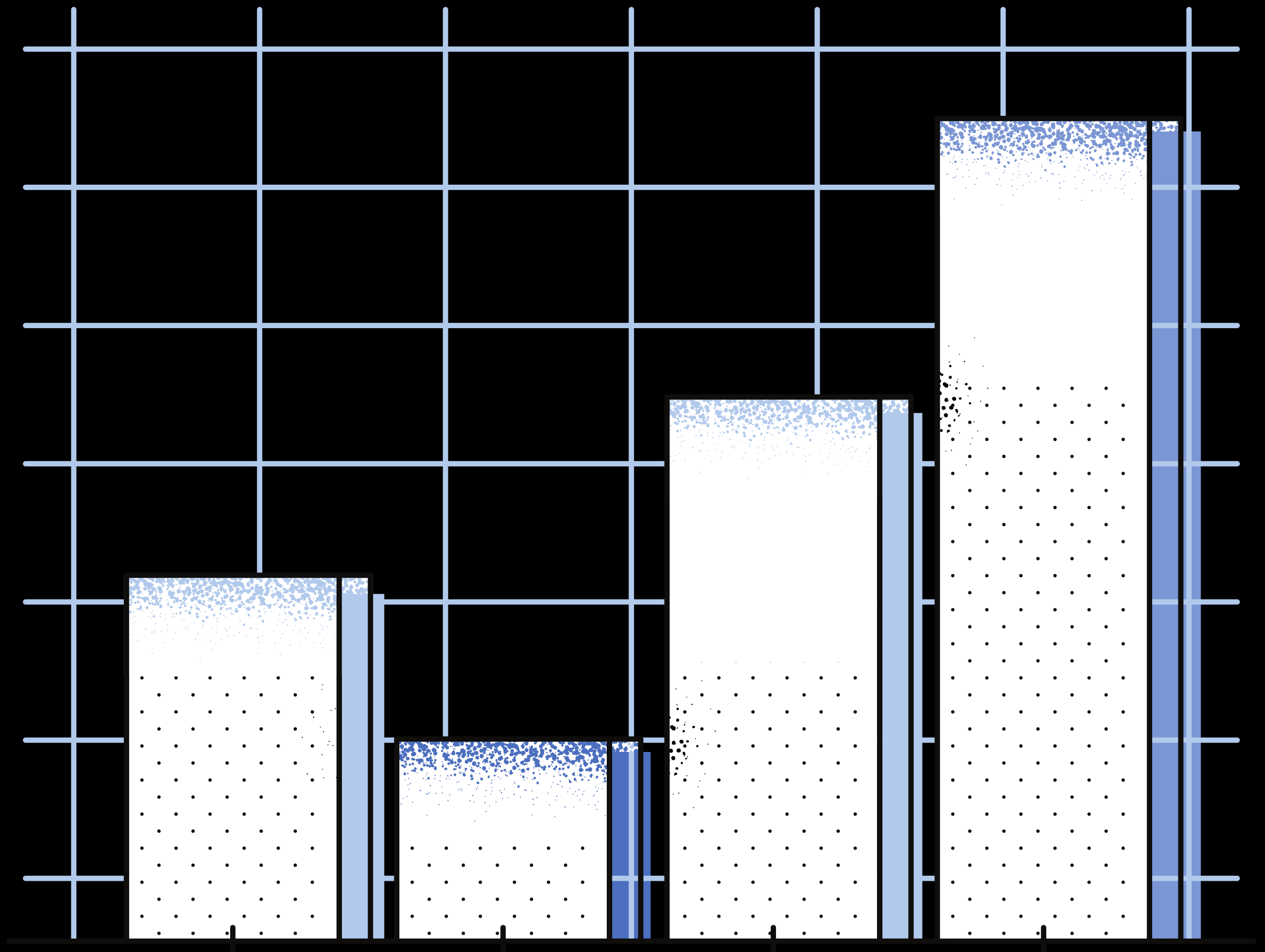
FULL STACK DATA ANALYTICS : WEEK 4-5

SQL



Anang Hendro Wibowo - Section Barcelona

dataset overview

San Fransisco Bikeshare

used in Question 1-4 of Intermediate assignment and Question 1 of Advanced assignment

- bikeshare_regions
- bikeshare_stations_info
- bikeshare_stations_status
- bikeshare_trips

Hacker News

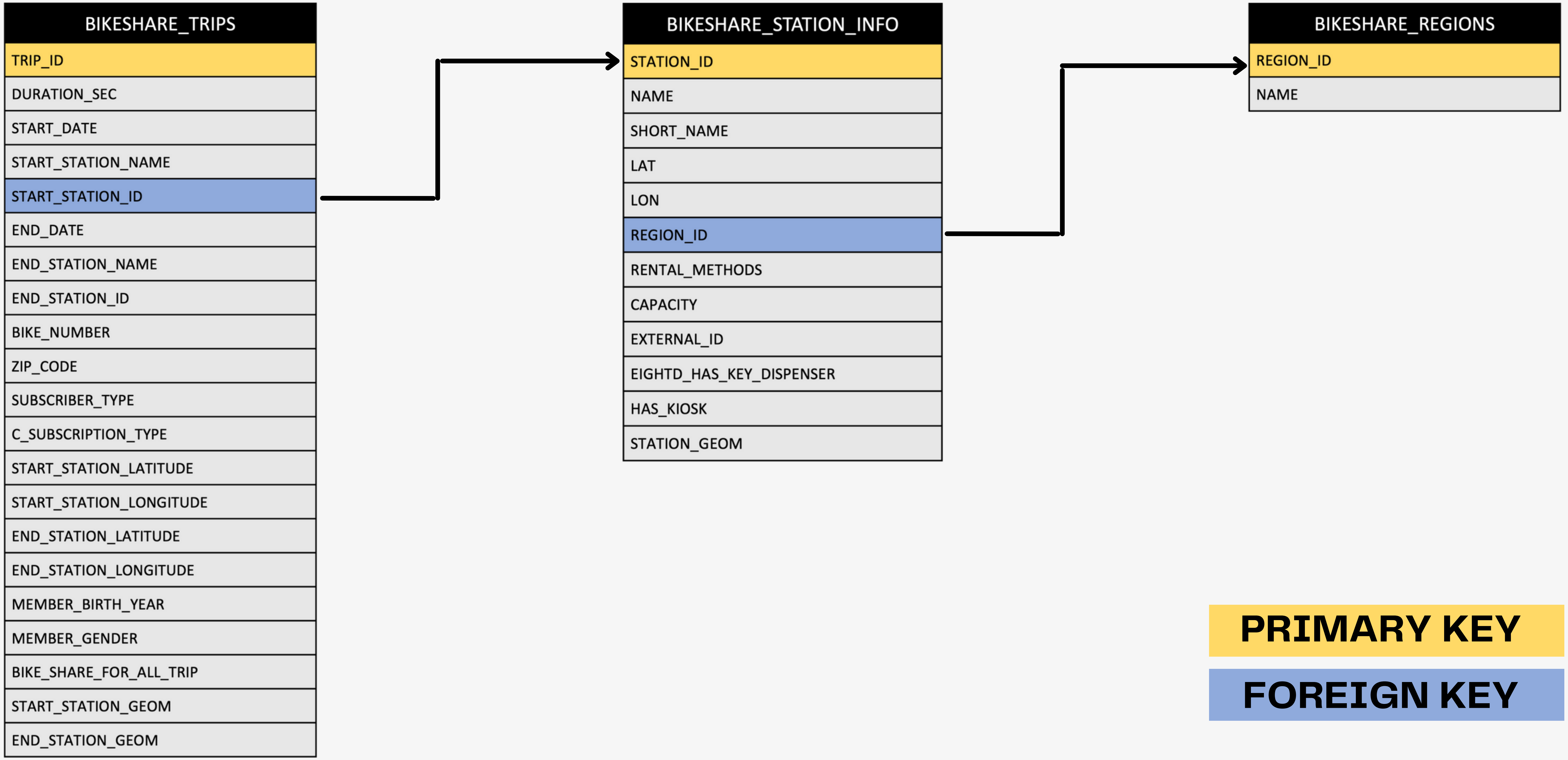
used in Question 2 in Advanced assignment

- comments
- full
- stories

tool

Google Big Query

ERD of San Fransisco Bikeshare



BIKESHARE_TRIPS
TRIP_ID
DURATION_SEC
START_DATE
START_STATION_NAME
START_STATION_ID
END_DATE
END_STATION_NAME
END_STATION_ID
BIKE_NUMBER
ZIP_CODE
SUBSCRIBER_TYPE
C_SUBSCRIPTION_TYPE
START_STATION_LATITUDE
START_STATION_LONGITUDE
END_STATION_LATITUDE
END_STATION_LONGITUDE
MEMBER_BIRTH_YEAR
MEMBER_GENDER
BIKE_SHARE_FOR_ALL_TRIP
START_STATION_GEOM
END_STATION_GEOM

BIKESHARE_STATION_INFO
STATION_ID
NAME
SHORT_NAME
LAT
LON
REGION_ID
RENTAL_METHODS
CAPACITY
EXTERNAL_ID
EIGHTD_HAS_KEY_DISPENSER
HAS_KIOSK
STATION_GEOM

BIKESHARE_REGIONS
REGION_ID
NAME

PRIMARY KEY

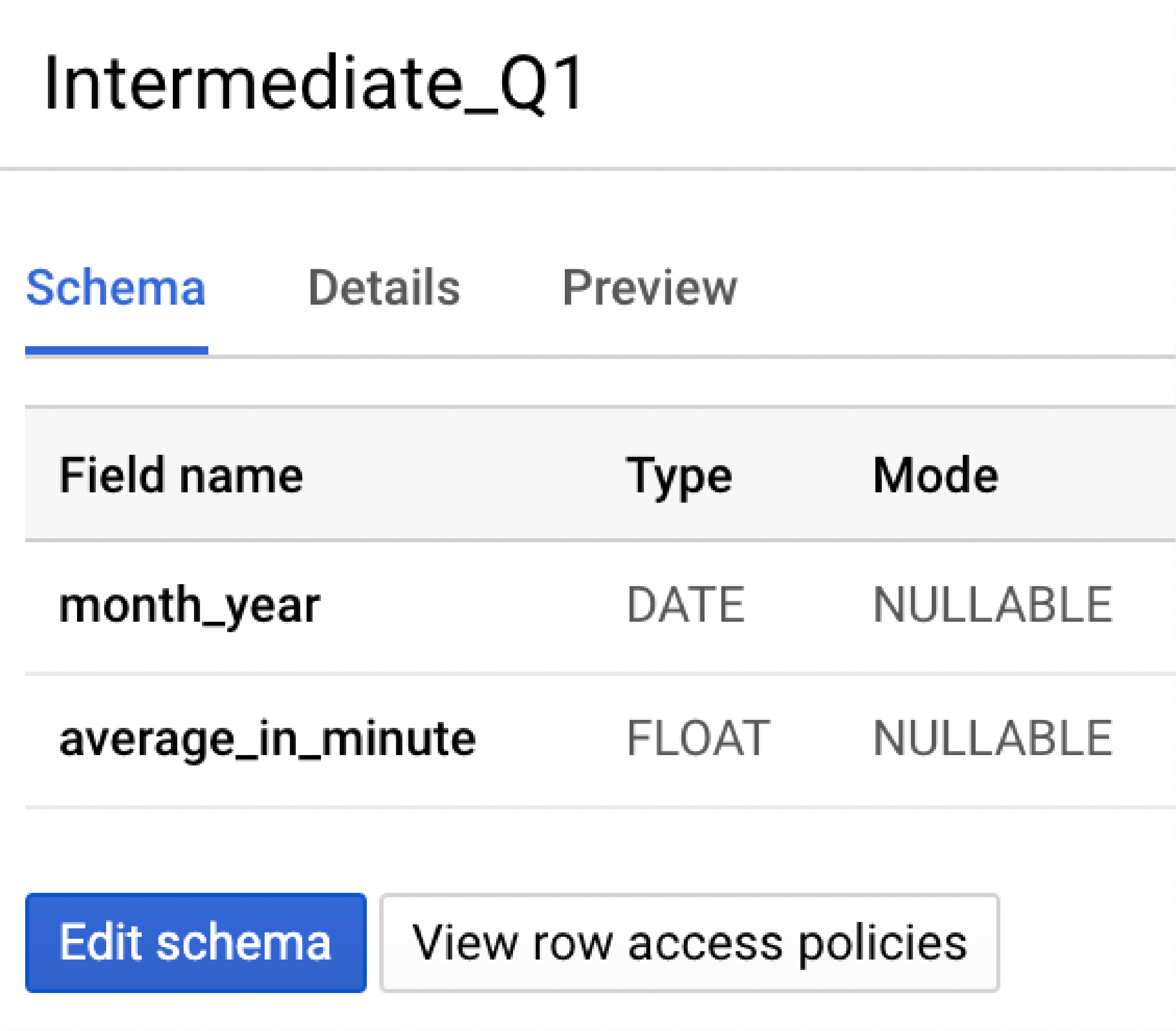
FOREIGN KEY

Intermediate Assignment

FULL STACK DATA ANALYTICS | WEEK 4-5

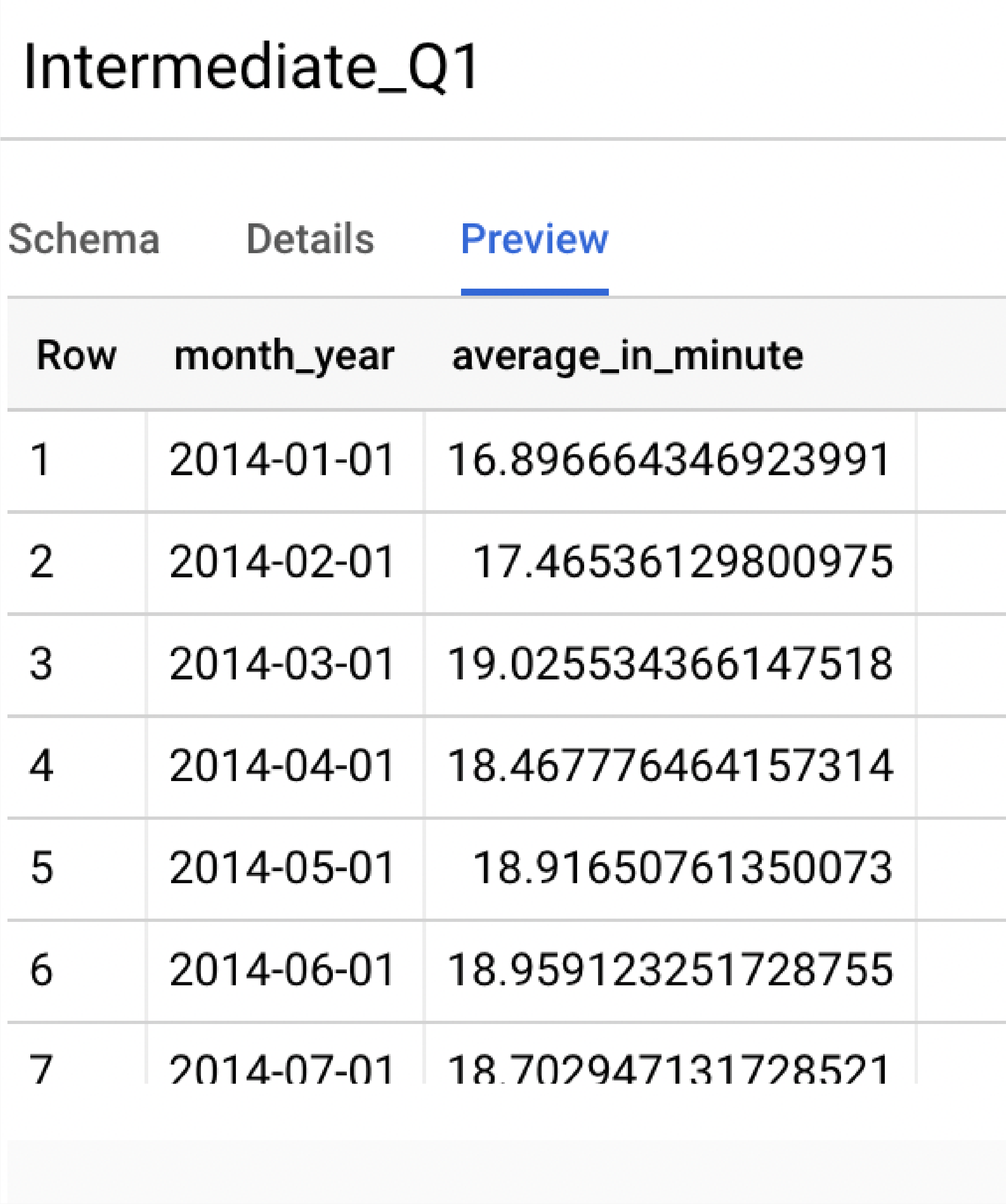
Question 1 : Table and Schema

Create a query to get average amount of duration (in minutes) per month (2014-2017)



Intermediate_Q1

Field name	Type	Mode
month_year	DATE	NULLABLE
average_in_minute	FLOAT	NULLABLE



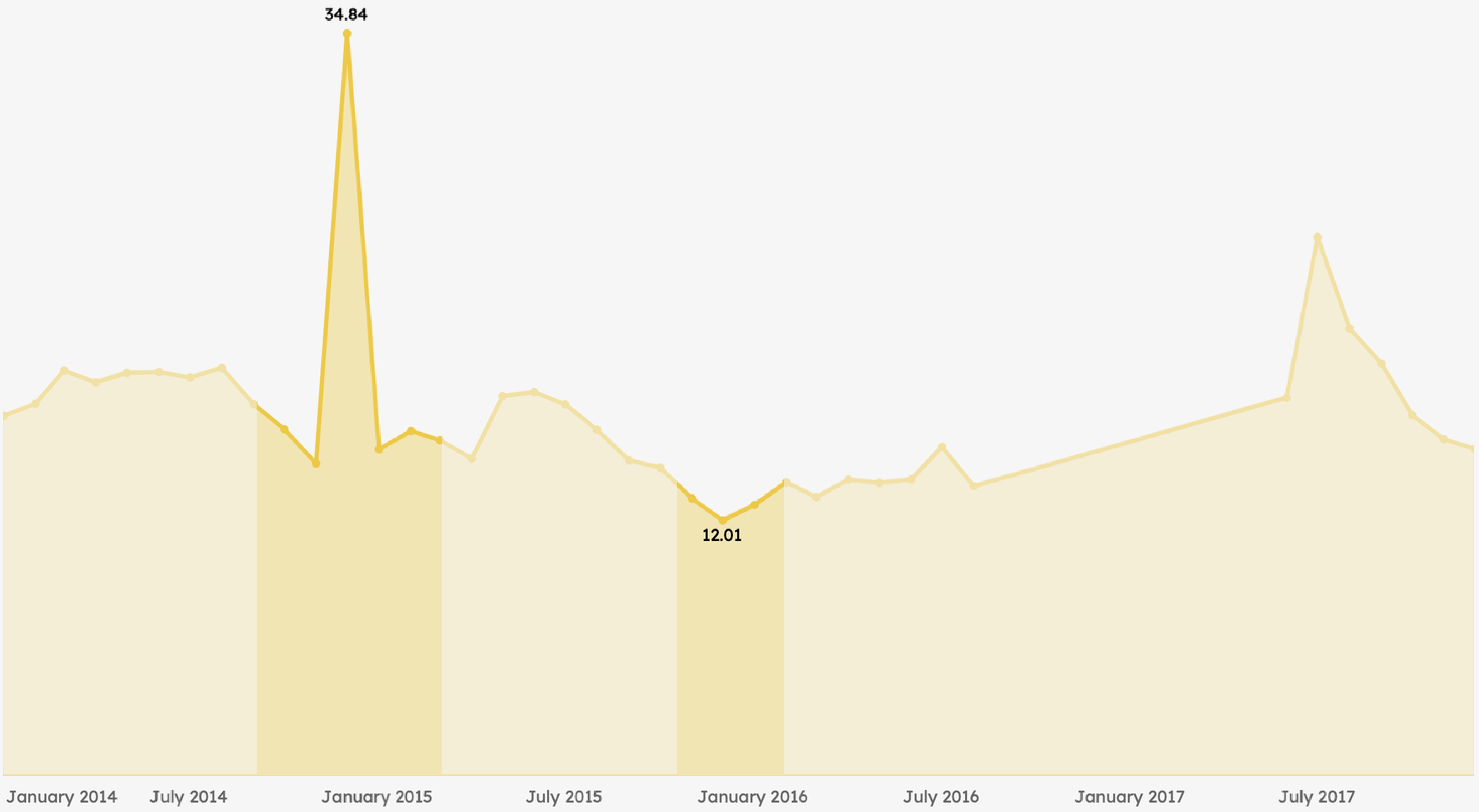
Intermediate_Q1

Row	month_year	average_in_minute
1	2014-01-01	16.896664346923991
2	2014-02-01	17.46536129800975
3	2014-03-01	19.025534366147518
4	2014-04-01	18.467776464157314
5	2014-05-01	18.91650761350073
6	2014-06-01	18.959123251728755
7	2014-07-01	18.702947131728521

Question 1 : Syntax

```
SELECT
 DATE(DATE_TRUNC(start_date,MONTH)) AS month_year,
 AVG(duration_sec/60) AS average_in_minute
FROM
 ` bigquery-public data.san_francisco_bikeshare.bikeshare_trips`
WHERE
 start_date BETWEEN '2014-01-01' AND '2017-12-31'
GROUP BY 1
ORDER BY 1
```

Question 1 : Visualization and Insight

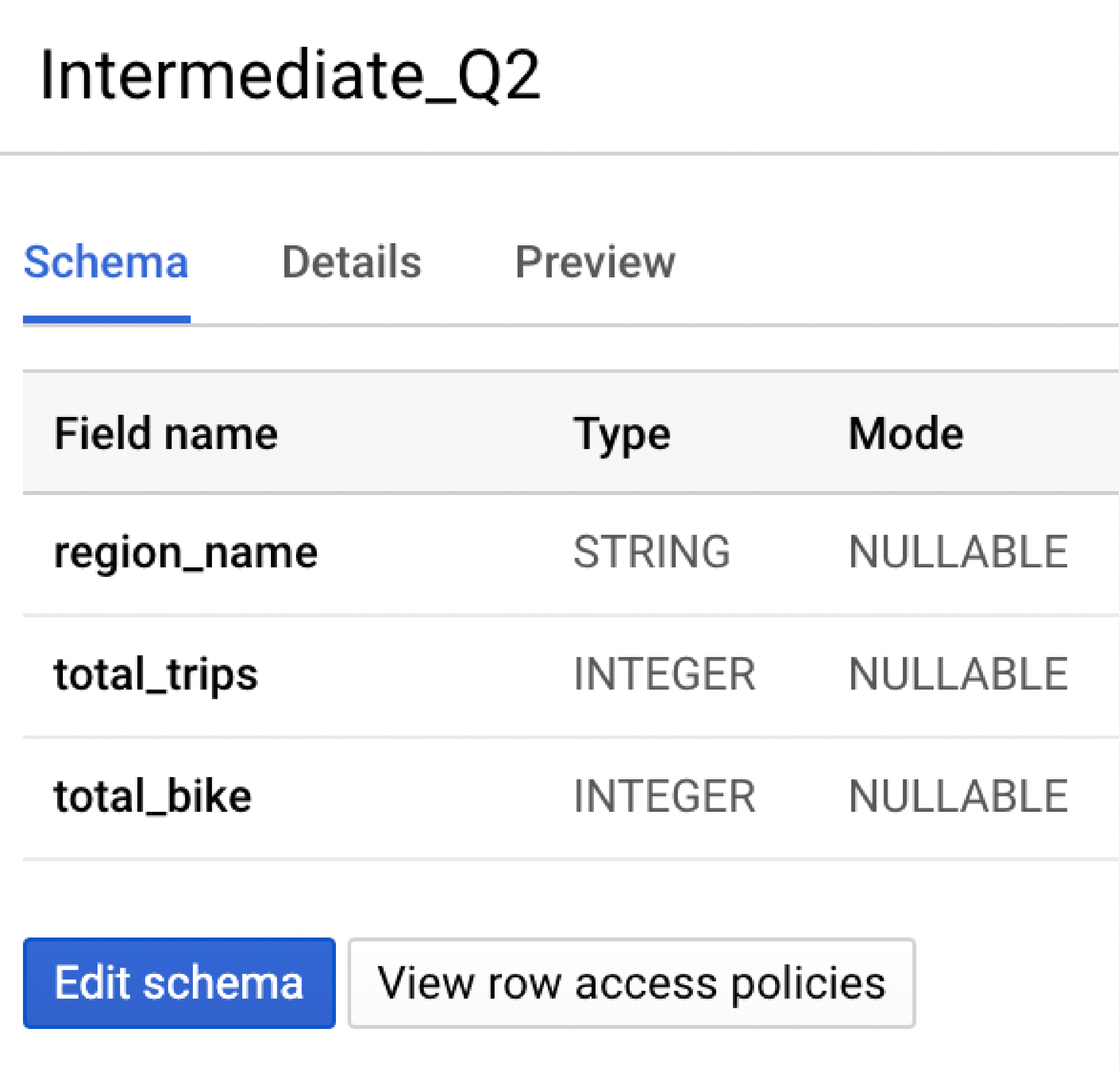


- The highest average of trips (in minutes) was shown in **December 2014** (~34 minutes)

- The lowest average of trips (in minutes) was shown in **December 2015** (~12 minutes)

Question 2 : Table and Schema

Create a query to get total trips and total number of unique bikes grouped by region name



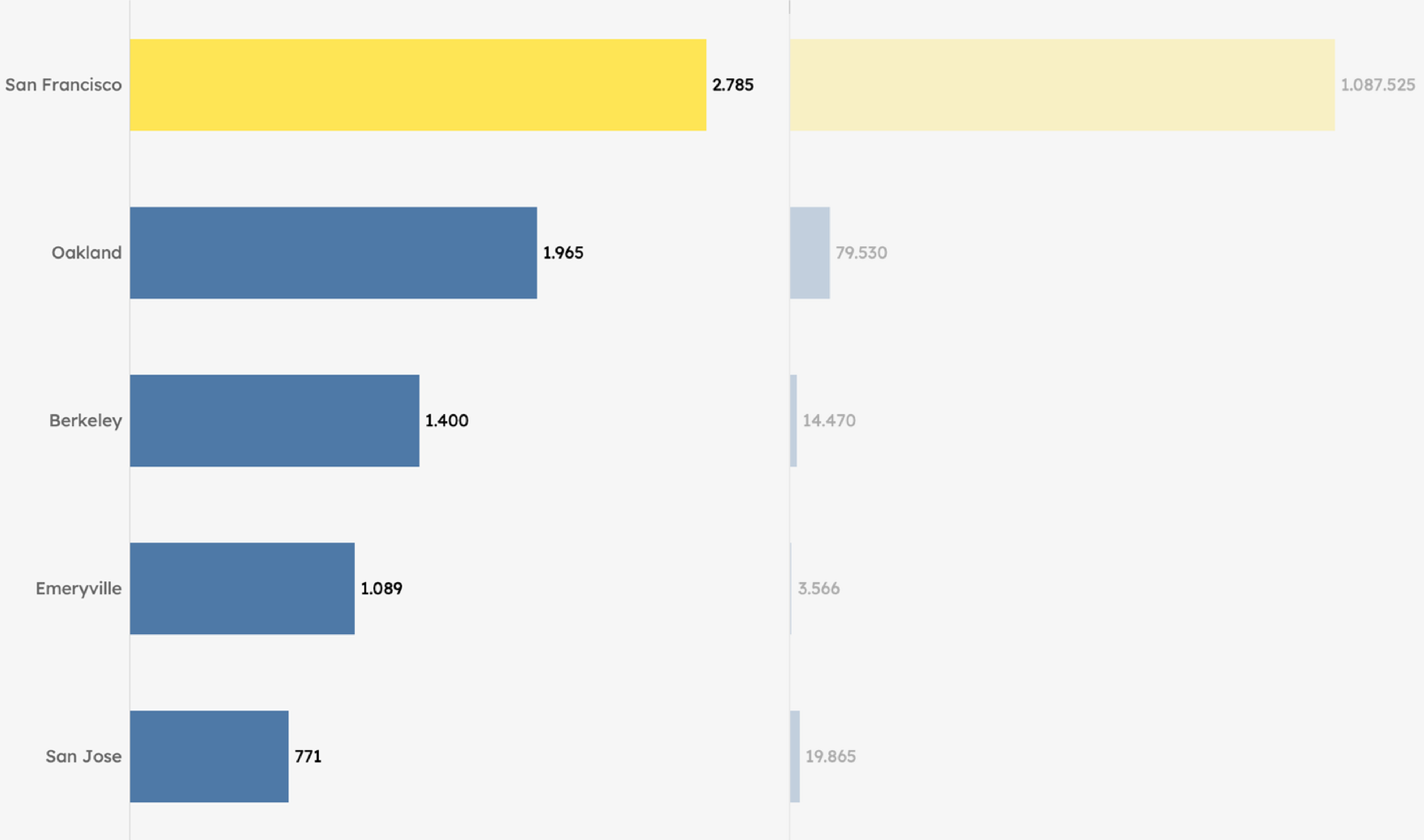
Field name	Type	Mode
region_name	STRING	NULLABLE
total_trips	INTEGER	NULLABLE
total_bike	INTEGER	NULLABLE

region_name	total_trips	total_bike
Berkeley	14470	1400
Emeryville	3566	1089
Oakland	79530	1965
San Francisco	1087525	2785
San Jose	19865	771

Question 2 : Syntax

```
SELECT
  regions.name AS region_name,
  COUNT(DISTINCT(trip_id)) AS total_trips,
  COUNT(DISTINCT(bike_number)) AS total_bike
FROM
  `bigquery-public-data.san_francisco_bikeshare.bikeshare_trips` trips
INNER JOIN
  `bigquery-public-data.san_francisco_bikeshare.bikeshare_station_info` info
ON
  trips.start_station_id = info.station_id
INNER JOIN
  `bigquery-public-data.san_francisco_bikeshare.bikeshare_regions` regions
ON
  info.region_id=regions.region_id
WHERE
  trips.start_date BETWEEN '2014-01-01' AND '2017-12-31'
GROUP BY 1
ORDER BY 1
```

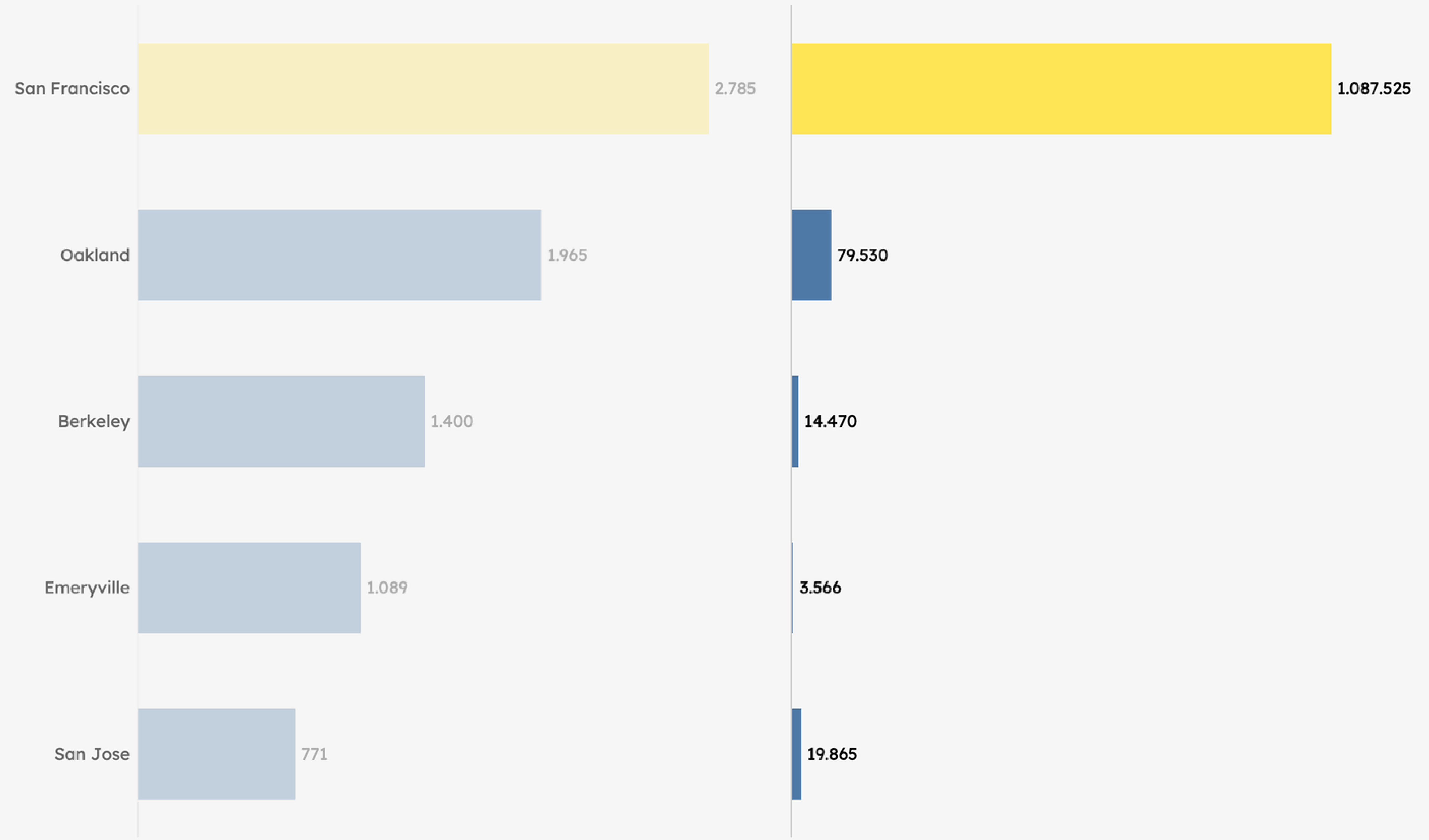
Question 2 : Visualization and Insight



- The region with the highest total trips from 2014 to 2017 is **San Fransisco** with 1087525 (90,3%)

- The region with the lowest total trips from 2014 to 2017 is **Emeryville** with 3566 (0,3%)

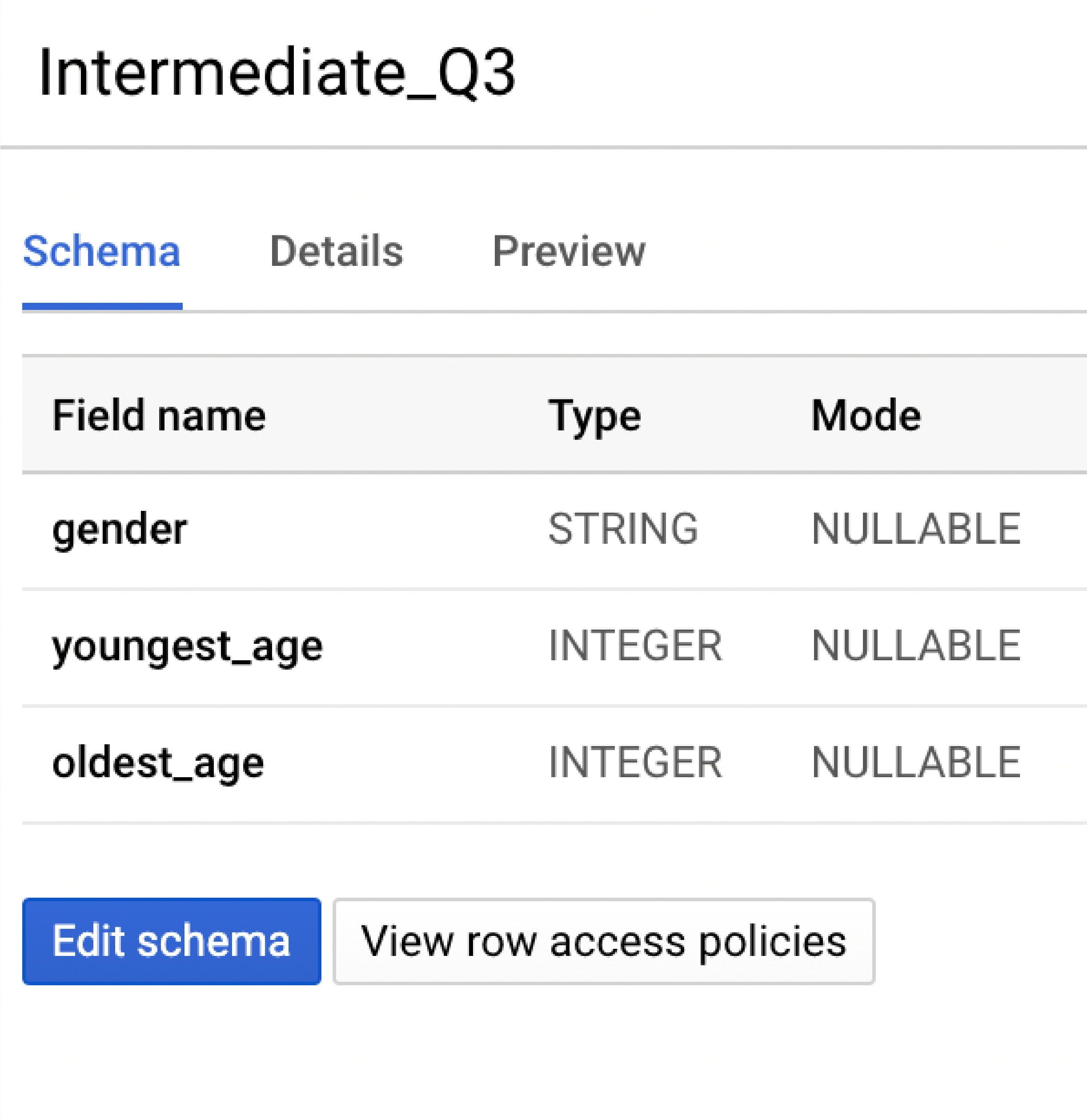
Question 2 : Visualization and Insight



- The region with the highest total bike from 2014 to 2017 is **San Fransisco** with 2785 (34,8%)
- The region with the lowest total bike from 2014 to 2017 is San Jose 771 (9,6%)

Question 3 : Table and Schema

Find the youngest and oldest age of the members for each gender (assume the present year is 2022)



Intermediate_Q3

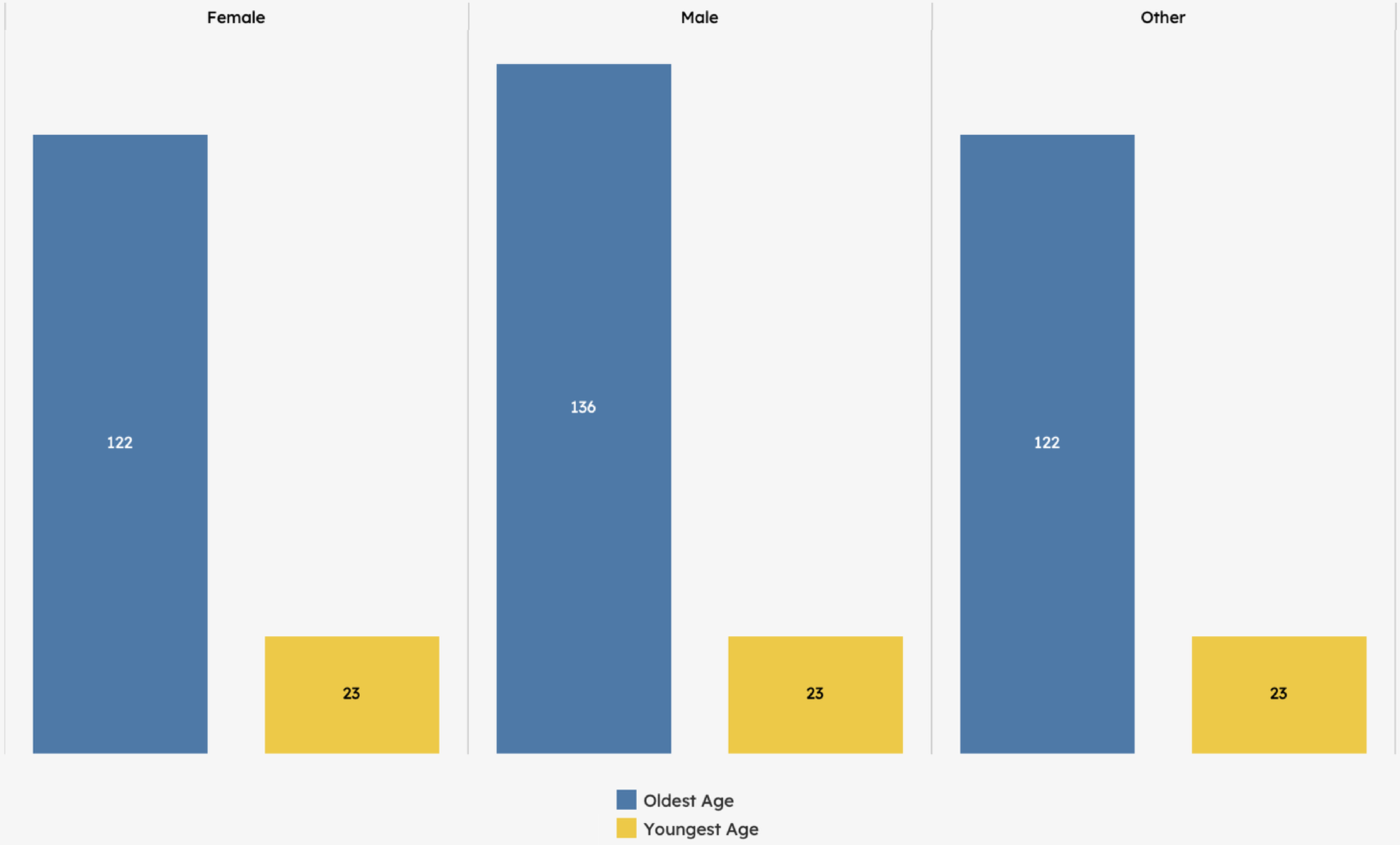
Field name	Type	Mode
gender	STRING	NULLABLE
youngest_age	INTEGER	NULLABLE
oldest_age	INTEGER	NULLABLE

gender	youngest_age	oldest_age
Female	23	122
Male	23	136
Other	23	122

Question 3 : Syntax

```sql
SELECT
  DISTINCT(member_gender) AS gender,
  MIN(2022-member_birth_year) OVER (PARTITION BY member_gender) AS youngest_age,
  MAX(2022-member_birth_year) OVER (PARTITION BY member_gender) AS oldest_age
FROM
  ` bigquery-public-data.san_francisco_bikeshare.bikeshare_trips`
WHERE
  start_date BETWEEN '2014-01-01'
  AND '2017-12-31'
  AND member_gender IS NOT NULL
ORDER BY 1
```

Question 3 : Visualization and Insight

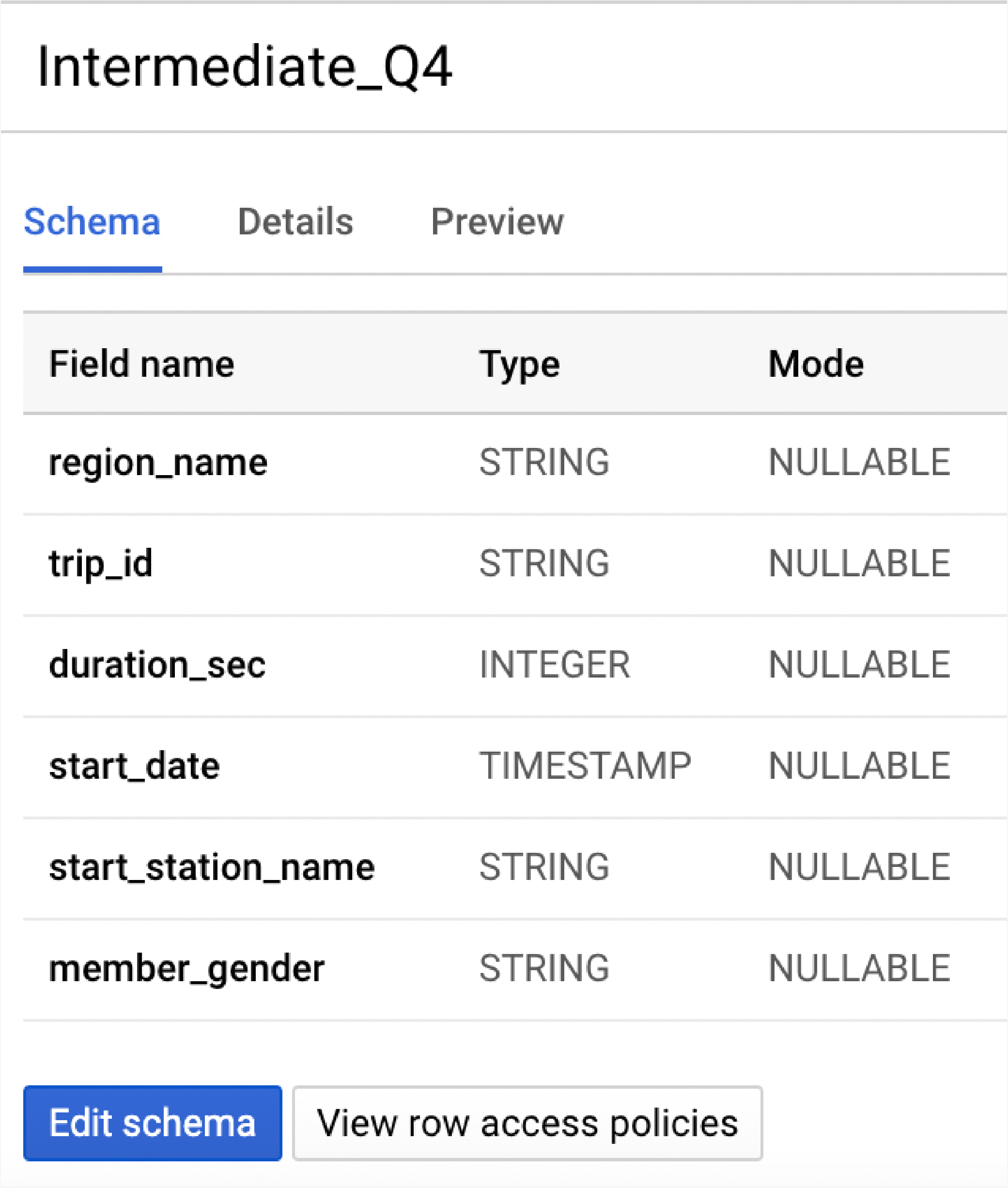


- The youngest age of female users is 23 and the oldest is 122

- The youngest age of male users is 23 and the oldest is 136

Question 4 : Table and Schema

Get the latest detailed trip in each region



Intermediate_Q4

Field name	Type	Mode
region_name	STRING	NULLABLE
trip_id	STRING	NULLABLE
duration_sec	INTEGER	NULLABLE
start_date	TIMESTAMP	NULLABLE
start_station_name	STRING	NULLABLE
member_gender	STRING	NULLABLE

region_name	trip_id	duration_sec	start_date	start_station_name	member_gender
Berkeley	12832017123023081100	380	2017-12-30 23:08:11	North Berkeley BART Station	Male
Emeryville	35882017123022082200	1258	2017-12-30 22:08:22	Stanford Ave at Hollis St	Male
Oakland	29272017123023190000	232	2017-12-30 23:19:00	19th Street BART Station	Male
San Francisco	16422017123023461300	3456	2017-12-30 23:46:13	Market St at Franklin St	Male
San Jose	4542017123021551700	234	2017-12-30 21:55:17	San Jose Diridon Station	Male

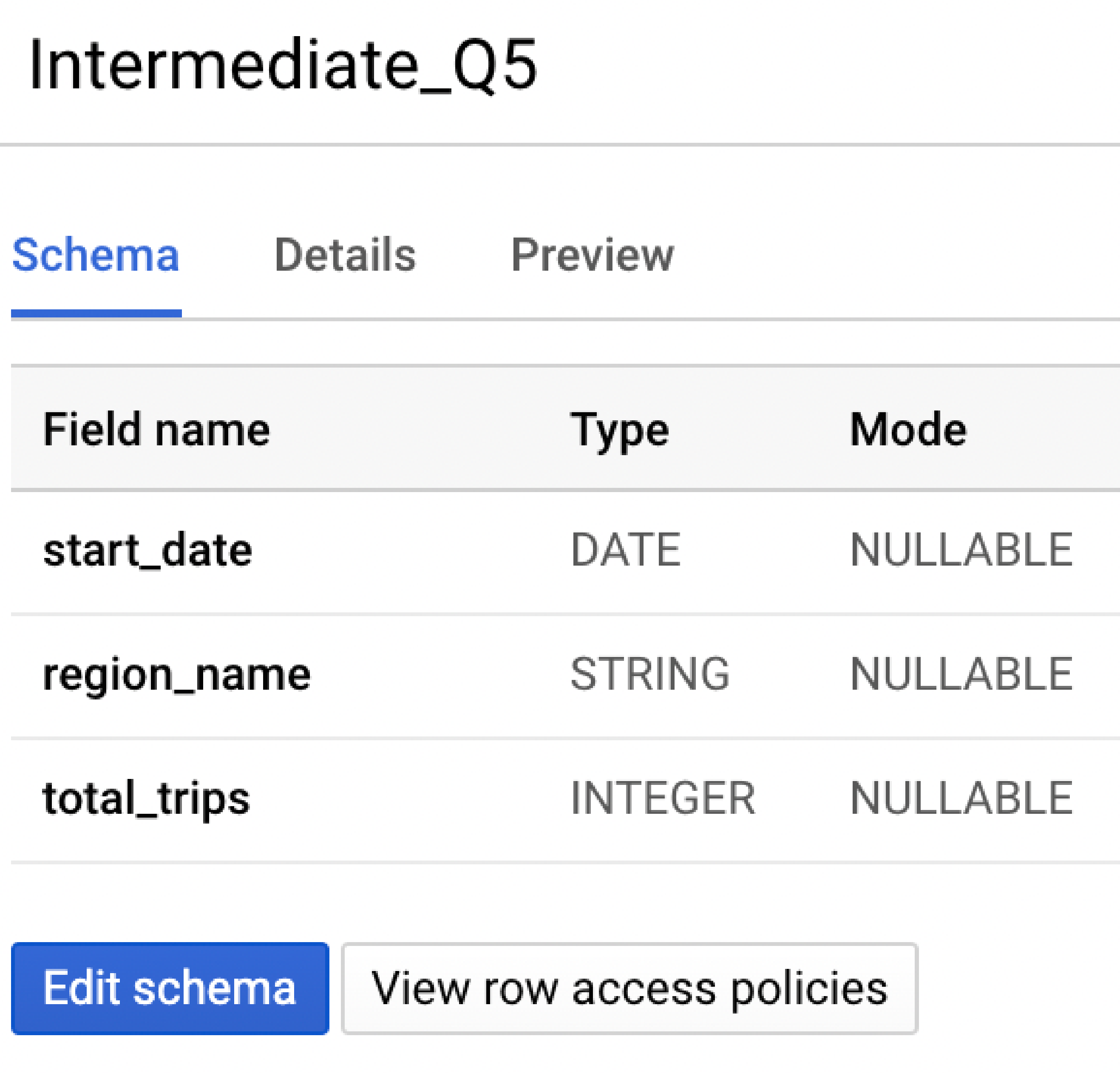
Question 4 : Syntax

```
WITH
 temporary AS (
 SELECT
  C.name AS region_name,
  trip_id,
  duration_sec,
  start_date,
  start_station_name,
  member_gender
 FROM
  `bigquery-public-data.san_francisco_bikeshare.bikeshare_trips` A
 INNER JOIN
  `bigquery-public-data.san_francisco_bikeshare.bikeshare_station_info` B
 ON
  A.start_station_id = B.station_id
 INNER JOIN
  `bigquery-public-data.san_francisco_bikeshare.bikeshare_regions` C
 ON
  B.region_id=C.region_id
 WHERE
  start_date BETWEEN '2014-01-01'
  AND '2017-12-31'
  AND member_gender IS NOT NULL )
```

```
SELECT
 region_name,
 trip_id,
 duration_sec,
 start_date,
 start_station_name,
 member_gender
FROM (
 SELECT *,
  MAX(start_date) OVER (PARTITION BY (region_name)) AS latest_trip
 FROM
  temporary )
WHERE
 start_date = latest_trip
ORDER BY
 1
```

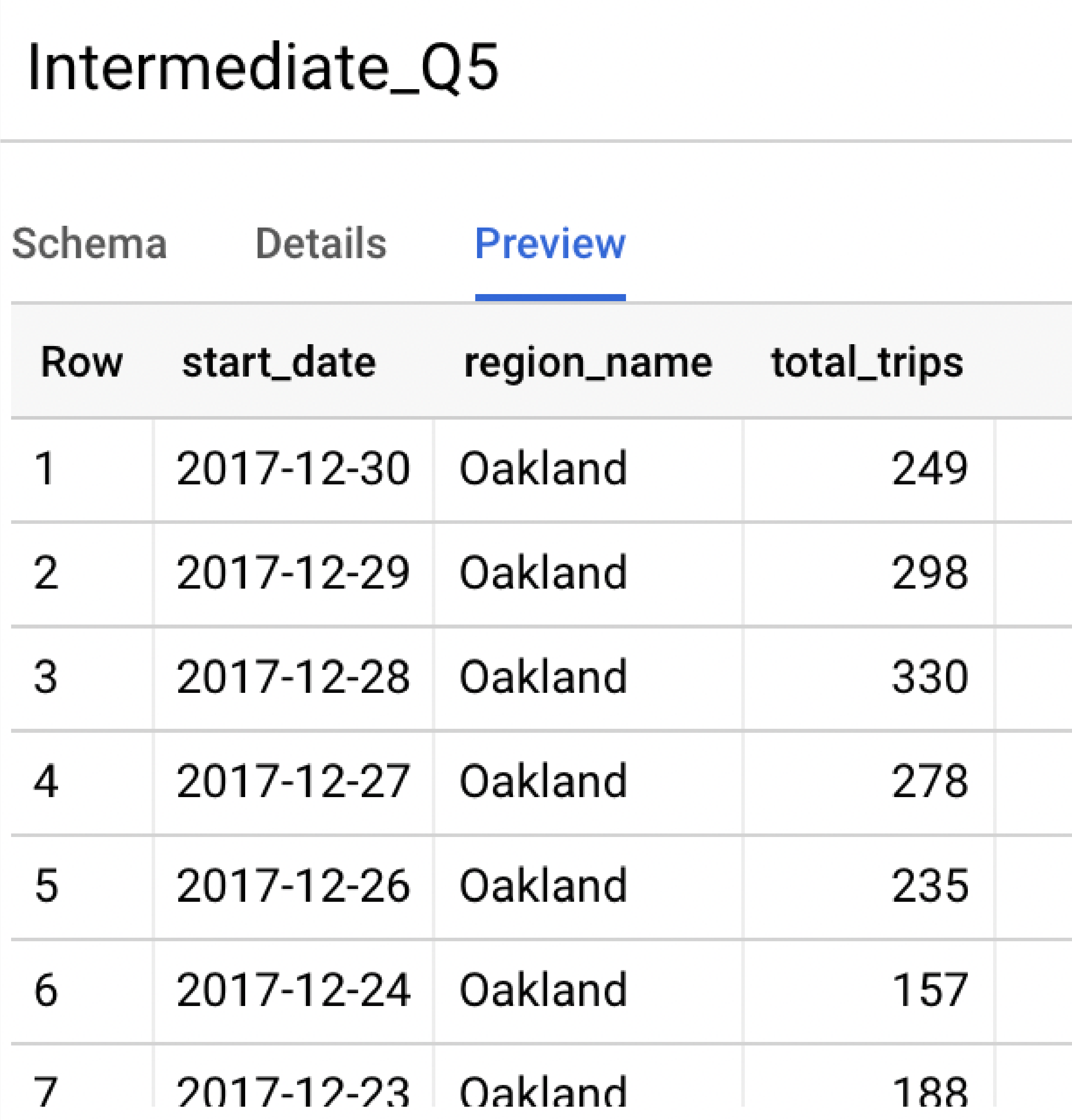
Question 5 : Table and Schema

Create a query to get Month to Date of total trips in each region breakdown by date



Intermediate_Q5

Field name	Type	Mode
start_date	DATE	NULLABLE
region_name	STRING	NULLABLE
total_trips	INTEGER	NULLABLE



Intermediate_Q5

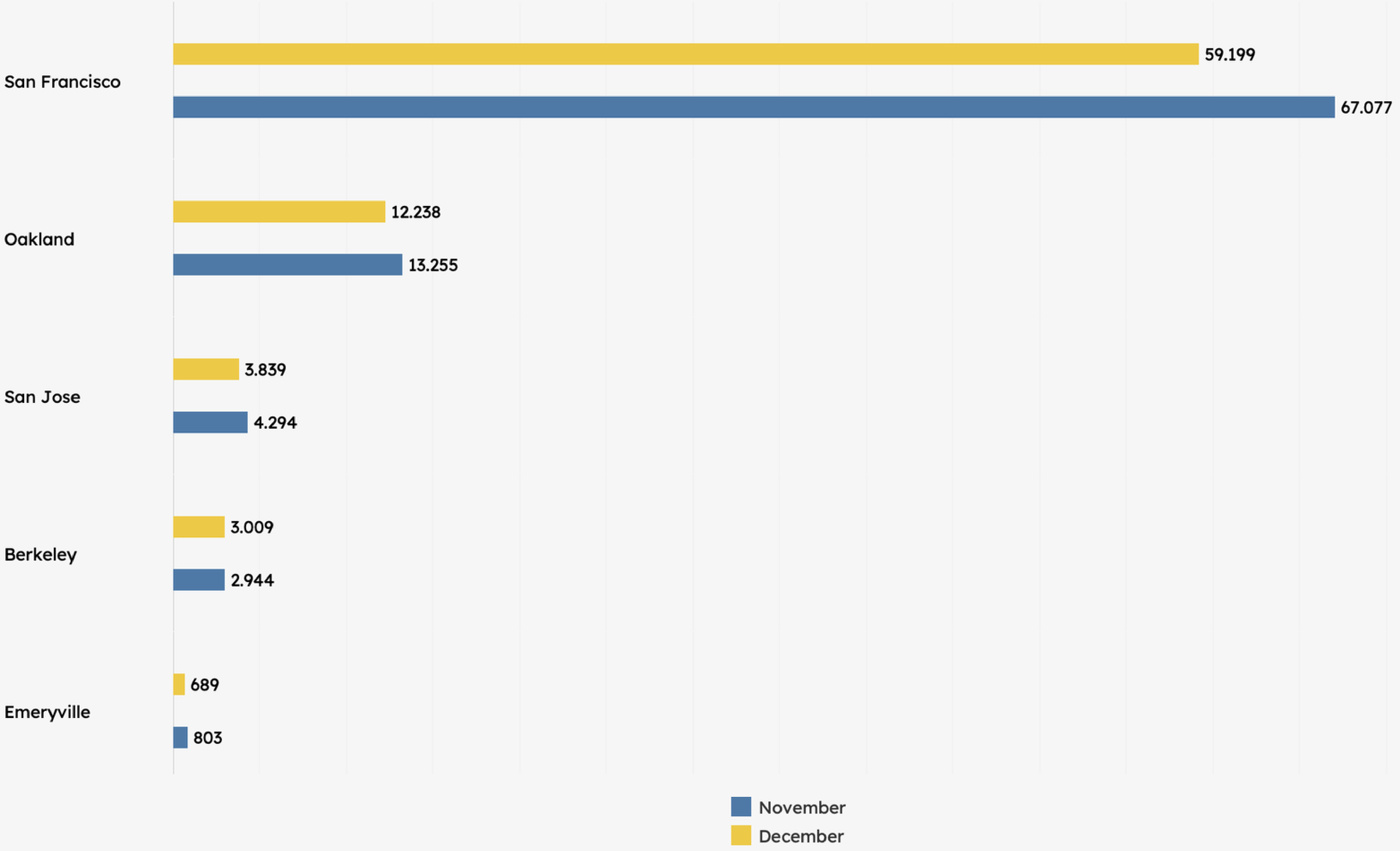
Row	start_date	region_name	total_trips
1	2017-12-30	Oakland	249
2	2017-12-29	Oakland	298
3	2017-12-28	Oakland	330
4	2017-12-27	Oakland	278
5	2017-12-26	Oakland	235
6	2017-12-24	Oakland	157
7	2017-12-23	Oakland	188

Question 5 : Syntax

```
WITH
 total AS(
 SELECT
  DATE(DATE_TRUNC(start_date,DAY)) AS start_date,
  region_table.name AS region_name,
  COUNT(DISTINCT(trip_id)) AS total_trips
 FROM
  `bigquery-public-data.san_francisco_bikeshare.bikeshare_trips`  AS trip_table
 INNER JOIN
  `bigquery-public-data.san_francisco_bikeshare.bikeshare_station_info`  AS info_table
 ON
  trip_table.start_station_id=info_table.station_id
 INNER JOIN
  `bigquery-public-data.san_francisco_bikeshare.bikeshare_regions` AS region_table
 ON
  info_table.region_id=region_table.region_id
 WHERE
  start_date BETWEEN '2017-11-01'
  AND '2017-12-31'
 GROUP BY 1,2)

SELECT *
FROM total
```

Question 5 : Visualization and Insight



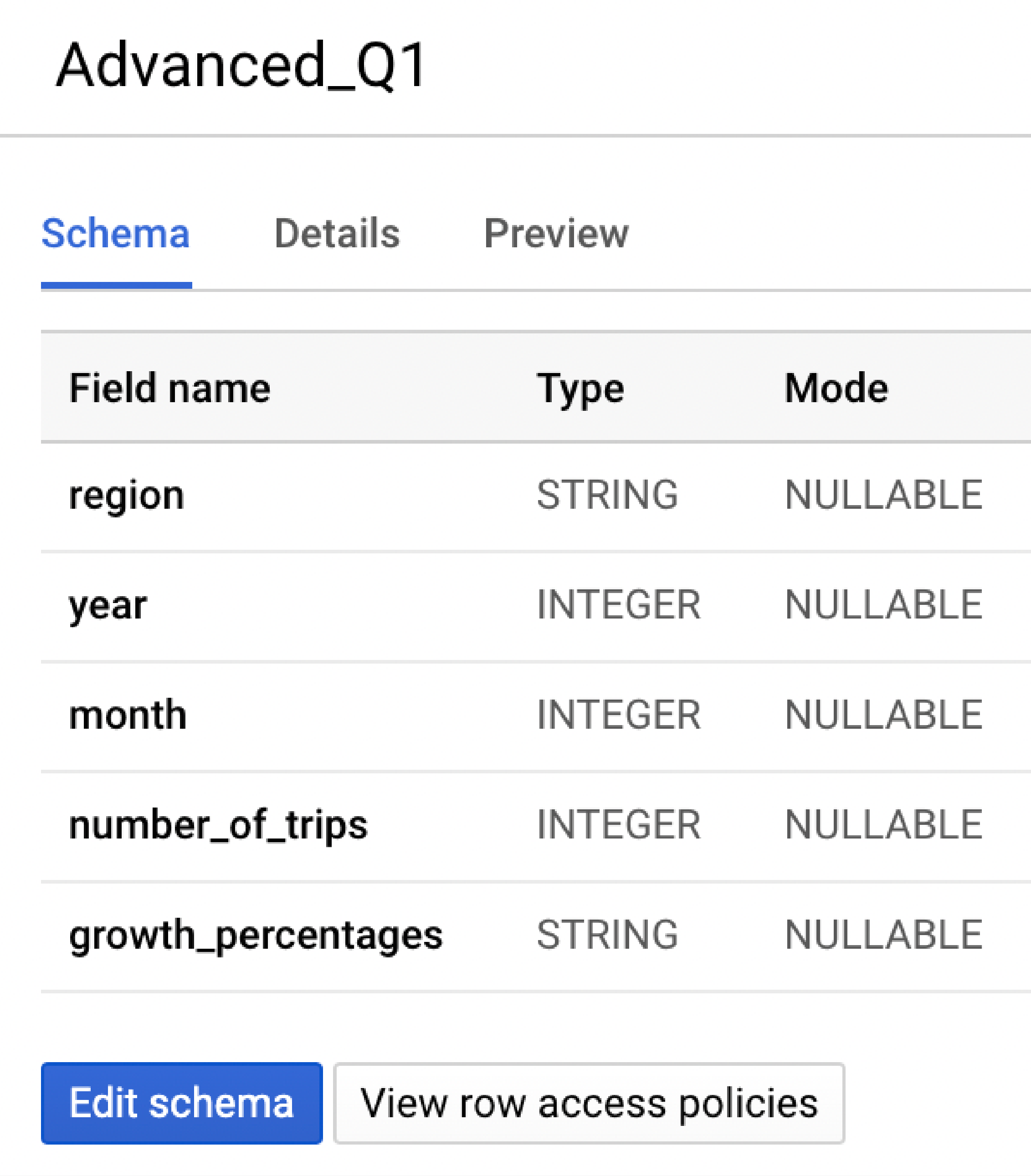
- The total trips in **December** are generally decline compared with the total trips in **November**

- The total trips of Berkeley slightly increased compared to the previous month

Advanced Assignment

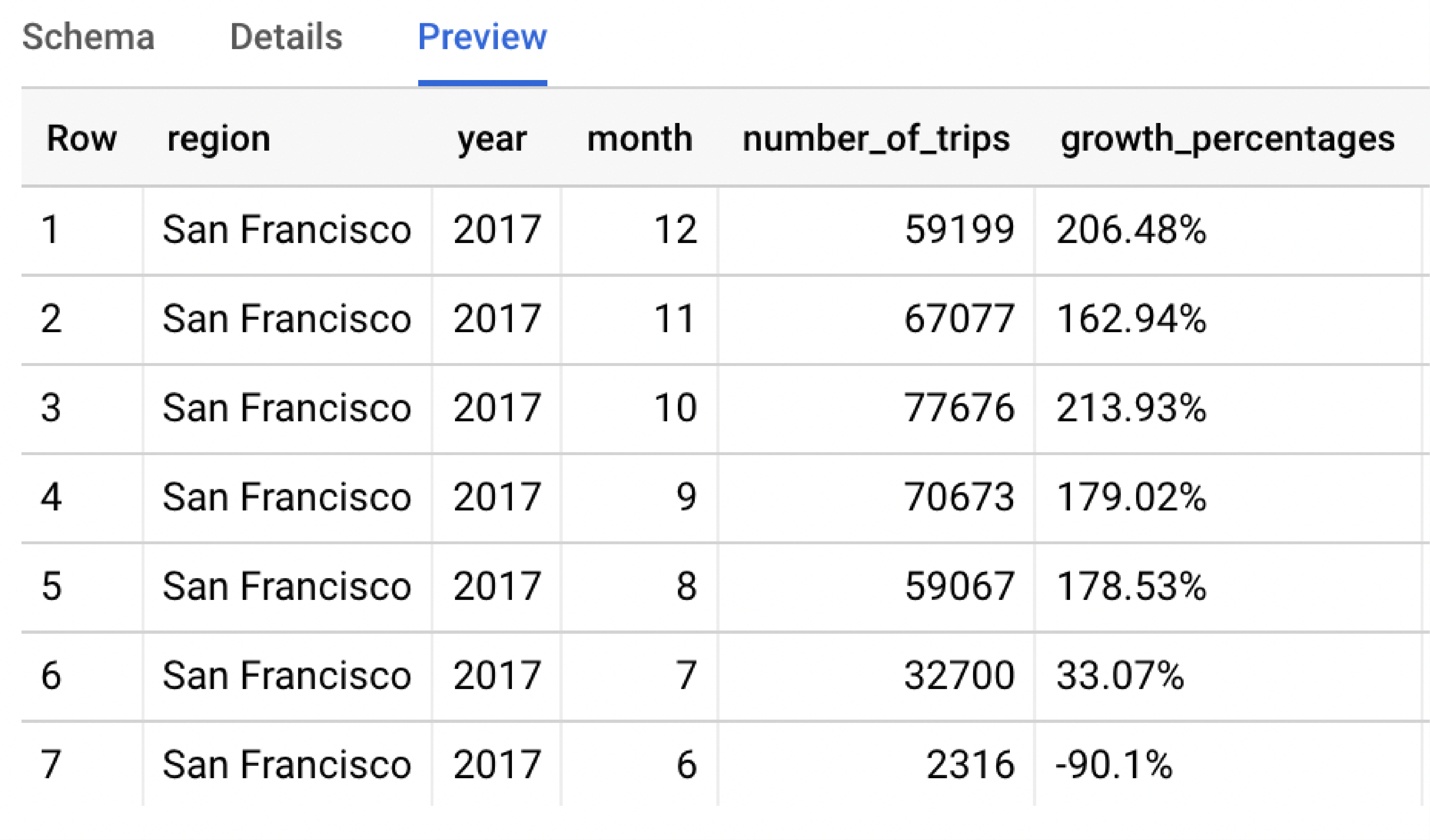
Question 6 : Table and Schema

Find monthly growth of trips in percentage, order by time descendingly (only for trips from the region with highest total number of trips)



Advanced_Q1

Field name	Type	Mode
region	STRING	NULLABLE
year	INTEGER	NULLABLE
month	INTEGER	NULLABLE
number_of_trips	INTEGER	NULLABLE
growth_percentages	STRING	NULLABLE



Row	region	year	month	number_of_trips	growth_percentages
1	San Francisco	2017	12	59199	206.48%
2	San Francisco	2017	11	67077	162.94%
3	San Francisco	2017	10	77676	213.93%
4	San Francisco	2017	9	70673	179.02%
5	San Francisco	2017	8	59067	178.53%
6	San Francisco	2017	7	32700	33.07%
7	San Francisco	2017	6	2316	-90.1%

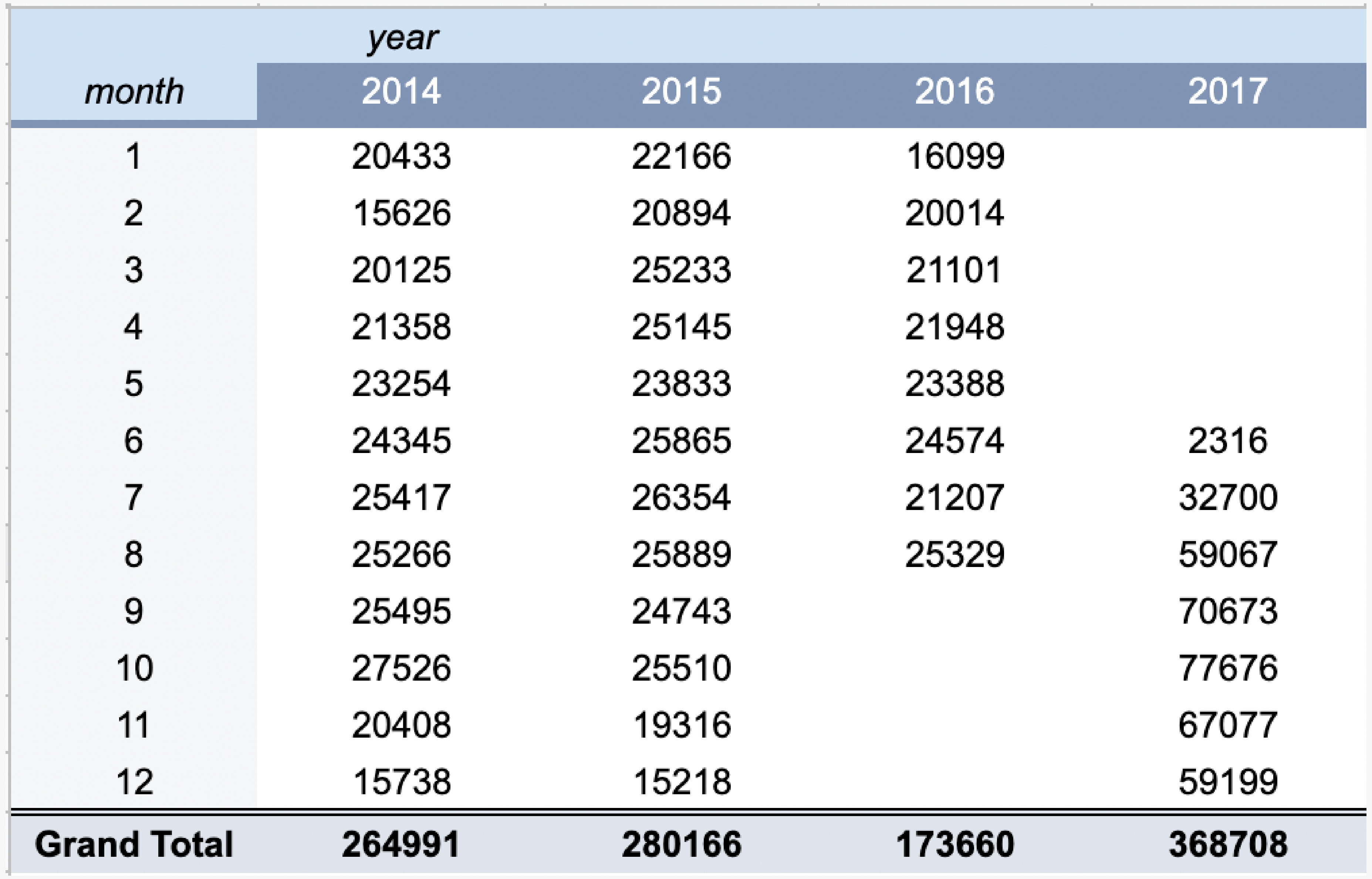
Question 6 : Syntax

```
WITH
 highest_region_trip AS(
 SELECT
  region_table.name AS region_name,
  COUNT(DISTINCT(trip_id)) AS number_of_trips,
  ROW_NUMBER()OVER(ORDER BY (COUNT(DISTINCT(trip_id)))DESC) AS rank
 FROM
  ` bigquery-public-data.san_francisco_bikeshare.bikeshare_trips`  AS trip_table
 INNER JOIN
  ` bigquery-public-data.san_francisco_bikeshare.bikeshare_station_info`  AS info_table
 ON
  trip_table.start_station_id=info_table.station_id
 INNER JOIN
  ` bigquery-public-data.san_francisco_bikeshare.bikeshare_regions` AS region_table
 ON
  info_table.region_id=region_table.region_id
 GROUP BY
  1),
```

```
helper_table AS(
 SELECT
  region_table.name AS region_name,
  COUNT(DISTINCT(trip_id)) AS number_of_trips,
  EXTRACT(MONTH FROM start_date)AS month,
  EXTRACT(YEAR FROM start_date)AS year,
 FROM
  ` bigquery-public-data.san_francisco_bikeshare.bikeshare_trips`  AS trip_table
 INNER JOIN
  ` bigquery-public-data.san_francisco_bikeshare.bikeshare_station_info`  AS info_table
 ON
  trip_table.start_station_id=info_table.station_id
 INNER JOIN
  ` bigquery-public-data.san_francisco_bikeshare.bikeshare_regions` AS region_table
 ON
  info_table.region_id=region_table.region_id
 WHERE
  start_date BETWEEN '2014-01-01'
  AND '2017-12-31'
  AND region_table.name IN (
  SELECT
   region_name
  FROM
   highest_region_trip
  WHERE
   rank = 1)
 GROUP BY
  1,3,4
 ORDER BY 1 )
```

```
SELECT
 region_name AS region,
 year AS year,
 month AS month,
 number_of_trips,
 growth_percentages
FROM (
 SELECT
  *,
  CONCAT(ROUND(((number_of_trips) - LEAD(number_of_trips)OVER(ORDER BY month
DESC))/LEAD(number_of_trips)OVER(ORDER BY month DESC)*100,2),'%') AS growth_percentages
 FROM
  helper_table)
ORDER BY
 2 DESC,
 3 DESC
```

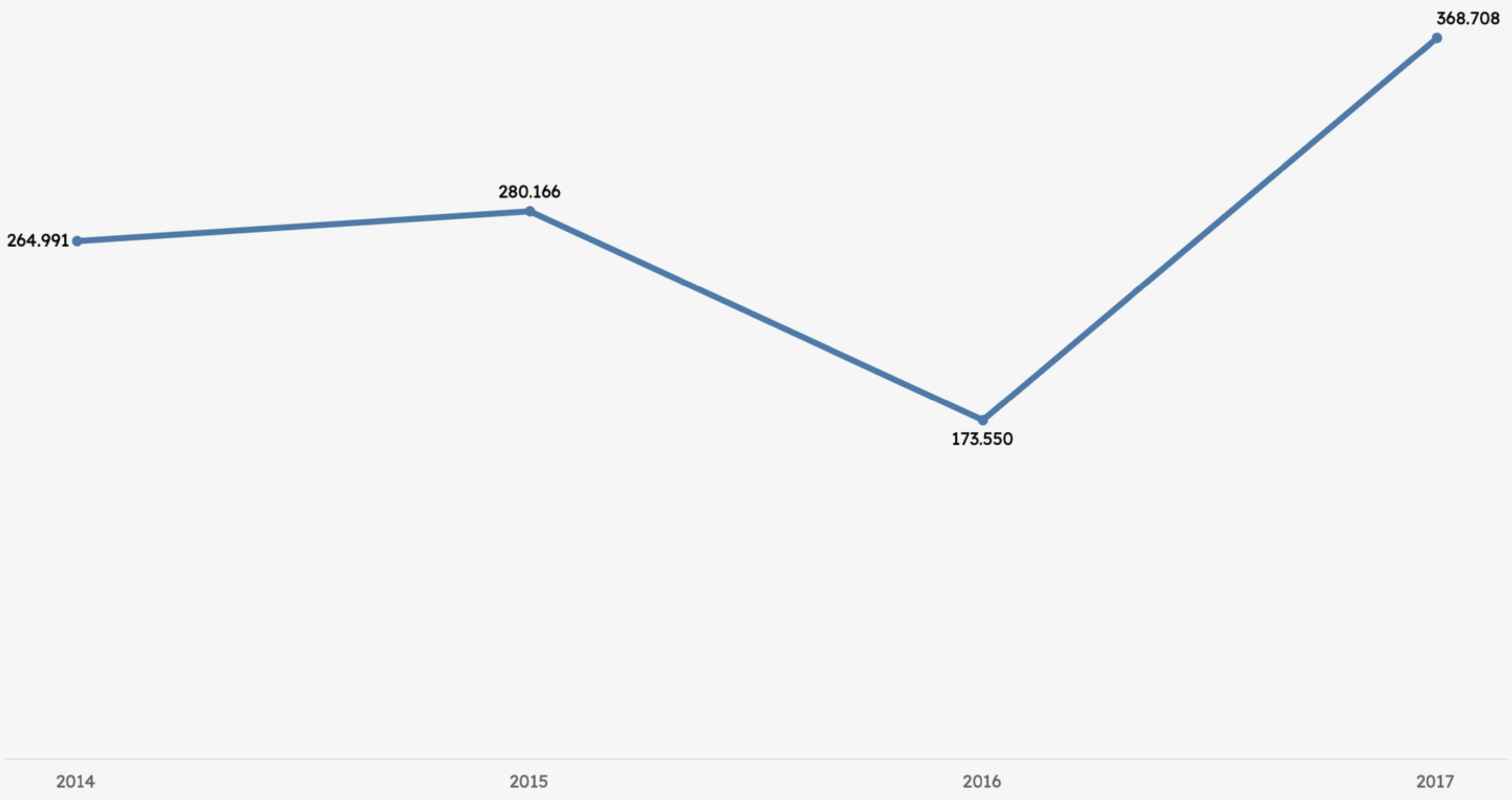
Question 6 : Visualization and Insight



	year			
month	2014	2015	2016	2017
1	20433	22166	16099	
2	15626	20894	20014	
3	20125	25233	21101	
4	21358	25145	21948	
5	23254	23833	23388	
6	24345	25865	24574	2316
7	25417	26354	21207	32700
8	25266	25889	25329	59067
9	25495	24743		70673
10	27526	25510		77676
11	20408	19316		67077
12	15738	15218		59199
Grand Total	**264991**	**280166**	**173660**	**368708**

With a simple Pivot table, we can look into the total trips of each month throughout the years. We can see that from September 2016 until May 2017, we got a missing value of trips. It could be an error in the data input process/record, so we should confirm with the related team first (data engineer or database PIC) before doing further analysis.

Question 6 : Visualization and Insight



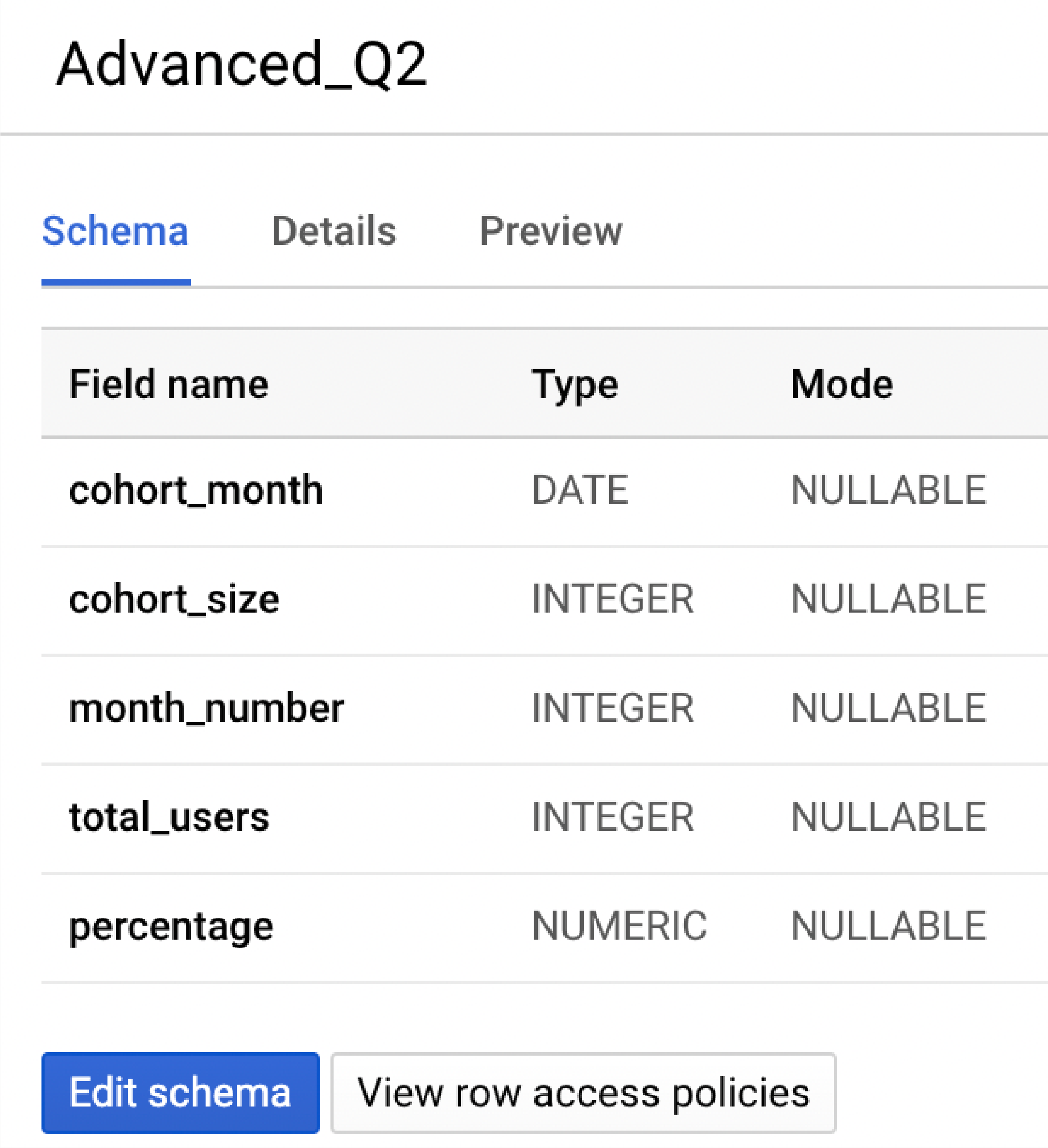
- Based on the chart made with the available data, **the number of trips shows an increment each** year except in 2016.

- The low number of trips in 2016 could be contributed to the missing data values

- The **highest number of trips occurred in 2017** with 368,708 trips, even with a couple of months of missing data. So we expect the total number could be even bigger.

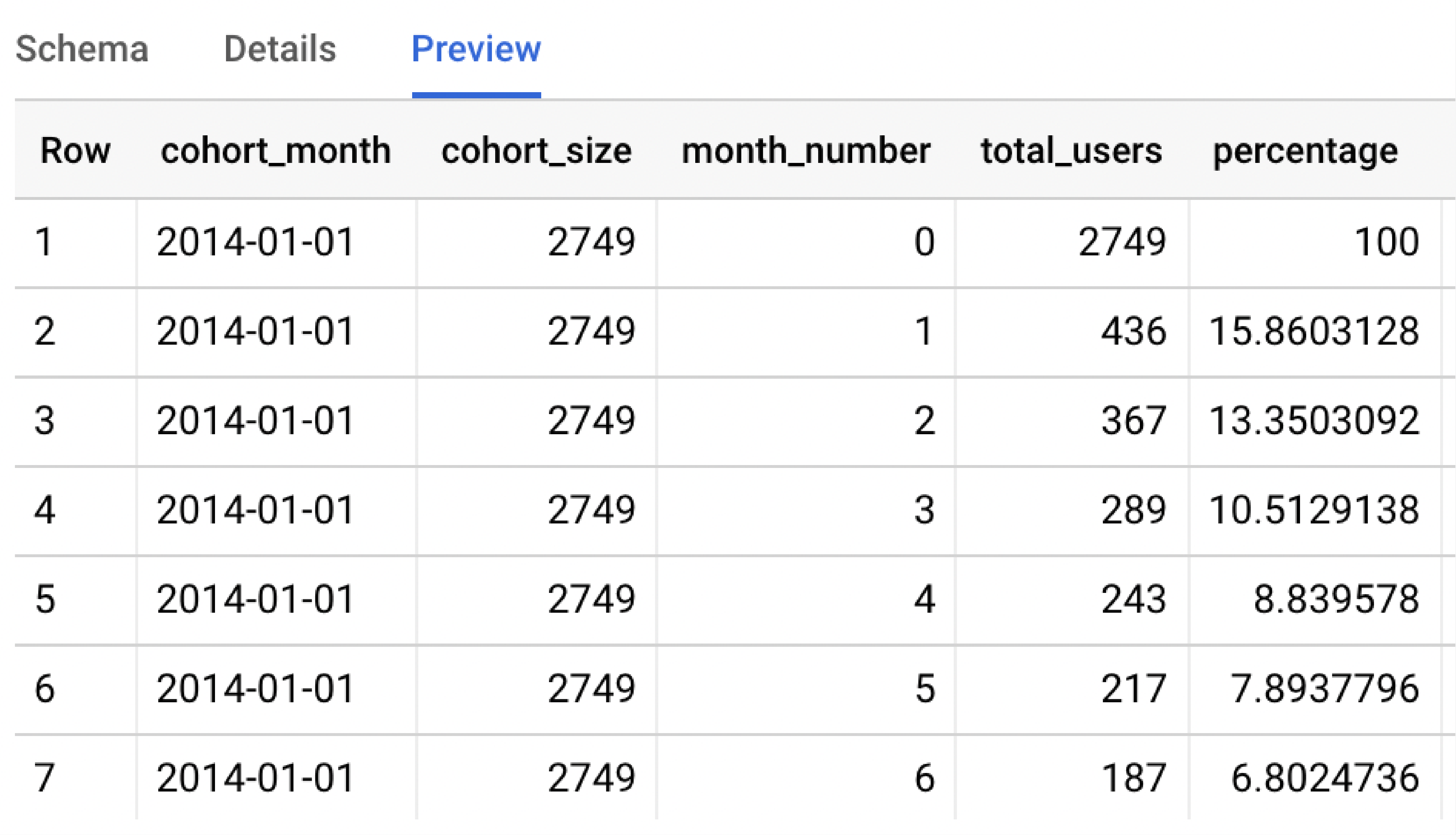
Question 7 : Table and Schema

Create monthly retention Cohorts using table “Stories” (Hacker News Dataset) to find how many authors coming back for the following months



Advanced_Q2

Field name	Type	Mode
cohort_month	DATE	NULLABLE
cohort_size	INTEGER	NULLABLE
month_number	INTEGER	NULLABLE
total_users	INTEGER	NULLABLE
percentage	NUMERIC	NULLABLE



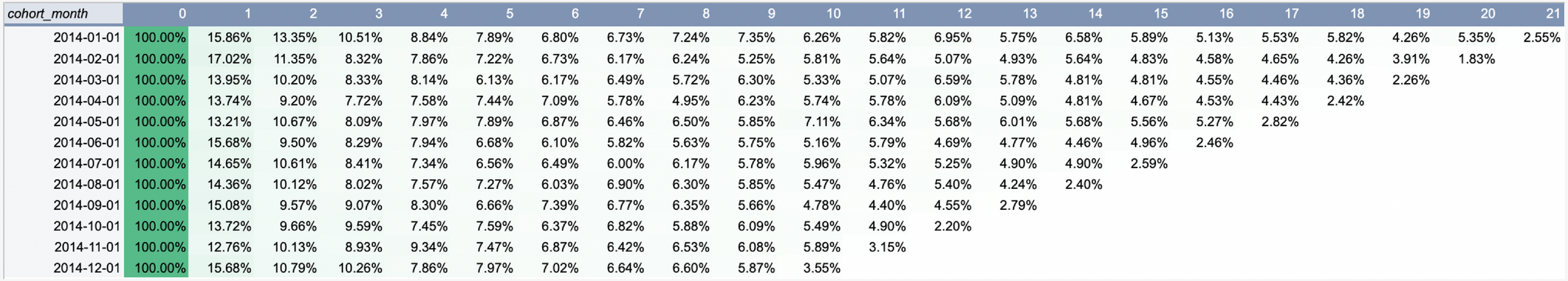
Row	cohort_month	cohort_size	month_number	total_users	percentage
1	2014-01-01	2749	0	2749	100
2	2014-01-01	2749	1	436	15.8603128
3	2014-01-01	2749	2	367	13.3503092
4	2014-01-01	2749	3	289	10.5129138
5	2014-01-01	2749	4	243	8.839578
6	2014-01-01	2749	5	217	7.8937796
7	2014-01-01	2749	6	187	6.8024736

Question 7 : Syntax

```
WITH
 cohort_items AS(
 SELECT
  author AS author,
  MIN(DATE(DATE_TRUNC(time_ts,MONTH))) AS cohort_month,
 FROM
  ` bigquery-public-data.hacker_news.stories`
 GROUP BY 1),
 user_activities AS (
 SELECT
  act.author AS author,
  DATE_DIFF(DATE(DATE_TRUNC(time_ts,MONTH)), cohort.cohort_month, MONTH ) AS month_number,
 FROM
  ` bigquery-public-data.hacker_news.stories`  act
 LEFT JOIN
  cohort_items AS cohort
 ON
  act.author = cohort.author
 WHERE
  EXTRACT(year FROM cohort.cohort_month) IN (2014)
 GROUP BY 1,2),
 cohort_size AS (
 SELECT
  cohort_month,
  COUNT(1) AS num_users
 FROM
  cohort_items
 GROUP BY
  1
 ORDER BY
  1),
```

```
retention_table AS (
 SELECT
  C.cohort_month,
  A.month_number AS month_number,
  COUNT(1) AS num_users
 FROM
  user_activities A
 LEFT JOIN
  cohort_items C
 ON
  A.author = C.author
 GROUP BY 1,2)
SELECT
 B.cohort_month,
 S.num_users AS cohort_size,
 B.month_number,
 B.num_users AS total_users,
 CAST(B.num_users AS decimal)/ S.num_users*100 AS percentage
FROM
 retention_table B
LEFT JOIN
 cohort_size S
ON
 B.cohort_month = S.cohort_month
WHERE
 B.cohort_month IS NOT NULL
ORDER BY  1,3
```

Question 7 : Visualization and Insight



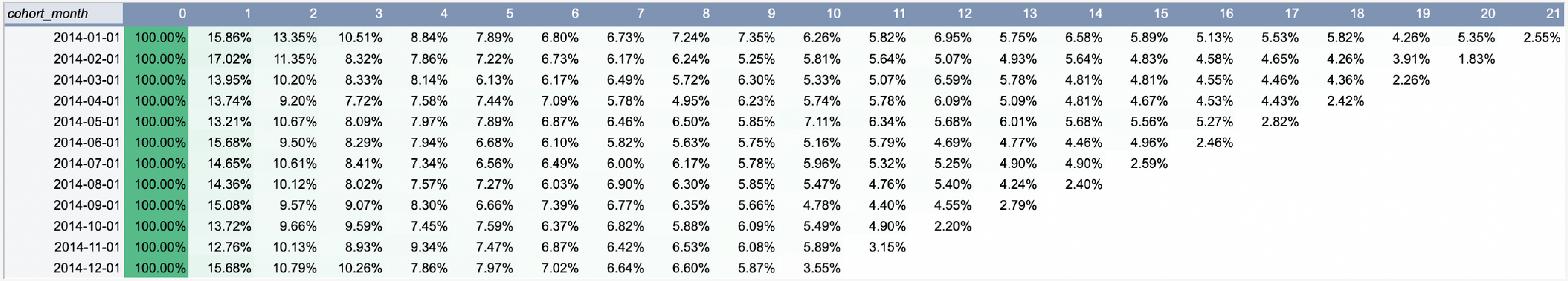
cohort_month	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
2014-01-01	100.00%	15.86%	13.35%	10.51%	8.84%	7.89%	6.80%	6.73%	7.24%	7.35%	6.26%	5.82%	6.95%	5.75%	6.58%	5.89%	5.13%	5.53%	5.82%	4.26%	5.35%	2.55%
2014-02-01	100.00%	17.02%	11.35%	8.32%	7.86%	7.22%	6.73%	6.17%	6.24%	5.25%	5.81%	5.64%	5.07%	4.93%	5.64%	4.83%	4.58%	4.65%	4.26%	3.91%	1.83%	
2014-03-01	100.00%	13.95%	10.20%	8.33%	8.14%	6.13%	6.17%	6.49%	5.72%	6.30%	5.33%	5.07%	6.59%	5.78%	4.81%	4.81%	4.55%	4.46%	4.36%	2.26%		
2014-04-01	100.00%	13.74%	9.20%	7.72%	7.58%	7.44%	7.09%	5.78%	4.95%	6.23%	5.74%	5.78%	6.09%	5.09%	4.81%	4.67%	4.53%	4.43%	2.42%			
2014-05-01	100.00%	13.21%	10.67%	8.09%	7.97%	7.89%	6.87%	6.46%	6.50%	5.85%	7.11%	6.34%	5.68%	6.01%	5.68%	5.56%	5.27%	2.82%				
2014-06-01	100.00%	15.68%	9.50%	8.29%	7.94%	6.68%	6.10%	5.82%	5.63%	5.75%	5.16%	5.79%	4.69%	4.77%	4.46%	4.96%	2.46%					
2014-07-01	100.00%	14.65%	10.61%	8.41%	7.34%	6.56%	6.49%	6.00%	6.17%	5.78%	5.96%	5.32%	5.25%	4.90%	4.90%	2.59%						
2014-08-01	100.00%	14.36%	10.12%	8.02%	7.57%	7.27%	6.03%	6.90%	6.30%	5.85%	5.47%	4.76%	5.40%	4.24%	2.40%							
2014-09-01	100.00%	15.08%	9.57%	9.07%	8.30%	6.66%	7.39%	6.77%	6.35%	5.66%	4.78%	4.40%	4.55%	2.79%								
2014-10-01	100.00%	13.72%	9.66%	9.59%	7.45%	7.59%	6.37%	6.82%	5.88%	6.09%	5.49%	4.90%	2.20%									
2014-11-01	100.00%	12.76%	10.13%	8.93%	9.34%	7.47%	6.87%	6.42%	6.53%	6.08%	5.89%	3.15%										
2014-12-01	100.00%	15.68%	10.79%	10.26%	7.86%	7.97%	7.02%	6.64%	6.60%	5.87%	3.55%											

100%	75%	50%	25%	0%

Insight :

- The cohorts show that over the year, the monthly retention generally declined. As shown in month 1, it even jumps down to 15,86% from the initial month (churn rate ~85%)
- The retention rate constantly shows decrement reaching 2,55% in month 21
- The monthly retentions across the tables show a low retention rate with the value is not even reaching 20% every month.

Question 7 : Visualization and Insight



cohort_month	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
2014-01-01	100.00%	15.86%	13.35%	10.51%	8.84%	7.89%	6.80%	6.73%	7.24%	7.35%	6.26%	5.82%	6.95%	5.75%	6.58%	5.89%	5.13%	5.53%	5.82%	4.26%	5.35%	2.55%
2014-02-01	100.00%	17.02%	11.35%	8.32%	7.86%	7.22%	6.73%	6.17%	6.24%	5.25%	5.81%	5.64%	5.07%	4.93%	5.64%	4.83%	4.58%	4.65%	4.26%	3.91%	1.83%	
2014-03-01	100.00%	13.95%	10.20%	8.33%	8.14%	6.13%	6.17%	6.49%	5.72%	6.30%	5.33%	5.07%	6.59%	5.78%	4.81%	4.81%	4.55%	4.46%	4.36%	2.26%		
2014-04-01	100.00%	13.74%	9.20%	7.72%	7.58%	7.44%	7.09%	5.78%	4.95%	6.23%	5.74%	5.78%	6.09%	5.09%	4.81%	4.67%	4.53%	4.43%	2.42%			
2014-05-01	100.00%	13.21%	10.67%	8.09%	7.97%	7.89%	6.87%	6.46%	6.50%	5.85%	7.11%	6.34%	5.68%	6.01%	5.68%	5.56%	5.27%	2.82%				
2014-06-01	100.00%	15.68%	9.50%	8.29%	7.94%	6.68%	6.10%	5.82%	5.63%	5.75%	5.16%	5.79%	4.69%	4.77%	4.46%	4.96%	2.46%					
2014-07-01	100.00%	14.65%	10.61%	8.41%	7.34%	6.56%	6.49%	6.00%	6.17%	5.78%	5.96%	5.32%	5.25%	4.90%	4.90%	2.59%						
2014-08-01	100.00%	14.36%	10.12%	8.02%	7.57%	7.27%	6.03%	6.90%	6.30%	5.85%	5.47%	4.76%	5.40%	4.24%	2.40%							
2014-09-01	100.00%	15.08%	9.57%	9.07%	8.30%	6.66%	7.39%	6.77%	6.35%	5.66%	4.78%	4.40%	4.55%	2.79%								
2014-10-01	100.00%	13.72%	9.66%	9.59%	7.45%	7.59%	6.37%	6.82%	5.88%	6.09%	5.49%	4.90%	2.20%									
2014-11-01	100.00%	12.76%	10.13%	8.93%	9.34%	7.47%	6.87%	6.42%	6.53%	6.08%	5.89%	3.15%										
2014-12-01	100.00%	15.68%	10.79%	10.26%	7.86%	7.97%	7.02%	6.64%	6.60%	5.87%	3.55%											

100%	75%	50%	25%	0%

Suggestion :

- With a low retention rate, it is known that users/authors do not actively share their stories within the community so the platform needs to find a way to engage its users to increase the retention rate
- We can concentrate the analysis on month 1 where the highest drop in retention rate occurred, and find out why the users were not engaged anymore for the following months

I DON'T ALWAYS TYPE IN ALL CAPS,
BUT WHEN I DO, I'M PROBABLY WRITING A SQL QUERY
imgflip.com