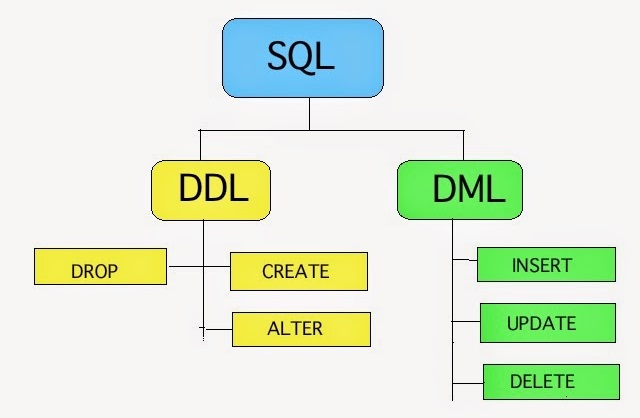
PENGANTAR SQL

1. Definisi SQL

SQL (atau query languange) adalah sebuah bahasa yang digunakan untuk merequest informasi dari basis data; bahasa ini sudah disediakan oleh DBMS. Konstruksi SQL terbentuk atas kombinasi relational-algebra dan relational-calculus.

SQL terdiri dari beberapa komponen, dua di antaranya yang sangat fundamental adalah DDL : memanipulasi definisinya (Data Definition Language) dan DML : memanipulasi datanya (Data Manipulation Language). Lalu apakah perbedaan antara DDL dan DML?



1. DDL

DDL menyediakan perintah-perintah untuk menambah, mengubah dan menghapus data dalam basis data. Contoh perintah DDL adalah CREATE, ALTER, DROP

1. Menambah/menciptakan data data

Untuk menambah data dalam DDL menggunakan perintah CREATE diikuti oleh nama database.

Berikut adalah contoh perintah untuk menciptakan database dengan nama “RPL”

CREATE DATABASE RPL;

Berikut adalah contoh perintah untuk menciptakan tabel dengan nama siswa yang diikuti dengan nama field

CREATE TABLE siswa (

nis CHAR(12) NOT NULL,

nama VARCHAR(60) NOT NULL,

alamat VARCHAR(120),

PRIMARY KEY(nis)

);

**Latihan 1**

Sekarang, coba tuliskan perintah untuk

1. Membuat database dengan nama kelas beserta no. absen (contoh XIRPL3\_20).
2. Membuat tabel siswa dengan field nis, nama dan alamat.
3. Memodifikasi data

Untuk mengubah data dalam DDL menggunakan perintah ALTER.

Berikut adalah contoh mengubah nama tabel dari RPL menjadi RPL1

ALTER TABEL RPL rename RPL1;

Selain itu, perintah ALTER juga dapat digunakan untuk memodifikasi data, misalnya untuk menambah field dalam suatu tabel.

Berikut adalah contoh perintah untuk menambah field jenis kelamin pada tabel siswa

ALTER TABLE siswa ADD jk CHAR(2);

Untuk menghapus field dapat menggunakan perintah sebagai berikut:

ALTER TABLE nama\_tabel DROP COLUMN nama\_field

Contoh berikut akan menghapus field jk dari tabel.

ALTER TABLE siswa DROP COLUMN jk;

**Latihan 2**

Sekarang, coba tuliskan perintah untuk:

1. Mengubah nama tabel siswa menjadi student.
2. Menambahkan field no\_hp pada tabel siswa.
3. Menambahkan field tgl\_lahir pada tabel siswa dimana field tersebut berada pada posisi setelah field nama.
4. Hapus field no\_hp pada tabel siswa.
5. Menghapus data

Penghapusan data dapat menggunakan perintah DROP

Berikut contoh perintah untuk menghapus database:

DROP DATABASE RPL;

1. DML

DML menyediakan perintah-perintah untuk menambah, memodifikasi, menghapus, dan mengambil data di dalam basis data.

2Contoh perintah DML adalah : INSERT, UPDATE, DELETE, dan SELECT

1. Penambahan Data

Sintaks operasi penambahan data dapat dituliskan sebagai berikut:

INSERT INTO r (A1, A2, …, An)

VALUES (V1,V2,…., An)

Dimana :

r adalah relasi (tabel)

A adalah atribut (field atau kolom)

V adalah value (nilai yang diisikan)

Misalkan tabel pelajaran memiliki field id dan nama\_pel, maka pernyataan SQL untuk penambahan data dituliskan sebagai berikut :

INSERT INTO pelajaran (id,nama\_pel)

VALUES (V1,V2,…., An)

**Modifikasi Data**

Sintaks operasi modifikasi data dapat dituliskan sebagai berikut :

UPDATE r

SET A1-V1, A2-V2, …, An-Vn

WHERE P

Di mana :

P adalah predikat

Misalkan nilai field nama\_pel untuk data dengan id 1 ingin diubah menjadi Basis Data, berikut pernyataan SQL-nya:

UPDATE pelajaran

SET nama\_pel = “Basis Data”

WHERE id = 1

**Penghapusan Data**

Sintaks operasi penghapusan data dapat dituliskan sebagai berikut :

DELETE FROM r

WHERE P

Misalkan nilai data dengan id ingin dihapus, penyataan SQL-nya dituliskan

sebagai berikut :

DELETE FROM pelajaran

WHERE id = 1

Perhatikan juga, operasi penghapusan tanpa menyertakan klausa WHERE akan

mengakibatkan dihapusnya seluruh data di table. Jadi, jika Anda hanya ingin

menghapus data tertentu gunakan selalu klausa WHERE.

**Seleksi Data**

Sintaks operasi seleksi atau pengambilan data dapat dituliskan sebagai berikut :

SELECT A1, A2, …, An

FROM r1, r2, r3, …, rn

WHERE P

Misalkan ingin menampilkan data id dan nama\_pel, pernyataan SQL-nya

dituliskan sebagai berikut :

SELECT id, nama\_pel

FROM pelajaran

Sebagai alternative, tersedia karakter asterik (\*) yang menyatakan seluruh field.

Jadi, penyataan di bawah ini ekuivalen dengan pernyataan di atas.

SELECT \*

FROM pelajaran

**Latihan 3**

Buatlah perintah DML untuk

1. Membuat tabel guru yang memiliki field id\_guru, nama\_guru, dan alamat beserta isi datanya seperti tabel di bawah ini.

|  |  |  |
| --- | --- | --- |
| ID Guru | Nama Guru | Alamat |
| 1 | Nina | Jl. Bandung |
| 2 | Sari | Jl. Jakarta |
| 3 | Toni | Jl. Jombang |
| 4 | Supri | Jl. Jakarta |

1. Ubahlah nama guru yang id\_gurunya 2 dengan nama “Rani”.
2. Seleksi data pada tabel guru yang beralamat di Jl. Jakarta.
3. Hapus data pada tabel guru yang id\_gurunya adalah 1.

**Seleksi dengan Operator LIKE**

Operator LIKE memungkinkan pengambilan data yang mirip atau mendekati kata kunci.

Contoh : dapatkan data pelajaran yang nama pelajarannya berawalan kata “Sistem”.

SELECT \* FROM pelajaran WHERE nama\_pel LIKE “Sistem%”;

**Pengurutan Data**

Untuk mengurutkan data, kita memanfaatkan klausa ORDER BY, dimana nilai

defaultnya adalah ASC (*ascending* atau pengurutan menaik).

Contoh:

1. Urutkan data pelajaran berdasarkan nama pelajaran dengan urutan menaik.

SELECT \* FROM pelajaran ORDER BY nama\_pel ASC;

1. Urutkan data pelajaran berdasarkan nama pelajaran dengan urutan menurun

SELECT \* FROM pelajaran ORDER BY nama\_pel DESC;

**Latihan 4**

Tuliskan perintah DML untuk

1. Dapatkan data guru yang namanya mengandung kata “Supri”.
2. Dapatkan data guru yang alamatnya berakhiran kata “Jombang”
3. Urutkan data guru berdasarkan id\_guru secara menurun.
4. Urutkan data guru berdasarkan nama\_guru secara menaik.