Operation Analytics and Investigating metric spike:

Operation Analytics and Investigating metric spike case study helps the company and analyst team to understand the company growth and day to day operations.
This analysis will give insights on user growth , user engagement with the product /services.

Calculate the number of jobs reviewed per hour per day for November 2020?

```
SELECT ds AS dates,
    ROUND((COUNT(job_id)/SUM(time_spent))*3600) AS No_of_jobs_reviewed
FROM primary_data
WHERE ds BETWEEN '2020-11-01' AND '2020-11-30'
GROUP BY ds;
```

Below you will find the number of jobs reviewed per hour per day. On Nov 28 maximum jobs got reviewed.

dates	No_of_jobs_reviewed
2020-11-30	180
2020-11-29	180
2020-11-28	218
2020-11-27	35
2020-11-26	64
2020-11-25	80

Calculate 7 day rolling average of throughput? For throughput, do you prefer daily metric or 7-day rolling and why?

```
SELECT ROUND(COUNT(event)/SUM(time_spent),2) AS Weekly_AVG_Throughput
FROM primary_data;
```

Weekly average of throughput was 0.03

Weekly_AVG_Throughput
0.03

```
SELECT ds AS dates,
    ROUND(COUNT(event)/SUM(time_spent),2) AS daily_metric
FROM primary_data
```

```
GROUP BY ds
ORDER BY ds;
```

Weekly throughput metric will show overall value but daily will give a more clear picture on events which are happening per second. As below result shows on NOV 28 has throughput avg highest and NOV 27 has lowest. Analyzing events data for those days will give the team more insights.

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

How will you display duplicates from the table?

By using job_id and actor _id we can derive the duplicate rows.

```
SELECT job_id,
    COUNT(*) AS duplicate_rows
FROM primary_data
GROUP BY job_id
HAVING COUNT(*)>1;
```

job_id	duplicate_rows
23	3

```
SELECT actor_id,
    COUNT(*) AS duplicate_rows
FROM primary_data
GROUP BY actor_id
HAVING COUNT(*)>1;
```

actor_id	duplicate_rows
1003	2

CASE STUDY - 2

Calculate the weekly user engagement?

```
SELECT EXTRACT(WEEK FROM occurred_at) AS week_no,
    COUNT(user_id) AS weekly_user
FROM events
WHERE event_type='engagement'
GROUP BY 1;
```

Analysis of user engagement daily or weekly is important because this will help to decide if the company provides quality product or service to the customer or not. If not how they can improve the quality.

week_no	weekly_user
17	627
18	1626
19	1594
20	1636
21	1685
23	1876
22	1929
24	2114
25	1854
29	1752
26	1988
30	2200
28	1920
27	2004
31	1857
32	1982
33	2111
34	2212
35	156

Calculate the user growth for a product?

```
SELECT Month_no,
      ROUND(((total_users/LAG(total_users,1) OVER (ORDER BY Month_no)-1)*100),2) AS
user_growth
FROM (
     SELECT EXTRACT(MONTH FROM created_at) AS Month_no,
           COUNT(activated_at) AS total_users
            FROM users
     WHERE activated_at NOT IN ("")
     GROUP BY 1
     ORDER BY 1
     )subquery ;
```

Month_no	user_growth
1	NULL
2	-3.79
3	11.68
4	18.56
5	9.48
6	9.37
7	17.96
8	5.15
9	-75.50
10	18.18
11	2.31
12	21.80

Calculate the weekly retention of users-sign up cohort?

```
SELECT signup_week AS "week_number",
 SUM(CASE WHEN week_no=0 THEN 1 ELSE 0 END) AS "WEEK 1",
 SUM(CASE WHEN week_no=1 THEN 1 ELSE 0 END) AS "WEEK 2",
 SUM(CASE WHEN week_no=2 THEN 1 ELSE 0 END) AS "WEEK 3",
 SUM(CASE WHEN week_no=3 THEN 1 ELSE 0 END) AS "WEEK 4",
 SUM(CASE WHEN week_no=4 THEN 1 ELSE 0 END) AS "WEEK 5",
 SUM(CASE WHEN week_no=5 THEN 1 ELSE 0 END) AS "WEEK 6",
 SUM(CASE WHEN week_no=6 THEN 1 ELSE 0 END) AS "WEEK 7",
 SUM(CASE WHEN week_no=7 THEN 1 ELSE 0 END) AS "WEEK 8",
 SUM(CASE WHEN week_no=8 THEN 1 ELSE 0 END) AS "WEEK 9",
 SUM(CASE WHEN week_no=9 THEN 1 ELSE 0 END) AS "WEEK 10",
 SUM(CASE WHEN week_no=10 THEN 1 ELSE 0 END) AS "WEEK 11",
 SUM(CASE WHEN week_no=11 THEN 1 ELSE 0 END) AS "WEEK 12",
 SUM(CASE WHEN week_no=12 THEN 1 ELSE 0 END) AS "WEEK 13",
 SUM(CASE WHEN week_no=13 THEN 1 ELSE 0 END) AS "WEEK 14",
 SUM(CASE WHEN week_no=14 THEN 1 ELSE 0 END) AS "WEEK 15",
 SUM(CASE WHEN week_no=15 THEN 1 ELSE 0 END) AS "WEEK 16",
 SUM(CASE WHEN week_no=16 THEN 1 ELSE 0 END) AS "WEEK 17",
 SUM(CASE WHEN week_no=17 THEN 1 ELSE 0 END) AS "WEEK 18",
 SUM(CASE WHEN week_no=18 THEN 1 ELSE 0 END) AS "WEEK 19"
FROM (
    SELECT
login.user_id,login.login_week,signup.signup_week,login.login_week-signup.signup_week as
week_no
```

```
    FROM(SELECT user_id, MIN(EXTRACT(WEEK FROM occurred_at)) AS signup_week
FROM events GROUP BY 1)signup,
                (SELECT user_id, EXTRACT(WEEK FROM occurred_at) AS login_week
FROM events GROUP BY 1,2)login
        WHERE login.user_id=signup.user_id
)sub
GROUP BY signup_week
ORDER BY signup_week;
```

week_number	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
17	87	9	3	4	2	2	2	1	0	0	0	0	0	1
18	188	15	4	3	0	1	1	0	0	1	1	0	0	0
19	193	13	9	7	2	1	1	1	1	0	0	0	0	0
20	177	13	8	3	3	1	0	0	0	0	0	0	0	0
21	183	19	7	1	2	0	0	0	1	0	0	0	0	0
22	197	16	5	5	4	0	0	0	1	0	0	0	0	0
23	196	15	6	3	1	2	1	0	0	0	0	0	0	0
24	230	26	8	4	2	2	1	2	0	0	0	0	0	0
25	207	15	6	2	1	0	1	1	0	0	0	0	0	0
26	201	16	8	2	2	0	0	0	0	0	0	0	0	0
27	222	11	5	2	3	0	1	1	0	0	0	0	0	0

Calculate the weekly engagement per device?

User engagement for each product will help the team to decide which product requires improvement and to understand which products are most liked by the customers.

```
SELECT DISTINCT device from events;

SELECT EXTRACT(WEEK FROM occurred_at) AS week_no,
    COUNT(DISTINCT CASE WHEN device IN("dell inspiron notebook") THEN user_id
ELSE NULL END) AS "dell inspiron notebook",
    COUNT(DISTINCT CASE WHEN device IN("iphone 5") THEN user_id ELSE NULL
END) AS "iphone 5",
    COUNT(DISTINCT CASE WHEN device IN("iphone 4s") THEN user_id ELSE NULL
END) AS "iphone 4s",
        COUNT(DISTINCT CASE WHEN device IN("windows surface") THEN user_id ELSE
NULL END) AS "windows surface",
    COUNT(DISTINCT CASE WHEN device IN("macbook air") THEN user_id ELSE NULL
END) AS "windows surface",
```

```
    COUNT(DISTINCT CASE WHEN device IN("iphone 5s") THEN user_id ELSE NULL
END) AS "iphone 5s",
        COUNT(DISTINCT CASE WHEN device IN("macbook pro") THEN user_id ELSE
NULL END) AS "macbook pro",
        COUNT(DISTINCT CASE WHEN device IN("kindle fire") THEN user_id ELSE
NULL END) AS "kindle fire",
        COUNT(DISTINCT CASE WHEN device IN("ipad mini") THEN user_id ELSE NULL
END) AS "ipad mini",
        COUNT(DISTINCT CASE WHEN device IN("nexus 7") THEN user_id ELSE NULL
END) AS "nexus 7",
    COUNT(DISTINCT CASE WHEN device IN("nexus 5") THEN user_id ELSE NULL END)
AS "nexus 5",
    COUNT(DISTINCT CASE WHEN device IN("samsung galaxy s4") THEN user_id ELSE
NULL END) AS "samsung galaxy s4",
    COUNT(DISTINCT CASE WHEN device IN("lenovo thinkpad") THEN user_id ELSE
NULL END) AS "lenovo thinkpad",
    COUNT(DISTINCT CASE WHEN device IN("samsumg galaxy tablet") THEN user_id
ELSE NULL END) AS "samsumg galaxy tablet",
    COUNT(DISTINCT CASE WHEN device IN("acer aspire notebook") THEN user_id ELSE
NULL END) AS "acer aspire notebook",
    COUNT(DISTINCT CASE WHEN device IN("asus chromebook") THEN user_id ELSE
NULL END) AS "asus chromebook",
    COUNT(DISTINCT CASE WHEN device IN("samsung galaxy note") THEN user_id ELSE
NULL END) AS "samsung galaxy note",
    COUNT(DISTINCT CASE WHEN device IN("mac mini") THEN user_id ELSE NULL
END) AS "mac mini",
    COUNT(DISTINCT CASE WHEN device IN("hp pavilion desktop") THEN user_id ELSE
NULL END) AS "hp pavilion desktop",
    COUNT(DISTINCT CASE WHEN device IN("ipad air") THEN user_id ELSE NULL END)
AS "ipad air",
    COUNT(DISTINCT CASE WHEN device IN("htc one") THEN user_id ELSE NULL END)
AS "htc one",
    COUNT(DISTINCT CASE WHEN device IN("dell inspiron desktop") THEN user_id ELSE
NULL END) AS "dell inspiron desktop",
    COUNT(DISTINCT CASE WHEN device IN("amazon fire phone") THEN user_id ELSE
NULL END) AS "amazon fire phone",
    COUNT(DISTINCT CASE WHEN device IN("acer aspire desktop") THEN user_id ELSE
NULL END) AS "acer aspire desktop",
```

```
    COUNT(DISTINCT CASE WHEN device IN("nokia lumia 635") THEN user_id ELSE
NULL END) AS "nokia lumia 635",
    COUNT(DISTINCT CASE WHEN device IN("nexus 10") THEN user_id ELSE NULL
END) AS "nexus 10"
FROM events
WHERE event_type="engagement"
GROUP BY 1
ORDER BY 1;
```

week_no	dell inspiron notebook	iphone 5	iphone 4s	windows surface	windows surface	iphone 5s	macbook pro	kindle fire	ipad mini	nexus 7	nexus 5	samsung galaxy s4
17	4	11	3	0	4	5	14	0	3	4	4	7
18	12	7	4	1	18	9	43	5	7	5	9	13
19	7	19	7	3	23	9	38	4	1	7	16	14
20	9	16	5	2	17	13	39	2	6	3	10	10
21	12	24	9	3	13	10	33	3	4	3	16	9
22	16	11	9	2	31	13	33	3	4	8	15	11
23	11	20	8	2	22	13	34	5	6	5	8	11
24	11	24	8	7	28	12	42	4	5	10	11	19
25	21	19	4	5	16	7	42	2	3	7	8	18
26	11	25	8	5	21	17	40	2	4	10	7	12
27	8	26	16	5	17	14	53	4	2	9	11	16
28	16	18	7	5	25	10	40	6	4	5	10	22
29	15	22	8	3	26	12	41	7	7	6	10	16
30	16	24	12	2	15	17	38	1	5	6	19	16

Calculate the email engagement metrics?

Email engagement metrics can help to increase the rate of active users by analyzing how many users are interacting with the emails. This will help companies to understand which product/service the users are most interested in. Team can increase user growth by sending the emails to inactive users.

```
SELECT DISTINCT action FROM email_events;
```

```
SELECT week_no,
      ROUND((100*sent_weekly_digest/total),2) AS weekly_digest_percentage,
      ROUND((100*email_open/total),2) AS email_open_percentage,
      ROUND((100*email_clickthrough/total),2) AS email_clickthroughs_percentage,
      ROUND((100*sent_reengagement_email/total),2) AS sent_reengagement_email_percentage
FROM(
    SELECT EXTRACT(WEEK FROM occurred_at) AS week_no,
    COUNT(CASE WHEN action='sent_weekly_digest' THEN user_id ELSE NULL END) AS
sent_weekly_digest,
```

```
    COUNT(CASE WHEN action='email_open' THEN user_id ELSE NULL END) AS
email_open,
    COUNT(CASE WHEN action='email_clickthrough' THEN user_id ELSE NULL END) AS
email_clickthrough,
    COUNT(CASE WHEN action='sent_reengagement_email' THEN user_id ELSE NULL
END) AS sent_reengagement_email,
        COUNT(user_id) AS total
        FROM email_events
    GROUP BY week_no
    )sub
GROUP BY week_no
ORDER BY week_no;
```

week_no	weekly_digest_percentage	email_open_percentage	email_clickthroughs_percentage	sent_reengagement_email_percentage
17	62.32	21.28	11.39	5.01
18	63.45	22.24	10.49	3.83
19	62.16	22.67	11.13	4.04
20	61.62	22.64	11.43	4.31
21	63.52	22.82	9.97	3.69
22	63.59	21.56	10.66	4.19
23	62.39	22.34	11.18	4.09
24	61.61	22.92	10.99	4.48
25	63.77	21.79	10.54	3.90
26	62.99	22.22	10.61	4.18
27	62.24	22.49	11.37	3.90
28	62.92	22.48	10.77	3.83
29	63.98	21.71	10.51	3.79
30	62.29	23.24	10.59	3.88
31	65.27	23.25	7.66	3.82
32	66.59	22.85	7.14	3.42
33	64.73	23.10	7.91	4.26
34	64.33	23.91	7.67	4.08
35	0.00	32.28	29.92	37.80