



# First- Churn Rates With Codeflix

Learn SQL from Scratch

Gray Palmeter

September 2018

# Table of Contents

1. Get familiar with the company
  - a. How many different segments are there?
  - b. How many months has the company been operating?
  - c. Which months do you have enough information to calculate a churn rate?
2. What is the overall churn trend since the company started?
3. Optimize the campaign budget.
  - a. Which segment of users should the company focus on expanding?

**1. Get familiar with the  
company**

# 1.1 How many segments are there?

- First, need to know the number of segments that are in the data.

id	subscription_start	subscription_end	segment
1	2016-12-01	2017-02-01	87
13	2016-12-01		30

Conclusion:

There are two segments found in the data which are segments 30 and 87.

```
SELECT *  
From subscriptions  
LIMIT 100;
```

## 1.2 How many months has the company been operating?

- Second, we need to know how many months that the company has been operating in order to understand the range of months that complete data is available so an accurate calculation of churn can be completed.

id	start	End
1991	2016-12-01	2017-03-30

Conclusion:

The company started 12/1/2016 and the data found runs to 3/30/2017.

```
SELECT id, MIN (subscription_start) AS start, MAX (subscription_start) AS End  
FROM subscriptions;
```

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

- Third, we need to know which months do we have enough available information in order to calculate churn rate.

Conclusion:

There are no subscriptions ending in December 2016, so the only months we have enough data to calculate churn for are for the months January 2017, February 2017 and March 2017.

```
SELECT *  
From subscriptions;
```

**2. What is the overall churn trend since the company has started?**

## 2. What is the overall churn trend since the company has started?

id	month	segment	is_active	is_canceled
1	2017-01-01	87	1	0
1	2017-02-01	87	0	1
1	2017-03-01	87	0	0
2	2017-01-01	87	1	1
2	2017-02-01	87	0	0
2	2017-03-01	87	0	0
3	2017-01-01	87	1	0
3	2017-02-01	87	1	0
3	2017-03-01	87	1	1

### Conclusion:

The overall churn trend is leaning towards a higher volume of churn within segment 87.

```
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,
  segment,
  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)
SELECT *
  FROM status;
```



**3. Which segment of users should the company focus on expanding?**

### 3. Which segment of users should the company focus on expanding?

churn_rate_87	churn_rate_30
35.2673529237363	8.875962789457

#### Conclusion:

The company should focus on expanding the users of segment 87 due to the extremely high volume of churn found between 1/1/2017 to 3/31/2017 of which had an extremely high volume of churn at 35.27% compared to only an 8.88 churn for the same date range for segment 30.

**3. Bonus: How would I  
modify the code to support  
a large number of  
segments?**

### 3. How would I modify the code to support a large number of segments?

Answer:

When writing a query contains a large number of hard coded statements, it is extremely important to display each statement after they have been written to make sure that there are no errors before proceeding to writing the next statement. Not displaying each statement after they are written can waste many man hours to go through a completed query with numerous hard coded statements in order to possibly find only 1 error that could have been avoided if each statement was displayed after they were written.