TUGAS KELOMPOK PRAKTIKUM MANAJEMEN BASIS DATA PERTEMUAN-5 TENTANG KOMPILASI QUERY



Asisten Praktikum:

Alya Safina & Izazava Putri Asari

Disusun oleh Kelompok 6 Lab A2:

Aditya Arif Setiawan (24060121140144)

Alfonso Michael N H (24060123140156)

Anggita Mahardika Syarifa (24060119130089)

DEPARTEMEN ILMU KOMPUTER/INFORMATIKA FAKULTAS SAINS DAN MATEMATIKA UNIVERSITAS DIPONEGORO 2025

TOPIK 6 Kompilasi Query

1. iii. Buat view vw_account_card_transaction_detail yang menggabungkan data akun, kartu, dan transaksi yang terkait.

contoh:

Account Id	Account Number	Account Type	Balance	Card Number	Card Type	Transact ion Id	Transact ion Type	Amount
abcd	1234	savings	8000	4111	debit	0001	deposit	7500

```
Pert-05_Kel-06.sql...7F6QIKPV\adity (69)) + X
   ⊟-- iii. Detail Gabungan Akun, Kartu, dan Transaksi
     -- Drop view jika sudah ada
   FIF OBJECT_ID('vw_account_card_transaction_detail', 'V') IS NOT NULL
      DROP VIEW vw_account_card_transaction_detail;
     -- Insert ke transaction_types (jika belum ada)
   ☐ IF NOT EXISTS (SELECT 1 FROM transaction_types WHERE transaction_type_id IN (1, 2, 3))
   BEGIN
        INSERT INTO transaction_types (transaction_type_id, name) VALUES
         (1, 'deposit'),
         (2, 'transfer'),
         (3, 'withdrawal');
     -- Insert 1 dummy customer (karena account_id butuh relasi customer_id)
     DECLARE @customer_id CHAR(36) = NEWID();
   ☐ INSERT INTO customers (customer_id, first_name, last_name, email, phone_number, address)
     VALUES (
         @customer_id, 'Dummy', 'User', 'dummy@email.com', '081234567890', '123 Main Street');
     -- Insert dummy account
     DECLARE @account_id CHAR(36) = NEWID();
   INSERT INTO accounts (account_id, customer_id, account_number, account_type, balance)
81 % ~
Messages
   Commands completed successfully.
   Completion time: 2025-05-27T11:40:56.6593343+07:00
```

```
Pert-05 Kel-06.sql...7F6QIKPV\adity (69)) = X
   🖃 -- iii. Detail Gabungan Akun, Kartu, dan Transaksi
    -- Drop view jika sudah ada
   IF OBJECT_ID('vw_account_card_transaction_detail', 'V') IS NOT NULL
        DROP VIEW vw_account_card_transaction_detail;
     -- Insert ke transaction_types (jika belum ada)
   ☐IF NOT EXISTS (SELECT 1 FROM transaction_types WHERE transaction_type_id IN (1, 2, 3))
   BEGIN
        INSERT INTO transaction_types (transaction_type_id, name) VALUES
        (1, 'deposit'),
        (2, 'transfer'),
        (3, 'withdrawal');
     -- Insert 1 dummy customer (karena account_id butuh relasi customer_id)
     DECLARE @customer_id CHAR(36) = NEWID();
   JINSERT INTO customers (customer_id, first_name, last_name, email, phone_number, address)
     VALUES (
         @customer_id, 'Dummy', 'User', 'dummy@email.com', '081234567890', '123 Main Street');
     -- Insert dummy account
    DECLARE @account_id CHAR(36) = NEWID();
   ☐INSERT INTO accounts (account id, customer id, account number, account type, balance)
    VALUES /
81 %
Messages
   Commands completed successfully.
   Completion time: 2025-05-27T11:42:23.0837441+07:00
```

```
Pert-05_Kel-06.sql...7F6QIKPV\adity (69)) + X
      -- Insert 1 dummy customer (karena account_id butuh relasi customer_id)
      DECLARE @customer_id CHAR(36) = NEWID();
     INSERT INTO customers (customer_id, first_name, last_name, email, phone_number, address)
      VALUES (
        @customer_id, 'Dummy', 'User', 'dummy@email.com', '081234567890', '123 Main Street');
      -- Insert dummy account
      DECLARE @account_id CHAR(36) =
      DECLARE @account_id CHAR(36) = NEWID();
INSERT INTO accounts (account_id, customer_id, account_number, account_type, balance)
      VALUES (
        @account_id, @customer_id, '1234567890', 'savings', 10000.00
      - Insert dummy card
      DECLARE @card_id CHAR(36)
      INSERT INTO cards (card_id, account_id, card_number, card_type, expiration_date)
      VALUES (
        @card_id, @account_id, '4111222233334444', 'debit', '2026-12-31'
      -- Insert dummy transaction
DECLARE @transaction_id CHAR(36) = NEWID();
      INSERT INTO transactions (transaction_id, account_id, transaction_type_id, amount, transaction_date, description, reference_account)
      VALUES (
81 %
Messages
   (1 row affected)
   (1 row affected)
   Completion time: 2025-05-27T11:43:38.4793842+07:00
```

```
Pert-05_Kel-06.sql...7F6QIKPV\adity (69)) 🕒 🔀
                transactions (transaction_id, account_id, transaction_type_id, amo
    VALUES (
        @transaction_id, @account_id, 1, 7500.00, GETDATE(), 'Setoran awal', NULL
    );
     -- Buat ulang view
   CREATE VIEW vw_account_card_transaction_detail AS
     SELECT
         a.account_id AS [Account Id],
        a.account_number AS [Account Number],
         a.account_type AS [Account Type],
         a.balance AS [Balance],
        c.card_number AS [Card Number],
        c.card_type AS [Card Type],
        t.transaction_id AS [Transaction Id],
        tt.name AS [Transaction Type],
         t.amount AS [Amount]
     FROM accounts a
     JOIN cards c ON a.account_id = c.account_id
     JOIN transactions t ON a.account_id = t.account_id
     JOIN transaction_types tt ON t.transaction_type_id = tt.transaction_type_id;
     -- Tampilkan hasil view
     SELECT * FROM vw_account_card_transaction_detail;
81 %
Commands completed successfully.
   Completion time: 2025-05-27T11:44:05.4795613+07:00
```

```
Pert-05_Kel-06.sql...7F6QIKPV\adity (69)) - X
     VALUES (
          @transaction_id, @account_id, 1, 7500.00, GETDATE(), 'Setoran awal', NULL
      -- Buat ulang view
    CREATE VIEW vw_account_card_transaction_detail_AS
          a.account_id AS [Account Id],
          a.account_number AS [Account Number],
a.account_type AS [Account Type],
          a.balance AS [Balance],
c.card_number AS [Card Number],
c.card_type AS [Card Type],
          t.transaction_id AS [Transaction Id],
tt.name AS [Transaction Type],
          t.amount AS [Amount]
     FROM accounts a

JOIN cards c ON a.account_id = c.account_id
     JOIN transactions t ON a.account_id = t.account_id
JOIN transaction_types tt ON t.transaction_type_id = tt.transaction_type_id;
     -- Tampilkan hasil view

SELECT * FROM vw_account_card_transaction_detail;
Results Messages
                                                    Account Number
                                                                     Account Type Balance
      FBDD3C42-207F-4AE6-8E75-8EBD2C9A6F20 1234567890
                                                                                     10000.00 4111222233334444 debit
                                                                                                                               723F15CA-0F7B-402E-BCA3-D83AC7E19EBC Deposit
                                                                      savings
                                                                      savings
      7A466116-6C20-453C-B5D9-3E1FF56107DD
                                                  1234567890
                                                                                     10000.00 4111222233334444 debit
                                                                                                                               E049C075-F0C2-4FC5-8809-61D54753534C
                                                                                                                                                                            Deposit
                                                                                                                                                                                               7500.00
```

2. iv. Buat procedure sp_get_transactions_report dengan parameter opsional @account_id, @date_from, @date_to. Jika parameter tidak diisi, gunakan default (misal 30 hari terakhir semua akun).

```
Pert-05_Kel-06.sql...7F6QIKPV\adity (69)) + X
     -- iv: Laporan Transaksi Dinamis
   ☐ IF OBJECT_ID('sp_get_transactions_report', 'P') IS NOT NULL
         DROP PROCEDURE sp_get_transactions_report;
     GO
   □ CREATE PROCEDURE sp_get_transactions_report
         @account_id CHAR(36) = NULL,
         @date_from DATE = NULL,
         @date_to DATE = NULL
     AS
   ⊨BEGIN
         -- Default: 30 hari terakhir jika tidak diisi
         IF @date_from IS NULL SET @date_from = DATEADD(DAY, -30, GETDATE());
         IF @date_to IS NULL SET @date_to = GETDATE();
         -- Tampilkan transaksi
         SELECT *
         FROM transactions
         WHERE (@account_id IS NULL OR account_id = @account_id)
           AND transaction_date BETWEEN @date_from AND @date_to;
     END;
     GO
81 %
Messages
   Commands completed successfully.
   Completion time: 2025-05-27T11:47:28.0354019+07:00
```

```
Pert-05_Kel-06.sql...7F6QIKPV\adity (69)) + X
     -- iv: Laporan Transaksi Dinamis
   ☐ IF OBJECT_ID('sp_get_transactions_report', 'P') IS NOT NULL
        DROP PROCEDURE sp_get_transactions_report;
     GO
   ☐ CREATE PROCEDURE sp_get_transactions_report
        @account_id CHAR(36) = NULL,
        @date_from DATE = NULL,
        @date_to DATE = NULL
     AS
   BEGIN
         -- Default: 30 hari terakhir jika tidak diisi
        IF @date_from IS NULL SET @date_from = DATEADD(DAY, -30, GETDATE());
        IF @date_to IS NULL SET @date_to = GETDATE();
        -- Tampilkan transaksi
        SELECT *
        FROM transactions
        WHERE (@account_id IS NULL OR account_id = @account_id)
          AND transaction_date BETWEEN @date_from AND @date_to;
    END;
     GO
     -- Contoh pemanggilan: Semua transaksi 30 hari terakhir
   EXEC on got transactions renort.
81 %
Messages
   Commands completed successfully.
   Completion time: 2025-05-27T11:47:57.5661897+07:00
```

```
Pert-05_Kel-06.sql...7F6QIKPV\adity (69)) = ×
   GO
         DROP PROCEDURE sp_get_transactions_report;
   \begin{tabular}{ll} $\sqsubseteq$ CREATE PROCEDURE sp\_get\_transactions\_report \end{tabular}
        @account_id CHAR(36) = NULL,
         @date_from DATE = NULL,
        @date_to DATE = NULL
   BEGIN
         -- Default: 30 hari terakhir jika tidak diisi
         IF @date_from IS NULL SET @date_from = DATEADD(DAY, -30, GETDATE());
         IF @date_to IS NULL SET @date_to = GETDATE();
         -- Tampilkan transaksi
         FROM transactions
         WHERE (@account_id IS NULL OR account_id = @account_id)
          AND transaction_date BETWEEN @date_from AND @date_to;
    END;
      - Contoh pemanggilan: Semua transaksi 30 hari terakhir
   EXEC sp_get_transactions_report;
81 % 🔻 🖣
Results Messages
                                           account_id transaction_type_id amount
                                                                                                     description
                                                                                                                 reference_account
                                                         ________ апточти uransaction_date description referer

5000.00 2025-05-21 14:21:20.357 Test berhasil NULL
     382D1116-3045-4493-B11B-40E93F828FFE ACC001 3
      9BA28F66-94B5-4017-8404-31AFDE661B9B ACC001 3
                                                                     150000.00 2025-05-21 14:20:54.777 Test gagal NULL
```

 vi. Buat function yang mengembalikan statistik berikut dalam satu record: total akun, total saldo, rata-rata saldo untuk customer_id tertentu (gunakan RETURN table).
 Contoh:

Customer Id	Total Accounts	Total Balance	Average Balance
abcd	3	43500000.00	14500000.00

```
Pert-05_Kel-06.sql...7F6QIKPV\adity (69)) 	₱ ×
     -- vi: Statistik Akun dan Saldo Customer
   ☐ IF OBJECT_ID('fn_customer_account_stats', 'IF') IS NOT NULL
        DROP FUNCTION fn_customer_account_stats;
     GO
   GREATE FUNCTION fn_customer_account_stats (@customer_id CHAR(36))
     RETURNS TABLE
     AS
     RETURN (
        SELECT
             @customer_id AS customer_id,
             COUNT(*) AS total_accounts,
             SUM(balance) AS total_balance,
             AVG(balance) AS average_balance
         FROM accounts
        WHERE customer_id = @customer_id
    );
    GO
     -- Contoh pemanggilan (gunakan ID yang valid)

☐SELECT TOP 1 customer_id FROM customers;

   ⇒SELECT *
     FROM fn_customer_account_stats('079cd6ce-04a8-4c55-8c2b-b93189e9050a');
        ▼ 4
81 %
Commands completed successfully.
   Completion time: 2025-05-27T11:53:46.0494789+07:00
```

```
Pert-05_Kel-06.sql...7F6QIKPV\adity (69)) + X
    -- vi: Statistik Akun dan Saldo Customer
   ☐ IF OBJECT_ID('fn_customer_account_stats', 'IF') IS NOT NULL
         DROP FUNCTION fn_customer_account_stats;
     GO
   ☐ CREATE FUNCTION fn_customer_account_stats (@customer_id CHAR(36))
     RETURNS TABLE
     AS
     RETURN (
         SELECT
             @customer_id AS customer_id,
            COUNT(*) AS total_accounts,
             SUM(balance) AS total_balance,
             AVG(balance) AS average_balance
         FROM accounts
         WHERE customer_id = @customer_id
     -- Contoh pemanggilan (gunakan ID yang valid)

□ SELECT TOP 1 customer_id FROM customers;
   ⇒SELECT *
     FROM fn_customer_account_stats('079cd6ce-04a8-4c55-8c2b-b93189e9050a');
81 %
Messages
   Commands completed successfully.
   Completion time: 2025-05-27T11:54:16.8194797+07:00
```

```
Pert-05_Kel-06.sql...7F6QIKPV\adity (69)) + ×
    -- vi: Statistik Akun dan Saldo Customer
   ☐ IF OBJECT_ID('fn_customer_account_stats', 'IF') IS NOT NULL
        DROP FUNCTION fn_customer_account_stats;
    GO
   GREATE FUNCTION fn_customer_account_stats (@customer_id CHAR(36))
    RETURNS TABLE
    RETURN (
        SELECT
            @customer_id AS customer_id,
            COUNT(*) AS total_accounts,
            SUM(balance) AS total_balance,
            AVG(balance) AS average_balance
         FROM accounts
        WHERE customer_id = @customer_id
    );
    GO
    -- Contoh pemanggilan (gunakan ID yang valid)
   SELECT TOP 1 customer_id FROM customers;
   ⇒SELECT *
    FROM fn_customer_account_stats('079cd6ce-04a8-4c55-8c2b-b93189e9050a');
81 %
     Results Messages
     customer id
     079cd6ce-04a8-4c55-8c2b-b93189e9050a
```

```
-- vi: Statistik Akun dan Saldo Customer
   ☐IF OBJECT_ID('fn_customer_account_stats', 'IF') IS NOT NULL
        DROP FUNCTION fn_customer_account_stats;
    GO
   CREATE FUNCTION fn_customer_account_stats (@customer_id CHAR(36))
    RETURNS TABLE
    AS
    RETURN (
        SELECT
            @customer_id AS customer_id,
            COUNT(*) AS total_accounts,
            SUM(balance) AS total_balance,
            AVG(balance) AS average_balance
        FROM accounts
        WHERE customer_id = @customer_id
    );
    GO
    -- Contoh pemanggilan (gunakan ID yang valid)

☐ SELECT TOP 1 customer_id FROM customers;

   SELECT *
    FROM fn_customer_account_stats('079cd6ce-04a8-4c55-8c2b-b93189e9050a');
81 %
     Results Messages
     customer id
                                    total_accounts total_balance average_balance
     079cd6ce-04a8-4c55-8c2b-b93189e9050a 3
                                               588000.00
                                                          196000.000000
```

4. viii. Buat cursor yang membaca semua akun credit, lalu kurangi saldo mereka 2% sebagai biaya bulanan.

```
Pert-05_Kel-06.sql...7F6QIKPV\adity (69))* + X
   😑-- viii. Potongan 2% Saldo Akun Kredit (dengan pengecekan sebelum dan sesudah)
     -- Akun credit dummy
     DECLARE @customer_id CHAR(36) = (SELECT TOP 1 customer_id FROM customers);
   ☐ IF @customer_id IS NULL
   ⊟BEGIN
         SET @customer_id = NEWID();
         INSERT INTO customers (customer_id, first_name, last_name, email, phone_number, address)
        VALUES (@customer_id, 'Credit', 'User', 'credit@example.com', '08999999999', 'Jl. Kredit No. 1');
     -- Tambahkan akun kredit (jika belum ada)
     DECLARE @credit_account_id CHAR(36) = NEWID();
   ☐ IF NOT EXISTS (SELECT 1 FROM accounts WHERE account_type = 'credit')
        INSERT INTO accounts (account_id, customer_id, account_number, account_type, balance)
        VALUES (
            @credit_account_id, @customer_id, '9999999999', 'credit', 200000.00
     -- Simpan kondisi sebelum pemotongan
     IF OBJECT_ID('tempdb..#before_cut') IS NOT NULL DROP TABLE #before_cut;
   SELECT account_id, balance AS balance_before
    INTO #before_cut
     FROM accounts
     WHERE account type - 'chedit'
81 %
Messages
   Commands completed successfully.
   Completion time: 2025-05-27T12:09:08.9255158+07:00
```

```
Pert-05_Kel-06.sql...7F6QIKPV\adity (69))* 		□ ×
    -- Simpan kondisi sebelum pemotongan
    IF OBJECT_ID('tempdb..#before_cut') IS NOT NULL DROP TABLE #before_cut;
   INTO #before_cut
    FROM accounts
    WHERE account_type = 'credit';
     -- Tampilkan sebelum potong
    SELECT * FROM #before_cut;
    -- Jalankan cursor untuk potong 2%
   □DECLARE credit_cursor CURSOR FOR
        SELECT account_id, balance FROM accounts WHERE account_type = 'credit';
    DECLARE @account_id CHAR(36), @balance DECIMAL(18,2);
    OPEN credit cursor;
    FETCH NEXT FROM credit_cursor INTO @account_id, @balance;
   ⊨WHILE @@FETCH_STATUS = 0
   ⊟BEGIN
        UPDATE accounts
        SET balance = balance - (@balance * 0.02)
        WHERE account_id = @account_id;
        EFTCH NEVT FROM chadit curson TNTO Account id Abalanca
      ▼ 4
81 %
Results Messages
     account_id
                                       balance_before
     0A3E91AA-697D-4B76-AABC-78A649A90918 188238.40
     3CDF3899-56AF-4519-809E-C0AF9EA4575C 188238.40
3
     9795E48B-346B-443E-8834-041198D51016
                                       192080.00
     DA7CA675-DD19-4A61-A252-F690AB26A76E 188238.40
```

```
Pert-05_Kel-06.sql...7F6QIKPV\adity (69))* = X
     -- Jalankan cursor untuk potong 2%
   DECLARE credit cursor CURSOR FOR
         SELECT account_id, balance FROM accounts WHERE account_type = 'credit';
     DECLARE @account_id CHAR(36), @balance DECIMAL(18,2);
     OPEN credit cursor;
     FETCH NEXT FROM credit_cursor INTO @account_id, @balance;
   ⊢WHILE @@FETCH_STATUS = 0
   BEGIN
         UPDATE accounts
        SET balance = balance - (@balance * 0.02)
       WHERE account_id = @account_id;
        FETCH NEXT FROM credit_cursor INTO @account_id, @balance;
     END;
     CLOSE credit_cursor;
    DEALLOCATE credit_cursor;
     -- Tampilkan sesudah potong
   SELECT
        a.account_id,
        b.balance_before,
        a halanca AS halanca after
81 %
Messages
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   Completion time: 2025-05-27T12:10:50.1638011+07:00
```

```
Pert-05_Kel-06.sql...7F6QIKPV\adity (69))* + X
    OPEN credit_cursor;
    FETCH NEXT FROM credit_cursor INTO @account_id, @balance;
   ⊟WHILE @@FETCH_STATUS = 0
   ĖBEGIN
        UPDATE accounts
        SET balance = balance - (@balance * 0.02)
        WHERE account_id = @account_id;
        FETCH NEXT FROM credit_cursor INTO @account_id, @balance;
    END;
    CLOSE credit_cursor;
    DEALLOCATE credit_cursor;
     -- Tampilkan sesudah potong
   SELECT
        a.account_id,
        b.balance_before,
        a.balance AS balance_after,
         (b.balance_before - a.balance) AS amount_deducted
     FROM accounts a
     JOIN #before_cut b ON a.account_id = b.account_id
     WHERE a.account_type = 'credit';
81 %
        ▼ 4
Results Messages
      account_id
                                          balance_before balance_after amount_deducted
     0A3E91AA-697D-4B76-AABC-78A649A90918 188238.40
                                                       184473.63
                                                                   3764.77
2
     3CDF3899-56AF-4519-809E-C0AF9EA4575C 188238.40
                                                       184473.63
                                                                   3764.77
      9795E48B-346B-443E-8834-041198D51016
                                          192080.00
                                                       188238.40
                                                                   3841.60
      DA7CA675-DD19-4A61-A252-F690AB26A76E 188238.40
                                                       184473.63
                                                                   3764.77
```