1)How many Distinct Items are ordered

```
SELECT count(distinct name) FROM `quantum-theme-344315.projectsql.items` LIMIT 1000
```

2) How many items are veg and non-veg

```
SELECT is_veg,count(name) as items FROM `quantum-theme-344315.projectsql.items`

group by is_veg
```

3)Not a veg item just to check the anomaly data

```
SELECT * FROM `quantum-theme-344315.projectsql.items`

where is_veg=2
```

4)How many orders contain the word chicken

```
SELECT * FROM `quantum-theme-344315.projectsql.items`

where name like '%Chicken%'
```

5)Same as question 4

```
SELECT * FROM `quantum-theme-344315.projectsql.items`

where name like '%Paratha%'
```

6)Average items per order

```
SELECT count(name)/count(distinct order_id) as avgitemsperorder FROM
`quantum-theme-344315.projectsql.items`
```

7)How many times each item is ordered in descending order

```
SELECT name,count(*) FROM `quantum-theme-344315.projectsql.items`
group by name
order by count(*) desc
```

8)Distinct rain modes in the data

```
SELECT distinct rain_mode FROM `quantum-theme-344315.projectsql.orders` LIMIT 1000
```

9)How many distinct restaurants we have ordered from

```
SELECT count(distinct restaurant_name) FROM
`quantum-theme-344315.projectsql.orders` LIMIT 1000
```

10)How many times we have ordered from each restaurant in descending order

```
SELECT restaurant_name,count(*) FROM `quantum-theme-344315.projectsql.orders`

group by restaurant_name

order by count(*) desc
```

11)Which Month had the most orders were placed in descending order of the order_id

```
SELECT format_date('%Y-%m',order_time),count(distinct order_id) FROM
`quantum-theme-344315.projectsql.orders`

group by  format_date('%Y-%m',order_time)

order by count(distinct order_id) desc
```

12)Last order made time

```
SELECT max(order_time) FROM `quantum-theme-344315.projectsql.orders`
```

13)How much revenue was made by swiggy each month with total revenue in descending

```
SELECT format_date('%Y-%m',order_time),sum(order_total) as totalrevenue
FROM `quantum-theme-344315.projectsql.orders`
group by format_date('%Y-%m',order_time)
order by totalrevenue desc
```

14)Average order value per order

```
SELECT sum(order_total)/count(distinct order_id) as aov
FROM `quantum-theme-344315.projectsql.orders`
```

15)Change on revenue year by year

```
SELECT format_date('%Y',order_time),sum(order_total) as revenue
FROM `quantum-theme-344315.projectsql.orders`

group by  format_date('%Y',order_time)
```

16)To show the comparison of revenue between years and able to see the changes in revenue

```
with final as (
SELECT format_date('%Y',order_time) as yearorder,sum(order_total) as revenue
FROM `quantum-theme-344315.projectsql.orders`
group by  format_date('%Y',order_time))

select yearorder,revenue,lag(revenue) over (order by yearorder) as previousrevenue
from
final
```

16) Rank the best revenue year wise with 1 being the highest revenue

```
with final as (
SELECT format_date('%Y',order_time) as yearorder,sum(order_total) as revenue
FROM `quantum-theme-344315.projectsql.orders`
group by  format_date('%Y',order_time))

select yearorder,revenue,
rank() over (order by revenue desc) as ranking from final
```

17) Rank the best revenue restaurant wise in descending order with 1 being the highest revenue

```
with final as (
SELECT restaurant_name,sum(order_total) as revenue
FROM `quantum-theme-344315.projectsql.orders`
group by  restaurant_name)

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

18)In each order what items were ordered using JOIN table function and get a combination of data

```
SELECT a.name,a.is_veg,b.restaurant_name,b.order_id,b.order_time FROM
`quantum-theme-344315.projectsql.items` a
join `quantum-theme-344315.projectsql.orders`  b
on a.order_id=b.order_id
```

19)Adds two items and create a single order using concat function and shows them separately also in 2 different columns with distinct order_items

```
SELECT a.order_id,a.name,b.name as name2,concat(a.name,"-",b.name) FROM
`quantum-theme-344315.projectsql.items` a

join `quantum-theme-344315.projectsql.items` b

on a.order_id=b.order_id
where a.name!=b.name
and a.name<b.name
```