**Part A**

SELECT \* FROM subscriptions

inner join plans on subscriptions.plan\_id=plans.plan\_id

limit 20;

A screenshot of a data table

Description automatically generated

We quickly see that all customers started with a trial subscription before signing up for other subcriptions

Cust 1 is a monthly user after the trial period ended.

Cust 2 is pro annual user after the trial period ended

Cust 3 is monthly user

Cust 4 was a monthly users before ending his subscription in 3 months on 21/04

Cust 5 is a monthly user

Cust 6 used to be a monthly user and kept his monthly subscription active for 3 months

Cust 7 was in basic monthly plan for 3 month before switching to pro monthly

Cust 8 upgraded to pro monthly after 2 months of basic monthly plan.

Part B

Query : SELECT count(distinct(customer\_id)) FROM subscriptions;

A screenshot of a computer

Description automatically generated

1. 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

SELECT month(start\_date) as 'month',count(distinct(customer\_id)) FROM subscriptions

inner join plans on subscriptions.plan\_id=plans.plan\_id

where subscriptions.plan\_id=0

group by month

order by month ;

A screenshot of a data

Description automatically generated

1. What plan start\_date values occur after the year 2020 for our dataset? Show the breakdown by count of events for each plan\_name

SELECT count(plan\_name),plan\_name,subscriptions.plan\_id FROM subscriptions

inner join plans on subscriptions.plan\_id=plans.plan\_id

where year(start\_date)>2020

group by plan\_name

order by subscriptions.plan\_id;

A screenshot of a computer

Description automatically generated

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

with cte1 as

(

select count(distinct(customer\_id)) as cust\_count,

sum(case

when subscriptions.plan\_id=4 then 1

else 0

end) as churn\_cust

from subscriptions

inner join plans on plans.plan\_id=subscriptions.plan\_id

)

select churn\_cust,round((churn\_cust/cust\_count)\*100,1)

from cte1;

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

with cte1 as

(

select customer\_id,sum(price) as price\_paid from subscriptions

inner join plans on subscriptions.plan\_id=plans.plan\_id

group by customer\_id

order by customer\_id

)

select count(customer\_id) as cust\_count, (count(customer\_id)/1000)\*100 as churn\_percentage from cte1

where price\_paid=0;

A screenshot of a computer

Description automatically generated

Why this work out ?

Customers can sign up to an initial 7 day free trial will automatically continue with the pro monthly subscription plan unless they cancel, downgrade to basic or upgrade to an annual pro plan at any point during the trial.

So if they do not cancel after their free trial they the price paid is not 0 else the price paid is 0 which indicates after trial they have chosen to churn else they would be billed for the pro monthly subscriptions.

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

7. What is the customer count and percentage breakdown of all 5 plan\_name values at 2020-12-31?

WITH latest\_plan\_cte AS

(SELECT \*,

row\_number() over(PARTITION BY customer\_id

ORDER BY start\_date DESC) AS latest\_plan

FROM subscriptions

JOIN plans USING (plan\_id)

WHERE start\_date <='2020-12-31' )

SELECT plan\_id,

plan\_name,

count(customer\_id) AS customer\_count,

round(100\*count(customer\_id) /

(SELECT COUNT(DISTINCT customer\_id)

FROM subscriptions), 2) AS percentage\_breakdown

FROM latest\_plan\_cte

WHERE latest\_plan = 1

GROUP BY plan\_id

ORDER BY plan\_id;

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

select count(distinct(customer\_id)) as cust\_count from subscriptions

inner join plans on plans.plan\_id=subscriptions.plan\_id

where subscriptions.plan\_id=3 and year(start\_date)=2020;

A white background with a black line

Description automatically generated

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

with cte1 as

(

select customer\_id,subscriptions.plan\_id,start\_date as pro\_annual\_date from subscriptions

inner join plans on plans.plan\_id=subscriptions.plan\_id

where subscriptions.plan\_id=3

group by customer\_id

),

cte2 as

(

select customer\_id,subscriptions.plan\_id,start\_date as trial\_date from subscriptions

inner join plans on plans.plan\_id=subscriptions.plan\_id

where subscriptions.plan\_id=0

group by customer\_id

)

select avg(datediff(pro\_annual\_date,trial\_date)) from cte1

inner join cte2 on cte1.customer\_id=cte2.customer\_id;

A screenshot of a computer

Description automatically generated

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

select count(customer\_id),quarter(start\_date) as quarter\_report from subscriptions

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

where plans.plan\_id=3

group by quarter(start\_date)

order by quarter\_report;

A screenshot of a computer

Description automatically generated

Further breakdown

with cte1 as

(

select customer\_id,plan\_name,price,start\_date,plans.plan\_id,

case

when abs(datediff(start\_date,'2020-01-01')>=0) and abs(datediff(start\_date,'2020-01-01')<=30) then '0-30 days'

when abs(datediff(start\_date,'2020-01-01')>=31) and abs(datediff(start\_date,'2020-01-01')<=60) then '31-60 days'

when abs(datediff(start\_date,'2020-01-01')>=61) and abs(datediff(start\_date,'2020-01-01')<=120) then '61-120 days'

when abs(datediff(start\_date,'2020-01-01')>=121) and abs(datediff(start\_date,'2020-01-01')<=150) then '121-150 days'

when abs(datediff(start\_date,'2020-01-01')>=151) and abs(datediff(start\_date,'2020-01-01')<=180) then '151-180 days'

when abs(datediff(start\_date,'2020-01-01')>=181) and abs(datediff(start\_date,'2020-01-01')<=210) then '181-210 days'

when abs(datediff(start\_date,'2020-01-01')>=211) and abs(datediff(start\_date,'2020-01-01')<=240) then '211-240 days'

when abs(datediff(start\_date,'2020-01-01')>=241) and datediff(start\_date,'2020-01-01')<=270 then '241-270 days'

when datediff(start\_date,'2020-01-01')>=271 and datediff(start\_date,'2020-01-01')<=300 then '271-300 days'

when datediff(start\_date,'2020-01-01')>=301 and datediff(start\_date,'2020-01-01')<=330 then '301-330 days'

when abs(datediff(start\_date,'2020-01-01')>=331) and abs(datediff(start\_date,'2020-01-01')<=361) then '331-360 days'

else ">361 days"

end as date\_diff\_cat

from subscriptions

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

where plans.plan\_id=3

)

select count(date\_diff\_cat),date\_diff\_cat as breakdown from cte1

group by date\_diff\_cat

order by breakdown;

A screenshot of a number

Description automatically generated

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

with cte1 as

(

select price-lag(price,1) over(order by customer\_id) as price\_lag from subscriptions

inner join plans on plans.plan\_id=subscriptions.plan\_id

where year(start\_date)=2020

)

select count(price\_lag) from cte1

where price\_lag=-10;

A screenshot of a computer

Description automatically generated