Department of Computer Engineering
University of Peradeniya
CO527 Advanced Database Systems

Lab Number : 02
Topic   : Indexing
Due Date   : 28th November 2022 before 11:50 PM

Objective   :
The objective of this lab assignment is to be focusing on an important technique called **indexing** which is used for improving performance of query processing.

Submission :
Submit a single E/1X/XXXlab02.pdf file including all your .sql scripts and explanations. You can discuss among your classmates, but the submitted assignments should be your own work. Please note that committing plagiarism or assisting others to commit plagiarism will result in zero marks.

Lab Task :

Refer to the same Company database you created for the previous lab(Lab 01 - Review on SQL) and answer the following questions.

1. Assuming no indexes are used, record the query execution time for retrieving all the employees by first name in ascending order.

2. Create an index called *fname_index* on the *first_name* of the employee table. Retrieve all the employees by first name and record the query execution time. Observe the performance improvement gained when accessing with index.

3. Which indexing technique has been used when creating the above index? Hint: You can use *SHOW INDEX FROM [mytable]*; to see details of your indexes.

4. Create a unique index on *emp_no*, *first_name* and *last_name* of employees table. Retrieve all the employees by *emp_no*, *first_name* and *last_name*.

Observe if there is any performance improvement with respect to question1. If not, explain any possible reason.

5. Take the following 3 queries.
   A. select distinct emp_no from dept_manager where from_date>= '1985-01-01' and dept_no>= 'd005';
   B. select distinct emp_no from dept_manager where from_date>= '1996-01-03' and dept_no>= 'd005';
   C. select distinct emp_no from dept_manager where from_date>= '1985-01-01' and dept_no<= 'd009';
      I. Choose one single simple index(i.e index on one attribute) that is most likely to speed up all 3 queries giving reasons for your selection.
      II. For each of the 3 queries, check if MySQL storage engine used that index. If not, give a short explanation why not. You can prefix your select queries with *EXPLAIN EXTENDED* or with *EXPLAIN* to display a query execution plan.

      (Note that in MySQL InnoDB engine uses a clustered index usually on the primary key of the table, by default. We only care about the index you create.)

6. Consider the queries you wrote for questions 2 - 10 in Lab 01 assignment. Give with short explanations, which attributes on which relations should be used for creating indexes that could speed up your queries.

7. Assume that most of the queries on a relation are insert/update/delete. What will happen to the query execution time if that relation has an index created?