

Modul II

Perintah SELECT dan ORDER BY

A. Kompetensi Dasar:

1. Mahasiswa dapat merumuskan perintah select untuk menampilkan data dengan benar.
2. Mahasiswa dapat menyusun perintah Select dengan berbagai kriteria.

B. Teori Singkat :

1. Bentuk dasar (sederhana):

SELECT T1.A1 [T1.A2, ...,T1.An], **FROM** T1 [, T2, ...,Tn][**WHERE** T1.Ax=T2.Ax] **ORDER BY**/ **GROUP BY**

keterangan :

- a. **SELECT** : untuk menampilkan data dari satu atau beberapa tabel.
 - b. **A1, ..., An** : daftar kolom yang ingin ditampilkan. Jika ingin menampilkan semua atribut cukup dituliskan dengan tanda bintang (*)
 - c. **T1, ..., Tn** : daftar tabel
 - d. **kondisi** : syarat data yang akan ditampilkan
 - e. **[]** : tanda opsional (boleh digunakan boleh tidak)
2. **SELECT** dengan opsi **DISTINCT**, digunakan untuk menampilkan baris-baris dari suatu tabel, tetapi jika ada baris yang nilainya sama maka hanya akan ditampilkan satu kali.
 - a. **Sintaks** : **SELECT DISTINCT** nama_kolom **FROM** nama_tabel
 - b. **Ketr.** : nama_kolom adalah kolom yang dievaluasi, ada nilai yang sama atau tidak. Opsi **DISTINCT** dapat digunakan bersama-sama dengan opsi/ekspresi yang lain misalnya **WHERE**, **GROUP BY**, **HAVING** atau **ORDER BY**
 - c. **Contoh** :
 - ▲ **SELECT DISTINCT** Kode_Prodi **from** Mahasiswa
 - ▲ **SELECT DISTINCT** Lokasi **from** Fakultas **ORDER BY** Lokasi
 3. **SELECT** dengan opsi **AS**, digunakan untuk mengganti tampilan judul kolom tanpa mengubah nama kolom
 - a. **Sintaks** : **SELECT** A1 **AS** judul1 [, A2 **AS** judul2, ...] **FROM** nama_tabel
 - b. **Ketr.** :
 - ▲ A1... An adalah nama kolom yang akan ditampilkan
 - ▲ judul1, ..., judul N adalah judul baru yang akan menggantikan nama kolom
 - c. **Contoh** :
 - ▲ **SELECT** kode_prodi **AS** KODE PRODI, nama_prodi **AS** PROGRAM STUDI, kode_fak **AS** KODE FAK **from** PRODI
 4. **SELECT** dengan klausa **WHERE**, digunakan untuk menampilkan data-data berdasarkan kriteria/kondisi tertentu.
 - a. **Keterangan** : Menampilkan semua data dari sebuah tabel tidak selamanya menarik, bahkan sering-sering harus dihindari. Untuk memilih data tertentu, harus ditentukan kondisi yang diperlukan. Misalnya, untuk menampilkan daftar semua fakultas yang ada di kampus III, maka kondisinya adalah Lokasi = 'Kampus III'.
 - b. **Contoh** :
 - ▲ Untuk menampilkan daftar mahasiswa yang kode_prodi nya 016, perintah SQLnya :
SELECT * FROM Mahasiswa **WHERE** kode_prodi = 'P01'
 - ▲ Untuk menampilkan daftar fakultas yang Lokasi-nya adalah Kampus III :
SELECT * FROM Fakultas **WHERE** Lokasi = 'Kampus III'
 5. Operator **LIKE**, untuk atribut yang bertipe string dapat dilakukan pencarian dengan pola tertentu, dengan memanfaatkan karakter persen '%' dan atau garis bawah '_', serta dengan menambahkan operator **LIKE** pada klausa **WHERE**. **Contoh** :

- a. Untuk menampilkan data mahasiswa yang nama depannya Andi, maka kriterianya bisa dituliskan Nama LIKE 'Andi%', sehingga perintah SQLnya :
SELECT * FROM Mahasiswa where Nama LIKE 'Andi%'
 - b. Untuk menampilkan data mahasiswa yang namanya mengandung deretan string Nak, maka kriterianya bisa dituliskan Nama LIKE '%Nak%', sehingga perintah SQLnya :
SELECT * FROM Mahasiswa where Nama LIKE '%Nak%'
 - c. Untuk menampilkan data mahasiswa yang huruf keduanya 'a', maka kriterianya bisa dituliskan Nama LIKE '_a%', sehingga perintah SQLnya :
SELECT * FROM Mahasiswa where Nama LIKE '_a%'
6. Kondisi JAMAK . Kriteria pencarian dapat disusun dari beberapa kondisi yang dihubungkan dengan operator logika AND , BETWEEN atau OR. Contoh :
- a. Untuk menampilkan data buku yang huruf keduanya 'a', dan diterbitkan oleh Gava Media, maka perintah SQLnya :
SELECT * FROM Buku where Judul LIKE '_a%' AND Penerbit='Gava Media'
 - b. Untuk menampilkan data peminjaman buku yang tanggal harus kembalinya antara 01 Maret 2005 sampai 30 Maret 2007, maka perintah SQLnya :
SELECT * FROM Pinjam WHERE tgl_hrs_kbl BETWEEN '2007-04-19' AND '2007-04-22'
 - c. Untuk menampilkan data fakultas yang lokasinya di Kampus I atau Kampus III, maka perintah SQLnya :
SELECT * FROM Fakultas WHERE Lokasi = 'Kampus I' or Lokasi = 'Kampus III'

C. Langkah Praktikum :

- 1. Hapuslah database DIGILAB yang telah Anda buat pada praktikum I ! dengan perintah
DROP DATABASE DIGILAB ;
- 2. Buatlah database baru dengan nama DIGILAB, menggunakan perintah seperti pada modul II!
- 3. Buatlah tabel FAKULTAS dengan perintah :
create table FAKULTAS (kode_fak char(3) primary key, nama_fakultas varchar(30) NOT NULL,Lokasi varchar (20) NOT NULL);
- 4. Masukkan data dibawah ini pada table FAKULTAS
Contoh : insert into FAKULTAS values ('F01'FAK. MIPA','Kampus III');

kode_fak	Nama_fakultas	Lokasi
F02	Fak. Ekonomi	Kampus I
F03	Fak. Teknik	Kampus III
F04	Fak. Sastra	Kampus II
F05	Fak. Hukum	Kampus II
F06	Fak. Psikokogi	Kampus I

- 5. Buatlah tabel PRODI dengan perintah :
Create table PRODI (kode_prodi char(3) primary key, Nama_prodi varchar(30), kode_fak char(3), foreign key (kode_fak) references FAKULTAS (kode_fak));
- 6. Masukkan data dibawah ini pada tabel PRODI
Contoh : insert into PRODI values('P01', 'Ilmu Komputer','F01');

kode_prodi	Nama_prodi	Kode_fak
P02	Ekonomi	F02

P03	Teknik Industri	F03
P04	Sastra Inggris	F04
P05	Hukum	F05
P06	Psikologi	F06

7. Buatlah tabel MAHASISWA dengan perintah :
- CREATE TABLE Mahasiswa (nim CHAR(8) PRIMARY KEY, Nama VARCHAR(30), kode_prodi CHAR(3) , FOREIGN KEY (kode_prodi) REFERENCES prodi (kode_prodi));**
8. Masukkan data dibawah ini pada tabel MAHASISWA
- Contoh : insert into MAHASISWA values('04001001', 'Andi Herwanto','P01');

nim	Nama	Kode_prodi
04002001	Didik cahyono	P02
02003001	Fajar Yuda	P03
05004001	Yanti	P04
03005001	Tanakung	P05
01006001	Andrian	P06

9. Buatlah tabel BUKU dengan perintah :
- CREATE TABLE Buku (kode_buku CHAR(3) PRIMARY KEY, Judul VARCHAR(50), penulis VARCHAR (30), penerbit VARCHAR(30));**
10. Masukkan data dibawah ini pada tabel BUKU
- Contoh : insert into BUKU values('B01', 'basis data', 'Tawar AG, M.Si', 'Andi offset');

Kode_buku	Judul	Penulis	Penerbit
B02	Matematika	Sugiyarto M.Si	Gava Media
B03	Organisasi Komputer	Imam Riadi, SPd, M.Kom	Andi Offset
B04	Struktur data	Suprihatin S.Kom	Andi Offset
B05	Mikrobiologi	Hadi Sasaongko Msi	Elex Media
B06	Akutansi	Sugiyarto M.Si	Yudistira

11. Buatlah tabel PINJAM (nim, kode_buku, tgl_pjm, tgl_hrs_kbl, tgl_kbl, kembali) dengan perintah berikut
- CREATE TABLE pinjam (nim CHAR (8) , kode_buku CHAR (3) , tgl_pjm DATE, tgl_hrs_kbl DATE, tgl_kbl DATE, kembali BOOL, PRIMARY KEY(nim, kode_buku) , FOREIGN KEY (nim) REFERENCES mahasiswa(nim), FOREIGN KEY(kode_buku) REFERENCES buku(kode_buku));**

12. Masukkan data dibawah ini pada tabel PINJAM

Contoh :insert into PINJAM values('02003001', 'B02','2007-04-13','2007-04-20','2007-04-17','1');

Nim	Kode_buku	Tgl_pjm	Tgl_hrs_kbl	Thl_kbl	kembali
03005001	B03	2007-04-17	2007-04-24	2007-04-21	1
04001001	B03	2007-04-15	2007-04-22	2007-04-20	1
01006001	B06	2007-04-12	2007-04-19	2007-04-16	1

13. Tampilkan masing-masing isi tabel yang telah Anda buat!!!
14. Tampilkan data MAHASISWA dengan kriteria mahasiswa yang kode_prodi-nya “P05”
15. Tampilkan data nama MAHASISWA yang huruf keduanya ‘a’!
16. Tampilkan data FAKULTAS yang berlokasi dikampus I atau dikampus II !
17. Tampilkan daftar FAKULTAS yang diurutkan berdasarkan lokasinya
18. Tampilkan data BUKU dengan nama pengarang yang huruf pertamanya ‘S’, dan diterbitkan oleh Yudistira.

D. Tugas :

- 1. Jelaskan Perintah-perintah SQL yang anda praktekan secara mendetail
- 2. Tuliskan/ buatlah ringkasan perintah-perintah select

Note :

- 1. Perintah-perintah SQL dalam MySQL selalu diakhiri dengan tanda titik koma (;)
- 2. Perintah-perintah SQL dalam penulisan . (Titik) dan , (koma) harus diperhatikan
- 3. Perintah-perintah SQL dalam pembuatan Database, Table dan Nama kolom tidak boleh menggunakan Spasi (diganti dengan tanda _)
- 4. Perintah-perintah SQL dapat disimpan dalam file teks dan dijalankan dengan mengetikkan “\.**Path-lengkap**\nama_file_lengkap”,
Contoh : \. D:\tawar\sql\create.txt