

585. Investments in 2016

May 13, 2017 | 18.2K views

★★★★★
Average Rating: 4.20 (15 votes)

Write a query to print the sum of all total investment values in 2016 (**TIV_2016**), to a scale of 2 decimal places, for all policy holders who meet the following criteria:

1. Have the same **TIV_2015** value as one or more other policyholders.
2. Are not located in the same city as any other policyholder (i.e.: the (latitude, longitude) attribute pairs must be unique).

Input Format:

The **insurance** table is described as follows:

Column Name	Type
PID	INTEGER(11)
TIV_2015	NUMERIC(15,2)
TIV_2016	NUMERIC(15,2)
LAT	NUMERIC(5,2)
LON	NUMERIC(5,2)

where **PID** is the policyholder's policy ID, **TIV_2015** is the total investment value in 2015, **TIV_2016** is the total investment value in 2016, **LAT** is the latitude of the policy holder's city, and **LON** is the longitude of the policy holder's city.

Sample Input

PID	TIV_2015	TIV_2016	LAT	LON
1	10	5	10	10
2	20	20	20	20
3	10	30	20	20
4	10	40	40	40

Sample Output

TIV_2016
45.00

Explanation

The first record in the table, like the last record, meets both of the two criteria. The **TIV_2015** value '10' is as the same as the third and forth record, and its location (LAT, LON) pair is unique.

The second record does not meet any of the two criteria. Its **TIV_2015** is not like any other record, and its location is the same with the third record, which makes the third record fail.

So, the result is the sum of **TIV_2016** of the first and last record, which is 45.

Solution

Approach: Using **GROUP BY** and **COUNT** [Accepted]

Intuition

To decide whether a value in a column is unique or not, we can use **GROUP BY** and **COUNT**.

Algorithm

Check whether the value of a record's **TIV_2015** is unique, if it is not unique, and at the same time, its location (LAT, LON) pair is unique, then this record meeting the criteria. So it should be counted in the sum.

MySQL

```
SELECT
    SUM(insurance.TIV_2016) AS TIV_2016
FROM
    insurance
WHERE
    insurance.TIV_2015 IN
    (
        SELECT
            TIV_2015
        FROM
            insurance
        GROUP BY TIV_2015
        HAVING COUNT(*) > 1
    )
    AND CONCAT(LAT, LON) IN
    (
        SELECT
            CONCAT(LAT, LON)
        FROM
            insurance
        GROUP BY LAT , LON
        HAVING COUNT(*) = 1
    )
;
```

Tips: Concat the **LAT** and **LON** as a whole to represent the location information.

Note: These two criteria should be met without an order, so if you attempt to filter data using criteria #1 first and then criteria #2, you will get a wrong result.

Taking the sample input as an example, the data set will be as following after taking the first criteria.

PID	TIV_2015	TIV_2016	LAT	LON
1	10	5	10	10
3	10	30	20	20
4	10	40	40	40

Then, the second criteria cannot filter any records on this data set. So the result is 75(5+30+40), which is obviously wrong since the location of record with PID '3' is actually the same with the record having been filtered by the first criteria.


PID	TIV_2015	TIV_2016	LAT	LON
2	20	20	20	20
3	10	30	20	20

Rate this article: ★★★★★


PreviousNext

Comments: 31

Sort By

- 

Type comment here... (Markdown is supported)


PreviewPost
- 

meganlee ★961 · August 12, 2018 11:11 PM

CONCAT(LAT, LON) needs to be changed to **CONCAT(LAT, CONCAT(' ', LON))** with a separator ' ' in the middle: think about the following example

Lat = 13, Lon = 3

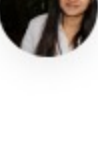
45 · Share · Reply

SHOW 1 REPLY
- 

tyumneva ★83 · July 16, 2018 11:23 PM

Works without concat():

```
select sum(TIV_2016) TIV_2016
from insurance
where TIV_2015 in (select TIV_2015 from insurance group by TIV_2015 having count(1)>1)
```

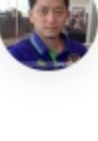
27 · Share · Reply
- 

mbodke41 ★75 · May 10, 2019 6:41 AM

My Solution:

```
select sum(tiv_2016) as TIV_2016
from insurance
where tiv_2015 not in ( select distinct tiv_2015
```

5 · Share · Reply


SHOW 2 REPLIES
- 

DennyZhang ★223 · March 16, 2018 9:57 PM

CONCAT(LAT, LON) may not be right.

e.g


11, 2

3 · Share · Reply
- 

adityajhanwar ★45 · May 23, 2019 8:14 AM


Although my solution uses correlated subqueries, I feel that it is a bit more readable compared to the official solution

```
select
round(sum(tiv_2016)/2) as 'TIV_2016'
```

2 · Share · Reply
- 

ehsanfar ★6 · April 26, 2018 4:39 AM


select sum(TIV_2016) as TIV_2016
from insurance as I1
where TIV_2015 in (select TIV_2015
from insurance
group by TIV_2015

1 · Share · Reply
- 

kyuuuuuuuuuu ★3 · May 26, 2019 1:16 AM

can anyone help with my solution?


```
select round(sum(a.TIV_2016)/2) as TIV_2016
from insurance a, insurance b, insurance c
where
a.pid != b.pid and a.pid != c.pid and
```

0 · Share · Reply
- 

Elaine66 ★10 · May 13, 2019 7:15 AM

Use two partition by is also working:


```
select sum(TIV_2016) as TIV_2016
from
(select PID, TIV_2015, TIV_2016, LAT, LON,
```

0 · Share · Reply
- 

nitinchoudhry ★0 · March 28, 2019 1:21 AM

Why this is incorrect?

```
SELECT
SUM(TIV_2016) AS TIV_2016
FROM
insurance
```

0 · Share · Reply
- 

sasikumardevarajan ★1 · February 28, 2019 1:46 AM

select PID,TIV_2015,TIV_2016 from (select PID,TIV_2015,TIV_2016,LAT,LON,COUNT(TIV_2015) OVER (PARTITION BY TIV_2015) AS CUMU_COUNT,COUNT(LAT) OVER (PARTITION BY LAT) AS LAT_COUNT,COUNT(LON) OVER (PARTITION BY LON) AS LON_COUNT from insurance) a where CUMU_COUNT > 1 AND (LAT_COUNT=1 OR LON_COUNT =1);

0 · Share · Reply