

Laporan Praktikum 9

Konsep Basis Data

STORED PROCEDURE



Nama : Taufan Ali
NIM : 2215016135
Kelas Praktikum : C

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI TERAPAN
UNIVERSITAS AHMAD DAHLAN
YOGYAKARTA
2023

A. Kompetensi Dasar :

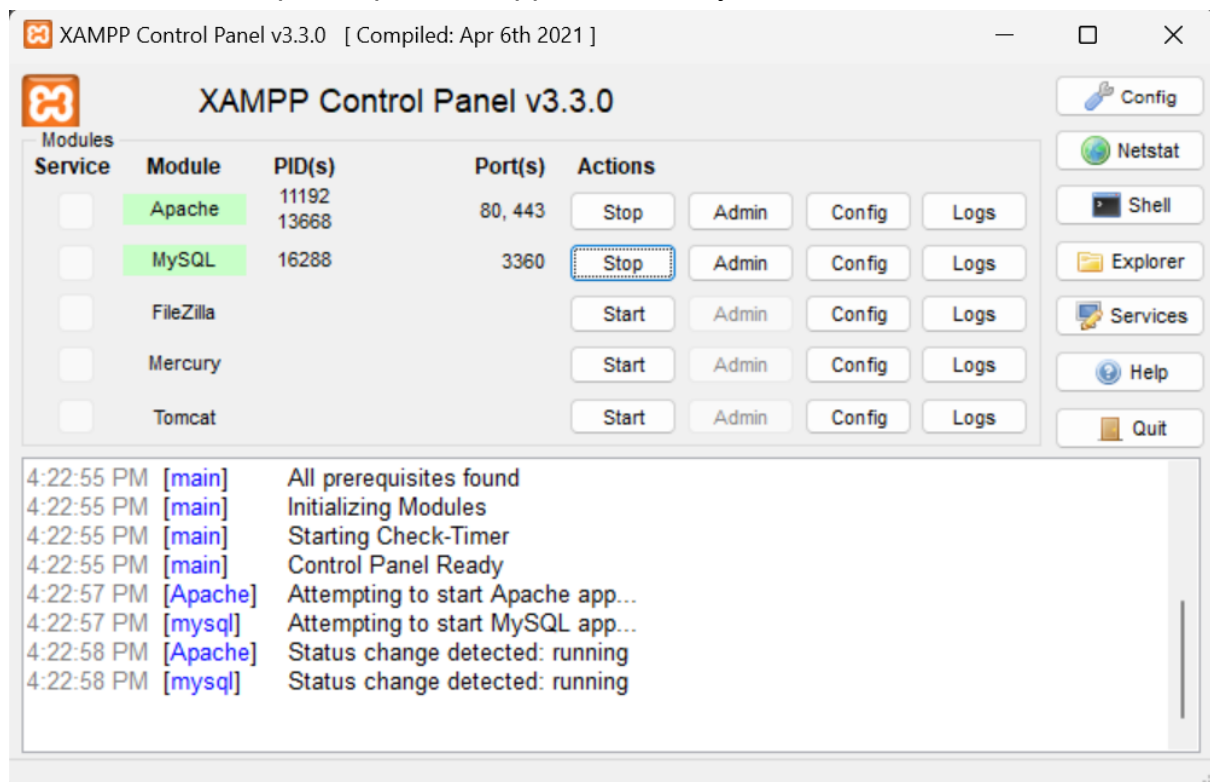
- Mahasiswa mampu memahami implementasi stored procedure di dalam basis data
- Mahasiswa mampu menyelesaikan operasi - operasi data spesifik dengan memanfaatkan stored procedure

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.

C. Percobaan :

- Sebelum membuat database, pastikan terlebih dahulu bahwa mysql dan apache pada xampp sudah menyala



2. Masuk ke folder bin yang ada pada folder mysql yang berada pada folder xampp

```
Lenovo Ideapad@AlienzWindows C: > cd xampp/mysql/bin
Lenovo Ideapad@AlienzWindows C: > xampp > mysql > bin
```

3. Login ke mysql dengan menggunakan perintah **mysql -u *username_anda***(default=root) atau **mysql -u root -p**(jika menggunakan password)

```
Lenovo Ideapad@AlienzWindows C: > xampp > mysql > bin mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.33 MySQL Community Server - GPL

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> |
```

4. Gunakan tabel digilabs yang telah dibuat pada saat praktikum 2

```
MariaDB [(none)]> use digilabs;
Database changed
MariaDB [digilabs]> |
```

5. Membuat Stored Procedure selectMhs() untuk mendapatkan seluruh NIM dan Nama Mahasiswa

```
MariaDB [digilab]> delimiter //
MariaDB [digilab]> create procedure selectMhs()
→ begin
→ select nim, nama from mahasiswa;
→ end //
Query OK, 0 rows affected (0.033 sec)
```

6. Untuk memanggil procedure tersebut gunakan query CALL

```
MariaDB [digilab]> call selectMhs();
```

nim	nama
01006001	Andrian
02003001	Fajar Yuda
03005001	Tanakung
04001001	Andi Herwanto
04002001	Didik Cahyono
05004001	Yanti

```
6 rows in set (0.001 sec)
```

Query OK, 0 rows affected (0.011 sec)

7. Membuat DML dengan stored procedure. Pada praktikum 9 dilakukan pembuatan procedure untuk menambahkan data pada tabel mahasiswa

```
MariaDB [digilab]> create procedure insertMhs (nim int(8), n  
ama varchar(30), kode_prodi char(3))  
→ begin  
→ insert into mahasiswa values (nim, nama, kode_prodi);  
  
→ end //
```

Query OK, 0 rows affected (0.010 sec)

8. Untuk memanggil procedure tersebut gunakan query CALL. Namun pada pemanggilan procedure insertMhs, isikan juga nim, nama, serta kode prodi yang akan dimasukkan

```
MariaDB [digilab]> call insertMhs('06002001', 'Agung', 'p02');
Query OK, 1 row affected (0.008 sec)

MariaDB [digilab]> select * from mahasiswa;
+-----+-----+-----+
| nim      | Nama      | kode_prodi |
+-----+-----+-----+
| 01006001 | Andrian   | P06        |
| 02003001 | Fajar Yuda | P03        |
| 03005001 | Tanakung  | P05        |
| 04001001 | Andi Herwanto | P01        |
| 04002001 | Didik Cahyono | P02        |
| 05004001 | Yanti     | P04        |
| 6002001  | Agung     | p02        |
+-----+-----+-----+
7 rows in set (0.001 sec)
```

9. Stored Procedure dengan parameter. Pada praktikum 9 dilakukan pembuatan procedure cariLokasiFak yang menggunakan parameter lokasi untuk mencari dimana lokasi fakultas yang dicari

```
MariaDB [digilab]> delimiter //
MariaDB [digilab]> create procedure cariLokasiFak(in lokasiFakultas varchar(20))
    → begin
    → select * from fakultas where lokasi = lokasiFakultas;

    → end //
Query OK, 0 rows affected (0.010 sec)
```

10. Untuk memanggil procedure tersebut gunakan query CALL dan isi parameternya dengan unit kampus

```
MariaDB [digilab]> call cariLokasiFak('Kampus I');
+-----+-----+-----+
| kode_fak | nama_fakultas | Lokasi |
+-----+-----+-----+
| F02      | FAK. Ekonomi  | Kampus I |
| F06      | FAK. Psikologi | Kampus I |
+-----+-----+-----+
2 rows in set (0.010 sec)

Query OK, 0 rows affected (0.016 sec)
```

11. Tugas Praktikum :

- 11.1. Buat stored procedure untuk mendapatkan data buku berdasarkan penerbitnya. Buat procedure dengan nama penerbitBuku dengan parameter namaPenerbit. Untuk query yang digunakan untuk mencari buku berdasarkan penerbit dapat dilihat pada gambar di bawah

```
MariaDB [digilab]> delimiter //
MariaDB [digilab]> create procedure penerbitBuku(in namaPenerbit varchar(20))
→ begin
→ select * from buku where penerbit = namaPenerbit;
→ end //
Query OK, 0 rows affected (0.008 sec)
```

```
MariaDB [digilab]> call penerbitBuku('Andi Offset');
+-----+-----+-----+-----+
| kode_buku | Judul          | penulis          | penerbit | ju |
+-----+-----+-----+-----+
| B01      | Basis Data    | Tawar AG, M.Si  | Andi Offset | 25 |
| B03      | Organisasi Komputer | Imam Riadi, SPd, M.Kom | Andi Offset | 75 |
+-----+-----+-----+-----+
2 rows in set (0.001 sec)

Query OK, 0 rows affected (0.007 sec)
```


- 11.2. Buat stored procedure untuk mendapatkan semua data mahasiswa di semester tertentu. Karena sebelumnya pada tabel mahasiswa belum terdapat kolom semester, oleh karena itu harus melakukan alter table dengan tujuan menambah kolom baru dengan nama semester. Tak lupa pula gunakan update untuk menambahkan jumlah semester tiap mahasiswa pada tabel mahasiswa. Setelah ditambahkan kolom dan data semester pada tabel mahasiswa, barulah buat procedure mhsBySemester yang mengambil parameter semester dan query yang digunakan untuk mengambil mahasiswa dengan semester tertentu dapat dilihat pada gambar di bawah

```
MariaDB [digilab]> alter table mahasiswa add column semester  
varchar(5);
```

```
MariaDB [digilab]> update mahasiswa set semester = '5' where  
nim = 03005001;  
Query OK, 0 rows affected (0.001 sec)  
Rows matched: 1 Changed: 0 Warnings: 0
```

```
MariaDB [digilab]> select * from mahasiswa;
```

nim	Nama	kode_prodi	semester
01006001	Andrian	P06	3
02003001	Fajar Yuda	P03	3
03005001	Tanakung	P05	5
04001001	Andi Herwanto	P01	5
04002001	Didik Cahyono	P02	5
05004001	Yanti	P04	3
6002001	Agung	p02	3

```
7 rows in set (0.001 sec)
```

```
MariaDB [digilab]> create procedure mhsBySemester(in semester  
rMahasiswa varchar(5))  
→ begin  
→ select * from mahasiswa where semester = semesterMaha  
siswa;  
→ end //  
Query OK, 0 rows affected (0.011 sec)
```

```
MariaDB [digilab]> call mhsBySemester('5');
```

nim	Nama	kode_prodi	semester
03005001	Tanakung	P05	5
04001001	Andi Herwanto	P01	5
04002001	Didik Cahyono	P02	5

```
3 rows in set (0.001 sec)  
  
Query OK, 0 rows affected (0.010 sec)
```