**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

1. **Rumusan Masalah**

Buatlah Stored Procedure untuk:

* 1. Menampilkan data pasien yang diurutkan berdasarkan nama kode pasien dan nama pasien.

1. Diurutkan berdasarkan kode pasien kemudian nama pasien.
2. Diurutkan berdasarkan nama pasien kemudian kode pasien.
   1. Jumlah karyawan untuk masing-masing nama kode karyawan (jumlah karyawan keseluruhan).
   2. Jumlah pasien untuk masing-masing nama kode pasien (jumlah pasien keseluruhan).
   3. Pencarian data pasien berdasarkan nama.
   4. Pencarian data karyawan berdasarkan nama kode karyawan.
   5. 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.
   6. Data jadwal shift karyawan per minggu untuk masing-masing karyawan.
   7. Data obat yang sudah kadaluarsa per hari (berdasarkan tanggal hari ini).
   8. Data riwayat kesehatan untuk masing-masing pasien.
   9. Menampilkan nama obat yang paling banyak diberikan kepada pasien untuk masing-masing kode obat.
   10. Buatlah minimal 3 Stored Procedure sesuai dengan studi kasus yang pernah anda buat di mata kuliah Basis Data!
3. **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.

* + - 1. 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.

* + - 1. 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:

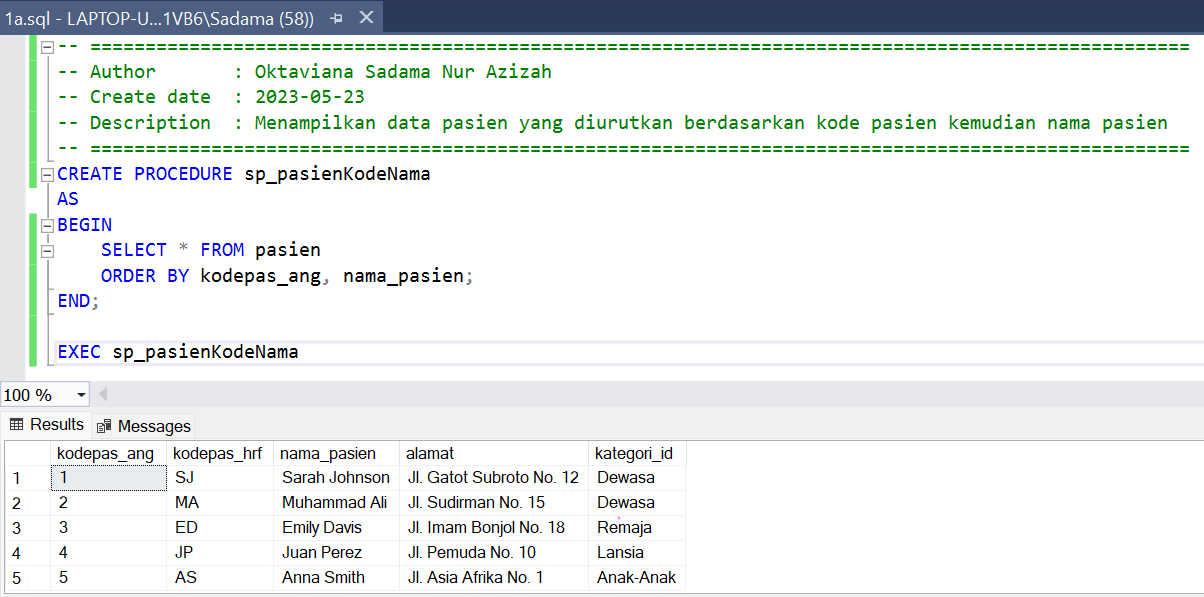
Peningkatan beban pada database server. Sebagian besar pekerjaan dilakukan di sisi server dan kurang di sisi client.

Pengulangan logika aplikasi di dua tempat yang berbeda: kode bahasa pemrograman dan kode Stored Procedure membuat pemeliharaan aplikasi menjadi lebih sulit.

Migrasi ke sistem manajemen basis data yang berbeda berpotensi menjadi lebih sulit.

# BAB III PEMBAHASAN

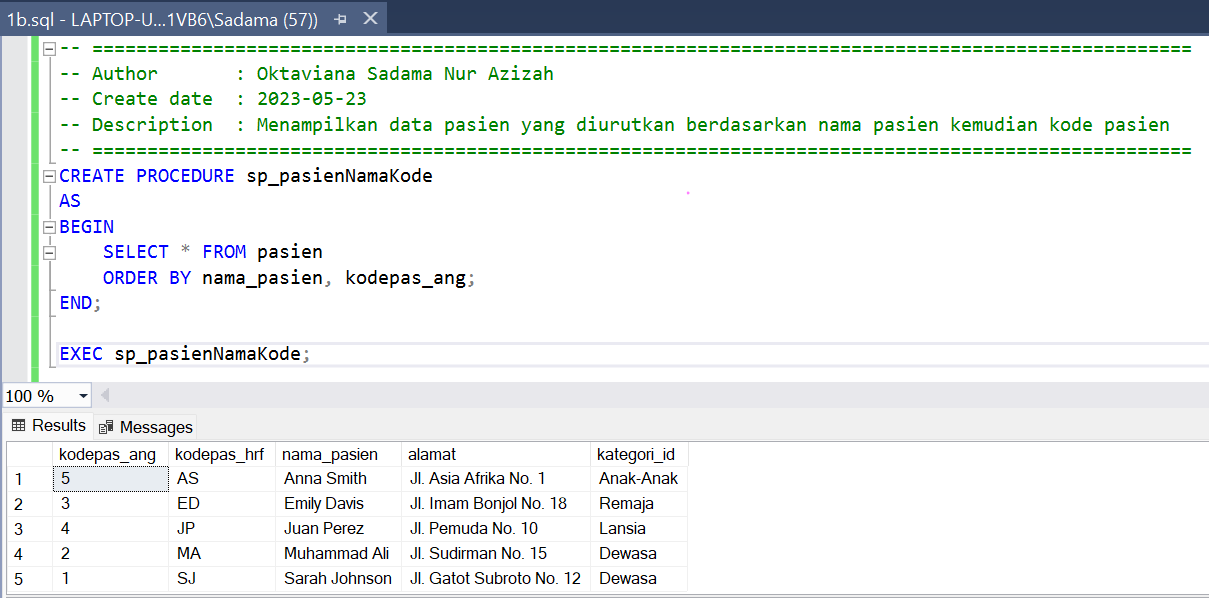
1. Menampilkan data pasien yang diurutkan berdasarkan nama kode pasien dan nama pasien.
2. Diurutkan berdasarkan kode pasien kemudian nama pasien.



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.

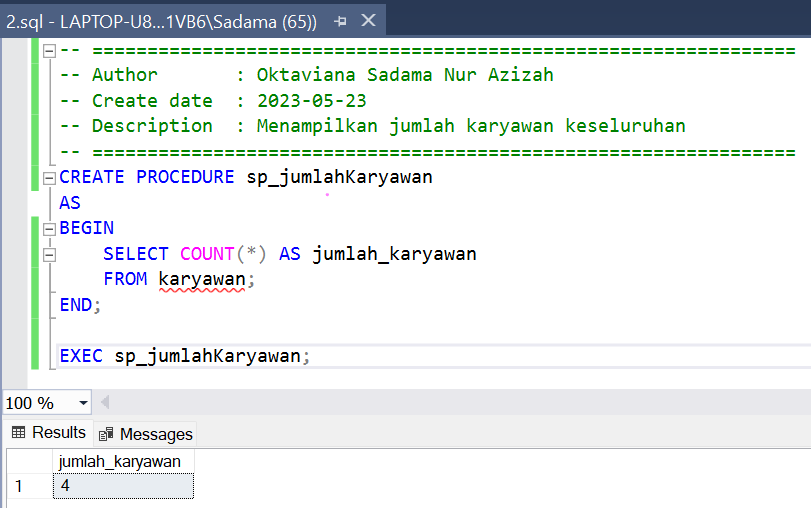
1. Diurutkan berdasarkan nama pasien kemudian kode pasien.



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.

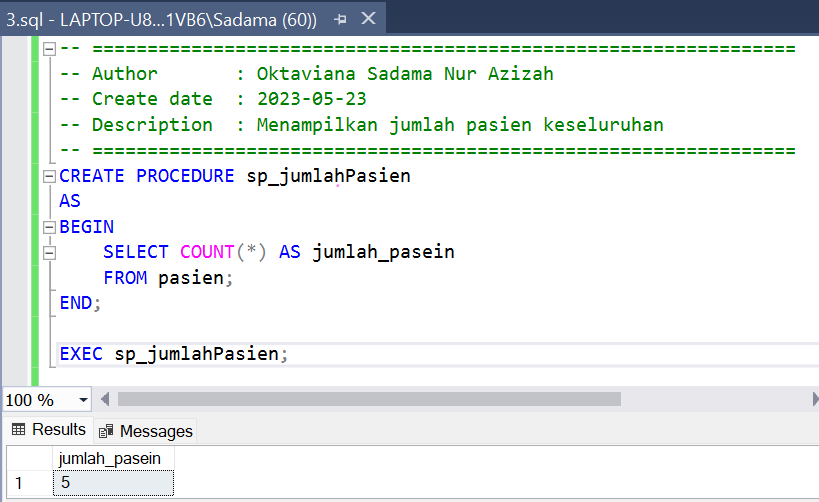
1. Jumlah karyawan untuk masing-masing nama kode karyawan (jumlah karyawan keseluruhan).



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.

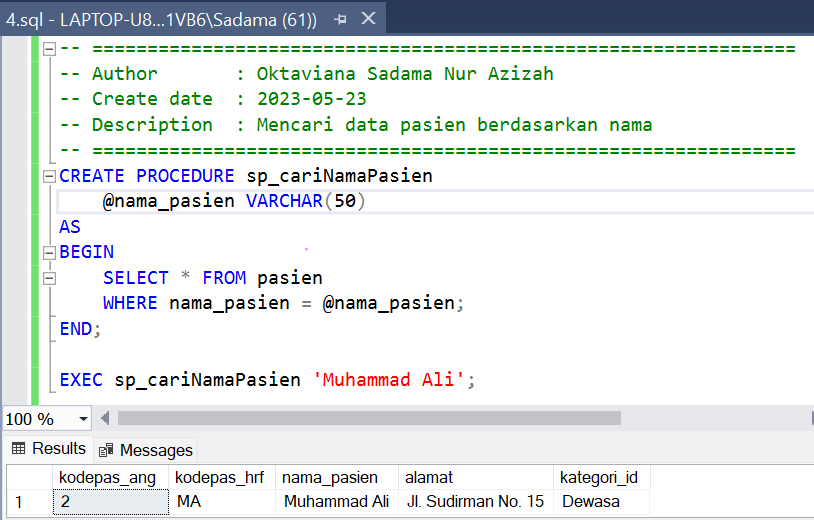
1. Jumlah pasien untuk masing-masing nama kode pasien (jumlah pasien keseluruhan).



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.

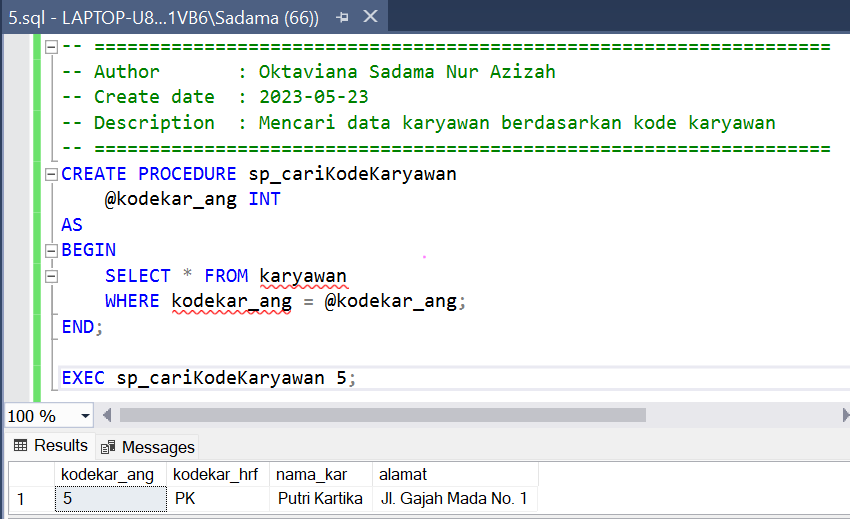
1. Pencarian data pasien berdasarkan nama.



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’.

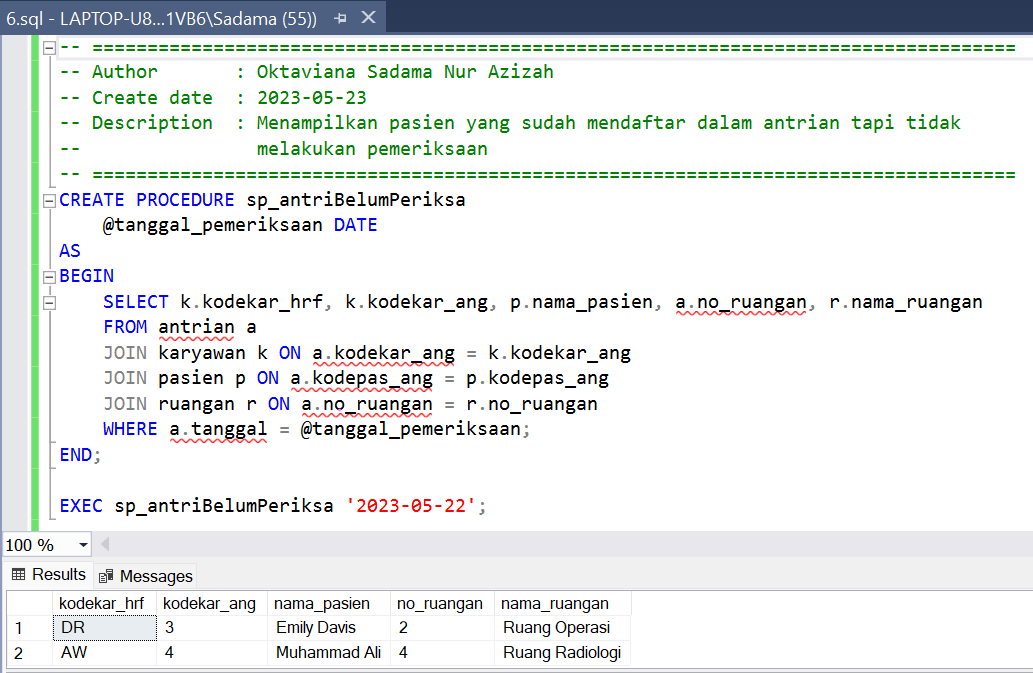
1. Pencarian data karyawan berdasarkan nama kode karyawan.



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.

1. 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.

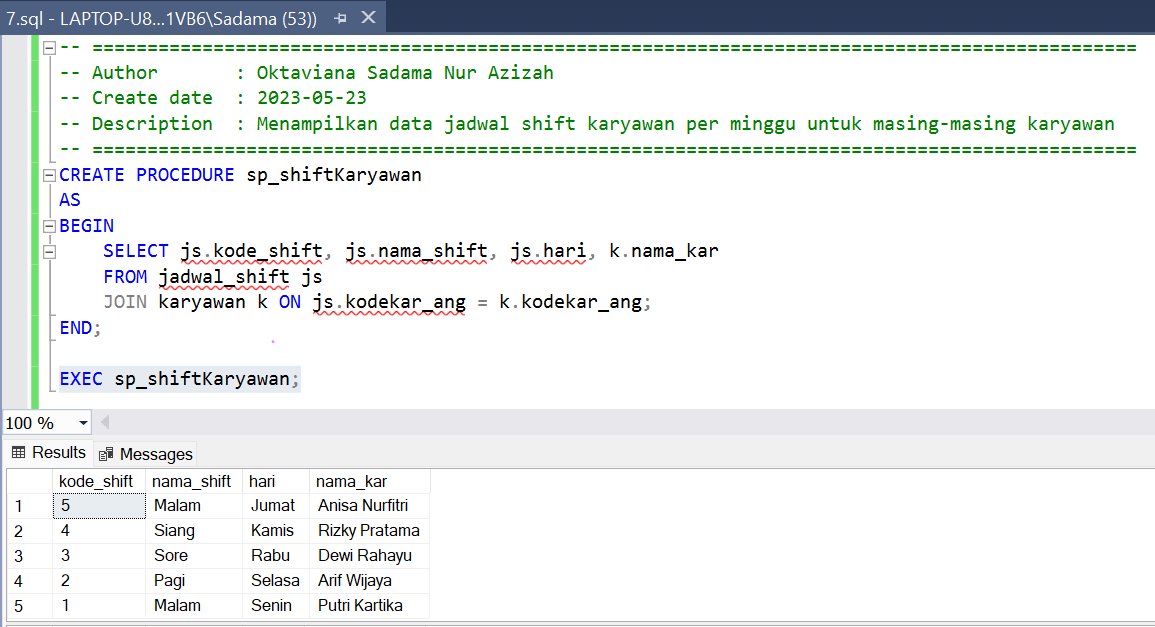


Penjelasan:

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 rungan 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\_antriBelum Periksa ‘2023-05-22’.

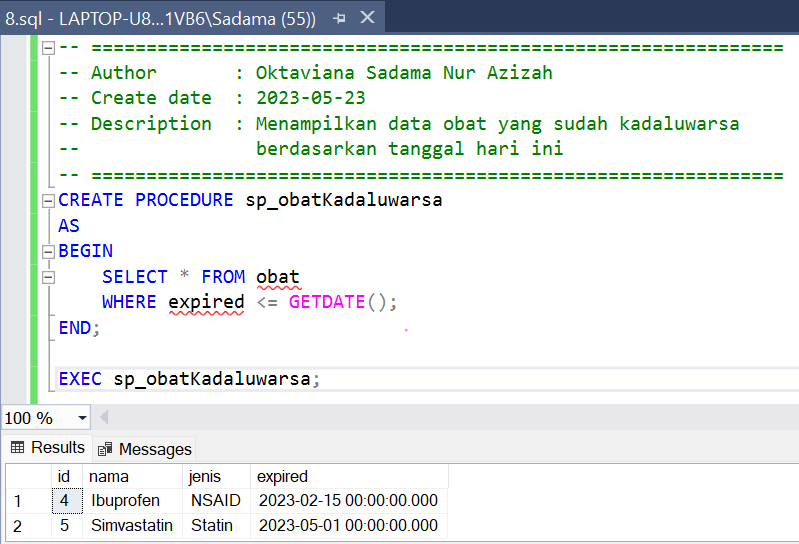
1. Data jadwal shift karyawan per minggu untuk masing-masing karyawan.



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.

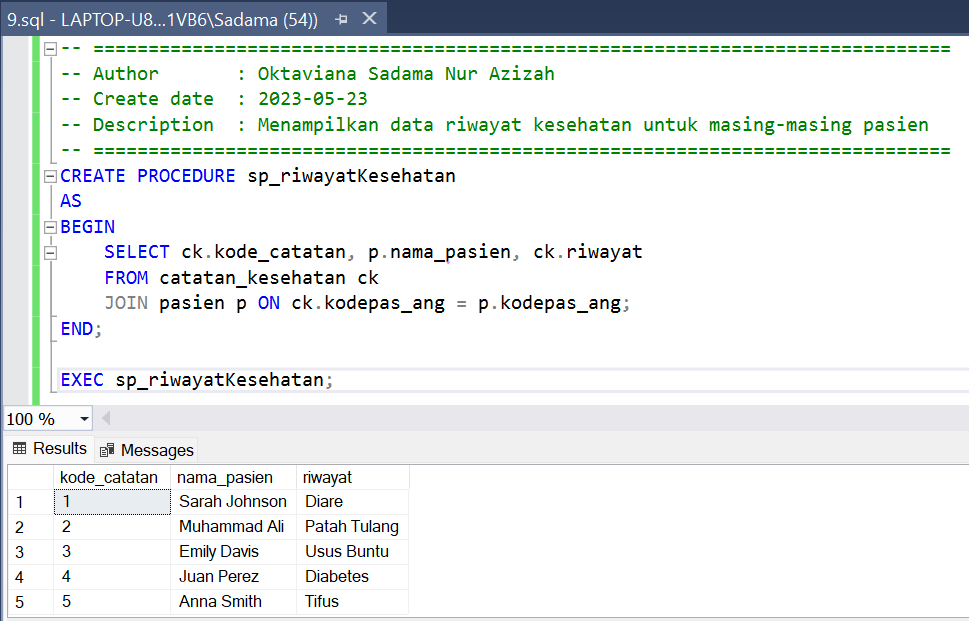
1. Data obat yang sudah kadaluarsa per hari (berdasarkan tanggal hari ini).



Penjelasan:

Stored Procedure dibuat dengan perintah CREATE PROCEDURE dengan nama ‘sp\_obatKadaluwarsa’. 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\_obatKadaluwarsa.

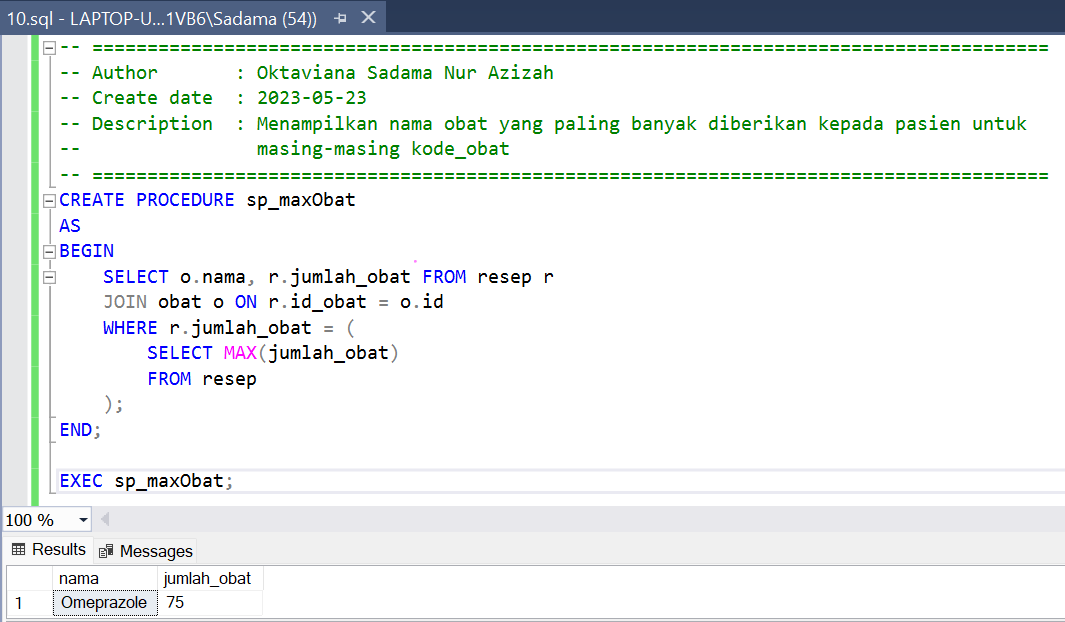
1. Data riwayat kesehatan untuk masing-masing pasien.



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\_riwayatKesehatan.

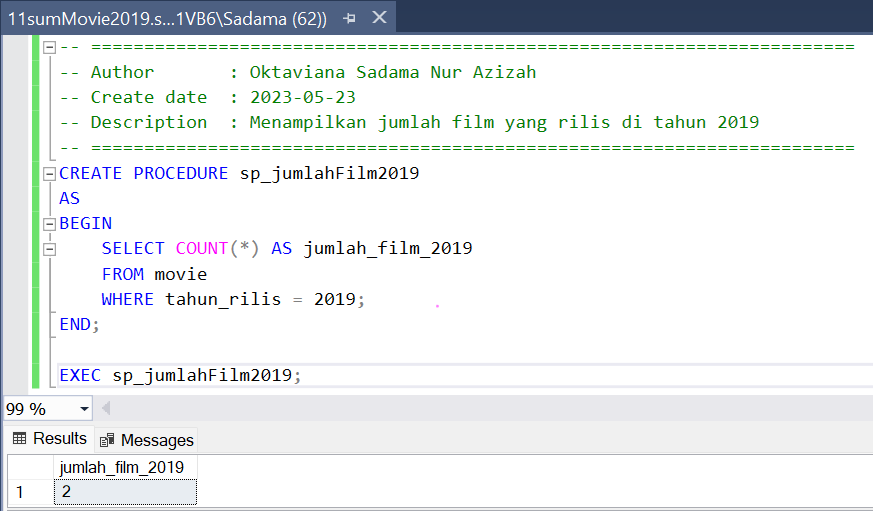
1. Menampilkan nama obat yang paling banyak diberikan kepada pasien untuk masing-masing kode obat.



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.

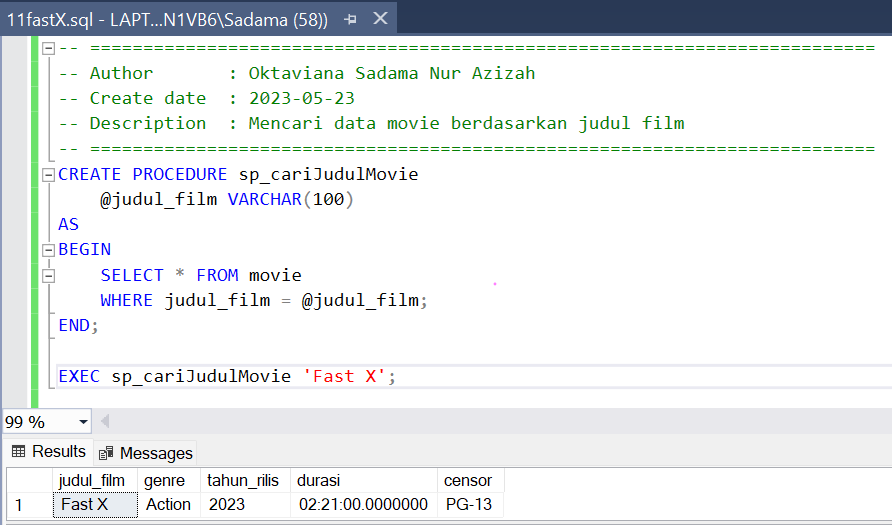
1. Buatlah minimal 3 Stored Procedure sesuai dengan studi kasus yang pernah anda buat di mata kuliah Basis Data!
2. Menampilkan jumlah film yang rilis di tahun 2019



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.

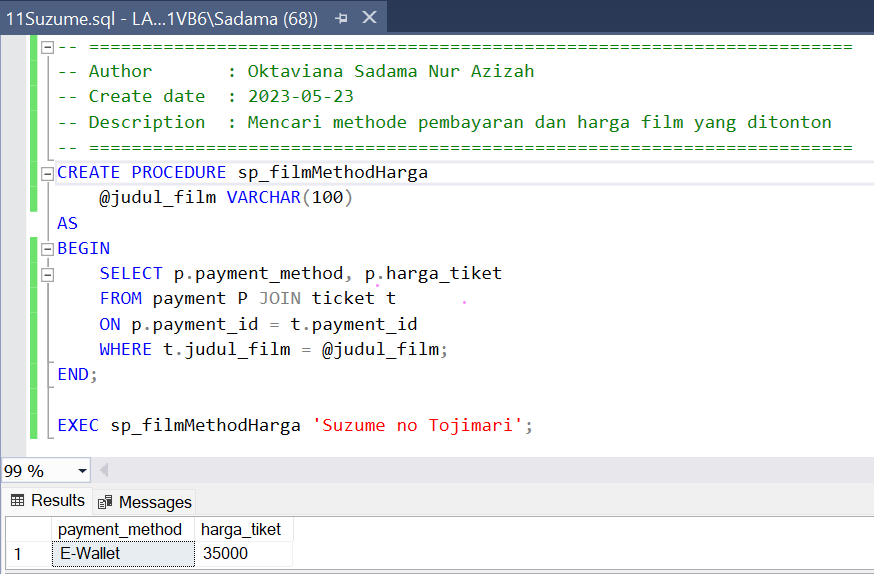
1. Pencarian data movie berdasarkan judul film



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’.

1. Mencari methode pembayaran dan harga film yang ditonton



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

# 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.