

**LAPORAN PRAKTIKUM**  
**PRAKTIKUM 3:**  
**“STORED PROCEDURE”**



**Disusun Oleh :**

Oktaviana Sadama Nur Azizah  
24060121130060

**PRAKTIKUM MANAJEMEN BASIS DATA**  
**LAB A2**

**DEPARTEMEN ILMU KOMPUTER / INFORMATIKA**  
**FAKULTAS SAINS DAN MATEMATIKA**  
**UNIVERSITAS DIPONEGORO**  
**SEMARANG**  
**2023**

# **BAB I**

## **PENDAHULUAN**

### **A. Rumusan Masalah**

Buatlah Stored Procedure untuk:

1. Menampilkan data pasien yang diurutkan berdasarkan nama kode pasien dan nama pasien.
  - a) Diurutkan berdasarkan kode pasien kemudian nama pasien.
  - b) Diurutkan berdasarkan nama pasien kemudian kode pasien.
2. Jumlah karyawan untuk masing-masing nama kode karyawan (jumlah karyawan keseluruhan).
3. Jumlah pasien untuk masing-masing nama kode pasien (jumlah pasien keseluruhan).
4. Pencarian data pasien berdasarkan nama.
5. Pencarian data karyawan berdasarkan nama kode karyawan.
6. Data pasien (Kode Karyawan Huruf, Kode Karyawan Angka, Nama Pasien, No Ruangan, Nama Ruangan) yang sudah mendaftar dalam antrian tetapi tidak melakukan pemeriksaan untuk masing-masing tanggal pemeriksaan.
7. Data jadwal shift karyawan per minggu untuk masing-masing karyawan.
8. Data obat yang sudah kadaluarsa per hari (berdasarkan tanggal hari ini).
9. Data riwayat kesehatan untuk masing-masing pasien.
10. Menampilkan nama obat yang paling banyak diberikan kepada pasien untuk masing-masing kode obat.
11. Buatlah minimal 3 Stored Procedure sesuai dengan studi kasus yang pernah anda buat di mata kuliah Basis Data!

### **B. Tujuan**

1. Mahasiswa memiliki pemahaman fungsi dari Stored Procedure dan proses perancangannya menggunakan Microsoft SQL Server.
2. Mahasiswa mampu merancang Stored Procedure dengan parameter sesuai dengan studi kasus yang diberikan.
3. Mahasiswa mampu merancang Stored Procedure tanpa parameter.

## BAB II

### DASAR TEORI

Stored Procedure adalah sebuah fungsi yang berisi kode SQL yang dapat digunakan kembali dengan cara memanggil atau *execute* Stored Procedure yang telah dibuat. Dalam Stored Procedure juga dapat dimasukkan parameter sehingga fungsi dapat digunakan lebih dinamis berdasarkan parameter tersebut.

#### Stored Procedure Syntax

```
CREATE PROCEDURE nama_prosedur  
AS  
BEGIN  
    SQL_statement  
END;
```

#### Pemanggilan Stored Procedure

```
EXEC nama_prosedur;
```

Adapun kelebihan menggunakan Stored Procedure adalah sebagai berikut:

1. Berbagi logik dengan aplikasi lainnya  
Stored Procedure merangkum fungsionalitas untuk memastikan akses data dan manipulasi koheren antara aplikasi yang berbeda.
2. Keamanan  
Stored Procedure mengisolasi pengguna dari tabel data. Fitur ini memberi kemampuan untuk memberikan akses ke Stored Procedure yang memanipulasi data namun tidak secara langsung ke tabel.
3. Performa  
Stored Procedure dikompilasi yang selanjutnya dicache dan akan digunakan kembali. Dengan begitu waktu respon dan performa menjadi lebih cepat karena Stored Procedure yang sama dieksekusi kembali.

Selain kelebihan Stored Procedure yang sudah disebutkan diatas, terdapat pula kekurangan dari Stored Procedure sebagai berikut:

1. Peningkatan beban pada database server. Sebagian besar pekerjaan dilakukan di sisi server dan kurang di sisi client.
2. Pengulangan logika aplikasi di dua tempat yang berbeda: kode bahasa pemrograman dan kode Stored Procedure membuat pemeliharaan aplikasi menjadi lebih sulit.
3. Migrasi ke sistem manajemen basis data yang berbeda berpotensi menjadi lebih sulit.

## BAB III

### PEMBAHASAN

1. Menampilkan data pasien yang diurutkan berdasarkan nama kode pasien dan nama pasien.
  - a) Diurutkan berdasarkan kode pasien kemudian nama pasien.

```
1a.sql - LAPTOP-U...1VB6\Sadama (58) X
-- Author      : Oktaviana Sadama Nur Azizah
-- Create date  : 2023-05-23
-- Description  : Menampilkan data pasien yang diurutkan berdasarkan kode pasien kemudian nama pasien
-- =====
CREATE PROCEDURE sp_pasienKodeNama
AS
BEGIN
    SELECT * FROM pasien
    ORDER BY kodepas_ang, nama_pasien;
END;
EXEC sp_pasienKodeNama
```

100 %

	kodepas_ang	kodepas_hrf	nama_pasien	alamat	kategori_id
1	1	SJ	Sarah Johnson	Jl. Gatot Subroto No. 12	Dewasa
2	2	MA	Muhammad Ali	Jl. Sudirman No. 15	Dewasa
3	3	ED	Emily Davis	Jl. Imam Bonjol No. 18	Remaja
4	4	JP	Juan Perez	Jl. Pemuda No. 10	Lansia
5	5	AS	Anna Smith	Jl. Asia Afrika No. 1	Anak-Anak

Penjelasan:

Stored Procedure dibuat dengan perintah `CREATE PROCEDURE` dengan nama `'sp_pasienKodeNama'`. Store Procedure diatas menampilkan semua isi tabel pasien dengan perintah `SELECT *` pasien dan mengurutkan data tabel pasien secara ascending menggunakan perintah `ORDER BY` berdasarkan kode pasien kemudian nama pasien. Stored Procedure dijalankan dengan memanggil nama Store Procedure yang telah dibuat menggunakan perintah `EXEC`, yaitu `EXEC sp_pasienKodeNama`.

- b) Diurutkan berdasarkan nama pasien kemudian kode pasien.

```
1b.sql - LAPTOP-U...1VB6\Sadama (57) X
-- Author      : Oktaviana Sadama Nur Azizah
-- Create date  : 2023-05-23
-- Description  : Menampilkan data pasien yang diurutkan berdasarkan nama pasien kemudian kode pasien
-- =====
CREATE PROCEDURE sp_pasienNamaKode
AS
BEGIN
    SELECT * FROM pasien
    ORDER BY nama_pasien, kodepas_ang;
END;
EXEC sp_pasienNamaKode;
```

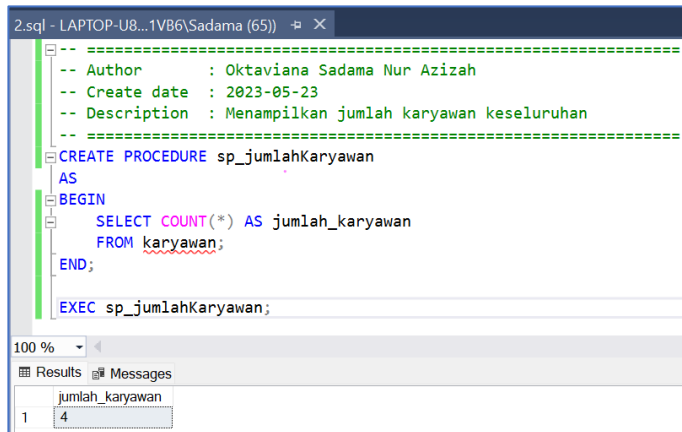
100 %

	kodepas_ang	kodepas_hrf	nama_pasien	alamat	kategori_id
1	5	AS	Anna Smith	Jl. Asia Afrika No. 1	Anak-Anak
2	3	ED	Emily Davis	Jl. Imam Bonjol No. 18	Remaja
3	4	JP	Juan Perez	Jl. Pemuda No. 10	Lansia
4	2	MA	Muhammad Ali	Jl. Sudirman No. 15	Dewasa
5	1	SJ	Sarah Johnson	Jl. Gatot Subroto No. 12	Dewasa

Penjelasan:

Stored Procedure dibuat dengan perintah `CREATE PROCEDURE` dengan nama `'sp_pasienNamaKode'`. Store Procedure diatas menampilkan semua isi tabel pasien dengan perintah `SELECT *` pasien dan mengurutkan data tabel pasien secara ascending menggunakan perintah `ORDER BY` berdasarkan nama pasien kemudian kode pasien. Stored Procedure dijalankan dengan memanggil nama Store Procedure yang telah dibuat menggunakan perintah `EXEC`, yaitu `EXEC sp_pasienNamaKode`.

2. Jumlah karyawan untuk masing-masing nama kode karyawan (jumlah karyawan keseluruhan).



```
-- =====
-- Author      : Oktaviana Sadama Nur Azizah
-- Create date   : 2023-05-23
-- Description   : Menampilkan jumlah karyawan keseluruhan
-- =====

CREATE PROCEDURE sp_jumlahKaryawan
AS
BEGIN
    SELECT COUNT(*) AS jumlah_karyawan
    FROM karyawan;
END;

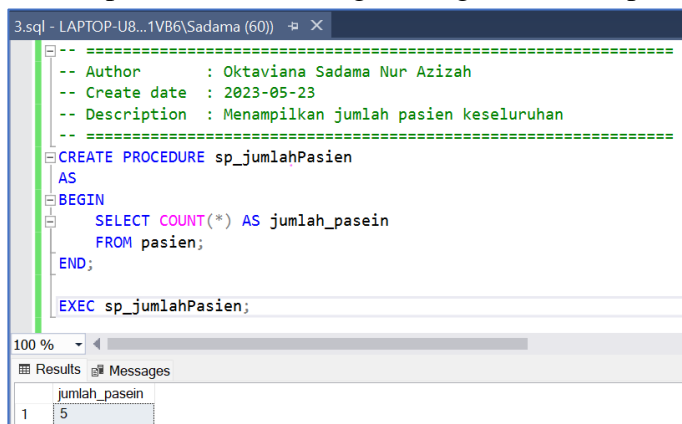
EXEC sp_jumlahKaryawan;
```

jumlah_karyawan
4

Penjelasan:

Stored Procedure dibuat dengan perintah `CREATE PROCEDURE` dengan nama `'sp_jumlahKaryawan'`. Store Procedure diatas menampilkan jumlah karyawan dengan cara menghitung jumlah karyawan menggunakan perintah `COUNT(*)` dan menyimpan hasilnya dalam kolom `jumlah_karyawan`. Stored Procedure dijalankan dengan memanggil nama Store Procedure yang telah dibuat menggunakan perintah `EXEC`, yaitu `EXEC sp_jumlahKaryawan`.

3. Jumlah pasien untuk masing-masing nama kode pasien (jumlah pasien keseluruhan).



```
-- =====
-- Author      : Oktaviana Sadama Nur Azizah
-- Create date   : 2023-05-23
-- Description   : Menampilkan jumlah pasien keseluruhan
-- =====

CREATE PROCEDURE sp_jumlahPasien
AS
BEGIN
    SELECT COUNT(*) AS jumlah_pasein
    FROM pasien;
END;

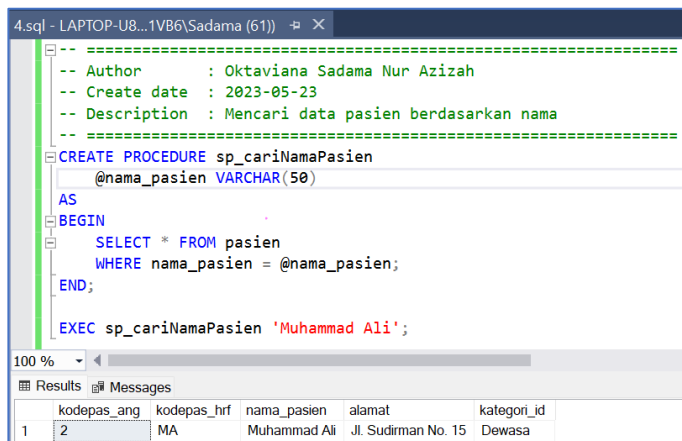
EXEC sp_jumlahPasien;
```

jumlah_pasein
5

Penjelasan:

Stored Procedure dibuat dengan perintah `CREATE PROCEDURE` dengan nama `'sp_jumlahPasien'`. Store procedure diatas menampilkan jumlah pasien dengan cara menghitung jumlah pasien menggunakan perintah `COUNT(*)` dan menyimpan hasilnya dalam kolom `jumlah_pasien`. Stored Procedure dijalankan dengan memanggil nama Store Procedure yang telah dibuat menggunakan perintah `EXEC`, yaitu `EXEC sp_jumlahPasien`.

#### 4. Pencarian data pasien berdasarkan nama.



```
-- Author      : Oktaviana Sadama Nur Azizah
-- Create date   : 2023-05-23
-- Description  : Mencari data pasien berdasarkan nama
-- =====
CREATE PROCEDURE sp_cariNamaPasien
    @nama_pasien VARCHAR(50)
AS
BEGIN
    SELECT * FROM pasien
    WHERE nama_pasien = @nama_pasien;
END;

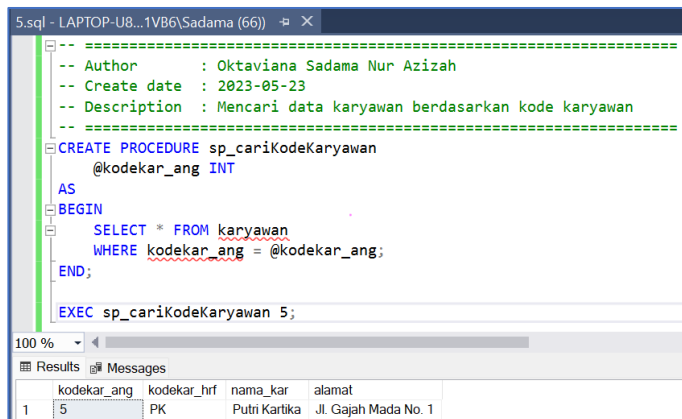
EXEC sp_cariNamaPasien 'Muhammad Ali';
```

	kodepas_ang	kodepas_hrf	nama_pasien	alamat	kategori_id
1	2	MA	Muhammad Ali	Jl. Sudirman No. 15	Dewasa

##### Penjelasan:

Stored Procedure dibuat dengan perintah `CREATE PROCEDURE` dengan nama `'sp_cariNamaPasien'` menggunakan parameter `@nama_pasien` dengan tipe data `VARCHAR(50)`. Parameter ini digunakan untuk menginputkan nama pasien yang kemudian akan ditampilkan data pasien yang dipilih menggunakan perintah `SELECT * FROM pasien`. Stored Procedure dijalankan dengan memanggil nama Store Procedure yang telah dibuat menggunakan perintah `EXEC` diikuti parameter `@nama_pasien`, yaitu `EXEC sp_cariNamaPasien 'Muhammad Ali'`.

#### 5. Pencarian data karyawan berdasarkan nama kode karyawan.



```
-- Author      : Oktaviana Sadama Nur Azizah
-- Create date   : 2023-05-23
-- Description  : Mencari data karyawan berdasarkan kode karyawan
-- =====
CREATE PROCEDURE sp_cariKodeKaryawan
    @kodekar_ang INT
AS
BEGIN
    SELECT * FROM karyawan
    WHERE kodekar_ang = @kodekar_ang;
END;

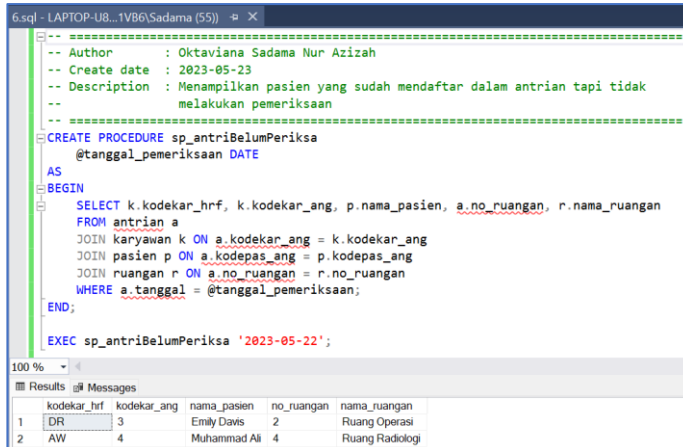
EXEC sp_cariKodeKaryawan 5;
```

	kodekar_ang	kodekar_hrf	nama_kar	alamat
1	5	PK	Putri Kartika	Jl. Gajah Mada No. 1

##### Penjelasan:

Stored Procedure dibuat dengan perintah `CREATE PROCEDURE` dengan nama `'sp_cariKodeKaryawan'` menggunakan parameter `@kodekar_ang` dengan tipe data `INT`. Parameter ini digunakan untuk menginputkan kode karyawan yang kemudian akan menampilkan data karyawan yang dipilih menggunakan perintah `SELECT * FROM karyawan`. Stored Procedure dijalankan dengan memanggil nama Store Procedure yang telah dibuat menggunakan perintah `EXEC` diikuti parameter `@kodekar_ang`, yaitu `EXEC sp_cariKodeKaryawan 5`.

6. Data pasien (Kode Karyawan Huruf, Kode Karyawan Angka, Nama Pasien, No Ruangan, Nama Ruangan) yang sudah mendaftar dalam antrian tetapi tidak melakukan pemeriksaan untuk masing-masing tanggal pemeriksaan.



```
-- Author      : Oktaviana Sadama Nur Azizah
-- Create date   : 2023-05-23
-- Description  : Menampilkan pasien yang sudah mendaftar dalam antrian tapi tidak
--               melakukan pemeriksaan
-- =====
CREATE PROCEDURE sp_antriBelumPeriksa
    @tanggal_pemeriksaan DATE
AS
BEGIN
    SELECT k.kodekar_hrf, k.kodekar_ang, p.nama_pasien, a.no_ruangan, r.nama_ruangan
    FROM antrian a
    JOIN karyawan k ON a.kodekar_ang = k.kodekar_ang
    JOIN pasien p ON a.kodepas_ang = p.kodepas_ang
    JOIN ruangan r ON a.no_ruangan = r.no_ruangan
    WHERE a.tanggal = @tanggal_pemeriksaan;
END;

EXEC sp_antriBelumPeriksa '2023-05-22';
```

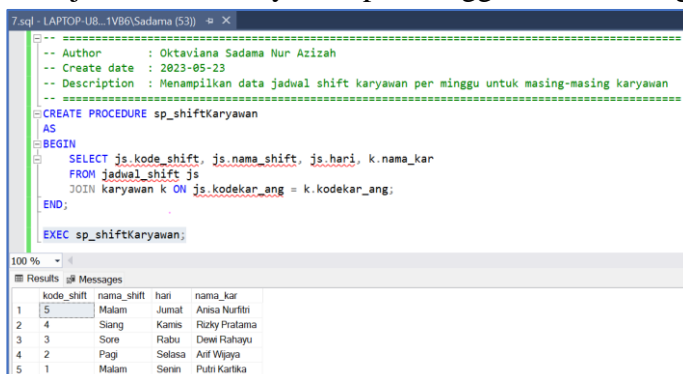
	kodekar_hrf	kodekar_ang	nama_pasien	no_ruangan	nama_ruangan
1	DR	3	Emily Davis	2	Ruang Operasi
2	AW	4	Muhammad Ali	4	Ruang Radiologi

Penjelasan:

Stored Procedure dibuat dengan perintah CREATE PROCEDURE dengan nama 'sp\_antriBelumPeriksa' menggunakan parameter @tanggal\_pemeriksaan dengan tipe data DATE. Parameter ini digunakan untuk menginputkan tanggal pemeriksaan yang kemudian akan menampilkan data pasien berupa kodekar\_hrf, kodekar\_ang, nama\_pasien, no\_ruangan, dan nama\_ruangan yang dipilih menggunakan perintah SELECT.

Perintah JOIN digunakan karena menggabungkan kolom dari beberapa tabel, seperti tabel karyawan, pasien, antrian, dan ruangan kemudian dipilih tanggal antrian yang bernilai sama dengan parameter yang diinputkan. Stored Procedure dijalankan dengan memanggil nama Store Procedure yang telah dibuat menggunakan perintah EXEC diikuti parameter @tanggal\_pemeriksaan, yaitu EXEC sp\_antriBelumPeriksa '2023-05-22'.

7. Data jadwal shift karyawan per minggu untuk masing-masing karyawan.



```
-- Author      : Oktaviana Sadama Nur Azizah
-- Create date   : 2023-05-23
-- Description  : Menampilkan data jadwal shift karyawan per minggu untuk masing-masing karyawan
-- =====
CREATE PROCEDURE sp_shiftKaryawan
AS
BEGIN
    SELECT js.kode_shift, js.nama_shift, js.hari, k.nama_kar
    FROM jadwal_shift js
    JOIN karyawan k ON js.kodekar_ang = k.kodekar_ang;
END;

EXEC sp_shiftKaryawan;
```

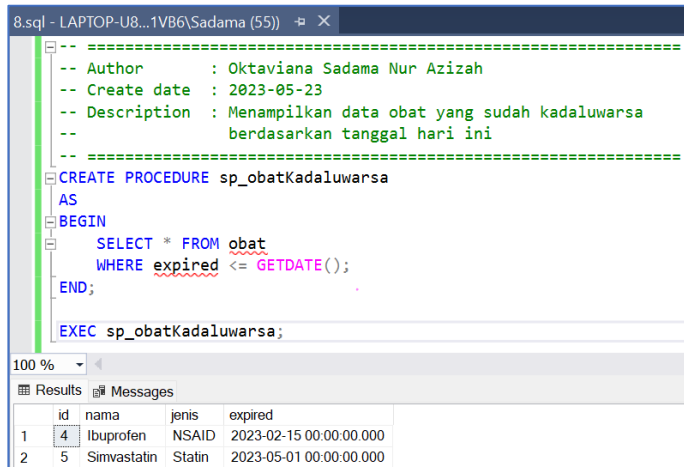
	kode_shift	nama_shift	hari	nama_kar
1	5	Malam	Jumat	Arisa Nurfitri
2	4	Siang	Kamis	Riky Pratama
3	3	Sore	Rabu	Dewi Rahayu
4	2	Pagi	Selasa	Arif Wijaya
5	1	Malam	Senin	Putri Kartika

Penjelasan:

Stored Procedure dibuat dengan perintah CREATE PROCEDURE dengan nama 'sp\_shiftKaryawan'. Store Procedure diatas melakukan SELECT kolom kode\_shift, nama\_shift, hari, dan nama\_kar dari dua tabel, yaitu tabel

jadwal\_shift dan karyawan. Oleh karena itu digunakan perintah JOIN untuk menampilkan kolom dari tabel yang berbeda. Kedua tabel dihubungkan dengan kolom kodekar\_ang. Stored Procedure dijalankan dengan memanggil nama Store Procedure yang telah dibuat menggunakan perintah EXEC, yaitu EXEC sp\_shiftKaryawan.

8. Data obat yang sudah kadaluarsa per hari (berdasarkan tanggal hari ini).



```
-- Author      : Oktaviana Sadama Nur Azizah
-- Create date   : 2023-05-23
-- Description    : Menampilkan data obat yang sudah kadaluarsa
--                  berdasarkan tanggal hari ini
-- =====
CREATE PROCEDURE sp_obatKadaluarsa
AS
BEGIN
    SELECT * FROM obat
    WHERE expired <= GETDATE();
END;

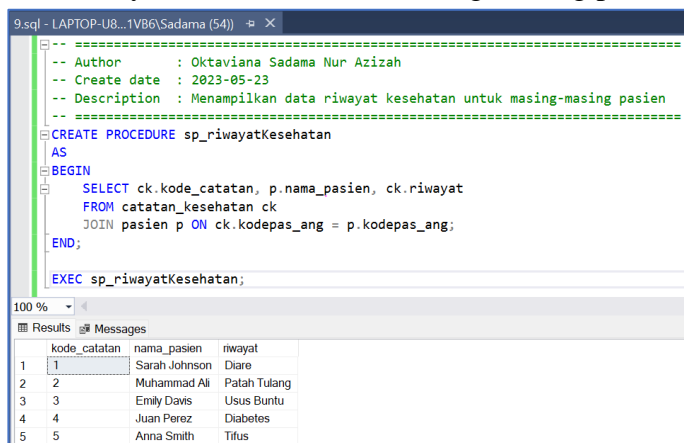
EXEC sp_obatKadaluarsa;
```

	id	nama	jenis	expired
1	4	Ibuprofen	NSAID	2023-02-15 00:00:00.000
2	5	Simvastatin	Statin	2023-05-01 00:00:00.000

Penjelasan:

Stored Procedure dibuat dengan perintah CREATE PROCEDURE dengan nama 'sp\_obatKadaluarsa'. Store Procedure diatas menampilkan data obat dengan perintah SELECT \* FROM obat dimana dipilih tanggal expired yang kurang dari atau sama dengan tanggal hari ini [GETDATE() ]. Stored Procedure dijalankan dengan memanggil nama Store Procedure yang telah dibuat menggunakan perintah EXEC, yaitu EXEC sp\_obatKadaluarsa.

9. Data riwayat kesehatan untuk masing-masing pasien.



```
-- Author      : Oktaviana Sadama Nur Azizah
-- Create date   : 2023-05-23
-- Description    : Menampilkan data riwayat kesehatan untuk masing-masing pasien
-- =====
CREATE PROCEDURE sp_riwayatKesehatan
AS
BEGIN
    SELECT ck.kode_catatan, p.nama_pasien, ck.riwayat
    FROM catatan_kesehatan ck
    JOIN pasien p ON ck.kodepas_ang = p.kodepas_ang;
END;

EXEC sp_riwayatKesehatan;
```

	kode_catatan	nama_pasien	riwayat
1	1	Sarah Johnson	Diare
2	2	Muhammad Ali	Patah Tulang
3	3	Emily Davis	Usus Buntu
4	4	Juan Perez	Diabetes
5	5	Anna Smith	Tifus

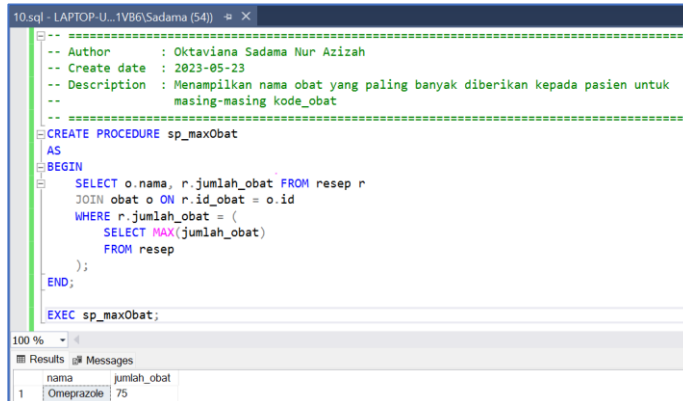
Penjelasan:

Stored Procedure dibuat dengan perintah CREATE PROCEDURE dengan nama 'sp\_riwayatKesehatan'. Store Procedure diatas melakukan SELECT kolom kode\_catatan, nama\_pasien, dan riwayat dari dua tabel, yaitu tabel catatan\_kesehatan dan pasien. Oleh karena itu digunakan perintah JOIN untuk menampilkan kolom dari tabel yang berbeda. Kedua tabel dihubungkan dengan kolom



kodepas\_ang. Stored Procedure dijalankan dengan memanggil nama Store Procedure yang telah dibuat menggunakan perintah EXEC, yaitu EXEC sp\_riwatKesehatan.

10. Menampilkan nama obat yang paling banyak diberikan kepada pasien untuk masing-masing kode obat.



```
-- Author      : Oktaviana Sadama Nur Azizah
-- Create date   : 2023-05-23
-- Description  : Menampilkan nama obat yang paling banyak diberikan kepada pasien untuk
--               masing-masing kode_obat
-- =====
CREATE PROCEDURE sp_maxObat
AS
BEGIN
    SELECT o.nama, r.jumlah_obat FROM resep r
    JOIN obat o ON r.id_obat = o.id
    WHERE r.jumlah_obat = (
        SELECT MAX(jumlah_obat)
        FROM resep
    );
END;

EXEC sp_maxObat;
```

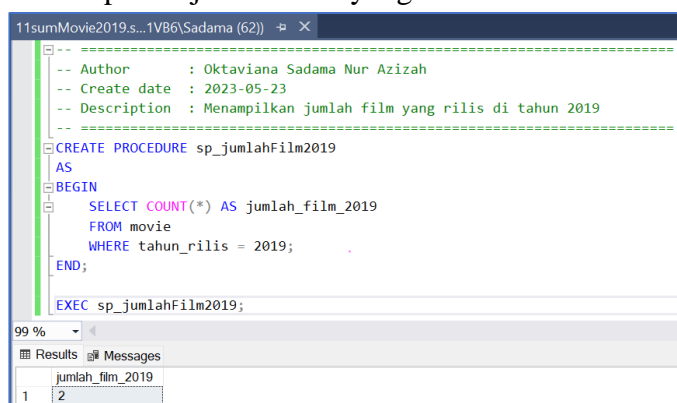
nama	jumlah_obat
Omeprazole	75

Penjelasan:

Stored Procedure dibuat dengan perintah CREATE PROCEDURE dengan nama 'sp\_maxObat'. Store Procedure diatas melakukan SELECT kolom nama\_obat dan jumlah\_obat dari dua tabel, yaitu tabel obat dan resep. Oleh karena itu digunakan perintah JOIN untuk menampilkan kolom dari tabel yang berbeda. Kedua tabel dihubungkan dengan kolom id\_obat. Digunakan fungsi MAX() untuk menampilkan jumlah obat yang paling banyak. Stored Procedure dijalankan dengan memanggil nama Store Procedure yang telah dibuat menggunakan perintah EXEC, yaitu EXEC sp\_maxObat.

11. Buatlah minimal 3 Stored Procedure sesuai dengan studi kasus yang pernah anda buat di mata kuliah Basis Data!

a) Menampilkan jumlah film yang rilis di tahun 2019



```
-- Author      : Oktaviana Sadama Nur Azizah
-- Create date   : 2023-05-23
-- Description  : Menampilkan jumlah film yang rilis di tahun 2019
-- =====
CREATE PROCEDURE sp_jumlahFilm2019
AS
BEGIN
    SELECT COUNT(*) AS jumlah_film_2019
    FROM movie
    WHERE tahun_rilis = 2019;
END;

EXEC sp_jumlahFilm2019;
```

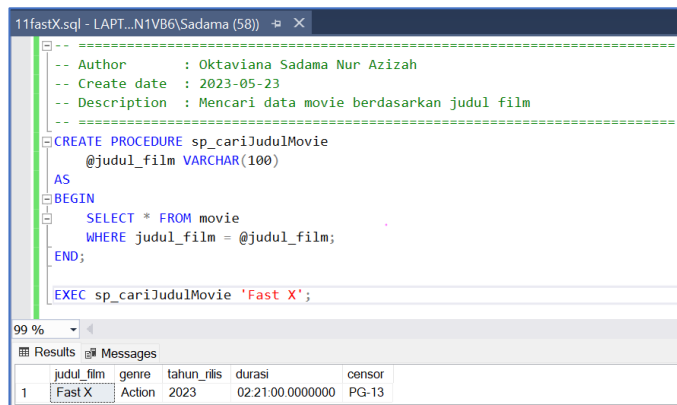
jumlah_film_2019
2

Penjelasan:

Stored Procedure dibuat dengan perintah CREATE PROCEDURE dengan nama 'sp\_jumlahFilm2019'. Store Procedure diatas menghitung jumlah film menggunakan perintah COUNT(\*) dan dipilih film yang rilis di tahun 2019 kemudian menyimpan hasilnya dalam kolom jumlah\_film\_2019. Stored

Procedure dijalankan dengan memanggil nama Store Procedure yang telah dibuat menggunakan perintah EXEC, yaitu EXEC sp\_jumlahFilm2019.

b) Pencarian data movie berdasarkan judul film



```
-- Author      : Oktaviana Sadama Nur Azizah
-- Create date   : 2023-05-23
-- Description  : Mencari data movie berdasarkan judul film
-- =====
CREATE PROCEDURE sp_cariJudulMovie
    @judul_film VARCHAR(100)
AS
BEGIN
    SELECT * FROM movie
    WHERE judul_film = @judul_film;
END;

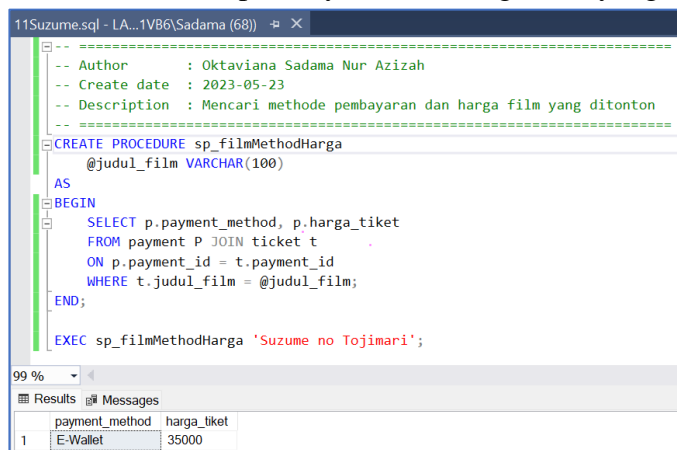
EXEC sp_cariJudulMovie 'Fast X';
```

judul_film	genre	tahun_rilis	durasi	censor
1 Fast X	Action	2023	02:21:00.0000000	PG-13

Penjelasan:

Stored Procedure dibuat dengan perintah CREATE PROCEDURE dengan nama 'sp\_cariJudulMovie' menggunakan parameter @judul\_film dengan tipe data VARCHAR(100). Parameter ini digunakan untuk menginputkan judul film yang kemudian akan menampilkan data movie yang dipilih menggunakan perintah SELECT \* FROM movie. Stored Procedure dijalankan dengan memanggil nama Store Procedure yang telah dibuat menggunakan perintah EXEC diikuti parameter @judul\_film, yaitu EXEC sp\_cariJudulMovie 'Fast X'.

c) Mencari method pembayaran dan harga film yang ditonton



```
-- Author      : Oktaviana Sadama Nur Azizah
-- Create date   : 2023-05-23
-- Description  : Mencari method pembayaran dan harga film yang ditonton
-- =====
CREATE PROCEDURE sp_filmMethodHarga
    @judul_film VARCHAR(100)
AS
BEGIN
    SELECT p.payment_method, p.harga_tiket
    FROM payment P JOIN ticket t
    ON p.payment_id = t.payment_id
    WHERE t.judul_film = @judul_film;
END;

EXEC sp_filmMethodHarga 'Suzume no Tojimari';
```

payment_method	harga_tiket
1 E-Wallet	35000

Penjelasan:

Stored Procedure dibuat dengan perintah CREATE PROCEDURE dengan nama 'sp\_filmMethodHarga' menggunakan parameter @judul\_film dengan tipe data VARCHAR(100). Parameter ini digunakan untuk menginputkan judul film yang kemudian akan menampilkan payment\_method dan juga harga\_tiket dengan perintah JOIN. Stored Procedure dijalankan dengan memanggil nama Store Procedure yang telah dibuat menggunakan perintah EXEC diikuti parameter @judul\_film, yaitu EXEC sp\_cariJudulMovie 'Suzume no Tojimari'.

## **BAB IV**

### **PENUTUP**

#### **A. Kesimpulan**

Stored Procedure adalah sebuah fungsi yang berisi kode SQL yang dapat digunakan kembali dengan cara memanggil Stored Procedure yang telah dibuat dengan perintah `EXEC`. Dalam Stored Procedure juga dapat dimasukkan parameter sehingga fungsi dapat digunakan lebih dinamis berdasarkan parameter tersebut.