**MODUL III**

**DML (Data Manipulation Language)**

## TUJUAN

* + 1. Mengetahui dan memahami penggunaan DML dalam MySQL
    2. Mengetahui perbedaan beberapa perintah-perintah DML.
    3. Mampu mengimplementasikan DML dalam pembuatan suatu database.

## DASAR TEORI

## Pengertian DML

## Data Manilation 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.

## Perintah-perintah dalam DML

## SELECT

## Select merupakan perintah dalam DML yang digunakan untuk mengambil data dari database

## Bentuk umum penggilan SELECT adalah

|  |
| --- |
| SELECT {\*|daftar\_kolom} from nama\_tabel [where kondisi\_where] |

## Keterangan :

## \* digunakan untuk menampilkan semua kolom yang ada `dalam tabel

## Daftar\_kolom digunakan untuk menampilkan kolom tertentu saja

## Where digunakan jika ingin membatasi data yang ditampilkan

## Contoh

|  |
| --- |
| Select\*from member; |

## Menhasilkan :

## 

## Gambar 3.1 tampilan select

## INSERT

## INSERT merupakan perintah yang digunakan untuk menambahkan data ke suatu tabel.

## Bentuk umum pemanggilan insert adalah:

|  |
| --- |
| INSERT [INTO] nama\_tabel [(nama\_kolom1[,nama\_kolom2,….])] {value | values} (isi\_kolom1[,isi\_kolom2, …]) |

## Keterangan :

## Into bersifat opsional. Boleh ditulis boleh tidak.

## Daftar nama kolom boleh ditulis, boleh tidak. Jika ditulis, maka dianggap sesuai urutan nama kolom dalam pendefinisian table (lihat dengan perintah desc namatabel. Gunakan koma sebagai pemisah)

## {value|values}, wajib ditulis. Boleh value atau values.

## Isi kolom harus sesuai dengan urutan daftar nama kolom. Boleh diisi ekspresi atau default. Gunakan koma sebagai pemisah.

## DELETE

## DELETE merupakan perintah yang digunakan untuk melakukan penghapusan data dalam sebuah tabel

## Bentuk umum pemanggilan delete adalah:

|  |
| --- |
| delete from namatabel; |

## Contoh :

|  |
| --- |
| Delete from member; |

## Maka akan menampilan empty set atau tabel kosong.

## REPLACE

## Replace bekerjta seperti insert. Perbedaannya adalah jika terjadi error karena duplicate key, maka data lama akan dihapus dan data baru diinsertkan.

## Bentuk umum pemanggilan replace adalah:

|  |
| --- |
| Replace [into] namatabel [(namakolom1[,namakolom2, …])] {value | values} (isikolom1[,isikolom2, …]) |

## Keterangan:

## Pemanggilannya mirip dengan insert, begitu juga aturan pengisian datanya.

## Contoh:

|  |
| --- |
| create table demoreplace(id int primary key, nama char(10)); insert into demoreplace values(1,'AA'),(2,'BB'),(3,'CC'),(4,'DD');select \* from demoreplace; |

## Hasil :

## 

## Gambar 3.2 hasil replace

## TRUNCATE

## Truncate merupakan perintah yang digunakan untuk melakukan penghapusan semua baris mirip seperti delete.

## Bentuk pemanggilan truncate adalah:

|  |
| --- |
| Truncate [tabel] namatabel; |

## Perbedaan antara truncate dan delete

## Jika penghapusan menggunakan delete maka nilai auto increment tetap berlaku dan akan digunakan untuk insert berikutnya. Tetapi jika menggunakan truncate, maka nilai auto increment akan Kembali ke nilai awal.

## Penghapusan dengan delete dilakukan dengan melakukan penghapusan per baris, tetapi truncate akan melakukan penghapusan dengna drop tabel dan kemudian create tabel baru. Hal ini akan menyebabkan eksekusi truncate akan lebih cepat dibandingkan dengan delete.

## Contoh:

|  |
| --- |
| Create table demotruncate( id int primary key auto\_increment, nama varchar(5));Insert into demotruncate(nama) values('aa'),('bb'),('cc'); Truncate table demotruncate; Insert into demotruncate (nama) values('dd'); |

## UPDATE

## Update merupakan perintah yang digunakan untuk mengisi data lama dengan data baru.

## Bentuk umum pemanggilan update adalah:

|  |
| --- |
| UPDATE nama\_tabel  SET kolom1={isi|DEFAULT} [, kolom2={expr2|DEFAULT}] ...]  [WHERE kondisi\_where]  [ORDER BY nama\_kolom\_pengurutan]  [LIMIT banyak\_baris] |

## Keterangan:

## Where digunakan untuk membatasi banyaknya barus yang diupdate

## Order by digunakan untuk melakukan pengurutan data yang akan diupdate

## Limit digunakan untuk membatasi baris yang akan diupdate (setelah data dibatasi dengan where).

## Contoh :

|  |
| --- |
| Update member set naam=upper (nama); |

## Akan menampilkan update seluruh data karena tidak menggunakan batas.

## STUDI KASUS

## Daftar Persediaan Barang di Gudang

## DATA HASIL

## Entity Reletionship Diagram

## 

Gambar 3.3 ERD

1. **Jobsheet**

|  |  |  |  |
| --- | --- | --- | --- |
| No | | Latihan | Query dan Hasil |
| 1 | Membuat perintah SQL menampilkan salah satu tabel pada database | | Select\*from Latihan; |
|  |
| Pada percobaan pertama ini yaitu membuat perintah SQL menampilkan salah satu tabel pada database dengan sintax Select\*from nama\_tabel; , pada gambar di atas tampil Empety yang berarti tabel tersebut masih default kosong belum diisi; |
| 2 | Membuat perintah SQL memasukkan data minimal 3 data pada salah satu tabel | | insert into latihan values("Ahmad Satriadi","F1B020010","Teknik Elektro");  insert into latihan values("Wilad Surya Mandala","F1B020148","Teknik Elektro");  insert into latihan values("Nurul Watoni ","F1B020125","Teknik Elektro"); |
|  |
| Pada percobaan kedua yaitu membuat perintah SQL memasukkan data minimal 3 data pada salah satu tabel dengan sintax insert into nama\_tabel values(value); , pada gambar diatas tampil query ok yang berarti memasukkan data berhasil, |
| 3 | Membuat perintah SQL menghapus data minimal 1 data | | delete from latihan where NIM="F1B020125"; |
|  |
| Pada percobaan ketiga yaitu membuat perintah SQL menghapus data minimal 1 data dengan sintax delete from nama\_tabel where nama\_kolom\_primarykey=isi\_kolom;, dengan sintax tersebut akan menghapus data dengan berpatokan pada primarykey. Pada query daiatas data yang akan dihapus yaitu yang memiliki primarykey “F1B020125”. |
| 4 | Membuat perintah SQL mengubah data minimal 1 data menggunakan query REPLACE | | replace into latihan values ("Wilad Sampurna", "F1B020148", "Teknik Elektro"); |
|  |
| Pada percobaan keempat yaitu membuat perintah SQL mengubah data minimal 1 data menggunakan query REPLACE dengan sintax replace into nama\_tabel values (value);, dengan sintax tersebut maka akan mengganti(replace) data pada tabel yang diinginkan, perlu diingat untuk bisa mengganti suatu data pada tabel, harus memiliki primary key. Pada query diatas nama awalnya Wilad surya mandala diganti dengan Wilad Sampurna begitu juga dengan NIM dan Jurusan juga dirubah. |
| 5 | Membuat perintah SQL mengubah data minimal 1 data menggunakan query UPDATE | | update latihan set nama="Wilad Surya Mandala" where nama is "Wilad Sampurna"; |
|  |
| Pada percobaan kelima yaitu membuat perintah SQL mengubah data minimal 1 data menggunakan query UPDATE dengan sintax update nama\_tabel set nama\_kolom="value\_baru" where nama\_kolom is "value\_lama";, dengan sintax tersebut akan mengupdate isi yang lama menjadi isi atau data yang baru pada satu data, jika ingin mengupdate kolom keseluruhan selain primary key dengan sintax update nama\_tabel set nama\_kolom="value\_baru" tanpa menggunakan where maka satu kolom semua sama, namun sintax ini tidak berlaku untuk kolom primary key. Pada query diatas mengubah nama wilad sampurna menjadi Wilad Surya Mandala pada tabel latihan. |

**3.5 KESIMPULAN**

1. penggunaan DML dalam MySQL yaitu untuk membantu pengguna memanipulasi data dalam database. Manipulasi ini dilakukan dengan memasukkan data ke dalam tabel database, mengambil data yang ada, kemudian menghapus data dari tabel yang ada, dan memodifikasi data yang ada.
2. Select digunakan untuk mengambil data dari database, INSERT digunakan untuk menambahkan data ke suatu tabel, DELETE digunakan untuk melakukan penghapusan data dalam sebuah tabel, Replace digunakan untuk merubah data, Truncate digunakan untuk melakukan penghapusan semua baris mirip seperti delete.
3. Megimplementasikan SELECT dengan sintax Select\*from nama\_tabel;, INSERT dengan sintax insert into nama\_tabel values(value);, UPDATE dengan sintax update nama\_tabel set nama\_kolom="value\_baru" where nama\_kolom is "value\_lama";, DELETE dengan sintax from nama\_tabel where nama\_kolom\_primarykey=isi\_kolom;.

**DAFTAR PUSTAKA**

Andri Heryandi, 2010. SISTEM BASIS DATA. Teknik Informatika Universitas Komputer Indonesia

Anonim, 2022. PRAKTIKUM BASIS DATA. Laboratorium Komputer dan Jaringan, Fakultas Teknik Universitas Mataram.

Didik Setiyadi, Fata Nidaul Khasanah, Henderi, 2019. DATA MANIPULATION LANGUAGE (DML) DATABASE PENJADWALAN DOSEN MENGGUNAKAN SQL SERVER 2008. Teknik Informatika Universitas Bina Insani.

Fitria Nurhasanah, Rahmania Sri Untari, 2019. BUKU AJAR BASIS DATA. Universitas Muhammadiyah Sidoarjo