Order of Query Execution

Timeline

Description automatically generated

* Use inner joins wherever possible to avoid having any unnecessary rows in the resultant table.
* Apply all the required filters to get only the required data values from multiple tables.
* Index the columns that are frequently used in the WHERE clause.
* Avoid using DISTINCT while using the GROUP BY clause, as it slows down query processing.
* Avoid using SELECT \* as much as possible. Select only the required columns.
* Use the ORDER BY clause only if it is absolutely necessary, as it is processed late in a query.
* Avoid using LIMIT and OFFSET as much as possible. Instead, apply appropriate filters using the WHERE clause.

In this video, you learnt the use of the EXPLAIN keyword. You learnt that the 'explain' keyword tells us how many rows were scanned to display the output that we requested. Using the 'explain' keyword, you saw that in the first example when there was no index created, SQL scanned four rows to display the desired result but, after creating the index, only one row was scanned. This factor can reduce the query execution time by a large factor in the case of large databases.