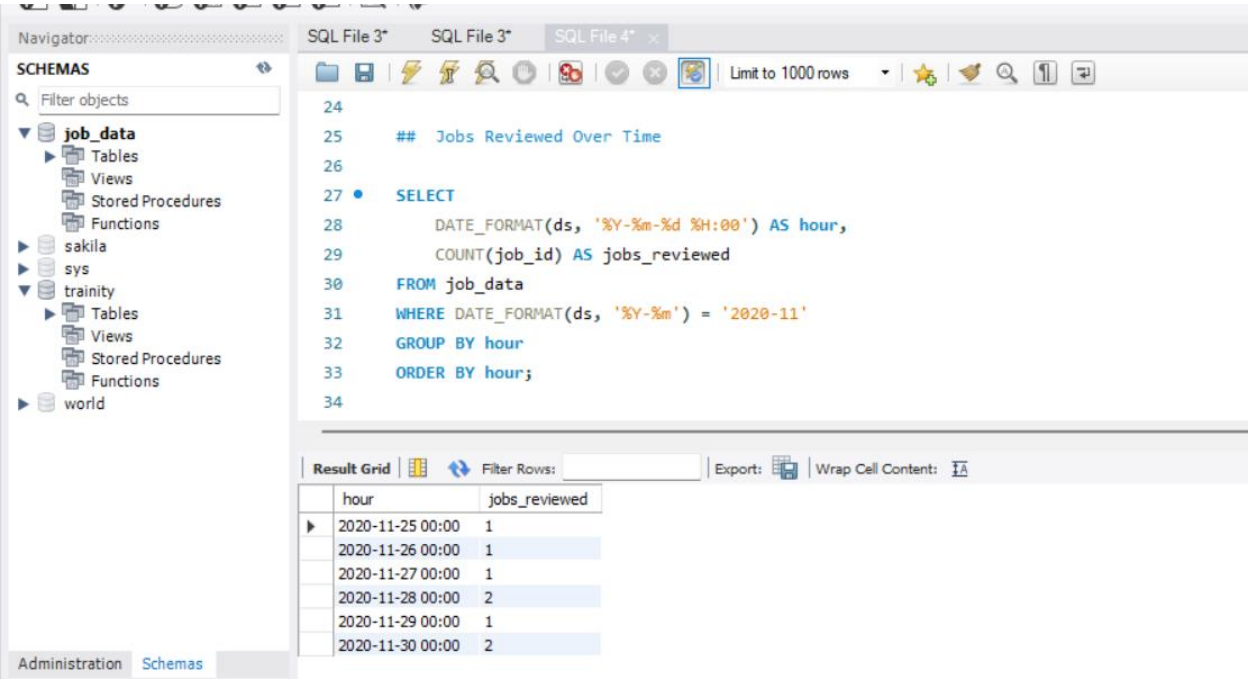
1: Jobs Reviewed Over Time

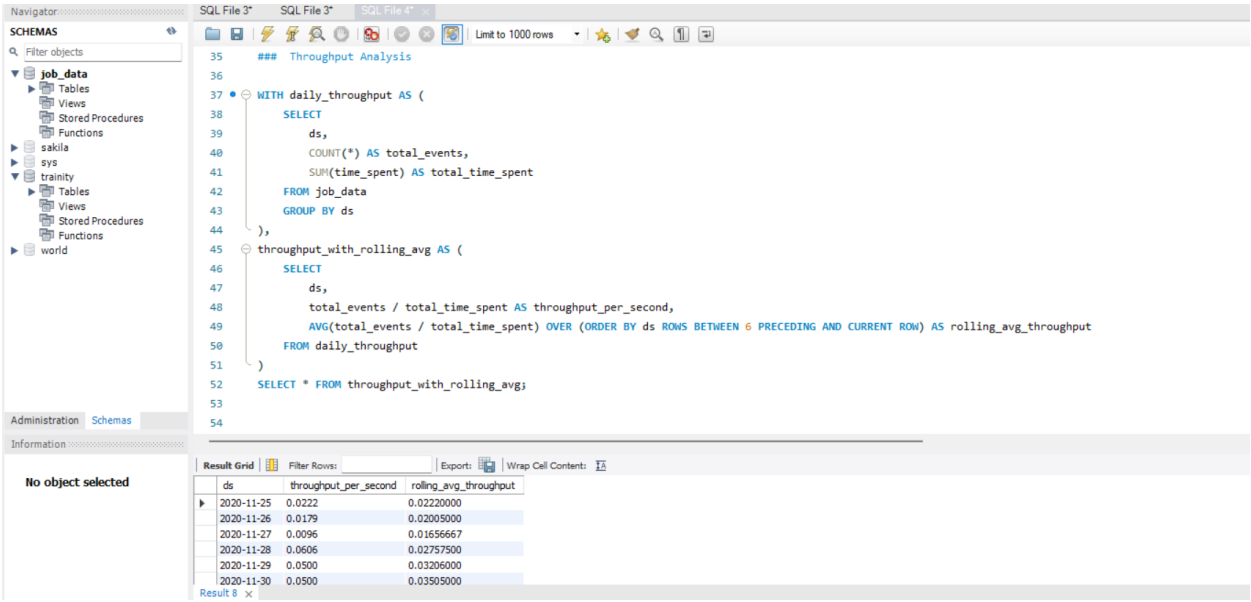
```
SELECT
    DATE_FORMAT(ds, '%Y-%m-%d %H:00') AS hour,
    COUNT(job_id) AS jobs_reviewed
FROM job_data
WHERE DATE_FORMAT(ds, '%Y-%m') = '2020-11'
GROUP BY hour
ORDER BY hour;
```



2: Throughput Analysis

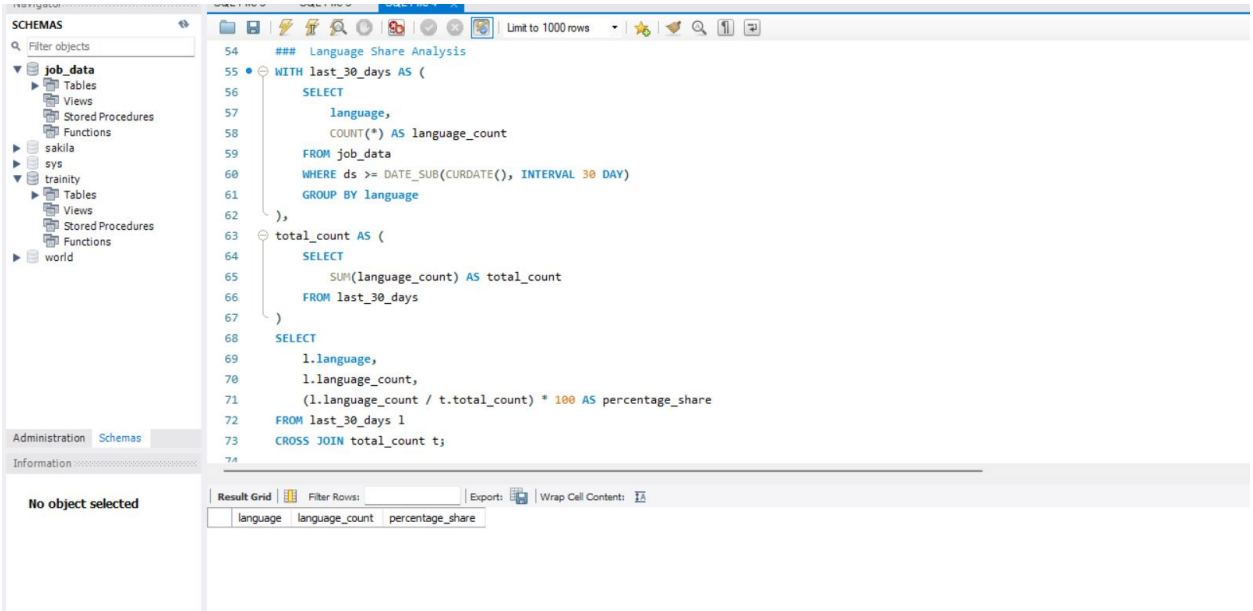
```
WITH daily_throughput AS (
    SELECT
        ds,
```

```
        COUNT(*) AS total_events,
        SUM(time_spent) AS total_time_spent
    FROM job_data
    GROUP BY ds
),
throughput_with_rolling_avg AS (
    SELECT
        ds,
        total_events / total_time_spent AS throughput_per_second,
        AVG(total_events / total_time_spent) OVER (ORDER BY ds ROWS BETWEEN 6 PRECEDING AND CURRENT ROW) AS rolling_avg_throughput
    FROM daily_throughput
)
SELECT * FROM throughput_with_rolling_avg;
```



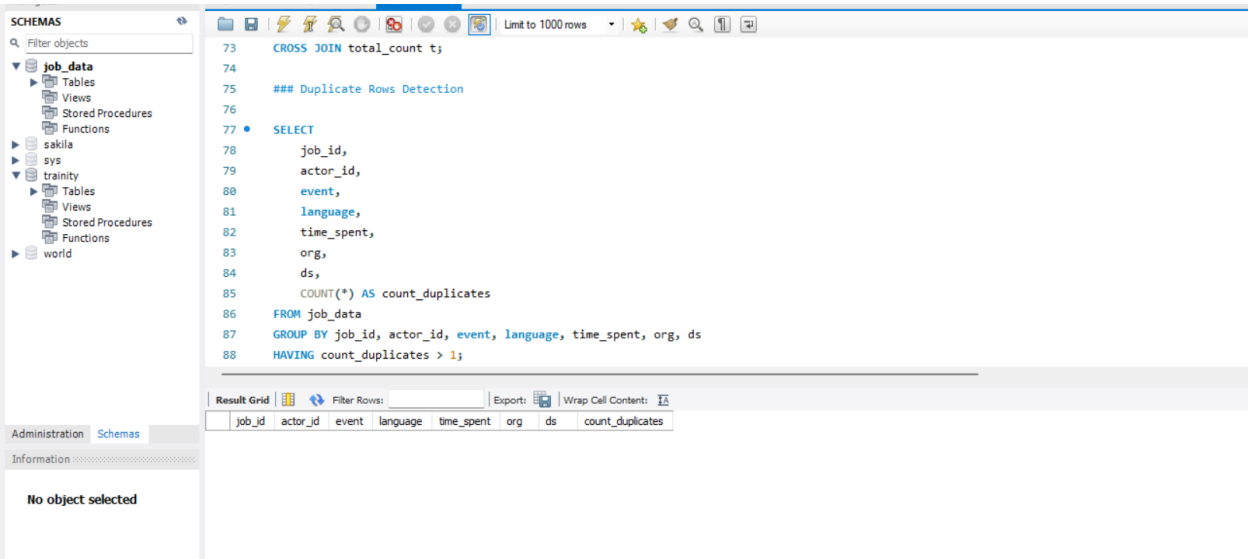
3: Language Share Analysis

```
WITH last_30_days AS (
    SELECT
        language,
        COUNT(*) AS language_count
    FROM job_data
    WHERE ds >= DATE_SUB(CURDATE(), INTERVAL 30 DAY)
    GROUP BY language
),
total_count AS (
    SELECT
        SUM(language_count) AS total_count
    FROM last_30_days
)
SELECT
    l.language,
    l.language_count,
    (l.language_count / t.total_count) * 100 AS percentage_share
FROM last_30_days l
CROSS JOIN total_count t;
```



4: Duplicate Rows Detection

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



Project Description:

This project involves operational analytics where we analyze the company's end-to-end operations using data. The objective is to derive insights from the dataset related to job events, helping different departments to understand and improve various metrics.

Approach:

Data Understanding: Initially, the dataset was loaded and inspected to understand its structure and content.

Data Preprocessing: The date column was converted to a standard format suitable for SQL operations.

Analysis Execution: SQL queries were crafted to address the project tasks, including temporal analysis, rolling average calculation, language distribution, and duplicate detection.

Tech-Stack Used

MySQL Workbench: Used for querying and analyzing the dataset.

Insights

Jobs Reviewed Over Time: This analysis provides a detailed breakdown of job reviews per hour, helping to identify peak review times.

Throughput Analysis: By calculating the 7-day rolling average, we get a smoothed view of throughput, which is useful for identifying trends without the noise of daily fluctuations.

Language Share: Understanding the distribution of job languages helps in allocating resources effectively.

Duplicate Rows: Detecting duplicates ensures data integrity and accuracy in analysis.

Result

The project provided valuable insights into job reviews, throughput performance, language distribution, and data integrity. These insights can help in making informed decisions to enhance operational efficiency and resource allocation.