

8WeekSQLChallenge.com

CASE STUDY #5



DATA MART

fresh is best

DataWithDanny.com

Dataset link available here :- <https://8weeksqlchallenge.com/case-study-5/>

data_mart.weekly_sales

week_date	VARCHAR(7)
region	VARCHAR(13)
platform	VARCHAR(7)
segment	VARCHAR(4)
customer_type	VARCHAR(8)
transactions	INTEGER
sales	INTEGER

Data Cleansing Steps

1. Convert the week_date to a DATE format

```
alter table weekly_sales
```

```
alter column week_date type date USING week_date::date
```

2. Add a week_number as the second column for each week_date value.

```
alter table weekly_sales
```

```
add column week_number int
```

```
update weekly_sales
```

```
set week_number= extract(week from week_date)
```

3. Add a month_number with the calendar month for each week_date value.

```
alter table weekly_sales
```

```
add column month_number int
```

```
update weekly_sales
```

```
set month_number = extract(month from week_date)
```

4. Add a calendar_year column as the 4th column containing either 2018, 2019 or 2020 values.

```
alter table weekly_sales
```

```
add column calender_year numeric
```

```
update weekly_sales
```

```
set calender_year = extract(year from weekly_date)
```

5. Ensure all null string values with an "unknown" string value in the original segment column as well as the new age_band and demographic columns.

for demographic

alter table weekly_sales

add column demographic varchar(15)

update weekly_sales

set demographic =

case

when segment_text = 'C' then 'Couples'

when segment_text = 'F' then 'Families'

else 'UNKNOWN'

for segment

alter table weekly_sales

alter column segment to varchar(20) using segment :: varchar(20)

update weekly_sales

set segment =

case when segment is null then 'UNKNOWN' else segment end

6. Generate a new avg_transaction column as the sales value divided by transactions rounded to 2 decimal places for each record.

alter table weekly_sales

add column avg_transaction numeric

update weekly_sales

```
set avg_transaction = round(sales/transactions,1)
```

2. Data Exploration

1. What day of the week is used for each week_date value?

```
select to_char(week_date,'day')  
from weekly_sales
```

2. What range of week numbers are missing from the dataset?

```
create table weekly as  
select generate_series(1,52)  
  
select week_number  
from weekly  
where week_number < (select min(week_number) from weekly_sales)
```

3. How many total transactions were there for each year in the dataset?

```
select calender_year,count(transactions) as total_transactions  
from weekly_sales  
group by calender_year  
order by calender_year
```

4. What is the total sales for each region for each month?

```
select month_number,region,sum(sales) total_sales  
from weekly_sales  
group by month_number,region  
order by month_number
```

5. What is the total count of transactions for each platform

```
select platform,count(transactions) total_transactions
from weekly_sales
group by platform
order by total_transactions desc
```

6. What is the percentage of sales for Retail vs Shopify for each month?

```
select month_number,
round(((select sum(sales) from weekly_sales where platform = 'Shopify') * 100.0/(select sum(sales)
from weekly_sales)),2) as Shopify_percentage,
round(((select sum(sales) from weekly_sales where platform = 'Retail') * 100.0/(select sum(sales)
from weekly_sales)),2) as Retail_percentage
from weekly_sales
group by month_number
```

7. What is the percentage of sales by demographic for each year in the dataset?

```
select calender_year,
sum(case when demographic = 'Couples' then sales else 0 end)*100.0/sum(sales) as
percentage_Couples,
sum(case when demographic = 'Families' then sales else 0 end)*100.0/sum(sales) as
percentage_Families,
sum(case when demographic = 'UNKNOWN' then sales else 0 end)*100.0/sum(sales)
percentage_Unkowns
from weekly_sales
group by calender_year
order by calender_year
```

8. Which age_band and demographic values contribute the most to Retail sales?

```
select age_band,demographic,sum(sales) total_sales
from weekly_sales
where platform = 'Retail'
group by age_band,demographic
order by total_sales desc
limit 1
```

9. Can we use the avg_transaction column to find the average transaction size for each year for Retail vs Shopify? If not - how would you calculate it instead?

```
select calender_year,
avg(case when platform = 'Shopify' then sales else 0 end) avg_shopify_transaction,
avg(case when platform = 'Retail' then sales else 0 end) avg_retail_transaction
from weekly_sales
group by calender_year
```

10. What is the total sales for the 4 weeks before and after 2020-06-15?

BEFORE

```
select sum(sales) as before_4_week
from weekly_sales
where week_number = (select extract(week from week_date)
from weekly_sales
where week_date = '2020-06-15'
limit 1) - 4 and calender_year = '2020'
```

AFTER

```
select sum(sales) as After_4_week
from weekly_sales
```

```
where week_number = (select extract(week from week_date)
from weekly_sales
where week_date = '2020-06-15'
limit 1) + 4 and calender_year = '2020'
```

11. Which areas of the business have the highest negative impact in sales metrics performance in 2020 for the 12 week before and after period?

region

platform

age_band

demographic

customer_type

with mycte as

(select

date '2020-06-15' - interval '12 week' before_date,

date '2020-06-15' + interval '12 week' after_date

)

select

region,

platform,

age_band,

demographic,

customer_type,

sum(case when week_date >= before_date and week_date < '2020-06-15' then sales else 0 end) as
total_sales_before_date,

sum(case when week_date > '2020-06-15' and week_date < after_date then sales else 0 end) as
total_sales_after_date

from weekly_sales,mycte

group by region,platform,age_band,demographic,customer_type

order by total_sales_after_date, total_sales_before_date

limit 5

