8WEEKSQLCHALLENGE.COM

CASE STUDY #8

FRESH SEGMENTS

EXTRACT MAXIMUM VALUE

DATAWITHDANNY.COM

dataset link available on these site : - https://www.db-fiddle.com/f/jmnwogTsUE8hGqkZv9H7E8/17

QUESTION

CLEANING DATASETS

1. Update the fresh_segments.interest_metrics table by modifying the month_year column to be a date data type with the start of the month.

```
ALTER TABLE interest_metrics
ADD COLUMN month_year_date DATE;
```

```
UPDATE interest_metrics
SET month_year_date = CASE
                WHEN month_year ~ '^[0-9]{2}-[0-9]{4}$' THEN to_date(month_year, 'MM-YYYY')
                ELSE NULL
               END;
```

```
ALTER TABLE interest_metrics
DROP COLUMN month_year
```

```
ALTER TABLE interest_metrics
RENAME COLUMN month_year_date TO month_year;
```

2. What is count of records in the fresh_segments.interest_metrics for each month_year value sorted in chronological order (earliest to latest) with the null values appearing first?

```
select month_year ,count(*) count_of_record
from interest_metrics
group by month_year
order by month_year desc
```

3. How many interest_id values exist in the fresh_segments.interest_metrics table but not in the fresh_segments.interest_map table? What about the other way around?

```
select count(*) count_of_rows_do_not_exist
from
(select id from interest_map
except
select distinct(interest_id) :: int
 from interest_metrics) as except_
```

4. Summarise the id values in the fresh_segments.interest_map by its total record count in this table.

```
SELECT
   id,
   COUNT(*) AS total_record_count
FROM
   interest_map
GROUP BY
   id
ORDER BY
   id;
```

5. What sort of table join should we perform for our analysis and

Check your logic by checking the rows where interest_id = 21246 in your joined output and include all columns from interest_metrics and

all columns from interest_map except from the id column.

```
select *
from interest_map as i
join interest_metrics as im
on i.id = cast(im.interest_id as int )
where i.id = 21246
order by i.id desc
```

INTEREST ANALYSIS

1. Which interests have been present in all month_year dates in our dataset?

```
select
   im.interest_id :: int,
   ip.interest_name
from
(select
interest_id,
count(distinct(month_year)) as count_of_unique_product
from interest_metrics
group by interest_id
having count(distinct(month_year)) = (select count(distinct(month_year)) from interest_metrics)
) im
join interest_map as ip
on im.interest_id = ip.id :: varchar
order by im.interest_id :: int
```

2. Using this same total_months measure - calculate the cumulative percentage of all records starting at 14 months - which total_months value passes the 90% cumulative percentage value?

```
with mycte as
(select
month_year,
count(month_year) record_count
from interest_metrics
 group by month_year
) ,mycte2 as
(
select
month_year,
record_count,
sum(record_count) over(order by month_year asc )*100/ (select sum(record_count) from mycte) as cumulative_count
from mycte
)
select
month_year,
record_count,
cumulative_count
from
mycte2
where   cumulative_count >= 90
limit 1
```

3. If we were to remove all interest_id values which are lower than the total_months value we found in the previous question - how many total data points would we be removing?

```
with mycte as
(select
month_year,
count(month_year) record_count
from interest_metrics
```

```
 group by month_year
) ,mycte2 as
(
select
month_year,
record_count,
sum(record_count) over(order by month_year asc )*100/ (select sum(record_count) from mycte) as cumulative_percentage
from mycte
),mycte3 as
(
select
month_year,
record_count,
cumulative_percentage,
count(month_year) as count_of_month
from
mycte2
where   cumulative_percentage >= 90
group by month_year,
record_count,
cumulative_percentage
limit 1
)
select *
from mycte3
as m
join interest_metrics as im
on  im.interest_id = m.count_of_month :: varchar
where im.interest_id < m.count_of_month :: varchar
```

SEGMENT ANALYSIS

1. Using our filtered dataset by removing the interests with less than 6 months worth of data, which are the top 10 and

bottom 10 interests which have the largest composition values in any month_year? Only use the maximum composition value for each interest but you must keep the corresponding month_year.

```
with mycte as
(
        select
        interest_id
        from interest_metrics
        where extract(month from month_year) >= 6
),
mycte2 as
(
        select
        interest_id,
        max(composition) max_composition,
        max(month_year) max_month_year
        from interest_metrics
        where interest_id in (select interest_id from mycte)
        group by interest_id
), mycte3 as
(
        select
        interest_id,
        max_composition,
```

```
        row_number() over( order by max_composition desc ) max_compo_desc,
        row_number() over( order by max_composition asc )max_compo_asc,
        max_month_year
        from mycte2

)
select
interest_id,
max_composition,
max_month_year
from mycte3
WHERE
   max_compo_desc <= 10
  OR max_compo_asc <= 10
order by max_composition desc
```

Which 5 interests had the lowest average ranking value?

```
select ip.interest_name,round(avg(im.ranking),1) avg_ranking
from interest_metrics as im
join interest_map as ip
on im.interest_id = ip.id :: varchar
group by ip.interest_name
order by avg_ranking asc
limit 5
```

2. Which 5 interests had the largest standard deviation in their percentile_ranking value?

```
select interest_id, stddev(percentile_ranking) as standard_percentile
from interest_metrics
group by interest_id
```

```
order by standard_percentile desc
limit 5
```

3. For the 5 interests found in the previous question - what was minimum and maximum percentile_ranking values for each interest and its corresponding year_month value?

```
with mycte as
(
select interest_id, stddev(percentile_ranking) as standard_percentile
from interest_metrics
group by interest_id
order by standard_percentile desc
limit 5
)
select
interest_id,
max(percentile_ranking) as max_percentile,
min(percentile_ranking) min_percentile,
month_year
from interest_metrics
where interest_id in (select interest_id from mycte)
group by interest_id,month_year
order by interest_id
```

4. How would you describe our customers in this segment based off their composition and ranking values?

```
select
ranking,
count(ranking) count_of_ranking,
sum(composition) as total_copositions
from interest_metrics
```

```
group by
ranking
order by total_copositions desc
```

INDEX ANALYISIS

NOTE :- The index_value is a measure which can be used to reverse calculate the average composition for Fresh Segments' clients and

Average composition can be calculated by dividing the composition column by the index_value column rounded to 2 decimal places.

1. What is the top 10 interests by the average composition for each month?

```
select _month,sum(avg_composition) as interest_avg_comp
from interest_metrics
group by _month
order by interest_avg_comp desc
```

2. For all of these top 10 interests - which interest appears the most often?

```
select _month,count(interest_id) count_of_interest,sum(avg_composition) as interest_avg_comp
from interest_metrics
group by _month
order by interest_avg_comp desc
limit 1
```

3. What is the average of the average composition for the top 10 interests for each month?

```
with mycte as (
select
interest_id,
avg_composition,
extract(month from month_year) months,
row_number() over(partition by interest_id order by avg_composition desc)
from interest_metrics
)
select months,avg(avg_composition) as avg_comp
from mycte
where row_number < 10
group by months
```