# Project 03

# Operation Analytics and Investigating Metric Spike

## Project Description:

Operational Analytics is a crucial process that involves analysing a company's end-to-end operations. This analysis helps identify areas for improvement within the company. As a Data Analyst, you'll work closely with various teams, such as operations, support, and marketing, helping them derive valuable insights from the data they collect.

One of the key aspects of Operational Analytics is investigating metric spikes. This involves understanding and explaining sudden changes in key metrics, such as a dip in daily user engagement or a drop in sales. As a Data Analyst, you'll need to answer these questions daily, making it crucial to understand how to investigate these metric spikes.

In this project, you'll take on the role of a Lead Data Analyst at a company like Microsoft. You'll be provided with various datasets and tables, and your task will be to derive insights from this data to answer questions posed by different departments within the company. Your goal is to use your advanced SQL skills to analyse the data and provide valuable insights that can help improve the company's operations and understand sudden changes in key metrics.

### Case Study 1: Job Data Analysis

**Table: job_data**

- **job_id:** Unique identifier of jobs
- **actor_id:** Unique identifier of actor
- **event:** The type of event (decision/skip/transfer).
- **language:** The Language of the content
- **time_spent:** Time spent to review the job in seconds.
- **org:** The Organization of the actor
- **ds:** The date in the format yyyy/mm/dd (stored as text).

**Tasks:**

A. **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.

B. **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.

C. **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.

D. **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.

### Case Study 2: Investigating Metric Spike

**You will be working with three tables:**

- **users**: Contains one row per user, with descriptive information about that user’s account.
- **events**: Contains one row per event, where an event is an action that a user has taken (e.g., login, messaging, search).
- **email_events**: Contains events specific to the sending of emails.

**Tasks:**

A. **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.

B. **User Growth Analysis:**
   - Objective: Analyse the growth of users over time for a product.
   - Your Task: Write an SQL query to calculate the user growth for the product.

C. **Weekly Retention Analysis:**
   - Objective: Analyse 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.

D. **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.

E. **Email Engagement Analysis:**
   - Objective: Analyse how users are engaging with the email service.
   - Your Task: Write an SQL query to calculate the email engagement metrics.

### Approach

I have used relational database software to inspect and answer queries I was asked, business analytics tool for visualization of the insights, gathered the information and loopholes and jotted them down.

### Tech-Stack Used

- My SQL version 8.0 was used in this project.

- The MySQL workbench is used to write and execute queries.

- The entire dataset is separated by two different approaches, one is provided to analyse the end to end operation of the organization, case study1 and the other one is provided with information to investigate the metric spike, case study-2.

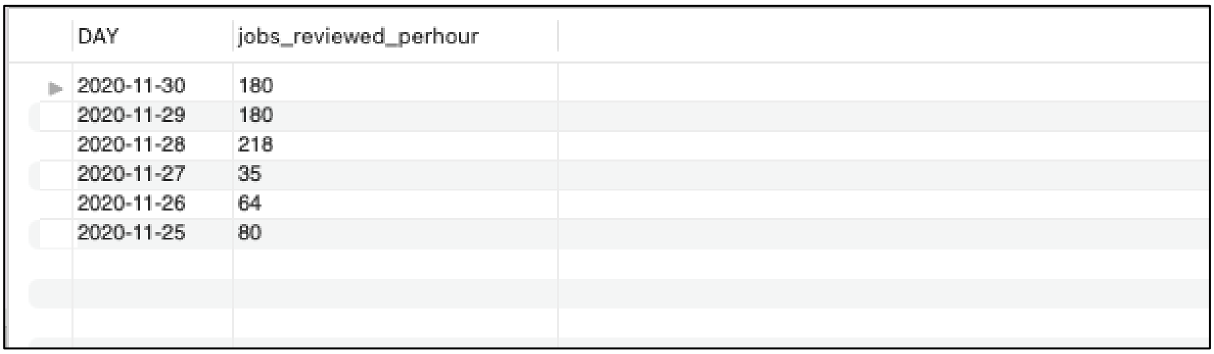
- A number of SQL functions and queries are used in this project.

**Case Study 1: Job Data Analysis**

## SQL Tasks:

A. **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.

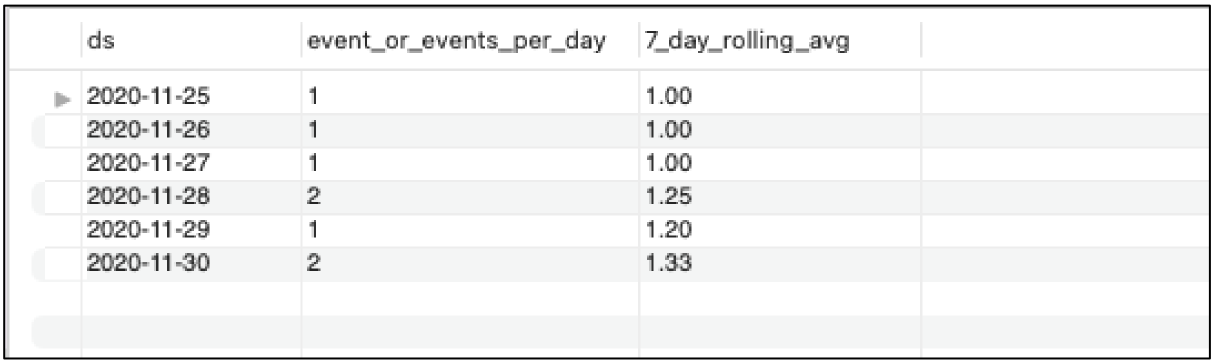
```
SELECT ds AS DAY,ROUND(COUNT(job_id)/SUM(time_spent)*3600) as
jobs_reviewed_perhour

FROM job_data

WHERE ds BETWEEN '2020-11-01' AND '2020-11-30'

GROUP BY ds;
```



| DAY | jobs_reviewed_perhour |
|---|---|
| 2020-11-30 | 180 |
| 2020-11-29 | 180 |
| 2020-11-28 | 218 |
| 2020-11-27 | 35 |
| 2020-11-26 | 64 |
| 2020-11-25 | 80 |

B. **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.

```
SELECT ds, event_or_events_per_day,

ROUND(AVG(event_or_events_per_day) OVER(ORDER BY ds ROWS BETWEEN 6 PRECEDING AND CURRENT ROW),2) AS 7_day_rolling_avg

FROM (SELECT ds, COUNT(DISTINCT event) AS event_or_events_per_day

FROM job_data

GROUP BY ds) AS temptable;
```

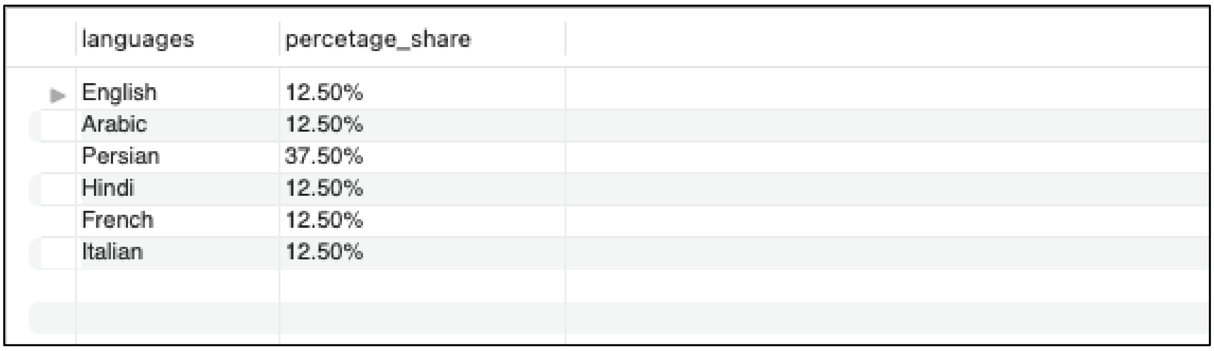


| ds | event_or_events_per_day | 7_day_rolling_avg |
|---|---|---|
| 2020-11-25 | 1 | 1.00 |
| 2020-11-26 | 1 | 1.00 |
| 2020-11-27 | 1 | 1.00 |
| 2020-11-28 | 2 | 1.25 |
| 2020-11-29 | 1 | 1.20 |
| 2020-11-30 | 2 | 1.33 |

C. **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

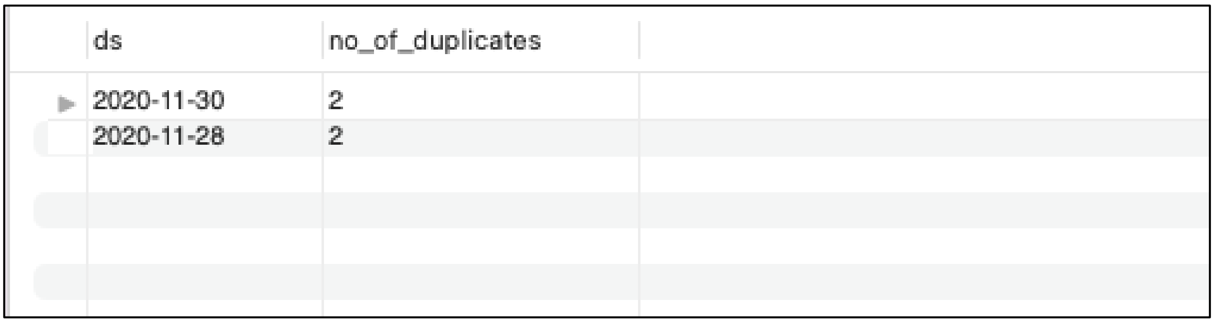
```
SELECT language AS languages, CONCAT(ROUND(COUNT(*)*100/(select COUNT(*)

FROM job_data),2),'%') AS percentage_share

FROM job_data

GROUP BY language;
```



| languages | percetage_share |
|---|---|
| English | 12.50% |
| Arabic | 12.50% |
| Persian | 37.50% |
| Hindi | 12.50% |
| French | 12.50% |
| Italian | 12.50% |

D. **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.

```
SELECT ds, COUNT(ds) AS no_of_duplicates

FROM job_data

GROUP BY ds

HAVING no_of_duplicates > 1;
```



| ds | no_of_duplicates |
|---|---|
| 2020-11-30 | 2 |
| 2020-11-28 | 2 |

### Case Study 2: Investigating Metric Spike

A. **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.

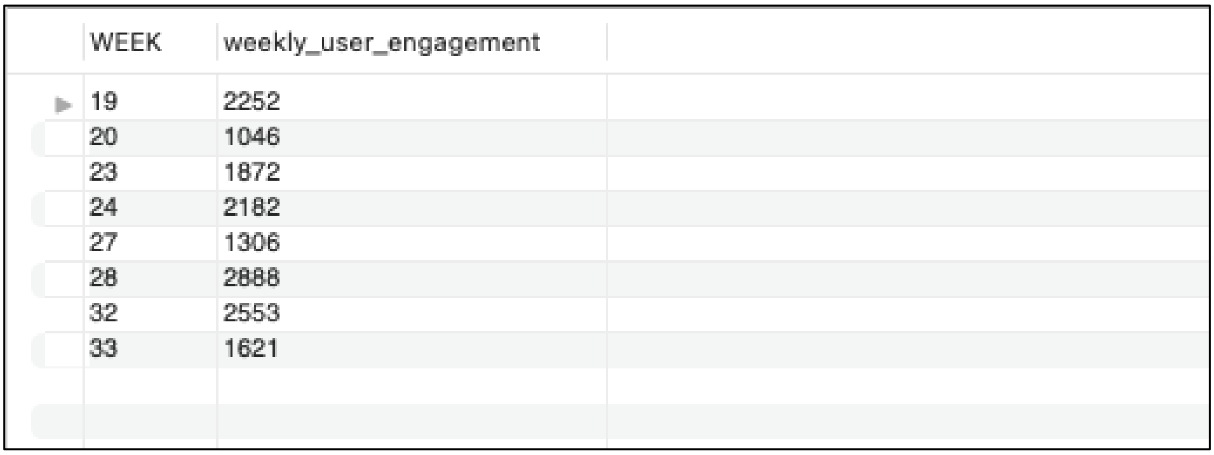
SELECT WEEK(occurred_at) AS WEEK,COUNT(DISTINCT user_id) AS weekly_user_engagement

FROM events

WHERE event_type='engagement'

GROUP BY WEEK(occurred_at)

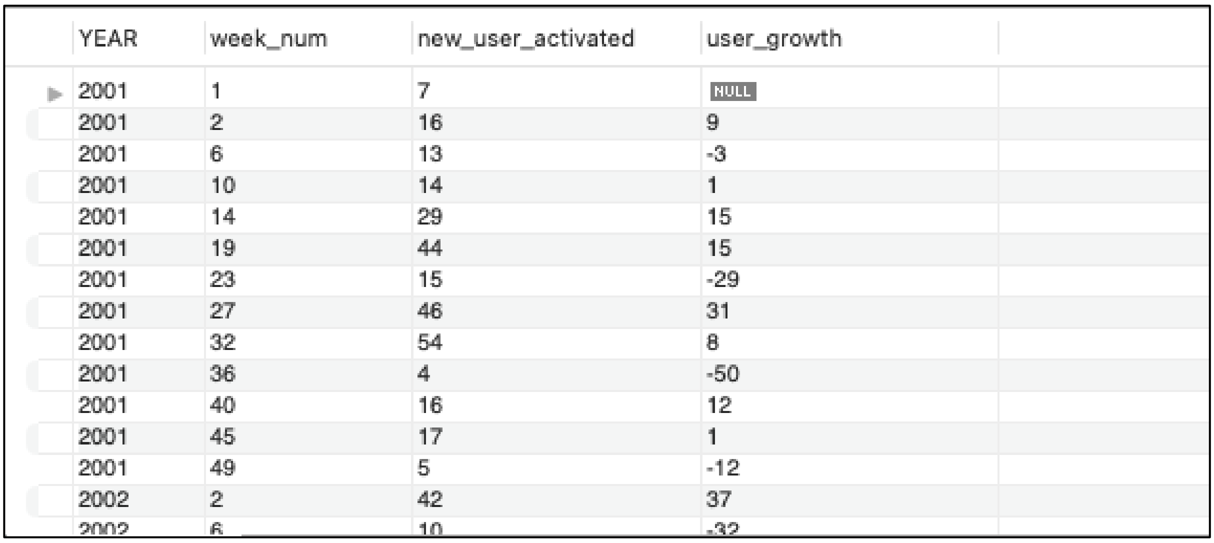
ORDER BY WEEK(occurred_at);



| WEEK | weekly_user_engagement |
|---|---|
| 19 | 2252 |
| 20 | 1046 |
| 23 | 1872 |
| 24 | 2182 |
| 27 | 1306 |
| 28 | 2888 |
| 32 | 2553 |
| 33 | 1621 |

B. **User Growth Analysis:**

- Objective: Analyse the growth of users over time for a product.
- Your Task: Write an SQL query to calculate the user growth for the product.

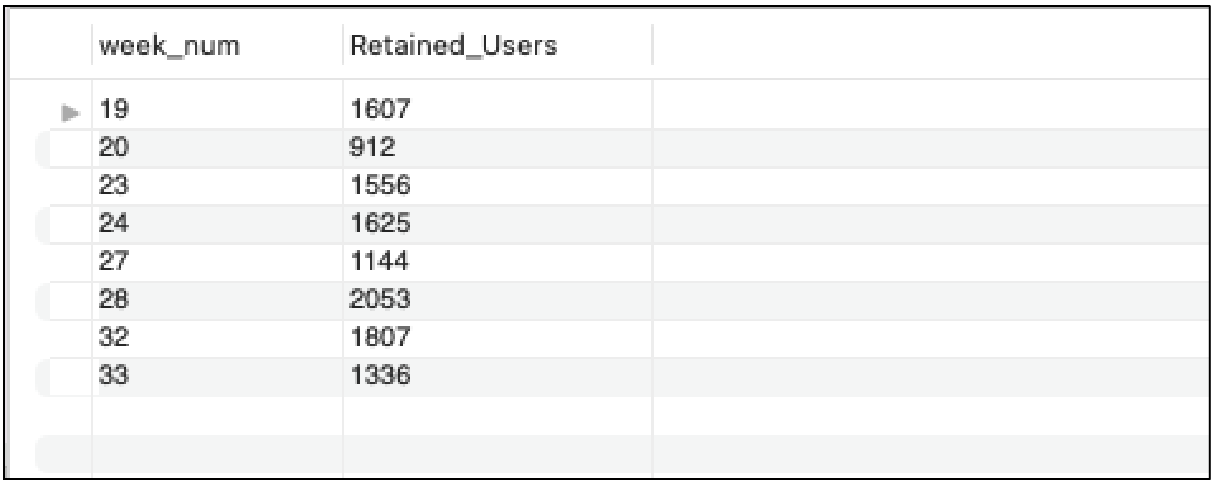


| YEAR | week_num | new_user_activated | user_growth |
|---|---|---|---|
| 2001 | 1 | 7 | NULL |
| 2001 | 2 | 16 | 9 |
| 2001 | 6 | 13 | -3 |
| 2001 | 10 | 14 | 1 |
| 2001 | 14 | 29 | 15 |
| 2001 | 19 | 44 | 15 |
| 2001 | 23 | 15 | -29 |
| 2001 | 27 | 46 | 31 |
| 2001 | 32 | 54 | 8 |
| 2001 | 36 | 4 | -50 |
| 2001 | 40 | 16 | 12 |
| 2001 | 45 | 17 | 1 |
| 2001 | 49 | 5 | -12 |
| 2002 | 2 | 42 | 37 |
| 2002 | 6 | 10 | -32 |

C. **Weekly Retention Analysis:**

- Objective: Analyse 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

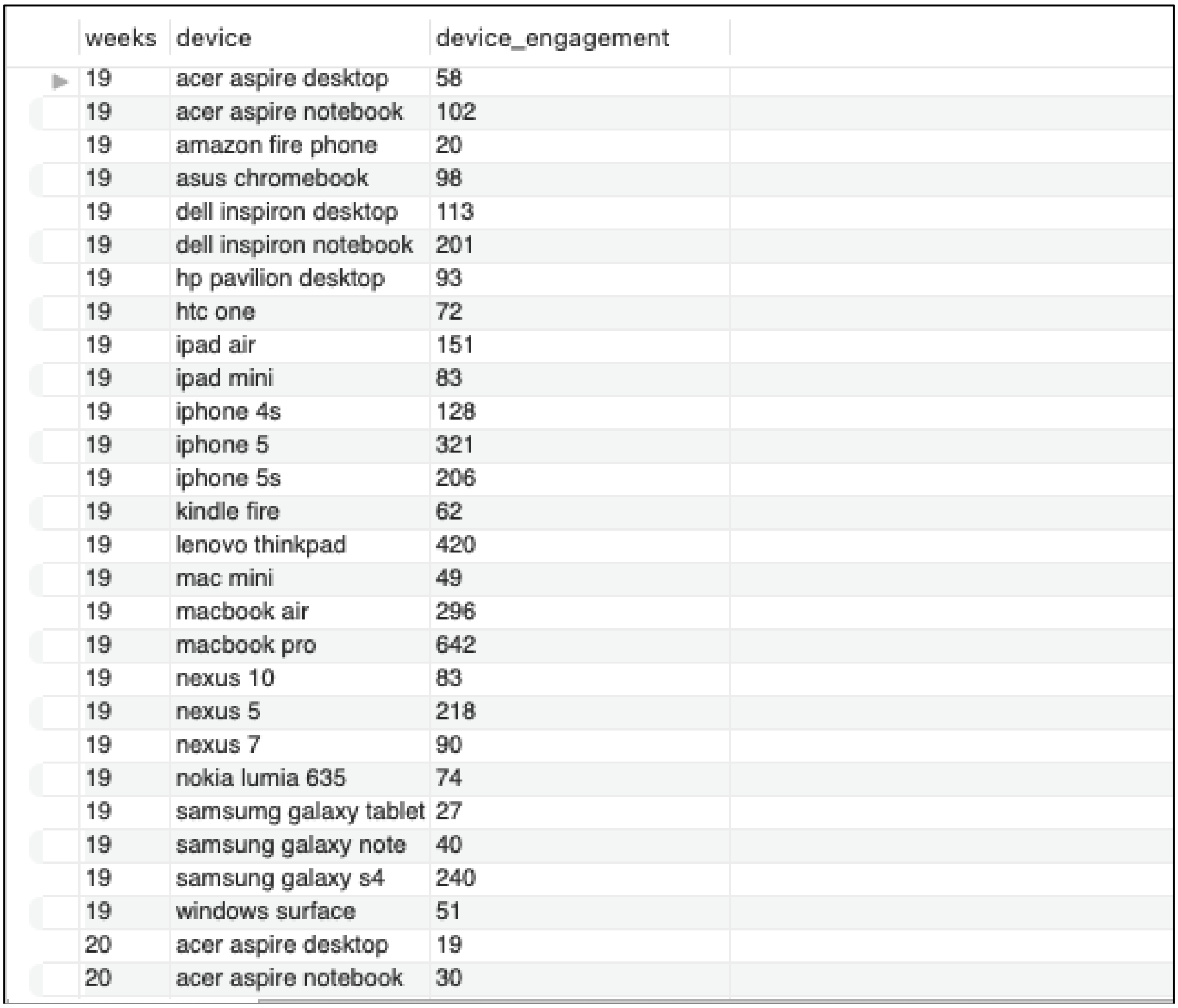
```
SELECT t1.week_num,(t2.old_users - t1.new_users)AS Retained_Users

FROM(SELECT WEEK(occurred_at) AS week_num,

COUNT(DISTINCT user_id) AS new_users

FROM events

WHERE event_type = "signup_flow"

GROUP BY week_num) AS t1

JOIN

(SELECT WEEK(occurred_at) AS week_num,

COUNT(DISTINCT user_id) AS old_users

FROM events

WHERE event_type = "engagement"

GROUP BY week_num) AS t2

ON t1.week_num = t2.week_num;
```



| week_num | Retained_Users |
|---|---|
| 19 | 1607 |
| 20 | 912 |
| 23 | 1556 |
| 24 | 1625 |
| 27 | 1144 |
| 28 | 2053 |
| 32 | 1807 |
| 33 | 1336 |

D. **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.

```
SELECT WEEK(occurred_at) AS weeks,device,COUNT(DISTINCT user_id) AS

device_engagement

FROM events

GROUP BY device, WEEK(occurred_at)

ORDER BY WEEK(occurred_at);
```

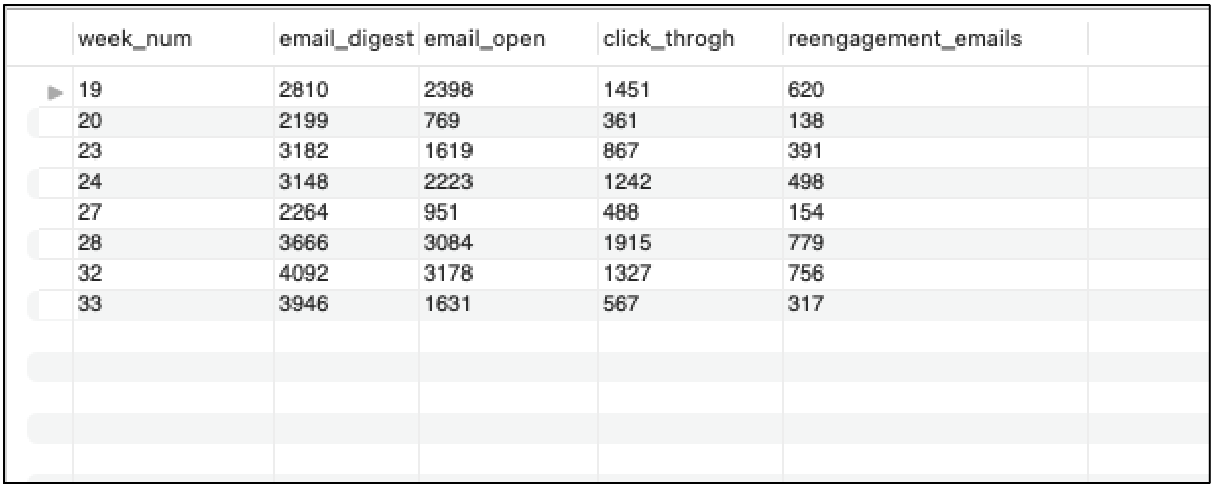


| weeks | device | device_engagement |
|---|---|---|
| 19 | acer aspire desktop | 58 |
| 19 | acer aspire notebook | 102 |
| 19 | amazon fire phone | 20 |
| 19 | asus chromebook | 98 |
| 19 | dell inspiron desktop | 113 |
| 19 | dell inspiron notebook | 201 |
| 19 | hp pavilion desktop | 93 |
| 19 | htc one | 72 |
| 19 | ipad air | 151 |
| 19 | ipad mini | 83 |
| 19 | iphone 4s | 128 |
| 19 | iphone 5 | 321 |
| 19 | iphone 5s | 206 |
| 19 | kindle fire | 62 |
| 19 | lenovo thinkpad | 420 |
| 19 | mac mini | 49 |
| 19 | macbook air | 296 |
| 19 | macbook pro | 642 |
| 19 | nexus 10 | 83 |
| 19 | nexus 5 | 218 |
| 19 | nexus 7 | 90 |
| 19 | nokia lumia 635 | 74 |
| 19 | samsumg galaxy tablet | 27 |
| 19 | samsung galaxy note | 40 |
| 19 | samsung galaxy s4 | 240 |
| 19 | windows surface | 51 |
| 20 | acer aspire desktop | 19 |
| 20 | acer aspire notebook | 30 |

E. **Email Engagement Analysis:**

- Objective: Analyse how users are engaging with the email service.
- Your Task: Write an SQL query to calculate the email engagement metrics.

```
SELECT DISTINCT WEEK(occured_at) AS week_num,

COUNT(DISTINCT CASE WHEN ACTION = 'sent_weekly_digest' THEN user_id END) AS
email_digest,

COUNT(DISTINCT CASE WHEN ACTION ='email_open' THEN user_id END) AS email_open,

COUNT(DISTINCT CASE WHEN ACTION = 'email_clickthrough' THEN user_id END) AS
click_throgh,
COUNT(DISTINCT CASE WHEN ACTION ='sent_reengagement_email' THEN user_id END) AS
reengagement_emails

FROM email_events

GROUP BY WEEK(occured_at);
```



| week_num | email_digest | email_open | click_throgh | reengagement_emails |
|---|---|---|---|---|
| 19 | 2810 | 2398 | 1451 | 620 |
| 20 | 2199 | 769 | 361 | 138 |
| 23 | 3182 | 1619 | 867 | 391 |
| 24 | 3148 | 2223 | 1242 | 498 |
| 27 | 2264 | 951 | 488 | 154 |
| 28 | 3666 | 3084 | 1915 | 779 |
| 32 | 4092 | 3178 | 1327 | 756 |
| 33 | 3946 | 1631 | 567 | 317 |

**Insights**

**Case Study 1: Job Data Analysis**

- In the given date range, the highest number of jobs were reviewed on 28th November,2020 which is 218.

- It seems that the difference between events happening each day and the throughput is not so big so we can prefer 7 day rolling over daily metrics.

- 37.5% of the language share is taken by Persian language which is the highest in the distribution.

- 28th and 30th November,2020 has two duplicate rows in the dataset.

**Case Study 2: Investigating Metric Spike**

- 30th week holds the highest user engagement,1467.
- In 2013,highest user growth was in 42th week and highest number of total new users were seen in 50th week whereas the same calculation comes for 32nd and 34th week in 2014,there is sharp drop in user growth on 20th week(2013) and 35th week(2014).
- With some small decrease, the overall number of retained users gradually increased till 30th week and then started decreasing, the last week (i.e : 35th week) has seen the lowest number of retained users, from 34th week to 35th week, there is a sharp drop.
- Samsung galaxy is the most used device and on 29th week it has seen highest use.
- Most number of email engagements were acted upon email weekly digest that holds almost 64% of total email engagements.

**Result**

- Data cleaning: this entire project has helped me learning the approaches to convert raw data into clean data.
- Practical knowledge: experienced a practical exposure with different SQL commands and their uses in real life industries..
- Uploading Large dataset: Learned the tricks and methods to upload large datasets into MySQL.
- Business optimization methods: gained an idea on how the industries and organizations optimize their business problems, calculate the metric spike and perform operation analytics that helps in the growth of an industry or organization.