

TEORI BASIS DATA

Pertemuan 12- DATA RETRIEVAL LANGUAGE (DRL)

Dosen Pengajar : Bapak Farid Angga Pribadi, S.Kom., M.Kom



Nama : Surya Rahmat Fatahillah

NIM : 2341760020

Prodi : Sistem Informasi Bisnis

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

2024

LATIHAN 2

1. Buatlah SQL untuk mengisi kolom sks pada tabel matakuliah! (Pastikan jumlah sks ada yang berbeda)
2. Buatlah SQL untuk mengisi kolom nilai _huruf dengan 3 record baru untuk mahasiswa yang mendapatkan nilai 'B+' pada tabel nilai!
3. Buatlah SQL untuk mencari kode matakuliah yang sudah diisi pada tabel nilai dan tabel matakuliah!
4. Buatlah SQL untuk mencari data yang tidak bernilai diantara C dan E!
5. Tampilkan semua kolom dalam tabel matakuliah yang mempunyai jumlah sks lebih dari jumlah sks yang dimiliki oleh nama mata kuliah Basis Data Dasar!

JAWABAN!

1. Berikut Query untuk mengisi kolom sks pada tabel matakuliah

```
UPDATE mata_kuliah
SET sks =
CASE
    WHEN kode = 'ASD' THEN 5
    WHEN kode = 'BDD' THEN 2
    WHEN kode = 'KCB' THEN 1
    WHEN kode = 'MMT' THEN 4
    WHEN kode = 'PBO' THEN 6
    WHEN kode = 'SPK' THEN 3
    ELSE sks
END
WHERE kode IN ('ASD', 'BDD', 'KCB', 'MMT', 'PBO', 'SPK');
```

Kemudian berikut hasilnya:

```
MariaDB [(none)]> use akademik_3;
Database changed
MariaDB [akademik_3]> select * from mata_kuliah;
```

kode	nama	sks
ASD	Algortima dan Struktur Data	5
BDD	Basis Data Dasar	2
KCB	Kecerdasan Buatan	1
MMT	Multimedia Terapan	4
PBO	Pemrograman Berorientasi Objek	6
SPK	Sistem Pendukung Keputusan	3

```
6 rows in set (0.000 sec)
```

Berikut adalah penjelasan langkah-langkahnya:

- **UPDATE mata_kuliah:** Mendefinisikan tabel yang akan diupdate, yaitu 'mata_kuliah'.
- **SET sks = ...:** Menentukan kolom 'sks' akan diupdate. Nilai yang akan diberikan pada kolom 'sks' bergantung pada kondisi yang ditentukan dalam CASE statement berikutnya.
- **CASE** digunakan untuk mengevaluasi setiap baris dalam tabel berdasarkan kondisi yang ditentukan, dan memberikan nilai yang sesuai sesuai dengan kondisi yang terpenuhi.
- **WHERE kode IN ('ASD', 'BDD', 'KCB', 'MMT', 'PBO', 'SPK');** Digunakan untuk memastikan bahwa update hanya dilakukan pada baris-baris di mana nilai kolom 'kode' adalah 'ASD', 'BDD', 'KCB', 'MMT', 'PBO', atau 'SPK'.

2. Berikut Query untuk mengisi kolom nilai_huruf dengan 3 record baru untuk mahasiswa yang mendapatkan nilai 'B+' pada tabel nilai

```
INSERT INTO nilai (NIM, kode_mata_kuliah, nilai_huruf)
SELECT NIM, 'ASD', 'B+'
FROM nilai
WHERE nilai_huruf = 'B+'
UNION ALL
SELECT NIM, 'BDD', 'B+'
FROM nilai
WHERE nilai_huruf = 'B+'
UNION ALL
SELECT NIM, 'MMT', 'B+'
FROM nilai
WHERE nilai_huruf = 'B+'
LIMIT 3;
```

Berikut hasilnya:

```
MariaDB [akademik_3]> select * from nilai;
```

id	NIM	kode_mata_kuliah	nilai_huruf
1	171401	KCB	B+
2	171402	ASD	A
3	171403	MMT	B
4	171404	PBO	A
5	171404	KCB	C

```
MariaDB [akademik_3]> select * from nilai;
```

id	NIM	kode_mata_kuliah	nilai_huruf
1	171401	KCB	B+
2	171402	ASD	A
3	171403	MMT	B
4	171404	PBO	A
5	171404	KCB	C
6	171401	ASD	B+
7	171401	BDD	B+
8	171401	MMT	B+

8 rows in set (0.000 sec)

Berikut penjelasannya:

- **INSERT INTO nilai (NIM, kode_mata_kuliah, nilai_huruf):** Digunakan untuk menyisipkan data ke dalam tabel **nilai**, dengan menyediakan nilai untuk kolom NIM, kode_mata_kuliah, dan nilai_huruf.
- **SELECT NIM, 'ASD', 'B+' FROM nilai WHERE nilai_huruf = 'B+':** Adalah subquery pertama yang mengambil NIM dari tabel nilai dimana nilai_huruf adalah 'B+'. Data tersebut akan disisipkan ke dalam tabel nilai dengan kode_mata_kuliah 'ASD' dan nilai_huruf 'B+'.
- **UNION ALL:** Adalah operator UNION yang digunakan untuk menggabungkan hasil dari tiga subquery, termasuk duplikat.
- **LIMIT 3:** Ini adalah klausa yang membatasi jumlah baris yang akan disisipkan.

3. Berikut Query untuk mencari kode matakuliah yang sudah diisi pada tabel nilai dan tabel matakuliah!

```
SELECT DISTINCT kode_mata_kuliah
FROM nilai
WHERE kode_mata_kuliah IN (SELECT DISTINCT kode_mata_kuliah FROM nilai);
```

Berikut hasilnya:

```
MariaDB [akademik_3]> select * from nilai;
+----+-----+-----+-----+
| id | NIM   | kode_mata_kuliah | nilai_huruf |
+----+-----+-----+-----+
| 1  | 171401 | KCB              | B+          |
| 2  | 171402 | ASD              | A           |
| 3  | 171403 | MMT              | B           |
| 4  | 171404 | PBO              | A           |
| 5  | 171404 | KCB              | C           |
+----+-----+-----+-----+
5 rows in set (0.000 sec)
```

```
MariaDB [akademik_3]> SELECT DISTINCT kode_mata_kuliah
-> FROM nilai
-> WHERE kode_mata_kuliah IN (SELECT DISTINCT kode_mata_kuliah FROM nilai);
+-----+
| kode_mata_kuliah |
+-----+
| KCB              |
| ASD              |
| MMT              |
| PBO              |
+-----+
4 rows in set (0.001 sec)

MariaDB [akademik_3]> |
```

Berikut penjelasannya:

- **SELECT DISTINCT kode_mata_kuliah:** Digunakan untuk memilih nilai unik dari kolom **kode_mata_kuliah** dalam tabel **nilai**.
- **WHERE kode_mata_kuliah IN (SELECT DISTINCT kode_mata_kuliah FROM nilai):** Adalah subquery yang digunakan untuk memfilter baris yang memiliki kode mata kuliah yang ada dalam tabel **nilai**. Dengan kata lain, subquery ini mengambil semua kode mata kuliah yang ada dalam tabel **nilai**.

4. Berikut Query untuk mencari data yang tidak bernilai diantara C dan E

```
select * from nilai
where nilai_huruf not between 'C' AND 'E';
```

Berikut hasilnya:

```
MariaDB [akademik_3]> select * from nilai;
+----+-----+-----+-----+
| id | NIM   | kode_mata_kuliah | nilai_huruf |
+----+-----+-----+-----+
| 1  | 171401 | KCB              | B+          |
| 2  | 171402 | ASD              | A           |
| 3  | 171403 | MMT              | B           |
| 4  | 171404 | PBO              | A           |
| 5  | 171404 | KCB              | C           |
+----+-----+-----+-----+
5 rows in set (0.000 sec)
```

```
+----+-----+-----+-----+
| id | NIM   | kode_mata_kuliah | nilai_huruf |
+----+-----+-----+-----+
| 1  | 171401 | KCB              | B+          |
| 2  | 171402 | ASD              | A           |
| 3  | 171403 | MMT              | B           |
| 4  | 171404 | PBO              | A           |
+----+-----+-----+-----+
```

Berikut penjelasannya:

- **SELECT * FROM nilai:** Ini adalah bagian utama yang menentukan bahwa kita ingin mengambil semua kolom dari tabel **nilai**.
- **WHERE nilai_huruf NOT BETWEEN 'C' AND 'E':** Adalah klausa WHERE yang memfilter baris-baris dalam tabel **nilai**. Kondisi NOT BETWEEN 'C' AND 'E' digunakan untuk mengambil baris-baris nilai_huruf tidak berada di antara 'C' dan 'E'.

5. Berikut Query untuk semua kolom dalam tabel matakuliah yang mempunyai jumlah sks lebih dari jumlah sks yang dimiliki oleh nama mata kuliah Basis Data Dasar

```
select * from mata_kuliah
where sks > (select sks from mata_kuliah where nama = 'Basis Data Dasar');
```

```
MariaDB [akademik_3]> select * from mata_kuliah;
```

kode	nama	sks
ASD	Algortima dan Struktur Data	5
BDD	Basis Data Dasar	2
KCB	Kecerdasan Buatan	1
MMT	Multimedia Terapan	4
PBO	Pemrograman Berorientasi Objek	6
SPK	Sistem Pendukung Keputusan	3

```
6 rows in set (0.000 sec)
```

```
MariaDB [akademik_3]> select * from mata_kuliah
-> where sks > (select sks from mata_kuliah where nama = 'Basis Data Dasar');
```

kode	nama	sks
ASD	Algortima dan Struktur Data	5
MMT	Multimedia Terapan	4
PBO	Pemrograman Berorientasi Objek	6
SPK	Sistem Pendukung Keputusan	3

```
4 rows in set (0.000 sec)
```

Berikut penjelasannya:

- **SELECT * FROM mata_kuliah:** Untuk mengambil semua kolom dari tabel **mata_kuliah**.
- **WHERE sks > (SELECT sks FROM mata_kuliah WHERE nama = 'Basis Data Dasar');** Adalah klausa WHERE yang memfilter baris-baris dalam tabel **mata_kuliah**. Subquery (**SELECT sks FROM mata_kuliah WHERE nama = 'Basis Data Dasar'**) digunakan untuk mengambil jumlah SKS dari mata kuliah yang memiliki nama 'Basis Data Dasar'. Kondisi **sks > ...** menentukan bahwa kita ingin mengambil baris-baris di mana jumlah SKS (**sks**) lebih besar dari jumlah SKS dari mata kuliah 'Basis Data Dasar'.