

---

# OPERATION ANALYTICS AND INVESTIGATING METRIC SPIKE

---

---

# AGENDA

---

INTRODUCTION

PROJECT DESCRIPTION

APPROACH

TECH-STACK USED

INSIGHTS

RESULT

CONCLUSION OF THE ANALYSIS

# 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 in which it must improve upon. I've worked with the ops team, support team, marketing team, etc. and helped them derive insights from the data they collect.
- Being one of the most essential 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 spikes is also an essential part of operation analytics as 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 plunge? etc. Questions like these must be answered daily and for that, it's very important to investigate metric spikes.

# APPROACH

**The project was executed in a systematic manner following standard data analysis protocols:**

- Designed and Implemented Database Architecture: Designed and created a database, including defining table structures and establishing table relationships, to store and organize provided data.
- Performed Data Analysis: Utilized SQL queries to thoroughly analyze data, answering key business questions. Joined relevant data tables, aggregated metrics, and filtered data to extract actionable insights from the dataset.
- Delivered Data-Driven Recommendations: Performed exploratory analysis to uncover trends and patterns. Presented analytical findings and data visualizations to provide business intelligence and evidence-based guidance to inform strategic decision-making.

---

# TECH-STACK USED

---

- MySQL Workbench (v8.0.33) enabled visual database management. SQL provided data query and analysis capabilities.
- MySQL Workbench allowed efficient import and inspection of the Instagram dataset.
- SQL was used to write optimized queries that extracted user metrics and insights from the database.
- These technologies were selected due to their widespread use in data analytics. MySQL is a leading relational database. SQL is the standard language for data manipulation and analysis.
- Together, MySQL Workbench and SQL provided the necessary data storage, querying, and analysis functionality required for this project in a reliable and scalable manner.

---

# **INSIGHTS**

---

## **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

# INSIGHTS

## CASE STUDY 1: JOB DATA ANALYSIS:

>\_ 1. Jobs Reviewed Over Time:  
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;
```

>\_ the number of jobs reviewed per hour for  
each day in November 2020.

	per_day_jobs
▶	0.0083

# INSIGHTS

>\_ 2. Throughput Analysis: Calculate the 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
) sub
ORDER BY ds;
```

>\_ 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.

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

# INSIGHTS

>\_ 3. Language Share Analysis: Calculate the percentage share of each language in the last 30 days.

```
WITH LanguageCounts AS (
  SELECT
    language,
    COUNT(language) AS total_language,
    SUM(COUNT(*)) OVER () AS total_count
  FROM job_data
  GROUP BY language
)
SELECT
  language,
  total_language,
  (total_language * 100.0) / total_count AS
  percentage
FROM LanguageCounts
ORDER BY language;
```

>\_ Persian Language had the highest share among other languages :

	language	total_language	percentage
▶	Arabic	1	12.50000
	English	1	12.50000
	French	1	12.50000
	Hindi	1	12.50000
	Italian	1	12.50000
	Persian	3	37.50000

# INSIGHTS

>\_ 4. Duplicate Rows Detection: Identify duplicate rows in the data.

```
WITH cte AS (
    SELECT *,
        ROW_NUMBER() OVER (PARTITION BY
job_id ORDER BY (SELECT 0)) AS row_num
    FROM job_data
)
SELECT *
FROM cte
WHERE row_num > 1;
```

>\_ There were two duplicate job id in the dataset :

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

# INSIGHTS

## Case Study 2: Investigating Metric Spike

>\_ 1. Weekly User Engagement: Measure the activeness of users on a weekly basis.

```
SELECT
    WEEK(STR_TO_DATE(occurred_at, '%d-%m-
%Y %H:%i')) AS week_number,
    COUNT(DISTINCT user_id) AS active_users
FROM
    events
GROUP BY
    week_number
ORDER BY
    week_number;
```

>\_ Week 30 posted the highest user engagement and week 35 posted the minimum user engagement:

	week_number	active_users
▶	17	129
	18	264
	19	279
	20	267
	21	282
	22	318
	23	298
	24	340
	25	324
	26	298
	27	327
	28	323
	29	327
	30	357
	31	299
	32	325
	33	339
	34	346
	35	21



# INSIGHTS

>\_ 2. User Growth Analysis: Analyze the growth of users over time for a product

```

SELECT
    YEAR(STR_TO_DATE(created_at, '%d-%m-%Y'))
AS year,
    MONTH(STR_TO_DATE(created_at, '%d-%m-
%Y')) AS month_number,
    COUNT(DISTINCT user_id) AS new_users
FROM
    users
GROUP BY
    year, month_number
ORDER BY
    year, month_number;
  
```

>\_ 2014 8TH WEEK HAS THE HIGH USER GROWTH :

	year	month_number	new_users
▶	2013	1	160
	2013	2	160
	2013	3	150
	2013	4	181
	2013	5	214
	2013	6	213
	2013	7	284
	2013	8	316
	2013	9	330
	2013	10	390
	2013	11	399
	2013	12	486
	2014	1	552
	2014	2	525
	2014	3	615
	2014	4	726
	2014	5	779
	2014	6	873
	2014	7	997
	2014	8	1031

# INSIGHTS

>\_ 3. Weekly Retention Analysis: Analyze the retention of users on a weekly basis after signing up for a product.

```
WITH cte1 AS (
  SELECT DISTINCT
    user_id,
    EXTRACT(WEEK FROM occurred_at) AS signup_week
  FROM
    events
  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
  WHERE
    event_type = 'engagement'
)
```

>\_ 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:

total_engaged_users	retained_users
317	236

# INSIGHTS

>\_ 3. Weekly Retention Analysis: Analyze the retention of users on a weekly basis after signing up for a product.

```
SELECT
  COUNT(user_id) AS 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
    ctel a
  LEFT JOIN
    cte2 b ON a.user_id = b.user_id
  ORDER BY
    a.user_id
) sub;
```

>\_ 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:

total_engaged_users	retained_users
317	236

# INSIGHTS

>\_ 4. Weekly Engagement Per Device: Measure the activeness of users on a weekly basis per device.

```
WITH cte AS (
  SELECT
    EXTRACT(YEAR FROM occurred_at) || '-' || device AS
    weeknum_device,
    COUNT(DISTINCT user_id) AS usercnt
  FROM
    events_tbl
  WHERE
    event_type = 'engagement'
  GROUP BY
    weeknum_device
  ORDER BY
    weeknum_device
)
SELECT
  weeknum_device,
  usercnt
FROM
  cte;
```

>\_ 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

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

# INSIGHTS

>\_ 5. Email Engagement Analysis: Analyze how users are engaging with the email service.

```
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 and email click rates

	email_open_rate	email_click_rate
▶	30.7087	9.8425

---

# RESULT

---

- Demonstrated proficiency in writing complex SQL queries across multiple tables to extract user data and metrics.
- Demonstrated advanced SQL proficiency through operational analytics.
- Calculated hourly job reviews and 7-day throughput to identify workflow optimization opportunities.
- Analyzed language share to expand job content reach.
- Diagnosed user engagement spike using cohort and device analysis. Linked issue to email effectiveness.
- Delivered actionable insights across operations, product and marketing to inform strategy.
- Developed skills in analyzing real-world production datasets at scale.
- Created an analytical methodology replicable across other business datasets.
- Showcased and strengthened core data analytics competencies in SQL and critical thinking.

# CONCLUSION OF THE ANALYSIS

The operational analytics project provided an opportunity to apply advanced SQL skills to uncover valuable insights for business optimization. Thorough analysis of job review data revealed workflow inefficiencies and data quality issues. Investigating the user engagement spike with retention and cohort analysis pointed to deficiencies in the email product as a likely cause.

- Advanced SQL enables granular analysis using techniques like aggregates, joins, window functions.
- Operational metrics should be monitored over time to detect abnormalities. Rolling averages smooth daily fluctuations.
- Noisy, duplicate data compromises analysis and should be addressed through data cleaning.
- Cohort analysis is pivotal for diagnosing issues exhibited by subsets of users.
- Visualizations and dashboards aid data-driven decision making when shared with stakeholders.

This project exemplified how leveraging data analytics can help diagnose issues, identify improvement opportunities, and empower organizations to enhance processes, products, and performance. The skills applied here will be invaluable for any analytical role.

---

# THANK YOU

---

PRESENTED BY: CIRIL SHAJU