Khoi Duong

Prof. VB

CS457L

9/17/2022

1. Add 7 more rows to make it 10 in total for weather_station. You can have two months of stats for each city. So, 20 rows for weather_stats

Source code:

```
INSERT INTO weather_station VALUES (37, 'San Jose', 'CA', 48, 164);
INSERT INTO weather_station VALUES (49, 'Chicago', 'IL', 51, 152);
INSERT INTO weather_station VALUES (56, 'Houston', 'TX', 25, 147);
INSERT INTO weather_station VALUES (12, 'Columbus', 'OH', 29, 135);
INSERT INTO weather_station VALUES (10, 'Nashville', 'TN', 14, 88);
INSERT INTO weather_station VALUES (87, 'Boston', 'MA', 27, 90);
INSERT INTO weather_station VALUES (94, 'Baltimore', 'MD', 99, 65);
```

Explanation: We insert 7 values/rows into weather_station table.

```
INSERT INTO weather_stats VALUES (37, 1, 34.8, 2.81);
INSERT INTO weather_stats VALUES (37, 7, 25.6, 1.15);
INSERT INTO weather_stats VALUES (49, 1, 19.6, 3.45);
INSERT INTO weather_stats VALUES (49, 7, 47.5, 0.48);
INSERT INTO weather_stats VALUES (56, 1, 64.3, 4.15);
INSERT INTO weather_stats VALUES (56, 7, 24.7, 3.45);
```

INSERT INTO weather_stats VALUES (12, 1, 78.9, 1.26);

INSERT INTO weather_stats VALUES (12, 7, 86.7, 0.59);

INSERT INTO weather_stats VALUES (10, 1, 13.3, 1.89);

INSERT INTO weather_stats VALUES (10, 7, 38.7, 2.42);

INSERT INTO weather_stats VALUES (87, 1, 98.6, 1.67);

INSERT INTO weather_stats VALUES (87, 7, 14.1, 0.69);

INSERT INTO weather_stats VALUES (94, 1, 77.2, 3.54);

INSERT INTO weather_stats VALUES (94, 7, 18.3, 1.24);

Explanation: We insert 14 values/rows into weather_stats table.

Run program & result:

weather_station

```
MariaDB [19610dm]> INSERT INTO weather_station VALUES (37, 'San Jose', 'CA', 48, 164);
INSERT INTO weather_station VALUES (87, 'Boston', 'MA', 27, 90);
INSERT INTO weather_station VALUES (94, 'Baltimore', 'MD', 99, 65);

MariaDB [19610dm]> INSERT INTO weather_station VALUES (49, 'Chicago', 'IL', 51, 152);


MariaDB [19610dm]> INSERT INTO weather_station VALUES (56, 'Houston', 'TX', 25, 147);


MariaDB [19610dm]> INSERT INTO weather_station VALUES (12, 'Columbus', 'OH', 29, 135);


MariaDB [19610dm]> INSERT INTO weather_station VALUES (10, 'Nashville', 'TN', 14, 88);


MariaDB [19610dm]> INSERT INTO weather_station VALUES (87, 'Boston', 'MA', 27, 90);


MariaDB [19610dm]> INSERT INTO weather_station VALUES (94, 'Baltimore', 'MD', 99, 65);


MariaDB [19610dm]> select * from weather_station
    -> ;
+----+-----------+-------+-------+--------+
| ID | CITY      | STATE | LAT_N | LONG_W |
+----+-----------+-------+-------+--------+
| 10 | Nashville | TN    |    14 |     88 |
| 12 | Columbus  | OH    |    29 |    135 |
| 13 | Phoenix   | AZ    |    33 |    112 |
| 37 | San Jose  | CA    |    48 |    164 |
| 44 | Denver    | CO    |    40 |    105 |
| 49 | Chicago   | IL    |    51 |    152 |
| 56 | Houston   | TX    |    25 |    147 |
| 66 | Caribou   | ME    |    47 |     68 |
| 87 | Boston    | MA    |    27 |     90 |
| 94 | Baltimore | MD    |    99 |     65 |
+----+-----------+-------+-------+--------+
```

weather_stats

```
MariaDB [19610dm]> INSERT INTO weather_stats VALUES (12, 7, 86.7, 0.59);

MariaDB [19610dm]> INSERT INTO weather_stats VALUES (10, 1, 13.3, 1.89);

MariaDB [19610dm]> INSERT INTO weather_stats VALUES (10, 7, 38.7, 2.42);

MariaDB [19610dm]> INSERT INTO weather_stats VALUES (87, 1, 98.6, 1.67);

MariaDB [19610dm]> INSERT INTO weather_stats VALUES (87, 7, 14.1, 0.69);

MariaDB [19610dm]> INSERT INTO weather_stats VALUES (94, 1, 77.2, 3.54);

MariaDB [19610dm]> INSERT INTO weather_stats VALUES (94, 7, 18.3, 1.24);

MariaDB [19610dm]> select * from weather_stats;
+----+-------+--------+--------+
| ID | MONTH | TEMP_F | RAIN_I |
+----+-------+--------+--------+
| 10 |     1 |   13.3 |   1.89 |
| 10 |     7 |   38.7 |   2.42 |
| 12 |     1 |   78.9 |   1.26 |
| 12 |     7 |   86.7 |   0.59 |
| 13 |     1 |   57.4 |   0.31 |
| 13 |     7 |   91.7 |   5.15 |
| 37 |     1 |   34.8 |   2.81 |
| 37 |     7 |   25.6 |   1.15 |
| 44 |     1 |   27.3 |   0.18 |
| 44 |     7 |   74.8 |   2.11 |
| 49 |     1 |   19.6 |   3.45 |
| 49 |     7 |   47.5 |   0.48 |
| 56 |     1 |   64.3 |   4.15 |
| 56 |     7 |   24.7 |   3.45 |
| 66 |     1 |    6.7 |    2.1 |
| 66 |     7 |   65.8 |   4.52 |
| 87 |     1 |   98.6 |   1.67 |
| 87 |     7 |   14.1 |   0.69 |
| 94 |     1 |   77.2 |   3.54 |
| 94 |     7 |   18.3 |   1.24 |
+----+-------+--------+--------+
```

2. Use a shortcut (or a command like say distinct, etc) to remove the extra id column when selecting the columns from two tables.

In order to remove the extra id column without losing any of the data in the two data tables, first, we have to process ***weather_stats***.

Source code:

```
create table weather_stats1
select ID, GROUP_CONCAT(MONTH SEPARATOR ', ') AS MONTH,
GROUP_CONCAT(TEMP_F SEPARATOR ', ') AS TEMP_F,
 GROUP_CONCAT(RAIN_I SEPARATOR ', ') AS RAIN_I
from weather_stats group by ID;
```

Explanation: the code will concatenate the MONTH value (1, 7), the TEMP_F and RAIN_I value from table weather_stats (since we don't want to lose any data from the table) and save them as a new table called weather_stats1.

Result:

```
MariaDB [19610dm]> select * from weather_stats1;
+----+-------+------------+------------+
| ID | MONTH | TEMP_F     | RAIN_I     |
+----+-------+------------+------------+
| 10 | 1, 7  | 13.3, 38.7 | 1.89, 2.42 |
| 12 | 1, 7  | 78.9, 86.7 | 1.26, 0.59 |
| 13 | 1, 7  | 57.4, 91.7 | 0.31, 5.15 |
| 37 | 1, 7  | 34.8, 25.6 | 2.81, 1.15 |
| 44 | 1, 7  | 27.3, 74.8 | 0.18, 2.11 |
| 49 | 1, 7  | 19.6, 47.5 | 3.45, 0.48 |
| 56 | 1, 7  | 64.3, 24.7 | 4.15, 3.45 |
| 66 | 1, 7  | 6.7, 65.8  | 2.1, 4.52  |
| 87 | 1, 7  | 98.6, 14.1 | 1.67, 0.69 |
| 94 | 1, 7  | 77.2, 18.3 | 3.54, 1.24 |
+----+-------+------------+------------+
```

Next, we have to join the two tables together and create a table with all of the data and no extra column.

Source code:

```
select * from weather_station
inner join weather_stats1
on weather_station.ID = weather_stats1.ID group by weather_station.ID;
```

Explanation: the code will select all data from weather_station and join with the previous result from the above execution by the same ID values and group by the ID values.

Result:

ID	CITY	STATE	LAT_N	LONG_W	ID	MONTH	TEMP_F	RAIN_I
10	Nashville	TN	14	88	10	1, 7	13.3, 38.7	1.89, 2.42
12	Columbus	OH	29	135	12	1, 7	78.9, 86.7	1.26, 0.59
13	Phoenix	AZ	33	112	13	1, 7	57.4, 91.7	0.31, 5.15
37	San Jose	CA	48	164	37	1, 7	34.8, 25.6	2.81, 1.15
44	Denver	CO	40	105	44	1, 7	27.3, 74.8	0.18, 2.11
49	Chicago	IL	51	152	49	1, 7	19.6, 47.5	3.45, 0.48
56	Houston	TX	25	147	56	1, 7	64.3, 24.7	4.15, 3.45
66	Caribou	ME	47	68	66	1, 7	6.7, 65.8	2.1, 4.52
87	Boston	MA	27	90	87	1, 7	98.6, 14.1	1.67, 0.69
94	Baltimore	MD	99	65	94	1, 7	77.2, 18.3	3.54, 1.24

However, if we just want to select distinct values without caring about the data of the table, we can use the function GROUP BY:

```
select * from weather_station, weather_stats group by weather_station.ID;
```

Explanation: the code will select everything from table weather_station and weather_stats and group them by ID values, thus we will have only one row per ID value.

Result:

```
MariaDB [19610dm]>  select * from weather_station, weather_stats group by weather_station.ID;
+----+-----------+-------+-------+--------+----+-------+--------+--------+
| ID | CITY      | STATE | LAT_N | LONG_W | ID | MONTH | TEMP_F | RAIN_I |
+----+-----------+-------+-------+--------+----+-------+--------+--------+
| 10 | Nashville | TN    |    14 |     88 | 10 |     1 |   13.3 |   1.89 |
| 12 | Columbus  | OH    |    29 |    135 | 10 |     1 |   13.3 |   1.89 |
| 13 | Phoenix   | AZ    |    33 |    112 | 10 |     1 |   13.3 |   1.89 |
| 37 | San Jose  | CA    |    48 |    164 | 10 |     1 |   13.3 |   1.89 |
| 44 | Denver    | CO    |    40 |    105 | 10 |     1 |   13.3 |   1.89 |
| 49 | Chicago   | IL    |    51 |    152 | 10 |     1 |   13.3 |   1.89 |
| 56 | Houston   | TX    |    25 |    147 | 10 |     1 |   13.3 |   1.89 |
| 66 | Caribou   | ME    |    47 |     68 | 10 |     1 |   13.3 |   1.89 |
| 87 | Boston    | MA    |    27 |     90 | 10 |     1 |   13.3 |   1.89 |
| 94 | Baltimore | MD    |    99 |     65 | 10 |     1 |   13.3 |   1.89 |
+----+-----------+-------+-------+--------+----+-------+--------+--------+
10 rows in set (0.000 sec)
```

3. Use examples and explain the difference between group by and order by

GROUP BY:

```
MariaDB [19610dm]> select * from weather_stats group by ID;
+----+-------+--------+--------+
| ID | MONTH | TEMP_F | RAIN_I |
+----+-------+--------+--------+
| 10 |     1 |   13.3 |   1.89 |
| 12 |     1 |   78.9 |   1.26 |
| 13 |     1 |   57.4 |   0.31 |
| 37 |     1 |   34.8 |   2.81 |
| 44 |     1 |   27.3 |   0.18 |
| 49 |     1 |   19.6 |   3.45 |
| 56 |     1 |   64.3 |   4.15 |
| 66 |     1 |    6.7 |    2.1 |
| 87 |     1 |   98.6 |   1.67 |
| 94 |     1 |   77.2 |   3.54 |
+----+-------+--------+--------+
```

The GROUP BY expression will group the rows will the same ID value together, thus it will remove any extra rows, and data of different columns (e.g. we miss the data of MONTH 7 for weather_stats). Usually, the GROUP BY expression goes with the function COUNT() in order to count the number of times a data value exists in the data table.

ORDER BY:

```
MariaDB [19610dm]> select * from weather_stats ORDER by ID;
+----+-------+--------+--------+
| ID | MONTH | TEMP_F | RAIN_I |
+----+-------+--------+--------+
| 10 |     1 |   13.3 |   1.89 |
| 10 |     7 |   38.7 |   2.42 |
| 12 |     1 |   78.9 |   1.26 |
| 12 |     7 |   86.7 |   0.59 |
| 13 |     1 |   57.4 |   0.31 |
| 13 |     7 |   91.7 |   5.15 |
| 37 |     1 |   34.8 |   2.81 |
| 37 |     7 |   25.6 |   1.15 |
| 44 |     1 |   27.3 |   0.18 |
| 44 |     7 |   74.8 |   2.11 |
| 49 |     1 |   19.6 |   3.45 |
| 49 |     7 |   47.5 |   0.48 |
| 56 |     1 |   64.3 |   4.15 |
| 56 |     7 |   24.7 |   3.45 |
| 66 |     1 |    6.7 |    2.1 |
| 66 |     7 |   65.8 |   4.52 |
| 87 |     1 |   98.6 |   1.67 |
| 87 |     7 |   14.1 |   0.69 |
| 94 |     1 |   77.2 |   3.54 |
| 94 |     7 |   18.3 |   1.24 |
+----+-------+--------+--------+
```

ORDER BY expression only orders the result with the ID values, it does not remove any extra data from the table. For example, in this case, ORDER BY ID will list the data in the table in ascending order of ID values. We can try the same thing with ORDER BY MONTH:

```
MariaDB [19610dm]> select * from weather_stats order by MONTH;
+----+-------+--------+--------+
| ID | MONTH | TEMP_F | RAIN_I |
+----+-------+--------+--------+
| 10 |     1 |   13.3 |   1.89 |
| 94 |     1 |   77.2 |   3.54 |
| 87 |     1 |   98.6 |   1.67 |
| 66 |     1 |    6.7 |    2.1 |
| 56 |     1 |   64.3 |   4.15 |
| 49 |     1 |   19.6 |   3.45 |
| 44 |     1 |   27.3 |   0.18 |
| 37 |     1 |   34.8 |   2.81 |
| 12 |     1 |   78.9 |   1.26 |
| 13 |     1 |   57.4 |   0.31 |
| 37 |     7 |   25.6 |   1.15 |
| 10 |     7 |   38.7 |   2.42 |
| 87 |     7 |   14.1 |   0.69 |
| 66 |     7 |   65.8 |   4.52 |
| 12 |     7 |   86.7 |   0.59 |
| 56 |     7 |   24.7 |   3.45 |
| 49 |     7 |   47.5 |   0.48 |
| 13 |     7 |   91.7 |   5.15 |
| 44 |     7 |   74.8 |   2.11 |
| 94 |     7 |   18.3 |   1.24 |
+----+-------+--------+--------+
20 rows in set (0.000 sec)
```

We can see that the MONTH value will be ordered in ascending order.

4. Display and Show all the columns along with the avg(temp_c) in the metric_stats view.

Source code:

```
select ID,
GROUP_CONCAT(ROUND(TEMP_C,2) SEPARATOR ', ') AS TEMP_C,
GROUP_CONCAT(ROUND(RAIN_C,2) SEPARATOR ', ') AS RAIN_C,
ROUND(AVG(TEMP_C),2) FROM metric_stats GROUP BY ID;
```

Explanation: The code will select the ID values, concatenate the two rounding values of TEMP_C and RAIN_C (for 2 MONTH values), and calculate the average of TEMP_C by the ID values, rounding it to 2 decimal places from the table metric_stats.

Result:

```
MariaDB [1961Odm]> select ID,
    ->  GROUP_CONCAT(ROUND(TEMP_C,2) SEPARATOR ', ') AS TEMP_C,
    ->  GROUP_CONCAT(ROUND(RAIN_C,2) SEPARATOR ', ') AS RAIN_C,
    ->  ROUND(AVG(TEMP_C),2)
    ->  FROM metric_stats GROUP BY ID;
+----+--------------+-----------+----------------------+
| ID | TEMP_C       | RAIN_C    | ROUND(AVG(TEMP_C),2) |
+----+--------------+-----------+----------------------+
| 10 | -10.39, 3.72 | 0.74, 0.95 |                -3.33 |
| 12 | 26.06, 30.39 | 0.50, 0.23 |                28.22 |
| 13 | 14.11, 33.17 | 0.12, 2.03 |                23.64 |
| 37 | 1.56, -3.56  | 1.11, 0.45 |                -1.00 |
| 44 | -2.61, 23.78 | 0.07, 0.83 |                10.58 |
| 49 | -6.89, 8.61  | 1.36, 0.19 |                 0.86 |
| 56 | 17.94, -4.06 | 1.63, 1.36 |                 6.94 |
| 66 | -14.06, 18.78 | 0.83, 1.78 |                2.36 |
| 87 | 37.00, -9.94 | 0.66, 0.27 |                13.53 |
| 94 | 25.11, -7.61 | 1.39, 0.49 |                 8.75 |
+----+--------------+-----------+----------------------+
```