

DBMS Lab 8

2019

General Instruction

1. [P] marked questions are for your practice, need not be submitted. There are also some practice problems at the end. [B] marked questions are bonus. You will get 1 mark extra for bonus.
2. Make a single pdf file using screen shots. Work out the questions in the order given and also arrange them in the same order in the submitted pdf.
3. Error messages are also output, we need to check if you are getting correct error messages or not.

-
1. Import database from "largeRelationsInsertFile.sql"
 2. Check the status of student table.
 3. Profiler for MariaDB
 - a. Set *profiling* option on
 - b. Show profile after execution of query to fetch details of students from *student* table whose name is 'wood'.
 - c. What is bottleneck for this query ? [Which task required most of the time]
 - d. Show list of processes running in your DB
 4. Change indexing and check performance
 - a. List all the *storage engines*. Which one is default? Find out which storage engine support hash index.
 - b. Create a table *takes_hash* , cloned from *takes* table but use <ENGINE> as storage engine where <ENGINE> support hash index. Create an hash index '*take_hash_gr*' on *grade* of table *takes_hash*
 - c. Show the current default index present in *takes*. If you try to create an hash index '*take_gr*' on *grade* attribute of *takes*. What will happen? [check what kind of indexes are there in *takes* table and what is the type of '*take_gr*']. Report your observations.
 - d. Compare the performance of following SQL query in both table *takes* and *takes_hash* "select * from <table_name> where grade like '%C%'. Report your observation/
 - e. Try to create an unique index on *grade* of *takes*. Report your observation
 - f. Create an composite index on *ID* and *course_id* in *takes*
 - g. Check the present indexes in *takes*

5. Physical level parameter
 - a. Check the data directory
 - b. Create a test database, and a table candidates with columns id and name. Insert values (1,tom),(2,jerry). Check the files for candidates.
 - c. Change the location of storage and create table places with columns city and country. Insert two records. Check the files for places.
 - d. Create a view view_location by merging candidates and places tables side by side that is four columns and matching the records with the order of input. Check the location of the view, and how it is stored. Compare the size of files corresponding to views and tables.
 - e. Create a trigger check_city that will update entries for cities to Proper casing (that is palakkad -> Palakkad). Check the location where this trigger is created, and the information is stored about the trigger with the table.
 - f. Check the following system variables
 - i. Default storage engine
 - ii. Buffer size
 - iii. Change the buffer size half of current buffer size.
6. [B] Open question: Show something interesting by yourself regarding physical level of MariaDB. Explain what you have shown, and the significance of it. Some examples:
 - a. Partitioning big tables for faster access
 - b. Copying tables through files without going through sql
 - c. Checking logs for executed sql queries
 - d. ...