Project on

Operation Analytics and Investigating Metric Spike

<u>Project Description</u> :

- ### Operation Analytics:

Operational Analytics is a category of business analytics that's focused on syncing data from your warehouse directly to your business tools – thus ensuring that everyone across your organization has access to the same data regardless of their technical skills. Operations analytics is a subfield of data analytics that focuses on using data to optimize and improve operational processes within an organization .

- ### Metric Spike:

A metric spike refers to a sudden and significant increase in a particular metric or measurement that is being tracked or analyzed. This can occur in a wide range of data sets and can be observed over a relatively short period of time.

This Project focuses on analyzing the data which is provided by the company. With the help of this, the company then finds the areas on which it must improve upon. My task is to derive insights and answer the question asked by different department like -

In case study-1 the insights are found based on following questions:

A.Number of jobs reviewed: Amount of jobs reviewed over time.

B.Throughput: It is the no. of events happening per second.

C.Percentage share of each language: Share of each language for different contents.

D.Duplicate rows: Rows that have the same value present in them.

In case study-2 the insights are found based on following questions:

User Engagement: To measure the activeness of a user. Measuring if the user finds quality in a product/service.

User Growth: Amount of users growing over time for a product.

Weekly Retention: Users getting retained weekly after signing-up for a product.

Weekly Engagement: To measure the activeness of a user. Measuring if the user finds quality in a product/service weekly.

Email Engagement: Users engaging with the email service.

Approach:

- Download all Data provided
- Create database with the dataset provided for Case Study 1 (Job Data)
- Upload datasets of Case Study 2 (Investigating metric spike) to MySQL Workbench
- Gather insights of database
- Write queries to find answers
- Analyze the information and make proper decision

Tech-Stack Used:

MYSQL Workbench version 8.0.30. The main purpose of using MySQL workbench is that it provides the console to simply edit 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:

The Knowledge I gained from this Project is how to Create a database with the reference of Excel sheet provided and import large / Uploaded datasets (Investigating metric spike) to MySQL Workbench as all the questions asked could be answered through SQL .

Result :

By looking at the questions asked we get following output queries

Data Analysis:

Case Study 1 (Job Data)

A. **Number of jobs reviewed:** Amount of jobs reviewed over time.

-- 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 'JObs reviewed
    per hour per day'
FROM
    job_data
WHERE
    ds BETWEEN '2020-11-01' AND '2020-11-30'
GROUP BY ds
order by ds;
```

Output:

#	DATES	JObs reviewed per hour per day
1	2020-11-25	80
2	2020-11-26	64
3	2020-11-27	35
4	2020-11-28	218
5	2020-11-29	180
6	2020-11-30	180

Conclusion:

On the date 2020-11-28 there is a maximum number of jobs reviewed that is 218.

B.Throughput: It is the no. of events happening per second.

-- 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 throughput'

  From job_data;
```

Output:

#	weekly throughput
1	0.03

Conclusion:

The weekly throughput is 0.03 .

```
select ds as dates,round(Count(event)/sum(time_spent),2)as 'Daily
Throughput'  From job_data

  group by ds

  order by ds ;
```

Output:

#	dates	Daily Throughput
1	2020-11-25	0.02
2	2020-11-26	0.02
3	2020-11-27	0.01
4	2020-11-28	0.06
5	2020-11-29	0.05
6	2020-11-30	0.05

Conclusion:

On the date 2020-11-28 the throughput is highest i.e 0.06.

C.Percentage share of each language: Share of each language for different contents.

QUERY :

-- C. Calculate the percentage share of each language in the last 30 days?

```
select language,count(job_data.language)as
```

```
total,((count(job_data.language)/(select count(*)from
job_data))*100)per_share_of_each_lang

from job_data

group by job_data.language;
```

Output:

#	language	total	per_share_of_each_lang
1	English	1	12.5000
2	Arabic	1	12.5000
3	Persian	3	37.5000
4	Hindi	1	12.5000
5	French	1	12.5000
6	Italian	1	12.5000

Conclusion:

Persian language is highest with 37.5%

D.Duplicate rows: Rows that have the same value present in them.

QUERY :

-- D.Let's say you see some duplicate rows in the data. How will you display duplicates from the table?

```
SELECT * FROM(SELECT*,ROW_NUMBER() OVER(PARTITION BY JOB_ID) AS
ROWNUM FROM job_data)A WHERE

ROWNUM>1;
```

Output:

#	ds	job_id	actor_id	event	language	time_spent	org	ROWNUM
1	2020-11-28	23	1005	transfer	Persian	22	D	2
2	2020-11-26	23	1004	skip	Persian	56	A	3

Conclusion:

Job_id number 23 has duplicate rows.

Case Study 2 (Investigating metric spike)

A.User Engagement: To measure the activeness of a user. Measuring if the user finds quality in a product/service.

QUERY :

-- 1. Calculate the weekly user engagement?

```
select extract(week from occurred_at)as week_number,

count(distinct user_id)as users

from events

group by week_number;
```

Output:

#	week_number	users
1	17	740
2	18	1260
3	19	1287
4	20	1351
5	21	1299
6	22	1381
7	23	1446
8	24	1471
9	25	1459
10	26	1509
11	27	1573
12	28	1577
13	29	1607
14	30	1706
15	31	1514
16	32	1454
17	33	1438
18	34	1443
19	35	118

 Conclusion: On the 30th week the highest number of users were active i.e 1706

B.User Growth: Amount of users growing over time for a product.

QUERY :

-- 2.Calculate the user growth for the product?

```sql
select years,week_number,num_users,

sum(num_users)over(order by years,week_number rows between unbounded
preceding and current row) as cumulative_active_usurs

from

(

select extract(year from activated_at) as years,

extract(week from activated_at) as week_number,

count(distinct user_id)as num_users

from users

where state='active'

group by years, week_number

order by years, week_number)A;
```

Output:

years	week_number	num_users	cumulative_active_usurs
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
2013	12	33	468
2013	13	39	507
2013	14	35	542
2013	15	43	585
2013	16	46	631
2013	17	49	680
2013	18	44	724
2013	19	57	781
2013	20	25	806

C.Weekly Retention: Users getting retained weekly after signing-up for a product

-- 3.Calculate the weekly retention of users-sign up cohort?

```
SELECT EXTRACT(YEAR FROM OCCURRED_AT)AS YEAR,

EXTRACT(WEEK FROM OCCURRED_AT)AS WEEK_NUMBER,

DEVICE, COUNT(DISTINCT USER_ID) USER_TYPE

FROM events

WHERE EVENT_TYPE ='ENGAGEMENT'

GROUP BY 1,2,3

ORDER BY 1,2,3;
```

Output:

YEAR	WEEK_NUMBER	DEVICE	USER_TYPE
2014	17	acer aspire desktop	9
2014	17	acer aspire notebook	20
2014	17	amazon fire phone	4
2014	17	asus chromebook	21
2014	17	dell inspiron desktop	18
2014	17	dell inspiron notebook	46
2014	17	hp pavilion desktop	14
2014	17	htc one	16
2014	17	ipad air	27
2014	17	ipad mini	19
2014	17	iphone 4s	21
2014	17	iphone 5	65
2014	17	iphone 5s	42
2014	17	kindle fire	6
2014	17	lenovo thinkpad	86
2014	17	mac mini	6
2014	17	macbook air	54
2014	17	macbook pro	143
2014	17	nexus 10	16
2014	17	nexus 5	40
2014	17	nexus 7	18

D.Weekly Engagement: To measure the activeness of a user. Measuring if the user finds quality in a product/service weekly.

QUERY :

-- 4.Calculate the weekly engagement per device?

```
SELECT DISTINCT device FROM events;

SELECT WEEK(occurred_at) AS week_of_theYEAR,

COUNT(DISTINCT user_id) AS weekly_users,

COUNT(DISTINCT CASE WHEN device IN ('macbook pro', 'acer aspire
notebook','acer aspire desktop','lenovo thinkpad', 'mac mini', 'dell
inspiron desktop','dell inspiron notebook','windows surface','macbook
air','asus chromebook','hp pavilion desktop') THEN user_id ELSE NULL
END) AS computer,

COUNT(DISTINCT CASE WHEN device IN ('iphone 5s','nokia lumia
635','amazon fire phone','iphone 4s','htc one','iphone 5','samsung
galaxy s4') THEN user_id ELSE NULL END) AS phone,

COUNT(DISTINCT CASE WHEN device IN ('kindle fire','samsung galaxy
note','ipad mini','nexus 7','nexus 10','samsung galaxy tablet','nexus
5','ipad air') THEN user_id ELSE NULL END) AS tablet

FROM events

WHERE event_type = 'engagement'

AND event_name = 'login'

GROUP BY 1

ORDER BY 1;
```

Output:

week_of_theYEAR	weekly_users	computer	phone	tablet
17	663	399	217	137
18	1068	724	375	250
19	1113	736	394	274
20	1154	767	414	286
21	1121	734	391	259
22	1186	805	400	286
23	1232	819	450	283
24	1275	836	441	311
25	1264	858	425	294
26	1302	840	486	294
27	1372	912	497	294
28	1365	934	494	275
29	1376	932	503	288
30	1467	979	500	314
31	1299	938	417	242
32	1225	900	360	219
33	1225	899	350	220
34	1204	887	364	219
35	104	69	21	14

E.Email Engagement: Users engaging with the email service.

QUERY :

-- 5. Calculate the email engagement metrics?

```
SELECT ACTION,

EXTRACT(MONTH FROM OCCURRED_AT) AS MONTH,

COUNT(ACTION) AS NUMBER_OF_MAILS

FROM email_events

GROUP BY ACTION,MONTH

ORDER BY ACTION,MONTH;
```

Output:

ACTION	MONTH	NUMBER_OF_MAILS
email_clickthrough	5	2023
email_clickthrough	6	2274
email_clickthrough	7	2721
email_clickthrough	8	1992
email_open	5	4212
email_open	6	4658
email_open	7	5611
email_open	8	5978
sent_reengagement_email	5	758
sent_reengagement_email	6	889
sent_reengagement_email	7	933
sent_reengagement_email	8	1073
sent_weekly_digest	5	11730
sent_weekly_digest	6	13155
sent_weekly_digest	7	15902
sent_weekly_digest	8	16480

How this project helped me : This project helps me to understand the importance of operation analytics. Through this project I am able to understand how the companies use metric spike as a secret weapon.