

OPERATION ANALYTICS AND INVESTIGATION MATRIC SPIKE

By- ANJALI RAJPUT

PROJECT DESCRIPTION

THE MAIN PURPOSE OF THE PROJECT IS TO UNDERSTAND SUDDEN CHANGES IN THE KEY INDICATORS AND DISCOVER MEANINGFUL INSIGHTS THAT WILL ENRICH THE OPERATION OF THE BUSINESS. WE LOOK AT THIS INFORMATION FROM THE FOLLOWING PERSPECTIVE:

CASE STUDY 1: JOB DATA ANALYSIS

- JOBS REVIEWED OVER TIME
- THROUGHPUT ANALYSIS
- LANGUAGE SHARE ANALYSIS
- DUPLICATE ROWS DETECTION

CASE STUDY 2: INVESTIGATING METRIC SPIKE

- WEEKLY USER ENGAGEMENT
- USER GROWTH ANALYSIS
- WEEKLY RETENTION ANALYSIS
- WEEKLY ENGAGEMENT PER DEVICE
- EMAIL ENGAGEMENT ANALYSIS

APPROACH:

1. READ THE PROVIDES DATA DESCRIPTION AND UNDERSTAND THE PROBLEM.
2. CAREFULLY GO THROUGH THE DATA AND MAKE SURE YOU UNDERSTAND THE PROVIDED VARIABLES AND ATTRIBUTE.
3. I IMPORTED FILES BY USING MYSQL WORKBENCH INTO THE OPERATIONAL ANALYTICS DATABASES AND STARTED MAKING QUERIES WITH THE QUESTIONS PRESENTED TO GET THE DESIRED OUTCOMES.
4. I WRITE QUERIES AND EXECUTE THEM IN ORDER TO GET RESULTANT OUTCOME.
5. FINALLY, TAKE SCREENSHOTS AND DOCUMENT THEM TOGETHER.

TECH USE IN THIS PROJECT:



CASE STUDY 1: JOB DATA ANALYSIS

JOBS REVIEWED OVER TIME:

OBJECTIVE: CALCULATE THE NUMBER OF JOBS REVIEWED PER HOUR FOR EACH DAY IN NOVEMBER 2020.

YOUR TASK: WRITE AN SQL QUERY TO CALCULATE THE NUMBER OF JOBS REVIEWED PER HOUR FOR EACH DAY IN NOVEMBER 2020.

```
13 • SELECT count(distinct(job_id))/(30*24)) AS job_reviewed_perhour  
14   From job_data;  
15
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
job_reviewed_perhour			
▶ 6			

THROUGHPUT ANALYSIS:

OBJECTIVE: CALCULATE THE 7-DAY ROLLING AVERAGE OF THROUGHPUT (NUMBER OF EVENTS PER SECOND).

YOUR TASK: WRITE AN 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.

```
1 • select ds , job_nums, total_time,
2     sum(job_nums) over (order by ds rows between 6 preceding and current row)/sum(total_time) as 7day_rolling_average
3 from
4 (
5     select ds ,count(job_id) as job_nums,sum(time_spent) as total_time
6     from job_data where ds >='2020-11-01' and ds <='2020-11-30'
7     group by ds
8 ) a
9 group by
10 ds
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	ds	job_nums	total_time	7day_rolling_average
▶	2020-11-25	1	45	0.0222
	2020-11-26	1	56	0.0357
	2020-11-27	1	104	0.0288
	2020-11-28	2	33	0.1515
	2020-11-29	1	20	0.3000
	2020-11-30	2	40	0.2000

LANGUAGE SHARE ANALYSIS:

OBJECTIVE: CALCULATE THE PERCENTAGE SHARE OF EACH LANGUAGE IN THE LAST 30 DAYS.

YOUR TASK: WRITE AN SQL QUERY TO CALCULATE THE PERCENTAGE SHARE OF EACH LANGUAGE OVER THE LAST 30 DAYS.

```
2 • select language, lang_count,  
3    sum(lang_count)/(sum(lang_count)over(order by language rows between unbounded preceding and unbounded following )) * 100.0 as perc_langu  
4 from  
5 (  
6    select language, count(language) as lang_count  
7    from job_data  
8    group by language  
9 ) a  
10
```

Result Grid | Filter Rows: | Exports | Wrap Cell Contents

	language	lang_count	perc_language
▶	Arabic	1	12.50000
	English	1	12.50000
	French	1	12.50000
	Hindi	1	12.50000
	Italian	1	12.50000
	Persian	3	37.50000

DUPLICATE ROWS DETECTION:

OBJECTIVE: IDENTIFY DUPLICATE ROWS IN THE DATA.

YOUR TASK: WRITE AN SQL QUERY TO DISPLAY DUPLICATE ROWS FROM THE JOB_DATA TABLE.

```
28 • select job_id, count(job_id) as duplicate_count
29   from job_data
30  group by job_id
31  having duplicate_count >1
32
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
job_id	duplicate_count		
23	3		



CASE STUDY 2: INVESTIGATING METRIC SPIKE

WEEKLY USER ENGAGEMENT:

OBJECTIVE: MEASURE THE ACTIVENESS OF USERS ON A WEEKLY BASIS.

YOUR TASK: WRITE AN SQL QUERY TO CALCULATE THE WEEKLY USER ENGAGEMENT.

```
34 • select
35     extract(year FROM occurred_at) as year,
36     extract(week from occurred_at) as weeknum,
37     count(distinct user_id) AS user_engagement
38 FROM
39     events
40 group by year, weeknum
41 order by year, weeknum;
```

Result Grid		Filter Rows:		Export:		W
	year	weeknum	user_engagement			
▶	NULL	NULL	3800			

USER GROWTH ANALYSIS:

OBJECTIVE: ANALYZE THE GROWTH OF USERS OVER TIME FOR A PRODUCT.

YOUR TASK: WRITE AN SQL QUERY TO CALCULATE THE USER GROWTH FOR THE PRODUCT.

```
1 use opr_n_ma ;
2
3 select
4     year,
5     weeknum,
6     new_active_user,
7     sum(new_active_user)over(order by year,weeknum rows between unbounded preceding and current row) as cum_active_user
8 FROM
9 (
10  SELECT
11      extract(year from activated_at) as year,
12      extract(week from activated_at) as weeknum,
13      count(distinct user_id) as new_active_user
14  FROM
15      users
16  where
17      state = 'active'
18  group by
19      year,
20      weeknum
21 ) a
22
```

Result Grid | Filter Rows: | Exports: | Wrap Cell Content: |

	year	weeknum	new_active_user	cum_active_user
▶	2013	0	23	23
	2013	1	30	53
	2013	2	48	101
	2013	3	36	137
	2013	4	30	167

WEEKLY RETENTION ANALYSIS:

OBJECTIVE: ANALYZE THE RETENTION OF USERS ON A WEEKLY BASIS AFTER SIGNING UP FOR A PRODUCT.

YOUR TASK: WRITE AN SQL QUERY TO CALCULATE THE WEEKLY RETENTION OF USERS BASED ON THEIR SIGN-UP COHORT.

```
1 SELECT user_id, count(user_id),
2 sum(case when retention_week = 1 then 1 else 0 end) as week_1,
3 sum(case when retention_week = 2 then 1 else 0 end) as week_2,
4 sum(case when retention_week = 3 then 1 else 0 end) as week_3,
5 sum(case when retention_week = 4 then 1 else 0 end) as week_4,
6 sum(case when retention_week = 5 then 1 else 0 end) as week_5,
7 sum(case when retention_week = 6 then 1 else 0 end) as week_6,
8 sum(case when retention_week = 7 then 1 else 0 end) as week_7,
9 sum(case when retention_week = 8 then 1 else 0 end) as week_8,
10 sum(case when retention_week = 9 then 1 else 0 end) as week_9
11 from
12 (
13 SELECT
14     a.user_id,
15     a.signup_week,
16     b.engagement_week,
17     b.engagement_week - a.signup_week as retention_week
18 FROM
```

Result Grid | Filter Rows: | Exports: | Wrap Cell Content: |

	user_id	count(user_id)	week_1	week_2	week_3	week_4	week_5	week_6	week_7	week_8	week_9
	11920	1	0	0	0	0	0	0	0	0	0
▶	11924	1	0	0	0	0	0	0	0	0	0
	11926	8	1	1	1	1	1	1	1	0	0
	11928	8	0	0	0	0	0	0	0	0	1
	11929	1	0	0	0	0	0	0	0	0	0
	11931	6	1	1	1	1	0	0	0	0	0
	11933	6	1	1	1	1	1	0	0	0	0
	11936	3	0	0	1	0	0	0	0	0	0

WEEKLY ENGAGEMENT PER DEVICE:

OBJECTIVE: MEASURE THE ACTIVENESS OF USERS ON A WEEKLY BASIS PER DEVICE.

YOUR TASK: WRITE AN SQL QUERY TO CALCULATE THE WEEKLY ENGAGEMENT PER DEVICE.

```
1 • select
2     extract(year from occurred_at) as year,
3     extract(week from occurred_at) as week,
4     device,
5     count(distinct user_id) as user_count
6 FROM
7     events
8 where
9     event_type = 'engagement'
10 GROUP BY
11     year, week, device
12 order by
13     year, week, device
```

Result Grid				
Filter Rows:				
Exports: Wrap Cell Contents				
year	week	device	user_count	
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	

EMAIL ENGAGEMENT ANALYSIS:

OBJECTIVE: ANALYZE HOW USERS ARE ENGAGING WITH THE EMAIL SERVICE.

YOUR TASK: WRITE AN SQL QUERY TO CALCULATE THE EMAIL ENGAGEMENT METRICS.

```
1 • select
2     extract(year from occurred_at) as year,
3     extract(week from occurred_at) as week,
4     device,
5     count(distinct user_id) as user_count
6 from
7     events
8 where
9     event_type = 'engagement'
10 group by
11     year, week, device
12 order by
13     year, week, device
```

Result Grid

	year	week	device	user_count
▶	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

The background is a gradient of deep blue and purple, speckled with white dots resembling stars. On the right side, there are faint, light blue geometric patterns: a large circular scale with degree markings (90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 210) and arrows, and a smaller circular diagram with concentric circles and arrows. In the bottom left corner, there are more faint circular and curved line patterns.

THANK YOU