OPERATIONAL ANALYTICS AND INVESTIGATING METRIC

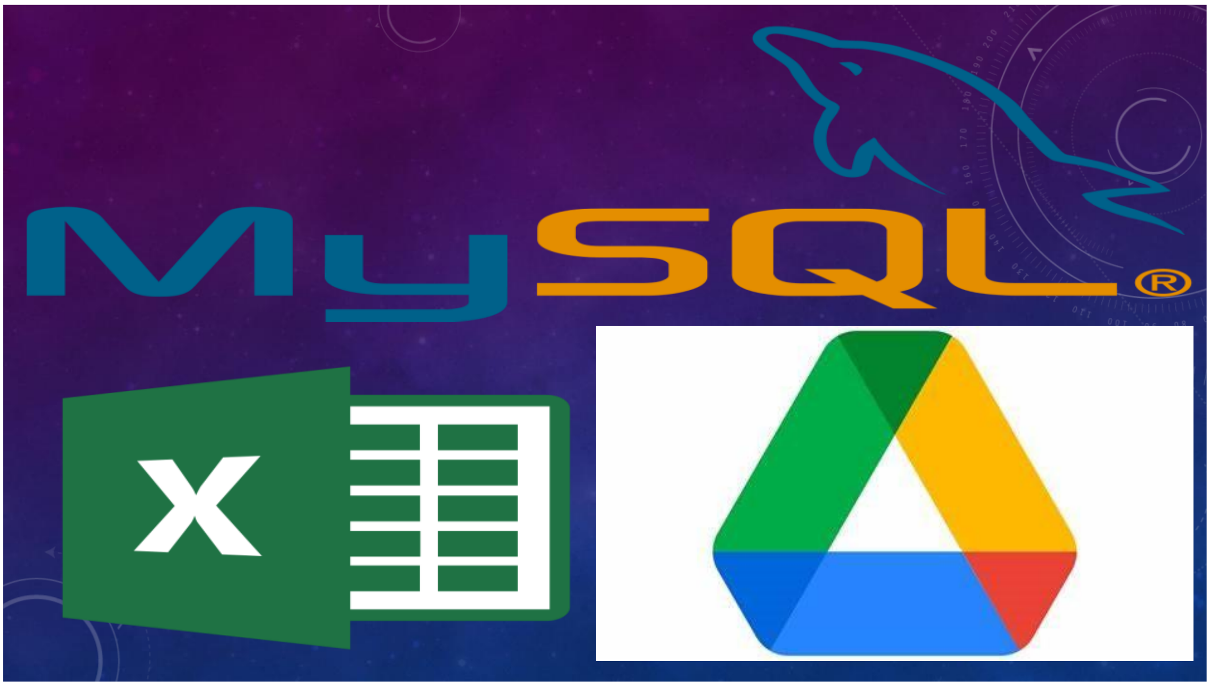
PROJECT DESCRIPTION: The objective of the project is to analyse the company's end to end operations which helps identifying the area for improvement. As the Data Analyst of Microsoft, I will work closely with the marketing, operation and support team to derive valuable insights from the collected data source. Th e project also focuses on investing metric spikes in daily engagement and sales, investing their causes and providing insights to address any dip in these metrices

APPROACH:

1. Project involves understanding and gathering data sets and tables from various sources and departments within company
2. Making sure that the given variable and attribute can be understood properly
3. Importing files into my SQL Workbench and through SQL the collected data will be analysed to give insights and patterns
4. Key performance metrices are such as daily engagement, sales and customer satisfaction will be identified in collaboration with cross-functional teams
5. Cross checking the queries one or two times so that the code will run without errors
6. Once the questions are answered it will be saved with screenshot
7. Finally, it will be added to the document

TECH-STACK USED:

8. MY SQL Community server-GPL Version 8.0.29 and connector version c++8.0.29 for creating my project as we know MY SQL community server is an open-source relational database management system that uses SQL



INSIGHTS:

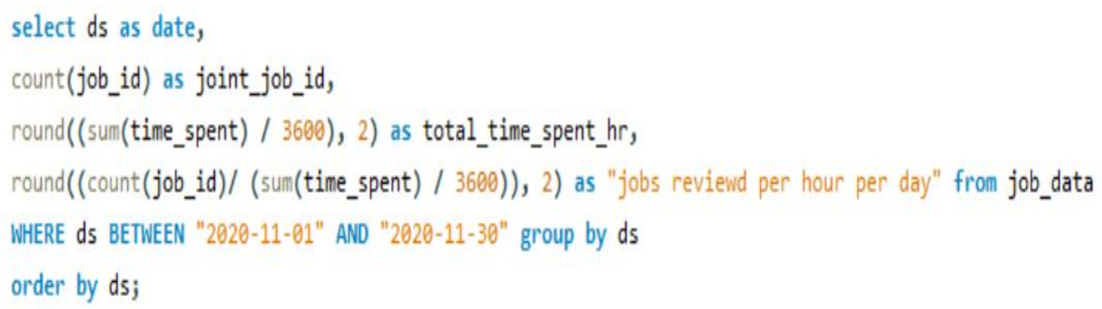
- While making the project I learn about SQL how to implement the queries and about the various built-in functions that can be used to get desired output
- I got good bit of exposure about SQL and the functions of that can be used in analysing data from the dataset and also the queries how they worked and executed.

RESULTS:

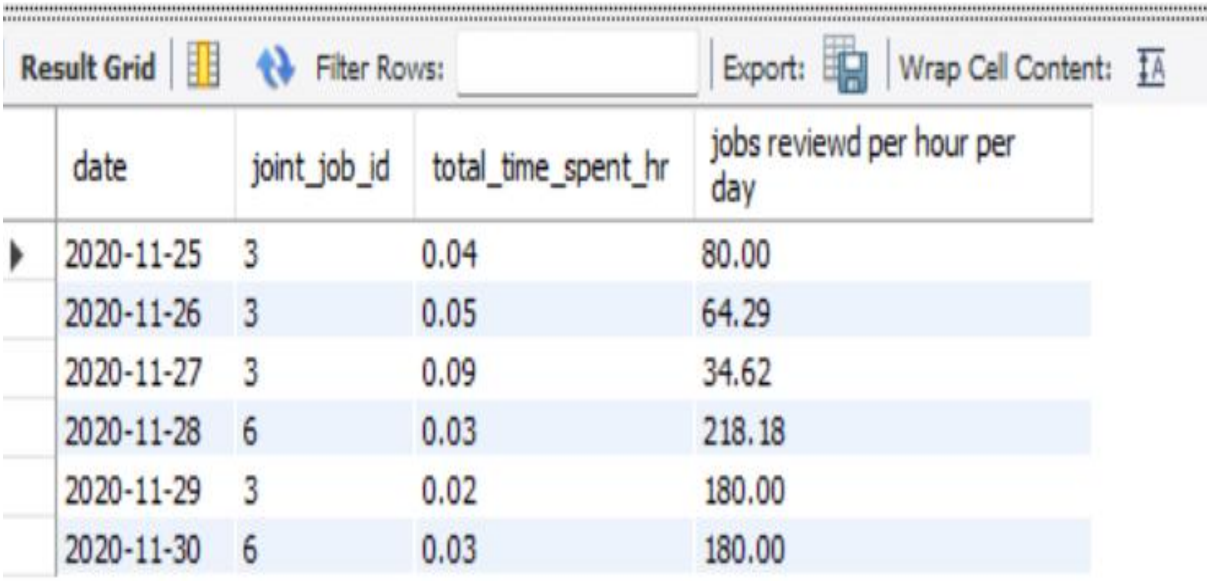
CASE STUDY 1: JOB DATA ANALYSIS

A. JOBS REVIEWED OVER TIME:

Task: Write an SQL query to calculate the number of jobs reviewed per hour for each day in November 2020.



```
select ds as date,
count(job_id) as joint_job_id,
round((sum(time_spent) / 3600), 2) as total_time_spent_hr,
round((count(job_id)/ (sum(time_spent) / 3600)), 2) as "jobs reviewd per hour per day" from job_data
WHERE ds BETWEEN "2020-11-01" AND "2020-11-30" group by ds
order by ds;
```

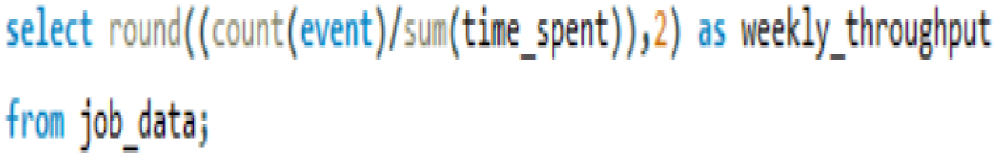


date	joint_job_id	total_time_spent_hr	jobs reviewd per hour per day
2020-11-25	3	0.04	80.00
2020-11-26	3	0.05	64.29
2020-11-27	3	0.09	34.62
2020-11-28	6	0.03	218.18
2020-11-29	3	0.02	180.00
2020-11-30	6	0.03	180.00

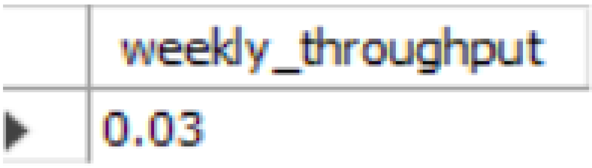
B. THROUGHPUT ANALYSIS:

Task: Write an SQL query to calculate the 7-day rolling average of throughput. Additionally, explain whether you prefer using the daily metric or the 7-day rolling average for throughput, and why.

WEEKLY THROUGHPUT:

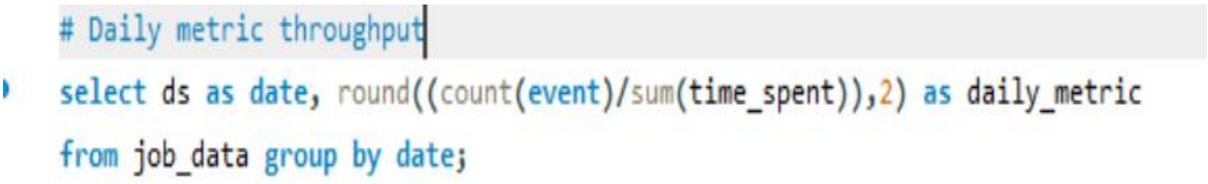


```
select round((count(event)/sum(time_spent)),2) as weekly_throughput
from job_data;
```

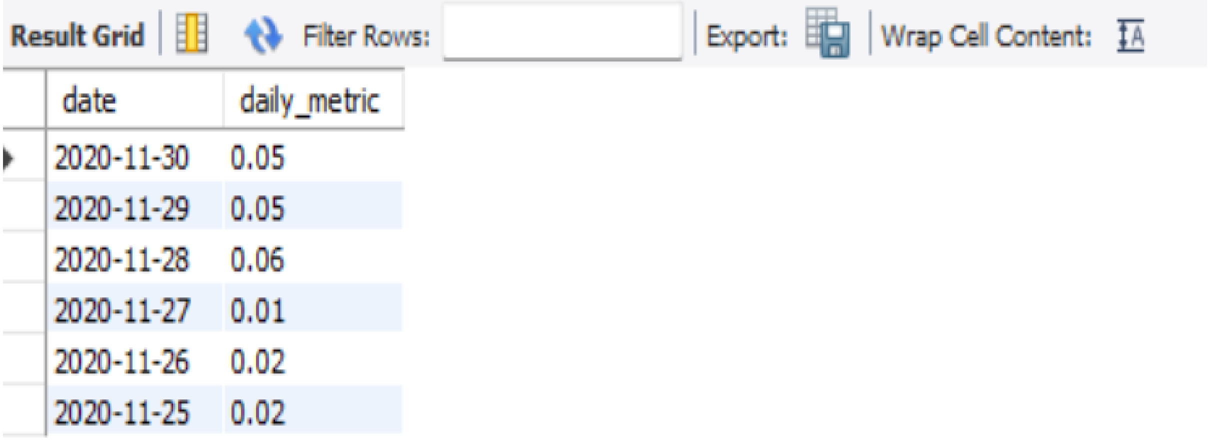


	weekly_throughput
▶	0.03

DAILY THROUGHPUT:



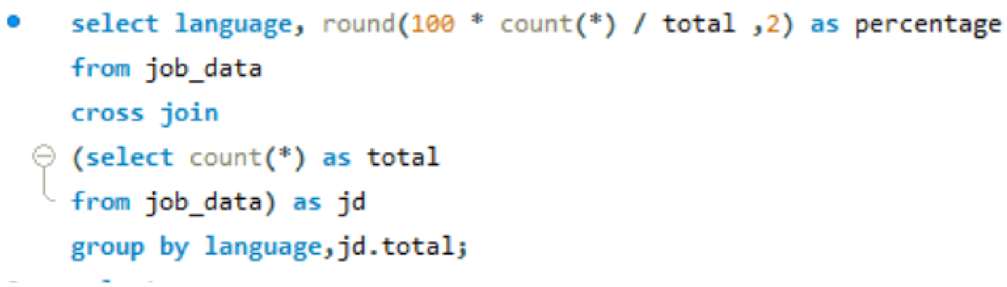
```
# Daily metric throughput
select ds as date, round((count(event)/sum(time_spent)),2) as daily_metric
from job_data group by date;
```



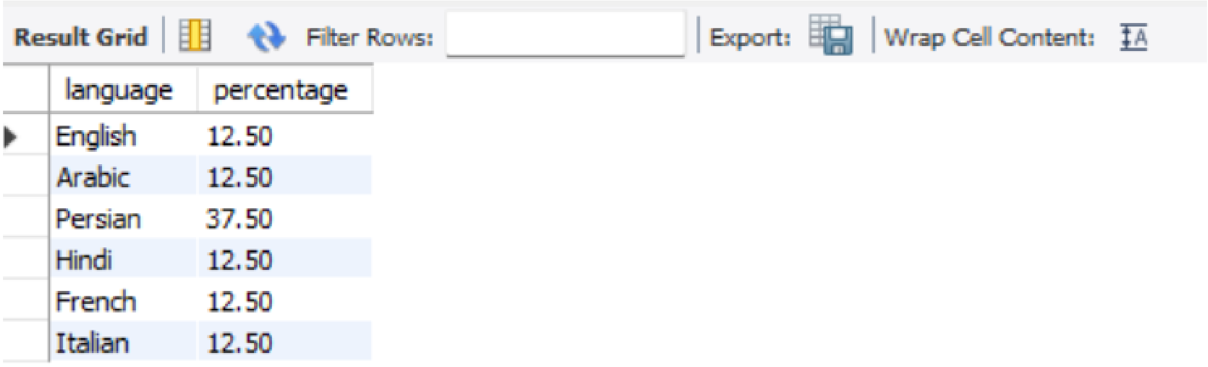
	date	daily_metric
▶	2020-11-30	0.05
	2020-11-29	0.05
	2020-11-28	0.06
	2020-11-27	0.01
	2020-11-26	0.02
	2020-11-25	0.02

C. **LANGUAGE SHARE ANALYSIS:**

Task: Write an SQL query to calculate the percentage share of each language over the last 30 days.



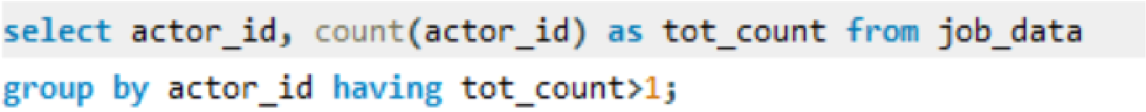
```
select language, round(100 * count(*) / total ,2) as percentage
from job_data
cross join
(select count(*) as total
from job_data) as jd
group by language,jd.total;
```



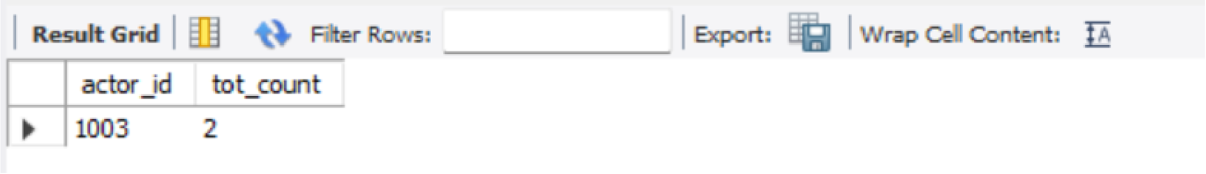
language	percentage
English	12.50
Arabic	12.50
Persian	37.50
Hindi	12.50
French	12.50
Italian	12.50

D. **DUPLICATE ROWS DETECTION**

Task: Write an SQL query to display duplicate rows from the job_data table.



```
select actor_id, count(actor_id) as tot_count from job_data
group by actor_id having tot_count>1;
```

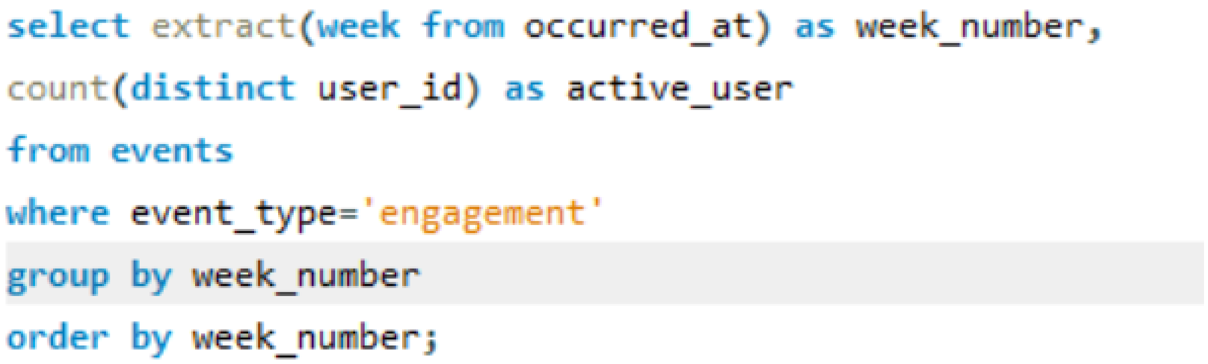


actor_id	tot_count
1003	2

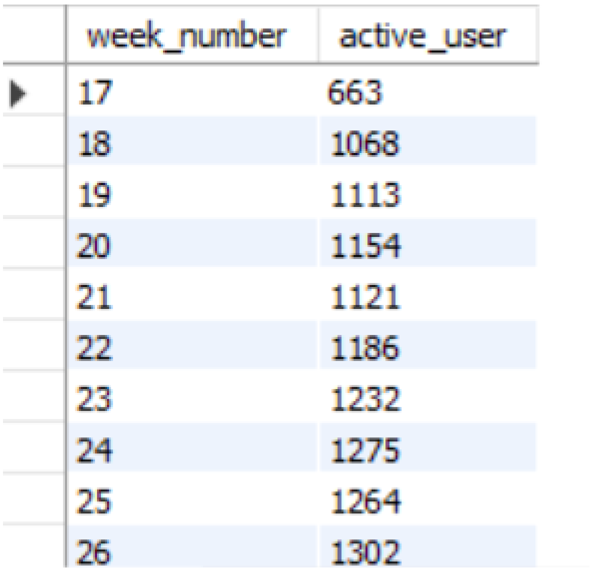
Case Study 2: INVESTIGATING METRIC SPIKE

1. WEEKLY USER ENGAGEMENT:

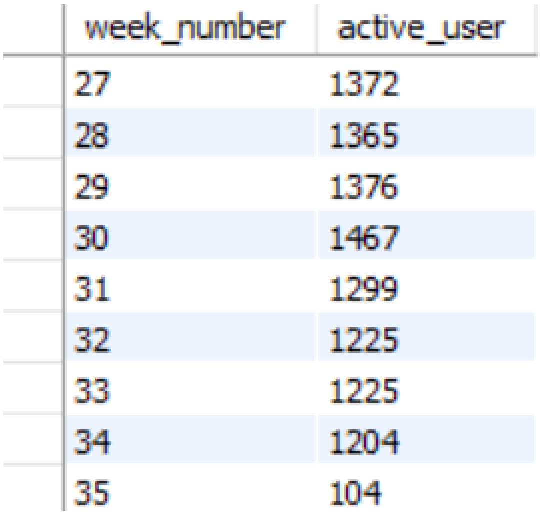
Task: Write an SQL query to calculate the weekly user engagement.



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



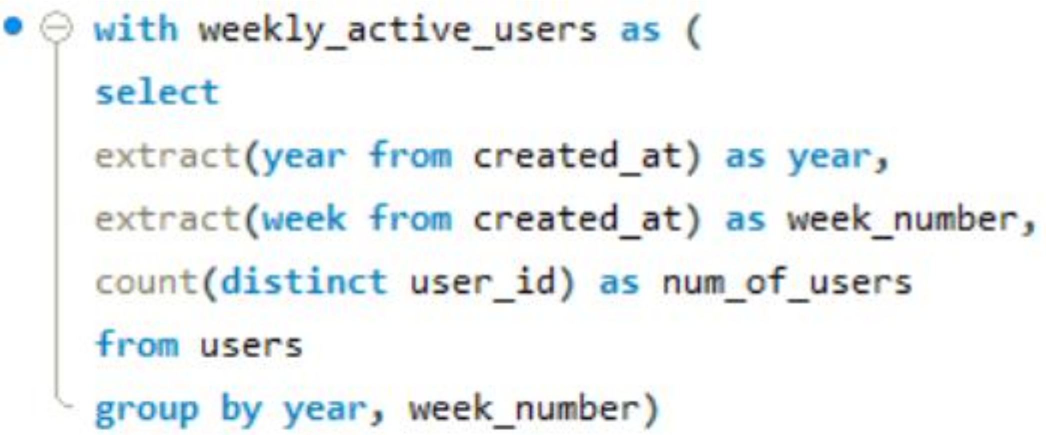
week_number	active_user
17	663
18	1068
19	1113
20	1154
21	1121
22	1186
23	1232
24	1275
25	1264
26	1302



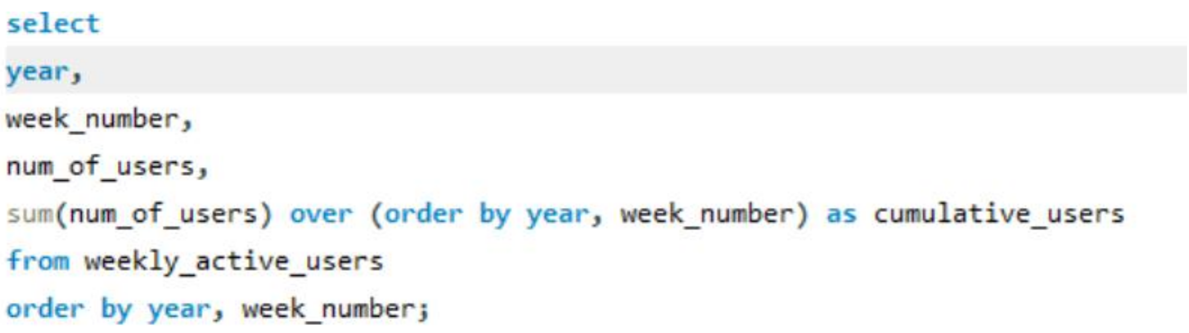
week_number	active_user
27	1372
28	1365
29	1376
30	1467
31	1299
32	1225
33	1225
34	1204
35	104

2. USER GROWTH ANALYSIS:

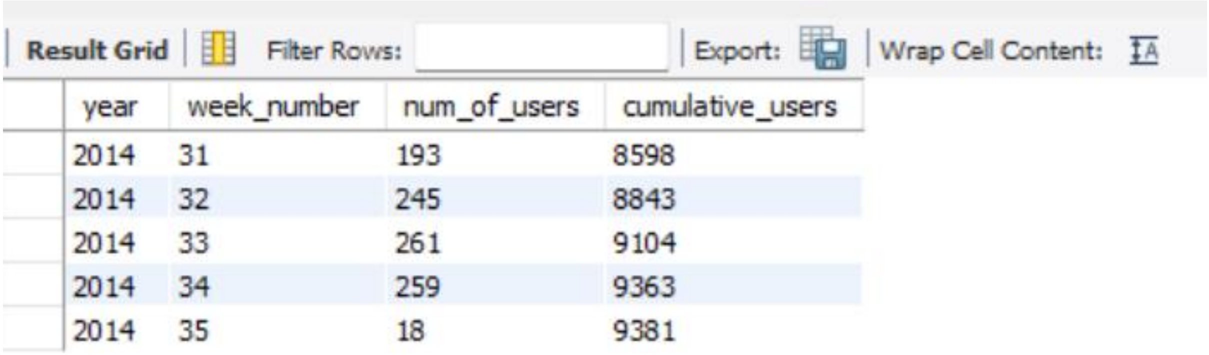
Task: Write an SQL query to calculate the user growth for the product.



```
with weekly_active_users as (
select
extract(year from created_at) as year,
extract(week from created_at) as week_number,
count(distinct user_id) as num_of_users
from users
group by year, week_number)
```



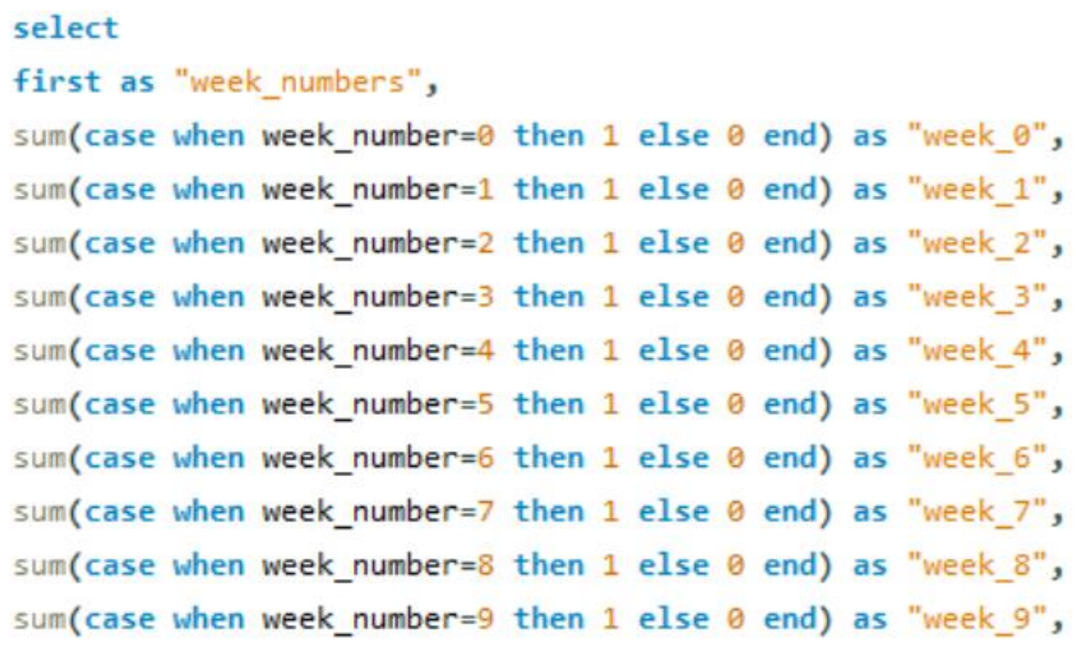
```
select
year,
week_number,
num_of_users,
sum(num_of_users) over (order by year, week_number) as cumulative_users
from weekly_active_users
order by year, week_number;
```



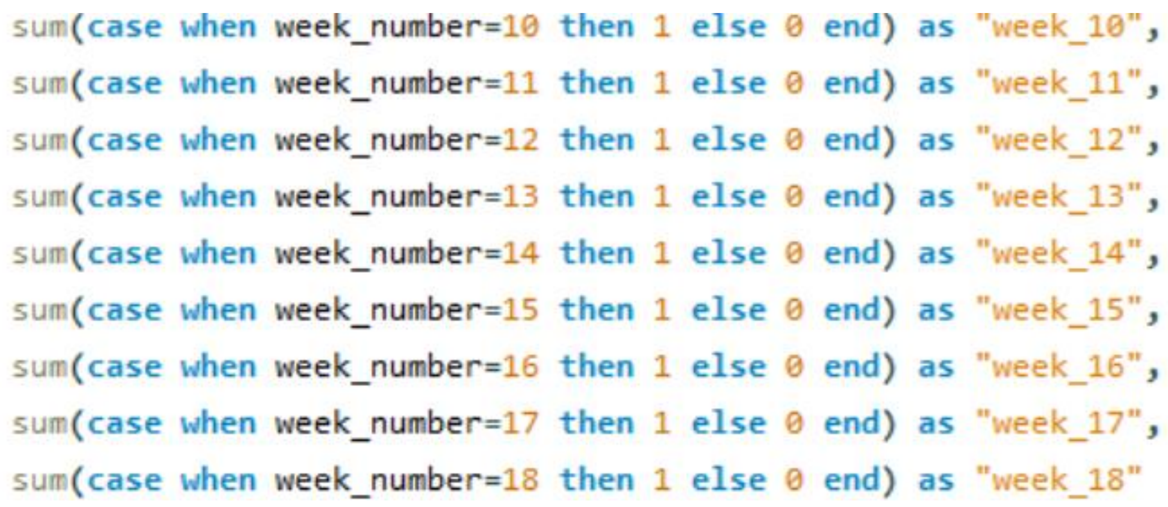
year	week_number	num_of_users	cumulative_users
2014	31	193	8598
2014	32	245	8843
2014	33	261	9104
2014	34	259	9363
2014	35	18	9381

3. WEEKLY RETENTION ANALYSIS:

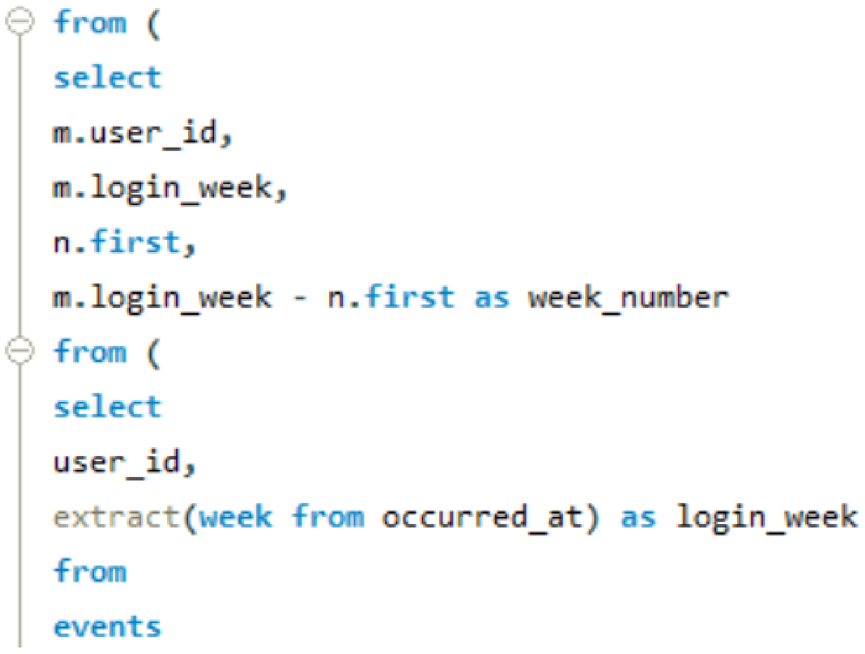
Your Task: Write an SQL query to calculate the weekly retention of users based on their sign-up cohort.



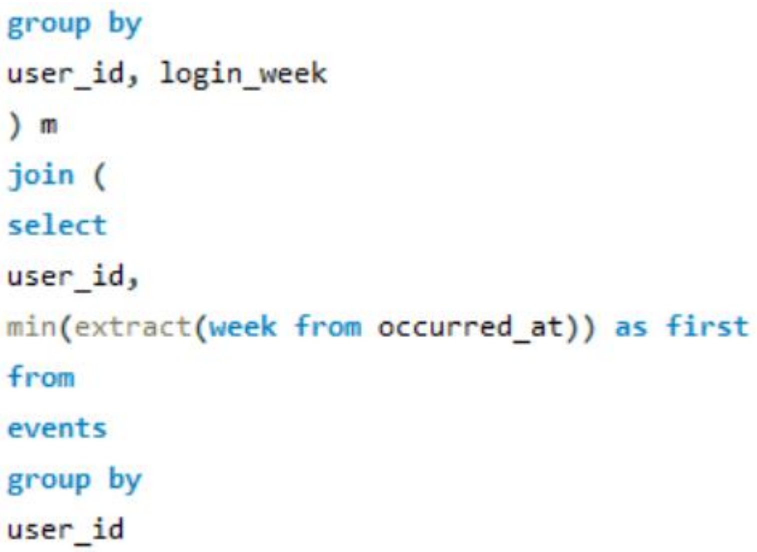
```
select
first as "week_numbers",
sum(case when week_number=0 then 1 else 0 end) as "week_0",
sum(case when week_number=1 then 1 else 0 end) as "week_1",
sum(case when week_number=2 then 1 else 0 end) as "week_2",
sum(case when week_number=3 then 1 else 0 end) as "week_3",
sum(case when week_number=4 then 1 else 0 end) as "week_4",
sum(case when week_number=5 then 1 else 0 end) as "week_5",
sum(case when week_number=6 then 1 else 0 end) as "week_6",
sum(case when week_number=7 then 1 else 0 end) as "week_7",
sum(case when week_number=8 then 1 else 0 end) as "week_8",
sum(case when week_number=9 then 1 else 0 end) as "week_9",
```



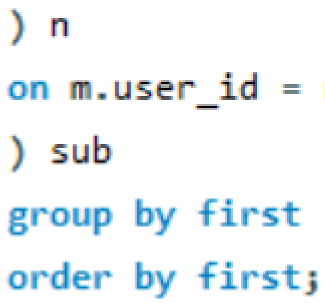
```
sum(case when week_number=10 then 1 else 0 end) as "week_10",
sum(case when week_number=11 then 1 else 0 end) as "week_11",
sum(case when week_number=12 then 1 else 0 end) as "week_12",
sum(case when week_number=13 then 1 else 0 end) as "week_13",
sum(case when week_number=14 then 1 else 0 end) as "week_14",
sum(case when week_number=15 then 1 else 0 end) as "week_15",
sum(case when week_number=16 then 1 else 0 end) as "week_16",
sum(case when week_number=17 then 1 else 0 end) as "week_17",
sum(case when week_number=18 then 1 else 0 end) as "week_18"
```



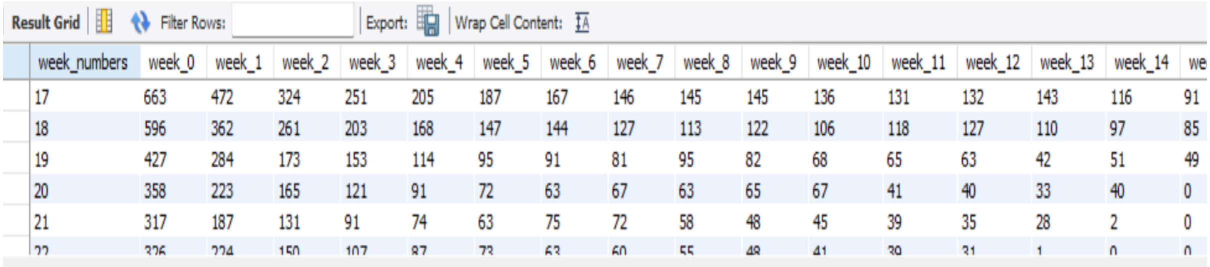
```sql
from (
select
m.user_id,
m.login_week,
n.first,
m.login_week - n.first as week_number
from (
select
user_id,
extract(week from occurred_at) as login_week
from
events
```



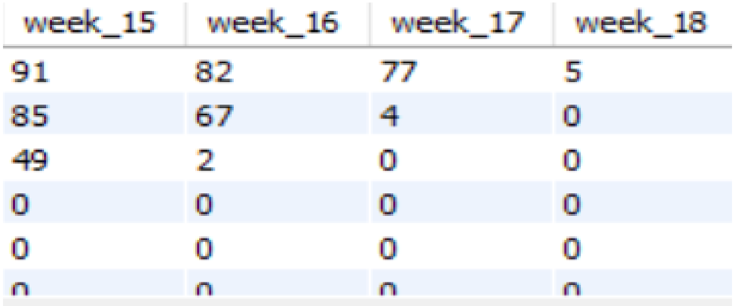
```sql
group by
user_id, login_week
) m
join (
select
user_id,
min(extract(week from occurred_at)) as first
from
events
group by
user_id
```



```sql
) n
on m.user_id = n.user_id
) sub
group by first
order by first;
```



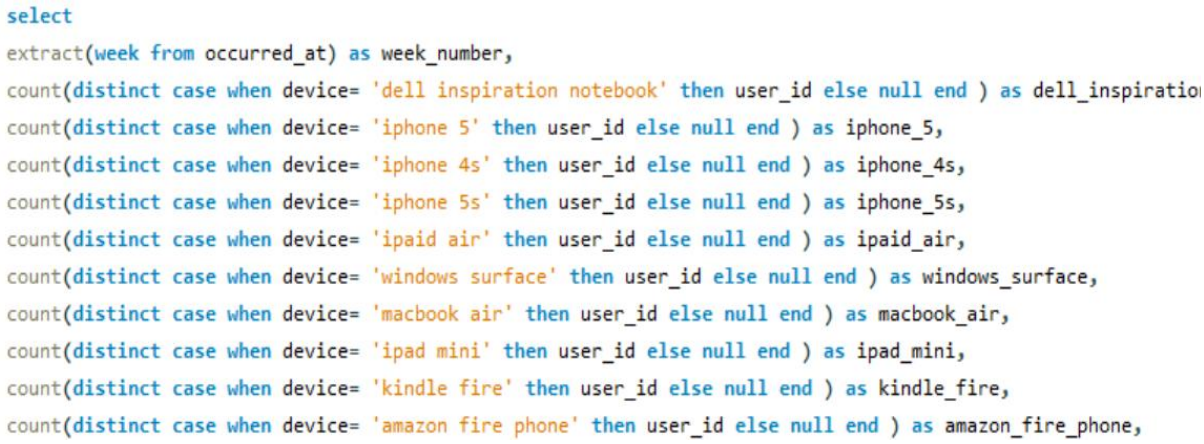
week_numbers	week_0	week_1	week_2	week_3	week_4	week_5	week_6	week_7	week_8	week_9	week_10	week_11	week_12	week_13	week_14	we
17	663	472	324	251	205	187	167	146	145	145	136	131	132	143	116	91
18	596	362	261	203	168	147	144	127	113	122	106	118	127	110	97	85
19	427	284	173	153	114	95	91	81	95	82	68	65	63	42	51	49
20	358	223	165	121	91	72	63	67	63	65	67	41	40	33	40	0
21	317	187	131	91	74	63	75	72	58	48	45	39	35	28	2	0
22	326	224	150	107	87	73	63	60	55	48	41	39	31	1	0	0



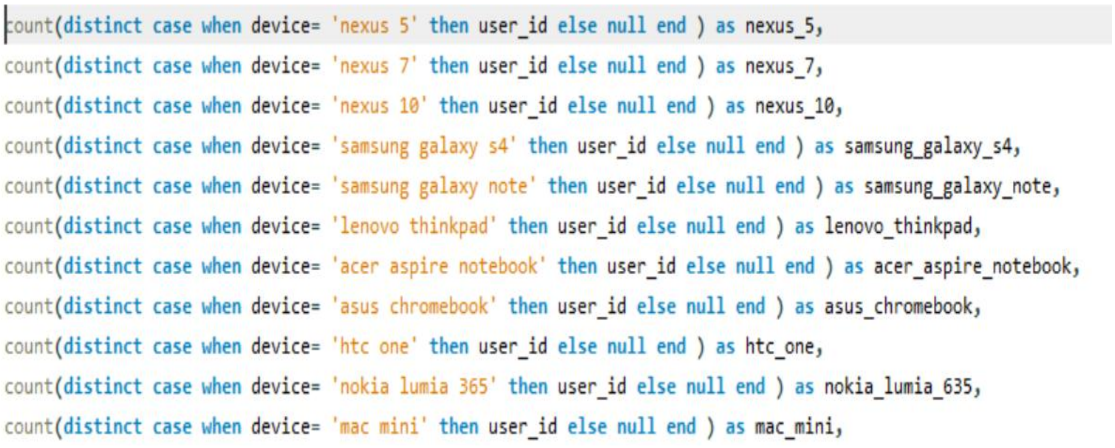
week_15	week_16	week_17	week_18
91	82	77	5
85	67	4	0
49	2	0	0
0	0	0	0
0	0	0	0
0	0	0	0

4. WEEKLY ENGAGEMENT PER DEVICE:

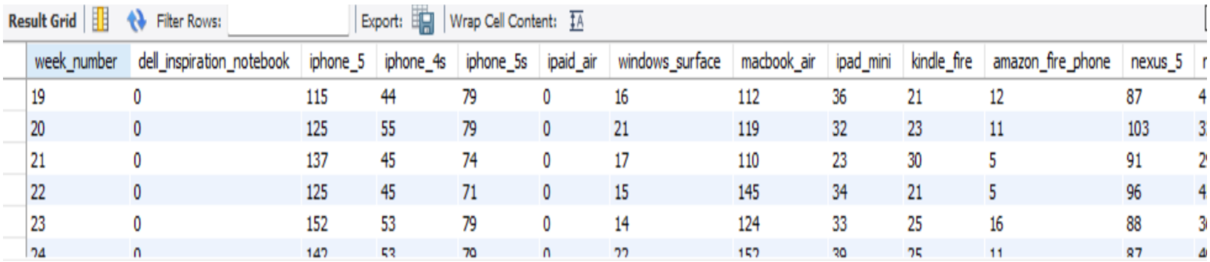
Task: Write an SQL query to calculate the weekly engagement per device.



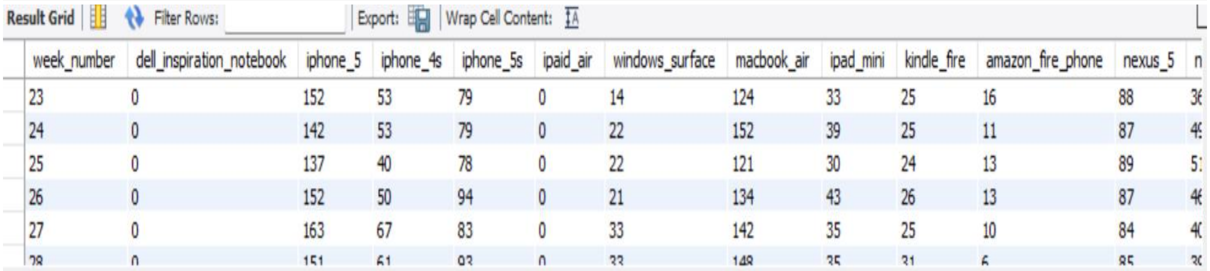
```
select
extract(week from occurred_at) as week_number,
count(distinct case when device= 'dell inspiration notebook' then user_id else null end ) as dell_inspiratio
count(distinct case when device= 'iphone 5' then user_id else null end ) as iphone_5,
count(distinct case when device= 'iphone 4s' then user_id else null end ) as iphone_4s,
count(distinct case when device= 'iphone 5s' then user_id else null end ) as iphone_5s,
count(distinct case when device= 'ipaid air' then user_id else null end ) as ipaid_air,
count(distinct case when device= 'windows surface' then user_id else null end ) as windows_surface,
count(distinct case when device= 'macbook air' then user_id else null end ) as macbook_air,
count(distinct case when device= 'ipad mini' then user_id else null end ) as ipad_mini,
count(distinct case when device= 'kindle fire' then user_id else null end ) as kindle_fire,
count(distinct case when device= 'amazon fire phone' then user_id else null end ) as amazon_fire_phone,
```



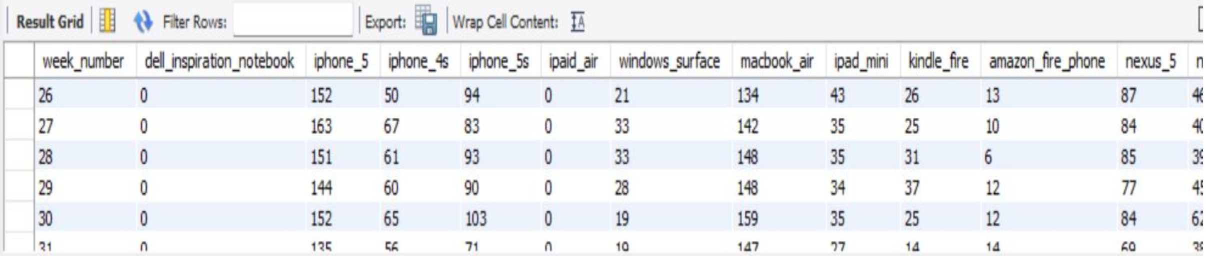
```
count(distinct case when device= 'nexus 5' then user_id else null end ) as nexus_5,
count(distinct case when device= 'nexus 7' then user_id else null end ) as nexus_7,
count(distinct case when device= 'nexus 10' then user_id else null end ) as nexus_10,
count(distinct case when device= 'samsung galaxy s4' then user_id else null end ) as samsung_galaxy_s4,
count(distinct case when device= 'samsung galaxy note' then user_id else null end ) as samsung_galaxy_note,
count(distinct case when device= 'lenovo thinkpad' then user_id else null end ) as lenovo_thinkpad,
count(distinct case when device= 'acer aspire notebook' then user_id else null end ) as acer_aspire_notebook,
count(distinct case when device= 'asus chromebook' then user_id else null end ) as asus_chromebook,
count(distinct case when device= 'htc one' then user_id else null end ) as htc_one,
count(distinct case when device= 'nokia lumia 365' then user_id else null end ) as nokia_lumia_635,
count(distinct case when device= 'mac mini' then user_id else null end ) as mac_mini,
count(distinct case when device= 'hd pavilion desktop' then user_id else null end ) as hd_pavilion_desktop,
count(distinct case when device= 'dell inspiron desktop' then user_id else null end ) as dell_inspiron_desktop
from
events
where
event_type='engagement'
group by
week_number
order by
week_number;
```



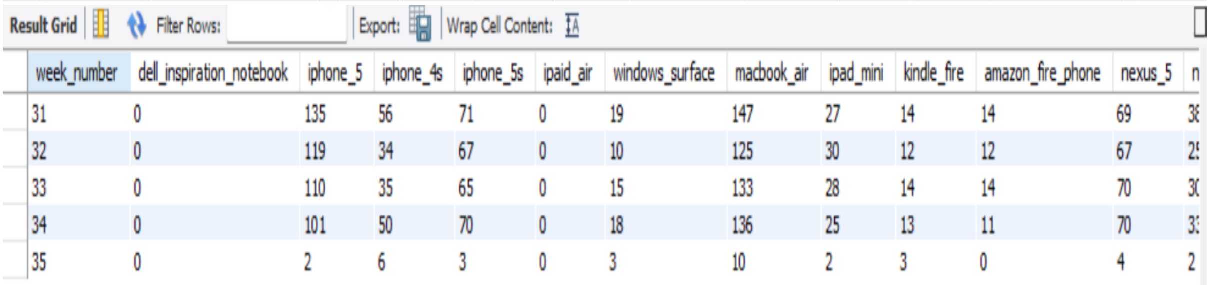
week_number	dell_inspiration_notebook	iphone_5	iphone_4s	iphone_5s	ipaid_air	windows_surface	macbook_air	ipad_mini	kindle_fire	amazon_fire_phone	nexus_5	n
19	0	115	44	79	0	16	112	36	21	12	87	4
20	0	125	55	79	0	21	119	32	23	11	103	3
21	0	137	45	74	0	17	110	23	30	5	91	2
22	0	125	45	71	0	15	145	34	21	5	96	4
23	0	152	53	79	0	14	124	33	25	16	88	3
24	0	142	53	79	0	22	152	39	25	11	87	4



week_number	dell_inspiration_notebook	iphone_5	iphone_4s	iphone_5s	ipaid_air	windows_surface	macbook_air	ipad_mini	kindle_fire	amazon_fire_phone	nexus_5	n
23	0	152	53	79	0	14	124	33	25	16	88	36
24	0	142	53	79	0	22	152	39	25	11	87	49
25	0	137	40	78	0	22	121	30	24	13	89	51
26	0	152	50	94	0	21	134	43	26	13	87	46
27	0	163	67	83	0	33	142	35	25	10	84	40
28	0	151	61	93	0	33	148	35	31	6	85	39



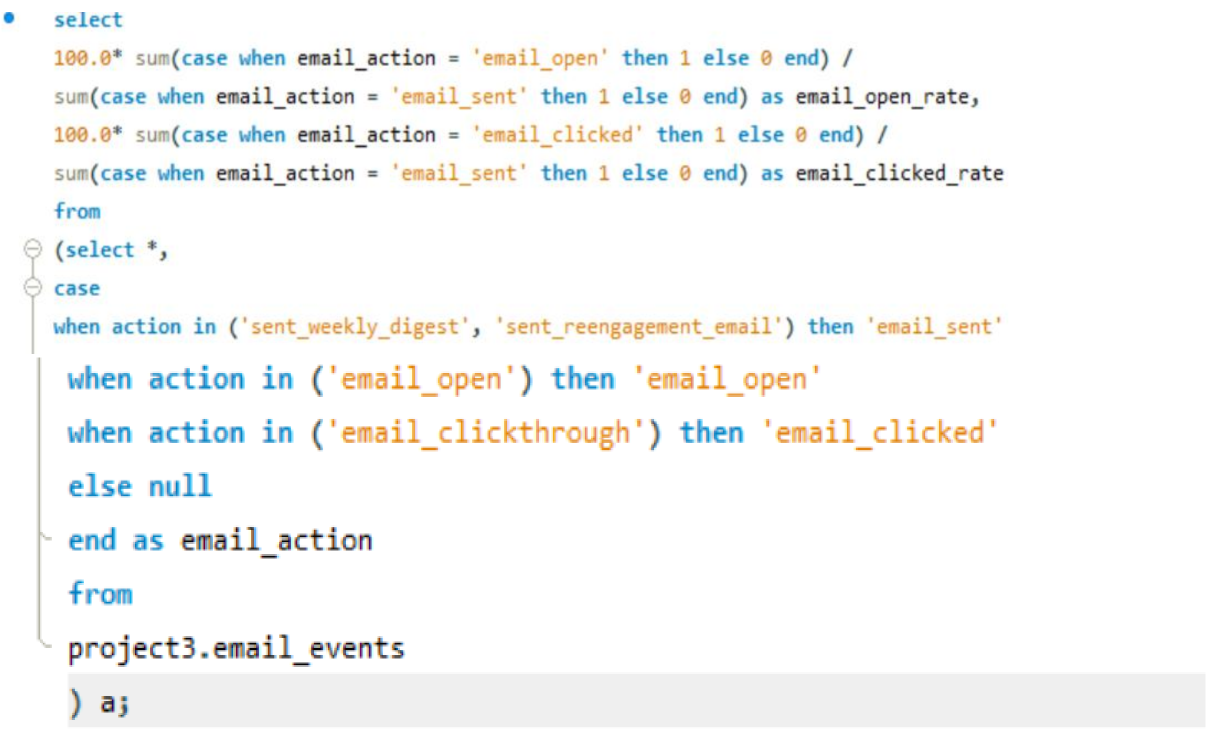
week_number	dell_inspiration_notebook	iphone_5	iphone_4s	iphone_5s	ipaid_air	windows_surface	macbook_air	ipad_mini	kindle_fire	amazon_fire_phone	nexus_5	n
26	0	152	50	94	0	21	134	43	26	13	87	46
27	0	163	67	83	0	33	142	35	25	10	84	40
28	0	151	61	93	0	33	148	35	31	6	85	39
29	0	144	60	90	0	28	148	34	37	12	77	45
30	0	152	65	103	0	19	159	35	25	12	84	62
31	0	135	56	71	0	19	147	27	14	14	69	38



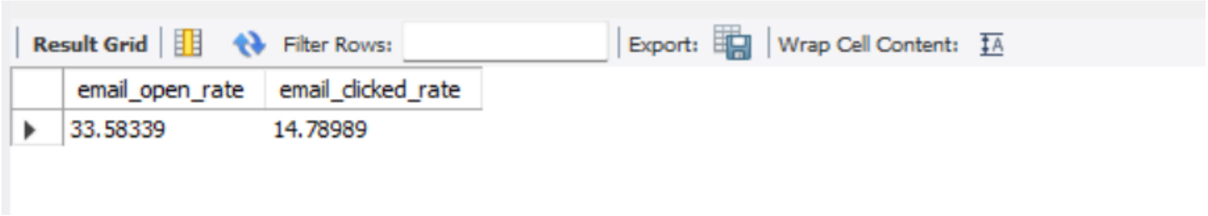
week_number	dell_inspiration_notebook	iphone_5	iphone_4s	iphone_5s	ipaid_air	windows_surface	macbook_air	ipad_mini	kindle_fire	amazon_fire_phone	nexus_5	n
31	0	135	56	71	0	19	147	27	14	14	69	38
32	0	119	34	67	0	10	125	30	12	12	67	25
33	0	110	35	65	0	15	133	28	14	14	70	30
34	0	101	50	70	0	18	136	25	13	11	70	33
35	0	2	6	3	0	3	10	2	3	0	4	2

5. EMAIL ENGAGEMENT ANALYSIS:

Task: Write an SQL query to calculate the email engagement metrics.



```
select
100.0* sum(case when email_action = 'email_open' then 1 else 0 end) /
sum(case when email_action = 'email_sent' then 1 else 0 end) as email_open_rate,
100.0* sum(case when email_action = 'email_clicked' then 1 else 0 end) /
sum(case when email_action = 'email_sent' then 1 else 0 end) as email_clicked_rate
from
(select *,
case
when action in ('sent_weekly_digest', 'sent_reengagement_email') then 'email_sent'
 when action in ('email_open') then 'email_open'
 when action in ('email_clickthrough') then 'email_clicked'
 else null
 end as email_action
 from
 project3.email_events
 ) a;
```



email_open_rate	email_clicked_rate
33.58339	14.78989

THANK YOU