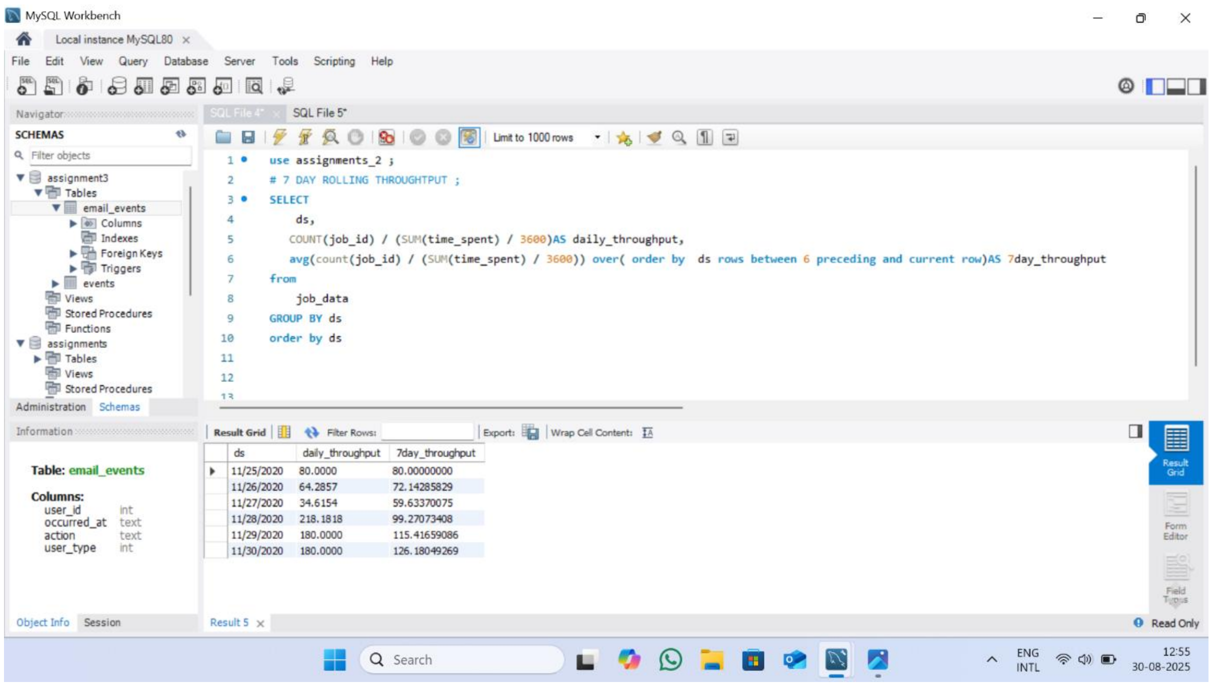
Operation Analytics and Investigating Metric Spike

Case Study 1: Job Data Analysis

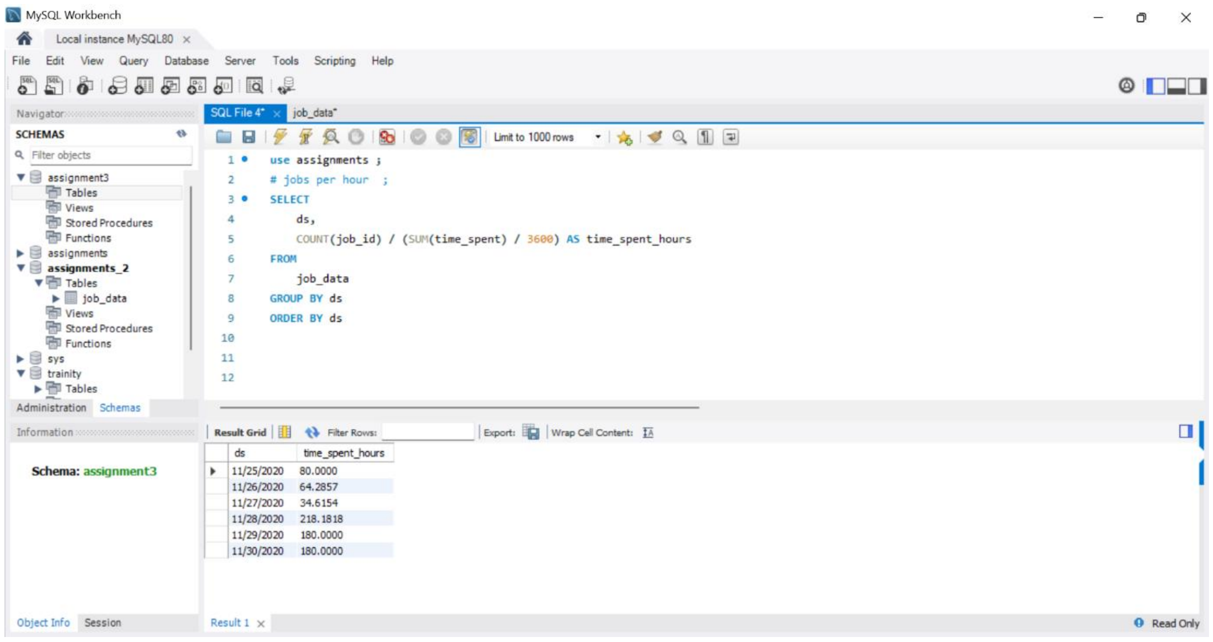
1. Write an SQL query to calculate the number of jobs reviewed per hour for each day in November 2020.



```
use assignments_2 ;
# 7 DAY ROLLING THROUGHTPUT ;
SELECT
    ds,
   COUNT(job_id) / (SUM(time_spent) / 3600)AS daily_throughput,
   avg(count(job_id) / (SUM(time_spent) / 3600)) over( order by  ds rows between 6 preceding and current row)AS 7day_throughput
from
    job_data
GROUP BY ds
order by ds
```

ds	daily_throughput	7day_throughput
11/25/2020	80.0000	80.00000000
11/26/2020	64.2857	72.14285829
11/27/2020	34.6154	59.63370075
11/28/2020	218.1818	99.27073408
11/29/2020	180.0000	115.41659086
11/30/2020	180.0000	126.18049269

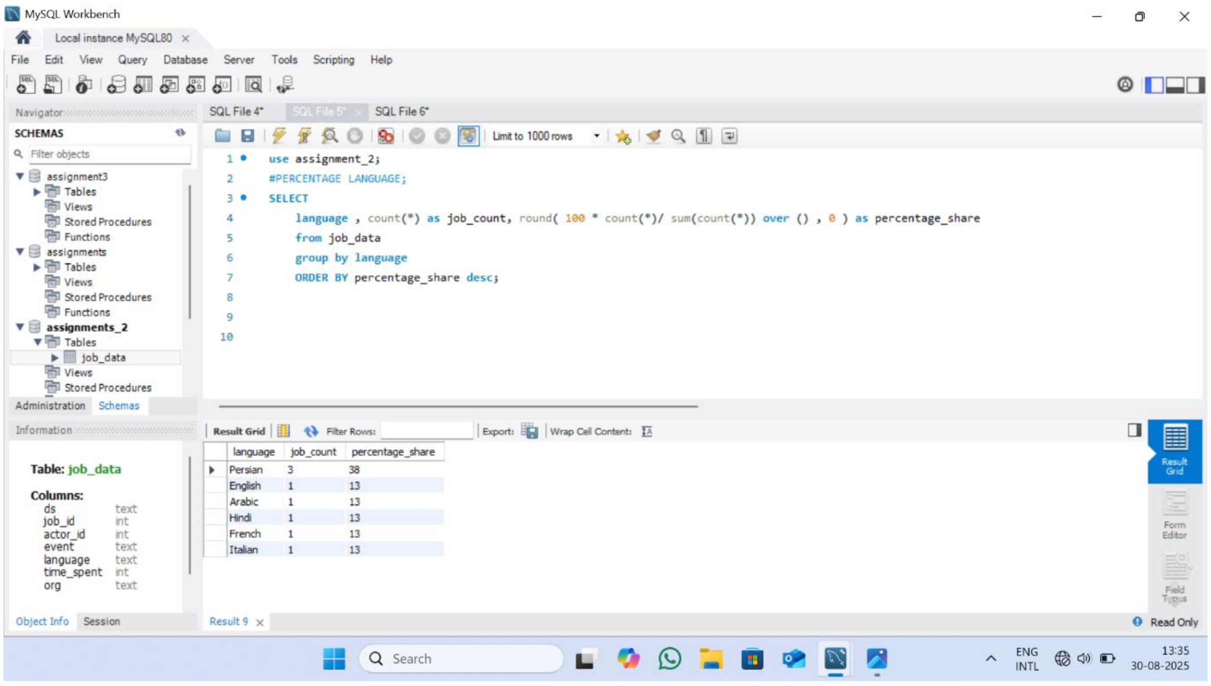
2. Write an SQL query to calculate the 7-day rolling average of throughput



```
use assignments ;
# jobs per hour  ;
SELECT
    ds,
    COUNT(job_id) / (SUM(time_spent) / 3600) AS time_spent_hours
FROM
    job_data
GROUP BY ds
ORDER BY ds
```

ds	time_spent_hours
11/25/2020	80.0000
11/26/2020	64.2857
11/27/2020	34.6154
11/28/2020	218.1818
11/29/2020	180.0000
11/30/2020	180.0000

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



```sql
use assignment_2;
#PERCENTAGE LANGUAGE;
SELECT
    language , count(*) as job_count, round( 100 * count(*)/ sum(count(*)) over () , 0 ) as percentage_share
    from job_data
    group by language
    ORDER BY percentage_share desc;
```

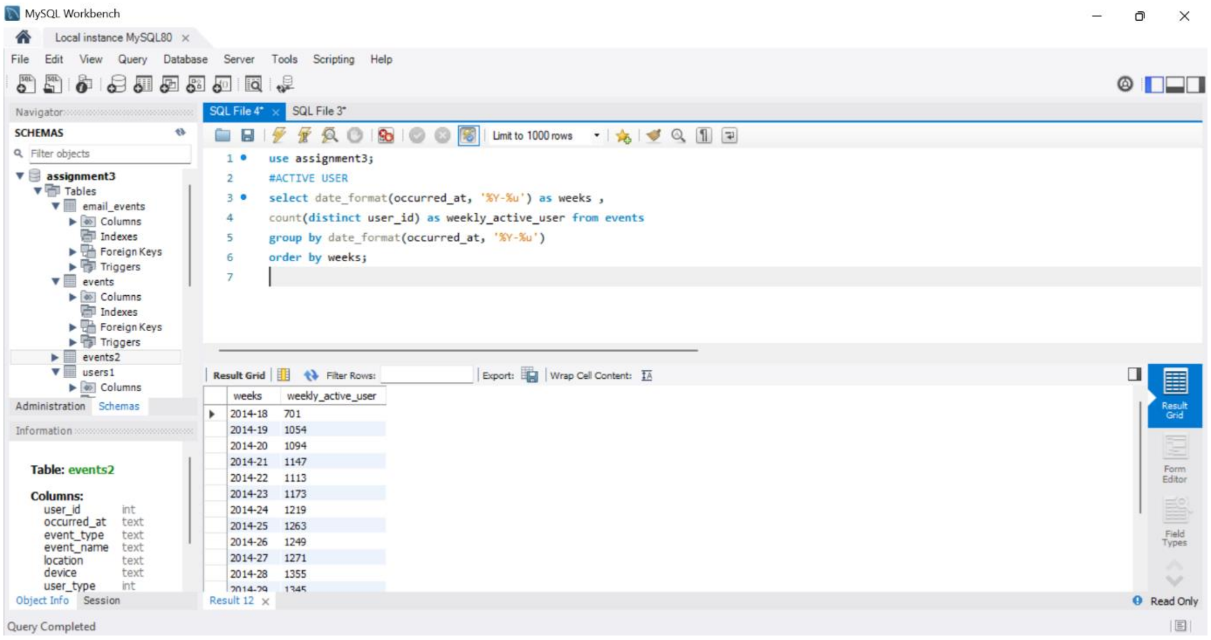
Table: job_data

Columns:

ds	text
job_id	int
actor_id	int
event	text
language	text
time_spent	int
org	text

language	job_count	percentage_share
Persian	3	38
English	1	13
Arabic	1	13
Hindi	1	13
French	1	13
Italian	1	13

4. Write an SQL query to display duplicate rows fr om the job_data table .



```sql
use assignment3;
#ACTIVE USER
select date_format(occurred_at, '%Y-%u') as weeks ,
count(distinct user_id) as weekly_active_user from events
group by date_format(occurred_at, '%Y-%u')
order by weeks;
```

Table: events2

Columns:

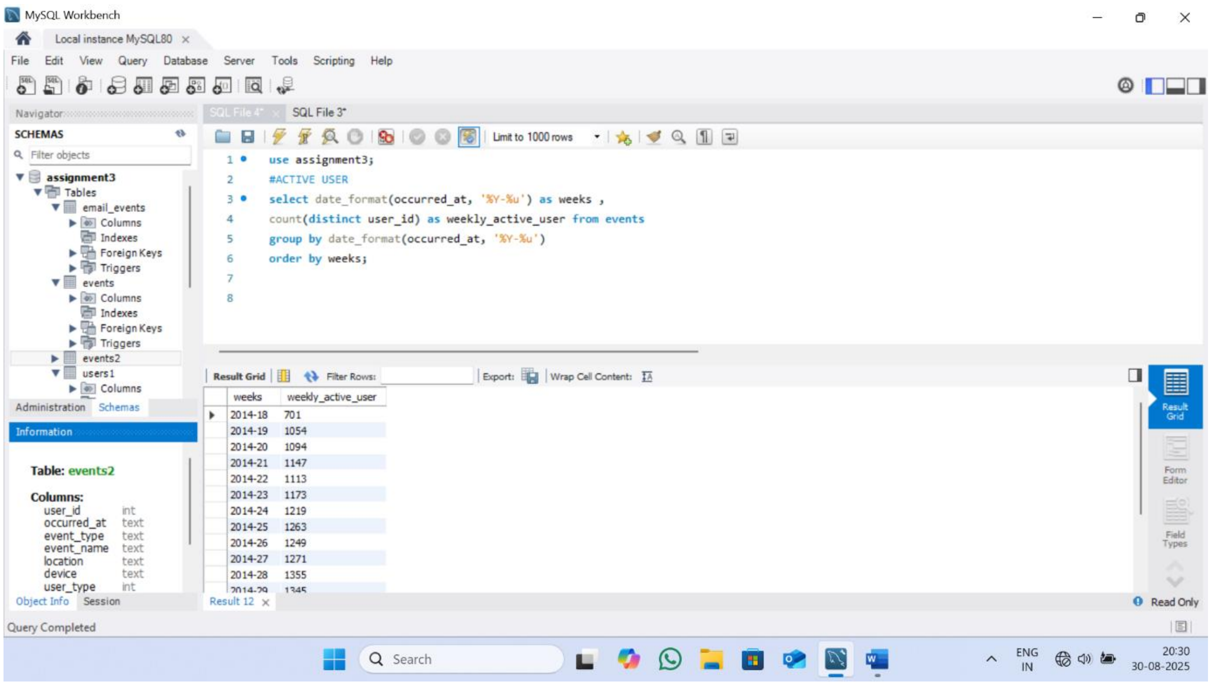
user_id	int
occurred_at	text
event_type	text
event_name	text
location	text
device	text
user_type	int

weeks	weekly_active_user
2014-18	701
2014-19	1054
2014-20	1094
2014-21	1147
2014-22	1113
2014-23	1173
2014-24	1219
2014-25	1263
2014-26	1249
2014-27	1271
2014-28	1355
2014-29	1345

Query Completed

Case Study 2: Investigating Metric Spike

1. Write an SQL query to calculate the weekly user engagement.

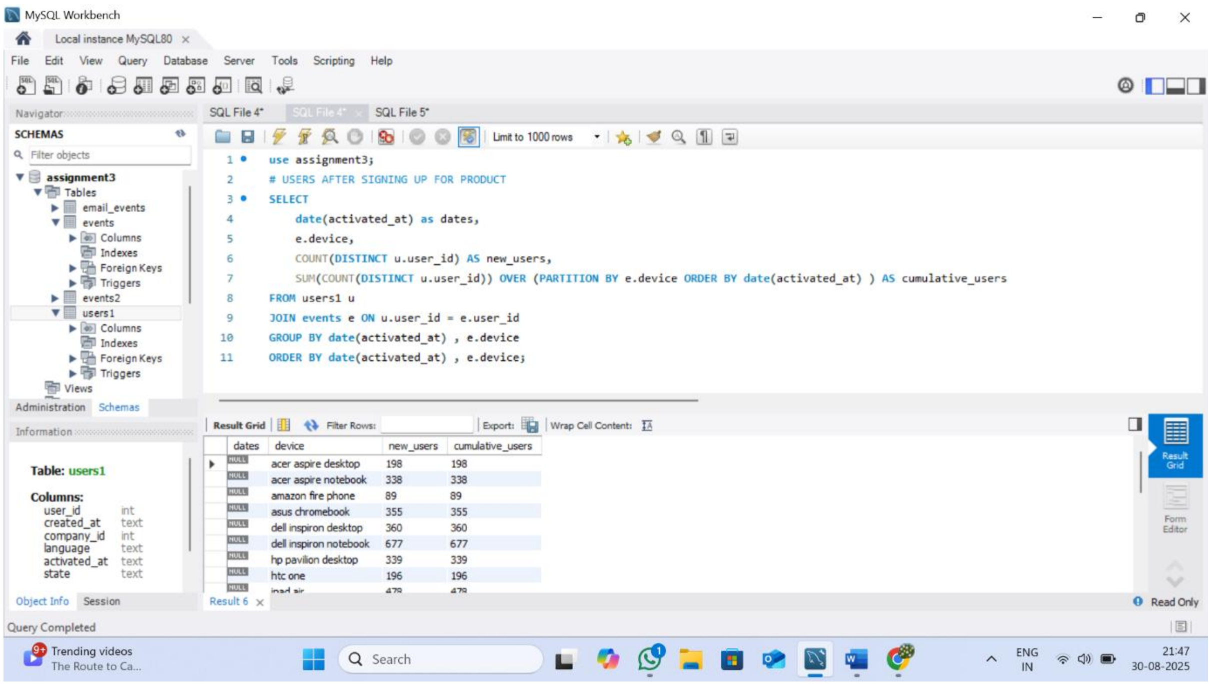


```sql
use assignment3;
#ACTIVE USER
select date_format(occurred_at, '%Y-%u') as weeks ,
count(distinct user_id) as weekly_active_user from events
group by date_format(occurred_at, '%Y-%u')
order by weeks;
```

weeks	weekly_active_user
2014-18	701
2014-19	1054
2014-20	1094
2014-21	1147
2014-22	1113
2014-23	1173
2014-24	1219
2014-25	1263
2014-26	1249
2014-27	1271
2014-28	1355
2014-29	1345

Query Completed

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

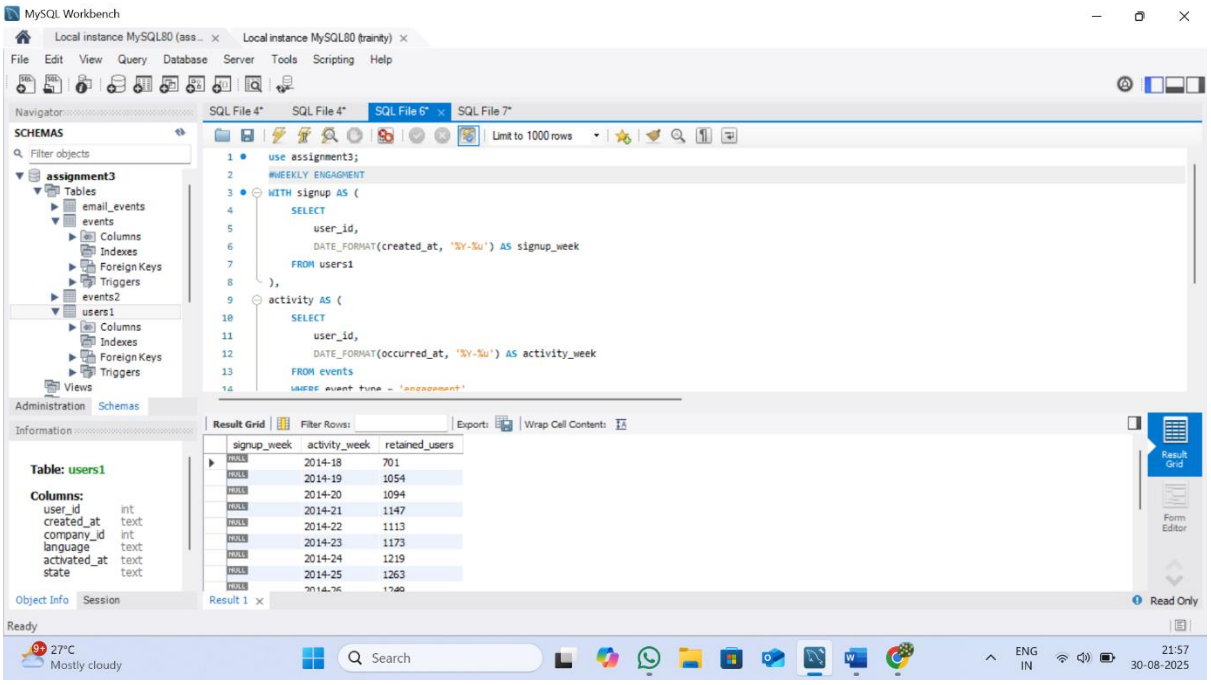


```sql
use assignment3;
# USERS AFTER SIGNING UP FOR PRODUCT
SELECT
    date(activated_at) as dates,
    e.device,
    COUNT(DISTINCT u.user_id) AS new_users,
    SUM(COUNT(DISTINCT u.user_id)) OVER (PARTITION BY e.device ORDER BY date(activated_at) ) AS cumulative_users
FROM users1 u
JOIN events e ON u.user_id = e.user_id
GROUP BY date(activated_at) , e.device
ORDER BY date(activated_at) , e.device;
```

dates	device	new_users	cumulative_users
NULL	acer aspire desktop	198	198
NULL	acer aspire notebook	338	338
NULL	amazon fire phone	89	89
NULL	asus chromebook	355	355
NULL	dell inspiron desktop	360	360
NULL	dell inspiron notebook	677	677
NULL	hp pavilion desktop	339	339
NULL	htc one	196	196
NULL	ipad air	478	478

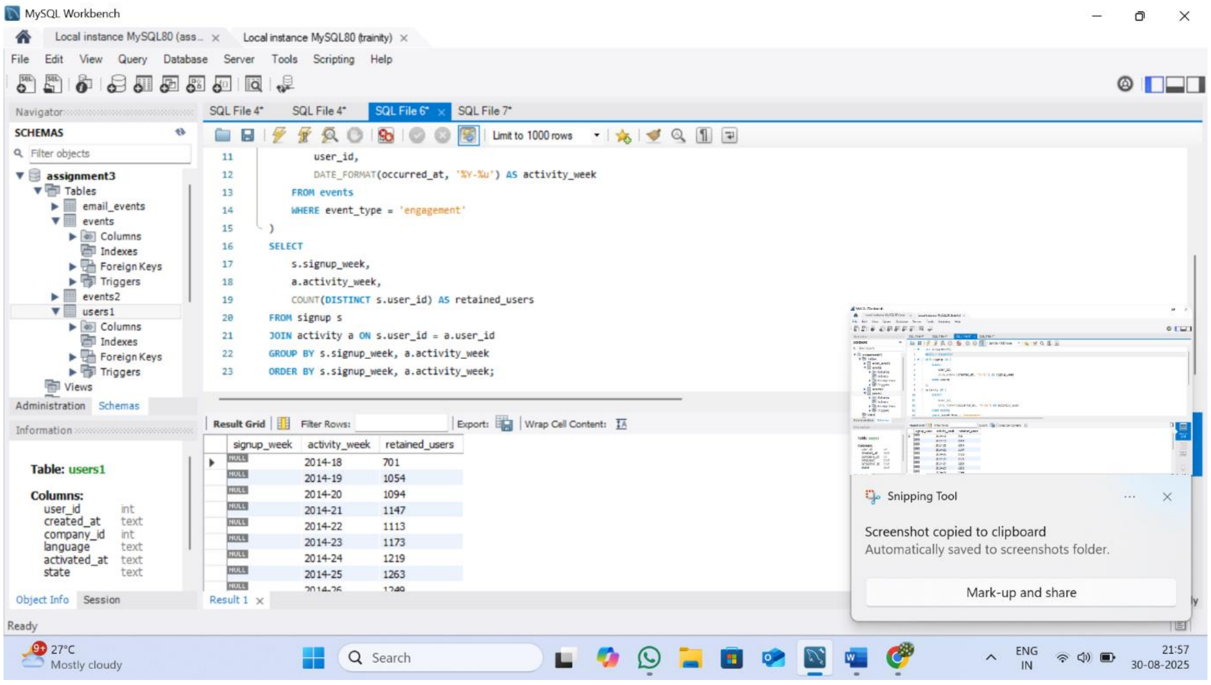
Query Completed

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



```sql
use assignment3;
#WEEKLY ENGAGMENT
WITH signup AS (
    SELECT
        user_id,
        DATE_FORMAT(created_at, '%Y-%u') AS signup_week
    FROM users1
),
activity AS (
    SELECT
        user_id,
        DATE_FORMAT(occurred_at, '%Y-%u') AS activity_week
    FROM events
```

signup_week	activity_week	retained_users
NULL	2014-18	701
NULL	2014-19	1054
NULL	2014-20	1094
NULL	2014-21	1147
NULL	2014-22	1113
NULL	2014-23	1173
NULL	2014-24	1219
NULL	2014-25	1263



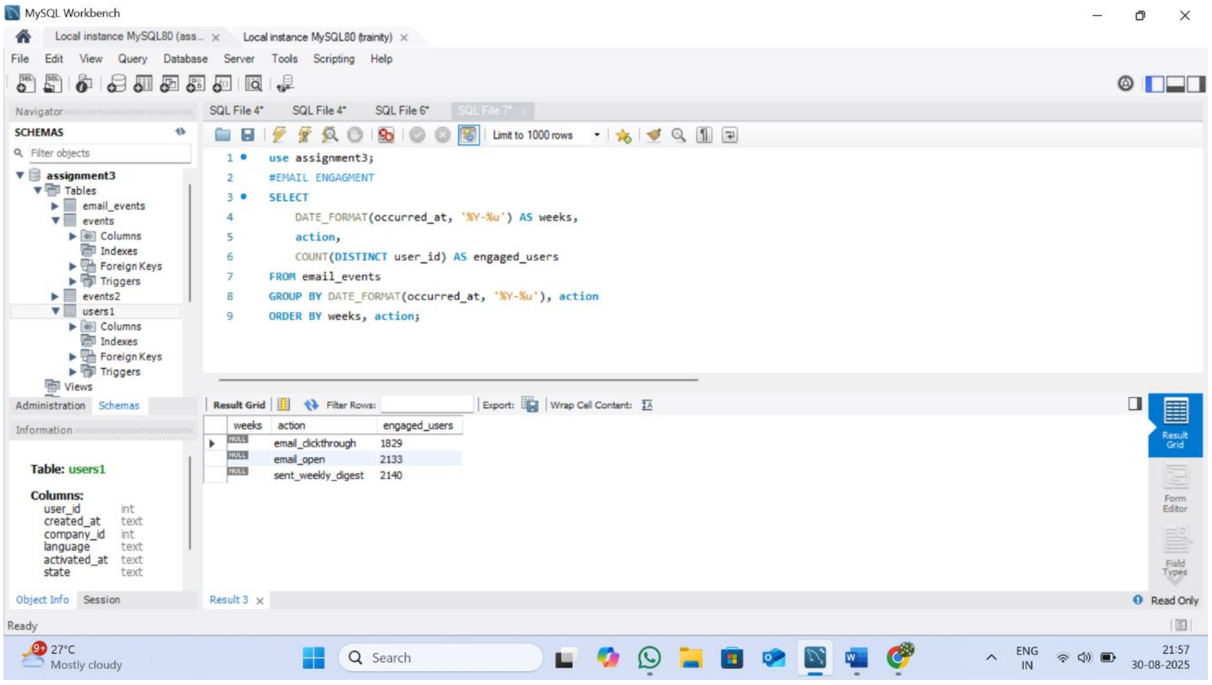
```sql
        user_id,
        DATE_FORMAT(occurred_at, '%Y-%u') AS activity_week
    FROM events
    WHERE event_type = 'engagement'
)
SELECT
    s.signup_week,
    a.activity_week,
    COUNT(DISTINCT s.user_id) AS retained_users
FROM signup s
JOIN activity a ON s.user_id = a.user_id
GROUP BY s.signup_week, a.activity_week
ORDER BY s.signup_week, a.activity_week;
```

signup_week	activity_week	retained_users
NULL	2014-18	701
NULL	2014-19	1054
NULL	2014-20	1094
NULL	2014-21	1147
NULL	2014-22	1113
NULL	2014-23	1173
NULL	2014-24	1219
NULL	2014-25	1263

Snipping Tool

Screenshot copied to clipboard
Automatically saved to screenshots folder.

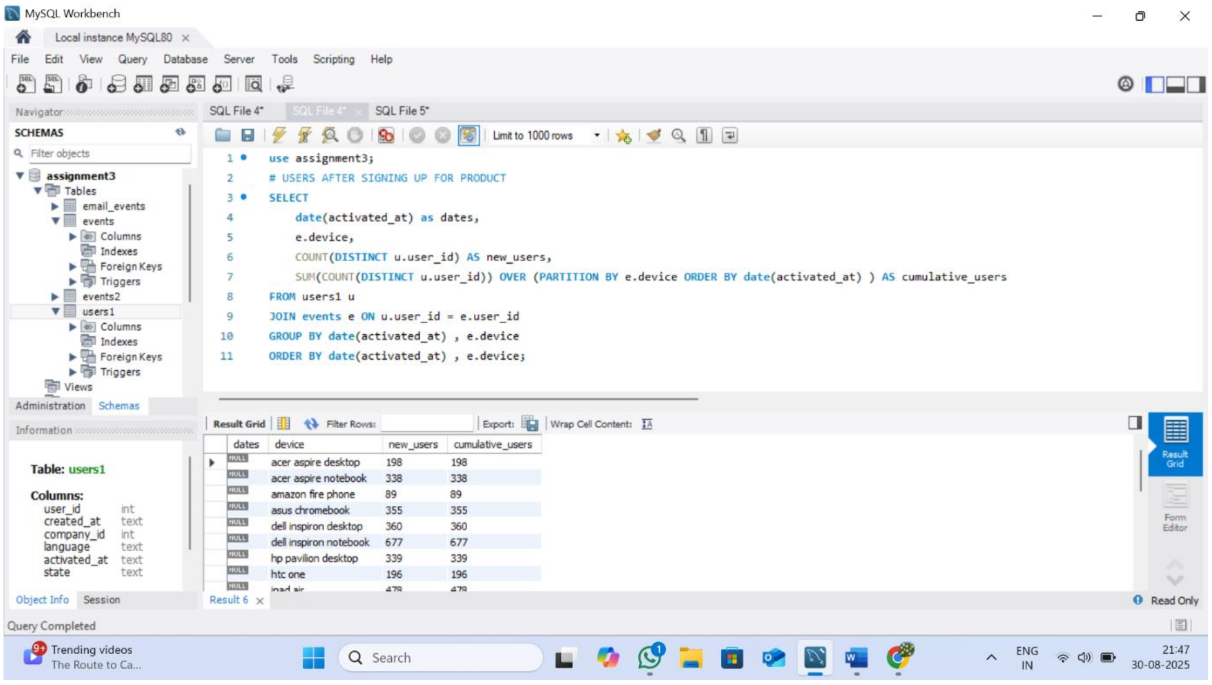
4. Write an SQL query to calculate the email engagement metrics.



```sql
use assignment3;
#EMAIL ENGAGMENT
SELECT
    DATE_FORMAT(occurred_at, '%Y-%u') AS weeks,
    action,
    COUNT(DISTINCT user_id) AS engaged_users
FROM email_events
GROUP BY DATE_FORMAT(occurred_at, '%Y-%u'), action
ORDER BY weeks, action;
```

weeks	action	engaged_users
NULL	email_clickthrough	1829
NULL	email_open	2133
NULL	sent_weekly_digest	2140

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



```sql
use assignment3;
# USERS AFTER SIGNING UP FOR PRODUCT
SELECT
    date(activated_at) as dates,
    e.device,
    COUNT(DISTINCT u.user_id) AS new_users,
    SUM(COUNT(DISTINCT u.user_id)) OVER (PARTITION BY e.device ORDER BY date(activated_at) ) AS cumulative_users
FROM users1 u
JOIN events e ON u.user_id = e.user_id
GROUP BY date(activated_at) , e.device
ORDER BY date(activated_at) , e.device;
```

dates	device	new_users	cumulative_users
NULL	acer aspire desktop	198	198
NULL	acer aspire notebook	338	338
NULL	amazon fire phone	89	89
NULL	asus chromebook	355	355
NULL	dell inspiron desktop	360	360
NULL	dell inspiron notebook	677	677
NULL	hp pavilion desktop	339	339
NULL	htc one	196	196
NULL	ipad air	478	478