SWIGGY DATA ANALYSIS – BigQuery Cloud Console

Link to the file –
https://console.cloud.google.com/bigquery?sq=87064456029:10d9c5f02e1a412b823f26aef4db6bfd

```sql
SELECT * FROM swiggy.items;

SELECT * FROM swiggy.orders;

##Distinct Food Items Ordered
SELECT COUNT(DISTINCT name)
FROM swiggy.items;



##Group vegetarian and meat items together
SELECT is_veg, COUNT(name) AS item
FROM swiggy.items
GROUP BY is_veg;

SELECT * FROM swiggy.items
WHERE is_veg = 2;



##Count the number of unique orders
SELECT COUNT(DISTINCT order_id)
FROM swiggy.items;



##Show items containing chicken in their name
SELECT * FROM swiggy.items
WHERE name like '%Chicken%';



## Find item names with Paratha
SELECT * FROM swiggy.items
WHERE name like '%Paratha%';



##Average Items per Order
SELECT COUNT(name)/COUNT(DISTINCT order_id) AS avg_items_per_order
FROM swiggy.items;



##Item ordered the most number of times
SELECT name, COUNT(*)
FROM swiggy.items
```

```sql
GROUP BY name
ORDER BY COUNT(*) DESC;



##Orders during rainy times
SELECT COUNT(*)
FROM `swiggy.orders`
WHERE rain_mode != 0;


##Unique restaurant names
SELECT COUNT(DISTINCT restaurant_name)
FROM swiggy.orders;



##Restaurant with most orders
SELECT restaurant_name, COUNT(*)
FROM swiggy.orders
GROUP BY restaurant_name
ORDER BY COUNT(*) DESC;



##Orders placed per month and year
SELECT format_date('%Y-%m', order_time), COUNT(DISTINCT order_id)
FROM swiggy.orders
GROUP BY format_date('%Y-%m', order_time)
ORDER BY COUNT(DISTINCT order_id) DESC;



##Revenue made by month
SELECT format_date('%Y-%m', order_time), SUM(order_total) AS TotalRevenuePerMonth
FROM swiggy.orders
GROUP BY format_date('%Y-%m', order_time)
ORDER BY TotalRevenuePerMonth DESC;



##Average Order Value
SELECT ROUND((SUM(order_total)/COUNT(DISTINCT order_id)),2) AS AverageOrderValue
FROM swiggy.orders;



##YOY Change in revenue using lag function and ranking the highest year
SELECT format_date('%Y', order_time) AS Year, SUM(order_total) AS Revenue
FROM swiggy.orders
GROUP BY format_date('%Y', order_time);

WITH Final AS(
  SELECT format_date('%Y', order_time) AS Year, SUM(order_total) AS Revenue
FROM swiggy.orders
GROUP BY format_date('%Y', order_time)
)
SELECT Year, Revenue, LAG(Revenue) OVER (ORDER BY Year) AS PreviousRevenue
```

```sql
FROM Final;

WITH Final AS(
  SELECT format_date('%Y', order_time) AS Year, SUM(order_total) AS Revenue
FROM swiggy.orders
GROUP BY format_date('%Y', order_time)
)
SELECT Year, Revenue, RANK() OVER (ORDER BY Revenue DESC) AS Ranking FROM Final;



## Restaurant with highest revenue ranking
with final as (
SELECT restaurant_name,sum(order_total) as revenue
FROM swiggy.orders
group by  restaurant_name)

select restaurant_name,revenue,
rank() over (order by revenue desc) as ranking from final
order by revenue desc;



##Join order and item tables and find product combos using self join
SELECT a.name, a.is_veg, b.restaurant_name,b.order_id,b.order_time
FROM swiggy.items AS a
INNER JOIN swiggy.orders AS b
ON a.order_id = b.order_id;

SELECT a.order_id,a.name,b.name as name2,concat(a.name,"-",b.name)
FROM swiggy.items a
join swiggy.items b
on a.order_id=b.order_id
where a.name!=b.name
and a.name<b.name;
```