SQL Case Study - Foodie-Fi

by Akif Fazal

Table of Contents

01 About Foodi-Fi

02 Case Study Objectives

03 Data Overview

04 Business Problem

05 Case Study Questions

Foodie-Fi

- Foodie-Fi is a subscription-based streaming platform dedicated to food-related content, founded in 2020 by Danny and his team.
- Foodie-Fi aims to fill the market gap for a platform focused solely on cooking shows, similar to Netflix but with a culinary focus.
- The company embraces a data-driven approach to make informed decisions and investments.

Case Study Objectives

- **Customer Insights:** Understand Foodie-Fi's customer base dynamics, including total count, churn rates, and subscription behaviors.
- **Subscription Pattern Analysis:** Examine trial plan sign-ups, plan start dates, and upgrades/downgrades to discern trends and patterns over time.
- **Retention and Conversion Assessment:** Evaluate churn rates, trial conversion rates, and post-trial plan preferences to enhance customer retention strategies and optimize subscription offerings.
- **Performance Metrics Calculation:** Calculate key performance indicators such as average time to upgrade, annual plan adoption rates, and customer plan distributions to inform strategic decision-making and business growth initiatives.

Data Overview: Foodie-Fi Subscription Database

Plans Table: Contains information about different subscription plans offered by Foodie-Fi, including plan ID, name, and price.

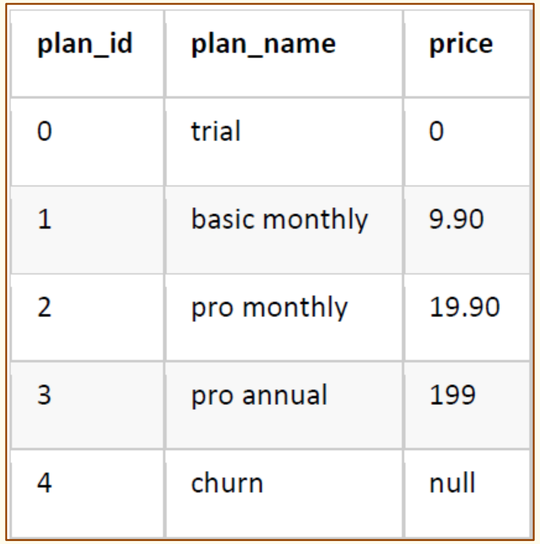
Subscriptions Table: Includes data on customer subscriptions, including customer ID, plan ID, and subscription start date.

Customer ID: Unique identifier for each customer.

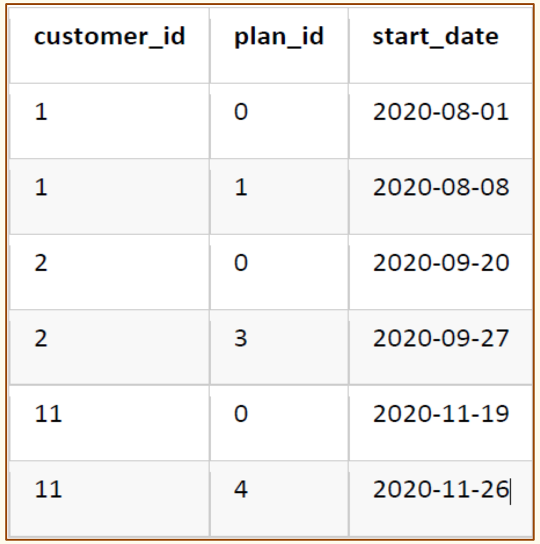
Plan ID: Identifier for each subscription plan, including trial, basic monthly, pro monthly, pro annual, and churn.

Start Date: Date when the subscription plan starts for each customer.

Churn: Indicates when a customer cancels their subscription, with a null price but access until the end of the billing period.



plan_id	plan_name	price
0	trial	0
1	basic monthly	9.90
2	pro monthly	19.90
3	pro annual	199
4	churn	null



customer_id	plan_id	start_date
1	0	2020-08-01
1	1	2020-08-08
2	0	2020-09-20
2	3	2020-09-27
11	0	2020-11-19
11	4	2020-11-26

Business Problem

Subscriber Acquisition and Retention:

- Attract and retain subscribers amidst intense competition.
- Understand customer preferences and improve content curation.
- Enhance user experience to reduce churn rates.

Data-Driven Decision-Making:

- Utilize data analytics for informed decision-making.
- Build robust data infrastructure and implement predictive analytics.
- Foster a culture of data-driven innovation for sustainable growth.

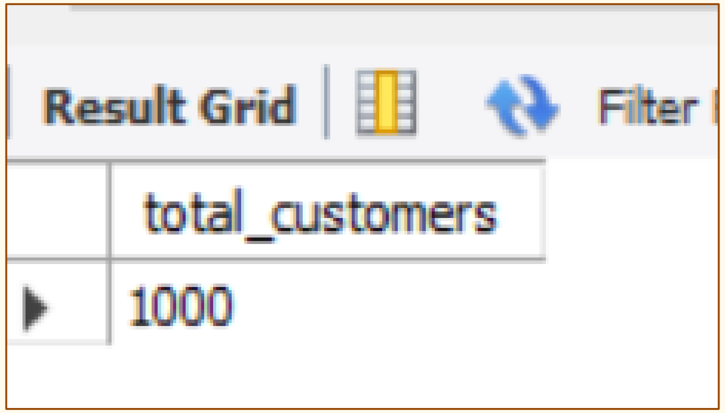
Case Study Questions..

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

SQL Query:

```
SELECT
    COUNT(DISTINCT customer_id) AS total_customers
FROM subscriptions;
```

Result:



total_customers
1000

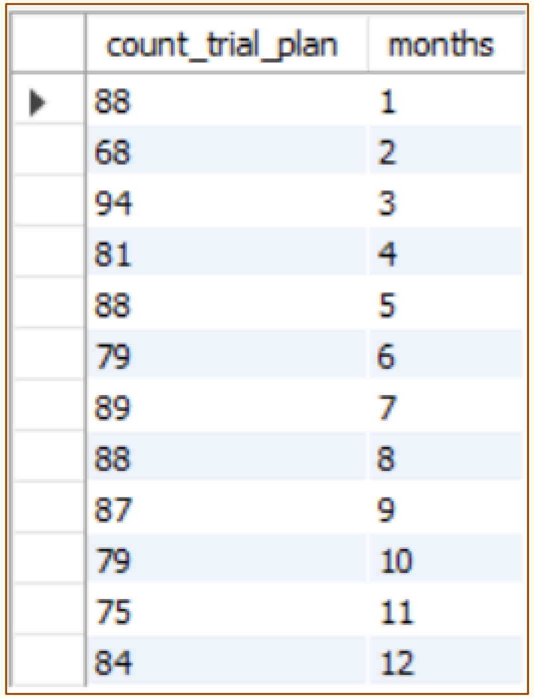
Insight: Foodie-Fi has engaged with 1000 unique customers over its operational period, suggesting a notable level of initial customer interest and adoption of the streaming service.

2. 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

SQL Query:

```
SELECT
    count(plan_id) as count_trial_plan,
    month(start_date) as months
FROM subscriptions
    GROUP BY months, plan_id
HAVING plan_id=0;
```

Result:



count_trial_plan	months
88	1
68	2
94	3
81	4
88	5
79	6
89	7
88	8
87	9
79	10
75	11
84	12

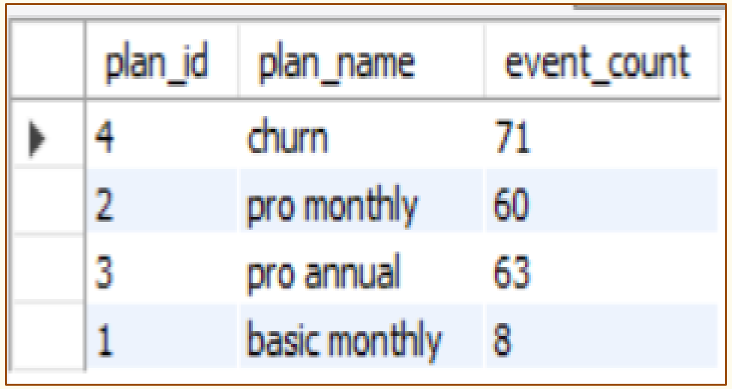
Insight: The monthly distribution of trial plan start dates reveals varying levels of customer engagement throughout the year, with peaks observed in March and August.

3. What plan start_date values occur after the year 2020 for our dataset? Show the breakdown by count of events for each plan_name.

SQL Query:

```
SELECT
        s.plan_id,
        p.plan_name,
        count(p.plan_name) as event_count
FROM subscriptions s INNER JOIN plans p
        on s.plan_id = p.plan_id
        WHERE YEAR(s.start_date) > 2020
GROUP BY s.plan_id, p.plan_name;
```

Result:



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

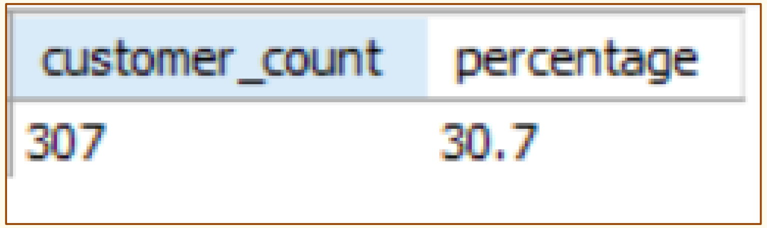
Insight: Post-2020, churn events dominate, indicating a considerable number of subscription cancellations. Pro monthly and Pro annual plans show sustained interest, while Basic monthly plan activations remain low.

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

SQL Query:

```
SELECT
	COUNT(DISTINCT customer_id) as customer_count,
	ROUND((count(DISTINCT customer_id)/(SELECT COUNT(DISTINCT customer_id) FROM subscriptions)) * 100, 1)
	AS percentage
from subscriptions
	WHERE plan_id=4;
```

Result:



customer_count	percentage
307	30.7

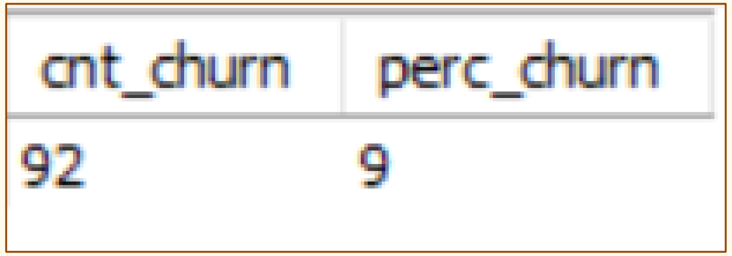
Insight: Approximately 30.7% of Foodie-Fi's customer base, totaling 307 customers, have churned from the subscription service, indicating a noteworthy churn rate requiring attention.

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

SQL Query:

```
WITH cte_churn AS (
      SELECT *, LAG(plan_id, 1)  OVER(PARTITION BY customer_id) AS prev_plan
      FROM subscriptions)
SELECT
      COUNT(prev_plan) AS cnt_churn,
      ROUND(COUNT(*) * 100/(SELECT COUNT(DISTINCT customer_id)
FROM subscriptions),0) AS perc_churnFROM cte_churn
WHERE plan_id = 4 and prev_plan = 0;
```

Result:



cnt_churn	perc_churn
92	9

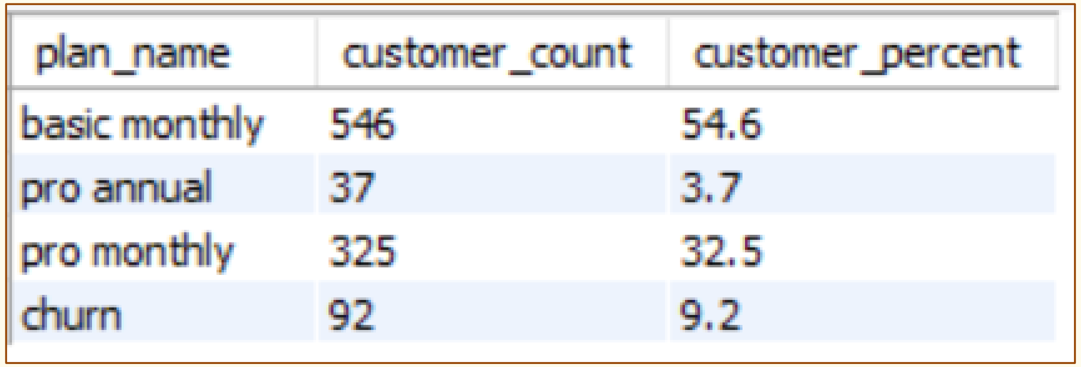
Insight: Post-2020, churn events dominate, indicating a considerable number of subscription cancellations. Pro monthly and Pro annual plans show sustained interest, while Basic monthly plan activations remain low.

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

SQL Query:

```
WITH cte_next_plan AS (
	SELECT *,
		LEAD(plan_id, 1) OVER(PARTITION BY customer_id ORDER BY plan_id) AS next_plan
FROM subscriptions)
SELECT
	next_plan,
	COUNT(*) AS num_cust,
	ROUND(COUNT(*) * 100/(SELECT COUNT(DISTINCT customer_id) FROM subscriptions),1) AS perc_next_plan
FROM cte_next_plan
	WHERE next_plan is not null and plan_id = 0
	GROUP BY next_plan;
```

Result:



plan_name	customer_count	customer_percent
basic monthly	546	54.6
pro annual	37	3.7
pro monthly	325	32.5
churn	92	9.2

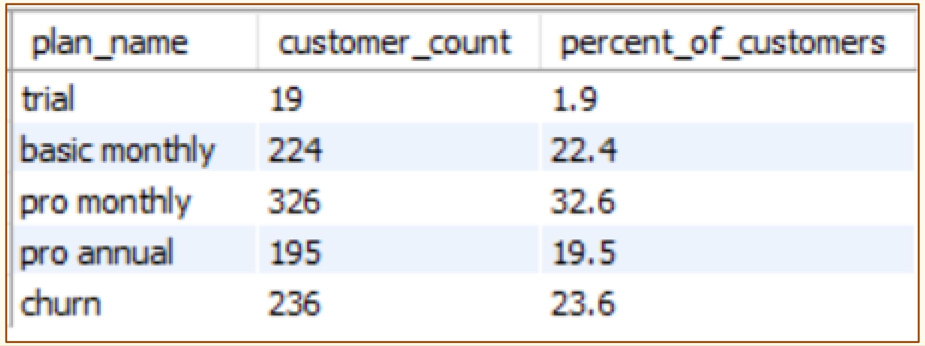
Insight: Post-free trial, 54.6% of customers opt for the Basic monthly plan, followed by 32.5% selecting the Pro monthly plan. Only 3.7% opt for the Pro annual plan, while 9.2% churn from the service.

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

SQL Query:

```
WITH My_CTE AS (
      SELECT
                 *,
                 ROW_NUMBER() OVER(PARTITION BY customer_id ORDER BY start_date DESC) as rwnmbr
      FROM subscriptions
      WHERE start_date <= '2020-12-31')
 SELECT
      plan_name,
      COUNT(customer_id) as customer_count,
      ROUND((COUNT(customer_id)/(SELECT COUNT(DISTINCT customer_id) FROM My_CTE))*100,1) as percent_of_customers
FROM My_CTE mc INNER JOIN plans as P ON mc.plan_id = P.plan_id
      WHERE rwnmbr = 1
      GROUP BY plan_name;
```

Result:



plan_name	customer_count	percent_of_customers
trial	19	1.9
basic monthly	224	22.4
pro monthly	326	32.6
pro annual	195	19.5
churn	236	23.6

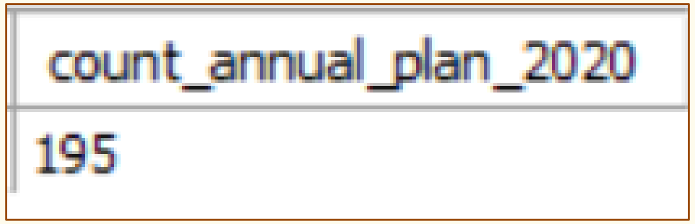
Insight: As of December 31, 2020, Pro monthly is the most subscribed plan, constituting 32.6% of customers, followed by Basic monthly at 22.4%. Pro annual and churn plans each represent 19.5% and 23.6% of the customer base, respectively, while the trial plan has the lowest subscription rate at 1.9%.

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

SQL Query:

```
SELECT
	count(*) AS count_annual_plan_2020
FROM subscriptions
WHERE Year(start_date) = 2020 and plan_id = 3;
```

Result:



count_annual_plan_2020
195

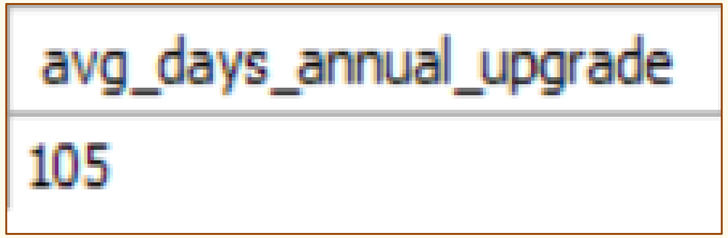
Insight: In 2020, 195 customers upgraded to an annual plan, showcasing a notable preference for long-term subscription commitments, likely driven by perceived value and cost-efficiency.

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

SQL Query:

```
WITH trail_plan AS (
      SELECT
                customer_id,
                start_date AS trail_dates
      FROM subscriptions
      WHERE plan_id=0),
annual_plan as (
      select
                customer_id,
                Start_date as annual_dates
      from subscriptions
      where plan_id=3)
SELECT
      ROUND(AVG(DATEDIFF(annual_dates, trail_dates)),0) AS avg_days_annual_upgrade
FROM annual_plan ap JOIN trail_plan tp
ON ap.customer_id = tp.customer_id;
```

Result:



avg_days_annual_upgrade
105

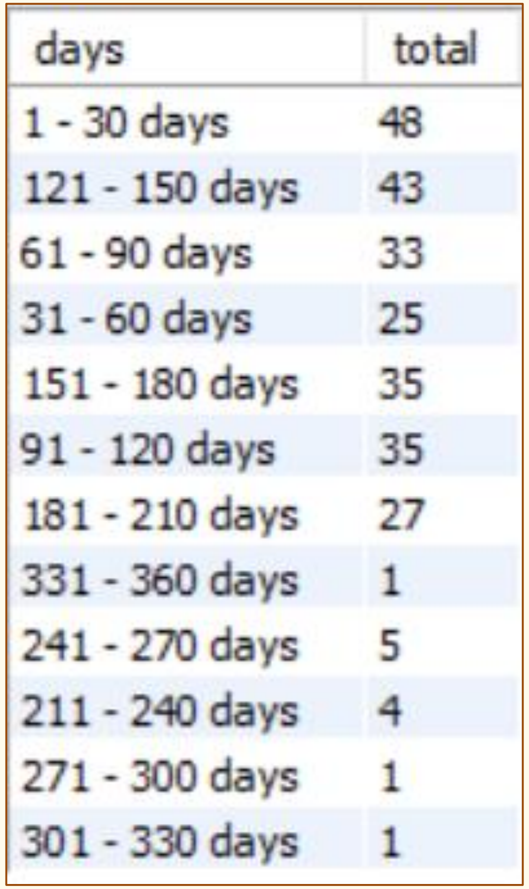
Insight: The average time for customers to upgrade from the trial plan to an annual subscription on Foodie-Fi is approximately 105 days. This indicates a considerable consideration period before committing to a long-term subscription, highlighting the importance of strategic engagement and marketing efforts to prompt timely upgrades.

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

SQL Query:

```
WITH annual_plan AS (
      SELECT customer_id, start_date AS annual_date
      FROM subscriptions
      WHERE plan_id = 3),
trial_plan AS (
      SELECT   customer_id, start_date AS trial_date
      FROM subscriptions
      WHERE plan_id = 0),
day_period AS (
      SELECT   DATEDIFF(annual_date, trial_date) AS diff
      FROM trial_plan tp LEFT JOIN annual_plan ap
      ON tp.customer_id = ap.customer_id
      WHERE annual_date is not null),
bins AS (
      SELECT   *, FLOOR(diff/30) AS bins
FROM day_period)
SELECT CONCAT((bins * 30) + 1, ' - ', (bins + 1) * 30, ' days ') AS days,
      COUNT(diff) AS total
FROM bins GROUP BY bins;
```

Result:



days	total
1 - 30 days	48
121 - 150 days	43
61 - 90 days	33
31 - 60 days	25
151 - 180 days	35
91 - 120 days	35
181 - 210 days	27
331 - 360 days	1
241 - 270 days	5
211 - 240 days	4
271 - 300 days	1
301 - 330 days	1

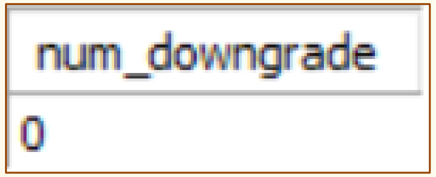
Insight: Customers show varied engagement periods when upgrading from the trial plan to an annual subscription. Notably, a significant transition occurs within the first 30 days, while engagement remains consistent between 31-180 days. Fewer upgrades occur beyond 180 days, indicating the importance of timely engagement strategies for subscription upgrades.

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

SQL Query:

```
WITH next_plan AS (
SELECT
      *,
      LEAD(plan_id, 1) OVER(PARTITION BY customer_id ORDER BY start_date, plan_id) AS plan
FROM subscriptions)
SELECT
      COUNT(DISTINCT customer_id) AS num_downgrade
FROM next_plan np LEFT JOIN plans p
      ON p.plan_id = np.plan_id
WHERE p.plan_name = 'pro monthly' AND np.plan = 1 AND start_date <= '2020-12-31';
```

Result:



num_downgrade
0

Insight: In 2020, there were no downgrades from the Pro monthly to the Basic monthly plan, indicating strong satisfaction or perceived value among Pro monthly subscribers. This underscores the importance of delivering compelling features to retain subscribers in higher-tier plans.