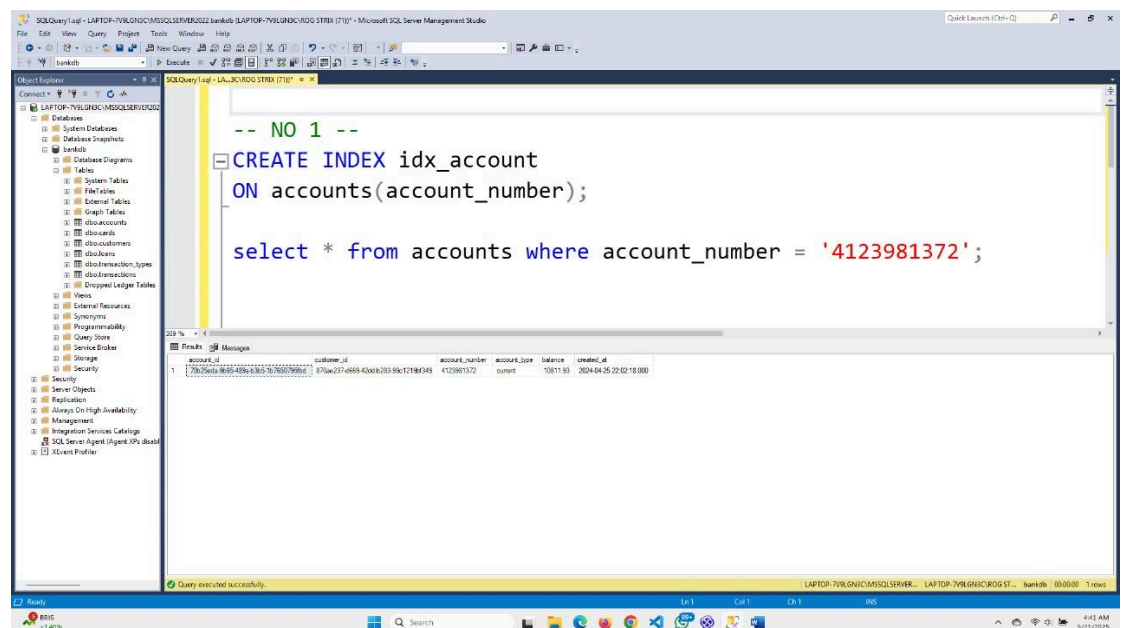


Kelompok 2

LAB A2

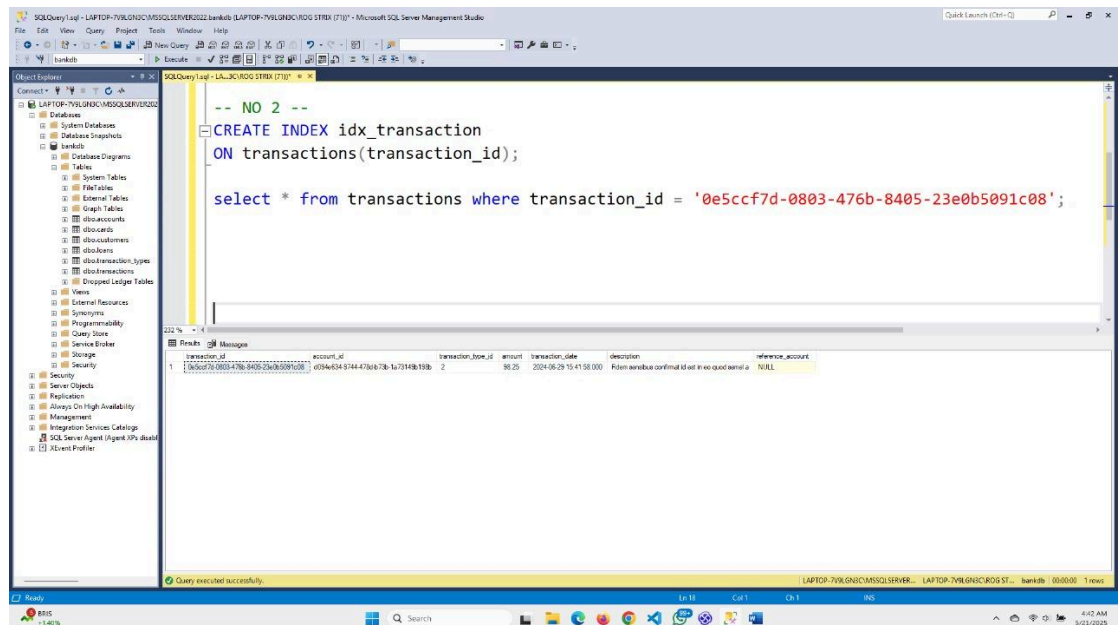
Nama : 1. Ananda Rachmawati Purwanto (24060123130061)  
2. Atika Mutia Sabila (24060123130089)  
3. Aulia Raffa Satria (24060123140050)  
4. Bramantyo Kunni Nurrisqi (24060123130091)  
5. Muhammad Irfan Irsyad (24060123130085)

1. Buatlah index pada tabel accounts untuk mempercepat pencarian. Pilih salah satu kolom pada tabel tersebut dan berikan alasan pemilihan kolom tersebut sebagai index!



Alasan: Kami memilih `account_number` sebagai index di tabel `accounts` karena `account_number` memiliki sifat unik. `account_number` juga sering digunakan untuk pencarian atau transaksi eksternal. `account_number` adalah atribut statis yang tidak berubah seiring waktu, sehingga cocok dijadikan index.

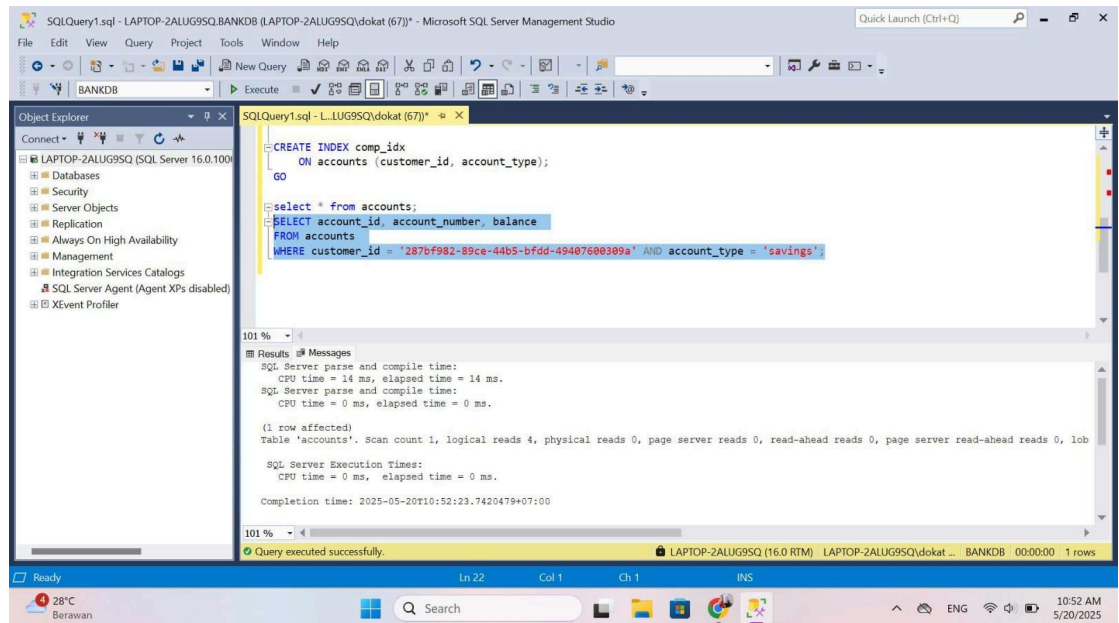
2. Buatlah index pada tabel transactions untuk mempercepat pencarian. Pilih salah satu kolom pada tabel tersebut dan berikan alasan pemilihan kolom tersebut sebagai index!



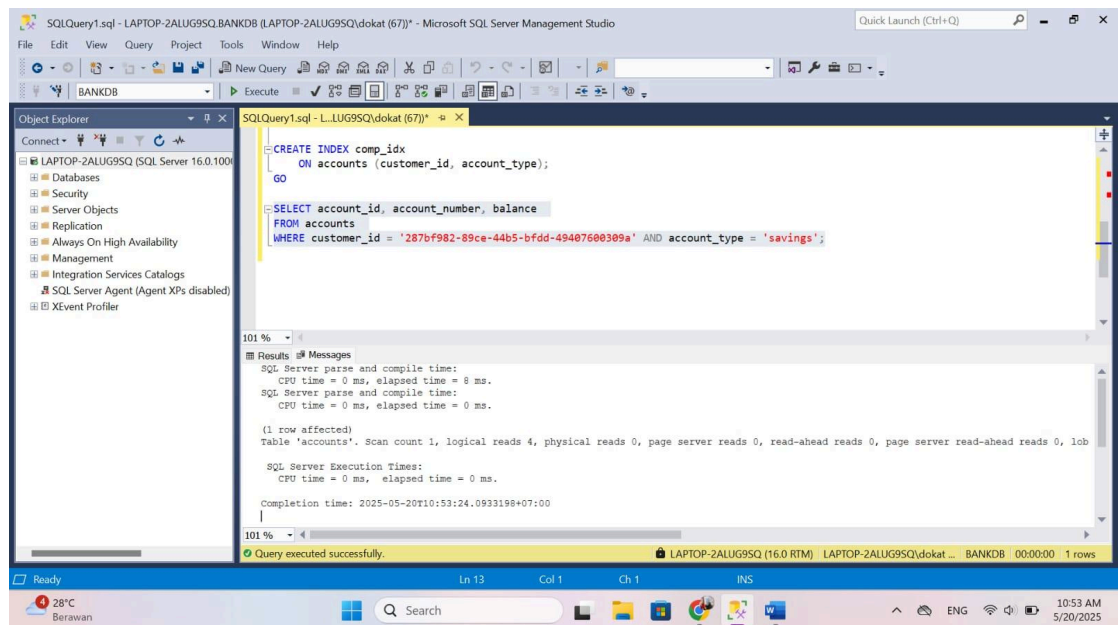
Alasan: Kami menjadikan transaction\_id sebagai index karena transaction\_id sering digunakan dalam query untuk mengakses data transaksi secara spesifik, seperti pada operasi SELECT, DELETE, dan UPDATE. Sehingga dengan adanya index dapat mempercepat proses pencarian dan meningkatkan performa database secara keseluruhan.

3. Buatlah composite index pada tabel accounts. Pilih lebih dari satu kolom pada tabel tersebut dan uji performa query sebelum dan setelah menggunakan index! (query pengujian berdasarkan kolom yang dipilih sebagai index)

Sebelum: **14 ms**



Sesudah: **8 ms**



4. Buat cursor untuk membaca semua akun dari tabel accounts, lalu periksa balance masing-masing. Jika  $\text{balance} < 9.000$ , tampilkan pesan: "Saldo rendah untuk account\_id: [account\_id]".

```
DECLARE @account_id VARCHAR (255);
DECLARE @balance DECIMAL (10,2);

DECLARE read_cursor CURSOR FOR
SELECT account_id, balance FROM accounts
WHERE balance < 9000;

OPEN read_cursor;
FETCH NEXT FROM read_cursor INTO @account_id, @balance;

WHILE @@FETCH_STATUS = 0
BEGIN
    PRINT 'Saldo rendah untuk account_id = ' + @account_id;
    FETCH NEXT FROM read_cursor INTO @account_id, @balance;
END;

CLOSE read_cursor;
DEALLOCATE read_cursor;
```

100 %

Messages

Saldo rendah untuk account\_id = 07a36a4b-7943-4634-b17a-7c7eb4407cd3  
Saldo rendah untuk account\_id = 0937445f-d5eb-4ef9-887f-4173fe662dd4  
Saldo rendah untuk account\_id = 1d97bce1-5c94-49e4-8e2a-6ad56b99afb0  
Saldo rendah untuk account\_id = 2fdf706a-7187-470c-a1c3-f3e747b72f20  
Saldo rendah untuk account\_id = 318c43bf-9511-4030-bd13-81a47a4b8cac  
Saldo rendah untuk account\_id = 34e9be76-4674-4d0f-b7c0-a39e43548747  
Saldo rendah untuk account\_id = 36f3d280-2886-4e51-adcd-dfc644e59d80  
Saldo rendah untuk account\_id = 410c132e-fc1d-4237-b410-c0dc04f5b00e

5. Buat cursor untuk membaca semua pelanggan (customers) dan gabungkan first\_name dan last\_name untuk ditampilkan dengan format: "Customer: [Nama Lengkap]".

```
DECLARE @first_name VARCHAR (255);
DECLARE @last_name VARCHAR (255);

DECLARE cust_cursor CURSOR FOR
SELECT first_name, last_name FROM customers;

OPEN cust_cursor;

FETCH NEXT FROM cust_cursor INTO @first_name, @last_name;

WHILE @@FETCH_STATUS = 0
BEGIN
    PRINT 'Customer: ' + @first_name + ' ' + @last_name;
    FETCH NEXT FROM cust_cursor INTO @first_name, @last_name;
END;

CLOSE cust_cursor;
DEALLOCATE cust_cursor;
```

100 %

Messages

Customer: Sergeant Maith  
Customer: Shaine Matic  
Customer: Buddie Sandison  
Customer: Stephanus Frear  
Customer: Corette Vicent  
Customer: Devon McQuode  
Customer: Jacquenette Hilldrup

100 %