Month-over-month

SELECT DATE\_FORMAT(host\_since, '%Y-%m') as yearmonth, neighbourhood,

COUNT(DISTINCT host\_id) as countid,

FORMAT(100 \* (COUNT(DISTINCT host\_id) - LAG(COUNT(id), 1) OVER (PARTITION BY neighbourhood

ORDER BY date\_format(host\_since, '%Y-%m'))) / LAG(COUNT(DISTINCT host\_id), 1) OVER

(PARTITION BY neighbourhood ORDER BY DATE\_FORMAT(host\_since, '%Y-%m')), 2)

AS growth

FROM listings

WHERE host\_since BETWEEN '2018-06-01' AND '2018-08-31'

AND neighbourhood <> ''

GROUP BY 1,2

ORDER BY 1

;

Except

CREATE TABLE if not exists Listings1 AS

SELECT \* from listings

EXCEPT

SELECT l.\*

FROM listings l

JOIN reviews r ON l.id = r.listing\_id

WHERE r.date BETWEEN '2018-01-01' and '2018-12-31'

;

SELECT COUNT(id), host\_response\_time

FROM LISTINGS1

GROUP BY 2

ORDER BY 1 DESC

;

SELECT COUNT(id), neighbourhood,

host\_identity\_verified,room\_type

FROM listings

WHERE neighbourhood <> ''

AND room\_type = 'Private room'

GROUP BY 2,3,4

ORDER BY 1 DESC

LIMIT 2;

**PART ONE |** Analyzing Dates

**Accounts**

|  |  |  |
| --- | --- | --- |
| account\_id | created\_date | campaign\_id |
|  |  |  |

**Campaigns**

|  |  |  |  |
| --- | --- | --- | --- |
| campaign\_id | created\_date | revenue | Product |
|  |  |  |  |

1. [Write a query] For each campaign *without an account,* what was the number of campaigns created in December 2018?
2. [Write a query] For each campaign *with an account*, what is the number of days between a campaign created and an account created?
3. [Open Answer] Provide at least 3 potential data quality issues from your answer to #2 that would warrant further investigation.

1

Select count(campaign\_id)

From campaigns

Where created\_date between ‘12-01-2018’ and ’12-31-2018’

;

2

Select datediff(days, c.created\_date , a.created\_date) as num\_days

From campaigns c

Join account a

On c.campaign\_id = a.campaign\_id;

3

1. If the two date columns do not have same data type.
2. The date format is different for both columns
3. Duplicate records may be present.

**PART TWO |** Analyzing Revenue

**Revenue**

|  |  |  |
| --- | --- | --- |
| **account\_id** | **revenue\_2017** | **revenue\_2018** |
| 1111 | 1000 | 1300 |
| 1112 | 500 | 800 |
| 1113 | 750 | 1200 |
| 1114 | 3200 | 3500 |
| … |  |  |

**Team**

|  |  |  |
| --- | --- | --- |
| **account\_id** | **team** | **region** |
| 1111 | Tech | NAMER-US |
| 1112 | Services | NAMER-CA |
| 1113 | Government | LATAM |
| 1114 | Education | NAMER-CA |
| … |  |  |

1. [Write a query] What is the 2017 to 2018 year-over-year growth rate by account?
2. [Write a query] We suspect the “team” table may have a data quality issue. Write a query that identifies the top 5 teams by 2018 revenue that are mapped to more than one region and list the # of regions in the same row.
3. [Write a query] Show two columns in the output where one is the total 2018 revenue from the “Tech” Team “NAMER-US” accounts and another is the total 2018 revenue from “Tech” Team “NAMER-CA” accounts.

1

Select account\_id, ((revenue\_2018 – revenue\_2017)/revenue\_2018) \* 100 as growth

From revenue;

2

Select t.team, count(t.region), r.revenue\_2018

From team t

Join

Revenue r

On t.account\_id = r.account\_id

Group by t.team

Order by r.revenue\_2018

limit 5

;

3

Select sum(r.revenue\_2018) as rev\_namerus, sum(r.revenue\_2018) as rev\_namerca

From revenue r

Join team t

On

r.account\_id = t.account\_id

where t.team = ‘namer\_us’ and ‘Namer\_ca’

;

select a1.campaign ,a2.name as username,a1.device\_id,a3.total\_amt

from ((SELECT campaign,ud.device\_id,ud.userid from attribution a , user\_device ud where a.device\_id =ud.device\_id

group by a.campaign ,ud.device\_id,ud.userid) a1

inner join

(select id,name from [User] u group by id,name) a2

on a1.userid = a2.id )

inner join (select s.userid , SUM(s.amount) as total\_amt from sale s , user\_device ud where ud.userid =s.userid

group by s.userid ) a3

on a2.id = a3.userid

order by 1,2,3

;

1. SQL

* On a typical day, what share of registrants are coming from each acquisition

source?

select u.reg\_device,round(cast(Total as Float)/sum(Frequency),2) as 'Average Share of Registrants'

from (select reg\_device,SUM(1) as 'Total' from Users group by reg\_device) u inner join (select reg\_device ,convert(varchar(12),reg\_ts,101) as 'reg\_date' ,1 as 'Frequency' from Users group by reg\_device,convert(varchar(12),reg\_ts,101)) t on u.reg\_device=t.reg\_device  
group by u.reg\_device,u.Total , t.Frequency  
;

* What is the typical Active Lifetime of a user by device, where Active Lifetime is defined as time from registration to last activity?

select u.user\_id,u.reg\_device,DATEDIFF(MINUTE,u.reg\_ts,max(a.actvy\_ts)) as 'Activity Lifetime (Minutes)' from Users u inner join Activity a  
on a.user\_id = u.user\_id  
group by u.user\_id ,u.reg\_device, u.reg\_ts

order by u.user\_id,u.reg\_device ;

* What is the typical Churned Lifetime overall, where Churned Lifetime is defined as time from registration to last activity for users who have not been active for at least 180 days?

select u.user\_id,DATEDIFF(DAY,u.reg\_ts,max(a.actvy\_ts)) as 'Churned Lifetime (Days)' from Users u inner join Activity a  
on a.user\_id = u.user\_id  
group by u.user\_id, u.reg\_ts

having (DATEDIFF(DAY,u.reg\_ts,max(a.actvy\_ts)))>180 order by u.user\_id  
;

2. Considering the data available in these tables, what characteristics of user behavior would you want to examine in this dataset? Propose 3-5 theories you have about user engagement and describe how you would explore them.

1. **Device type** 
   1. Checking the type of device where the maximum traffic comes from. Checking

which device is being used maximum whether a phone, tablet or a computer.

* 1. Improving the app/website to be more user friendly depending upon where

maximum traffic is coming from.

1. **Session activity** 
   1. Checking the user interaction with app/website by tracking the user activity from

the time they login into the app/website until they quit.

* 1. Watching the user first session activity and see how the user interacts and why he

never used the app/website after the first session.

* 1. Watching the session activity of regular users to see what works the best for them

and how to improve/add features.

1. **Conversion funnels**Tracking user behavior across different stages of a funnel such as payment stage, in-app purchases, onboarding, etc. and time taken for each stage.

i. Measuring time between different stages in the funnel  
ii. Understanding if the users are leaving upon increasing the in-app purchases what

made them to leave without converting.  
iii. Session recording for login and creating an account and improving onboardings.

d. **Action Cohorts**

1. Tracking actions to analyze the relation between different related actions to see

the trends and engagement of the user over a given period of time.

1. Tracking user behavior from time of onboarding to time of account creation.
2. Tracking relationship between first and successive sessions.
3. Tracking user cohorts who created the account and then left without making any

in-app purchases or using any other service to increase user retention.

1

* What are the 5 most listened to playlists?

SELECT playlist\_id, count(playlist\_id) as most\_listened from plays

GROUP BY playlist\_id

ORDER BY most\_listened DESC

limit 5

;

* Given a user X, what are their 3 most listened to playlists?

SELECT user\_id, playlist\_id, count(playlist\_id)

FROM plays

WHERE user\_id = X

GROUP BY playlist\_id

ORDER BY playlist\_id DESC

limit 3

;

* Which users have uploaded a majority of tracks belonging to ‘HipHop & R&B’ or ‘Dance & Electronic’ genre categories? (please answer this question with and without the use of a window/analytical function)

**Without window function**

SELECT user\_id, count(track\_id) as max\_tracks

FROM tracks

WHERE track\_genre\_category = 'HipHop & R&B' or track\_genre\_category = 'Dance & Electronic'

GROUP BY user\_id

ORDER BY max\_tracks DESC

;

**With window function**

SELECT user\_id, track\_id, track\_genre\_category,

COUNT(track\_id) OVER (PARTITION BY track\_genre\_category) total\_count

FROM tracks

;

* Which uploaders have churned recently?

For this question I used the tracks table and perform self -join on it to see how many users were not active in last 90 days i.e., which users did not upload any tracks.

SELECT t.user\_id

FROM tracks t

JOIN tracks tr ON t.user\_at = tr.user\_ID

GROUP BY t.user\_id

HAVING MAX(tr.created\_at) < CURDATE() - INTERVAL 90 DAY

OR MAX(tr.created\_at) IS NULL

;

* For a user X, recommend 5 playlists they would like that they haven’t heard before. Please attempt this using SQL and (where appropriate) common table expressions. Please explain your logic.

For this question I first try to find what is the most listened to tracks by user by genre category.

I want to find the most listed tracks belonging to a particular genre. For that I use following query to get ouput for user\_id = 1.

SELECT p.user\_id, COUNT(p.track\_id), t.track\_genre\_category

FROM plays p

JOIN tracks t

ON p.track\_id = t.track\_id

WHERE p.user\_id = 1

GROUP by 1, 3

ORDER BY 2 DESC

;

If the above output says ‘Classic is the most listened I can get the track\_id from ‘Classic’ genre from tracks table that were not listened by the user.

SELECT p.user\_id, p.track\_id, t.track\_id, t.track\_genre\_category from plays p

RIGHT JOIN tracks t

ON p.track\_id = t.track\_id

WHERE t.track\_genre\_category = ‘Classic’

;

2. There might be several reasons why the numbers might be less today. Some of them are:

1. **Technical difficulties with website**: Checking if the website was down or were the pages broken, was there any maintenance scheduled, longer load times, Error 404.
2. **Recent website changes**: Checking if any recent changes were made to the website, any features added or deleted. Change in UI which is not very user friendly for some users.
3. **Server Overload**: Investigating if the web servers were overloaded which might have crashed the website for a few hours.
4. **Competition from other websites**: It might be possible the competitor website might be having a better and a greater number of tracks collection and more user-friendly UI. Monitoring and analyzing the competitors content marketing, social media activity.
5. **Monotonous**: Investigating whether any new tracks were uploaded or not. Are the tracks becoming repetitious, are users tired of listening to same old tracks?
6. **Poor promotion**: Poorly promoting a new uploaders track. Users not getting suggestions as per their music choices.

3. Visualization done in tableau with output in ppt file.

4. Some of the important KPIs to track and analyze the health of subscription business

1. Active Subscriber Count (ASC)
2. Churn
3. Average Revenue Per User (ARPU)
4. Monthly Recurring Revenue (MRR)
5. Customer Acquisition Cost (CAC)
6. Lifetime Value (LTV)

a. **Active Subscriber Count (ASC)**

An active subscriber is currently using the service over a given period of time, but customers can always cancel the service whenever they want. Important points to consider:

1. Most profitable subscriber
2. Subscriber most engaged with the service.
3. Who are most likely to stay and most likely to churn?

ASC is calculated from Net Subscriber Count (NSC) which is:

(No. of subscribers acquired during a period) – (No. of subscribers churned in same period).

b. **Churn**

The churn is the rate at which the subscribers leaves in a given month or year. Churn rate must be always less then new subscriber signing in. If we lose 1000 subscribers, we need to get more than 1000 subscribers.

Churn = (No. of subscribers left) / (Total subscribers) x 100

c. **Average Revenue Per User (ARPU)**

ARPU helps in determining the average revenue earned from each subscriber.

ARPU = (Total revenue) / (Total subscribers)

d. **Monthly Recurring Revenue (MRR)**

MRR is the total revenue from different subscription plans and is calculated by:

(Monthly revenue) x (No. of active subscribers)

e. **Customer Acquisition Cost (CAC)**

CAC is the total amount spent in running different marketing campaigns to acquire a single subscriber. CAC helps in understanding how effective our marketing strategies and marketing campaigns is.

CAC = (Amount spent on marketing) / New subscribers acquired)

We can use a more complex model by attributing costs at individual level like which campaign helped in acquiring the subscriber.

f. **Lifetime Value (LTV)**

It helps in predicting total business that would be received from the subscriber. It helps in determining what subscriber are most valuable based on recent purchases. The more the subscriber spends in a given period of time the higher will be its LTV. Higher LTV can justify spending more Customer Acquisition. Together LTV and CAC can help determine how long it takes to get back the investment required to acquire a customer.

LTV = (ARPU) x (Retention)