

## Modul 1: MySQL – Data Manipulation Language (DML) - INSERT

### Topik

Penggunaan SQL Statement INSERT pada DBMS MySQL.

### Tujuan

Setelah mempelajari modul ini, mahasiswa diharapkan dapat:

1. Memahami penggunaan SQL statement INSERT.
2. Menambahkan data ke dalam tabel menggunakan berbagai format INSERT.

### Pendahuluan

DML merupakan istilah untuk beberapa sintaksis (syntax) dari SQL yang digunakan untuk melakukan perubahan pada data (isi tabel-tabel) dalam suatu database. DML terdiri dari 3 klausa utama yaitu:

1. **INSERT** : Menambah baris baru pada sebuah tabel
2. **UPDATE** : Mengubah nilai suatu baris pada sebuah tabel.
3. **DELETE** : Menghapus suatu baris dari sebuah tabel.

Pada DML terdapat dua jenis bahasa, yaitu :

#### 1. High-Level(Non\_procedural) DML.

- Digunakan secara interaktif (interpreter)
  - Dapat dijadikan satu dengan general purpose programming language (embedded)
- High-Level DML yang biasa digunakan secara interaktif disebut "Query Language".

#### 2. Low-Level(Proedural) DML.

Digunakan secara embedded dalam suatu general purpose programming language

Bilamana kedua jenis DML diatas digunakan secara "embedded", maka : bahasa pemrograman yang digunakan disebut sebagai "Host Language" dan "DML-nya disebut "Sub Language"

### Operasi INSERT

Operasi insert bertujuan untuk menyisipkan satu tuple baru ke dalam suatu relasi R.

Klausa pembentuk:

1. **INSERT**
2. **INTO**
3. **VALUES**

Format:

1. **INSERT INTO** nama\_tabel (kolom1, kolom2, ...dst.) **VALUES** (nilai\_kolom1, nilai\_kolom2, ...dst.);
2. **INSERT INTO** nama\_tabel **VALUES** (nilai\_kolom1, nilai\_kolom2, ...dst.); 3. [Salah satu dari kedua format sebelumnya], (nilai\_kolom\_kolom\_baris1), (nilai\_kolom\_kolom\_baris2), ...dst.

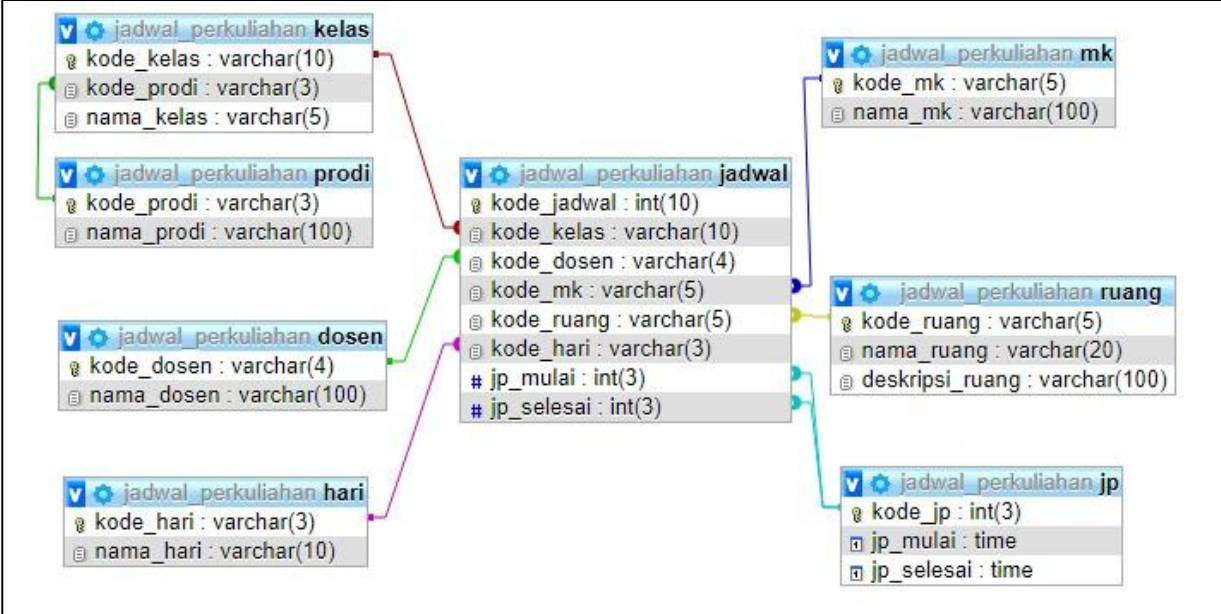
Operasi ini memungkinkan untuk melanggar empat jenis constraint sebagaimana dijelaskan berikut ini :

1. **DOMAIN Constraint** dapat dilanggar jika suatu nilai attribute yang diberikan tidak ada dalam domain yang berkorespondensi dengan attribute tadi.
2. **KEY Constraint** dapat dilanggar jika nilai key dalam tuple baru t sudah ada dalam tuple lain dalam relasi  $r(R)$ .
3. **ENTITY INTEGRITY** Constraint dapat dilanggar jika primary key dari tuple baru t adalah NULL
4. **REFERENTIAL INTEGRITY** Constraint dapat dilanggar jika nilai dari suatu foreign key dalam t mengacu ke suatu tuple yang tidak ada dalam relasi yang diacu.

Ada dua pilihan tindakan yang dapat dilakukan jika ada satu atau lebih constraint yang dilanggar akibat operasi insert, yaitu :

1. Menolak (reject) operasi insertion. Biasanya DBMS memberikan penjelasan mengapa proses insertion ditolak.
2. Berusaha memperbaiki alasan penolakan proses insertion. Dimana insertion akan diterima jika user melakukan perubahan nilai-nilai attribute sehingga insertion diterima.

## Praktikum – Bagian 1: Percobaan Statement INSERT

Langkah	Keterangan
	<p>Perhatikan skema/model relasional/EER diagram dari database berikut.</p> 
1.	<p>Untuk menambahkan data (mengisi) suatu tabel, digunakan statement (pernyataan) <b>INSERT</b>. Eksekusi SQL berikut untuk menambahkan 1 baris (record) baru pada tabel <b>mk</b>.</p> <pre data-bbox="358 1249 1555 1318">INSERT INTO mk (kode_mk, nama_mk) VALUES ('02010', 'Basis Data');</pre> <p>Statement SQL tersebut menambahkan 1 baris baru ke tabel <b>mk</b> pada kolom yang dinyatakan di dalam tanda kurung ( ) pertama.</p> <p>Untuk melihat hasil SQL yang kita eksekusi tersebut, gunakan statement <b>SELECT</b> seperti berikut. Pembahasan lebih lengkap mengenai SELECT dijadwalkan untuk disampaikan pada pertemuan berikutnya, namun secara umum, statement SELECT digunakan untuk menyajikan recordrecord yang ada pada suatu tabel. Karakter * akan menampilkan isi dari semua kolom yang ada pada tabel.</p> <pre data-bbox="358 1648 701 1717">SELECT * FROM mk</pre>

	<p>Apabila data di-insert-kan pada semua kolom tabel, maka kita dapat langsung menggunakan klausa VALUES tanpa harus menuliskan nama-nama kolom dahulu.</p>
2.	<div data-bbox="358 344 1286 432" data-label="Text"> <pre>INSERT INTO mk VALUES('02041', 'Teknologi Data');</pre> </div> <p>Statement SQL tersebut menambahkan 1 baris baru ke tabel <b>mk</b> pada kolom yang ada pada struktur tabel <b>mk</b> tanpa menyebutkan nama kolomnya.</p> <p>Untuk melihat hasil SQL yang kita eksekusi tersebut, gunakan statement <b>SELECT</b> seperti berikut.</p> <div data-bbox="358 646 701 711" data-label="Text"> <pre>SELECT * FROM mk</pre> </div>
3.	<p>Untuk menambahkan beberapa kolom sekaligus dalam 1 statement digunakan statement dengan format seperti berikut.</p> <div data-bbox="358 869 1302 1031" data-label="Text"> <pre>INSERT INTO mk VALUES ('02004', 'Aljabar Linier'), ('02005', 'Analisis Dan Desan Berorientasi Objek'), ('02006', 'Bahasa Indonesia');</pre> </div> <p>Statement SQL tersebut menambahkan 3 baris baru ke tabel <b>mk</b> pada kolom yang ada pada struktur tabel <b>mk</b> tanpa menyebutkan nama kolomnya.</p> <p>Untuk melihat hasil SQL yang kita eksekusi tersebut, gunakan statement <b>SELECT</b> seperti berikut.</p> <div data-bbox="358 1262 701 1327" data-label="Text"> <pre>SELECT * FROM mk</pre> </div>
	<p>Dan seperti berikut, jika hanya kolom tertentu saja yang akan diberi nilai dengan cara menyebutkan nama kolomnya.</p>
4.	<div data-bbox="358 1484 1164 1661" data-label="Text"> <pre>INSERT INTO mk (kode_mk, nama_mk) VALUES ('02001', 'Agama'), ('02002', 'Aljabar Linier'), ('02003', 'Algoritma dan Struktur Data');</pre> </div> <p>Statement SQL tersebut menambahkan 3 baris baru ke tabel <b>mk</b> pada kolom yang ada pada struktur tabel <b>mk</b>.</p> <p>Untuk melihat hasil SQL yang kita eksekusi tersebut, gunakan statement <b>SELECT</b> seperti berikut.</p>

	<pre>SELECT * FROM mk</pre>
5.	<p>Statement INSERT juga dapat dieksekusi dengan menggunakan klausa SET alih-alih VALUES.</p> <pre>INSERT INTO mk SET   kode_mk = '02012' ,   nama_mk = 'Desain Pemrograman Web' ;</pre> <p>Statement SQL tersebut menambahkan 1 baris baru ke tabel <b>mk</b> pada kolom yang ada pada struktur tabel <b>mk</b>. Untuk melihat hasil SQL yang kita eksekusi tersebut, gunakan statement <b>SELECT</b> seperti berikut.</p> <pre>SELECT * FROM mk</pre>
6.	<p>Pada statement INSERT juga dapat digunakan klausa SELECT. Misalnya kita ingin menyalin semua baris pada tabel <b>mk</b> ke tabel <b>mk_backup</b>, maka kita dapat menggunakan SQL berikut. (<b>Buat terlebih dahulu tabel “mk_backup” dengan struktur tabel yang sama dengan tabel “mk”</b>)</p> <pre>CREATE TABLE mk_backup (   kode_mk VARCHAR(5),   nama_mk VARCHAR(100) );</pre>
7.	<p>Kemudian masukkan data dari tabel <b>mk</b> ke tabel <b>mk_backup</b></p> <pre>INSERT INTO mk_backup   SELECT * FROM mk;</pre> <p>Statement SQL tersebut menambahkan data baru dari tabel <b>mk</b> ke tabel <b>mk_backup</b>. Untuk melihat hasil SQL yang kita eksekusi tersebut, gunakan statement <b>SELECT</b> seperti berikut.</p> <pre>SELECT * FROM mk_backup;</pre>
8.	<p>Setelah berhasil mengeksekusi SQL tersebut, lanjutkan ke <b>SubTopik</b> selanjutnya.</p>