Modul IX STORED PROCEDURE

A. Kompetensi Dasar:

- 1. Mahasiswa mampu memahami implementasi stored procedure di dalam basis data
- 2. Mahasiswa mampu menyelesaikan operasi operasi data spesifik dengan memanfaatkan stored procedure

B. Teori Singkat:

Stored Procedure adalah sebuah prosedur layaknya subprogram (subrutin) di dalam bahasa pemrograman reguler yang tersimpan di dalam katalog basis data.

Pada sebuah stored procedure dapat diberikan parameter sesuai dengan kebutuhan, sehingga dengan menerapkan stored procedure yang disimpan di dalam suatu database akan dapat digunakan lebih dinamis berdasarkan parameter yang diberikan, juga stored procedure dapat dikombinasikan dengan fungsi-fungsi pernyataan bersyarat dan berkondisi seperti fungsi IF, While.

Stored procedure dibuat dan disimpan hanya satu kali pada sebuah database, selanjutnya dapat dipanggil kembali, sehingga prosedur dengan nama yang sama dalam program dapat lebih cepat dieksekusi. Jenis perintah-perintah dalam stored procedure meliputi menambah data(insert), mengubah data (update), menghapus data(delete), dan memilih/menyeleksi data (select).

Beberapa kelebihan yang ditawarkan stored procedure antara lain : meningkatkan performa, mereduksi trafik jaringan, reusable, dan meningkatkan kontrol sekuriti.

Sintaks Stored Procedure

Untuk memanggil stored procedure, digunakan perintah CALL CALL nama_sp();

Dalam Implementasinya, penggunaan stored procedure sering melibatkan parameter.

Di MySQL, parameter stored procedure dibedakan menjadi tiga mode : IN, OUT, dan INOUT.

IN

Parameter yang merupakan mode default ini mengindikasikan bahwa sebuah parameter dapat di-pass ke dalam stored procedure tetapi nilainya tidak dapat diubah dari dalam stored procedure

OUT

Mode ini mengindikasikan bahwa stored procedure dapat mengubah parameter dan mengirimkan kembali ke program pemanggil

INOUT

Mode ini pada dasarnya merupakan kombinasi dari mode IN dan OUT.

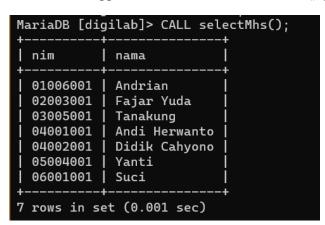
C. Langkah Praktikum:

 Membuat Stored Procedure selectMhs() untuk mendapatkan seluruh NIM dan NAMA mahasiswa

Sintaks:

```
DELIMITER //
CREATE PROCEDURE selectMhs()
BEGIN
SELECT nim, nama FROM mahasiswa;
END //
DELIMITER;
```

Untuk memanggil Stored Procedure selectMhs() gunakan perintah CALL



2. DML dengan Stored Procedure

Misal kita ingin memasukkan data mahasiswa dengan Stored Procedure

Sintaks:

```
DELIMITER //
CREATE PROCEDURE insertMhs (
    nim INT(8),
    nama VARCHAR(30),
    kode_prodi CHAR(3)
)
BEGIN
    INSERT INTO Mahasiswa VALUES (nim, nama, kode_prodi);
END //
DELIMITER;
```

Penggunaan

```
MariaDB [digilab] > CALL insertMhs('06002001','Agung','P02');
Query OK, 1 row affected (0.010 sec)
MariaDB [digilab]> select * from mahasiswa;
l nim
             nama
                               kode_prodi
  01006001
              Andrian
                                P06
  02003001
              Fajar Yuda
                                P03
  03005001
                                P05
              Tanakung
  04001001
              Andi Herwanto
                                P01
  04002001
              Didik Cahyono
                                P<sub>0</sub>2
  05004001
              Yanti
                                P<sub>0</sub>4
  06001001
              Suci
                                P07
  6002001
              Agung
                                P02
8 rows in set (0.001 sec)
```

Selanjutnya jika ingin memasukkan data mahasiswa ke table mahasiswa tidak perlu membuat kode INSERT berkali-kali. Kita bisa gunakan Stored Procedure insertMhs() untuk menggantikan proses INSERT yang biasanya kita gunakan

3. Stored Procedure dengan Parameter

Kita juga memasukkan parameter di Stored Procedure agar menjadi lebih dinamis

Contoh, kita ingin membuat kode SQL untuk mencari data Fakultas berdasarkan lokasi. (gunakan parameter IN)

sintaks:

```
DELIMITER //
```

CREATE PROCEDURE cariLokasiFak(IN lokasiFakultas VARCHAR(20)) BEGIN

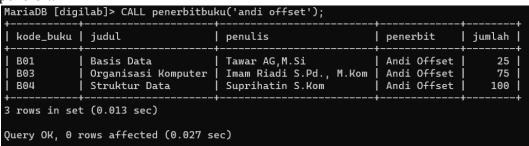
SELECT * FROM Fakultas WHERE lokasi = lokasiFakultas;

END //

DELIMITER;

D. Tugas

1. Buat stored procedure dimana kita bisa mendapatkan data buku berdasarkan penerbit.



- 2. Buat stored procedure untuk menambah data di tabel buku.
- 3. Buat stored procedure yang bisa mendapatkan semua data mahasiswa di semester tertentu (tambah kolom semester terlebih dahulu di tabel mahasiswa)

