

Data Manipulation Language (DML) Database Penjadwalan Dosen menggunakan SQL Server 2008

Didik Setiyadi^{1,*}, Fata Nidaul Khasanah¹, Henderi²

- ¹ Teknik Informatika; Universitas Bina Insani; Jl. Siliwangi No.6 Rawa Panjang Bekasi Timur 17114 Indonesia; Telp. (021) 824 36 886 / (021) 824 36 996. Fax. (021) 824 009 24; e-mail: didiksetiyadi@binainsani.ac.id, fatanidaul@binainsani.ac.id
- ² Teknik Informatika; Universitas Raharja; Jl.Jend.Sudirman No.40 Cikokol Kota Tangerang 15117 Indonesia; Telp. (021) 55749539, email: henderi@raharja.info

* Korespondensi: e-mail: didiksetiyadi@binainsani.ac.id

Diterima: 23 September 2019; Review: 29 September 2019; Disetujui: 2 Oktober 2019

Cara sitasi: Setiyadi D, Khasanah FN, Henderi, 2019. *Data Manipulation Language (DML) Database Penjadwalan Dosen menggunakan SQL Server 2008*. Bina Insani ICT Journal. 6(2): 145-154.

Abstrak: *Data Manipulation Language (DML)* merupakan bahasa basis data yang dipergunakan untuk melakukan modifikasi dan *retrieve* (pengambilan) data pada suatu basis data. DML yang sering digunakan dan terkenal adalah *Structured Query Language (SQL)* yang digunakan untuk mengambil dan memanipulasi data dalam database relasional. SQL terdiri dari sintaks sederhana dalam bentuk instruksi-instruksi dalam melakukan manipulasi data, instruksi tersebut sering disebut dengan *query*. Bahasa *query* tersebut digunakan untuk mengimplementasikan basis data Penjadwalan Dosen dengan menggunakan basis data SQL Server 2008. Langkah awal sebelum melakukan DML dengan menggunakan *query editor*, kita harus memiliki *database* dan data yang telah diimplementasikan dengan SQL Server 2008. Dalam pembahasan ini hanya akan menjelaskan 4 (empat) pernyataan DML yaitu *Select*, *Insert*, *Update* dan *Delete* dengan menggunakan database Penjadwalan Dosen pada RDBMS SQL Server 2008. Pernyataan *select* dimanfaatkan untuk menampilkan data dari suatu tabel (1 atau lebih) pada *database*. *Update* dipergunakan untuk melakukan perubahan data pada tabel dalam *database*. *Delete* dipergunakan menghapus data pada tabel di *database*.

Kata Kunci: DML, Query, Sql Server 2008, Diagram, Penjadwalan Dosen.

Abstract: *Data Manipulation Language (DML)* is a database language that is used to modify and retrieve data in a database. DML which is often used and well known is the *Structured Query Language (SQL)* which is used to retrieve and manipulate data in a relational database. SQL consists of a simple syntax in the form of instructions for manipulating data, these instructions are often called queries. The query language is used to implement the Lecturer Scheduling database using the SQL Server 2008 database. The initial step before doing DML using the query editor, we must have a database and data that have been implemented with SQL Server 2008. In this discussion we will only explain 4 (four) DML statements namely *Select*, *Insert*, *Update* and *Delete* by using the Lecturer Scheduling database on the SQL Server 2008 RDBMS. The *select* statement is used to display data from a table (1 or more) in the database. Updates are used to make changes to data in tables in the database. Delete is used to delete data in a table in the database.

Keywords: DML, Query, Sql Server 2008, Diagram, Lecturer Scheduling.

1. Pendahuluan

Database Management System (DBMS) merupakan kumpulan data yang saling terintegrasi dan dikategorikan dalam sebuah tabel atau beberapa tabel dan sebuah aplikasi

program yang mengatur cara melakukan akses data tersebut. Kumpulan dari data tersebut biasanya dikenal dengan *database* (basis data). Tujuan utama dari DBMS adalah untuk menyediakan sebuah cara untuk menyimpan dan mengambil informasi yang ada dalam basis data secara efektif dan efisien [Widodo, Agus and Kurnianingtyas, 2017].

Database atau basis data terdiri dari kata basis dan data, basis adalah suatu lokasi untuk menyimpan data, sedangkan data merupakan catatan atau kumpulan fakta yang mewakili suatu objek. Dengan demikian basis data adalah koleksi terpadu dari data yang saling terintegrasi yang disimpan pada tabel-tabel didalam suatu DBMS yang diorganisasi untuk memenuhi kebutuhan para *end user* dalam suatu perusahaan [Jayanti, Ari, Dewi, Ketut and Sumiari, Kadek, 2018].

Relational database sebenarnya adalah salah satu konsep dalam melakukan penyimpanan data. Dalam *database relasional*, data disimpan dalam bentuk relasi atau tabel dua dimensi dan antar tabel satu dengan lainnya terdapat *relationship* (hubungan) data. Sebenarnya fungsi RDBMS bukan hanya untuk membuat tabel, isi data, ubah dan hapus data. Untuk manajemen data dalam skala besar dan agar bisa mendukung proses bisnis yang berkelanjutan dan *real time*. Suatu RDBMS dipastikan mempunyai kemampuan dalam manajemen *user* dan keamanan data, *backup* dan *recovery* data serta kemampuan lainnya yang berkaitan dengan kecepatan pemrosesan data [Sutejo, 2010].

SQL Server 2018 adalah sebuah terobosan baru dalam *microsoft* bidang pengembangan *database*. *SQL Server* merupakan salah satu DBMS (*Database Management System*) yang diciptakan *Microsoft* untuk melakukan pengolahan data pada dunia usaha dan industri dalam persaingan pengelolaan basis data seperti IBM dan Oracle [Komputer, 2010]. *Microsoft SQL Server* atau sering disebut *SQL Server* mendukung *SQL* sebagai bahasa untuk melakukan penelusuran data (kueri) di dalam tabel-tabel yang ada di basis data. *Microsoft SQL Server* banyak digunakan pada dunia bisnis, pendidikan atau juga pemerintahan sebagai solusi *database* atau penyimpanan data [Nuryana and Sulistiyono, 2014].

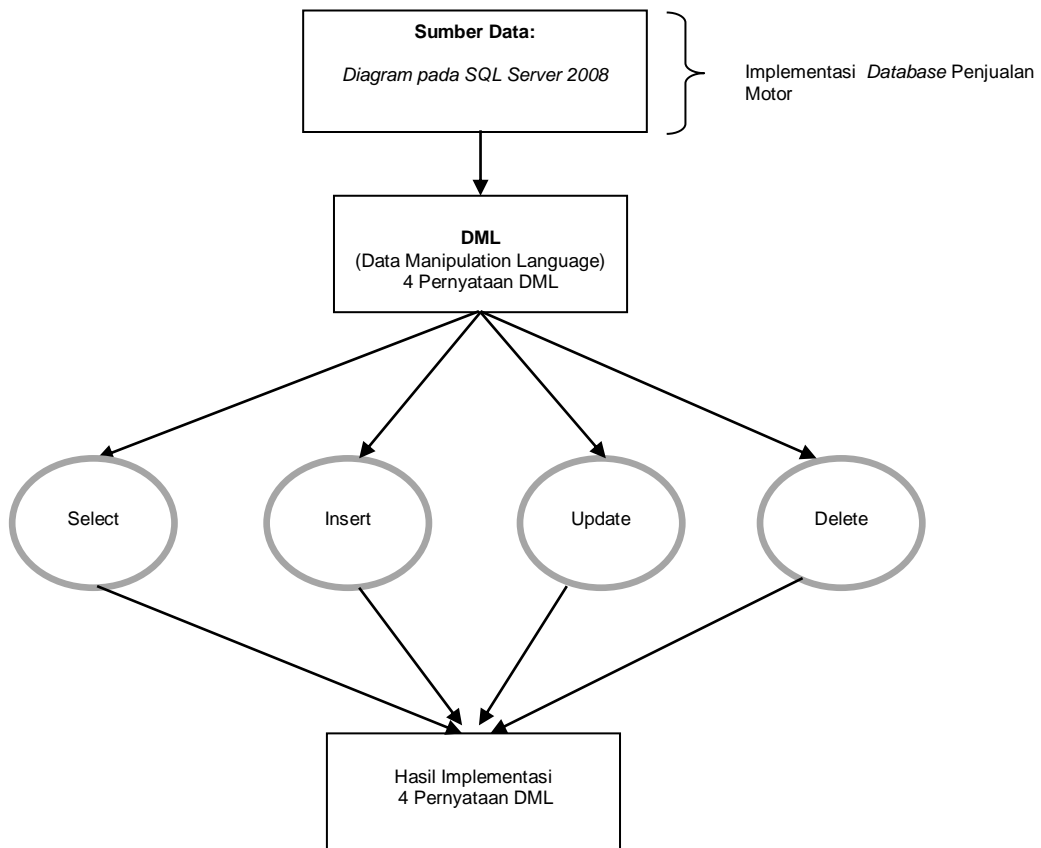
SQL (Structured Query Language) pada dasarnya adalah bahasa komputer standar yang ditetapkan untuk mengakses dan memanipulasi sistem *database*. Sebuah *database* berisi satu tabel atau lebih yang memiliki nama yang berbeda untuk masing-masing tabel. *Query* digunakan untuk mengakses dan mengolah *database*. *SQL* terdiri dari 5 (lima) bagian yaitu retrieving data, *data definition language* (DDL), *data manipulation language* (DML), *data control language* (DCL) dan *data transaction language* (DTL) [Utami and Sukrisno, 2008]

Berdasarkan uraian tersebut diatas, dalam penelitian ini membahas tentang bagaimana melakukan DML dengan menggunakan *query editor* untuk *database Penjadwalan Mengajar* dengan menggunakan RDBMS *SQL Server 2008*. Implementasi didasarkan dari Diagram dan data yang telah dibuat. Implementasi DML dengan *query* di *query editor* pada *SQL Server 2008* terdiri dari 4 (empat) pernyataan DML yaitu *Select*, *Insert*, *Update* dan *Delete*.

2. Metode Penelitian

Studi pustaka merupakan metode penelitian yang dipergunakan penulis dalam melakukan pengumpulan data dan fakta dalam melakukan penelitian ini. Pengumpulan data yang dilakukan dengan teknik studi pustaka diperoleh dari literatur buku-buku dan publikasi jurnal yang relevan dengan masalah yang akan dibahas.

Database System Development Life Cycle (DSDLC) merupakan metode yang digunakan dalam penelitian ini yang meliputi tahapan perencanaan basis data, definisi sistem, analisis dan pengumpulan kebutuhan, desain basis data, seleksi basis data, desain aplikasi, prototipe, implementasi, konversi data, pengujian dan pemeliharaan operasional [Indrajani, 2018]. Pada penelitian ini hanya dibatasi pada tahapan implementasi basis data hanya pada 4 (empat) pernyataan DML yaitu *Select*, *Insert*, *Update* dan *Delete* yang menggunakan *SQL Server 2008*. Berikut ini adalah kerangka pemikiran yang merupakan langkah-langkah yang dilakukan dalam mengimplementasikan DML untuk *database Penjadwalan Dosen* yang dilakukan dengan menggunakan RDBMS *SQL Server 2008*.

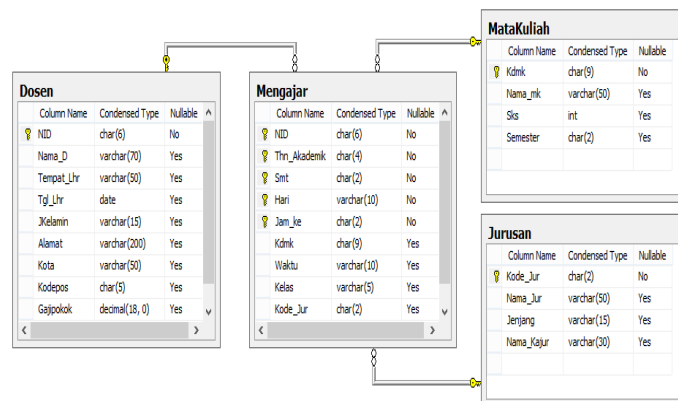


Sumber: Hasil Penelitian (2019)

Gambar 1. Kerangka Pemikiran

Penjelasan dari Gambar 1 adalah bahwa untuk melakukan implementasi 4 (empat) pernyataan DML dengan *database* Penjualan Motor ke *SQL Server 2008*, tahapan awal yaitu dengan mendapatkan sumber data *Diagram* (table dan data-datanya) dari *database Penjadwalan Mengajar*. Selanjutnya dilakukan implementasi DML dengan menggunakan RDBMS *SQL Server 2008* dengan editor *query* dari 4 (empat) pernyataan DML *select*, *insert*, *update* dan *delete* untuk *database Penjadwalan Mengajar* tersebut.

Sumber data diperoleh dari basis data *Penjadwalan Mengajar* yang terdiri dari 4 (empat) tabel yaitu *Dosen*, *Matakuliah*, *Jurusan* dan *Mengajar*.



Sumber: Hasil Penelitian (2019)

Gambar 2. Diagram Penjadwalan Mengajar

Adapun data-data dari 4 (empat) tabel pada gambar 2 diatas akan dijelaskan lebih detail pada tabel 1, 2, 3 dan 4. Gambar 3 menjelaskan detail data yang ada di dalam tabel dosen. Jumlah data diinput sebanyak 13 *record*.

	NID	Nama_D	Tempat_Lhr	Tgl_Lhr	JKelamin	Alamat	Kota	Kodepos	Gajipokok
1	00001	Riswoko Sasono,MMSi	Bogor	1974-12-24	Pria	Jl.Catur 3 No.23	Bekasi Utara	52123	1300000
2	00002	Hasta Riyanti,Ir,MM	Jakarta	1969-05-27	Wanita	Jl.Delima 2 No.2	Bekasi Timur	45612	1150000
3	01001	Cokro Diningrat,Ir,MT	Bekasi	1966-12-19	Pria	Jl.Dayang Raya 2 No.12	Cibitung	54456	1200000
4	01002	Sakib Aljaber,MT	Cikarang	1970-05-20	Pria	Jl.Kemunig 1 No.1	Cikarang	56234	1150000
5	95001	Bambang Sutedjo,Ir,MMSi	Jakarta	1958-02-23	Pria	Jl.Mawar 6 No.2	Jakarta Selatan	23234	1300000
6	95002	Asri Kasetyaningsih, M.Kom	Semarang	1962-12-25	Wanita	Jl.Perjuangan 3 No.11	Bekasi Timur	54567	1200000
7	96001	Triyatno,Ir,MM,M.Kom	Bekasi	1967-05-14	Pria	Jl.Mawar Indah 1 No.1	Bekasi Barat	54356	1100000
8	96002	Diastuti Pujiningsih,MM,MT	Surabaya	1970-12-24	Wanita	Jl.Jarur Kuning 2 No.56	Cibitung	54566	1000000
9	97001	Endang Junianti,Ir,MMSi	Bandung	1967-05-20	Wanita	Jl.Kemukus 2 No 56	Cikarang	56555	1150000
10	97002	Djoko Pamungkas,M.Kom	Bogor	1971-05-23	Pria	Jl.Anggrek 7 No.1	Bekasi Timur	45666	1100000
11	98001	Didik Atmadja,Ir,MMSi	Bandung	1970-04-20	Pria	Jl.Cipete Raya No.6	Jakarta Selatan	24123	1250000
12	98002	Bagus Windarjo,M.Kom	Tangerang	1971-12-13	Pria	Jl.H.Ali No.21	Cibitung	54523	1150000
13	99001	Dewi Anjani,Ir,MM	Bekasi	1975-12-14	Wanita	Jl.Kemang Raya 2 No.1	Bekasi Barat	54567	1200000

Sumber: Hasil Penelitian (2019)

Gambar 3. Data pada tabel Dosen

Gambar 4 berisi penjelasan detail data yang ada di dalam tabel matakuliah. Jumlah data diinput sebanyak 19 *record*.

	Kdmk	Nama_mk	Sks	Semester
1	MKB331201	PENGANTAR TEKNOLOGI INFORMASI	3	1
2	MKB331203	PRAKTIKUM PAKET PROGRAM APLIKASI I	1	1
3	MKB331205	PERANCANGAN BASIS DATA	4	2
4	MKB331206	PAKET PROGRAM APLIKASI II	1	2
5	MKB331209	ALGORITMA PEMROGRAMAN I (PASCAL)	3	2
6	MKB331210	PRAK.ALGORITMA PEMROGRAMAN I (PASCAL)	1	2
7	MKK231201	DASAR MANAJEMEN DAN BISNIS I	2	1
8	MKK231202	DASAR MANAJEMEN DAN BISNIS II	2	2
9	MKK231203	MATEMATIKA DISKRIT	2	1
10	MKK231204	MATEMATIKA DISKRIT II	2	2
11	MKK231205	AKUTANSI DASAR I	2	1
12	MKK231206	AKUTANSI DASAR II	2	2
13	MKK231213	PENGANTAR EKONOMI	2	1
14	MPK131201	PENDIDIKAN AGAMA I	2	1
15	MPK131203	PENDIDIKAN PANCASILA	2	1
16	MPK131204	PENDIDIKAN KEWARGANEGARAAN	2	2
17	MPK131205	BAHASA INGGRIS I	2	1
18	MPK131206	BAHASA INGGRIS II	2	2
19	MPK231207	SISTEM BASIS DATA a	2	1

Sumber: Hasil Penelitian (2019)

Gambar 4. Data pada tabel Matakuliah

Gambar 5 berisi penjelasan detail data yang ada di dalam tabel mengajar. Jumlah data diinput sebanyak 10 *record*.

	NID	Thn_Akademik	Smt	Hari	Jam_ke	Kdmk	Waktu	Kelas	Kode_Jur
1	00001	2004	1	Rabu	1	MKB331205	8:00	T202	TI
2	00001	2004	1	Senin	1	MKB331201	8:00	M101	MI
3	00002	2004	2	Jum'at	1	MPK131204	8:00	S201	SI
4	00002	2004	2	Jum'at	2	MPK131204	10:00	S202	SI
5	95001	2004	1	Kamis	1	MKB331201	8:00	T101	TI
6	95001	2004	1	Senin	1	MKB331201	8:00	M101	MI
7	98002	2004	2	Rabu	1	MKB331206	8:00	S201	SI
8	98002	2004	2	Selasa	2	MPK131204	10:00	M201	MI
9	99001	2004	1	Senin	1	MKB331201	8:00	M102	MI
10	99001	2004	2	Selasa	1	MKB331206	8:00	T201	TI

Sumber: Hasil Penelitian (2019)

Gambar 5. Data pada tabel Mengajar

Gambar 6 berisi penjelasan detail data yang ada di dalam tabel jurusan. Jumlah data diinput sebanyak 5 record.

	Kode_Jur	Nama_Jur	Jenjang	Nama_Kajur
1	KA	Komputerisasi Akuntansi	Diploma 3	Rini Wulandari, MM, MMSi.
2	MI	Manajemen Informatika	Diploma 3	Wahono Diprojo, MM, M.Kom.
3	SI	Sistem Infomasi	Strata 1	Bagus Hermansyah, MM.Si, M.Kom
4	TI	Tekhnik Informatika	Starta 1	Fadjar Sasongko, MT, M.Kom.
5	TK	Tekhnik Komputer	Diploma 3	Agus Budiyantera, T, MT.

Sumber: Hasil Penelitian (2019)

Gambar 6. Data pada tabel Jurusan

3. Hasil dan Pembahasan

Pada bagian ini dijelaskan mengenai hasil dan pembahasan dari penelitian yang telah dilakukan.

Pernyataan DML Select, dalam melakukan pengambilan data dari tabel pada database, SQL menyediakan perintah *select* untuk melakukan akses data. Dengan sintaks sebagai berikut:

```
Select [All | Distinct ] Pilih_daftar_kolom
      [ Into [ Nama_tabel_baru ] ]
From Nama_tabel / Nama View
[ Where   Klausa ]
[ Group By Klausa ] .....(1)
[ Having  Klausa ]
[ Order By Klausa ]
[ Compute Klausa ]
```

Pilih_daftar_kolom menyatakan pilihan terhadap kolom atau atribut dari data yang dipilih. Sedangkan *nama_tabel* merupakan tabel yang akan diambil datanya.

Berikut ini contoh dari penggunaan sintaks query untuk select :

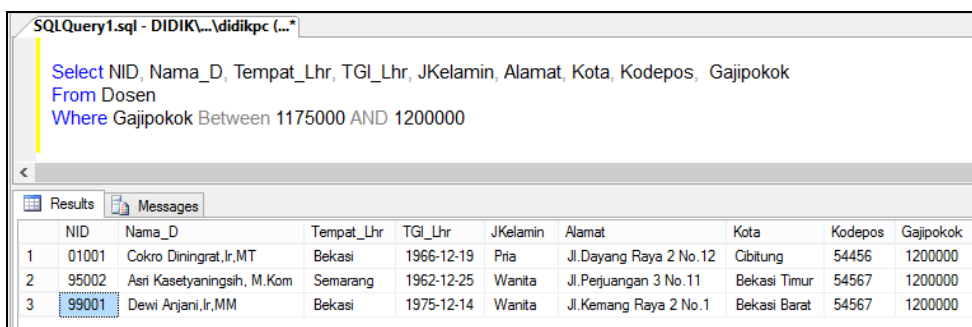
Perintah Select Untuk Memilih Semua Kolom, sebagai contoh misalkan akan menampilkan data pada *table* Dosen pada database Penjadwalan Mengajar, dapat menggunakan 2 (dua) cara: Dengan menggunakan tanda (*)



Sumber: Hasil Penelitian (2019)

Gambar 7. Select dengan tanda * dari 1 table dengan kriteria IN

Dengan memilih seluruh kolom

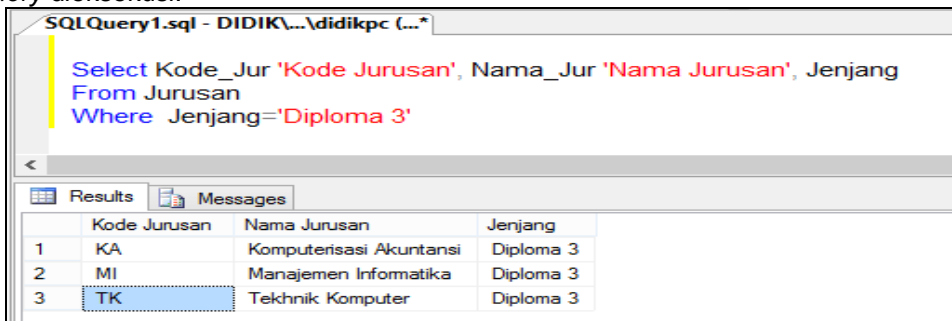


Sumber: Hasil Penelitian (2019)

Gambar 8. Select memilih seluruh kolom dari 1 table dengan kriteria Between

Perintah Select Untuk Memilih Kolom Tertentu Dan Membuat Alias Kolom

Sebagai contoh misalkan akan menampilkan data pada *table* Jurusan pada *database* Penjadwalan Mengajar, dimana kolom yang ditampilkan tertentu dan dirubah tampilan kolom saat *query* dieksekusi.



Sumber : Hasil Penelitian (2019)

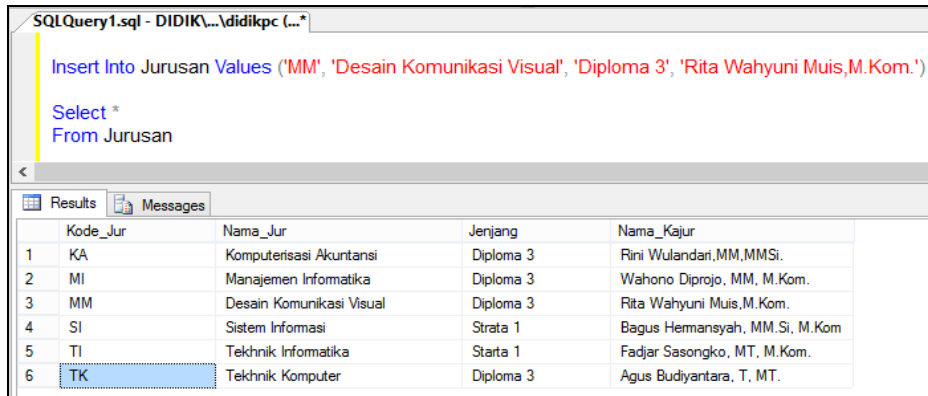
Gambar 9. Select untuk memilih kolom tertentu dan membuat alias kolom

Pernyataan DML Insert, terdapat 2 (dua) bentuk pernyataan kalimat *Insert*, yaitu bentuk pertama *Insert* dimana memungkinkan satu baris tunggal disisipkan kedalam table dan bentuk kedua *Insert*, dimana memungkinkan banyak baris sekaligus dikopikan kesatu table atau lebih. Bentuk pertama *Insert*. Setelah struktur tabel terbentuk, data dapat dimasukkan ke dalam tabel dengan perintah *Insert* dengan menggunakan *query*. Sintaks untuk melakukan insert data kealam suatu tabel adalah sebagai berikut:

Insert [Into] Nama Table [Daftar_Kolom] Value Daftar Nilai.....(2)

Insert, perintah ini mendiskripsikan penambahan data pada nama tabel yang akan dilakukan penambahan data tersebut. *Value*, perintah ini mendiskripsikan nilai data yang akan

ditambahkan pada tabel tersebut. **Daftar_Kolom**, merupakan daftar kolom yang dipisahkan oleh tanda koma menyatakan kolom-kolom yang akan diisi data. Jika tidak ada kolom yang dinyatakan, berarti semua kolom di dalam akan diisi data. Jika hanya sebagian daftar yang dinyatakan, nilai null atau nilai default akan diisikan ke kolom yang tidak disebutkan dalam daftar kolom. **Daftar Nilai**, daftar nilai untuk kolom tabel yang akan disisipkan sebagai sebuah baris data dalam tabel. Data yang diberikan pada daftar nilai harus sesuai dengan daftar kolom. Berikut ini adalah contoh melakukan *insert* data 1 record pada tabel Jurusan untuk seluruh kolom yang dilakukan *insert* datanya.



Sumber : Hasil Penelitian (2019)

Gambar 10. Bentuk pertama *insert*

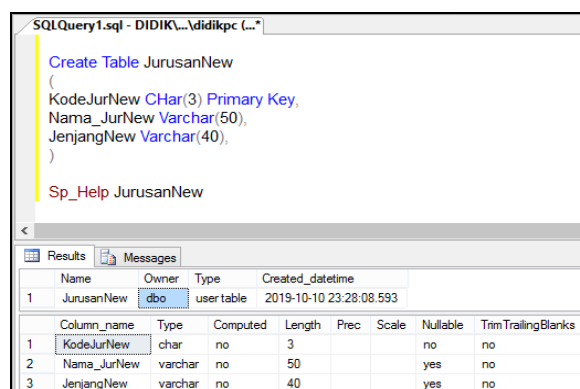
Bentuk kedua *Insert*, sintaks untuk melakukan *insert* untuk melakukan *copy data* kedalam suatu *table* adalah sebagai berikut:

```

Insert Into nama_table [(daftar_kolom)] ..... (3)
Select ...
```

Nama_table dan daftar_kolom dianalogikan semula dari 1 row tunggal disisipkan. Klausula *Select* berupa sembarang pernyataan *Select* absah. Batasan pada bentuk pertama dapat diterapkan pada bentuk kedua ini. Kalimat *Insert* mempunyai batasan – batasan dalam penggunaannya, yaitu 1). Kita harus selalu menspesifikasikan nilai – nilai untuk kunci utama dan kolom - kolom yang dinyatakan sebagai *Not Null*. 2). Data baru harus sesuai dengan tipe data untuk kolom yang dimaksud. 3). Dalam memanfaatkan kunci tamu (*foreign key*), harus menginputkan nilai – nilai absah pada table relasional. RDBMS melakukan pemeriksaan untuk menjamin ketentuan – ketentuan diatas dipenuhi. Ketika *insert* yang tidak absah dimasukkan, RDBMS akan memberikan kesalahan dan tidak mengijinkan perubahan.

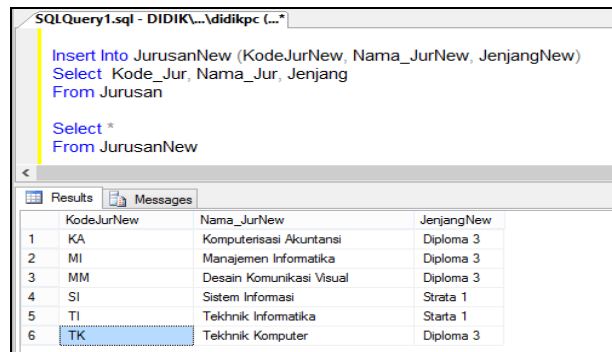
Sebagai contoh kita akan mengkopi (duplikasi) *table* baru dengan nama JurusanNew hasil copy dari table Jurusan pada database Jadwal Mengajar, Membuat *table* baru dengan nama *table* JurusanNew.



Sumber : Hasil Penelitian (2019)

Gambar 11. Table JurusanNew

Setelah itu buat *sintaks query* berikut ini untuk melakukan copi data dari *table* Jurusan yang dipindahkan datanya ke JurusanNew.



Sumber : Hasil Penelitian (2019)

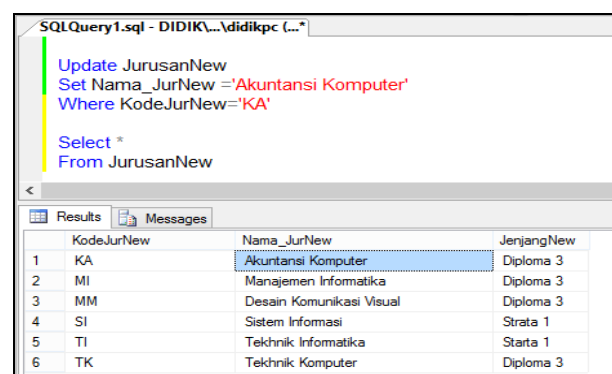
Gambar 12. *Insert data* JurusanNew dari *table* Jurusan

Pernyataan DML Update, kalimat *Update* memungkinkan kita memodifikasi satu nilai kolom atau lebih untuk data *table* yang telah ada. Perubahan data yang kita lakukan bisa dalam baris tertentu di *table* atau lebih. Sintaks dari kalimat *Update* tersebut adalah sebagai berikut:

```
Update table_name
Set column_1 = value_1
[,column_2 = value_2 [,...]] ..... (4)
Where condition
```

Sebagaimana kalimat *Insert*, beberapa batasan juga diterapkan terhadap kalimat *Update* dan juga menjadi sasaran batasan – batasan pada *Insert*. Pernyataan *where* bisa kita gunakan untuk melakukan pembatasan baris – baris yang akan dilakukan perubahan datanya. Ini memungkinkan kita menerapkan perubahan – perubahan secara selektif. *Update* memungkinkan kita memodifikasi kolom – kolom yang berisi kunci utama atau *foreign key* sepanjang hasil perubahan memenuhi *constraint* integritas yang diberlakukan. Misalkan kita akan melakukan perubahan data pada tabel JurusanNew pada gambar 7 diatas:

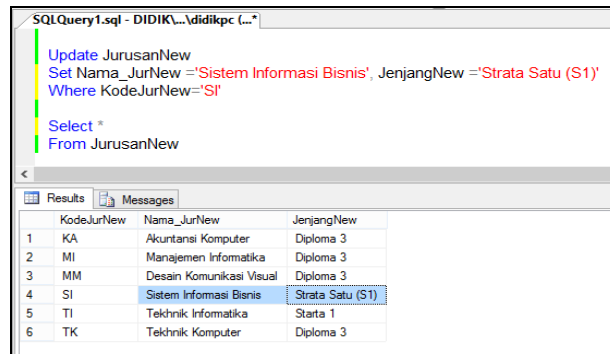
Update data dari 1 kolom, lakukan perubahan data nama jurusan 'Komputerisasi Akuntansi' menjadi 'Akuntansi Komputer' pada kode jurusan 'KA'.



Sumber : Hasil Penelitian (2019)

Gambar 13. *Update data* JurusanNew 1 kolom

Update data lebih dari 1 kolom, lakukan perubahan data nama jurusan 'Sistem Informasi' menjadi 'Sistem Informasi Bisnis', jenjang menjadi 'Strata Satu (S1)' pada kode jurusan 'SI'.



Sumber : Hasil Penelitian (2019)

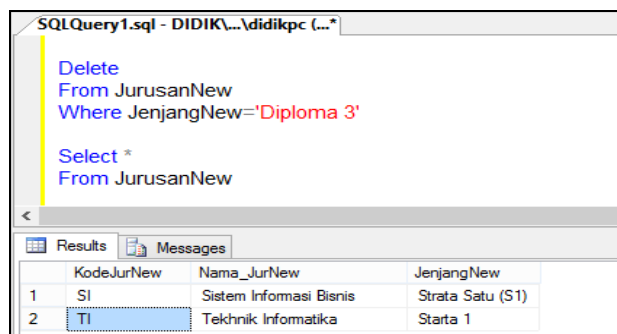
Gambar 14. Update data JurusanNew lebih dari 1 kolom

Pernyataan DML Delete, kalimat *Delete* memungkinkan kita menghapus satu *record* atau lebih pada suatu *table* dalam database. Sintaks dari kalimat *Delete* tersebut adalah sebagai berikut:

Delete From table_name..... (5)
Where condition

Pernyataan *Delete* dapat digunakan apabila perubahan – perubahan yang ditimbulkan tidak melanggar integritas dari data yang telah ada. **Table_name**, merupakan nama tabel yang akan dilakukan penghapusan data. Semua data akan terhapus jika kita tidak menggunakan pernyataan *Where* dalam melakukan penghapusan data.

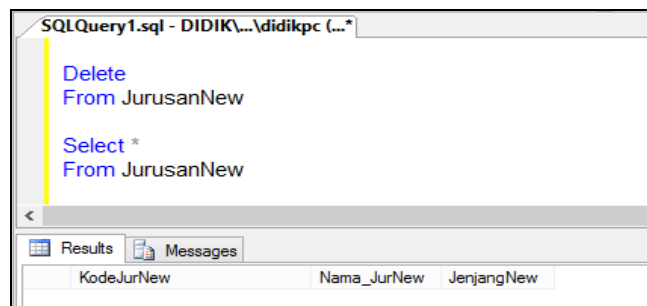
Delete baris (*record*) tertentu, misalkan kita akan menghapus record tertentu dimana Jenjang nya ‘Strata 1’, pada table JurusanNew yang telah kita buat dengan cara duplikasi (*copy*) sebelumnya, maka perintahnya adalah:



Sumber : Hasil Penelitian (2019)

Gambar 15. Delete baris (*record*) tertentu

Delete seluruh baris (*record*), misalkan kita akan menghapus seluruh *record* pada table JurusanNew yang telah kita buat dengan cara duplikasi (*copy*) sebelumnya, maka perintahnya adalah:



Sumber : Hasil Penelitian (2019)

Gambar 16. Delete Seluruh Baris (*record*)

4. Kesimpulan

Berdasarkan hasil implementasi SQL dengan menggunakan *query editor* pada SQL Server 2008 dari pernyataan DML *select*, *insert update* dan *delete* dari basis data *Penjadwalan Mengajar* yang telah dilakukan, dapat ditarik beberapa kesimpulan: 1). Implementasi pernyataan DML yang menggunakan RDBMS *SQL Server 2008* dapat dijalankan berdasarkan data dan struktur table *Penjadwalan Mengajar* yang terdiri dari tabel Dosen, Matakuliah, Jurusan dan Mengajar. yang telah dibuat sebagai dasar dalam melakukan DML (*select*, *insert update* dan *delete*). 2). Dalam pernyataan DML *Select* kita dapat melakukan pengambilan data dari 1 (satu) atau lebih tabel yang ada dengan berbagai kriteria yang diinginkan. 3). Pada pernyataan DML *insert* dapat dilakukan dengan melakukan pengisian data kedalam *record* pada tabel yang diinginkan dengan pengisian data seluruh kolom atau kolom tertentu. 4). Dalam pernyataan DML *update* dapat dilakukan dengan melakukan perubahan data pada satu atau lebih kolom yang akan dilakukan perubahan datanya. 5). Sedangkan pada pernyataan DML *delete* dapat dilakukan penghapusan data dari *record* tertentu berdasarkan kriteria *where* atau melakukan penghapusan seluruh record yang akan dihapus datanya.

Referensi

- Indrajani. 2018. Database Design All in One (Theory, Practice and Case Study). Jakarta: PT Elex Media Komputindo.
- Jayanti, Ari, Dewi, Ketut N, Sumiari, Kadek N. 2018. Teori Basis Data. Yogyakarta: Andi.
- Nuryana M, Sulistiyono. 2014. Analisa dan Perancangan Sistem Front Office Hotel Bidakara Serang. *Protekinfo* 1: 1–5.
- Sutejo B. 2010. Sukses Sertiikasi Oracle Certified Professional (OCP) Exam 1Z0-007 @ Exam 1Z0-051. Jakarta: PT Elex Media Komputindo.
- Utami E, Sukrisno. 2008. Mengoptimalkan Query pada Microsoft SQL Server. Yogyakarta: Andi.
- Widodo, Agus W, Kurnianingtyas D. 2017. NoSistem Basis Data. Malang: UB Press.