**How many months has the company been operating?**

--Check how many months CodeFlix has been operational

SELECT MIN(subscription\_start) AS 'First Month',

MAX(subscription\_start) AS 'Last Month'

FROM subscriptions;

**Which months do you have enough information to calculate a churn rate?**

--Find usable data to calculate churn rate

SELECT MIN(subscription\_end) AS 'First Sub End',

MAX(subscription\_end) AS 'Last Sub End'

FROM subscriptions;

**What segments of users exist?**

--Find existing consumer segment

SELECT DISTINCT segment AS 'Segment'

FROM subscriptions;

**What is the overall churn trend since the company started?**

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 (subscription\_start < first\_day)

AND (

subscription\_end > first\_day

OR subscription\_end IS NULL

) THEN 1

ELSE 0

END AS is\_active,

CASE

WHEN subscription\_end BETWEEN first\_day AND last\_day THEN 1

ELSE 0

END AS is\_canceled

FROM cross\_join

),

status\_aggregate AS (

SELECT

month,

SUM(is\_active) AS active,

SUM(is\_canceled) AS canceled

FROM status

GROUP BY month

)

SELECT

month,

1.0 \* canceled / active AS churn\_rate

FROM status\_aggregate;

**Compare the churn rates between user segments.**

--Create temporary table of months

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

),

--Create temporary table cross\_join from subscription and months

cross\_join AS

(SELECT \*

FROM subscriptions

CROSS JOIN months),

--Create temporary table, status, from cross\_join containing id selected from cross\_join, month as an alias of first\_day, is\_active\_87, is\_active\_30

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,

--Add an is\_cancelled\_87 and an is\_cancelled\_30 column to status temporary table

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),

--Create status\_aggregate temporary table that is a SUM of the active and cancelled subscriptions for each segment, for each month

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)

--Calculate churn tate for the two segments over the three month period

SELECT

month,

1.0 \* sum\_canceled\_87 / sum\_active\_87 AS churn\_rate\_87,

1.0 \* sum\_canceled\_30 / sum\_active\_30 AS churn\_rate\_30

FROM status\_aggregate;