

# OPERATION ANALYTICS AND INVESTGATING METRIC SPIKE

## PROJECT DESCRIPTION:

Operation Analytics is the process of analysing a company's overall operations from beginning to end to identify areas for improvement. As a Data Analyst, help them understand the data they collect and use it to make better decisions. The analysis helps the company to predict its growth or decline and optimize its workflows. Investigating metric spikes is also a critical part of this analysis to understand why certain metrics may have gone up or down.

## APPROACH;

The process of gathering information i.e by the reading the description, understanding the dataset and tasks to do. By using an IDE(DBeaver) further continued to write queries for the tasks. Created the table with SQL Query, performed the tasks, faced many errors when working, took a lot of time clearing those errors.

## TECH USED:

DBeaver – An IDE of SQL. The User Interface is easy to understand. No complications. Easy to handle for beginners.

## INSIGHTS:

Considering the first task of this task which was “Calculate the number of jobs reviewed per hour per day for November 2020?”. First I calculated the total hours of time spent for reviewing jobs for November. Then further lead to the calculation of per day. Then per hour, it took a lot of my thinking power to complete this project.

## RESULT:

How to proceed for a task, how to clear errors, how to check whether the output is correct. These were the things learned.

## QUERIES:

## CREATING TABLES:

DBeaver 22.3.5 - <operation\_analytics> Script-2

File Edit Navigate Search SQL Editor Database Window Help

SQL Commit Rollback Auto operation\_analytics operation\_analytics

Enter a part of object

- case\_study2 - localhost
- ig\_clone - localhost:3
- operation\_analytics
  - Databases
    - case\_study2
    - ig\_clone
    - operation\_analytics
      - Tables
      - Views
      - Indexes
      - Procedures
      - Triggers
      - Events
    - sys
    - Users
    - Administer
    - System Info

```
CREATE TABLE job_data
(
    ds DATE,
    job_id INT NOT NULL,
    actor_id INT NOT NULL,
    event VARCHAR(15) NOT NULL,
    language VARCHAR(15) NOT NULL,
    time_spent INT NOT NULL,
    org CHAR(2)
);

INSERT INTO job_data (ds, job_id, actor_id, event, language, time_spent, org)
VALUES ('2020-11-30', 21, 1001, 'skip', 'English', 15, 'A'),
('2020-11-30', 22, 1006, 'transfer', 'Arabic', 25, 'B'),
('2020-11-29', 23, 1003, 'decision', 'Persian', 20, 'C'),
('2020-11-28', 23, 1005, 'transfer', 'Persian', 22, 'D'),
('2020-11-28', 25, 1002, 'decision', 'Hindi', 11, 'B'),
('2020-11-27', 11, 1007, 'decision', 'French', 104, 'D'),
('2020-11-26', 23, 1004, 'skip', 'Persian', 56, 'A'),
('2020-11-25', 20, 1003, 'transfer', 'Italian', 45, 'C');
```

Statistics 1

Name	Value
Updated Rows	8
Query	INSERT INTO job_data (ds, job_id, actor_id, event, language, time_spent, org)
	VALUES ('2020-11-30', 21, 1001, 'skip', 'English', 15, 'A'),
	('2020-11-30', 22, 1006, 'transfer', 'Arabic', 25, 'B'),
	('2020-11-29', 23, 1003, 'decision', 'Persian', 20, 'C'),
	('2020-11-28', 23, 1005, 'transfer', 'Persian', 22, 'D'),
	('2020-11-28', 25, 1002, 'decision', 'Hindi', 11, 'B'),
	('2020-11-27', 11, 1007, 'decision', 'French', 104, 'D'),
	('2020-11-26', 23, 1004, 'skip', 'Persian', 56, 'A'),
	('2020-11-25', 20, 1003, 'transfer', 'Italian', 45, 'C')
Start time	Tue Mar 07 16:29:20 IST 2023
Finish time	Tue Mar 07 16:29:20 IST 2023

Project - G... x

Name DataSource

- Bookma
- Diagram
- Scripts

## CASE STUDY-1 (JOB DATA)

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

```
select count(job_id)/(30*24) as num_jobs_reviewed
```

```
from job_data where ds between "2020-11-01" and "2020-11-30";
```

The screenshot shows the DBeaver 22.3.5 interface. The SQL Editor window displays the following query:

```
select count(job_id)/(30*24) as num_jobs_reviewed
from job_data where ds between "2020-11-01" and "2020-11-30";
```

The Results window shows the following data:

num_jobs_reviewed
0.0111

The status bar at the bottom indicates: 1 row(s) fetched - 1ms, on 2023-03-07 at 16:38:12.

2. Let's say the above metric is called throughput. Calculate 7 day rolling average of throughput? For throughput, do you prefer daily metric or 7-day rolling and why?

```
select ds, jobs_reviewed, avg(jobs_reviewed)over(order by ds rows between 6
preceding and current row) as rolling_average from
```

(

```
select ds, count(distinct job_id) as jobs_reviewed
```

```
from job_data where ds between "2020-11-01" and "2020-11-30" group by ds
order by ds
```

```
)a;
```

DBeaver 22.3.5 - <operation\_analytics> Script-3

File Edit Navigate Search SQL Editor Database Window Help

SQL Commit Rollback Auto operation\_analytics operation\_analytics

D... x Pr... x \*<operation\_analytics> Script-2 \*<operation\_analytics> Script-3 x

Enter a part of object

- case\_study2 - localhost
- ig\_clone - localhost:3306
- operation\_analytics -
  - Databases
    - case\_study2
    - ig\_clone
    - operation\_analytics
      - Tables
      - Views
      - Indexes
      - Procedures
      - Triggers
      - Events
    - sys
    - Users
    - Administer
    - System Info

```

select ds, jobs_reviewed, avg(jobs_reviewed)over(order by ds rows between 6 preceding and current row) as rolling_average from
(
select ds, count(distinct job_id) as jobs_reviewed
from job_data where ds between "2020-11-01" and "2020-11-30" group by ds order by ds
)a;

```

job\_data 1 x

select ds, jobs\_reviewed, avg(jobs\_reviewed)over(order by ds) Enter a SQL expression to filter results (use Ctrl+Space)

ds	jobs_reviewed	rolling_average
2020-11-25	1	1
2020-11-26	1	1
2020-11-27	1	1
2020-11-28	2	1.25
2020-11-29	1	1.2
2020-11-30	2	1.3333

Refresh Save Cancel Export data 200 6 6 row(s) fetched - 2ms, on 2023-03-07 at 16:41:10

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

**SELECT** language,

num\_jobs,

**100** \* (num\_jobs / total\_jobs) **AS** percentage\_share

**FROM** (

**SELECT** language,

**COUNT**(job\_id) **AS** num\_jobs

**FROM** job\_data

**WHERE** ds **LIKE** '2020-11-%'

**GROUP BY** language

) a

**CROSS JOIN** (

**SELECT** **COUNT**(job\_id) **AS** total\_jobs

**FROM** job\_data

**WHERE** ds **LIKE** '2020-11-%'

) b;

The screenshot shows the DBeaver 22.3.5 interface. The top panel displays two SQL scripts. Script-2 contains a query to count distinct job IDs by date range. Script-3 contains a query to calculate the percentage share of jobs by language. The bottom panel shows the result of the query in Script-3, displayed in a table grid.

**Script-2:**

```
select ds, count(distinct job_id) as jobs_reviewed
from job_data where ds between "2020-11-01" and "2020-11-30" group by ds order by ds
)a;
```

**Script-3:**

```
SELECT language,
num_jobs,
100 * (num_jobs / total_jobs) AS percentage_share
FROM (
SELECT language,
COUNT(job_id) AS num_jobs
FROM job_data
WHERE ds LIKE '2020-11-%'
GROUP BY language
) a
CROSS JOIN (
SELECT COUNT(job_id) AS total_jobs
FROM job_data
WHERE ds LIKE '2020-11-%'
) b;
```

**Query Result Table:**

language	num_jobs	percentage_share
English	1	12.5
Arabic	1	12.5
Persian	3	37.5
Hindi	1	12.5
French	1	12.5
Italian	1	12.5

4. 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;
```

DBEaver 22.3.5 - <operation\_analytics> Script-3

File Edit Navigate Search SQL Editor Database Window Help

Auto operation\_analytics operation\_analytics

operation\_analytics> Script-2 operation\_analytics> Script-3

Enter a part of object

- case\_study2 - localhost:3306
- ig\_clone - localhost:3306
- operation\_analytics
  - Databases
    - case\_study2
    - ig\_clone
    - operation\_analytics
      - Tables
      - Views
      - Indexes
      - Procedures
      - Triggers
      - Events
    - sys
    - Users
    - Administer
    - System Info

```

SELECT language,
num_jobs,
100 * (num_jobs / total_jobs) AS percentage_share
FROM (
SELECT language,
COUNT(job_id) AS num_jobs
FROM job_data
WHERE ds LIKE '2020-11-%'
GROUP BY language
) a
CROSS JOIN (
SELECT COUNT(job_id) AS total_jobs
FROM job_data
WHERE ds LIKE '2020-11-%'
) b;

```

select \* from(select \*,row\_number()over(partition by job\_id) as rownum from job\_data)a where rownum>1;

job\_data 1

select \* from(select \*,row\_number()over(partition by job\_id) as rownum from job\_data)a where rownum>1;

ds	job_id	actor_id	event	language	time_spent	org	rownum
2020-11-28	23	1,005	transfer	Persian	22	D	2
2020-11-26	23	1,004	skip	Persian	56	A	3

Value X

2020-11-28

Project - G... x

Name DataSource

- Bookma
- Diagram
- Scripts

Refresh Save Cancel Export data 200 2 2 row(s) fetched - 2ms (1ms fetch), on 2023-03-07 at 16:48:04

## CASE STUDY 2 (INVESTIGATING METRIC SPIKE)

1. Calculate the weekly user engagement?

```
select extract(week from occurred_at) as weeknum,  
  
count(distinct user_id) from events group by weeknum;
```

The screenshot shows the DBeaver 22.3.5 interface. The SQL Editor contains the query: `select extract(week from occurred_at) as weeknum, count(distinct user_id) from events group by weeknum;`. The Results window displays 19 rows of data. The left sidebar shows the database structure with 'operation\_metrics' selected. The bottom status bar indicates '19 row(s) fetched - 431ms (1ms fetch), on 2023-03-07 at 17:19:41'.

weeknum	count(distinct user_id)
17	740
18	1,260
19	1,287
20	1,351
21	1,299
22	1,381
23	1,446
24	1,471
25	1,459
26	1,509
27	1,573
28	1,577
29	1,607

2. Calculate the user growth for product

```
select year, weeknum, num_active_user, sum(num_active_user) over(order by  
year, weeknum rows between unbounded preceding and current row) as  
cum_active_users
```

```
from (select extract(year from activated_at) as year, extract(week from  
activated_at) as weeknum, count(distinct user_id) as num_active_user
```

```
from users a where state="active" group by year, weeknum order by  
year, weeknum) a;
```

DBeaver 22.3.5 - <operation\_metrics> Script-4

File Edit Navigate Search SQL Editor Database Window Help

SQL Commit Rollback Auto operation\_metrics operation\_metrics

Database Navigator Projects

Enter a part of object name here

- case\_study2 - localhost:3306
- ig\_clone - localhost:3306
- operation\_metrics - localhost:3306
  - Databases
  - case\_study2
  - ig\_clone
  - operation\_metrics
    - Tables
    - email\_events
    - events
    - Columns
      - 123 user\_id (double)
      - 123 occurred\_at (varchar(50))
      - 123 event\_type (varchar(50))
      - 123 event\_name (varchar(50))
      - 123 location (varchar(50))
      - 123 device (varchar(50))
      - 123 user\_type (double)
    - Constraints
    - Foreign Keys
    - References
    - Triggers
    - Indexes

Script-2

```

weeknum,
year,
num_active_user,
SUM(num_active_user)over(order by year, weeknum rows between unbounded preceding and current row)as cum_active_users
FROM
(
select
extract(year from activated_at) as year,
extract(week from activated_at)as weeknum,
count(distinct user_id)as num_active_user
FROM
users
WHERE
state='active'
group by
weeknum,
year
order BY
weeknum,
year
)a

```

Results 1 x

select weeknum, year, num\_active\_user, SUM(num\_active\_user)over(order by year, weeknum rows between unbounded preceding and current row)as cum\_active\_users

	123 weeknum	123 year	123 num_active_user	123 cum_active_users
1	0	2,013	23	23
2	1	2,013	30	53
3	2	2,013	48	101
4	3	2,013	36	137
5	4	2,013	30	167
6	5	2,013	48	215
7	6	2,013	38	253
8	7	2,013	42	295
9	8	2,013	34	329
10	9	2,013	43	372
11	10	2,013	32	404
12	11	2,013	31	435
13	12	2,013	33	468

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

**SELECT**

**COUNT**(user\_id)**as** users\_\_,

**SUM**(**CASE WHEN** retention\_week = 1 **THEN** 1 **ELSE** 0 **END** ) **AS** week\_1,

**SUM**(**CASE WHEN** retention\_week = 2 **THEN** 1 **ELSE** 0 **END** ) **AS** week\_2,

**SUM**(**CASE WHEN** retention\_week = 3 **THEN** 1 **ELSE** 0 **END** ) **AS** week\_3,

**SUM**(**CASE WHEN** retention\_week = 4 **THEN** 1 **ELSE** 0 **END** ) **AS** week\_4,

**SUM**(**CASE WHEN** retention\_week = 5 **THEN** 1 **ELSE** 0 **END** ) **AS** week\_5

**FROM**

(

**SELECT** a.user\_id,

a.sign\_up\_week,

b.engagement\_week,

b.engagement\_week - a.sign\_up\_week **as** retention\_week

**FROM**

(



```
(select distinct user_id, extract(week from occurred_at) as sign_up_week
from events
where event_type='signup_flow'
and event_name='complete_signup'
and extract(week from occurred_at)=18
)a

left JOIN

(
select distinct user_id,
extract(week from occurred_at) as engagement_week
from events
where event_type='engagement'
)b

on a.user_id=b.user_id

)

order by
a.user_id )a
```

DBEAVER 22.3.5 - <operation\_metrics> Script-4

File Edit Navigate Search SQL Editor Database Window Help

Auto operation\_metrics operation\_metrics

Database Navigator Projects

Enter a part of object name here

- case\_study2 - localhost:3306
- ig\_clone - localhost:3306
- operation\_metrics - localhost:3306
  - Databases
    - case\_study2
    - ig\_clone
    - operation\_metrics
      - Tables
        - email\_events
        - events
          - Columns
            - 123 user\_id (double)
            - occurred\_at (varchar(50))
            - event\_type (varchar(50))
            - event\_name (varchar(50))
            - location (varchar(50))
            - device (varchar(50))
            - user\_type (double)
          - Constraints
          - Foreign Keys
          - References
          - Triggers
          - Indexes

Project - General

Name DataSource

Bookmarks

Diagrams

Scripts

```

SELECT
COUNT(user_id)as users___,
SUM(CASE WHEN retention_week = 1 THEN 1 ELSE 0 END ) AS week_1,
SUM(CASE WHEN retention_week = 2 THEN 1 ELSE 0 END ) AS week_2,
SUM(CASE WHEN retention_week = 3 THEN 1 ELSE 0 END ) AS week_3,
SUM(CASE WHEN retention_week = 4 THEN 1 ELSE 0 END ) AS week_4,
SUM(CASE WHEN retention_week = 5 THEN 1 ELSE 0 END ) AS week_5
FROM
(
SELECT a.user_id,
a.sign_up_week,
b.engagemet_week,
b.engagemet_week - a.sign_up_week as retention_week
FROM
(
(select distinct user_id, extract(week from occurred_at) as sign_up_week
from events
where event_type='signup_flow'
and event_name='complete_signup')

```

Results 1

Grid

	123 users___	123 week_1	123 week_2	123 week_3	123 week_4	123 week_5	Value
1	615	114	73	49	37	26	615

Refresh Save Cancel Export data 200 1 1 row(s) fetched - 601ms, on 2023-03-07 at 17:37:39

IST en Writable Smart Insert 30 : 1 [890] Set: 890 | 32

4. Calculate the weekly engagement per device?

```

select extract(year from occurred_at)as year,
extract(week from occurred_at)as week, device,
count(distinct user_id) from events
where event_type="engagement"
group by 1,2,3
order by 1,2,3;

```

DBeaver 22.3.5 - <operation\_metrics> Script-4

File Edit Navigate Search SQL Editor Database Window Help

Database Navigator × Projects × <none> Script-2 × <operation\_metrics> Script-4 ×

Enter a part of object name here

- > \ case\_study2 - localhost:3306
- > \ ig\_clone - localhost:3306
- > \ operation\_metrics - localhost:3306
  - > Databases
    - > case\_study2
    - > ig\_clone
    - > operation\_metrics
      - > Tables
        - > email\_events
        - > events
      - > Columns
        - 123 user\_id (double)
        - occurred\_at (varchar(50))
        - event\_type (varchar(50))
        - event\_name (varchar(50))
        - location (varchar(50))
        - device (varchar(50))
        - 123 user\_type (double)
      - > Constraints
      - > Foreign Keys
      - > References
      - > Triggers
      - > Indexes

users

```
WHERE
state='active'
group by
weeknum,
year
order BY
weeknum,
year
)a
```

```
select extract(year from occurred_at)as year,
extract(week from occurred_at)as week, device,
count(distinct user_id) from events
where event_type='engagement'
group by 1,2,3
order by 1,2,3;
```

events 1 ×

```
select extract(year from occurred_at)as year, extract(week from occurred_at)as week, device, count(distinct user_id)
```

year	week	device	count(distinct user_id)
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

Refresh Save Cancel Export data 200 200+ 200 row(s) fetched - 715ms, on 2023-03-07 at 17:32:11

IST en Writable Smart Insert 32:1 [197] Sel: 197 | 6

## 5. Calculate the email engagement metrics?

**select**

```
100.0 *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_opened_rate,
```

```
100.0* 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_clicked_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')

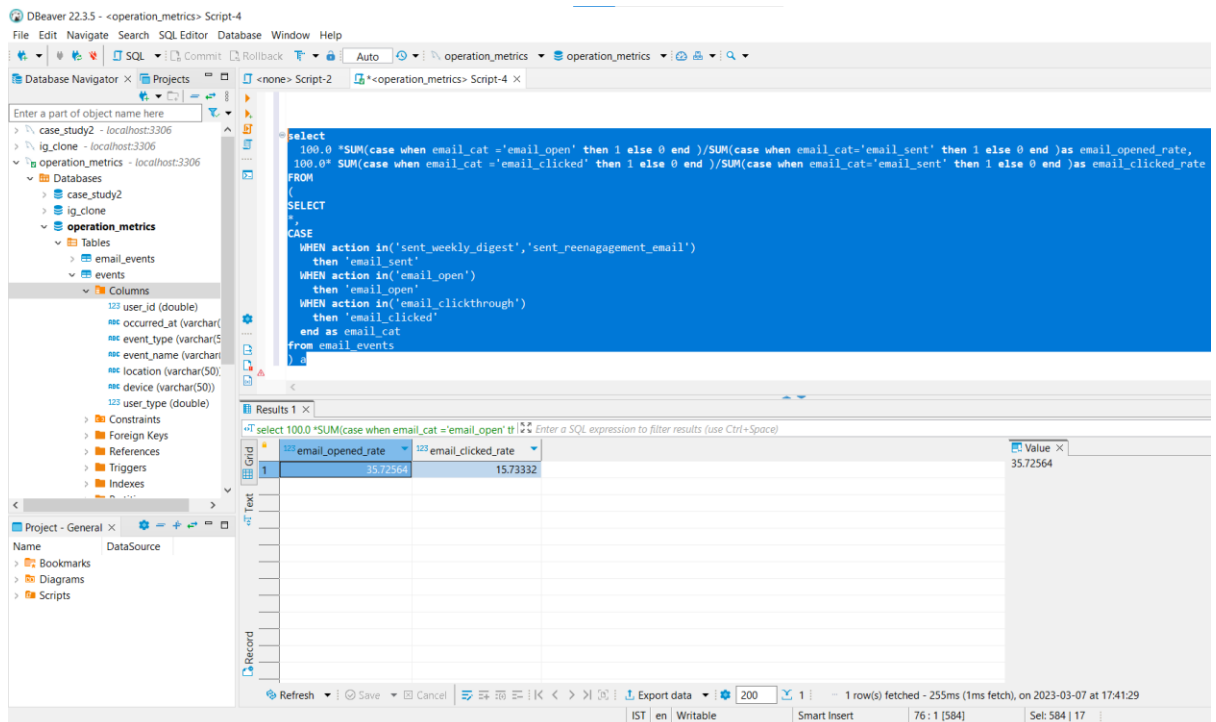
then 'email_clicked'

end as email_cat

from email_events

) a

```



**ALL THE QUERIES ARE EXECUTED SUCCESSFULLY AND GIVES A VALID OUTPUT!!!**

