**Case Study 1 (Job Data)**

**Below is the structure of the table with the definition of each column that you must work on:**

* **Table-1:**job\_data  
  + **job\_id:**unique identifier of jobs
  + **actor\_id:**unique identifier of actor
  + **event:**decision/skip/transfer
  + **language:**language of the content
  + **time\_spent:**time spent to review the job in seconds
  + **org:**organization of the actor
  + **ds:**date in the yyyy/mm/dd format. It is stored in the form of text and we use presto to run. no need for date function

Use the dataset attached in the Dataset section below the project images then answer the questions that follows

1. **Number of jobs reviewed:**Amount of jobs reviewed over time.  
   **Your task:** Calculate the number of jobs reviewed per hour per day for November 2020?

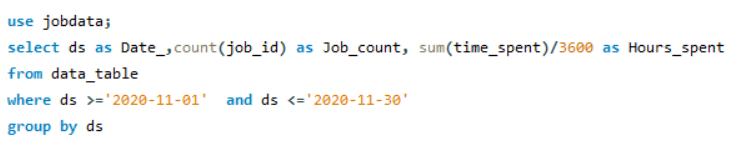
use jobdata;

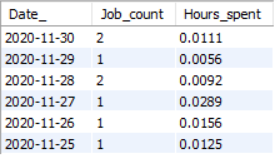
select ds as Date\_,count(job\_id) as Job\_count, sum(time\_spent)/3600 as Hours\_spent

from data\_table

where ds >='2020-11-01' and ds <='2020-11-30'

group by ds





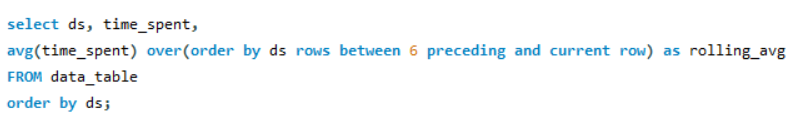
1. **Throughput:**It is the no. of events happening per second.  
   **Your task:** Let’s say the above metric is called throughput. Calculate 7 day rolling average of throughput? For throughput, do you prefer daily metric or 7-day rolling and why?

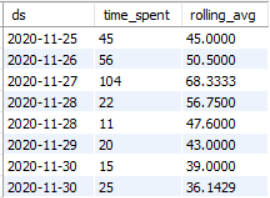
select ds, time\_spent,

avg(time\_spent) over(order by ds rows between 6 preceding and current row) as rolling\_avg

from data\_table

order by ds;





1. **Percentage share of each language:**Share of each language for different contents.  
   **Your task:** Calculate the percentage share of each language in the last 30 days?

Per day:

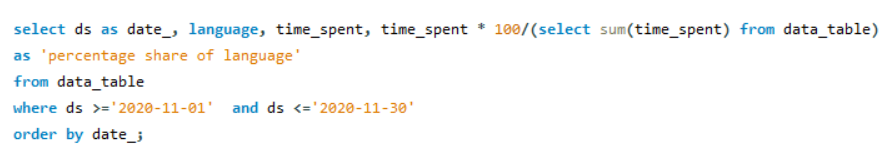
select ds as date\_, language, time\_spent, time\_spent \* 100/(select sum(time\_spent) from data\_table)

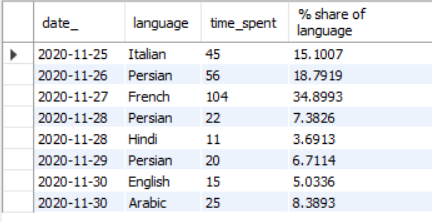
as 'percentage share of language'

from data\_table

where ds >='2020-11-01' and ds <='2020-11-30'

order by date\_;





Per month

SELECT language,

sum(time\_spent) as total\_time\_per\_language,

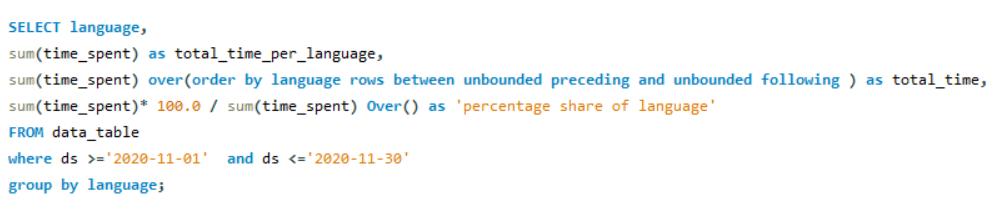
sum(time\_spent) over(order by language rows between unbounded preceding and unbounded following ) as total\_time,

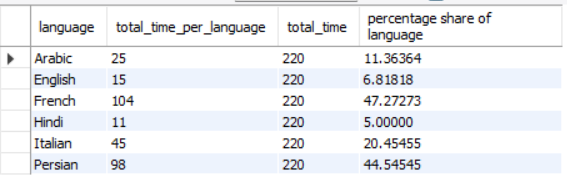
sum(time\_spent)\* 100.0 / sum(time\_spent) Over() as 'percentage share of language'

FROM data\_table

where ds >='2020-11-01' and ds <='2020-11-30'

group by language;





1. **Duplicate rows:**Rows that have the same value present in them.  
   **Your task:** Let’s say you see some duplicate rows in the data. How will you display duplicates from the table?

select \* from jobdata.data\_table;

use jobdata;

with cte as ( select \*,

row\_number() over (partition by ds, job\_id, actor\_id) as rownum from

data\_table)

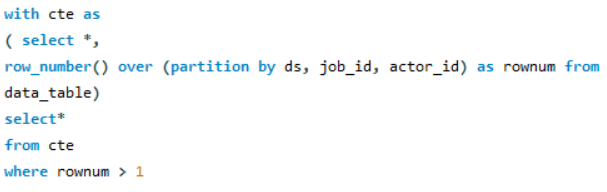
select\*

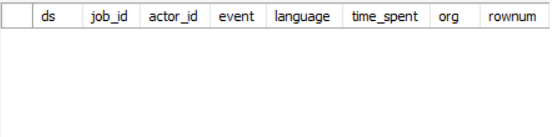
from

cte

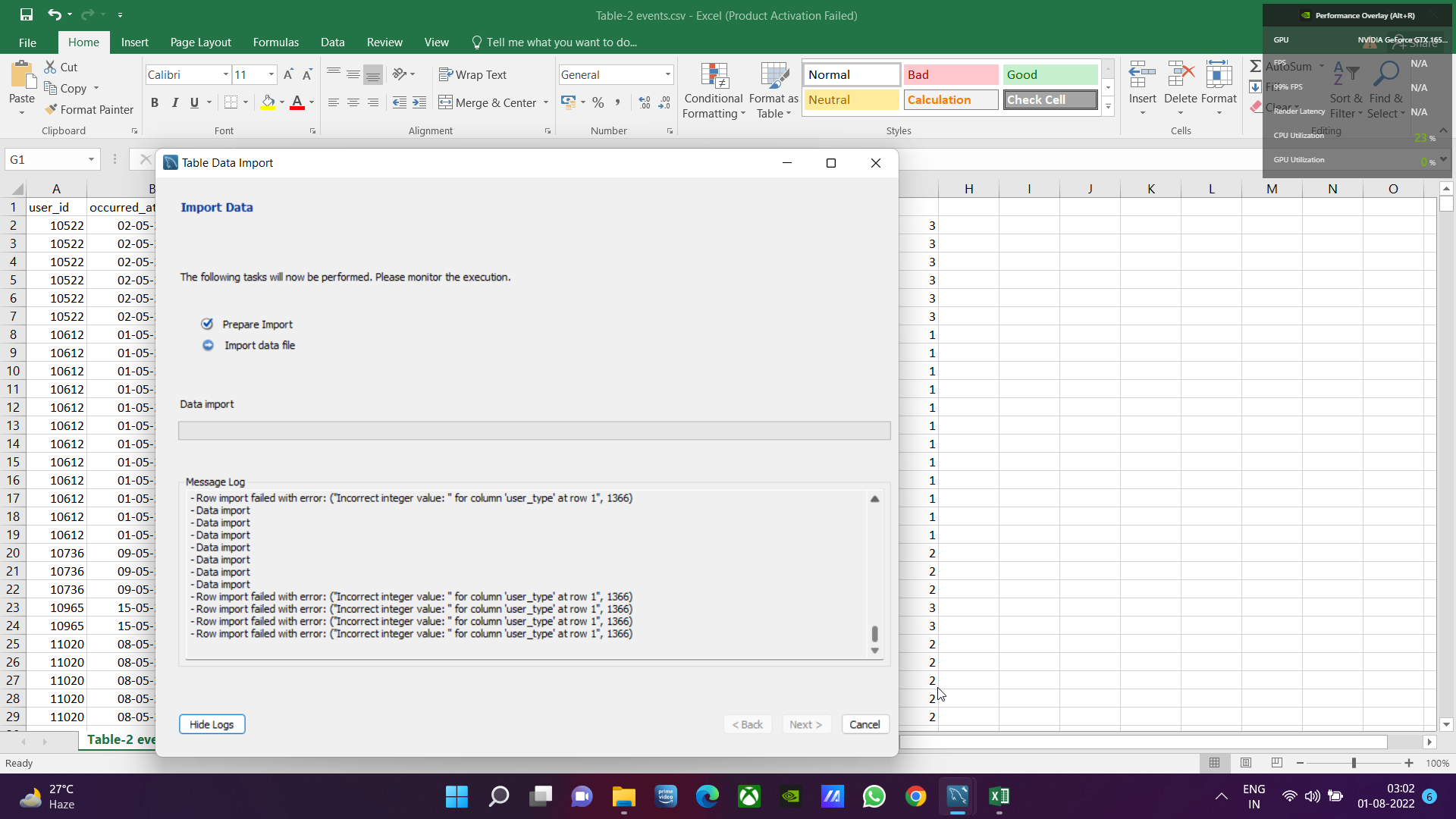
where

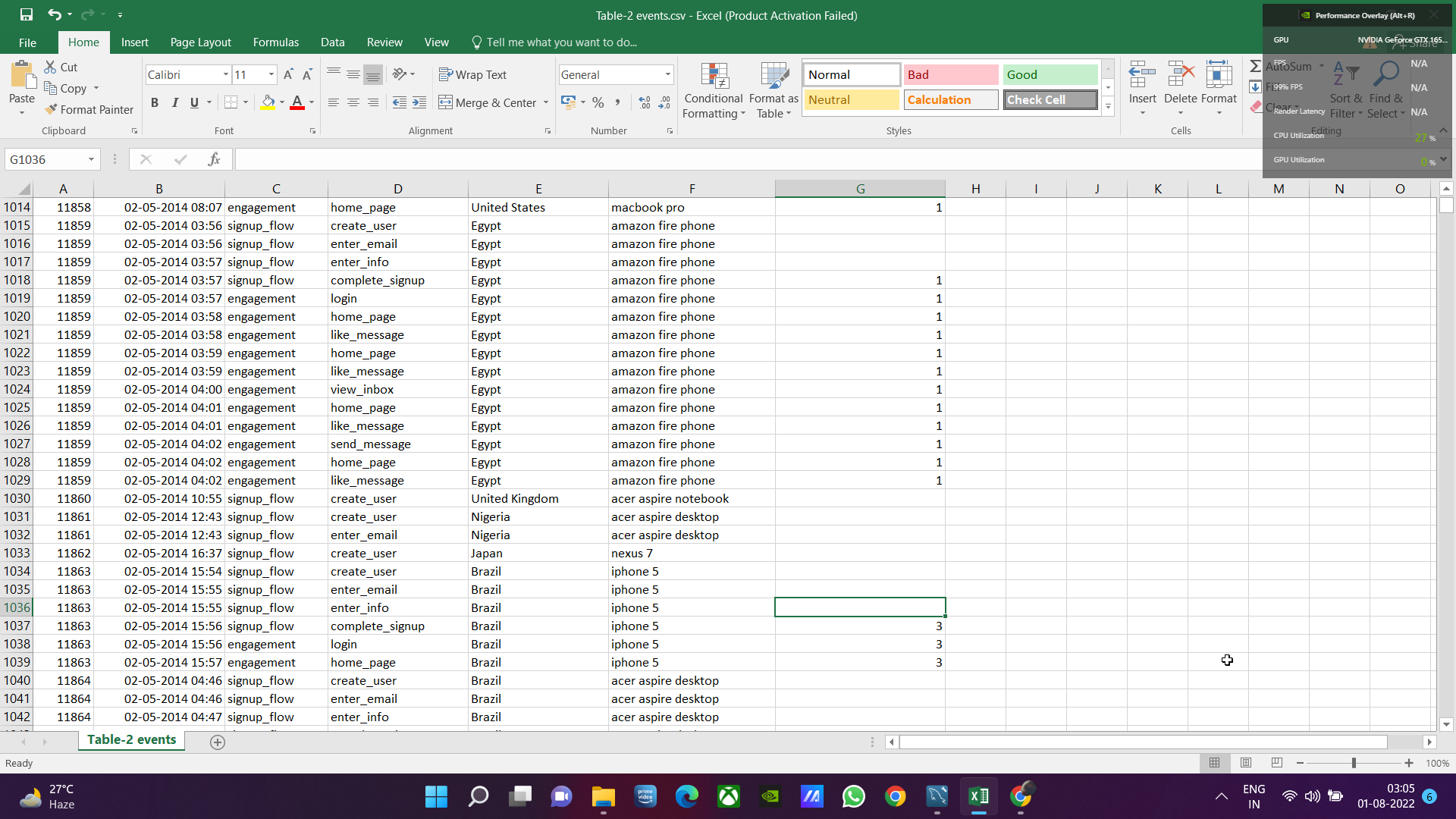
rownum > 1





**Case Study 2 (Investigating metric spike)**



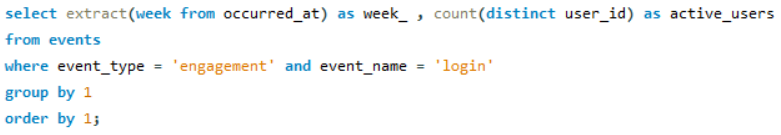


**The structure of the table with the definition of each column that you must work on is present in the project image**

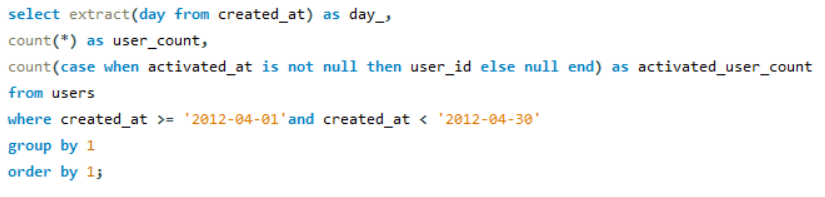
* **Table-1:**users  
  This table includes one row per user, with descriptive information about that user’s account.
* **Table-2:**events  
  This table includes one row per event, where an event is an action that a user has taken. These events include login events, messaging events, search events, events logged as users progress through a signup funnel, events around received emails.
* **Table-3:**email\_events  
  This table contains events specific to the sending of emails. It is similar in structure to the events table above.

Use the dataset attached in the Dataset section below the project images then answer the questions that follows

1. **User Engagement:**To measure the activeness of a user. Measuring if the user finds quality in a product/service.  
   **Your task:** Calculate the weekly user engagement?



1. **User Growth:**Amount of users growing over time for a product.  
   **Your task:** Calculate the user growth for product?



1. **Weekly Retention:**Users getting retained weekly after signing-up for a product.  
   **Your task:** Calculate the weekly retention of users-sign up cohort?

select date trunc (week', z.occurred at) as "week", avg (2.age at event) as "average age durig week",

count (distinct case when z.user\_age > 70 then z.user\_id else null end) as 10+ weeks", count (distinct case when z.user\_age < 70 and z.user\_age >=63 then 2.user id else null end) as 9 weeks", count (distinct case when 2.user\_age < 63 and 2.user age >-56

then z.user\_id else null end) as 8 weeks", count (distinct case when 2.user\_age < 56 and 2.user\_age >-49 then 2.user id else null end) as 17 weeks",

count (distinct case when z.user\_age 49 and z.user\_age >=42 then z.user\_id else null end) as 16 weeks",

count (distinct case when z.user\_age < 42 and 2.user\_age >=35

then 2.user\_id else null end) as 5 weeks",

count (distinct case when 2.user\_age 35 and 2.user\_age >=28 then z.user\_id else null end) as 4 weeks",

count (distinct case when z.user\_age < 28 and 2.user\_age >=21 then 2.user\_id else null end) as 3 weeks",

count (distinct case when 2.user\_age < 21 and 2.user age >-14

then 2.user\_id else null end) as 12 weeks",

count (distinct case when z.user\_age < 14 and z.user\_age >=7 then 2.user\_id else null end) as 1 weeks",

count (distinct case when z.user\_age < 7 and .user\_age >=63

then 2.user\_id else null end) as less than a week",

from ( select e.occurred at, u.user\_id, date trunc ("week", u.activated\_at) as activation week, extract('day' from e.occurred at u.activated at) as age at event, extract('day' from '201-09-01::timestamp -u.activated\_at) as user\_age

from tutorial.yammer users u

join tutorial.yammer events e on e.user\_id = u.user\_id

and e.event\_type = 'engagement

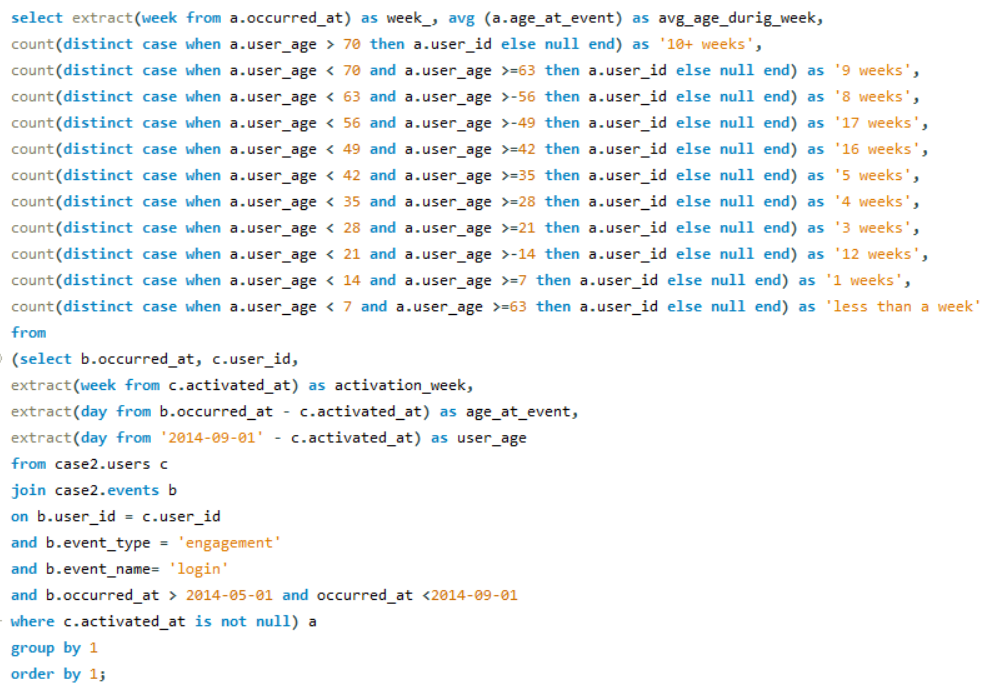
and e.evnetn\_name= 'login' and e.occurred at > 2014-05-01

and e.occurred at <2014-09-01 where u.activated at is not null

group by 1

order by 1

limit 100



1. **Weekly Engagement:**To measure the activeness of a user. Measuring if the user finds quality in a product/service weekly.  
   **Your task:** Calculate the weekly engagement per device?

select extract(week from occurred\_at) as week\_,

count(distinct user\_id) as weekly\_active\_users,

count(distinct case when device in

('macbook pro', 'lenovo thinkpad', 'macbook air',' dell inspiron notebook', 'asus chromebook', 'dell inspiron desktop',

'acer aspire notebook', 'hp pavilion desktop', 'acer aspire desktop', 'mac mini') then user\_id else null end) as computer,

count(distinct case when device in('iphone 5','samsung galaxy s4', 'nexus 5','iphone 5s', 'iphone 4s', 'nokia lumia 635',

'htc one','samsung galaxy note', 'amazon fire phone') then user\_id else null end) as phone,

count(distinct case when device in('ipad air', 'nexus 7','ipad mini', 'nexus 10', 'kindle fire',

'windows surface', 'samsung galaxy tablet') then user\_id else null end) as tablet

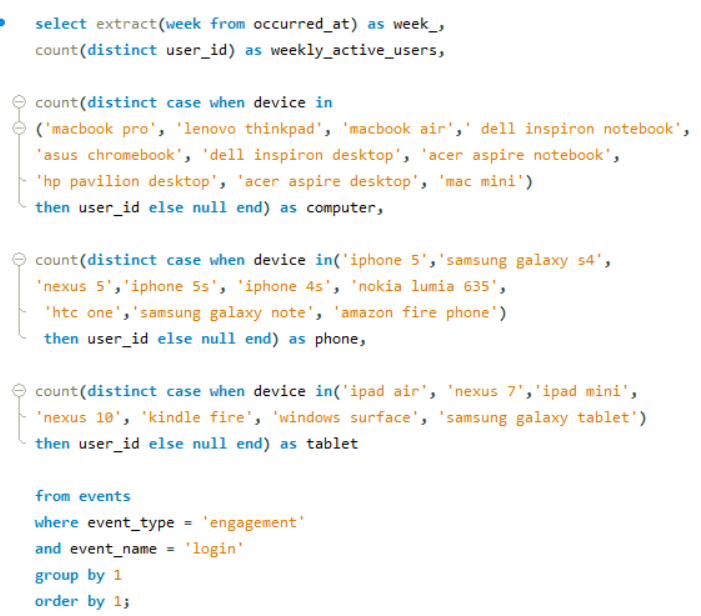
from events

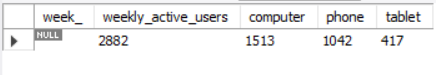
where event\_type = 'engagement'

and event\_name = 'login'

group by 1

order by 1;





1. **Email Engagement:**Users engaging with the email service.  
   **Your task:** Calculate the email engagement metrics?

select extract(week from occurred\_at) as week\_,

count(case when action = 'sent\_weekly\_digest' then user\_id else null end) as weekly\_email\_count,

count(case when action= 'email\_open' then user\_id else null end) as email\_open\_count,

count(case when action = 'email\_clickthrough' then user\_id else null end) as email\_clickthroughs\_count

from email\_events

group by 1;

