

## SQL Portfolio Assignment:

**GitHub:** <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:



The screenshot shows a 'Result Grid' with a 'Filter Rows' button. The grid contains one row with the column name 'Customers' and the value '1000'.

Customers
1000

### 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:

	plan_id	month	count
▶	0	1	88
	0	2	68
	0	3	94
	0	4	81
	0	5	88
	0	6	79
	0	7	89
	0	8	88
	0	9	87
	0	10	79
	0	11	75
	0	12	84

### 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:

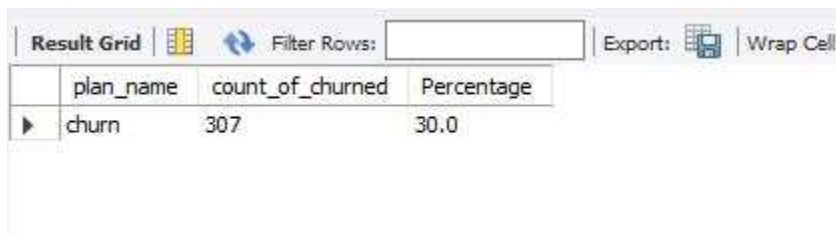
	plan_name	plan_id	year	Count_of_events
▶	basic monthly	1	2021	8
	churn	4	2021	71
	pro annual	3	2021	63
	pro monthly	2	2021	60

### 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:**

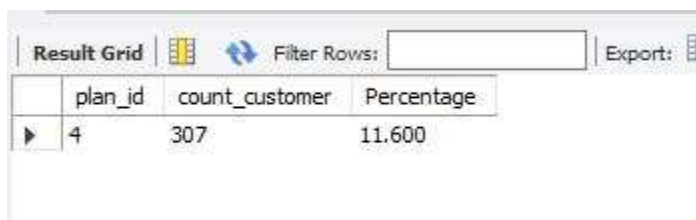
	plan_name	count_of_churned	Percentage
▶	churn	307	30.0

**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:**

	plan_id	count_customer	Percentage
▶	4	307	11.600


**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:**



	count_cust	Percentage
▶	1650	62.00

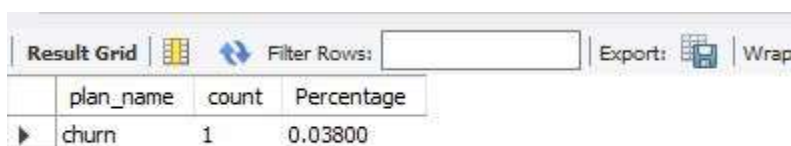
**Question No:07**

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:**



	plan_name	count	Percentage
▶	churn	1	0.03800


### 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:



	plan_name	count
▶	pro annual	195




### 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:

Result Grid			Filter Rows: <input type="text"/>	Export: 	Wrap Cell Content: <input type="checkbox"/>
	No_of_customers	AVG_Days_to_reach_annual_program			
▶	2392	116.6115			

### 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:

Result Grid			
		Filter Rows:	
		Export:	
		Wrap Cell Content:	
	period	No_of_customers	AVG_Days_to_reach_annual_program
▶	0-30 days	61	11.7541
	31-60 days	49	42.3265
	61-90 days	67	71.8209
	More than 90 days	1908	156.5074

### 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:

Result Grid		
		Filter Rows:
		Export:
		Wrap Cell Content:
	plan_id	downgraded_from_annual_to_basic
▶	1	163