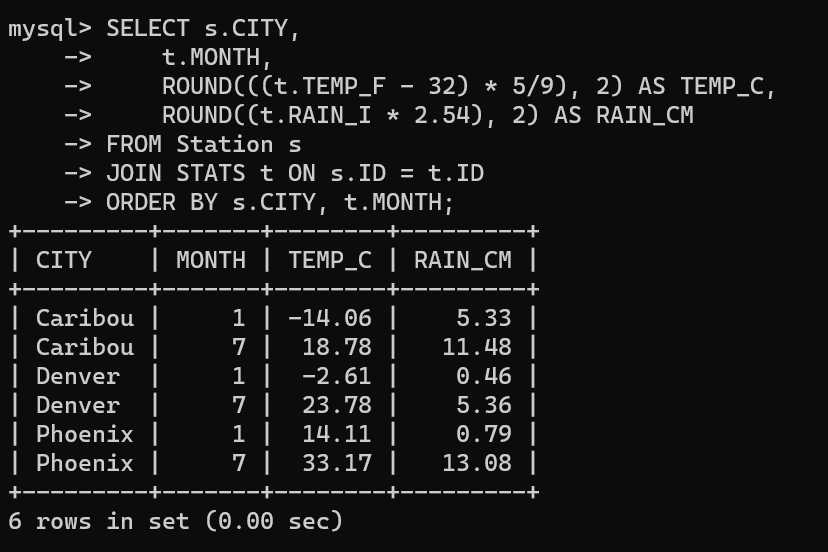
ROUND(((t.TEMP\_F - 32) \* 5/9), 2) AS TEMP\_C,

ROUND((t.RAIN\_I \* 2.54), 2) AS RAIN\_CM

FROM Station s

JOIN STATS t ON s.ID = t.ID

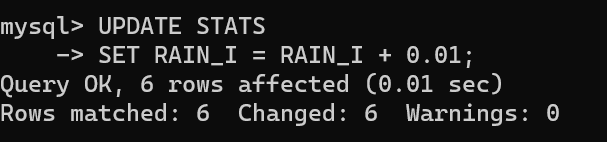
ORDER BY s.CITY, t.MONTH;



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

UPDATE STATS

SET RAIN\_I = RAIN\_I + 0.01;



13. Update Denver's July temperature reading as 74.9

UPDATE STATS

SET TEMP\_F = 74.9

WHERE ID = 44 AND MONTH = 7;

