use foodie_fi;

-- Based off the 8 sample customers provided in the sample from the subscriptions table,

-- write a brief description about each customer’s onboarding journey.

select customer_id, s.plan_id, plan_name, start_date

from subscriptions s

join plans p

on p.plan_id = s.plan_id;

customer_id	plan_id	plan_name	start_date
1	0	trial	2020-08-01
1	1	basic monthly	2020-08-08
2	0	trial	2020-09-20
2	3	pro annual	2020-09-27
3	0	trial	2020-01-13
3	1	basic monthly	2020-01-20
4	0	trial	2020-01-17
4	1	basic monthly	2020-01-24
4	4	churn	2020-04-21
5	0	trial	2020-08-03
5	1	basic monthly	2020-08-10
6	0	trial	2020-12-23
6	1	basic monthly	2020-12-30
6	4	churn	2021-02-26
7	0	trial	2020-02-05
7	1	basic monthly	2020-02-12
7	2	pro monthly	2020-05-22
8	0	trial	2020-06-11

-- How many customers has Foodie-Fi ever had?

select count(distinct customer_id) as cust_count

from subscriptions;

cust_count
1000

-- 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 extract(month from start_date) as month_id, monthname(start_date)as
month_name , count(customer_id) as cust_count
from subscriptions s
join plans p
on p.plan_id = s.plan_id
where plan_name = 'trial'
group by  extract(month from start_date), monthname(start_date)
order by extract(month from start_date);
```

month_id	month_name	cust_count
1	January	88
2	February	68
3	March	94
4	April	81
5	May	88
6	June	79
7	July	89
8	August	88
9	September	87
10	October	79
11	November	75
12	December	84

```
-- What plan start_date values occur after the year 2020 for our dataset?- Show the
breakdown by count of events for each plan_name
with cte_1 as (select distinct plan_id, extract(year from start_date) as years
                from subscriptions
                where extract(year from start_date) > 2020),

cte_2 as (select  p.plan_id, plan_name,count(distinct customer_id) as events
from plans p
join subscriptions s
on s.plan_id = p.plan_id
join cte_1 c1
on c1.plan_id = p.plan_id
```

where EXTRACT(YEAR FROM s.start_date) = c1.years

group by p.plan_id, plan_name)

select * from cte_2;

plan_id	plan_name	events
1	basic monthly	8
2	pro monthly	60
3	pro annual	63
4	churn	71

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

select plan_name, count(customer_id) as cust_count,

round(count(customer_id)*100/(select count(distinct customer_id) from subscriptions),1) as percentage

from plans p

join subscriptions s

on p.plan_id = s.plan_id

where plan_name = 'churn'

group by plan_name;

plan_name	cust_count	percentage
churn	307	30.7

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

with cte_1 as (select customer_id, plan_name, start_date,

lead(plan_name)over(partition by customer_id order by start_date) as leads

from plans p

join subscriptions s

on s.plan_id = p.plan_id)

select count(customer_id)as cust_count,

round(count(customer_id)/(select count(distinct customer_id) from subscriptions)*100) as percent

from cte_1 c1

where plan_name = "trial" and leads = "churn";

	cust_count	percent
▶	92	9

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

```
select  p.plan_id, plan_name, count(distinct customer_id) as cust_count,
        round(count(distinct customer_id)/ (select count(customer_id) from subscriptions) *100 , 2) as percent
from plans p
join subscriptions s
on p.plan_id = s.plan_id
where start_date <= "2020-12-31"
group by plan_id, plan_name;
```

	plan_id	plan_name	cust_count	percent
▶	0	trial	1000	37.74
	1	basic monthly	538	20.30
	2	pro monthly	479	18.08
	3	pro annual	195	7.36
	4	churn	236	8.91

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

```
select count(*) as cust_count
from
(select customer_id, plan_name, start_date,
lead(plan_name)over(partition by customer_id order by start_date) as lead_plan
from plans p
join subscriptions s
on s.plan_id = p.plan_id
where extract(year from start_date) =2020) x
where lead_plan = "pro annual";
```

	cust_count
▶	195

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

```
with cte_1 as (select customer_id,  min(start_date) as trial_date
from plans p
join subscriptions s
on s.plan_id = p.plan_id
group by customer_id),
cte_2 as (select customer_id, start_date
from plans p
join subscriptions s
on s.plan_id = p.plan_id
where plan_name = "pro annual")
select round(avg(datediff(start_date, trial_date)),2) as no_of_avg_days
from cte_2 c2
join cte_1 c1
on c1.customer_id = c2.customer_id;
```

	no_of_avg_days
▶	104.62

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

```
with cte_1 as (select customer_id, start_date, count(customer_id) as
pro_monthly_cust
from plans p
join subscriptions s
on s.plan_id = p.plan_id
where plan_name = 'pro monthly' and year(start_date) = 2020
group by customer_id, start_date),
cte_2 as (select customer_id, start_date, count(customer_id) as basic_monthly_cust
```

```
from plans p
join subscriptions s
on s.plan_id = p.plan_id
where plan_name = 'basic monthly' and year(start_date) = 2020
group by customer_id, start_date, plan_name)
select count(*) as customer_change
 from cte_1 c1
 join cte_2 c2
 on c1.customer_id = c2.customer_id
 where c2.start_date > c1.start_date;
```

	customer_change
▶	0

```
-- Can you further break down this average value into 30 day periods (i.e. 0-30 days, 31-60 days etc)
with cte_1 as (select customer_id,  min(start_date) as trial_date
from plans p
join subscriptions s
on s.plan_id = p.plan_id
group by customer_id),
cte_2 as (select customer_id, start_date
from plans p
join subscriptions s
on s.plan_id = p.plan_id
where plan_name = "pro annual"),
cte_3 as (select (FLOOR(datediff(start_date, trial_date) / 30) * 30, '-',
FLOOR(datediff( start_date, trial_date) / 30) * 30 + 30, ' days') AS period
from cte_1 c1
join cte_2 c2
on c1.customer_id = c2.customer_id
)
```

```
select period, round(avg(datediff(start_date, trial_date)),2) as no_of_avg_days
from cte_3
group by period;
-- How many customers have upgraded to an annual plan in 2020?
select s.*, p.*
from subscriptions s
join plans p
on p.plan_id  = s.plan_id;
```

customer_id	plan_id	start_date	plan_id	plan_name	price
1	0	2020-08-01	0	trial	0.00
1	1	2020-08-08	1	basic monthly	9.90
2	0	2020-09-20	0	trial	0.00
2	3	2020-09-27	3	pro annual	199.00
3	0	2020-01-13	0	trial	0.00
3	1	2020-01-20	1	basic monthly	9.90
4	0	2020-01-17	0	trial	0.00
4	1	2020-01-24	1	basic monthly	9.90
4	4	2020-04-21	4	churn	NULL
5	0	2020-08-03	0	trial	0.00
5	1	2020-08-10	1	basic monthly	9.90
6	0	2020-12-23	0	trial	0.00
6	1	2020-12-30	1	basic monthly	9.90
6	4	2021-02-26	4	churn	NULL

```
-- What is the number and percentage of customer plans after their initial free trial?
with cte_1 as (select   p.plan_id, plan_name, customer_id, start_date,
dense_rank()over(partition by customer_id order by start_date) as rnk
from plans p
join subscriptions s
on s.plan_id = p.plan_id
),
cte_2 as (select plan_name, count(customer_id) as plan_customer_count,
round(count(customer_id)/ (select count(distinct customer_id) from
subscriptions)*100,2)as percentage
from cte_1
where (rnk =1 or rnk= 2) and plan_name <> "trial"
```

```
group by plan_name)
select *
from cte_2
order by plan_customer_count desc
```

plan_name	plan_customer_count	percentage
basic monthly	546	54.60
pro monthly	325	32.50
churn	92	9.20
pro annual	37	3.70

```
-- What is the customer count and percentage breakdown of all 5 plan_name values at 2020-12-31?
select plan_name, count(customer_id) as cnt,
round(count(customer_id)/ (select count(distinct customer_id) from subscriptions)*100,2) as percent
from plans p
join subscriptions s
on p.plan_id = s.plan_id
where start_date <= '2020-12-31'
group by plan_name
```

plan_name	cnt	percent
trial	1000	100.00
basic monthly	538	53.80
pro annual	195	19.50
churn	236	23.60
pro monthly	479	47.90