Operation Analytics and Investigating Metric Spike

Description:

Operational Analytics is a crucial process that involves analyzing 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.

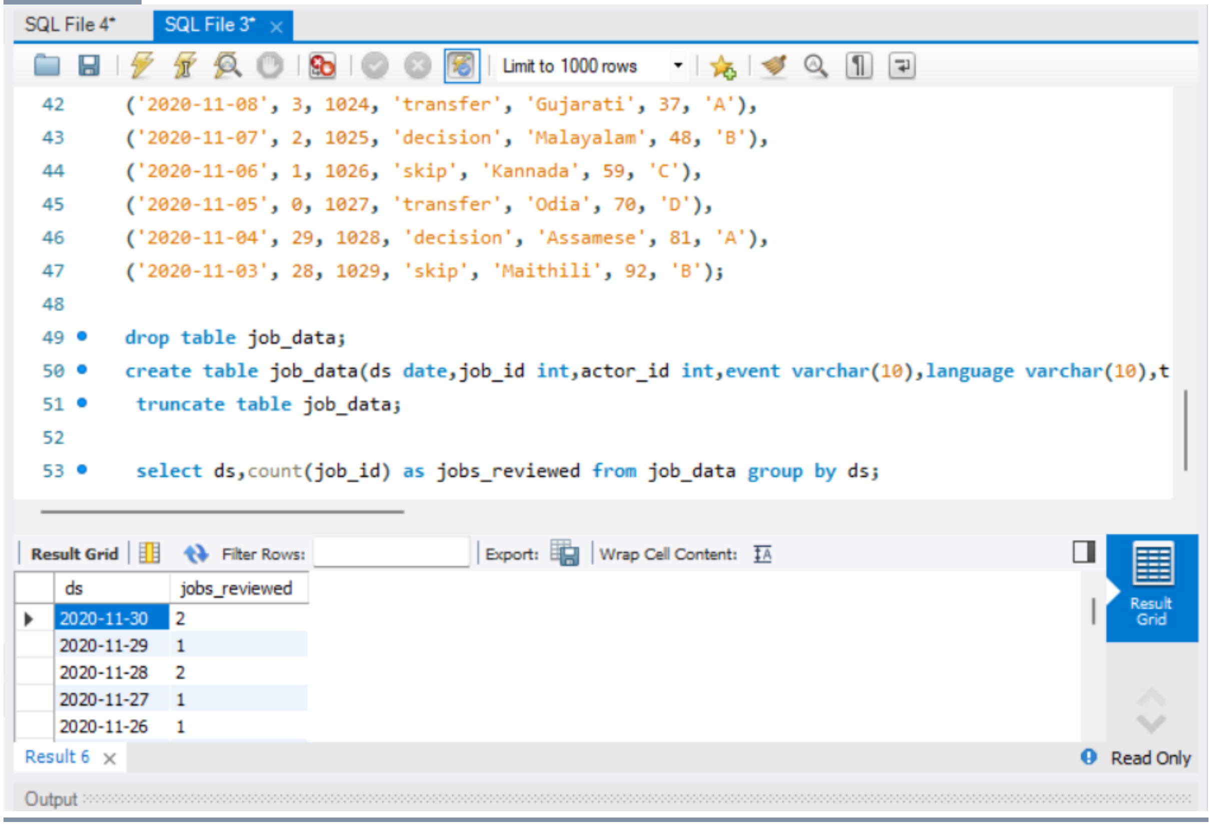
In this project, you'll take on the role of a Lead Data Analyst at a company like Microsoft. You'll be provided with various datasets and tables, and your task will be to derive insights from this data to answer questions posed by different departments within the company. Your goal is to use your advanced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

Tasks:

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.

Query:



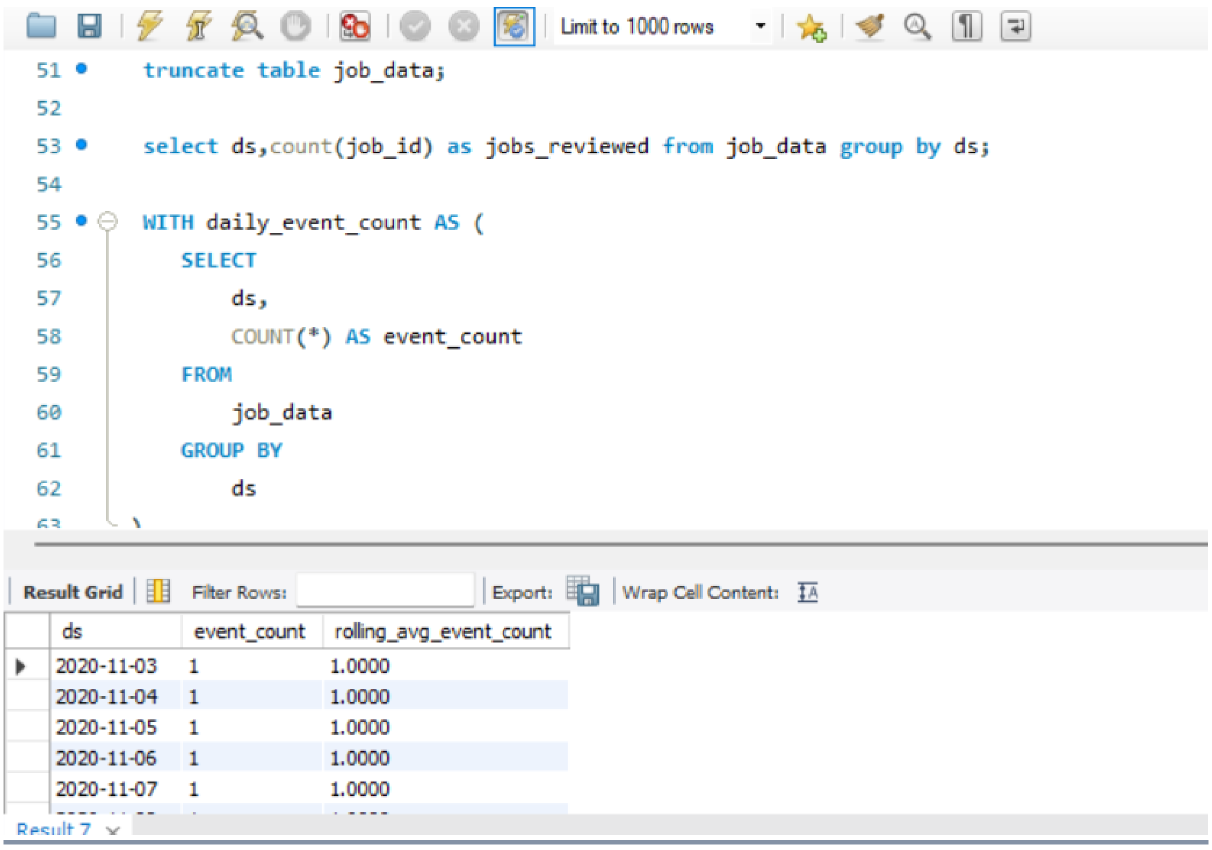
```
('2020-11-08', 3, 1024, 'transfer', 'Gujarati', 37, 'A'),
('2020-11-07', 2, 1025, 'decision', 'Malayalam', 48, 'B'),
('2020-11-06', 1, 1026, 'skip', 'Kannada', 59, 'C'),
('2020-11-05', 0, 1027, 'transfer', 'Odia', 70, 'D'),
('2020-11-04', 29, 1028, 'decision', 'Assamese', 81, 'A'),
('2020-11-03', 28, 1029, 'skip', 'Maithili', 92, 'B');

drop table job_data;
create table job_data(ds date,job_id int,actor_id int,event varchar(10),language varchar(10),t
 truncate table job_data;

 select ds,count(job_id) as jobs_reviewed from job_data group by ds;
```

ds	jobs_reviewed
2020-11-30	2
2020-11-29	1
2020-11-28	2
2020-11-27	1
2020-11-26	1

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.

Query:



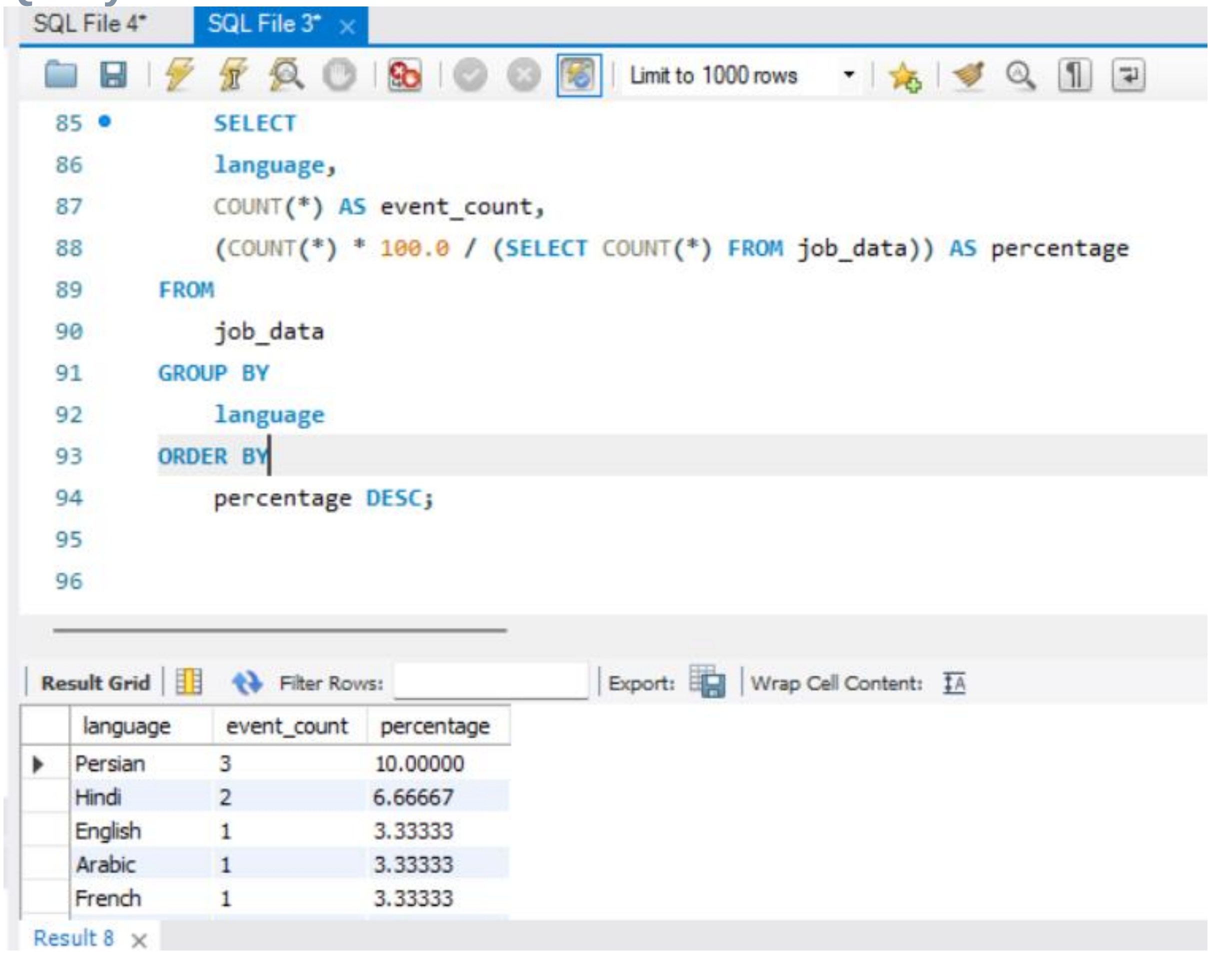
```sql
truncate table job_data;

select ds,count(job_id) as jobs_reviewed from job_data group by ds;

WITH daily_event_count AS (
    SELECT
        ds,
        COUNT(*) AS event_count
    FROM
        job_data
    GROUP BY
        ds
)
```

ds	event_count	rolling_avg_event_count
2020-11-03	1	1.0000
2020-11-04	1	1.0000
2020-11-05	1	1.0000
2020-11-06	1	1.0000
2020-11-07	1	1.0000

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.

Query:



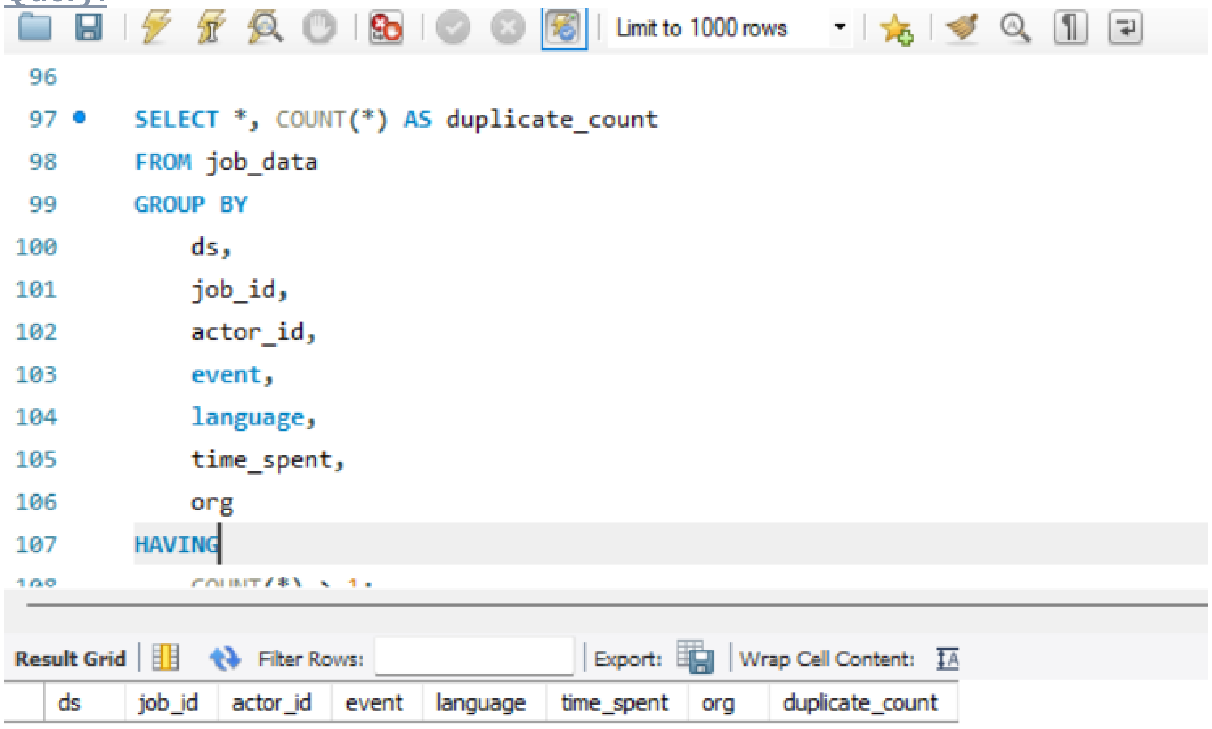
```
SELECT
language,
COUNT(*) AS event_count,
(COUNT(*) * 100.0 / (SELECT COUNT(*) FROM job_data)) AS percentage
FROM
job_data
GROUP BY
language
ORDER BY
percentage DESC;
```

language	event_count	percentage
Persian	3	10.00000
Hindi	2	6.66667
English	1	3.33333
Arabic	1	3.33333
French	1	3.33333

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.**

Query:



```
SELECT *, COUNT(*) AS duplicate_count
FROM job_data
GROUP BY
    ds,
    job_id,
    actor_id,
    event,
    language,
    time_spent,
    org
HAVING
    COUNT(*) > 1;
```

ds	job_id	actor_id	event	language	time_spent	org	duplicate_count