

Assignment 2

Original query was generated by chat GPT

Step 1: refactor

In this step I have changed a structure of query:

1. Replaced subqueries with CTEs
2. Simplified filtering logic
3. Changed complicated functions (case -> if)

Reasons:

1. Subqueries made the original query harder to read and maintain.
2. Some subqueries are not necessary and can be run as separate query if needed.
3. Simplifying logic with native MySQL functions (like `IF`) improved clarity and slightly boosted performance.

Effect: Query became readable, easier to understand and its performance became slightly better

Step 2: indexing

I have used imported database, so here I added:

1. Primary keys
2. Indexes

Reasons:

1. Primary keys helps to structure data inside the table and helps MySQL with joins
2. Indexes significantly reduce the number of rows scanned during filtering.

Effect: Less rows being checked during query running -> performance improved significantly

In this step I had to use **FORCE INDEX** because DBeaver used primary key. In my case it has enhanced performance (there was 6.6 million before and 860 thousand after)

Sometimes using **USE INDEX**, **FORCE INDEX** or **STRAIGHT_JOIN** can be more effective because SQL server can choose an optimal way as in my case.

Conclusions:

- The optimization process improved both readability and performance of the original query.
- Refactoring made the logic simpler and easier to follow.
- Indexing significantly reduced the number of rows scanned.
- **FORCE INDEX** helped override the default optimizer behavior for better performance.