Title: Query Performance Analysis and Optimization

1. Introduction

Objective: The purpose of this report is to analyze the performance of a specific SQL query, implement optimizations, and evaluate the performance improvements or regressions resulting from these optimizations.

2. Original Query Analysis

Query:

```
SELECT
    c.title,
    c.release_year,
    c.genre,
    COALESCE(ps.name, '(no production studio listed)') AS studio_name
FROM
    content c
LEFT JOIN
    production_studio ps
ON
    c.production_studio_id = ps.id;
```

Execution Plan: To understand how the database executes the query, we examined its execution plan using the EXPLAIN statement.

```
EXPLAIN SELECT
  c.title,
  c.release_year,
  c.genre,
  COALESCE(ps.name, '(no production studio listed)') AS studio_name
FROM
  content c
LEFT JOIN
  production_studio ps
ON
  c.production_studio_id = ps.id;
```

Analysis: The execution plan revealed the following points:

- **Full Table Scan:** The query performs a full table scan on the `content` table.
- **Join Operation:** The join between `content` and `production_studio` could be optimized with proper indexing.
- **Index Usage:** Lack of indexes on `production_studio_id` in the `content` table and `id` in the `production_studio` table.

3. Proposed Optimization

Optimization Strategy: To address the identified inefficiencies, we propose the following optimizations:

1. **Index Creation:** Adding indexes on the columns used in the join condition to improve the performance of the join operation.

SQL Commands:

```
CREATE INDEX idx_content_production_studio_id ON content (production_studio_id);

CREATE INDEX idx_production_studio_id ON production_studio (id);
```

Performance Measurement

Execution Time Before Optimization: The query execution time was measured before implementing the optimizations. The average execution time was recorded as follows:

- **Execution Time:** 16 milliseconds

Execution Time After Optimization: After applying the optimizations, the execution time was measured again. The new average execution time was recorded as follows:

- **Execution Time:** 0 milliseconds
- **Note:** The execution time is not actually 0 its just below the range that mySQL Workbench measures at.

Discussion

Performance Gains/Losses: The performance of the query improved after the optimizations were applied. The execution time decreased by 100%. A decrease by 100% is quite drastic, but the percentage will probably not be as large on much larger dataset (our dataset for this analysis only contained 500 values)

Explanation of Performance Changes: The observed performance improvement can be attributed to the following factors:

- **Index Utilization:** The newly created indexes allowed the database engine to quickly locate the rows involved in the join operation, reducing the need for full table scans.
- **Efficient Join:** With the indexes in place, the join operation became more efficient, leading to faster query execution times.