

# **TUNING DATABASE SYSTEM**

Diajukan kepada pengampu mata kuliah Manajemen Basis Data  
untuk memenuhi Tugas Besar



Dosen Pengampu :

Arief Ichwani, S.Kom. M.Cs.

Disusun oleh :

LATISYA

14117005

Kelas Manajemen Basis Data RC

**PROGRAM STUDI TEKNIK INFORMATIKA**

**INSTITUT TEKNOLOGI SUMATERA**

**2019**

## DAFTAR ISI

DAFTAR ISI.....	ii
DAFTAR GAMBAR .....	iii
BAB I STUDI LITERATUR .....	4
1.1 Tunning : Indexing .....	4
1.2 Tunning : Konfigurasi Database Management System (DBMS) .....	4
BAB II DESKRIPSI PERCOBAAN .....	6
2.1 Tuning Index .....	6
BAB III HASIL DAN PEMBAHASAN .....	9
3.1 Tabel Hasil.....	9
3.2 Grafik Hasil .....	11
3.3 Pembahasan Hasil.....	13
DAFTAR PUSTAKA .....	xiv
LAMPIRAN.....	xv

## DAFTAR GAMBAR

Gambar 1 Setting konfigurasi .....	5
Gambar 2 Generate file java .....	6
Gambar 3 Hasil Generate .....	6
Gambar 4 Proses import database melalui localhost PhpMyAdmin .....	7
Gambar 5 Proses import database melalui command line .....	7
Gambar 6 Data 1 sebelum tuning .....	xv
Gambar 7 Data 1 sesudah tuning .....	xv
Gambar 8 Data 2 sebelum tuning .....	xv
Gambar 9 Data 2 sesudah tuning .....	xvi
Gambar 10 Data 3 sebelum tuning .....	xvi
Gambar 11 Data 3 sesudah tuning .....	xvi
Gambar 12 Data 4 sebelum tuning .....	xvii
Gambar 13 Data 4 setelah tuning .....	xvii

# **BAB I**

## **STUDI LITERATUR**

### **1.1 Tunning : Indexing**

Index adalah daftar isi yang dapat dipakai oleh query tanpa perlu membaca isi tabel secara langsung. Tujuan adanya index dalam database antara lain adalah untuk mempercepat pencarian data berdasarkan kolom tertentu. Tanpa adanya sebuah index pada database, kinerja database sangat menurun dengan sangat signifikan. Hal ini dikarenakan resource CPU banyak digunakan untuk pencarian data atau pengaksesan query SQL dengan metode table-scan. Index membuat pencarian data akan lebih cepat dan tidak banyak menghabiskan resource CPU.

Database Tuning adalah sejumlah aktifitas yang dilakukan untuk memperbaiki atau meningkatkan kinerja atau performance sebuah database. Aktifitas tuning ini meliputi banyak aspek dari software hingga hardware, antara lain *I/O Tuning*, *DBMS Tuning*, *Query Tuning*, dan *Database Maintenance*. Tuning index merupakan salah satu teknik dari DBMS Tuning.

### **1.2 Tunning : Konfigurasi Database Management System (DBMS)**

SQL Server dapat memperoleh tingkat kinerja dengan relatif sedikit konfigurasi tuning yang sangat tinggi. Anda dapat memperoleh tingkat tinggi kinerja dengan menggunakan aplikasi yang baik dan desain database, dan bukan oleh konfigurasi ekstensif penyetelan. Lihat bagian "Referensi" dari artikel ini untuk informasi tentang cara memecahkan berbagai masalah kinerja SQL Server. Ketika Anda mengatasi masalah kinerja, tingkat perbaikan yang tersedia dari konfigurasi tuning biasanya sederhana kecuali jika Anda tidak memiliki sistem yang dikonfigurasi dengan benar. Di SQL Server versi 7.0 dan yang lebih baru, SQL Server menggunakan tuning otomatis konfigurasi dan sangat jarang bahwa pengaturan konfigurasi (Pengaturan lanjut khususnya) perlu perubahan apa pun. Umumnya, tidak membuat perubahan tanpa alasan yang sangat besar dan tidak tanpa berhati-hati metodis pengujian memverifikasi

kebutuhan untuk mengubah konfigurasi konfigurasi SQL Server. Anda harus menetapkan baseline sebelum mengubah konfigurasi sehingga Anda dapat mengukur manfaat setelah perubahan. Jika kita tidak memiliki SQL Server dikonfigurasi dengan benar, beberapa pengaturan mungkin memiliki server yang stabil atau mungkin membuat SQL Server berperilaku tak menentu. Beberapa dukungan dengan banyak lingkungan yang berbeda menunjukkan bahwa pengaturan konfigurasi tidak asli memungkinkan hasil yang netral untuk sangat negative. Setelah melakukan konfigurasi ada baiknya kita memeriksa apakah sudah benar atau tidak, apakah konfigurasi yang kita lakukan dapat meningkatkan kinerja database kita.

Setting konfigurasi yang digunakan :

```
innodb_log_file_size = 10M  
innodb_log_buffer_size = 10M  
innodb_flush_log_at_trx_commit = 1  
innodb_lock_wait_timeout = 100
```

*Gambar 1 Setting konfigurasi*

## BAB II

### DESKRIPSI PERCOBAAN

#### 2.1 Tuning Index

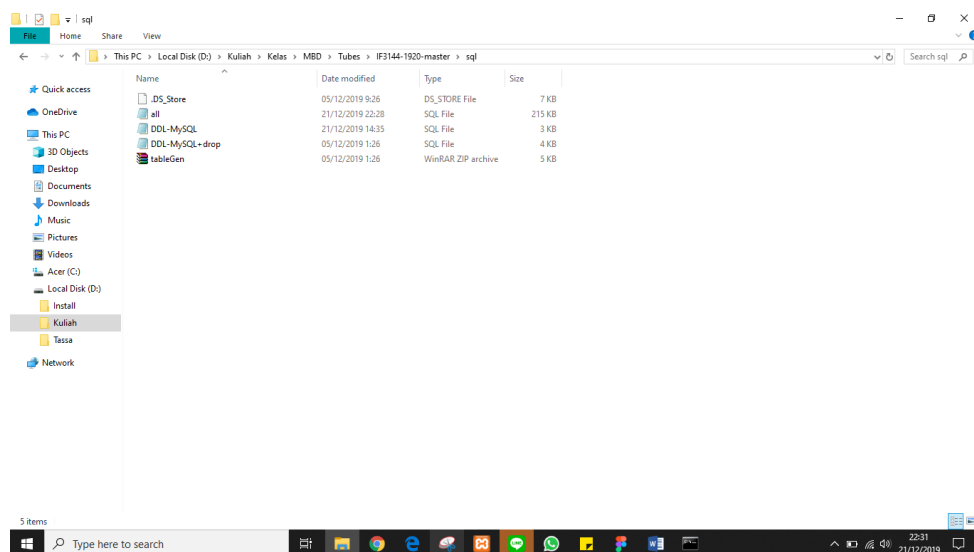
Tahapan tuning indexing adalah sebagai berikut :

- Melakukan generate file tableGen.java pada command line untuk mendapatkan data SQL yang akan dimasukkan ke dalam database.

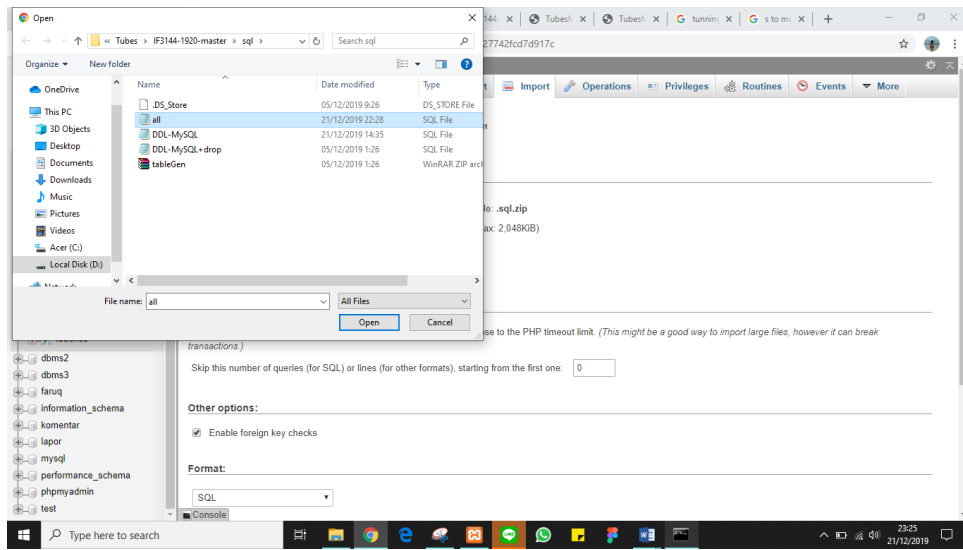
```
928', '469'  
'488', '600'  
'480', '336'  
'709', '552'  
'999', '536'  
'281', '880'  
'259', '552'  
'207', '775'  
'604', '928'  
'719', '968'  
'324', '731'  
'644', '910'  
'775', '138'  
'231', '215'  
'138', '112'  
'654', '731'  
'655', '707'  
'368', '176'  
'270', '469'  
'745', '769'  
'918', '428'  
'378', '968'  
'733', '782'  
'662', '608'  
'306', '552'  
'883', '268'  
'629', '581'  
'118', '214'  
D:\Kuliah\Kelas\MBD\Tubes\IF3144-1920-master>
```

Gambar 2 Generate file java

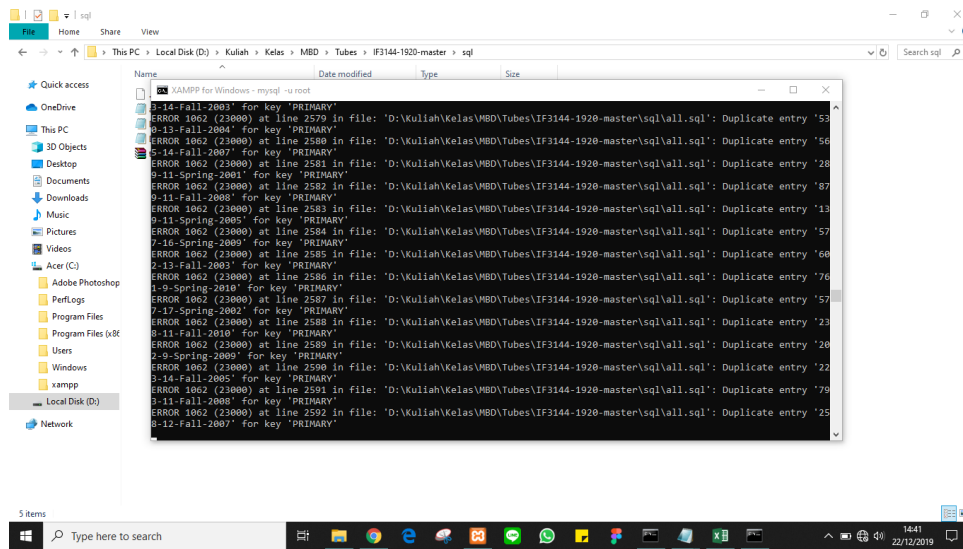
- Setelah itu kita buat database dan table sesuai dengan DDL-MySQL lalu import file all.sql yang sudah kita peroleh tadi ke dalam database



Gambar 3 Hasil Generate



Gambar 4 Proses import database melalui localhost PhpMyAdmin



Gambar 5 Proses import database melalui command line

- c. Ketikkan query yang diminta
- `SELECT * FROM student`
  - `SELECT * FROM student WHERE tot_cred > 30`
  - `SELECT name, dept_name FROM student WHERE tot_cred > 30`
  - `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;`
- d. Setelah query dijalankan, catat waktu eksekusi query ke dalam tabel sebelum melakukan tuning
- e. Lalu buat index pada field yang diinginkan. Index yang saya gunakan adalah sebagai berikut :
- `create index index_tot_cred on student(tot_cred);`
  - `create index index_course_dept_name on course(dept_name);`
  - `create index index_student_id on student(id);`
  - `create index index_takes_id on takes(id);`
  - `create index index_takes_course_id on takes(course_id);`
  - `create index index_section_course_id on section(course_id);`
  - `create index index_course_id on course(course_id);`
- f. Jalankan kembali query, catat waktu eksekusi query ke dalam table sesudah melakukan tuning
- g. Lakukan langkah tersebut ke data lainnya.



## BAB III

### HASIL DAN PEMBAHASAN

#### 3.1 Tabel Hasil

- advisor = 100, student = 100, section = 200, takes = 200

Query	Waktu sebelum Tunning (ms)	Waktu sesudah Tunning (ms)
Q1	0	0
Q2	1	0
Q3	0	0
Q4	4	2
Q5	70	5

- advisor = 200, student = 200, section = 400, takes = 400

Query	Waktu sebelum Tunning (ms)	Waktu sesudah Tunning (ms)
Q1	1	0
Q2	0	0
Q3	0	0
Q4	42	6
Q5	5	4

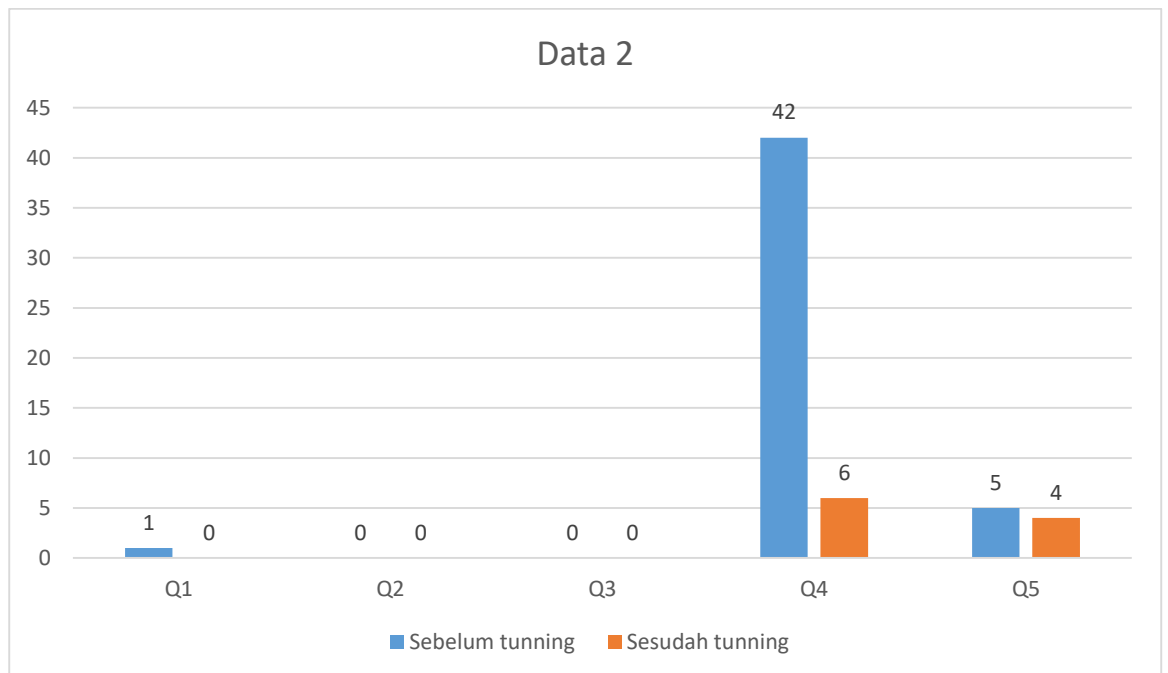
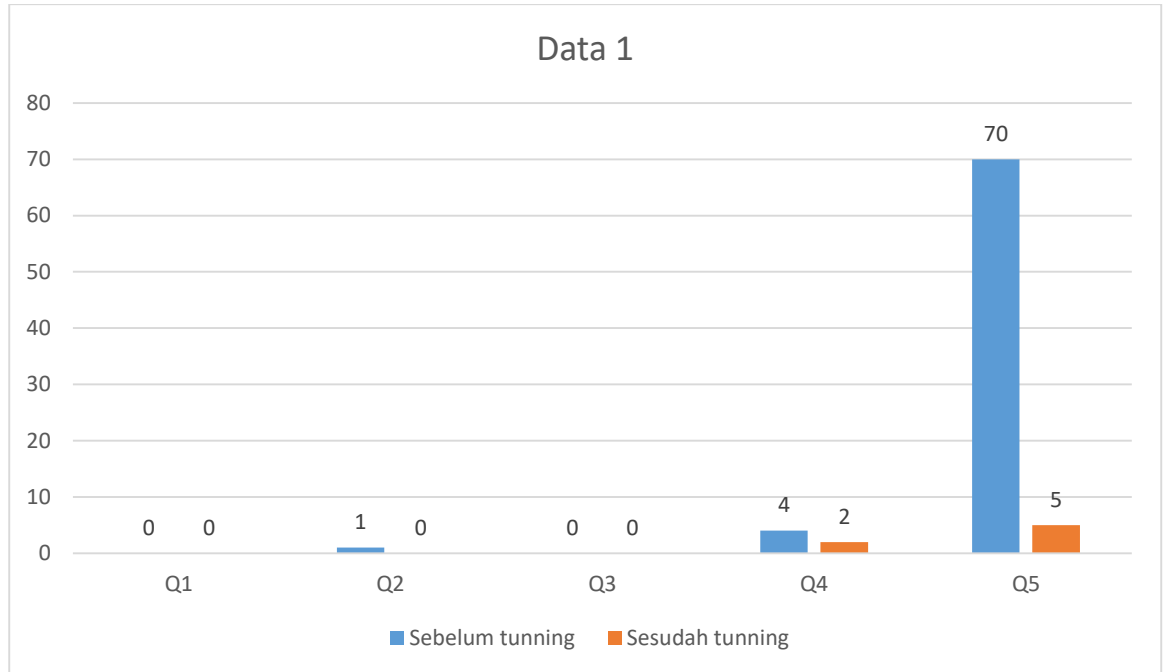
- advisor = 500, student = 500, section = 1000, takes = 1000

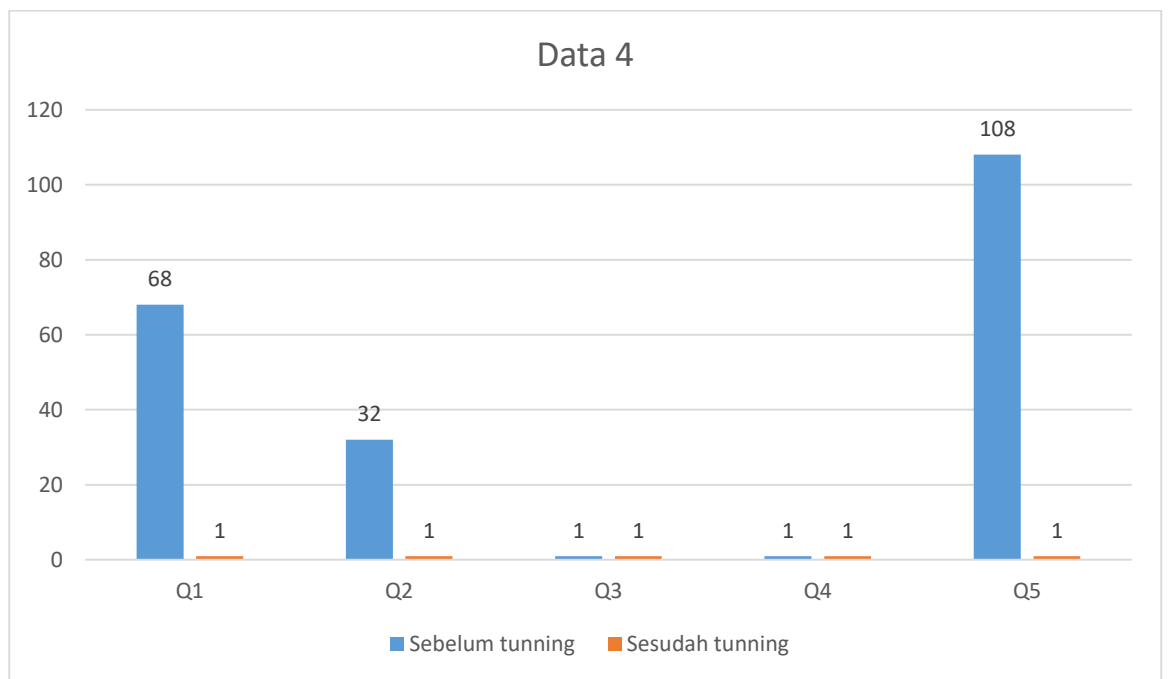
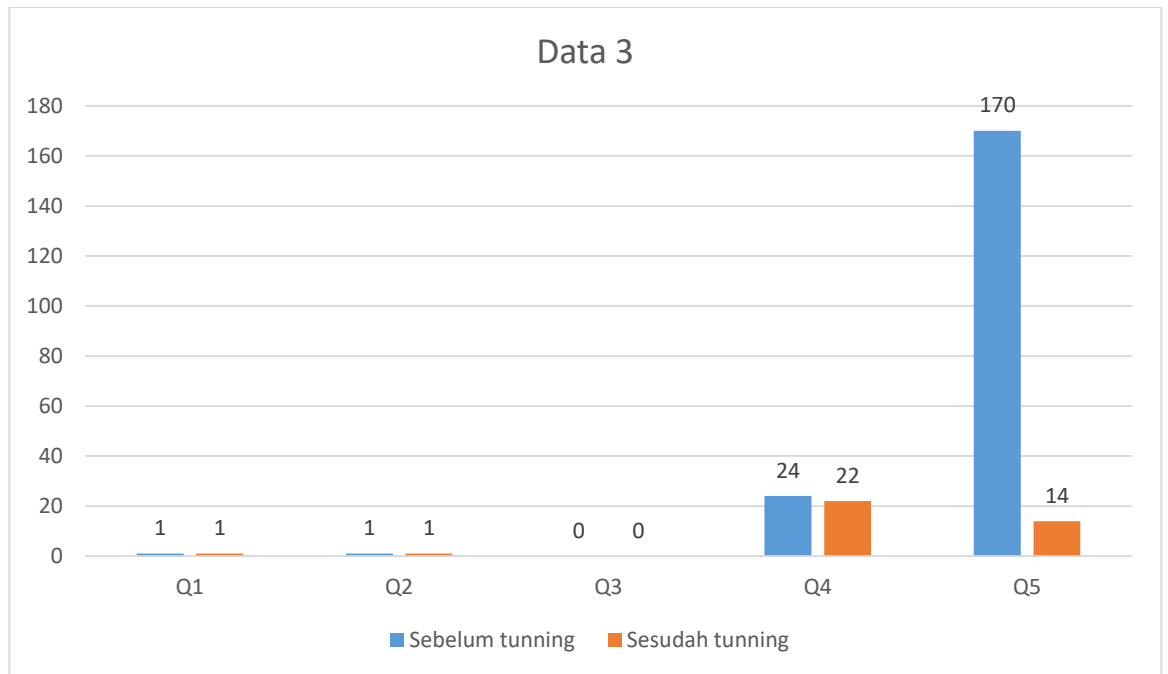
Query	Waktu sebelum Tunning (ms)	Waktu sesudah Tunning (ms)
Q1	1	1
Q2	1	1
Q3	0	0
Q4	24	22
Q5	170	14

- advisor = 700, student = 700, section = 20000,takes = 20000

Query	Waktu sebelum Tunning (ms)	Waktu sesudah Tunning (ms)
Q1	68	1
Q2	32	1
Q3	1	1
Q4	1	1
Q5	108	1

### 3.2 Grafik Hasil





### **3.3 Pembahasan Hasil**

Percobaan ini menggunakan teknik indexing untuk tuning dengan cara membuat index pada atribut dalam tabel yang diakses dalam query. Pada tabel dan grafik dapat kita lihat bahwa waktu eksekusi pada query yang dilakukan setelah indexing relatif lebih cepat dibanding sebelum melakukan indexing. Hal itu dikarenakan dengan adanya indexing, DBMS akan lebih mudah mencari dan mengakses data-data yang ada. Jadi, dapat disimpulkan bahwa tuning dapat digunakan untuk meningkatkan kecepatan dalam eksekusi data sehingga waktu eksekusi query pada data yang dihasilkan akan lebih cepat.

## DAFTAR PUSTAKA

- Inovasi informasi indonesia*. (2018). <https://www.i-3.co.id/2016/10/07/index-pada-database/>
- Microsoft*. (2018, April 22). Cara menentukan pengaturan konfigurasi SQL server yang tepat: <https://support.microsoft.com/id-id/help/319942/how-to-determine-proper-sql-server-configuration-settings>.

## LAMPIRAN

a. Data 1

```
| 3 | 0.00061890 | select * from student |
|
| 4 | 0.00103780 | SELECT * FROM student WHERE tot_cred > 30 |
|
| 5 | 0.00054090 | SELECT `name`, dept_name FROM student WHERE tot_cred > 30 |
|
| 6 | 0.00433510 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = s |
section.course_id |
|
| 7 | 0.07024410 | SELECT student.`name`,student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.roo |
m_number,section.building,course.course_id,course.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN sectio |
n ON takes.course_id = section.course_id JOIN course ON section.course_id = course.cou |
```

*Gambar 6 Data 1 sebelum tunning*

```

15 | 0.00054310 | select * from student
16 | 0.00066990 | SELECT * FROM student WHERE tot_cred > 30
17 | 0.00056500 | SELECT `name`, dept_name FROM student WHERE tot_cred > 30
18 | 0.00293240 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = s
section.course_id
19 | 0.00511690 | SELECT student.`name`, student.dept_name, takes.sec_id AS pengambilan, takes.semester, section.roo
m_number, section.building, course.course_id, course.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN sectio
n ON takes.course_id = section.course_id JOIN course ON section.course_id = course.cou

```

*Gambar 7 Data 1 sesudah tuning*

b. Data 2

```
| 3 | 0.00151950 | select * from student |
| 4 | 0.00075310 | SELECT * FROM student WHERE tot_cred > 30 |
| 5 | 0.00058930 | SELECT `name`, dept_name FROM student WHERE tot_cred > 30 |
| 6 | 0.04272540 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = s |
| section.course_id |
| 7 | 0.00500810 | SELECT student.`name`, student.dept_name, takes.sec_id AS pengambilan, takes.semester, section.roo |
| m_number, section.building, course.course_id, course.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN sectio |
| n ON takes.course_id = section.course_id JOIN course ON section.course_id = course.cou |
```

*Gambar 8 Data 2 sebelum tunning*

```

17 | 0.00084160 | select * from student
18 | 0.00068070 | SELECT * FROM student WHERE tot_cred > 30
19 | 0.00058550 | SELECT `name`, dept_name FROM student WHERE tot_cred > 30
20 | 0.00614820 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = s
section.course_id
21 | 0.00491350 | SELECT student.`name`,student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.roo
m_number,section.building,course.course_id,course.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN sectio
n ON takes.course_id = section.course_id JOIN course ON section.course_id = course.cou

```

*Gambar 9 Data 2 sesudah tuning*

### c. Data 3

```

23 | 0.00109550 | select * from student
24 | 0.00102290 | SELECT * FROM student WHERE tot_cred > 30
25 | 0.00078180 | SELECT `name`, dept_name FROM student WHERE tot_cred > 30
26 | 0.02460170 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = s
section.course_id
27 | 0.17083230 | SELECT student.`name`,student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.roo
m_number,section.building,course.course_id,course.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN sectio
n ON takes.course_id = section.course_id JOIN course ON section.course_id = course.cou

```

*Gambar 10 Data 3 sebelum tuning*

```

35 | 0.00105640 | select * from student
36 | 0.00103350 | SELECT * FROM student WHERE tot_cred > 30
37 | 0.00090040 | SELECT `name`, dept_name FROM student WHERE tot_cred > 30
38 | 0.02250850 | SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = s
section.course_id
39 | 0.01461870 | SELECT student.`name`,student.dept_name,takes.sec_id AS pengambilan,takes.semester,section.roo
m_number,section.building,course.course_id,course.dept_name FROM takes JOIN student ON takes.ID = student.ID JOIN sectio
n ON takes.course_id = section.course_id JOIN course ON section.course_id = course.cou

```

*Gambar 11 Data 3 sesudah tuning*



#### d. Data 4

8	0.06860830	SELECT * FROM student	
9	0.03292430	SELECT * FROM student WHERE tot_cred > 30	
10	0.00116560	SELECT `name`, dept_name FROM student WHERE tot_cred > 30	

Showing rows 0 - 24 (20000 total, Query took 0.0018 seconds.)

```
SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id
```

Showing rows 0 - 24 (2020414 total, Query took 0.1080 seconds.)

```
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
```

Gambar 12 Data 4 sebelum tuning

3	0.00138480	select * from student	
4	0.00136520	SELECT * FROM student WHERE tot_cred > 30	
5	0.00110350	SELECT `name`, dept_name FROM student WHERE tot_cred > 30	

Showing rows 0 - 24 (20000 total, Query took 0.0015 seconds.)

```
SELECT * FROM takes JOIN student ON takes.ID = student.ID JOIN section ON takes.course_id = section.course_id
```

Showing rows 0 - 24 (2020414 total, Query took 0.0017 seconds.)

```
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
```

Gambar 13 Data 4 setelah tuning