**LAPORAN RESMI**

**MODUL V**

**TYPE DATA/STORED PROCEDURE**

**SISTEM MANAJEMEN BASIS DATA**



**NAMA : Agus Putra Singkih**

**N.R.P : 200441100153**

**DOSEN : Fitri Damayanti, S.Kom., M.Kom.**

**ASISTEN :** **Muhammad Iqbal Firmansyah**

**TGL PRAKTIKUM : Rabu, 8 Mei 2024**

**Disetujui : … ..……… 2024**

**Asisten**

**Muhammad Iqbal Firmansyah**

**21.04.411.00084**

**LABORATORIUM BISNIS INTELIJEN SISTEM**

**PRODI SISTEM INFORMASI**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS TRUNOJOYO MADURA**

# **BAB 1 PENDAHULUAN**

## **Latar Belakang**

Sistem Manajemen Basis Data adalah perangkat lunak sistem yang memungkinkan para pemakai membuat, memelihara, mengontrol, dan mengakses sumber data dengan cara praktis dan efisien. DBMS dapat digunakan untuk mengakomodasikan berbagai macam pemakai yang memiliki kebutuhan akses yang berbeda-beda. DBMS pada umumnya menyediakan fasilitas atau fitur-fitur yang memungkinkan data dapat diakses dengan mudah, aman, dan cepat.

Dalam pengembangan perangkat lunak, data merupakan salah satu elemen penting yang harus dikelola dengan baik. Terdapat berbagai macam jenis data seperti teks, angka, tanggal, waktu, dan lain sebagainya. Dalam praktikum Type Data, peserta akan belajar tentang berbagai jenis tipe data dan cara menggunakannya dalam pengembangan perangkat lunak.

Selain itu, dalam praktikum Stored Procedure, peserta akan mempelajari tentang konsep stored procedure dalam database. Stored procedure adalah kumpulan perintah SQL yang diberi nama dan disimpan di dalam database. Stored procedure dapat membantu mempercepat eksekusi perintah dan memperbaiki efisiensi database.

Dalam modul praktikum ini mempelajari tentang Type data dan stored procedure. Tipe data merupakan bagian dari variabel yang mempengaruhi perilaku variabel. Dengan tipe data ini, bisa ditentukan nilai apa yang bisa disimpan didalam variabel tersebut. Dan satu variabel hanya bisa diberi satu tipe data. Ada lima jenis tipe data sesuai dengan SQL - ANSI 1993 yaitu character string, numeric, temporal, binary, dan boolean. Sedangkan Stored procedure adalah kumpulan pernyataan/procedure SQL yang disimpan dalam database. Stored procedure dapat menerima parameter, dan Anda dapat mengatur variabel, menulis pernyataan IF, dan lainnya dalam stored procedure.

Dalam praktikum ini, peserta akan belajar bagaimana membuat dan mengelola stored procedure serta mengintegrasikannya dengan aplikasi perangkat lunak. Dengan menguasai kedua praktikum ini, peserta diharapkan dapat mengembangkan perangkat lunak yang lebih efisien dan dapat diandalkan.

## **Tujuan**

1. Memahami dan Membuat Stored Procedure pada basis data
2. Mampu menggunakan tipe data sesuai kebutuhan pada Stored Procedure

# **BAB II DASAR TEORI**

## **Dasar Teori**

### **Tipe Data**

Tipe data merupakan bagian dari variabel yang mempengaruhi perilaku variabel. Dengan tipe data ini, bisa ditentukan nilai apa yang bisa disimpan didalam variabel tersebut. Dan satu variabel hanya bisa diberi satu tipe data.

Ada lima jenis tipe data sesuai dengan SQL - ANSI 1993 yaitu character string, numeric, temporal, binary, dan boolean.

1. Character String

Atribut seperti nama dan alamat direpresentasikan oleh character string. Ada 2 macam tipe data untuk merepresentasikan character string, yaitu:

* CHARACTER(<panjang>)

CHAR(<panjang>) menspesifikasikan karakter dengan panjang yang tetap. Sisa karakter yang tidak terpakai umumnya digantikan oleh padding characters (spasi).

* CHARACTER VARYING(<panjang>)

Atau VARCHAR(<panjang>) menspesifikasikan karakter dengan panjang yang fleksibel dan maksimum sesuai dengan <panjang>.

Sintak :

Type Data [(M)]

Misal :

CHAR [(M)]

CREATE TABLE contoh\_cha (cha CHAR(5), varcha

INSERT INTO contoh\_chaVARCHAR(5)); values('a ','a ');

INSERT INTO contoh\_cha values ('dunia','dunia');

INSERTINTO contoh\_cha VALUES ('basisdata','basis data') ;

1. Numeric

Data-data seperti usia dan gaji disimpan dalam bentuk angka. Penyimpanan dalam bentuk angka menggunakan tipe data numeric. Ada empat macam tipe data numeric, yaitu:

* INT, BIGINT, SMALLINT. SMALLINT mempunyai range ±2^15. INT mempunyai range ±2^31. BIGINT mempunyai range ±2^63.
* NUMERIC(<panjang>, [<presisi>]), DECIMAL(<panjang>, [<presisi>]) Mempunyai panjang karakter dan presisi (angka belakang koma) dengan range ±10^38. Sebagai contoh: NUMERIC (5,2) mempunyai panjang karakter 5 dengan tingkat presisi 2. Jadi nilai 100,52 bisa disimpan tetapi tidak untuk 1000,1. Nilai 0,00001 disimpan menjadi 0,00.
* REAL. Tingkat presisi tinggi dengan range -3,40E+38 s.d -1,18E-38 dan +1,18E-38 s.d +3,40E+38.
* FLOAT. Tingkat presisi tinggi dengan range -1,79E+308 s.d -2,23E-308; 0; +2,23E+308 s.d +1,79E+308.



Sintak :

Type Data [(M[,D])] [UNSIGNED] [ZEROFILL]

Misal :

INT/DECIMAL/FLOAT/DOUBLE[(M)] [UNSIGNED][ZEROFILL]

Keterangan :

* **M** : menunjukkan lebar karakter maksimum, jumlah digit keseluruhan
* **D** : Jumlah digit dibelakang koma
* **tanda [ dan ]** berarti pemakaiannya adalah optional

**Contoh :**



1. Temporal

Temporal merupakan tipe data yang menyimpan tanggal dan waktu yang disesuikan dengan system-timezone (komputer). Sebagai contoh data temporal adalah data tentang tanggal lahir. Ada dua macam tipe data temporal, yaitu:

* DATETIME. Tipe data ini menyimpan informasi tanggal, waktu atau bahkan keduanya. Dalam SQL Server, tipe data ini menyimpan dengan tingkat akurasi sampai 3,33 milidetik. Sedangkan untuk SMALLDATETIME hanya sampai 1 menit. Dalam tipe data ini, juga terdapat tipe data TIMESTAMP dengan tingkat akurasi sampai dengan 9 digit.
* INTERVAL. Umumnya digunakan untuk menyimpan periode seperti garansi. Ada 2 macam yaitu (1) YEAR-MONTH dan (2) DAY-TIME. SQL Server tidak mempunyai tipe data ini.



### **Stored Procedure**

Store Procedure adalah sebuah prosedur layaknya subprogram (subrtutin) di dalam bahasa pemrograman reguler yang tersimpan di dalam katalog basis data. Beberapa kelebihan yang ditawarkan stored procedure antara lain :

* meningkatkan performa.
* mereduksi trafik jaringan.
* Reusable.
* meningkatkan kontrol sekuriti.

Dibalik kelebihan tersebut, stored procedure juga memiliki kekurangan. Di anataranya adalah berpotensi meningkatkan beban server dan penulisannya tidak mudah (memerlukan pengetahuan yang spesifik).

Berikut adalah sintak pembuatan stored procedure:

****

**Aktivasi/pemanggilan stored procedure :**

****

**Menghapus Stored Procedure :**

****

**Pernyataan pembuatan stored procedure :**

DELIMITER $$

**CREATE PROCEDURE** set\_counter(**INOUT count INT**(4),**IN** inc **INT**(4)) **BEGIN**

**SET count** = **count** + inc;

**END**$$

DELIMITER ;

**Cara memanggilnya dengan mengset isi awal :**

**SET** @counter = 1;

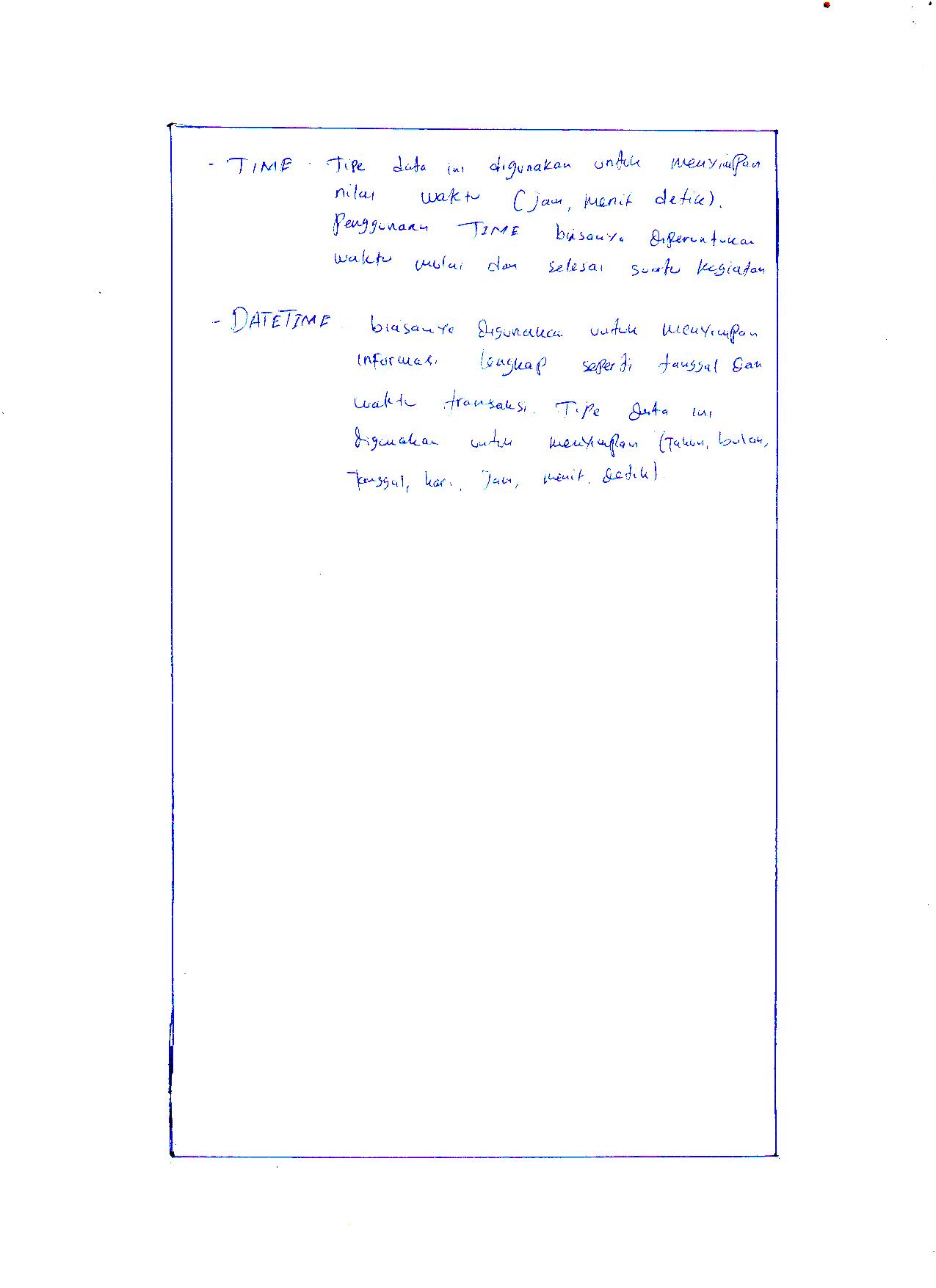
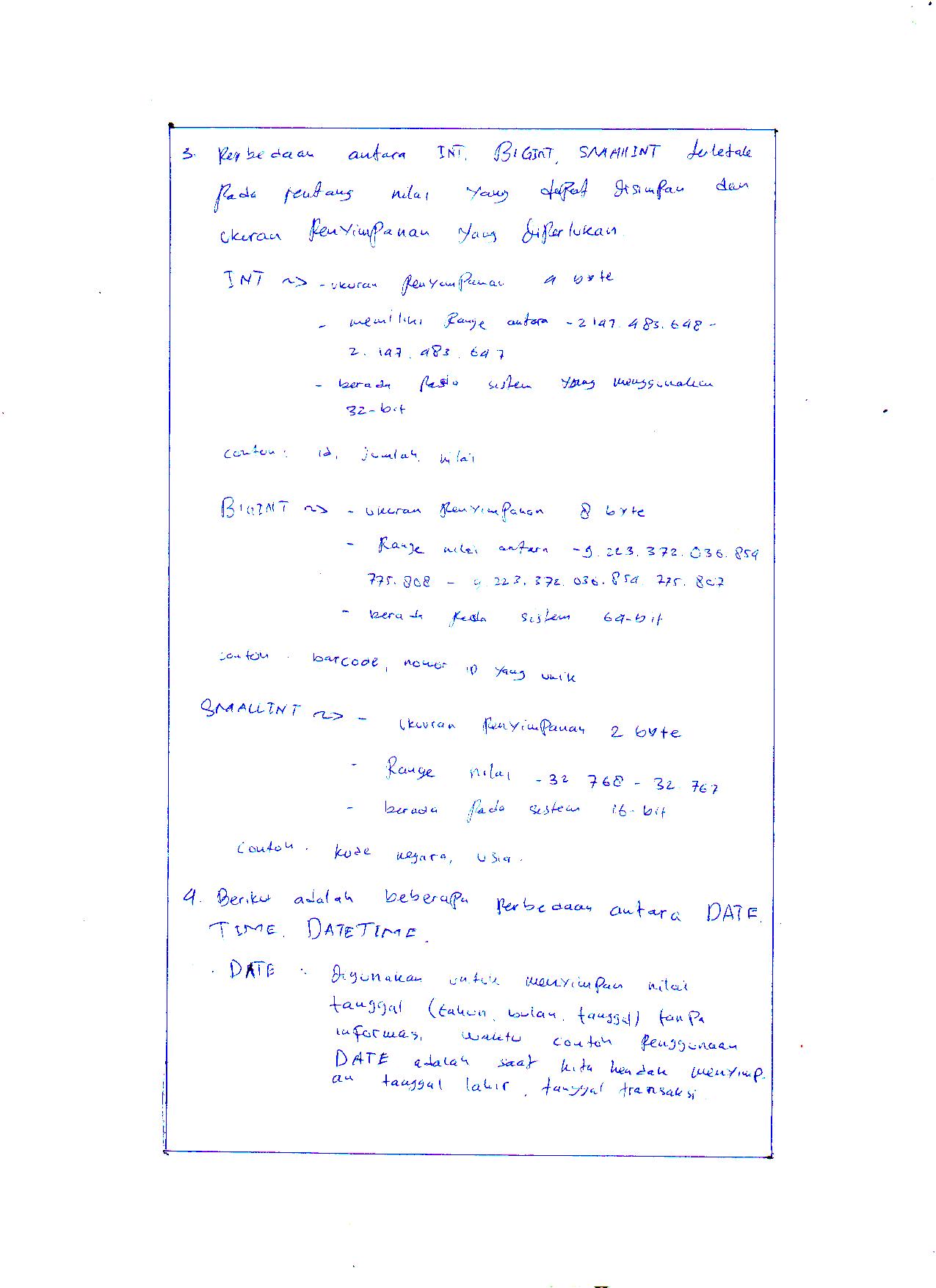
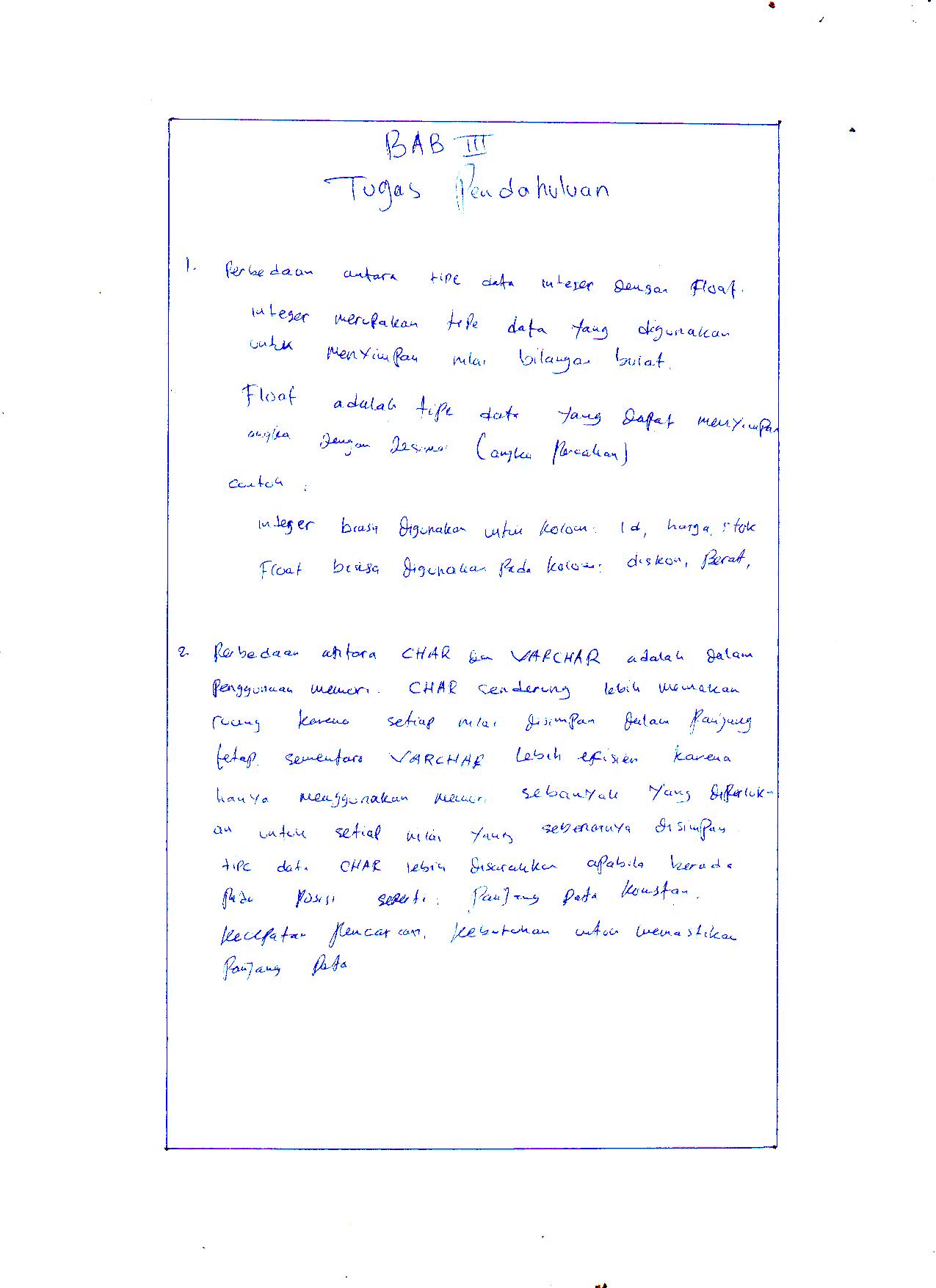
**CALL** set\_counter(@counter,1); *-- 2*

**CALL** set\_counter(@counter,2); *-- 3*

**CALL** set\_counter(@counter,9); *-- 12*

**SELECT** @counter; *-- 12*

# **BAB III TUGAS Pendahuluan**



# **BAB IV IMPLEMENTASI**

## **Implementasi**

1. Buatlah Stored Procedure untuk menghitung total harga dari semua transaksi yang dilakukan oleh seorang karyawan tertentu. Procedure harus menerapkan parameter IN dan parameter OUT.

Jawaban :

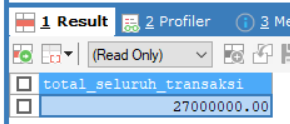
Query :

|  |
| --- |
| DELIMITER //  CREATE PROCEDURE soal11(  IN karyawan\_id INT,  OUT total\_harga DECIMAL(10, 2)  )  BEGIN  SELECT SUM(t.jumlah \* p.harga) INTO total\_harga  FROM transaksi t  JOIN produk p ON t.id\_produk = p.id\_produk  WHERE t.id\_karyawan = karyawan\_id;  END //  CALL soal11(3, @total\_harga);  SELECT @total\_harga AS total\_seluruh\_transaksi; |

Penjelasan :

Pernyataan SQL di atas adalah untuk membuat stored procedure untuk menghitung total harga transaksi yang dilakukan oleh karyawan denga ID 3. Pertama kita menggunakan paramter IN dengan nama karyawan\_id dan parameter OUT dengan nama total\_harga. Kemudian dilakukan perhitungan dengan menggunakan operasi SELECT. Dan menggunakan agregasi SUM untuk menjumlahkan harga pada tabel produk dan jumlah di tabel transaksi. Karena informasi tidak hanya ada pada satu tabel maka digunakan perintah JOIN. Dengan id\_produk yang sama pada tabel produk dan tabel transaksi dengan operasi WHERE untuk memfilter karyawan yang kita inginkan. Kemudian dilakukan pemanggilan dengan perintah CALL. Dan menampilkan hasil stored procedure dengan perintah SELECT @total\_harga.

Hasil:



2. Buatkanlah procedure untuk mengetahui berapa lama(hari) setiap produk berada digudang sejak tanggal masuk produk.

Jawaban :

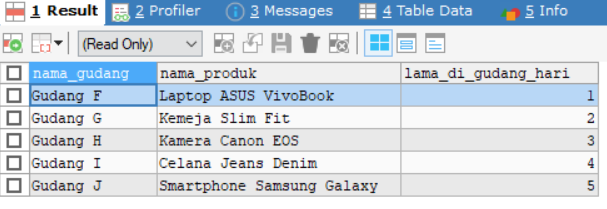
Query:

|  |
| --- |
| DELIMITER //  CREATE PROCEDURE CalculateProductDurationInWarehouse()  BEGIN  SELECT  g.nama AS nama\_gudang,  p.nama\_produk AS nama\_produk,  DATEDIFF(CURRENT\_DATE(), MAX(s.tanggal\_update)) AS lama\_di\_gudang\_hari  FROM  gudang g  INNER JOIN  stok s ON g.id\_gudang = s.id\_gudang  INNER JOIN  produk p ON s.id\_produk = p.id\_produk  GROUP BY  g.nama, p.nama\_produk;  END //  CALL CalculateProductDurationInWarehouse(); |

Penjelasan:

Pernyataan SQL di atas adalah membuat stored procedure untuk mengetahui berapa hari sebuah produk berada dalam gudang. Statement SELECT digunakan untuk mengambil data dari tabel. Kolom yang dipilih adalah nama\_gudang dari tabel gudang, nama\_produk dari tabel produk, dan perhitungan durasi menggunakan fungsi DATEDIFF. Fungsi DATEDIFF digunakan untuk menghitung selisih hari antara tanggal terkini (diperoleh melalui CURRENT\_DATE()) dan tanggal terbaru dari tanggal\_update untuk setiap produk di setiap gudang. Hasil dari DATEDIFF disebut sebagai lama\_di\_gudang\_hari.

Hasil:



3. Buatkan procedure untuk menghapus transaksi pada 1 bulan terakhir, tetapi jika total harga nya lebih dari 200 ribu maka tidak dapat di hapus.

Jawaban :

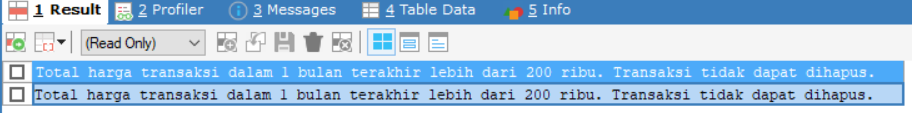
Query:

|  |
| --- |
| DELIMITER //  CREATE PROCEDURE DeleteTransactions()  BEGIN  DECLARE total\_harga DECIMAL(10, 2);    -- Menghitung total harga transaksi dalam 1 bulan terakhir  SELECT SUM(total\_harga) INTO total\_harga  FROM transaksi  WHERE tanggal\_transaksi >= DATE\_SUB(CURRENT\_DATE(), INTERVAL 1 MONTH);    -- Memeriksa apakah total harga lebih dari 200 ribu  IF total\_harga IS NOT NULL AND total\_harga <= 200000 THEN  -- Hapus transaksi dalam 1 bulan terakhir jika total harga kurang dari atau sama dengan 200 ribu  DELETE FROM transaksi  WHERE tanggal\_transaksi >= DATE\_SUB(CURRENT\_DATE(), INTERVAL 1 MONTH);  SELECT 'Transaksi dalam 1 bulan terakhir telah dihapus.';  ELSE  SELECT 'Total harga transaksi dalam 1 bulan terakhir lebih dari 200 ribu. Transaksi tidak dapat dihapus.';  END IF;  END //  DELIMITER ;  CALL DeleteTransactions(); |

Penjelasan:

Pernyataan SQL di atas adalah sebuah stored procedure dalam SQL yang bertujuan untuk menghapus transaksi yang terjadi dalam 1 bulan terakhir jika total harga transaksi dalam rentang waktu tersebut kurang dari atau sama dengan 200 ribu. Dalam BEGIN dengan END digunakan sintaks sebagai berikut yaitu Statement SELECT SUM(total\_harga) INTO total\_harga FROM transaksi WHERE tanggal\_transaksi >= DATE\_SUB(CURRENT\_DATE(), INTERVAL 1 MONTH); menghitung total harga transaksi dalam 1 bulan terakhir dan menyimpannya ke dalam variabel total\_harga. IF ... ELSE ... END IF digunakan untuk mengevaluasi kondisi tertentu. Di sini, kita memeriksa apakah total\_harga tidak kosong (IS NOT NULL) dan kurang dari atau sama dengan 200 ribu. Jika kondisi terpenuhi, maka perintah DELETE akan dijalankan untuk menghapus transaksi dalam 1 bulan terakhir. Jika tidak, pesan akan dikembalikan bahwa total harga transaksi dalam 1 bulan terakhir melebihi 200 ribu dan transaksi tidak dapat dihapus.

Hasil:



## **Penjelasan Hasil Praktikum**

Dalam praktikum ini, saya telah belajar tentang pentingnya tipe data dalam basis data dan bagaimana membuat stored procedure untuk mengelola data dengan lebih efisien. Melalui praktikum ini, saya menyadari bahwa tipe data memiliki peran yang penting dalam mengontrol perilaku variabel dan memastikan integritas data. Sementara itu, dengan membuat stored procedure, saya dapat menulis serangkaian perintah SQL yang kompleks dan dapat dijalankan secara terpisah untuk menjalankan tugas-tugas rutin atau pemrosesan data khusus. Pengalaman ini telah memberi saya pemahaman yang lebih dalam tentang penggunaan tipe data dan stored procedure dalam pengembangan basis data.

# **BAB V PENUTUP**

## **Analisa**

Dalam praktikum kali ini, praktikan dikenalkan dengan Type data/Stored Procedure. Dalam modul ini, praktikan dituntut untuk bisa membuat procedure untuk mengetahui umur buku jika dilihat dari tahun penerbitan, membuat procedure untuk menghitung berapa lama anggota meminjam sebuah buku, membuat procedure untuk mendelete daftar buku dengan ketentuan tertentu, membuat procedure untuk mengupdate jumlah buku pada tabel buku jika ada transaksi peminjaman maupun pengembalian buku, membuat procedure untuk mengetahui jumlah buku yang dipinjam oleh setiap anggota, dan juga perintah untuk menghapus procedure yang telah dibuat. Dengan tugas praktikum tersebut, praktikan dapat memahami dan membuat Stored Procedure pada basis data serta mampu menggunakan type data sesuai kebutuhan pada Stored Procedure.

## **Kesimpulan**

Dari hasil praktikum, dapat disimpulkan bahwa Type data merupakan bagian dari variabel yang mempengaruhi perilaku variabel. Dengan tipe data ini, bisa ditentukan nilai apa yang bisa disimpan didalam variabel tersebut. Dan satu variabel hanya bisa diberi satu tipe data. Ada lima jenis tipe data sesuai dengan SQL - ANSI 1993 yaitu character string, numeric, temporal, binary, dan boolean. Sedangkan Stored procedure adalah kumpulan pernyataan/procedure SQL yang disimpan dalam database. Stored procedure dapat menerima parameter, dan Anda dapat mengatur variabel, menulis pernyataan IF, dan lainnya dalam stored procedure.