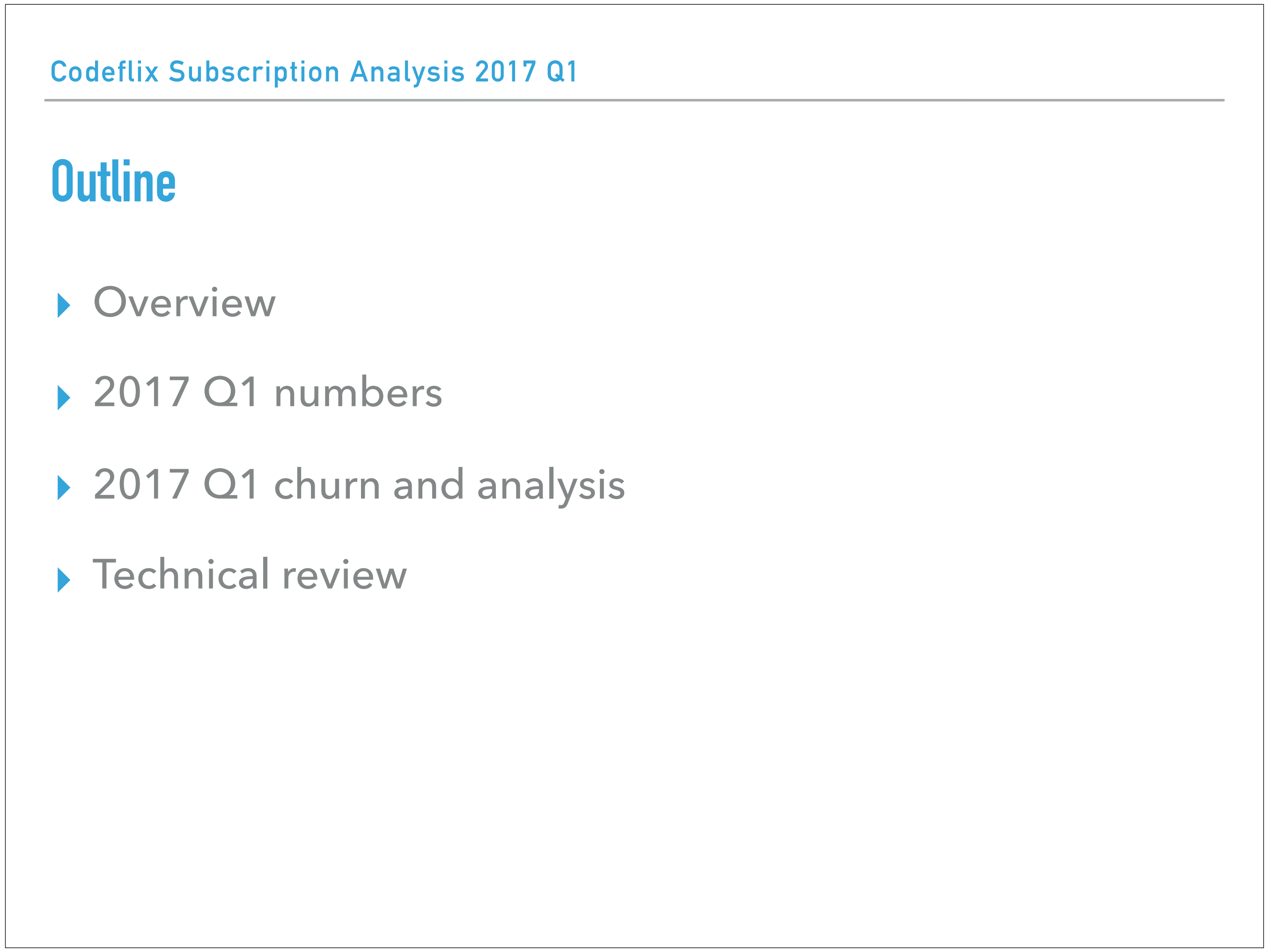
CODEFLIX

Subscription Analysis

2017 Q1

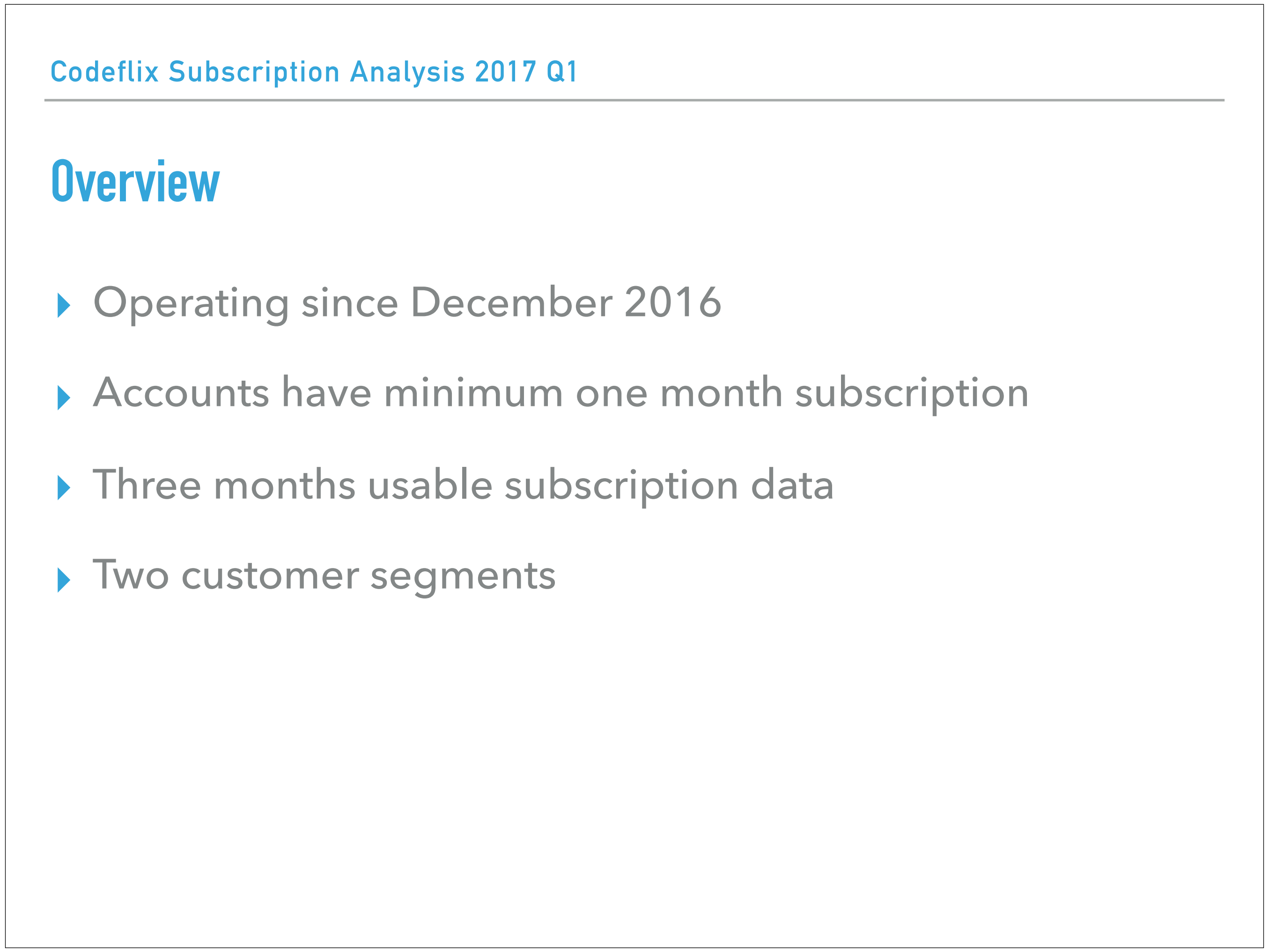
Greetings. This presentation outlines the first four months of Codeflix subscription operations.



Outline

- Overview
- 2017 Q1 numbers
- 2017 Q1 churn and analysis
- Technical review

This presentation is broken into five distinct parts. An overview of the company and the data involved. Presentation of specific numbers for 2017 Q1. Presentation of churn data and basic analysis for 2017 Q1. An optional technical review of how the information was collected and evaluated. Finally a question session to handle any outstanding queries.

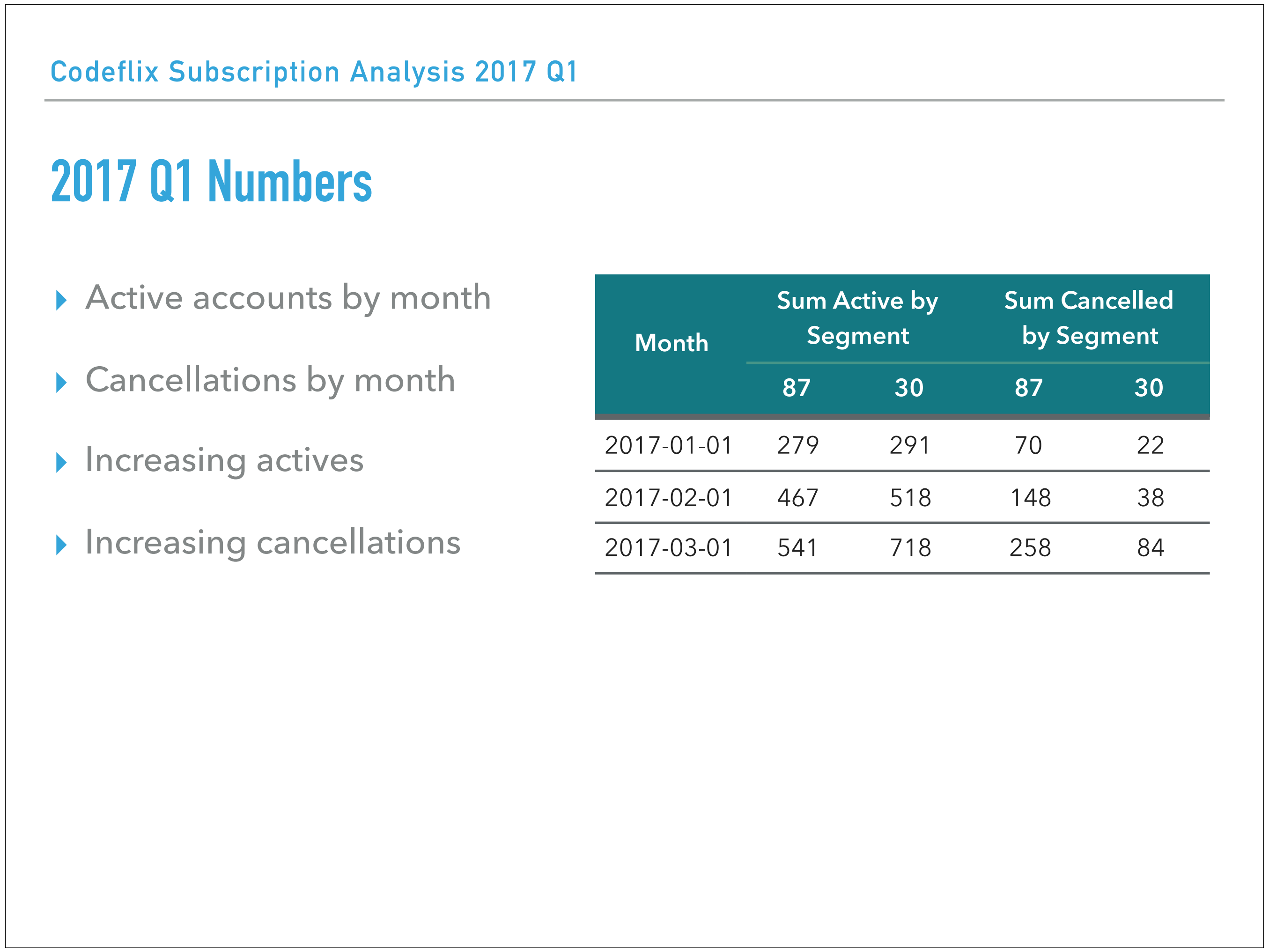


Overview

- Operating since December 2016
- Accounts have minimum one month subscription
- Three months usable subscription data
- Two customer segments

It has been four months since the launch on December 1, 2016. Since there is a minimum of a one month subscription, our overall analysis will focus on the first quarter of 2017 (January 1, 2017 to March 31, 2017) and the three months of usable data.

This analysis focuses on two segments, 30 and 87. The breakout of customers into specific segments was handled by marketing.

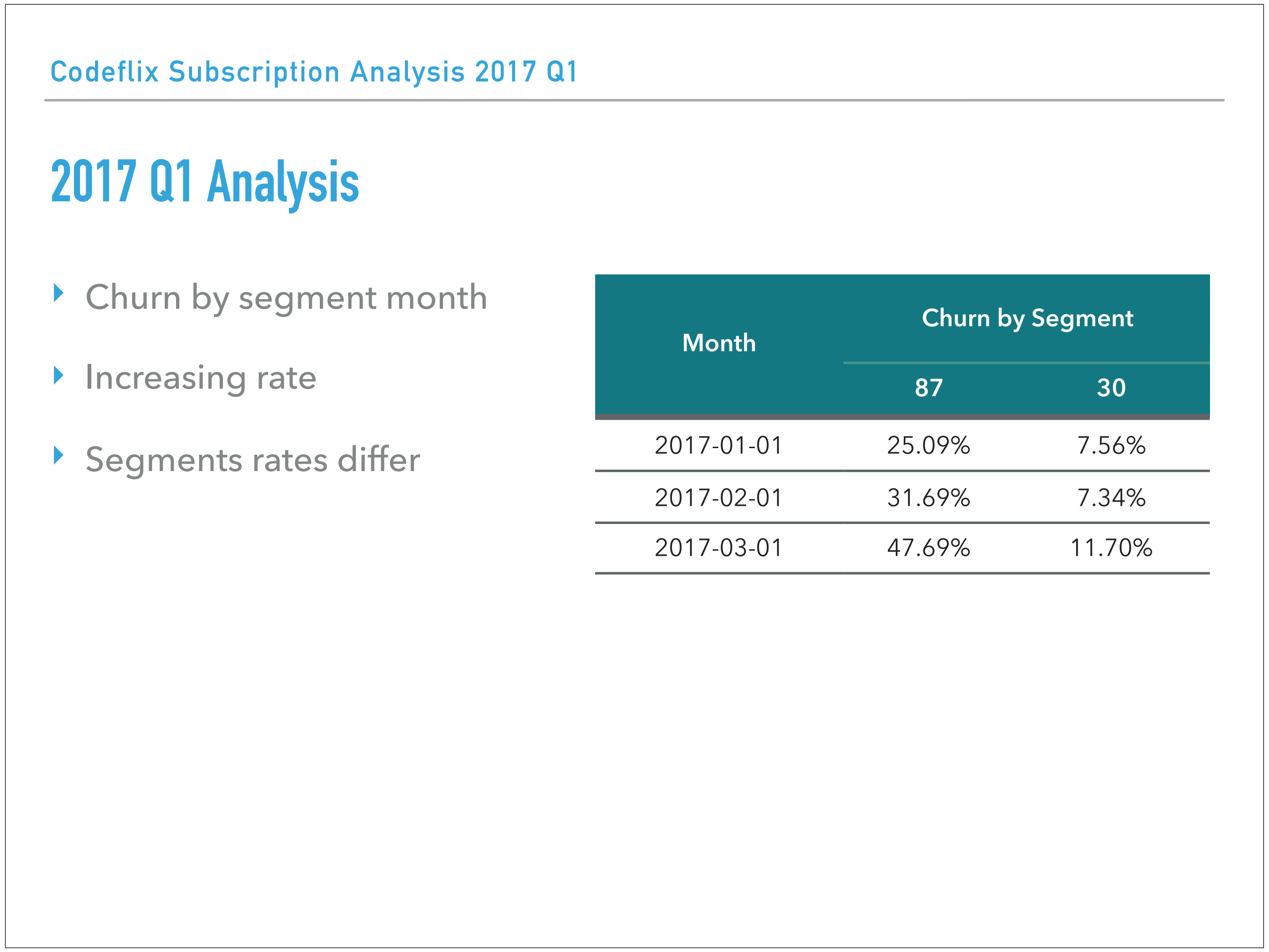


2017 Q1 Numbers

- Active accounts by month
- Cancellations by month
- Increasing actives
- Increasing cancellations

Month	Sum Active by Segment		Sum Cancelled by Segment	
	87	30	87	30
2017-01-01	279	291	70	22
2017-02-01	467	518	148	38
2017-03-01	541	718	258	84

The number of active user accounts increased throughout the quarter. This is offset somewhat by an increase in the cancellation rates, with a large surge of cancellations occurring in March in segment 87. [Note: detailed look on slide 9.]

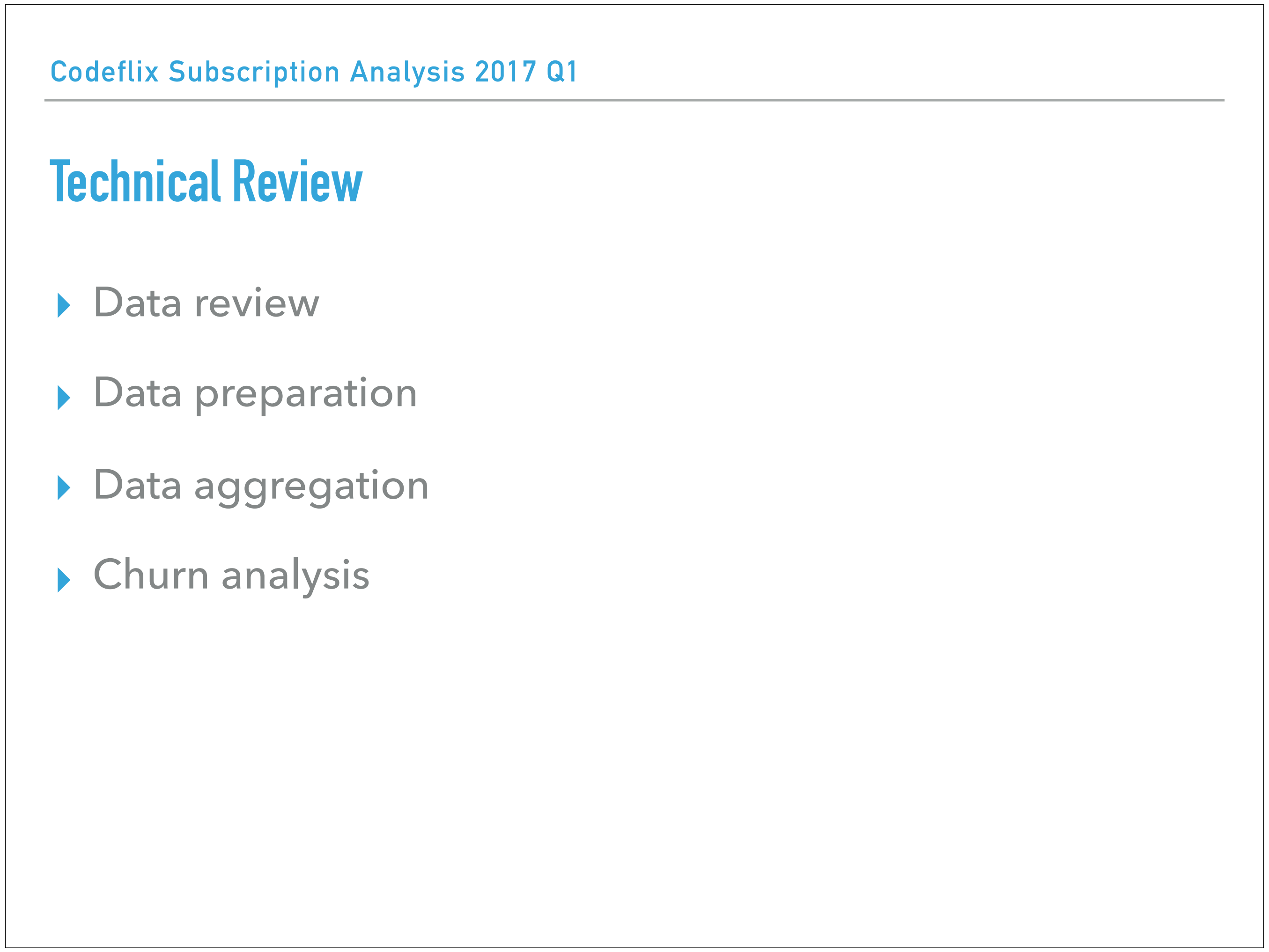


2017 Q1 Analysis

- Churn by segment month
- Increasing rate
- Segments rates differ

Month	Churn by Segment	
	87	30
2017-01-01	25.09%	7.56%
2017-02-01	31.69%	7.34%
2017-03-01	47.69%	11.70%

The overall trend in user churn is increasing. With a mild start in January, the loss rate has increased steadily over the quarter. Segment 87 has the highest churn rate and represents the majority of losses both in churn percentage and in hard numbers. Given the significant number of cancellations in segment 87 and the corresponding higher churn rate, it would be better to focus efforts on the demographics represented by segment 30 and increase that base to bring longer term stability. [Note: detailed look on slide 10.]



Technical Review

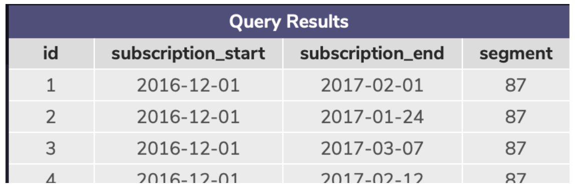
- Data review
- Data preparation
- Data aggregation
- Churn analysis

This segment covers the technical review and is used to answer specific questions or concerns.

Data review

Check first 100 rows of table

```
SELECT * FROM subscriptions LIMIT 100;
```

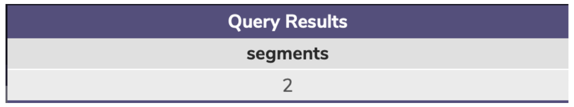


Query Results

id	subscription_start	subscription_end	segment
1	2016-12-01	2017-02-01	87
2	2016-12-01	2017-01-24	87
3	2016-12-01	2017-03-07	87
4	2016-12-01	2017-02-12	87

Check number of segments

```
SELECT COUNT(DISTINCT segment) as “segments"
FROM subscriptions;
```

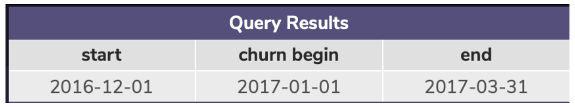


Query Results

segments
2

Check dates in table

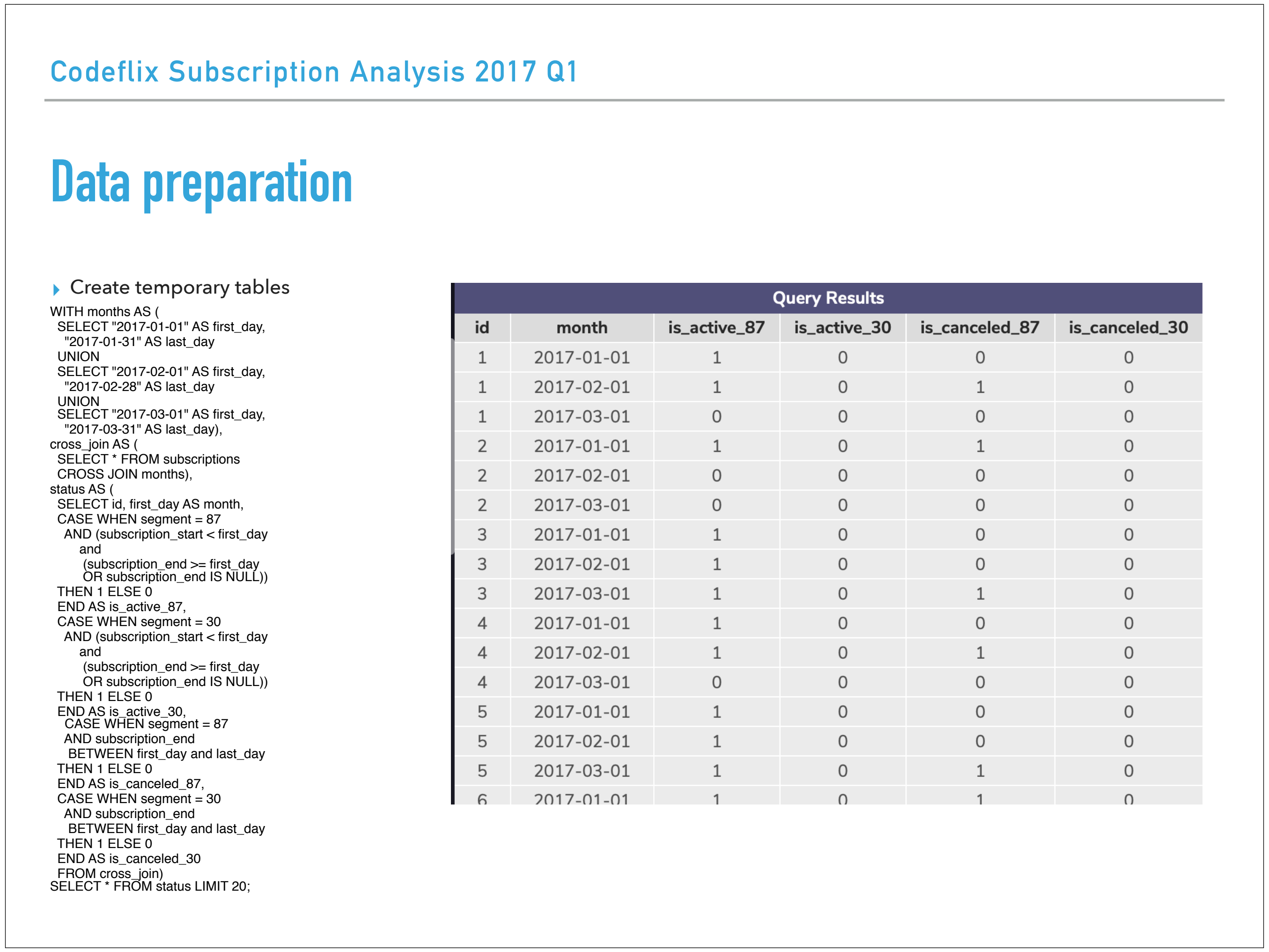
```
SELECT MIN(subscription_start) AS "start",
 MIN(subscription_end) AS "churn begin",
 MAX(subscription_end) AS "end"
FROM subscriptions;
```



Query Results

start	churn begin	end
2016-12-01	2017-01-01	2017-03-31

Initial review of table data, count of segments, and evaluation of dates involved.



Codeflix Subscription Analysis 2017 Q1

Data preparation

- Create temporary tables

```
WITH months AS (
  SELECT "2017-01-01" AS first_day,
    "2017-01-31" AS last_day
  UNION
  SELECT "2017-02-01" AS first_day,
    "2017-02-28" AS last_day
  UNION
  SELECT "2017-03-01" AS first_day,
    "2017-03-31" AS last_day),
cross_join AS (
  SELECT * FROM subscriptions
  CROSS JOIN months),
status AS (
  SELECT id, first_day AS month,
  CASE WHEN segment = 87
    AND (subscription_start < first_day
        and
         (subscription_end >= first_day
         OR subscription_end IS NULL))
  THEN 1 ELSE 0
  END AS is_active_87,
  CASE WHEN segment = 30
    AND (subscription_start < first_day
        and
         (subscription_end >= first_day
         OR subscription_end IS NULL))
  THEN 1 ELSE 0
  END AS is_active_30,
    CASE WHEN segment = 87
    AND subscription_end
     BETWEEN first_day and last_day
  THEN 1 ELSE 0
  END AS is_canceled_87,
  CASE WHEN segment = 30
    AND subscription_end
     BETWEEN first_day and last_day
  THEN 1 ELSE 0
  END AS is_canceled_30
  FROM cross_join)
SELECT * FROM status LIMIT 20;
```

Query Results

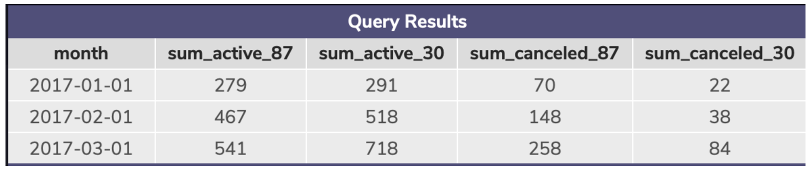
id	month	is_active_87	is_active_30	is_canceled_87	is_canceled_30
1	2017-01-01	1	0	0	0
1	2017-02-01	1	0	1	0
1	2017-03-01	0	0	0	0
2	2017-01-01	1	0	1	0
2	2017-02-01	0	0	0	0
2	2017-03-01	0	0	0	0
3	2017-01-01	1	0	0	0
3	2017-02-01	1	0	0	0
3	2017-03-01	1	0	1	0
4	2017-01-01	1	0	0	0
4	2017-02-01	1	0	1	0
4	2017-03-01	0	0	0	0
5	2017-01-01	1	0	0	0
5	2017-02-01	1	0	0	0
5	2017-03-01	1	0	1	0
6	2017-01-01	1	0	1	0

This shows a combination of several steps involving the creation of temporary tables used to sort relevant data. A month table is built and combined with the subscriptions table (cross_join). A status table is built from the cross_join table with columns added to show active or canceled status (using 0 and 1) for each segment.

Data aggregation

- Create sums from temporary tables

```
WITH months AS (…),

cross_join AS (…),

status AS (…),

status_aggregate AS (
  SELECT month,
    sum(is_active_87) AS sum_active_87,
    sum(is_active_30) AS sum_active_30,
    sum(is_canceled_87) AS sum_canceled_87,
    sum(is_canceled_30) AS sum_canceled_30
  FROM status
  GROUP BY month
)
SELECT * FROM status_aggregate;
```



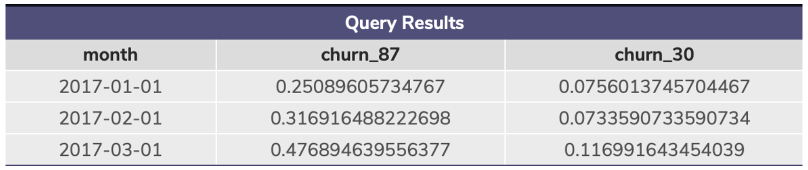
Query Results

month	sum_active_87	sum_active_30	sum_canceled_87	sum_canceled_30
2017-01-01	279	291	70	22
2017-02-01	467	518	148	38
2017-03-01	541	718	258	84

This table is an aggregation of data from the previously created temporary tables (Note: the code has been truncated for display purposes). The output shows the numbers of active subscriptions and numbers of cancellations, by segment, in a given month.

Churn analysis

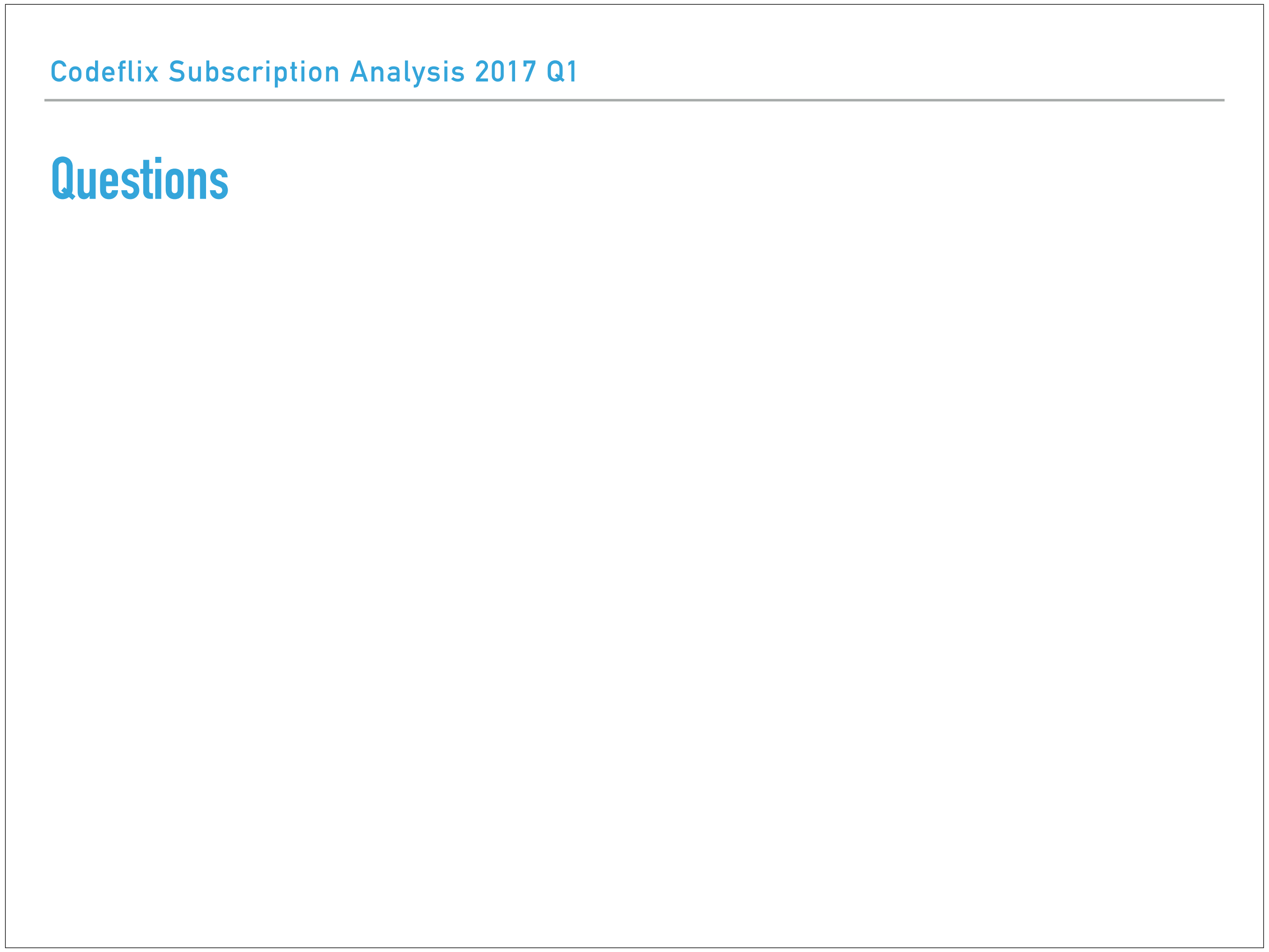
```
WITH months AS (…),

cross_join AS (…),

status AS (…),

status_aggregate AS (…)

SELECT month,
  1.0 * sum_canceled_87/sum_active_87 AS churn_87,
  1.0 * sum_canceled_30/sum_active_30 AS churn_30
FROM status_aggregate;
```



Query Results

month	churn_87	churn_30
2017-01-01	0.25089605734767	0.0756013745704467
2017-02-01	0.316916488222698	0.0733590733590734
2017-03-01	0.476894639556377	0.116991643454039

The analysis of churn data is completed using the prior temporary tables (Note: these are again truncated). The values were rounded off for the presentation, the full numbers are shown here.



Codeflix Subscription Analysis 2017 Q1

Questions

Are there any remaining questions?