**SQL Portfolio Assignment:**

**GitHub:** [**https://github.com/Hammad112/Foodie-Fi**](https://github.com/Hammad112/Foodie-Fi)

**Medium:** **https://medium.com/@hammadnasir797/sql-case-study-foodie-fi-3a2c579549d9**

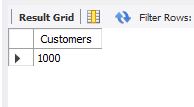
**Question No:01**

How many customers has Foodie-Fi ever had?

**Query:**

Select count(distinct customer\_id) as Customers from subscriptions;

**Output:**

****

**Question No:02**

What is the monthly distribution of trial plan start date values for our dataset - use the start of the month as the group by value.

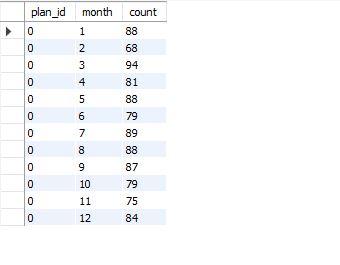
**Query:**

select plan\_id,month(start\_date) as month,count(month(start\_date)) as count from subscriptions

group by month(start\_date),plan\_id

having plan\_id=0;

**OUTPUT:**

****

**Question No:03**

What plan start date values occur after the year 2020 for our dataset? Show the breakdown by count of events for each plan name

**Query:**

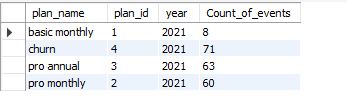
select plans.plan\_name,sp.plan\_id ,year(start\_date) as year,count(year(start\_date)) as Count\_of\_events from subscriptions as sp

join plans on sp.plan\_id=plans.plan\_id

where year(start\_date)>2020

group by plans.plan\_name,plan\_id,year(start\_date);

**OUTPUT:**

****

**Question No:04**

What is the customer count and percentage of customers who have churned rounded to 1 decimal place?

**Query:**

select plan\_name,

count((select count(distinct customer\_id) from subscriptions)) as count\_of\_churned,

ROUND(count(plans.plan\_name)/(select count(distinct customer\_id) from subscriptions),1)\*100 as Percentage

from subscriptions as sp

join plans on sp.plan\_id=plans.plan\_id

where plan\_name='churn';

**OUTPUT:**

**A screenshot of a computer

Description automatically generated**

**Question No:05**

How many customers have churned straight after their initial free trial - what percentage is this rounded to the nearest whole number?

**Query:**

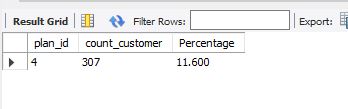
select plan\_id,count(customer\_id) as count\_customer,

Round(count(customer\_id)/(select count(customer\_id) from subscriptions),3)\*100 as Percentage

from subscriptions where plan\_id=4

and customer\_id in(select customer\_id from subscriptions where plan\_id=0);

**OUTPUT:**



**Question No:06**

What is the number and percentage of customer plans after their initial free trial?

**Query:**

SELECT COUNT(plan\_id) AS count\_cust,

ROUND(count(customer\_id)/(select count(customer\_id) from subscriptions),2)\*100 as Percentage

FROM subscriptions

WHERE plan\_id != 0;

**OUTPUT:**

A screenshot of a computer

Description automatically generated

**Question No:07**

What is the customer count and percentage breakdown of all 5 plan name values at 2020-12-31?

**Query:**

select plan\_name,count(customer\_id) as count,

ROUND(count(plan\_name)/(select count(customer\_id) from subscriptions),5)\*100 as Percentage

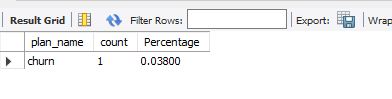
from subscriptions as sp

join plans as p on sp.plan\_id=p.plan\_id

where start\_date='2020-12-31'

group by plan\_name;

**OUTPUT:**

****

**Question No:08**

How many customers have upgraded to an annual plan in 2020?

**Query:**

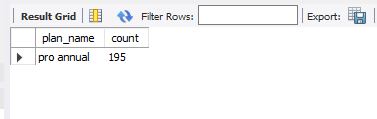
select plan\_name,count(customer\_id) as count

from subscriptions as sp

join plans as p on sp.plan\_id=p.plan\_id

where plan\_name='pro annual' AND year(start\_date)='2020';

**OUTPUT:**



**Question No:09**

How many days on average does it take for a customer to make an annual plan from the day they join Foodie-Fi?

**Query:**

Select count(customer\_id) as No\_of\_customers,AVG(DATEDIFF(

(Select Min(start\_date) from subscriptions as s1 where s1.customer\_id=s2.customer\_id and plan\_id=3),

(Select MIN(start\_date) from subscriptions as s3 where s3.customer\_id=s2.customer\_id )

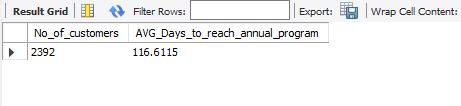
))

as AVG\_Days\_to\_reach\_annual\_program

from subscriptions as s2

where plan\_id !=3;

**OUTPUT:**



**Question No:10**

Can you further breakdown this average value into 30-day periods (i.e. 0-30 days, 31-60 days etc)

**Query:**

SELECT

CASE

WHEN days\_difference >= 0 AND days\_difference <= 30 THEN '0-30 days'

WHEN days\_difference > 30 AND days\_difference <= 60 THEN '31-60 days'

WHEN days\_difference > 61 AND days\_difference <= 90 THEN '61-90 days'

ELSE 'More than 90 days'

END AS period,

COUNT(customer\_id) AS No\_of\_customers,

AVG(days\_difference) AS AVG\_Days\_to\_reach\_annual\_program

FROM (

SELECT

s2.customer\_id,

DATEDIFF(

(SELECT MIN(start\_date) FROM subscriptions AS s1 WHERE s1.customer\_id = s2.customer\_id AND plan\_id = 3),

(SELECT MIN(start\_date) FROM subscriptions AS s3 WHERE s3.customer\_id = s2.customer\_id)) AS days\_difference

FROM subscriptions AS s2 WHERE plan\_id != 3 AND plan\_id != 4) AS differences

GROUP BY period

ORDER BY period;

**OUTPUT:**

A screenshot of a computer

Description automatically generated

**Question No:11**

How many customers downgraded from a pro monthly to a basic monthly plan in 2020?

**Query:**

select plan\_id,count(distinct customer\_id)as downgraded\_from\_annual\_to\_basic from subscriptions

where plan\_id =1

and customer\_id in

(select distinct customer\_id from subscriptions where plan\_id=2 and year(start\_date)=2020);

**OUTPUT:**

A computer screen shot

Description automatically generated