



Common Table Expression

Virtual Internship Experience



Outline

- 1 Konsep Dasar CTE**
- 2 Manfaat CTE**
- 3 Non-Recursive CTE**
- 4 Recursive CTE**

Disclaimer

“Dokumen ini memiliki hak cipta. Barang siapa yang menyebarluaskan atau menduplikasi tanpa izin dari instansi terkait dapat diproses sesuai dengan ketentuan hukum yang berlaku.”

Common Table Expression (CTE)

salah satu bentuk query SQL yang digunakan untuk **menyederhanakan JOIN** pada SQL kedalam subqueries dan mampu memberikan query yang bersifat **hierarki**. CTE dikenal dengan istilah Hierarchical and recursive queries in SQL



Manfaat CTE

- **Readability** - Penggunaan CTE memudahkan developer dalam membaca query kompleks yang telah dibuat
- **Recursion** - CTE mendukung pembuatan recursive queries, dimana sebuah query dapat memanggil dirinya sendiri.
- **Ranking** - kapanpun ingin menggunakan fungsi ranking maka dapat menggunakan ROW_NUMBER(), RANK(), NTILE(), dan sebagainya

Jenis-Jenis CTE

1. Non-Recursive CTE

CTE ini **tidak** menggunakan **pengulangan** dan tidak mereferensikan diri sendiri layaknya sebuah view yang sering kita buat dalam query SQL.

2. Recursive CTE

CTE yang mereferensikan dirinya sendiri dapat disebut sebagai Recursive CTE. Sejalan dengan dieksekusinya query, SQL akan mengulang set data yang dihasilkan CTE **sampai pada suatu kondisi terpenuhi**.

Non-Recursive CTE

```
WITH           CTE name
with engineers as (
    select *
    from employees
    where dept='Engineering'
)
select *
from engineers      CTE Usage
```

The diagram illustrates the structure of a Non-Recursive Common Table Expression (CTE). It shows the SQL code with annotations:

- WITH**: Points to the keyword **WITH** at the start of the CTE definition.
- CTE name**: Points to the alias **engineers** used in the **as** clause.
- CTE Body**: Points to the enclosed query block between the parentheses, which selects all columns from the **employees** table where the department is 'Engineering'.
- CTE Usage**: Points to the reference to the CTE alias **engineers** in the final **select** statement.

Recursive CTE

```
Declare @RowNo int =1;  
;with ROWCTE as  
(  
    SELECT @RowNo as ROWNO  
    UNION ALL  
    SELECT ROWNO+1  
    FROM ROWCTE  
    WHERE RowNo < 10  
)  
  
SELECT * FROM ROWCTE
```



ROWNO	ROWNO
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10

Recursive CTE

```
WITH cte_numbers(n, weekday)
AS (
    SELECT
        0,
        DATENAME(DW, 0)
    UNION ALL
    SELECT
        n + 1,
        DATENAME(DW, n + 1)
    FROM
        cte_numbers
    WHERE n < 6
)
SELECT
    weekday
FROM
    cte_numbers;
```



weekday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday

Implementation in Bank Muamalat



Source



<https://www.sqlshack.com/sql-server-common-table-expressions-cte/>

<https://www.sqlservertutorial.net/sql-server-basics/sql-server-recursive-cte/>

<https://www.belajararief.com/index.php/tulisan/tekno/214-common-table-expression-cte>

<https://mariadb.com/kb/en/non-recursive-common-table-expressions-overview/>



Assets



Logo Putih



Logo Berwarna



Thank You!

