OPERATIONAL ANALYTICS AND INVESTIGATING METRIC SPIKE

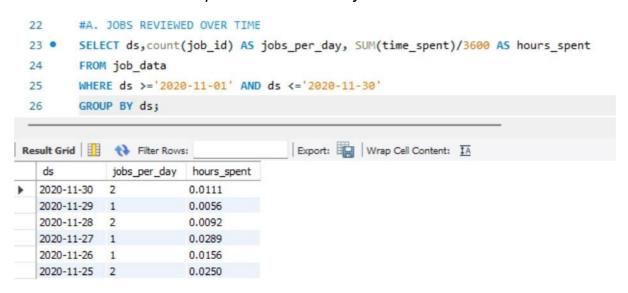
- 1) PROJECT DESCRIPTION: This project involves deriving insights and using those insights to work towards more efficient operations and understand sudden changes in key metrics.
- 2) APPROACH: The project was executed using Advanced SQL and MySQL workbench. Insights were derived from the data using queries.
- 3) **TECH-STACK USED:** My SQL Workbench 8.0 CE was used in analyzing this data. It's portable, lightweight and easy to use. It was used to execute queries, analyze data and derive insights.

4) INSIGHTS:

CASE STUDY 1: JOB DATA ANALYSIS

(A) JOBS REVIEWED OVER TIME

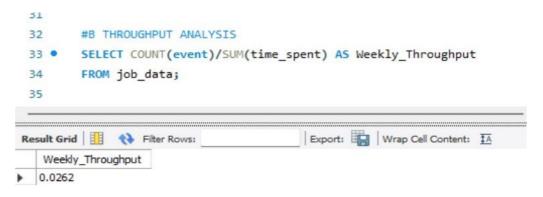
- Calculate the number of jobs reviewed per hour for each day in November 2020.
- Your task: Write a SQL Query to calculate the number of jobs reviewed per hour for each day in November 2020.



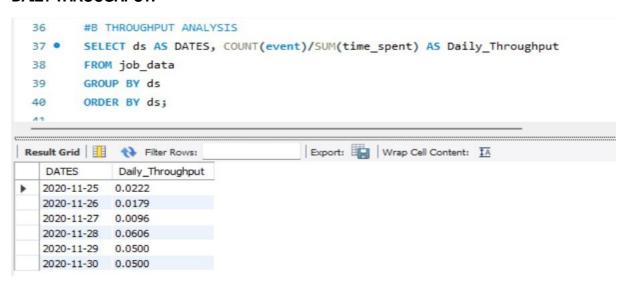
(B) THROUGHPUT ANALYSIS

- Objective: Calculate the 7-day rolling average of Throughput (number of events per second).
- Task: Write a 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:



DAILY THROUGHPUT:



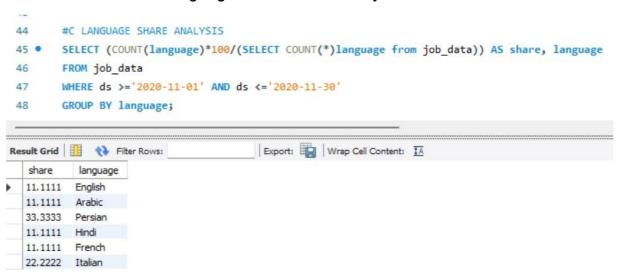
The 7-day rolling average throughput is 0.262. I **prefer the 7 day rolling average** metric over the daily one because it is less overwhelming to sort through and still gives us a useful insights that are just right enough to not be too macro. It is:

- Time-efficient, as there is less data to sort through than compared to the daily metric.
- Effective, is still detailed captures the small details and sorts off the unnecery details irrelevant to understand the bigger picture of things.

Which makes it the most efficient method for analyzing to gain valuable insights for improvement.

(C) LANGUAGE SHARE ANALYSIS

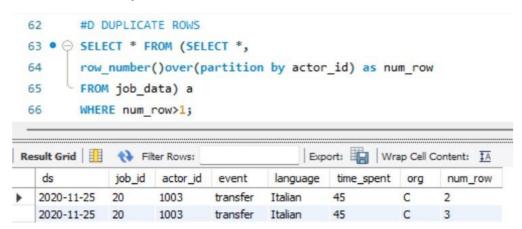
- Objective: Calculate the percentage share of each language in the last 30 days.
- Task: Write a SQL query to calculate the percentage share of each language over the last 30 days.



Persian has the **highest share** amongst all the other languages at 33.3%. The second highest share of language after Persian is **Italian at 22.2%**.

(D) DUPLICATE ROWS DETECTION

- Objective: Identify duplicate rows in the data.
- Task: Write a SQL Query to display duplicate rows from the job_data table.

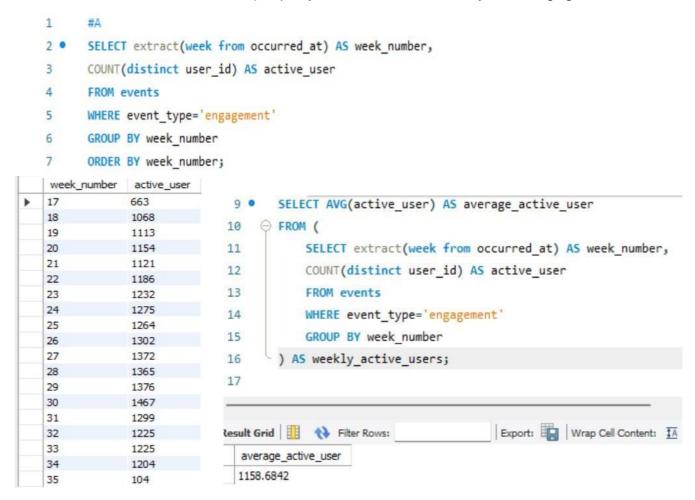


There are **2 duplicate rows** sharing the actor_id, 1003.

CASE STUDY 2: INVESTIGATING METRIC SPIKE

(A) WEEKLY USER ENGAGEMENT

- Objective: Measure the activeness of users on a weekly basis.
- Task: Write a SQL query to calculate the weekly user engagement.



There is a **steady increase** in the weekly user engagement up until **week 30** during which the highest number of users engaged with our platform at **1467 users**. Indicating that **unique users** engaged with our platform.

There is a drop in users after that however it does not go below our average of 1158 users.

There is a **significant dip** in users during **week 35** which saw the lowest user engagement at 104 users.

(B) USER GROWTH ANALYSIS

- o Objective: Analyse the growth of users over time for a product.
- Task: Write a SQL Query to calculate the user growth for a product over time.

	year	week_num	num_users	cum_users	year	week_num	num_users	cum_users	year	week_num	num_users	cum_users				
2	2013	0	23	23	2013	23	50	973	2013	45	91	2607				
2	2013	1	30	53	2013	24	45	1018	2013	46	88	2695	2014	15	164	5424
2	2013	2	48	101	2013	25	57	1075	2013	47	102	2797	2014	16	179	5603
2	2013	3	36	137	2013	26	56	1131	2013	48	97	2894	2014	17	170	5773
2	2013	4	30	167	2013	27	52	1183	2013	49	116	3010	2014	18	163	5936
2	2013	5	48	215	2013	28	72	1255	2013	50	124	3134	2014	19	185	6121
2	2013	6	38	253	2013	29	67	1322	2013	51	102	3236	2014	20	176	6297
2	2013	7	42	295	2013	30	67	1389	2013	52	47	3283	2014	21	183	6480
2	2013	8	34	329	2013	31	67	1456	2014	0	83	3366	2014	22	196	6676
2	2013	9	43	372	2013	32	71	1527	2014	1	126	3492	2014	23	196	6872
2	2013	10	32	404	2013	33	73	1600	2014	2	109	3601	2014	24	229	7101
2	2013	11	31	435	2013	34	78	1678	2014	3	113	3714	2014	25	207	7308
2	2013	12	33	468	2013	35	63	1741	2014	4	130	3844	2014	26	201	7509
2	2013	13	39	507	2013	36	72	1813	2014	5	133	3977	2014	27	222	7731
2	2013	14	35	542	2013	37	85	1898	2014	6	135	4112	2014	28	215	7946
2	2013	15	43	585	2013	38	90	1988	2014	7	125	4237	2014	29	221	8167
2	2013	16	46	631	2013	39	84	2072	2014	8	129	4366	2014	30	238	8405
2	2013	17	49	680	2013	40	87	2159	2014	9	133	4499	2014	31	193	8598
2	2013	18	44	724	2013	41	73	2232	2014	10	154	4653	2014	32	245	8843
2	2013	19	57	781	2013	42	99	2331	2014	11	130	4783	2014	33	261	9104
2	2013	20	39	820	2013	43	89	2420	2014	12	148	4931	2014	34	259	9363
2	2013	21	49	869	2013	44	96	2516	2014	13	167	5098	2021	-1	Lus	3000
2	2013	22	54	923	2013	45	91	2607	2014	14	162	5260				

The number of users is experiencing a steady growth despite a few fluctuations here and there.

The cumulative users have reached 9363 in over 1.5 years.

(C) WEEKLY RETENTION ANALYSIS

- Objective: Analyse the retention of users on a weekly basis after signing up.
- Task: Write a SQL Query to calculate the weekly retention of users based on their sign-up cohort.

```
WITH ctel AS (
                                                                                    signup_week
                                                                                                total_engaged_users
                                                                                                                  retained_use
 ELECT DISTINCT user id,
                                                                                    17
                                                                                                                  147
                                                                                                278
  XTRACT(week from occurred_at) AS signup_week
                                                                                               615
                                                                                                                  338
 ROM events
                                                                                                677
                                                                                                                  350
                                                                                    20
                                                                                               682
                                                                                                                  378
 IHERE event_type='signup_flow'
                                                                                                644
                                                                                                                  340
 ND event name='complete signup'),
                                                                                    22
                                                                                                694
                                                                                                                  356
c :e2 A5 (
                                                                                    23
                                                                                                707
                                                                                                                  365
  ELECT DISTINCT user_id,
                                                                                    24
                                                                                                700
                                                                                                                  320
  XTRACT(week from occurred at) as engagement week
                                                                                    25
                                                                                               671
                                                                                                                  299
                                                                                    26
 HERE event_type='engagement' )
                                                                                    27
                                                                                                697
                                                                                                                  314
                                                                                    28
                                                                                                596
                                                                                                                  220
                                                                                    29
                                                                                                588
                                                                                                                  207
SILECT signup_week, COUNT(user_id) total_engaged_users,
                                                                                    30
                                                                                               614
                                                                                                                  205
 UM(case when retention_week > 1 then 1 else 0 end) as retained_users
                                                                                    31
                                                                                                451
                                                                                                                  122
F tOM(
                                                                                    32
                                                                                                508
                                                                                                                  89
    SELECT a.user_id, a.signup_week,
                                                                                    33
                                                                                                456
    b.engagement_week, b.engagement_week - a.signup_week AS retention_week
                                                                                    34
                                                                                                302
                                                                                                                  0
    FROM ctel a
                                                                                   35
    LEFT JOIN cte2 b
    ON a.user_id = b.user_id
    ORDER BY a.user_id ) sub
G OUP BY signup_week
O DER BY signup week;
```

Weekly retention after signing up was at a steady rise up until week 20, which was its highest at 378 retained users. After that there were slight declines as well increase, which gradually became a significant decline and came down to 0 retained users for week 34 and 35.

(D) WEEKLY ENGAGEMENT PER DEVICE

- Objective: Measure the activeness of users on a weekly basis per device.
- Task: Write s SQL query to calculate the weekly engagement per device.

```
COUNT(event_type) AS event_num,

EXTRACT(week from occurred_at) AS weeknum,

device

from events

where event_type='engagement'

GROUP BY weeknum,device

ORDER BY weeknum, event_num;
```

Link to the table obtained:

https://drive.google.com/file/d/1QJLR8POU45tblcyQc9aLuJy0eHqspifD/view?usp =sharing

The **highest** engagement was on **week 31** with **3608** engagements on **MacBook Pro Max**.

The **lowest** engagement was on **week 35** with **4** engagements on **Dell Inspiron Desktop**.

(E) EMAIL ENGAGEMENT ANALYSIS

- o Objective: Analyse how users are engaging with the email service.
- o Task: Write a SQL query to calculate the email engagement metrics.

```
#E
  SELECT
  100*SUM(CASE WHEN email_at='email_open' then 1 else 0 end) /
  SUM(CASE WHEN email_at='email_sent' then 1 else 0 end) AS email_openrate,
  100*SUM(CASE WHEN email at='email clicked' then 1 else 0 end) /
  SUM(CASE WHEN email_at='email_sent' then 1 else 0 end) AS email_clickrate

⊖ FROM(
  SELECT *,

⊖ CASE

  WHEN action IN ('sent_weekly_digest', 'sent_reengagement_email') THEN 'email_sent'
  WHEN action = 'email open' THEN 'email open'
  WHEN action = 'email clickthrough' THEN 'email clicked'
  END AS email at
  FROM email_events) sub;
       email_openrate email_clickrate
      33.5834
                    14.7899
```

The email open rate is **33.58** and the click rate is **14.78**. Which means that a little less than half the opened emails had a clickthrough to their link.

(5) RESULT: I have improved my SQL skills and learned to write more Advanced queries using functions like subqueries, case statements, over() function etc. It has taught me how to fetch data needed for analysation in the most efficient way possible. Alongside this, I got valuable hands-on experience of the kind of work that is done by a Data Analyst on a daily basis.