

-- WALMART SALES DATA EXPLORATION

-- Select the data within the 3rd quarter of 2012

```
select * from `walmart-sales-data-352915.Project.Walmart`  
where year = 2012 and month between 6 and 9  
order by store, Date asc;
```

-- Visualizing total sales in Q3 of 2012 for each store

```
select Store, round(sum(Weekly_Sales)/1000000, 2) as Total_Sales_in_Million from  
`walmart-sales-data-352915.Project.Walmart`  
where year = 2012 and month between 6 and 9  
group by Store  
order by sum(Weekly_Sales) desc;
```

-- Calculating the growth rate per store in the 3rd quarter of 2012

-- 1. Year on year growth

```
with q2012 as  
(select Store, sum(Weekly_Sales) as Total_Sales from `walmart-sales-data-  
352915.Project.Walmart`  
where year = 2012 and month between 6 and 9  
group by Store  
order by Store),
```

```
q2011 as  
(select Store, sum(Weekly_Sales) as Total_Sales from `walmart-sales-data-  
352915.Project.Walmart`  
where year = 2011 and month between 6 and 9  
group by Store  
order by Store)
```

```
select q2012.Store as Store, round((q2012.Total_Sales -  
q2011.Total_Sales)/q2011.Total_Sales, 4) as Perc_Growth_YoY  
from q2012  
join q2011 on q2011.Store = q2012.Store  
order by Perc_Growth_YoY desc;
```

-- FINDINGS: Store 44 has the highest year on year growth rate of 0.1238% and store 45 experienced the highest loss of 0.0017%.

-- There are no significant growth and loss from the comparison, implying that the stores

-- remain consistent in maintaining their sales.

-- We Use Q3 as it is the most recent quarter with the most complete data.

-- 2. Quarter over quarter growth

```
with cte as (  
select  
    Store,  
    --EXTRACT(YEAR FROM DATE)  
    EXTRACT(QUARTER FROM Date) Quarter,  
    SUM(Weekly_Sales) Quarterly_Sales
```

```

from Project.Walmart
where EXTRACT(YEAR FROM Date) = 2012
group by 1,2
order by 1,2
),
x as (
select Store, Quarter, (quarterly_sales - lag(quarterly_sales,1) over(partition by
Store order by quarter))/lag(quarterly_sales,1) over(partition by Store order by
quarter) Growth
from cte
)
select Store, round(Growth, 4) as Growth
from x
where quarter = 3
order by growth desc;

```

-- FINDINGS: Only 10 stores have shown positive quarterly growth rate, the rest have negative growth.

-- Although store 4 has the highest sales on the 3rd quarter of 2012, it's growth rate is shown to be negative.

-- Some holidays have a positive/negative impact on sales.

-- Find out holidays which have higher sales than the mean sales in non-holiday season for all stores together

-- 1. Assigning Holiday Name to Each presented Date and the overall average sales

-- Super Bowl: 12-Feb-10, 11-Feb-11, 10-Feb-12, 8-Feb-13

-- Labour Day: 10-Sep-10, 9-Sep-11, 7-Sep-12, 6-Sep-13

-- Thanksgiving: 26-Nov-10, 25-Nov-11, 23-Nov-12, 29-Nov-13

-- Christmas: 31-Dec-10, 30-Dec-11, 28-Dec-12, 27-Dec-13

```

with Assign as
(select *,
case
when Date = "2010-02-12" or Date = "2011-02-11" or Date = "2012-02-10" or Date =
"2013-02-08" then "Super Bowl"
when Date = "2010-09-10" or Date = "2011-09-09" or Date = "2012-09-07" or Date =
"2013-09-06" then "Labour Day"
when Date = "2010-11-26" or Date = "2011-11-25" or Date = "2012-11-23" or Date =
"2013-11-29" then "Thanksgiving"
when Date = "2010-12-31" or Date = "2011-12-30" or Date = "2012-12-28" or Date =
"2013-12-27" then "Christmas"
else "None"
end as Occasion, (select avg(Weekly_Sales) from `walmart-sales-data-
352915.Project.Walmart`) as Avg_WS
from `walmart-sales-data-352915.Project.Walmart`
order by Store, Date)

```

-- For visualization: Walmart Total Sales from 2010 - 2012 by Occasion

-- select round(sum(Weekly_Sales)/1000000, 2) as Total_Sales_in_Million, Occasion, Year

-- from Assign

-- where Holiday_Flag = 1

-- group by Occasion, Year

-- order by Year, Occasion;

```
-- Selecting Store where there's holiday and the weekly sales is higher than the
overall average sales
select Store, round(Weekly_Sales/1000000, 2) as Total_Sales_in_Million, Occasion
from Assign
where Holiday_Flag = 1 and Weekly_Sales > Avg_WS
order by Occasion, Weekly_Sales;
```

-- Provide a monthly and semester view of sales in units and give insights
(Visualize on Tableau)

-- 1. Month with the highest sales from all stores

```
select Year,
case
when Month = 1 then "January"
when Month = 2 then "February"
when Month = 3 then "March"
when Month = 4 then "April"
When Month = 5 then "May"
when Month = 6 then "June"
when Month = 7 then "July"
when Month = 8 then "August"
when Month = 9 then "September"
When Month = 10 then "October"
When Month = 11 then "November"
When Month = 12 then "December"
end as MonthN, round(sum(Weekly_Sales)/1000000, 2) as Total_Sales_in_Million
from `walmart-sales-data-352915.Project.Walmart`
group by Year, Month
order by sum(Weekly_Sales) desc, Year, Month;
```

-- FINDINGS = The highest sales made by all of the stores are on December 2020.
-- The lowest sales made by all of the stores are on January 2011.

-- 2. Highest Monthly Sales by Store

```
select Store, Year,
case
when Month = 1 then "January"
when Month = 2 then "February"
when Month = 3 then "March"
when Month = 4 then "April"
When Month = 5 then "May"
when Month = 6 then "June"
when Month = 7 then "July"
when Month = 8 then "August"
when Month = 9 then "September"
When Month = 10 then "October"
When Month = 11 then "November"
When Month = 12 then "December"
end as MonthN, round(sum(Weekly_Sales)/1000000, 2) as Monthly_Sales_in_Million
from `walmart-sales-data-352915.Project.Walmart`
group by Store, Year, Month
order by sum(Weekly_Sales) desc;
```

```
-- FINDINGS = Store 20 has the highest monthly sales of $13.55 Million on December 2010.  
--           Store 33 has the lowest monthly sales of $0.98 Million on August and November 2010, January and June 2011.
```

```
-- 3. Monthly sales by year  
select case  
when Month = 1 then "January"  
when Month = 2 then "February"  
when Month = 3 then "March"  
when Month = 4 then "April"  
When Month = 5 then "May"  
when Month = 6 then "June"  
when Month = 7 then "July"  
when Month = 8 then "August"  
when Month = 9 then "September"  
When Month = 10 then "October"  
When Month = 11 then "November"  
When Month = 12 then "December"  
end as MonthN, Year, round(sum(Weekly_Sales)/1000000, 2) as  
Monthly_Sales_in_Million  
from `walmart-sales-data-352915.Project.Walmart`  
group by Month, Year  
order by Month, Year;
```

```
-- 4. Select top 3 Store with the highest total sales, view by weekly sales  
select Store, round(sum(Weekly_Sales)/1000000, 2) as Total_Sales_in_Million  
from `walmart-sales-data-352915.Project.Walmart`  
group by Store  
order by sum(Weekly_Sales) desc limit 3;
```

```
-- FINDINGS: Store 20, 4, and 14 have the highest total sales compared to other stores available.
```

```
-- Display the store with highest weekly sales (weekly sales progress from 2010 to 2012)  
select Store, Date, round(Weekly_Sales/1000000, 2) as Weekly_Sales_in_Million  
from `walmart-sales-data-352915.Project.Walmart`  
where Store = 20 or Store = 4 or Store = 14  
order by Store, Date;
```

```
-- Go back to R, select the data needed only (if CPI, Unemployment, & Fuel Price impact Weekly Sales, +Time)  
select Weekly_Sales, CPI, Unemployment, Fuel_Price, row_number() over(order by Date) as Time  
from `walmart-sales-data-352915.Project.Walmart`  
where Store = 1 and Year = 2010  
order by Date;
```

```
-- NOTE: The Date is changed to row_number because the regression equation can't input string  
--       and will make dummy encoding instead. Thus, each entry in the column date will be treated  
--       as different condition to each other instead of time frame (number).
```