

A faint background graphic featuring a bar chart with several bars of varying heights in shades of blue and yellow. A large, light orange arrow points diagonally upwards from the bottom left towards the top right, passing behind the main text.

Operation Analytics and Investigating Metric Spike

Advanced SQL

Swet

Project Description:

- ➔ Operation Analytics is the analysis done for the complete end to end operations of a company. With the help of this, the company then finds the areas on which it must improve upon. I've worked with the ops team, support team, marketing team, etc and help them derive insights out of the data they collect.
- ➔ Being one of the most important parts of a company, this kind of analysis is further used to predict the overall growth or decline of a company's fortune. It means better automation, better understanding between cross-functional teams, and more effective workflows.
- ➔ Investigating metric spike is also an important part of operation analytics as being a Data Analyst I've made the other teams understand questions like- Why is there a dip in daily engagement? Why have sales taken a dip? etc. Questions like these must be answered daily and for that its very important to investigate metric spike.

APPROACH

1. Created the Database and Tables: Created a database and then the tables using the structure and links provided.
2. Perform Analysis: Used SQL to perform entire analysis answering the questions asked. I went through the given dataset and then created the tables for calculating various queries.
3. Additionally, I joined the data bits and structured the tables to derive business insights, fetched the required results and hence, created useful insights for the company to take calculated and planned decisions.

TECH-STACK USED

Software And The Version Used While Making The Project :

1. **MySQL WorkBench 8.0** (For working, analysing and reporting insight\$
2. **Microsoft Power Point**(For presenting the detailed analysi\$

The objective of the project is to find out insights about following :

➔ **Case 1 : JOB DATA**

- i. **Number of jobs reviewed:** Number of jobs reviewed over time
- ii. **Throughput:** It is the no. of events happening per second.
- iii. **Percentage share of each language:** Share of each language for different contents.
- iv. **Duplicate rows:** Rows that have the same value present in them.

➔ **Case 2 : INVESTIGATING METRIC SPIKE**

- i. **User Engagement:** To measure the activeness of a user. Measuring if the user finds quality in a product/service.
- ii. **User Growth:** Number of users growing over time for a product.
- iii. **Weekly Retention:** Users getting retained weekly after signing up for a product.
- iv. **Weekly Engagement:** To measure the activeness of a user. Measuring if the user finds quality in a product/service weekly.
- v. **Email Engagement:** Users engaging with the email service.

Case study 1: JOB DATA

- Creating table job_data

job_id	actor_id	event	language	time_spent	org	ds
21	1001	skip	English	00:00:15	A	2020-11-30
22	1006	transfer	Arabic	00:00:25	B	2020-11-30
23	1003	decision	Persian	00:00:20	C	2020-11-29
23	1005	transfer	Persian	00:00:22	D	2020-11-28
25	1002	decison	Hindi	00:00:11	B	2020-11-28
11	1007	decision	French	00:01:04	D	2020-11-27
23	1004	skip	Persian	00:00:56	A	2020-11-26
20	1003	transfer	Italian	00:00:45	C	2020-11-25

Syntax used for creating database job_db and job_data table.

```
1 • create database job_db
2 ❌ use job_db
3 create table job_data
4 (
5     job_id int,
6     actor_id int,
7     event varchar(50),
8     language varchar(50),
9     time_spent time,
10    org varchar(100),
11    ds date
12 )
13 insert into job_data (job_id, actor_id, event, language, time_spent, org, ds)
14 values
15 ('21', '1001', 'skip', 'English', '15', 'A', '2020-11-30'),
16 ('22', '1006', 'transfer', 'Arabic', '25', 'B', '2020-11-30'),
17 ('23', '1003', 'decision', 'Persian', '20', 'C', '2020-11-29'),
18 ('23', '1005', 'transfer', 'Persian', '22', 'D', '2020-11-28'),
19 ('25', '1002', 'decison', 'Hindi', '11', 'B', '2020-11-28'),
20 ('11', '1007', 'decision', 'French', '104', 'D', '2020-11-27'),
21 ('23', '1004', 'skip', 'Persian', '56', 'A', '2020-11-26'),
22 ('20', '1003', 'transfer', 'Italian', '45', 'C', '2020-11-25')
23 ;
24 • select * from job_data
```

A. Number of jobs reviewed Objective Calculate the number of jobs reviewed per hour per day for November 2020

```
select count(distinct job_id)/(30*24)  
as per_day_jobs  
from job_data
```

	per_day_jobs
▶	0.0083

Less than 0.01 jobs were reviewed each hour of the day throughout the month of November.

B. Throughput (Number of events happening per second) Objective: Calculate 7 day rolling average of throughput

```
select ds, tot_events,  
avg(tot_events) over(order by ds rows between 6 preceding and current row) as 7_day_rolling_average  
from  
(select ds, count(distinct event) as tot_events  
from job_data  
group by ds  
order by ds)sub;
```

	ds	tot_events	7_day_rolling_average
▶	2020-11-25	1	1.0000
	2020-11-26	1	1.0000
	2020-11-27	1	1.0000
	2020-11-28	2	1.2500
	2020-11-29	1	1.2000
	2020-11-30	2	1.3333

Using a 7-day rolling average for throughput can be helpful in understanding trends over time, as it provides a longer-term perspective compared to a daily metric. This can help to smooth out any short-term fluctuations in the data and provide a clearer picture of the overall trend.

C. Percentage share of each language Objective

Calculate the percentage share of each language in the last 30 days

```
select language, count(language) as total_language,  
count(*)*100/sum(count(*))  
over() as percentage  
from job_data  
group by language  
order by language
```

	language	total_language	percentage
▶	Arabic	1	12.5000
	English	1	12.5000
	French	1	12.5000
	Hindi	1	12.5000
	Italian	1	12.5000
	Persian	3	37.5000

Persian Language had the highest share among other languages

D. Duplicate Rows Objective Display duplicate rows if any

```
with cte as (select *,  
row_number() over (partition by job_id) as row_num  
from job_data)  
select * from cte where row_num>1
```

job_id	actor_id	event	language	time_spent	org	ds	row_num
23	1005	transfer	Persian	00:00:22	D	2020-11-28	2
23	1004	skip	Persian	00:00:56	A	2020-11-26	3

The output showed two records as there were two duplicate job id in the dataset

Case Study 2 : Investigating metric spike

1. Creating Users Table

```
select * from users_tbl
```

	user_id	created_at	company_id	language	activated_at	state
▶	0	2013-01-01 20:59:39	5737	english	2013-01-01 21:01:07	active
	1	2013-01-01 13:07:46	28	english		pending
	2	2013-01-01 10:59:05	51	english		pending
	3	2013-01-01 18:40:36	2800	german	2013-01-01 18:42:02	active
	4	2013-01-01 14:37:51	5110	indian	2013-01-01 14:39:05	active
	5	2013-01-01 13:39:51	2463	spanish		pending
	6	2013-01-01 18:37:27	11699	english	2013-01-01 18:38:45	active
	7	2013-01-01 16:19:01	4765	french	2013-01-01 16:20:28	active
	8	2013-01-01 04:38:30	2698	french	2013-01-01 04:40:10	active
	9	2013-01-01 08:04:17	1	french		pending
	10	2013-01-01 09:36:41	10	arabic		pending
	11	2013-01-01 08:07:45	3745	english	2013-01-01 08:09:17	active
	12	2013-01-01 18:05:05	903	english		pending

2. Creating Events table

```
select * from events_tbl
```

	user_id	occurred_at	event_type	event_name	location	device	user_type
▶	10522	2014-05-02 11:02:39	engagement	login	Japan	dell inspiron notebook	3
	10522	2014-05-02 11:02:53	engagement	home_page	Japan	dell inspiron notebook	3
	10522	2014-05-02 11:03:28	engagement	like_message	Japan	dell inspiron notebook	3
	10522	2014-05-02 11:04:09	engagement	view_inbox	Japan	dell inspiron notebook	3
	10522	2014-05-02 11:03:16	engagement	search_run	Japan	dell inspiron notebook	3
	10522	2014-05-02 11:03:43	engagement	search_run	Japan	dell inspiron notebook	3
	10612	2014-05-01 09:59:46	engagement	login	Netherlands	iphone 5	1
	10612	2014-05-01 10:00:18	engagement	like_message	Netherlands	iphone 5	1
	10612	2014-05-01 10:00:53	engagement	send_message	Netherlands	iphone 5	1
	10612	2014-05-01 10:01:24	engagement	home_page	Netherlands	iphone 5	1
	10612	2014-05-01 10:01:52	engagement	like_message	Netherlands	iphone 5	1
	10612	2014-05-01 10:02:17	engagement	home_page	Netherlands	iphone 5	1
	10612	2014-05-01 10:02:51	engagement	view_inbox	Netherlands	iphone 5	1

events_tbl 34 x

3. Creating Email Events table

"LIVE" DATA

```
select * from email_events_tbl
```

	user_id	occurred_at	action	user_type
▶	0	2014-05-06 09:30:00	sent_weekly_digest	1
	0	2014-05-13 09:30:00	sent_weekly_digest	1
	0	2014-05-20 09:30:00	sent_weekly_digest	1
	0	2014-05-27 09:30:00	sent_weekly_digest	1
	0	2014-06-03 09:30:00	sent_weekly_digest	1
	0	2014-06-03 09:30:25	email_open	1
	0	2014-06-10 09:30:00	sent_weekly_digest	1
	0	2014-06-10 09:30:24	email_open	1
	0	2014-06-17 09:30:00	sent_weekly_digest	1
	0	2014-06-17 09:30:23	email_open	1
	0	2014-06-24 09:30:00	sent_weekly_digest	1
	0	2014-07-01 09:30:00	sent_weekly_digest	1
	0	2014-07-08 09:30:00	sent_weekly_digest	1

email_events_tbl 35 x

A. User Engagement Objective Calculate the weekly user engagement

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

week_num	num_users
17	663
18	1068
19	1113
20	1154
21	1121
22	1186
23	1232
24	1275
25	1264
26	1302
27	1372
28	1365

Week 31 posted the highest user engagement and week 18 posted the minimum user engagement

B. User Growth: Amount of users growing over time for a product. Your task: Calculate the user growth for product?

```
61
62 select year, week_num, num_users, sum(num_users)
63 over(order by year, week_num) as cum_users
64 from (
65 select extract(year from created_at) as year, extract(week from created_at) as week_num, count(distinct user_id) as num_users
66 from users_tbl
67 where state='active'
68 group by year, week_num
69 order by year, week_num)sub
```

Result Grid				
Filter Rows:				
	year	week_num	num_users	cum_users
▶	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

The 33th week of 2014 saw the greatest number of users actively engaging with the product or service, while the 35th week of 2014 had the lowest number of active users.

C. Weekly Retention Objective Calculate the weekly retention of the users sign-up cohort

```
with cte1 as (  
  select distinct user_id,  
    Extract(week from occurred_at) as signup_week  
  from events_tbl  
  where event_type = 'signup_flow'  
  and event_name = 'complete_signup' and extract (week from occurred_at) = 18 ),  
cte2 as (select distinct user_id,  
  Extract(week from occurred_at) as engagement_week  
  from events_tbl  
  where event_type = 'engagement')  
select count(user_id) total_engaged_users,  
  sum(case when retention_week > 0 then 1 else 0 end) as retained_users  
from (select a.user_id, a.signup_week,  
  b.engagement_week, b.engagement_week-a.signup_week as retention_week  
  from cte1 a  
  LEFT JOIN cte2 b  
  on a.user_id = b.user_id  
  order by a.user_id ) sub
```



total_engaged_users	retained_users
317	236

- 30% of the users retained in week 18 were retained only for the next 7 days.
- User 11816 was retained for the longest duration of 17 weeks

D. Weekly Engagement Objective Calculate the the weekly engagement per device

```
with cte as (select extract(year from occurred_at)|| '-' || extract(week from occurred_at) as weeknum
device, count(distinct user_id) as usercnt
from events_tbl
where event_type = 'engagement'
group by weeknum, device
order by weeknum)
select weeknum, device, usercnt
from cte
```

weeknum	device	usercnt
2014-18	acer aspire desktop	10
2014-18	acer aspire notebook	21
2014-18	amazon fire phone	4
2014-18	asus chromebook	23
2014-18	dell inspiron desktop	21

Weeks 31 & 32 of the year 2014 had the highest user engagement of 317 users each week for the product and the device being used was 'MacBook Pro ' for both the weeks

E. Email Engagement Objective Calculate the email engagement metrics

```
select
100 * sum(case when email_cat = 'email_open' then 1 else 0 end)/
    sum(case when email_cat = 'email_sent' then 1 else 0 end) as email_open_rate,
100 * sum(case when email_cat = 'email_clicked' then 1 else 0 end)/
    sum(case when email_cat = 'email_sent' then 1 else 0 end) as email_click_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'
    end as email_cat
from email_events ) sub
```

	email_open_rate	email_click_rate
▶	31.1921	10.4745

- Out of the total emails sent, around 35.73% of them were opened and only 15.74% of those emails were clicked

Insights

- Less than 0.01 jobs were reviewed each hour of the day throughout the month of November.
- 7 day rolling average is best for throughput.
- The Persian Language had the highest share among other languages.
- Out of the total emails sent, around 35.73% of them were opened and only 15.74% of those emails were clicked
- The weekly user engagement is highest in 31st week.
- 33rd and 35th week of 2014 were the highest and lowest of user activity engagement respectively.
- Maximum retained users were only retained for a week, the retention rates dropped weekly-week.
- Users who had the highest engagement with the product were operating on Macbook Pro'.
- During the month of August, users received the highest number of weekly digest emails.



Result

- The project's key results included the identification of reviewed jobs and their distribution across languages, the calculation of retention rates, and the identification of retained users through an in-depth analysis that relied on predefined assumptions. SQL is one of the most crucial skills for anyone in a data driven position. Additionally, this project helped me to gain insight of various factors which are crucially important for the business to run for a long period and grow as well. Brainstorming is the key to run successful business.

