# NAME : MUSKAN PRITAM

# PROJECT 3

# Operation Analytics and Investigating Metric Spike

##### The day to day life of data analytics includes cleaning of data to make if more efficient to use and working on it to perform analysis and work according to it.

#### Similarly in this project we were given with a data set and was asked to work with it along with few questions asked , for which we should find the most appropriate answers using SQL and MySql workbench . Also the  insights will help the product manager and the rest of the team make informed decisions about the future direction.

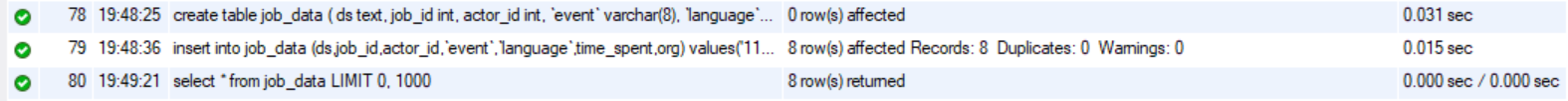
###### **Case Study 1: Job Data Analysis**

****You will be working with a table named**job\_data**with the following columns:****

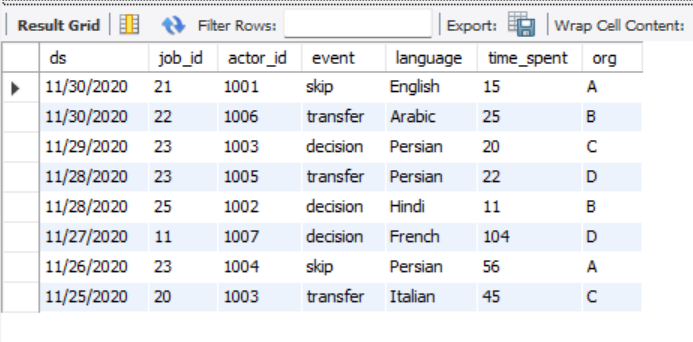
* ****job\_id:****Unique identifier of jobs
* ****actor\_id:****Unique identifier of actor
* ****event:****The type of event (decision/skip/transfer).
* ****language:****The Language of the content
* ****time\_spent:****Time spent to review the job in seconds.
* ****org:****The Organization of the actor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

Here we are creating a table named job\_data and also importing data into it.

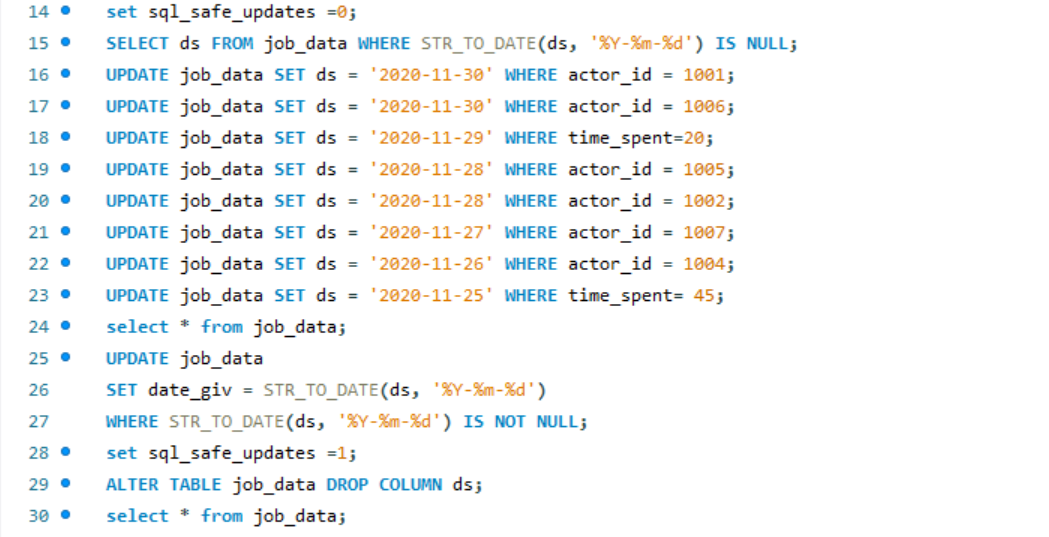




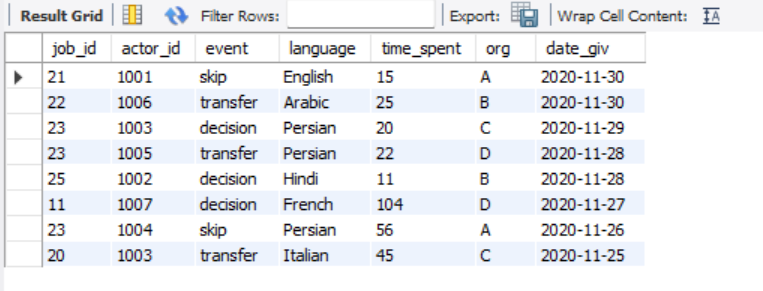
This is the original table data:



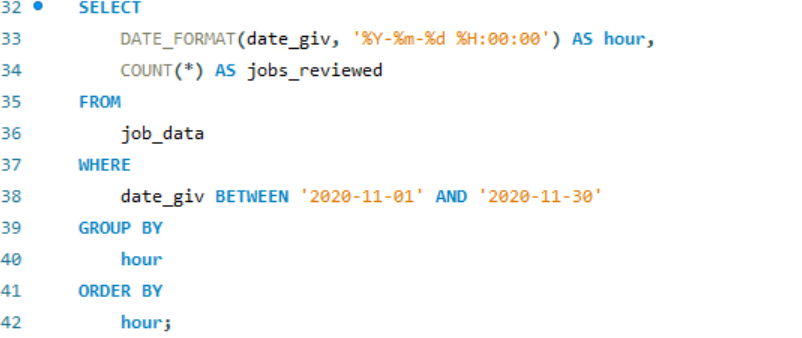
HERE, we are converting the DS column in the above original table which is in the data type of text into another column named DATE\_GIV of datatype date. Adding date\_giv column into the original table and deleting the previous ds one.



The table data after all these modifications:



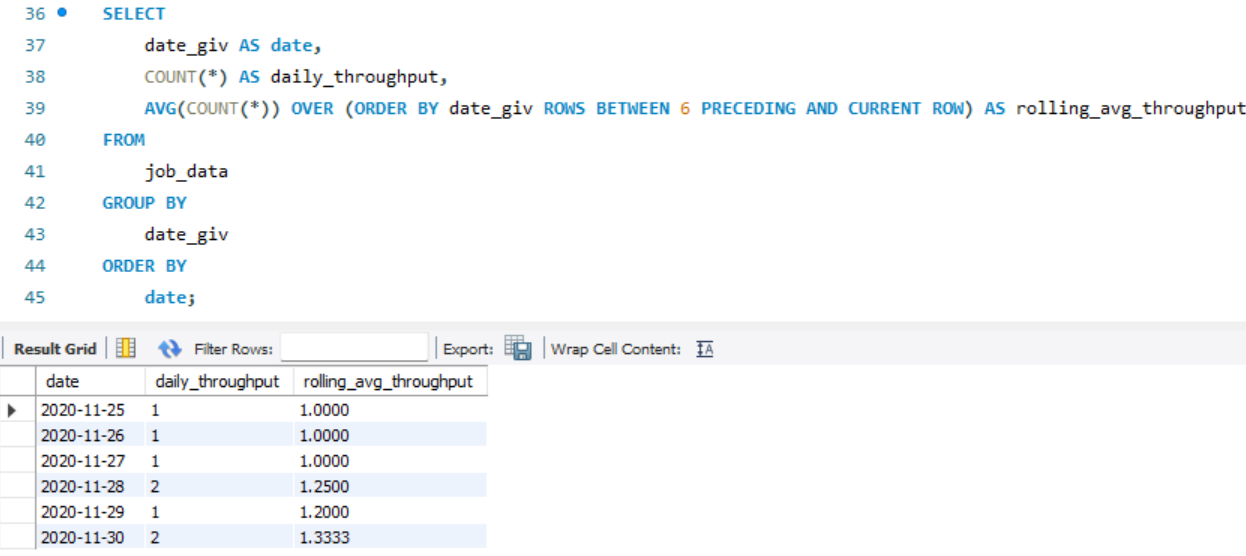
1. ****Jobs Reviewed Over Time:****
   1. Objective: Calculate the number of jobs reviewed per hour for each day in November 2020.
   2. Your Task: Write an SQL query to calculate the number of jobs reviewed per hour for each day in November 2020.





**Explaination**: in this the output shows the count of jobs that are reviewed on the following dates.

1. ****Throughput Analysis:****
   1. Objective: Calculate the 7-day rolling average of throughput (number of events per second).
   2. 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.

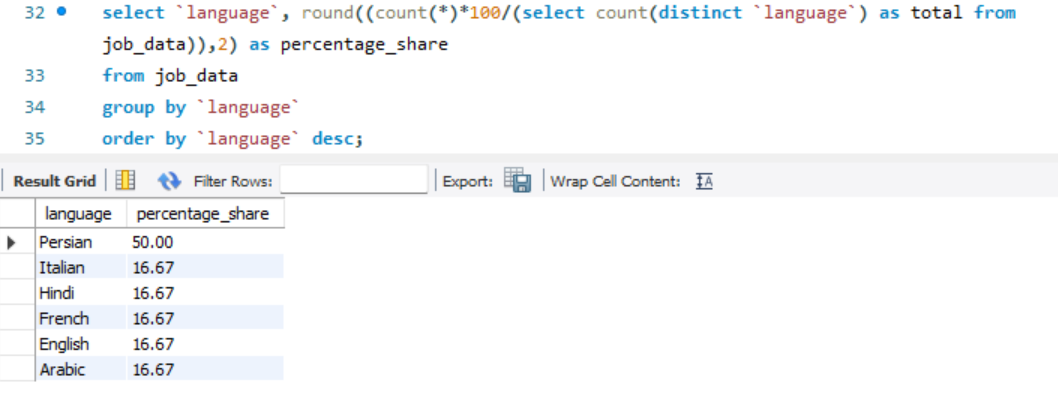


**Explaination**: the output shows the daily throughput also the rolling average throughput of the table data of job\_data. Throughput means number of events per second.

Also the choice depends on whether you need real-time monitoring (daily) or trend analysis (7-day average).

****3.Language Share Analysis:****

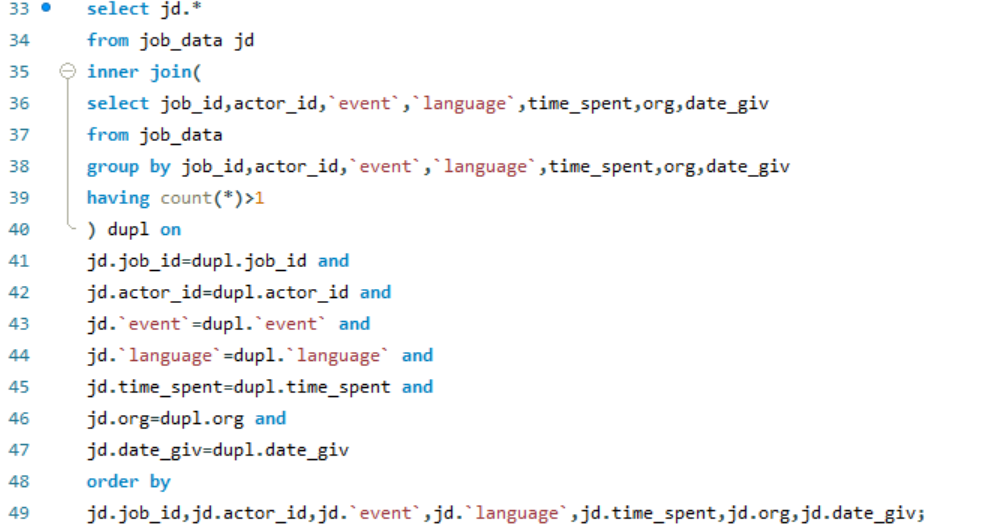
* 1. Objective: Calculate the percentage share of each language in the last 30 days.
  2. Your Task: Write an SQL query to calculate the percentage share of each language over the last 30 days.

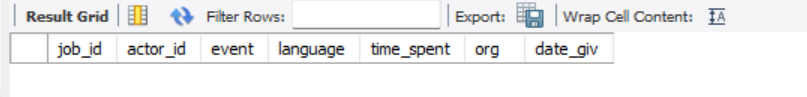


**Explaination**: the output shows the percentage of time each of these languages were used over the last 30 days with each of their names on the side. Persian being the most popular here.

****4.Duplicate Rows Detection:****

* 1. Objective: Identify duplicate rows in the data.
  2. Your Task: Write an SQL query to display duplicate rows from the job\_data table.





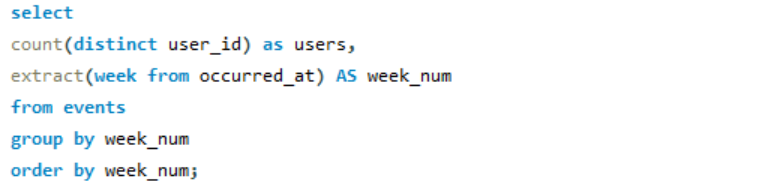
**Explaination**: here there are no duplicate rows as no two rows have the exact same values. We considered duplicates to be shown for the entire data of the rows considering all the columns and not only just few columns.

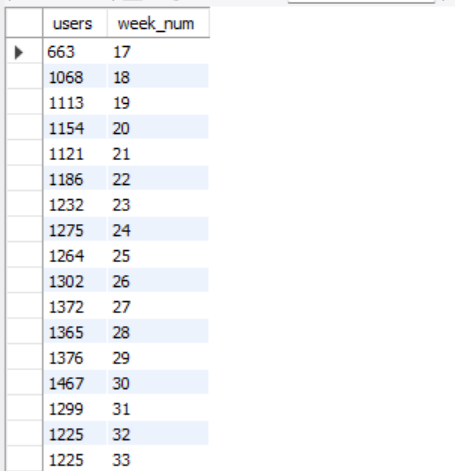
###### **Case Study 2: Investigating Metric Spike**

****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.

1. ****Weekly User Engagement:****
   1. Objective: Measure the activeness of users on a weekly basis.
   2. Your Task: Write an SQL query to calculate the weekly user engagement.

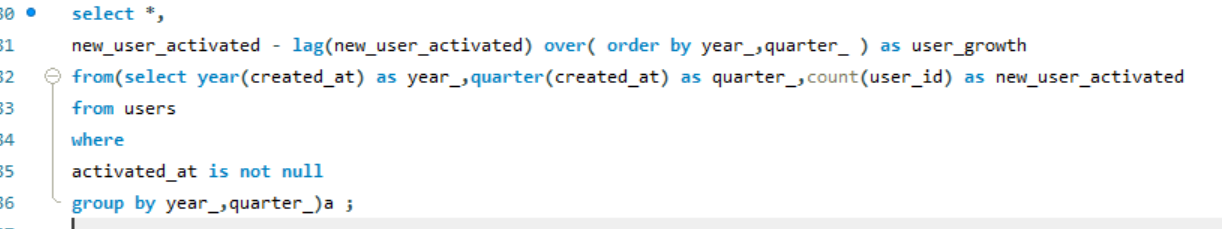


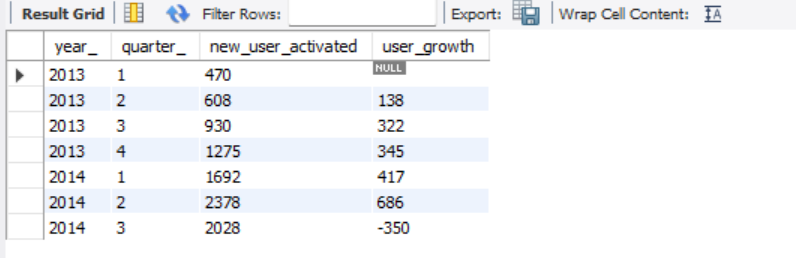




**Explaination**: this table shows the data where it shows the week numbers when the users were most active. This shows the weekly engagement of users on his/her account sending and receiving mails.

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

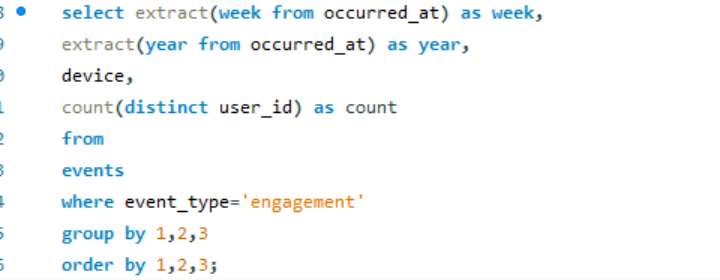




**Explaination**: this table shows the growth of the users on the product. When there is growth the user\_growth column is positive, when there is no growth the user\_growth column value is negative. From this table we can easily see which year resulted in how much growth.

****3.Weekly Engagement Per Device:****

* 1. Objective: Measure the activeness of users on a weekly basis per device.
  2. Your Task: Write an SQL query to calculate the weekly engagement per device.







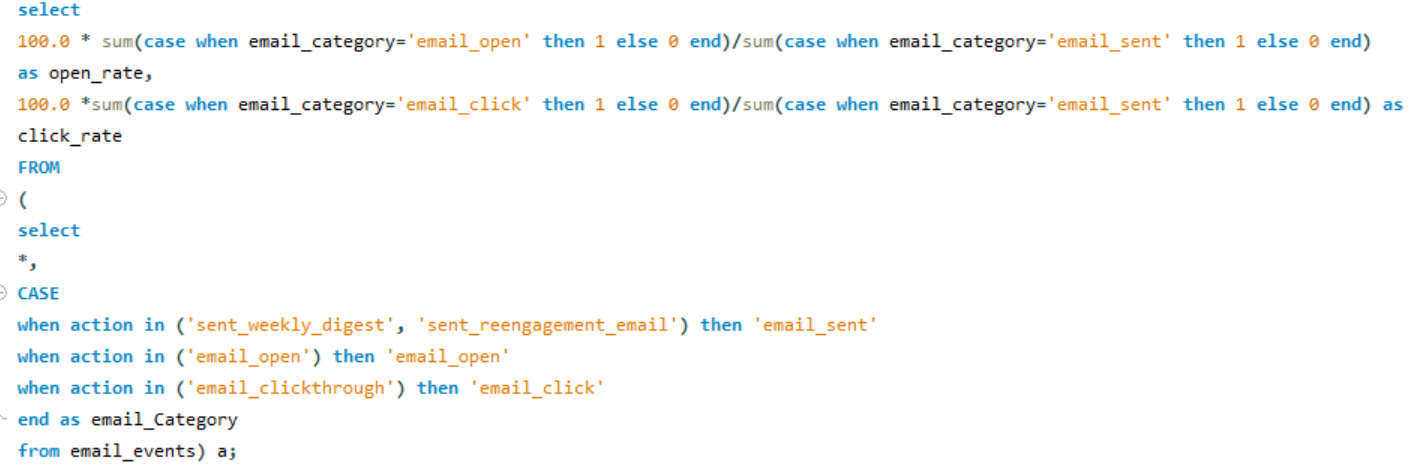


So on….

**Explaination**: the table was too long to fit in, thus only few screenshots have been added. Here in this query we saw the engagement on weekly basis according to different different devices.

****4.Email Engagement Analysis:****

* 1. Objective: Analyze how users are engaging with the email service.
  2. Your Task: Write an SQL query to calculate the email engagement metrics.





**Explaination**: this shows the type in which the users are engaging in the email services, based on email\_open, email\_sent and email\_click. We got the rates of open and click as output.

# !!THANK YOU!!