

**LAPORAN TUGAS BESAR  
DATABASE TUNING**



**DISUSUN OLEH:**

Nama : Rivaldo Fernandes

NIM : 14117103

Kelas : IF3144/ 4847-RA

Dosen : Ahmad Luky Ramdani, S.Komp., M.Kom.

**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI PRODUKSI DAN INDUSTRI  
INSTITUT TEKNOLOGI SUMATERA**

**2019**

## DAFTAR ISI

<b>BAB I.....</b>	<b>3</b>
<b>STUDI LITERATUR.....</b>	<b>3</b>
1.1    Database Tuning Indexing.....	3
1.2    DBMS Configuration .....	4
<b>BAB II .....</b>	<b>7</b>
<b>PERCOBAAN .....</b>	<b>7</b>
2.1    Pengujian Query Execution sebelum dituning .....	7
2.2    Indexing pada database .....	9
2.3    Konfigurasi DBMS.....	12
<b>BAB III.....</b>	<b>15</b>
<b>HASIL DAN PEMBAHASAN .....</b>	<b>15</b>
3.1    Time Eksekusi Query .....	15
3.2    Pembahasan .....	15
<b>DAFTAR PUSTAKA .....</b>	<b>16</b>

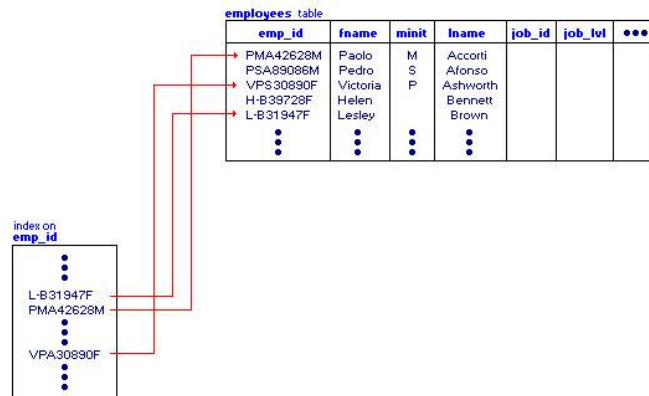
# **BAB I**

## **STUDI LITERATUR**

### **1.1 Database Tuning Indexing**

Tuning pada database adalah untuk meningkatkan performance dari database tersebut, sehingga respon dari database server bisa lebih cepat. misal untuk mempercepat query saat memproses suatu data. pada dasarnya tuning tidak akan terasa jika pada aplikasi database yang kecil, namun pada database dengan data yang banyak dan aktivitas pengambilan data yang cukup padat akan sangat terasa. parameter yang bisa di rubah pada tuning database antara lain config start up database, mempercepat query, index, struktur table dan lainnya.

Indeks dalam database serupa dengan indeks dalam buku. Di suatu buku, satu indeks memungkinkan anda untuk menemukan informasi dengan cepat tanpa membaca seluruh buku. Di suatu database, indeks memungkinkan program database menemukan data di suatu tabel tanpa menelusuri seluruh tabel. Satu indeks di suatu buku adalah daftar kata-kata dengan angka-angka halaman berisi masing-masing kata. Satu indeks di suatu database adalah daftar data tertentu dari tabel dengan lokasi penyimpanan baris dalam tabel berisi masing-masing nilai. indeks dapat diciptakan dimanapun suatu kolom atau suatu kombinasi dari kolom di suatu tabel dan diterapkan dalam wujud B-trees. Satu indeks berisi masukan dengan satu atau lebih kolom (kunci pencarian) dari masing-masing baris di suatu tabel. B-tree disortir di kunci pencarian, dan dapat dicari secara efisien di setiap subset yang terdepan dari kunci pencarian. Sebagai contoh, satu indeks di kolom A, B, C dapat dicari secara efisien di A, A, B, dan A, B, C. Sebagian besar buku berisi satu indeks dari kata-kata umum, nama, tempat, dan seterusnya. Database berisi indeks yang individu untuk jenis atau kolom yang terpilih dari data: ini serupa dengan sebuah buku yang berisi indeks untuk nama dari orang dan indeks lain untuk tempat. Ketika anda membuat suatu database dan menadaptkan kinerja, anda perlu membuat indeks untuk kolom yang digunakan di dalam query untuk menemukan data. Dalam contoh database, tabel karyawan mempunyai satu indeks di kolom emp\_id. Ilustrasi yang berikut menunjukkan bagaimana indeks menyimpan masing-masing nilai emp\_id dan poin-poin ke baris. Ketika aplikasi database melaksanakan suatu statemen untuk menemukan data dalam tabel karyawan berdasar suatu nilai emp\_id yang ditetapkan, mengenali indeks untuk kolom emp\_id dan menggunakan indeks itu untuk menemukan data. Jika indeks tidak ada, melaksanakan suatu permulaan scan tabel yang penuh pada awal tabel dan melangkah melalui masing-masing baris, mencari-cari nilai temp\_id yang ditetapkan.



Aplikasi database secara otomatis membuat indeks untuk jenis tertentu dari batasan (sebagai contoh, KUNCI UTAMA dan batasan UNIQUE ). Anda dapat lebih lanjut menyesuaikan definisi-definisi tabel dengan menciptakan indeks yang tidak terikat pada batasan. Manfaat kinerja dari indeks, bagaimanapun, memerlukan biaya. Tabel dengan indeks memerlukan lebih banyak ruang(space penyimpanan dalam database. Juga, perintah agar sisipan, pembaruan, atau penghapusan data mengambil lebih panjang dan memerlukan lebih banyak waktu proses untuk memelihara indeks. Ketika anda mendisain dan membuat indeks, anda perlu memastikan bahwa kinerja bermanfaat bagi beberapa lebih (dibanding) biaya tambahan dalam ruang(space penyimpanan dan memproses sumber daya.

## 1.2 DBMS Configuration

MySQL adalah salah satu DBMS yang paling populer dan paling banyak digunakan untuk menyimpan data-data baik itu untuk aplikasi desktop, mobile, maupun web. Untuk mempercepat kinerja dari database engine, maka terdapat konfigurasi yang dapat dilakukan yaitu :

### 1. InnoDB\_buffer\_pool\_size

Tips ini berlaku untuk database mysql yang menggunakan engine innodb, jadi jika server anda menggunakan engine InnoDB maka optimasi dasar yang harus anda lakukan adalah mengatur `innodb_buffer_pool_size`, kenapa? Karena pada dasarnya pemrosesan yang dilakukan oleh mysql baik itu insert delete update select itu banyak dilakukan oleh cpu sehingga apabila kita memproses data yang besar maka cpu akan mengalami kenaikan resource yang cukup tinggi dan akhirnya akan terjadi freeze atau crash. Oleh karena itu kita mengatur `innodb_buffer_pool_size` untuk membantu meningkatkan performa pemrosesan data utamanya di proses insert. Karena `innodb_buffer_pool_size` ini memanfaatkan ram yang pemrosesannya jauh lebih cepat daripada hardisk dan resourcenya bisa kita sesuaikan, saran saya adalah maksimal 80% dari ukuran RAM, contoh RAM 8GB maka `innodb_buffer_pool_size` adalah 6GB. Jadi nantinya proses seperti insert pemrosesannya akan dilakukan oleh RAM.

Ada 2 cara mengatur `innodb_buffer_pool_size`.

Pertama, dengan menggunakan perintah sql. Perintahnya `SET GLOBAL innodb_buffer_pool_size = 6G`. Ini jika kalian tidak ingin melakukan restart pada sql. Kedua, dengan menambahkan variabel `innodb_buffer_pool_size = 6G` di bawah konfigurasi-konfigurasi mysql yang sudah ada, kemudian restart service mysql anda. Silahkan gunakan cara manapun yang menurut anda lebih mudah, tapi ingat ukuran `innodb_buffer_pool_size` jangan lebih dari 80% agar sisanya dapat digunakan untuk sistem lain dan tidak menimbulkan crash.

## 2. `innodb_flush_log_at_trx_commit`

Melakukan konfigurasi pada `innodb_flush_log_at_trx_commit` sangat berpengaruh terhadap performa dari mysql, kenapa? karena pada dasarnya proses `innodb` adalah melakukan transaksi dan menyimpan catatan transaksinya. konfigurasi `innodb_flush_log_at_trx_commit` ini berpengaruh pada keamanan data, ada 3 pilihan setting yaitu 0, 1, 2. Saran saya adalah set variabelnya ke 0 atau 2 karena disini proses flush log akan dilakukan lebih cepat, sedangkan pilihan 1 proses flush akan memastikan keamanan data dalam proses transaksi. Namun bukan berarti pilihan 0 atau 2 tidak aman, memang ada resiko tapi itu sangat kecil sehingga lebih baik memilih pilihan 0 atau 2, saya sendiri menerapkan pilihan 2 karena cepat dan lebih aman daripada pilihan 0. Anda dapat merubah nilai variable dari `innodb_flush_log_at_trx_commit` seperti cara di tips no 2.

## 3. Skip Name Resolve

Pencarian DNS (DNS Lookup) untuk host MYSQL bisa dikatakan hampir tidak perlu, karena hanya menambahkan roundtrip tambahan untuk permintaan yang harus diselesaikan. Ini merupakan proses yang banyak berjalan dan cukup membuat delay karena server harus menyelesaikan pencarian DNS tersebut. Oleh karena itu lebih baik server tidak perlu melakukan pencarian DNS. Caranya adalah menggunakan perintah `skip-name-resolve` yang bisa dimasukkan pada file konfigurasi mysql.

## 4. Buat Slow Query Log

Membuat (mengaktifkan) log untuk menyimpan query yang berjalan lambat sangat bermanfaat untuk melakukan evaluasi query yang berjalan lambat dan membebani kinerja server. Untuk mengaktifkan slow query log, perlu ditambahkan konfigurasi berikut pada file konfigurasi mysql:

```
slow_query_log = 1
```

`long_query_time = 3`

`log_output = TABLE`

`Slow_query_log = 1` adalah perintah untuk mengaktifkan logging pada query. Kemudian `long_query_time = 3` adalah batas waktu query dijalankan apabila melebihi 3 detik maka masuk ke dalam slow query log, nilainya dapat diubah sesuai kebutuhan. Untuk `log_output = TABLE` adalah menentukan data query disimpan di dalam table `slow_log` yang ada pada database mysql ini saya sarankan karena lebih mudah daripada output dengan file.

## BAB II

### PERCOBAAN

#### 2.1 Pengujian Query Execution sebelum dituning

Sebelum melakukan tuning, kita eksekusi terlebih dahulu query untuk mengetahui lama proses sebelum melakukan tuning. Terdapat 4 buah dataset yang dapat dieksekusi antara lain :

No	Spesifikasi
1	advisor = 100, student = 100, section = 200,takes = 200
2	advisor = 200, student = 200, section = 400,takes = 400
3	advisor = 500, student = 500, section = 1000,takes = 1000
4	advisor = 700, student = 700, section = 20000,takes = 20000

Query yang akan diproses antara lain :

No	Query
1	SELECT * FROM student
2	SELECT * FROM student WHERE tot_cred > 30;
3	SELECT name, dept_name FROM student WHERE tot_cred > 30;
4	SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id
5	SELECT student.`name`,student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,section.building,course.course_id,course.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id JOIN course ON section.course_id = course.course_id

#### Hasil Pengujian

```

MariaDB [db_1]> show profiles;
+-----+-----+-----+
| Query_ID | Duration | Query |
+-----+-----+-----+
| 1 | 0.21057700 | SELECT * FROM student |
| 2 | 0.00038460 | SELECT * FROM student WHERE tot_cred > 30 |
| 3 | 0.00037970 | SELECT name, dept_name FROM student WHERE tot_cred > 30 |
| 4 | 0.12527170 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id |
| 5 | 0.08908250 | SELECT student.`name`,student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,section.building,course.course_id,course.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id JOIN course ON section.course_id = course.course_id |
+-----+-----+-----+
5 rows in set (0.000 sec)

MariaDB [db_1]>

```

#### Hasil Uji Query Dataset Pertama

```

23 | 0.10409950 | SELECT * FROM student
24 | 0.00048310 | SELECT * FROM student WHERE tot_cred > 30
25 | 0.00079710 | SELECT name, dept_name FROM student WHERE tot_cred > 30
26 | 0.14092480 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.cours
e_id
27 | 0.07765370 | SELECT student.`name`,student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,sec
tion.building,course.course_id,course.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = s
ection.course_id JOIN course ON section.course_id = course.cou
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
15 rows in set (0.000 sec)
MariaDB [db_2]>

```

## Hasil Uji Query Dataset Kedua

```

35 | 0.13079120 | SELECT * FROM student
36 | 0.00074710 | SELECT * FROM student WHERE tot_cred > 30
37 | 0.00054530 | SELECT name, dept_name FROM student WHERE tot_cred > 30
38 | 0.22982640 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.cours
e_id
39 | 0.13552930 | SELECT student.`name`,student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,sec
tion.building,course.course_id,course.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = s
ection.course_id JOIN course ON section.course_id = course.cou
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
15 rows in set (0.000 sec)
MariaDB [db_3]>

```

## Hasil Uji Query Dataset Ketiga

```

47 | 0.15938370 | SELECT * FROM student
48 | 0.00076230 | SELECT * FROM student WHERE tot_cred > 30
49 | 0.00071040 | SELECT name, dept_name FROM student WHERE tot_cred > 30
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
15 rows in set (0.000 sec)
MariaDB [db_4]>

53 | 6.96597440 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
15 rows in set (0.000 sec)
MariaDB [db_4]>

MariaDB [db_4]> show profiles;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Query_ID | Duration | Query |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | 3.96580160 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)
MariaDB [db_4]>

```

## Hasil Uji Query Keempat



## 2.2 Indexing pada database

Untuk melakukan indexing, kita dapat menggunakan perintah berikut:

```
MariaDB [db_1]> alter table instructor add index `dept_name` (`dept_name`) USING BTREE;
Query OK, 0 rows affected (0.144 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [db_1]> alter table student ADD INDEX `dept_name` (`dept_name`) USING BTREE;
Query OK, 0 rows affected (0.190 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [db_1]> alter table takes ADD INDEX `course_id` (`course_id`, `sec_id`, `semester`, `year`) USING BTREE;
Query OK, 0 rows affected (0.192 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [db_1]> alter table section ADD INDEX `building` (`building`, `room_number`) USING BTREE;
Query OK, 0 rows affected (0.175 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

Lakukan indexing untuk setiap pasangan primary key dan search value yang akan diakses query.

Setelah melakukan indexing saat nya melakukan uji eksekusi query

```
XAMPP for Windows - mysql -u root
MariaDB [db_1]> explain select count(*) from advisor
-> ;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | advisor | index | NULL | PRIMARY | 7 | NULL | 100 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)

MariaDB [db_1]> explain select count(*) from student;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | student | index | NULL | dept_name | 25 | NULL | 100 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)

MariaDB [db_1]> explain select count(*) from section;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | section | index | NULL | PRIMARY | 30 | NULL | 200 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)

MariaDB [db_1]> explain select count(*) from takes;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | takes | index | NULL | PRIMARY | 37 | NULL | 200 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)

MariaDB [db_1]>
```

## Dataset Pertama

```
MariaDB [db_1]> show profiles;
+-----+-----+-----+
| Query_ID | Duration | Query |
+-----+-----+-----+
| 1 | 0.00069710 | SELECT * FROM student |
| 2 | 0.00037930 | SELECT * FROM student WHERE tot_cred > 30 |
| 3 | 0.00044140 | SELECT name, dept_name FROM student WHERE tot_cred > 30 |
| 4 | 0.01508020 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id |
| 5 | 0.00220920 | SELECT student.`name`,student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,section.building,course.course_id,rse.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id JOIN course ON section.course_id = course.course_id |
+-----+-----+-----+
```

## Hasil Uji Query

```

MariaDB [db_2]> explain select count(*) from advisor;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | advisor | index | NULL | PRIMARY | 7 | NULL | 200 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.002 sec)

MariaDB [db_2]> explain select count(*) from student;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | student | index | NULL | dept_name | 25 | NULL | 200 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)

MariaDB [db_2]> explain select count(*) from section;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | section | index | NULL | PRIMARY | 30 | NULL | 400 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)

MariaDB [db_2]> explain select count(*) from takes;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | takes | index | NULL | PRIMARY | 37 | NULL | 400 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)

```

## Dataset Kedua

```

MariaDB [db_2]> show profiles;
+-----+-----+-----+
| Query_ID | Duration | Query |
+-----+-----+-----+
| 1 | 0.04164570 | SELECT * FROM student |
| 2 | 0.00084610 | SELECT * FROM student WHERE tot_cred > 30 |
| 3 | 0.00076500 | SELECT name, dept_name FROM student WHERE tot_cred > 30 |
| 4 | 0.17680260 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id |
| 5 | 0.00500910 | SELECT student.name, student.dept_name, takes.sec_id AS pengambilan, takes.semester, section.room_number, section.building, course.co |
| 6 | 0.00000000 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id JOIN course ON section.course_id = course.cou |
+-----+-----+-----+
5 rows in set (0.000 sec)

```

## Hasil Uji Query

```

MariaDB [db_3]> explain select count(*) from advisor;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | advisor | index | NULL | PRIMARY | 7 | NULL | 500 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)

MariaDB [db_3]> explain select count(*) from student;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | student | index | NULL | dept_name | 25 | NULL | 500 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.002 sec)

MariaDB [db_3]> explain select count(*) from section;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | section | index | NULL | PRIMARY | 30 | NULL | 1000 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)

MariaDB [db_3]> explain select count(*) from takes;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | takes | index | NULL | PRIMARY | 37 | NULL | 1000 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)

MariaDB [db_3]>

```

## Dataset Ketiga

```

MariaDB [db_3]> show profiles;
+-----+-----+-----+
| Query_ID | Duration | Query |
+-----+-----+-----+
| 1 | 0.00140230 | SELECT * FROM student |
| 2 | 0.00068670 | SELECT * FROM student WHERE tot_cred > 30 |
| 3 | 0.00058300 | SELECT name, dept_name FROM student WHERE tot_cred > 30 |
| 4 | 0.15644990 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.cour |
| 5 | 0.10606810 | SELECT student.`name`,student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,se |
| 6 | 0.00000000 | dent ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id JOIN course ON section.course_id = course.cou |
+-----+-----+-----+
5 rows in set (0.001 sec)

```

## Hasil Uji Query

## Dataset Keempat

```

MariaDB [db_4]> explain select count(*) from advisor;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | advisor | index | NULL | PRIMARY | 7 | NULL | 700 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)

MariaDB [db_4]> explain select count(*) from student;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | student | index | NULL | dept_name | 25 | NULL | 700 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)

MariaDB [db_4]> explain select count(*) from section;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | section | index | NULL | PRIMARY | 30 | NULL | 23601 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)

MariaDB [db_4]> explain select count(*) from takes;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | takes | index | NULL | PRIMARY | 37 | NULL | 19993 | Using index |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)

MariaDB [db_4]>

MariaDB [db_4]> show profiles
-> ;
+----+-----+-----+-----+
| Query_ID | Duration | Query |
+----+-----+-----+-----+
| 1 | 0.00093930 | SELECT * FROM student |
| 2 | 0.00117370 | SELECT * FROM student WHERE tot_cred > 30 |
| 3 | 0.00089530 | SELECT name, dept_name FROM student WHERE tot_cred > 30 |
| 4 | 2.30447590 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id |
+----+-----+-----+-----+
4 rows in set (0.000 sec)

+----+-----+-----+-----+
| 1 | 1.83667870 | SELECT student.`name`,student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,section.building,course.course_id,course.dep |
+----+-----+-----+-----+
1 row in set (0.000 sec)

```

## Hasil Uji Query

### 2.3 Konfigurasi DBMS

Untuk mengoptimasi kinerja engine, maka perlu dilakukan tuning melalui perubahan konfigurasi, dalam hal ini engine innodb. Berikut beberapa perubahan yang detuning sesuai dengan teori yang sebelumnya dibahas.

```

my.ini - Notepad
File Edit Format View Help
# Comment the following if you are using InnoDB tables
#skip-innodb
innodb_data_home_dir = "C:/xampp/mysql/data"
innodb_data_file_path = ibdata1:10M:autoextend
innodb_log_group_home_dir = "C:/xampp/mysql/data"
innodb_log_arch_dir = "C:/xampp/mysql/data"
## You can set ..buffer_pool_size up to 50 - 80 %
## of RAM but beware of setting memory usage too high
innodb_buffer_pool_size = 4G
## Set ..log_file_size to 25 % of buffer pool size
innodb_log_file_size = 1G
innodb_log_buffer_size = 2G
innodb_flush_log_at_trx_commit = 2
innodb_lock_wait_timeout = 50

## UTF 8 Settings
#init-connect='SET NAMES utf8'
#collation_server=utf8_unicode_ci
#character_set_server=utf8
#skip-character-set-client-handshake

```

## Konfigurasi Engine Innodb

```
MariaDB [(none)]> set global query_cache_size = 268435456;
Query OK, 0 rows affected (0.072 sec)
```

```
MariaDB [(none)]> set global query_cache_type=1;
Query OK, 0 rows affected (0.009 sec)
```

```
MariaDB [(none)]> set global query_cache_limit=1048576;
Query OK, 0 rows affected (0.000 sec)
```

Penambahan Cache untuk mempercepat akses data berulang

Setelah melakukan tuning engine innodb saat nya melakukan uji eksekusi query

```
|      16 | 0.00051260 | SELECT * FROM student |
|      17 | 0.00012430 | SELECT * FROM student WHERE tot_cred > 30 |
|      18 | 0.00011820 | SELECT name, dept_name FROM student WHERE tot_cred > 30 |
|      19 | 0.00012340 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.cours
e_id |
|      20 | 0.00023450 | SELECT student.`name`,student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,sec
tion.building,course.course_id,course.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = s
ection.course_id JOIN course ON section.course_id = course.cou |
+-----+-----+-----+
15 rows in set (0.000 sec)
MariaDB [db_1]>
```

### Hasil Uji Query Dataset Pertama

```
|      21 | 0.00042890 | SELECT * FROM student |
|      22 | 0.00011720 | SELECT * FROM student WHERE tot_cred > 30 |
|      23 | 0.00012310 | SELECT name, dept_name FROM student WHERE tot_cred > 30 |
|      24 | 0.00020020 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id |
|      25 | 0.00025340 | SELECT student.`name`,student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.room_number,section.building,course
.course_id,course.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id JOIN course ON section.
course_id = course.cou |
+-----+-----+-----+
15 rows in set (0.000 sec)
```

### Hasil Uji Query Dataset Kedua

```

40 | 0.00044670 | SELECT * FROM student
41 | 0.00012730 | SELECT * FROM student WHERE tot_cred > 30
42 | 0.00013510 | SELECT name, dept_name FROM student WHERE tot_cred > 30
43 | 0.00052650 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id
44 | 0.00050200 | SELECT student.`name`, student.dept_name, takes.sec_id AS pengambilan, takes.semester, section.room_number, section.building, course
.course_id, course.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id JOIN course ON section.
course_id = course.cou
+-----+
15 rows in set (0.000 sec)
MariaDB [db_3]>

```

## Hasil Uji Query Dataset Ketiga

```

50 | 0.00016320 | SELECT * FROM student
51 | 0.00012360 | SELECT * FROM student WHERE tot_cred > 30
52 | 0.00014950 | SELECT name, dept_name FROM student WHERE tot_cred > 30
+-----+
15 rows in set (0.000 sec)
MariaDB [db_4]>

MariaDB [db_4]> show profiles;
+-----+
| Query_ID | Duration | Query |
+-----+
| 1 | 4.79278000 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id |
+-----+
1 row in set (0.000 sec)

+-----+
| 1 | 2.30913940 | SELECT student.`name`, student.dept_name, takes.sec_id AS pengambilan, takes.semester, section.room_number, section.building, course
.course_id, course.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id JOIN course ON section.
course_id = course.cou |
+-----+
1 row in set (0.000 sec)
MariaDB [db_4]>

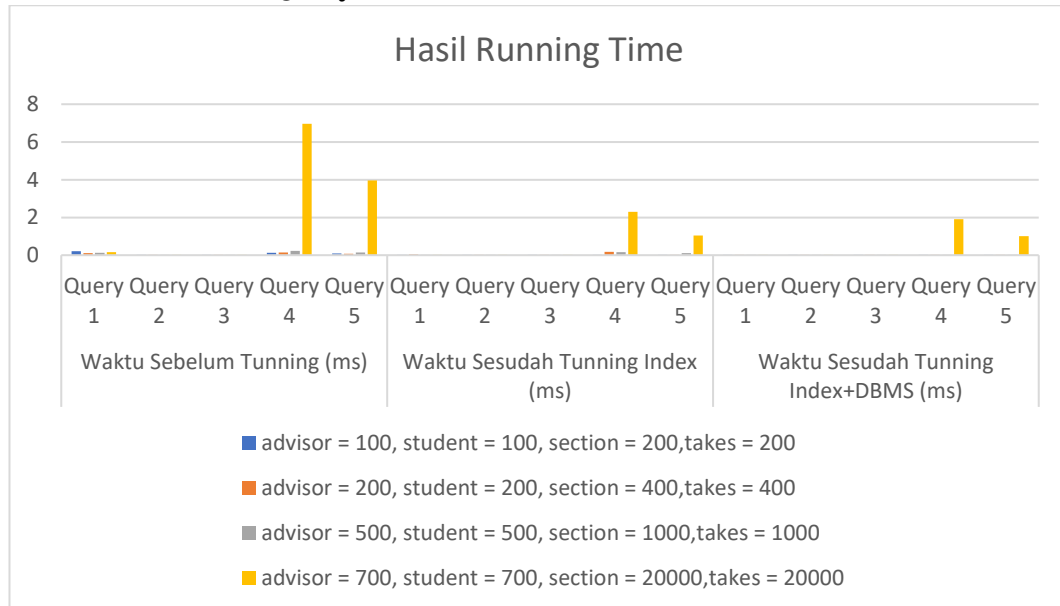
```

## Hasil Uji Query Dataset Keempat

## BAB III

### HASIL DAN PEMBAHASAN

#### 3.1 Time Eksekusi Query



Grafik Perbandingan Kondisi Query sebelum dan sesudah tuning

#### 3.2 Pembahasan

Dari percobaan yang telah dilakukan dapat dilihat bahwa proses tuning bertujuan untuk memaksimalkan kinerja DBMS, proses tuning dapat dilakukan pada berbagai aspek seperti query optimization, hardware tuning, relational schema dan lain-lain, namun dalam percobaan ini digunakan tuning menggunakan indexing dan dbms configuration. Pada tahap tuning pertama, menggunakan indexing, terlihat jelas perbedaan running timenya, dikarenakan proses tuning dengan indexing akan membuat proses pencarian yang tadinya full scan table, menjadi partial scan, sehingga data yang akan dicari akan langsung tertuju pada index yang ditetapkan. Lalu selanjutnya, pada tahap tuning selanjutnya, yaitu konfigurasi engine innodb cukup efektif untuk membantu mempercepat running time, karena konfigurasi dbms akan memaksimalkan kinerja hardware yang digunakan oleh dbms, sehingga lebih efisien dalam penggunaan sumber daya.

## **DAFTAR PUSTAKA**

- [1] MYSQLLAB, "InnoDB," 22 12 2019. [Online]. Available:  
<http://www.mysqlab.net/knowledge/kb/detail/topic/innodb/id/6553>.
- [2] NixCP, "Skip Name Resolve," 22 12 2019. [Online]. Available:  
<https://nixcp.com/skip-name-resolve/>.
- [3] "What is MySQL Query Caching?," 22 12 2019. [Online]. Available:  
<https://www.interserver.net/tips/kb/mysql-query-caching/>.