1. **Give the count of the minimum number of days for the time when temperature reduced**

**Solution:**

WITH T1 AS

(SELECT DATE, TEMPERATURE,

ROW\_NUMBER() OVER (ORDER BY DATE) AS ROW\_ID,

LEAD(TEMPERATURE) OVER (ORDER BY DATE) AS LEAD\_DATE

FROM weatherdata),

T2 AS

(SELECT ROW\_ID, DATE, TEMPERATURE, LEAD\_DATE,

ROW\_NUMBER() OVER (ORDER BY DATE) AS ROW\_NUM,

ROW\_ID - ROW\_NUMBER() OVER (ORDER BY DATE) AS DIFF

FROM T1

WHERE TEMPERATURE > LEAD\_DATE),

T3 AS

(SELECT \*, COUNT(\*) OVER (PARTITION BY DIFF) AS CONC\_DAYS

FROM T2)

SELECT MIN(CONC\_DAYS) FROM T3;

**Output:**

1

1. **Find the temperature as Cold / hot by using the case and avg of values of the given data set**

**Solution:**

SELECT

DATE, AVG(TEMPERATURE),

CASE

WHEN AVG(TEMPERATURE) > 50 THEN 'HOT'

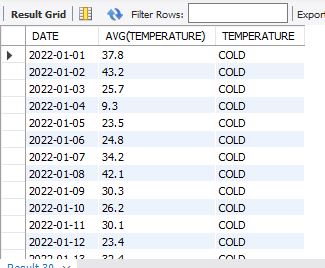
ELSE 'COLD'

END AS TEMPERATURE

FROM weatherdata

GROUP BY DATE;

**Output:**



1. **Can you check for all 4 consecutive days when the temperature was below 30 Fahrenheit**

**Solution:**

WITH T1 AS

(SELECT \*,

ROW\_NUMBER() OVER (ORDER BY DATE) AS ROW\_ID

FROM weatherdata), T2 AS

(SELECT DATE, TEMPERATURE, ROW\_ID,

ROW\_NUMBER() OVER (ORDER BY DATE) AS ROW\_NUM,

ROW\_ID - ROW\_NUMBER() OVER (ORDER BY DATE) AS DIFF

FROM T1

WHERE TEMPERATURE < 30), T3 AS

(SELECT \*, COUNT(\*) OVER (partition by DIFF) AS CONC\_DAYS FROM T2)

SELECT DATE, TEMPERATURE FROM T3 WHERE CONC\_DAYS = 4;

**Output:**

2022-01-03 25.7

2022-01-04 9.3

2022-01-05 23.5

2022-01-06 24.8

2022-03-30 23.3

2022-03-31 25.8

2022-04-01 24.8

2022-04-02 29.1

2022-11-14 25.7

2022-11-15 19.6

2022-11-16 19.2

2022-11-17 28.8

2023-01-23 27.1

2023-01-24 23.9

2023-01-25 25.6

2023-01-26 28.1

2024-01-09 18.9

2024-01-10 3.5

2024-01-11 11.7

2024-01-12 24

2024-12-22 11.1

2024-12-23 13.1

2024-12-24 18

2024-12-25 21.1

2025-12-08 26.8

2025-12-09 12.1

2025-12-10 18.3

2025-12-11 21.7

2026-03-22 26.2

2026-03-23 12.1

2026-03-24 8.6

2026-03-25 16

2026-11-21 13.4

2026-11-22 17.3

2026-11-23 20.1

2026-11-24 25.9

2028-01-20 27

2028-01-21 16.7

2028-01-22 13

2028-01-23 27.8

2029-01-14 29.8

2029-01-15 23

2029-01-16 20.7

2029-01-17 29.8

2029-04-28 29.2

2029-04-29 29.9

2029-04-30 29

2029-05-01 28.5

2029-12-05 28.4

2029-12-06 13.5

2029-12-07 2

2029-12-08 1.6

2030-01-24 20.6

2030-01-25 15.7

2030-01-26 12.7

2030-01-27 21.9

2030-02-23 20.2

2030-02-24 8.5

2030-02-25 14.8

2030-02-26 20.5

2030-12-04 25.5

2030-12-05 28.6

2030-12-06 21.4

2030-12-07 25.6

2030-12-21 15.3

2030-12-22 22.3

2030-12-23 25.7

2030-12-24 23.6

2031-12-01 28.4

2031-12-02 20.4

2031-12-03 12.7

2031-12-04 18.8

2032-03-02 11.6

2032-03-03 1

2032-03-04 5.9

2032-03-05 27

2032-03-13 24.1

2032-03-14 22.5

2032-03-15 24.1

2032-03-16 26.8

2032-10-27 15.3

2032-10-28 16.1

2032-10-29 5.7

2032-10-30 9.2

2032-12-14 23

2032-12-15 17.4

2032-12-16 11.5

2032-12-17 23.1

2033-02-17 24

2033-02-18 16

2033-02-19 11.7

2033-02-20 12

2033-02-23 29.4

2033-02-24 24.5

2033-02-25 16.6

2033-02-26 23.6

1. **Can you find the maximum number of days for which temperature dropped**

**Solution:**

WITH T1 AS

(SELECT DATE, TEMPERATURE,

ROW\_NUMBER() OVER (ORDER BY DATE) AS ROW\_ID,

LEAD(TEMPERATURE) OVER (ORDER BY DATE) AS LEAD\_DATE

FROM weatherdata),

T2 AS

(SELECT ROW\_ID, DATE, TEMPERATURE, LEAD\_DATE,

ROW\_NUMBER() OVER (ORDER BY DATE) AS ROW\_NUM,

ROW\_ID - ROW\_NUMBER() OVER (ORDER BY DATE) AS DIFF

FROM T1

WHERE TEMPERATURE > LEAD\_DATE),

T3 AS

(SELECT \*, COUNT(\*) OVER (PARTITION BY DIFF) AS CONC\_DAYS

FROM T2)

SELECT MAX(CONC\_DAYS) FROM T3;

**Output:**

7

1. **Can you find the average of average humidity from the dataset**

**( NOTE: should contain the following clauses: group by, order by, date )**

**Solution:**

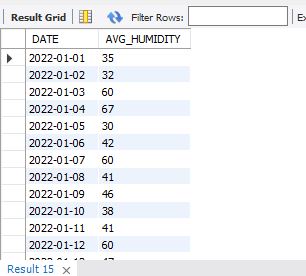
SELECT DATE, AVG(AVG\_HUMIDITY\_PERC) as AVG\_HUMIDITY

FROM weatherdata

GROUP BY DATE

ORDER BY DATE;

Output:



1. **Use the GROUP BY clause on the Date column and make a query to fetch details for average windspeed ( which is now windspeed done in task 3 )**

**Solution:**

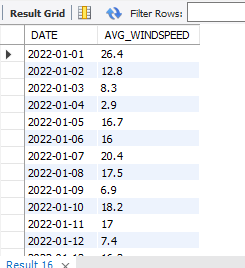
SELECT DATE, AVG(AVG\_WINDSPEED\_MPH) AS AVG\_WINDSPEED

FROM weatherdata

GROUP BY DATE

ORDER BY DATE;

**Output:**



1. **Please add the data in the dataset for 2034 and 2035 as well as forecast predictions for these years**

**( NOTE: data consistency and uniformity should be maintained )**

**Solution:**

WITH recursive T1 AS

(SELECT DATE, TEMPERATURE,

ROW\_NUMBER() OVER (ORDER BY DATE) AS ROW\_NUM

FROM weatherdata),

T2 AS

(SELECT SLOPE, y\_bar\_max - x\_bar\_max \* SLOPE as INTERCEPT

FROM

(SELECT

SUM((ROW\_NUM - x\_bar) \* (TEMPERATURE - y\_bar)) / SUM((ROW\_NUM - x\_bar) \* (ROW\_NUM- x\_bar)) as SLOPE,

MAX(x\_bar) as x\_bar\_max,

MAX(y\_bar) as y\_bar\_max

FROM

(SELECT ROW\_NUM, AVG(ROW\_NUM) OVER() as x\_bar,

TEMPERATURE, AVG(TEMPERATURE) OVER() as y\_bar

FROM T1) DATA1

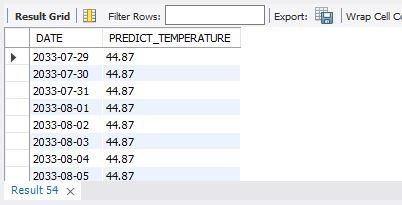
) DATA2)

SELECT predictiondata.DATE,

ROUND((predictiondata.ROW\_NUM \* (SELECT slope FROM T2) + (SELECT intercept FROM T2)),2) as PREDICT\_TEMPERATURE

FROM predictiondata;

**Output:**



1. **If the maximum gust speed increases from 55mph, fetch the details for the next 4 days**

**Solution:**

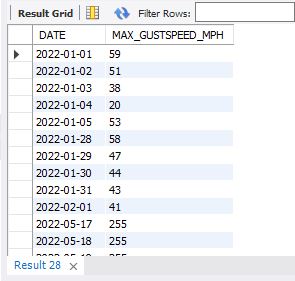
SELECT wd.DATE, wd.MAX\_GUSTSPEED\_MPH

FROM weatherdata wd

INNER JOIN (SELECT DATE FROM weatherdata WHERE MAX\_GUSTSPEED\_MPH > 55) sub

ON wd.DATE BETWEEN sub.DATE AND DATE\_ADD(sub.DATE, INTERVAL 4 DAY);

**OUTPUT:**



1. **Find the number of days when the temperature went below 0 degrees Celsius**

**Solution:**

SELECT COUNT(\*) FROM weatherdata

WHERE TEMPERATURE < 0;

**OUTPUT**:

14

1. **Create another table with a “Foreign key” relation with the existing given data set.**

**Solution:**

-- Created index on our weather analysis dataset.

CREATE INDEX idx\_date ON weatherdata (DATE);

-- Created another table with a “Foreign Key”

CREATE TABLE incharge\_details(

ID INT NOT NULL PRIMARY KEY,

INCHARGE\_NAME VARCHAR(50),

DEPARTMENT VARCHAR(50),

ENTRY\_DATE DATE,

FOREIGN KEY(ENTRY\_DATE) REFERENCES weatherdata(DATE)

);

**OUTPUT:**

18:34:09 Create Table incharge\_details( ID INT NOT NULL PRIMARY KEY, INCHARGE\_NAME VARCHAR(50), DEPARTMENT VARCHAR(50), ENTRY\_DATE DATE, FOREIGN KEY(ENTRY\_DATE) REFERENCES weatherdata(DATE) ) 0 row(s) affected 0.078 sec