

# WALMART DATA ANALYSIS USING SQL

Q.1 Find the different payment method and number of transaction , number of quantity sold

```
Select payment_method ,  
count(*) as no_payments ,  
sum(quantity) as no_quantity_sold  
from walmart  
group by payment_method
```

Q.2 Identify the highest rated category in each branch , displaying the branch , category

```
AVG Rating  
Select * from (  
SELECT  
branch ,  
category,  
avg(rating) as avg_rating ,  
rank() over(partition by branch order by avg(rating) desc) as rank  
from walmart  
group by 1,2  
) where rank = 1
```

--Q.3 Identify the busiest day for each branch based on the number of transactions

```
select * From  
(  
Select  
branch,  
TO_CHAR(date ,'day') as day_name,  
count(*) as no_transactions ,  
Rank() over (partition by branch ORDER by count(*) desc) as rank  
from walmart  
group by 1,2  
)  
where rank = 1
```

# WALMART DATA ANALYSIS USING SQL

--Q.4 calculate the total quantity of items sold per payment method . List payment\_method and total\_quantity

```
select payment_method ,  
--count(*) as no_payments ,  
sum(quantity) as no_qty_sold  
from walmart  
group by payment_method
```

--Q.5 Determine the average , minimum , and maximum rating of category for each city.

--List the city , average rating , min\_rating and max\_rating

```
Select  
city,  
category,  
avg(rating) as avg_rating ,  
min(rating) as min_rating ,  
max(rating) as max_rating  
from walmart  
group by 1,2
```

--Q.6 calculate the total profit for each category by considering total profit as (unit\_price , quantity and profit margin)

--List the category and total\_profit , ordered from highest to lowest profit

```
select  
category ,  
sum(total) as total_revenue,  
sum(total * profit_margin)  
from walmart  
group by 1
```

# WALMART DATA ANALYSIS USING SQL

--Q.7 Determine the most common payment method for each branch . Display branch and the preferred\_payment\_category

```
--Select * from walmart  
with cte  
as  
(select  
branch ,  
payment_method,  
count(*) as total_trans,  
Rank() over (partition by branch order by count(*) desc) as rank  
from walmart  
group by 1,2)  
select * from cte  
where rank = 1
```

--Q.8 categorize sales into 3 group Morning , afternoon , evening

-- find out each of the shift and number of invoices

```
select  
branch,  
case  
when extract (hour from (time::time)) < 12 Then 'Morning'  
when extract (hour from (time::time)) between 12 and 17 Then 'Afternoon'  
Else 'Evening'  
end day_time,  
count(*)  
from walmart  
group by 1 ,2  
order by 1,3 desc
```

# WALMART DATA ANALYSIS USING SQL

--Q.9 identify 5 branches with the highest decrease ratio in revenue compared to last year (current year 2023 and last year 2022)

-- revenue decrease ratio ==  $\frac{\text{last\_rev} - \text{current\_rev}}{\text{last\_rev}} * 100$

--2022 sales

with revenue\_2022

as

(

select

branch,

sum(total) as revenue

from walmart

where extract (year from date) = 2022

group by 1

),

revenue\_2023

as

(

select

branch,

sum(total) as revenue

from walmart

where extract (year from date) = 2023

group by 1

)

select

ls.branch ,

ls.revenue as last\_year\_revenue,

cs.revenue as cr\_year\_revenue,

# WALMART DATA ANALYSIS USING SQL

```
Round((ls.revenue -cs.revenue)::numeric/ ls.revenue::numeric * 100,
```

```
2) as rev_dec_ratio
```

```
from revenue_2022 as ls
```

```
join
```

```
revenue_2023 as cs
```

```
on ls.branch = cs.branch
```

```
where
```

```
ls.revenue > cs.revenue
```

```
order by 4 desc
```

```
limit 5
```