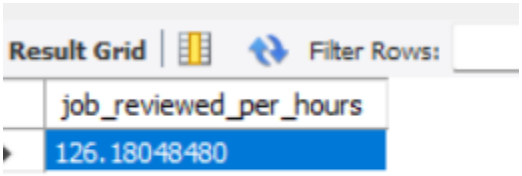
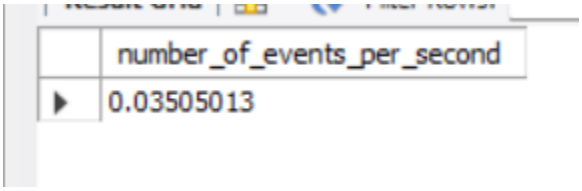
```
create view calc as(
select count(job_id) as total_jobs,sum(time_spent) as total_sec,ds
from job_data
group by ds);

select avg((total_jobs/ total_sec)*3600) as job_reviewed_per_hours
from calc;
```

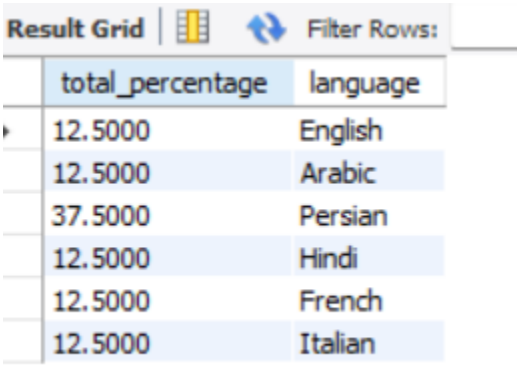


```
select avg((total_jobs/total_sec)) as number_of_events_per_second
from calc;
```



number_of_events_per_second
0.03505013

```
select count(language)/total*100 as total_percentage,language
from job_data
cross join
( select count(*)as total
from job_data ) as p
group by language,p.total;
```



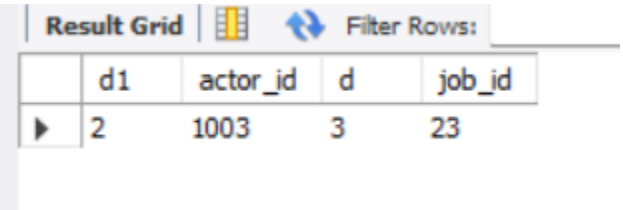
Result Grid | Filter Rows:

total_percentage	language
12.5000	English
12.5000	Arabic
37.5000	Persian
12.5000	Hindi
12.5000	French
12.5000	Italian

```
create view dupli as(
select count(actor_id) as Number_of_duplicate_of_actor_id,actor_id
from job_data
group by actor_id
having Number_of_duplicate_of_actor_id>1);

create view dupli_2 as (
select count(job_id) as Number_of_duplicate_of_job_id, job_id
from job_data
group by job_id
having Number_of_duplicate_of_job_id);

select * from dupli,dupli_2;
```



Result Grid | Filter Rows:

	d1	actor_id	d	job_id
▶	2	1003	3	23

Case Study 2: Investigating Metric Spike

Weekly user engagement.

```
select extract(week from occurred_at) as week,count(distinct user_id) as engagement
from events
where event_type ='engagement'
group by week;
```

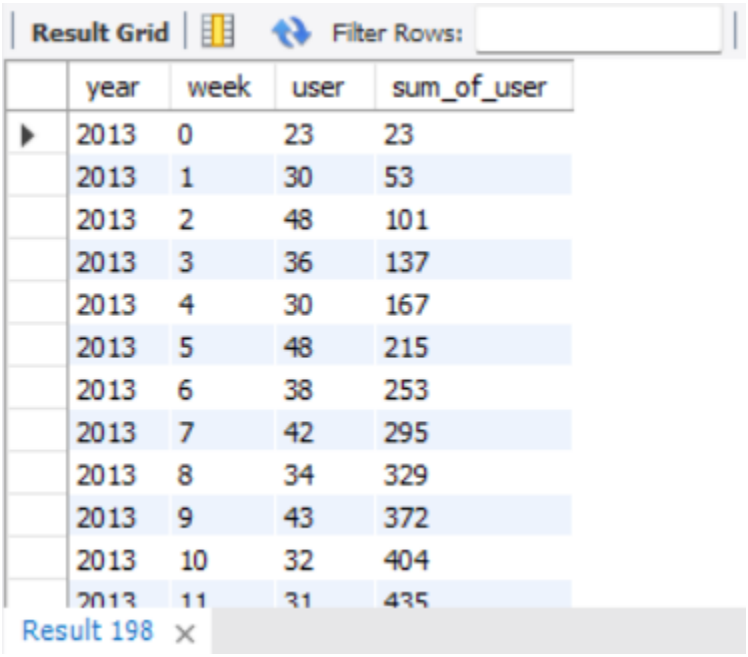
Result Grid | Filter Rov

week	engagement
17	663
18	1068
19	1113
20	1154
21	1121
22	1186
23	1232
24	1275
25	1264
26	1302
27	1372
28	1365

Result 195

User growth for the product.

```
select year,week,user,sum(user) over (order by year,week) as sum_of_user
from(
select extract(year from created_at) as year,extract(week from created_at) as week,count(distinct user_id) as user
from users
where state ='active'
group by year,week) a;
```



Result Grid | Filter Rows:

year	week	user	sum_of_user
2013	0	23	23
2013	1	30	53
2013	2	48	101
2013	3	36	137
2013	4	30	167
2013	5	48	215
2013	6	38	253
2013	7	42	295
2013	8	34	329
2013	9	43	372
2013	10	32	404
2013	11	31	435

Result 198

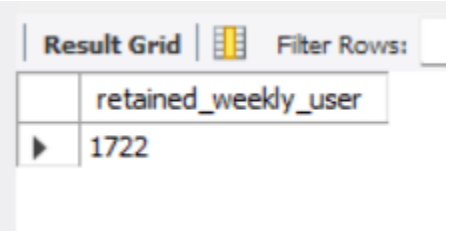
```
create view T1 as (
select user_id,extract(week from occurred_at) as weekly_signup
from events
where event_type = 'signup_flow' and event_name = 'complete_signup');

create view T2 as (
select user_id, extract(week from occurred_at) as weekly_engagement
from events
where event_type = 'engagement');

create view TT1 as (select T2.user_id,weekly_engagement-weekly_signup as ret
from T1,T2
where T2.user_id=T1.user_id);

with temp as (select user_id,count(ret) as weekly_retained_user
from TT1
where ret>1
group by user_id)

select count(weekly_retained_user) as retained_weekly_user
from temp;
```



the weekly engagement per device.

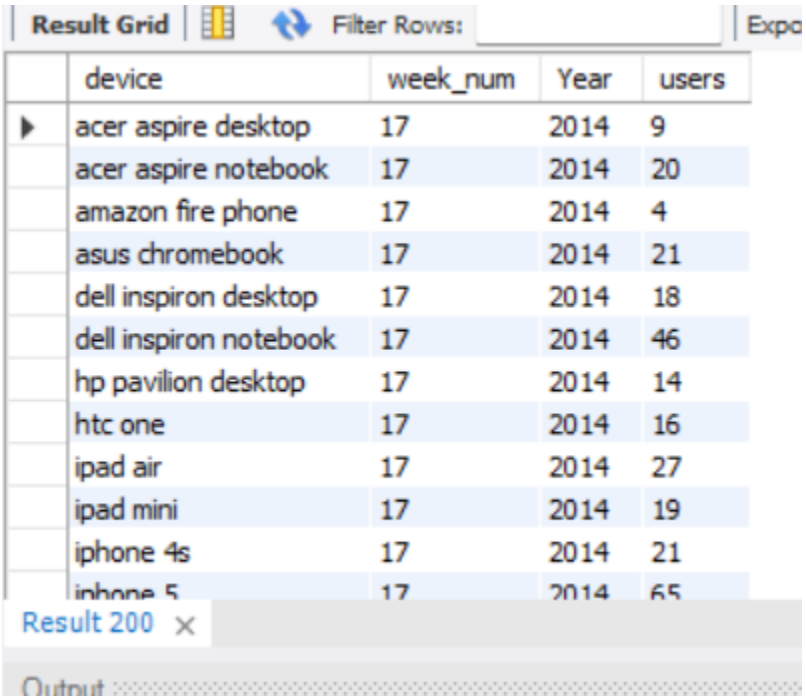
```
select distinct device,extract(week from occurred_at) as week_num,extract(year from occurred_at) as
Year,count(distinct user_id) as users

from events

where event_type ='engagement'

group by week_num,device, Year

order by week_num;
```



Result Grid | Filter Rows: | Expo

device	week_num	Year	users
acer aspire desktop	17	2014	9
acer aspire notebook	17	2014	20
amazon fire phone	17	2014	4
asus chromebook	17	2014	21
dell inspiron desktop	17	2014	18
dell inspiron notebook	17	2014	46
hp pavilion desktop	17	2014	14
htc one	17	2014	16
ipad air	17	2014	27
ipad mini	17	2014	19
iphone 4s	17	2014	21
iphone 5	17	2014	65

Result 200 ×

Output

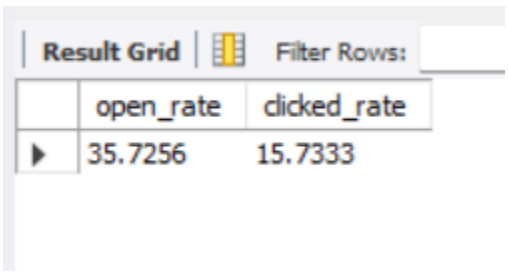
the email engagement metrics.

```
create view total_mail as (select
count(action) as total_email_sent
from email_events
where action ='sent_weekly_digest');

create view total_mail_opened as(
select count(action) as mail_opened
from email_events
where action ='email_open'
);
create view total_mail_email_clickthrough as (
select count(action) as clicked
from email_events
where action ='email_clickthrough');


with cte1 as(
select total_mail_opened.mail_opened/total_mail.total_email_sent *100 as open_rate,
 total_mail_email_clickthrough.clicked/total_mail.total_email_sent *100 as clicked_rate
 from total_mail,total_mail_opened,total_mail_email_clickthrough)

 select * from cte1;
```



Description

The project is based on Operation Analysis i.e., to perform end to end operations for company growth and which areas to improve on. I am going to find the following tasks.

I. The number of jobs reviewed per hour per day for November 2020.

II. 7 day rolling average of throughput.

III. The Percentage share of each language in the last 30 days.

IV. Displaying Duplicates.

V. The weekly user engagement.

VI. The user growth for the product. VII. The weekly retention of users-sign up cohort.

VIII. To measure the activeness of a user. IX. The email engagement metrics. From the Provided Database.

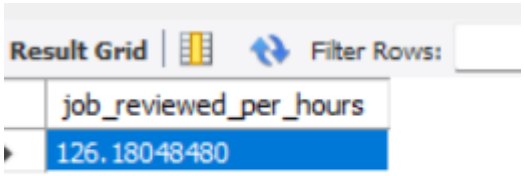
APPROACH:

1. I gather the information from the description provided i.e., which tasks I need to complete.
2. Using MYSQL Workbench I created new files and I started writing my queries to achieve the result.
3. Later I executed the Queries and if there are any errors in the code, I modified the code and fixed the code without any errors.
4. I Revised the code once after completion of execution.
5. Finally, I attached my code to the file.

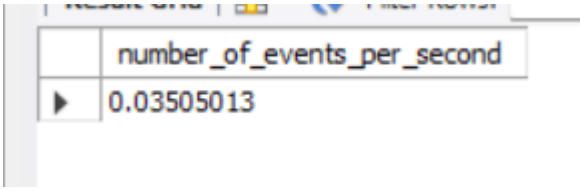
TECH STACK USED:

❖ SQL ❖ Development tool – MYSQL Workbench version 8.0.30. The main purpose of using MySQL workbench is that it provides the console to simply editable and administer the MYSQL environments and to gain better results and insights of the data. It provides data modeling, SQL development and connecting servers and is the best tool to design, generate and manage the databases.

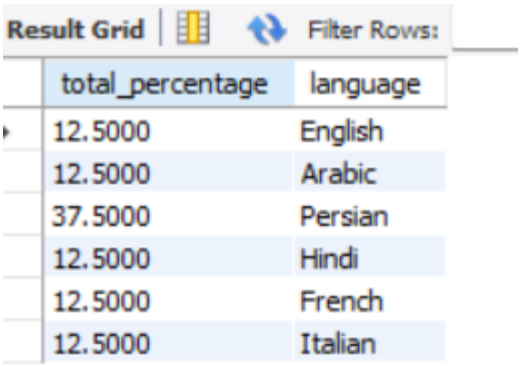
Insights



1- Number of jobs reviewed per hours stood to 126 as compared to standard laid out by management.



2- Number of jobs reviewed per hours stood to 126 as compared to standard laid out by management.



total_percentage	language
12.5000	English
12.5000	Arabic
37.5000	Persian
12.5000	Hindi
12.5000	French
12.5000	Italian

3- Persian language has the highest share among all other language

Result Grid | Filter Rows:

d1	actor_id	d	job_id
2	1003	3	23

5- **actor _id '1003' has 2 duplicates**
6- **Job_id – '23' has 3 duplicates**
7- **Weekly engagement is as follow**

Result Grid | Filter Rov

week	engagement
17	663
18	1068
19	1113
20	1154
21	1121
22	1186
23	1232
24	1275
25	1264
26	1302
27	1372
28	1365

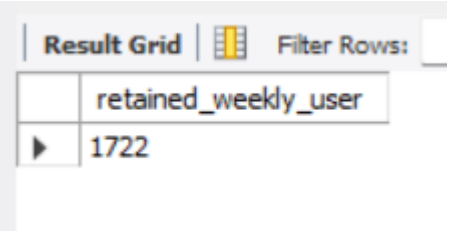
Result 195 ×

8- **weekly user growth is as follow**

Result Grid | Filter Rows:

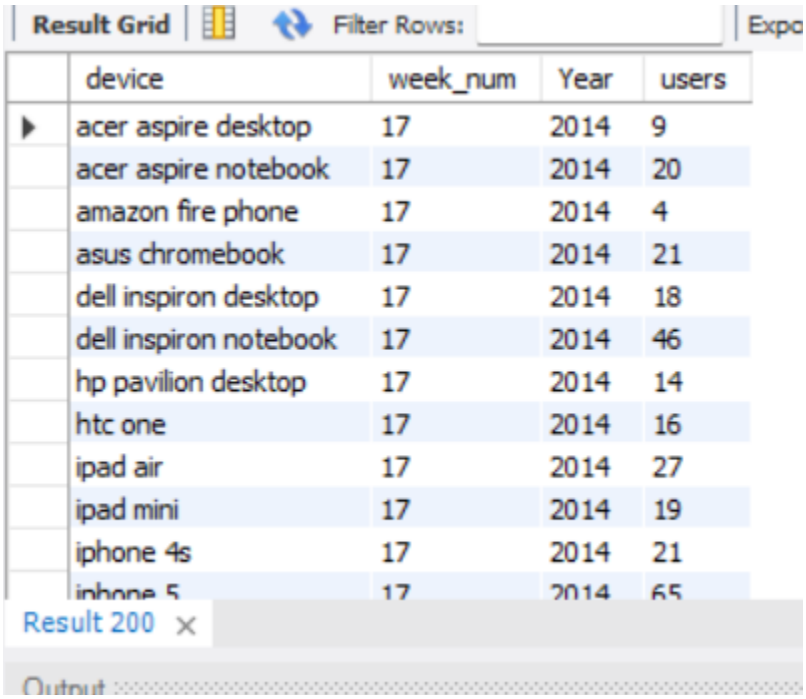
year	week	user	sum_of_user
2013	0	23	23
2013	1	30	53
2013	2	48	101
2013	3	36	137
2013	4	30	167
2013	5	48	215
2013	6	38	253
2013	7	42	295
2013	8	34	329
2013	9	43	372
2013	10	32	404
2013	11	31	435

Result 198 ×



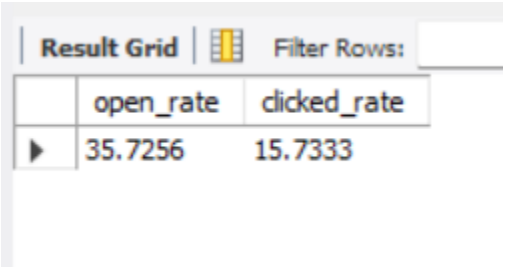
	retained_weekly_user
▶	1722

9 -weekly retained users stood to 1722 as compared total engaged users



	device	week_num	Year	users
▶	acer aspire desktop	17	2014	9
	acer aspire notebook	17	2014	20
	amazon fire phone	17	2014	4
	asus chromebook	17	2014	21
	dell inspiron desktop	17	2014	18
	dell inspiron notebook	17	2014	46
	hp pavilion desktop	17	2014	14
	htc one	17	2014	16
	ipad air	17	2014	27
	ipad mini	17	2014	19
	iphone 4s	17	2014	21
	iphone 5	17	2014	65

10- weekly users as per their devices and among all devices most are from those who are using MacBook pro.



	open_rate	clicked_rate
▶	35.7256	15.7333

11- In email engagement services

Open rate stood to - 35%

Clicked Rate stood to – 15%

9-

Result.

After working on and completing this project, My achievement is that I was able to perform the tasks and provide valid and precise insights to the team. I believe that I have done justice to the data that was demanded, performing all the mentioned operations on the data, collecting insights and answering the questions