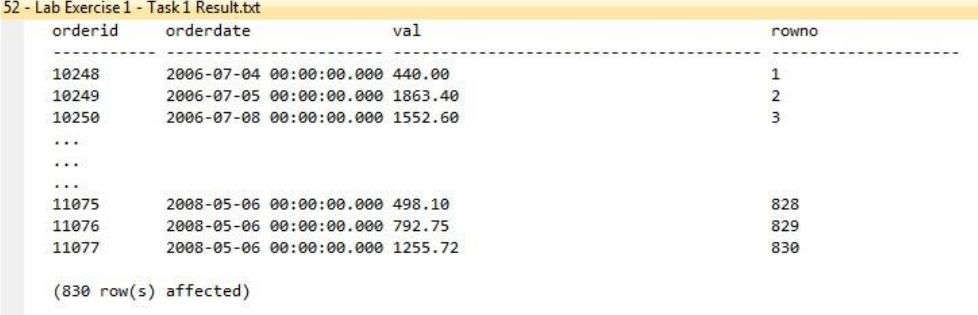



Nama : Maulana Arya Putra Nugraha

Kelas : TI 2D

NIM : 2241720199

### Praktikum – Bagian 1: Menulis Query Menggunakan Fungsi RANKING

Langkah	Keterangan
1	<p>Skenario :</p> <p>Bagian penjualan ingin menentukan pesanan berdasarkan nilai masing-masing pelanggan. Untuk itu diperlukan laporan menggunakan fungsi RANK (termasuk kolom hasil perhitungan yang menambahkan kolom hasil perhitungan untuk menampilkan nomor baris dengan klausa SELECT).</p> <p>Untuk melakukan percobaan pada praktikum bagian 1 ini terlebih dahulu lakukan login pada SQL Server Management Studio (SSMS). Pastikan database terhubung dengan "TSQL".</p>
2	<p><b>[Soal-1]</b> Tulislah pernyataan SELECT untuk mengambil kolom orderid, orderdate, dan val serta kolom hasil perhitungan bernama rowno dari view Sales.OrderValues! Gunakan fungsi ROW_NUMBER untuk mengembalikan rowno, urutkan nomor baris berdasarkan kolom orderdate!</p>   <pre>SELECT orderid, orderdate, val,        ROW_NUMBER() OVER (ORDER BY orderdate) AS rowno FROM Sales.OrderValues;</pre>

Nama : Maulana Arya Putra Nugraha

Kelas : TI 2D

NIM : 2241720199

100 %

Results Messages

	orderid	orderdate	val	rowno
1	10248	2006-07-04 00:00:00.000	440.00	1
2	10249	2006-07-05 00:00:00.000	1863.40	2
3	10250	2006-07-08 00:00:00.000	1552.60	3
4	10251	2006-07-08 00:00:00.000	654.06	4
5	10252	2006-07-09 00:00:00.000	3597.90	5
6	10253	2006-07-10 00:00:00.000	1444.80	6
7	10254	2006-07-11 00:00:00.000	556.62	7
8	10255	2006-07-12 00:00:00.000	2490.50	8
9	10256	2006-07-15 00:00:00.000	517.80	9
10	10257	2006-07-16 00:00:00.000	1119.90	10
11	10258	2006-07-17 00:00:00.000	1614.88	11

**[Soal-2]** Salin T-SQL pada soal no 1. Kemudian modifikasi dengan memasukkan kolom tambahan bernama rankno. Untuk membuat rankno gunakan fungsi RANK dengan urutan peringkat berdasarkan kolom orderdate!

3

53 - Lab Exercise1 - Task 2 Result.txt

orderid	orderdate	val	rowno	rankno
10248	2006-07-04 00:00:00.000	440.00	1	1
10249	2006-07-05 00:00:00.000	1863.40	2	2
10250	2006-07-08 00:00:00.000	1552.60	3	3
...				
...				
11075	2008-05-06 00:00:00.000	498.10	828	827
11076	2008-05-06 00:00:00.000	792.75	829	827
11077	2008-05-06 00:00:00.000	1255.72	830	827

(830 row(s) affected)

Nama : Maulana Arya Putra Nugraha

Kelas : TI 2D

NIM : 2241720199



	orderid	orderdate	val	rowno	rankno
1	10248	2006-07-04 00:00:00.000	440.00	1	1
2	10249	2006-07-05 00:00:00.000	1863.40	2	2
3	10250	2006-07-08 00:00:00.000	1552.60	3	3
4	10251	2006-07-08 00:00:00.000	654.06	4	3
5	10252	2006-07-09 00:00:00.000	3597.90	5	5
6	10253	2006-07-10 00:00:00.000	1444.80	6	6
7	10254	2006-07-11 00:00:00.000	556.62	7	7
8	10255	2006-07-12 00:00:00.000	2490.50	8	8
9	10256	2006-07-15 00:00:00.000	517.80	9	9
10	10257	2006-07-16 00:00:00.000	1119.90	10	10
11	10258	2006-07-17 00:00:00.000	1614.88	11	11

```
SELECT orderid, orderdate, val,  
       ROW_NUMBER() OVER (ORDER BY orderdate) AS rowno,  
       RANK() OVER (ORDER BY orderdate) AS rankno  
FROM Sales.OrderValues;
```

100 %

Results Messages

	orderid	orderdate	val	rowno	rankno
1	10248	2006-07-04 00:00:00.000	440.00	1	1
2	10249	2006-07-05 00:00:00.000	1863.40	2	2
3	10250	2006-07-08 00:00:00.000	1552.60	3	3
4	10251	2006-07-08 00:00:00.000	654.06	4	3
5	10252	2006-07-09 00:00:00.000	3597.90	5	5
6	10253	2006-07-10 00:00:00.000	1444.80	6	6
7	10254	2006-07-11 00:00:00.000	556.62	7	7
8	10255	2006-07-12 00:00:00.000	2490.50	8	8
9	10256	2006-07-15 00:00:00.000	517.80	9	9
10	10257	2006-07-16 00:00:00.000	1119.90	10	10
11	10258	2006-07-17 00:00:00.000	1614.88	11	11

**4** [Soal-3] Apakah perbedaan antara fungsi RANK dan fungsi ROW\_NUMBER?

- Perbedaan dari keduanya terletak pada cara mereka menangani nilai yang sama dalam data yang diurutkan: ROW\_NUMBER() memberikan nomor baris unik untuk setiap baris, sementara RANK() memberikan peringkat yang sama untuk nilai yang sama dan kemudian melompati peringkat berikutnya setelah nilai yang sama.

Nama : Maulana Arya Putra Nugraha

Kelas : TI 2D

NIM : 2241720199

[Soal-4] Tuliskan pernyataan SELECT untuk mengambil kolomorderid, orderdate, custid, dan val serta hitung kolom bernama orderrankno dari view Sales.OrderValues. Kolom orderrankno harus menampilkan rangking per pelanggan secara independen, berdasarkan pemesanan val dalam urutan menurun!

54 - Lab Exercise 1 - Task 3 Result.txt

orderid	orderdate	custid	val	orderrankno
11011	2008-04-09 00:00:00.000	1	933.50	1
10692	2007-10-03 00:00:00.000	1	878.00	2
10835	2008-01-15 00:00:00.000	1	845.80	3
...				
...				
10906	2008-02-25 00:00:00.000	91	427.50	5
10792	2007-12-23 00:00:00.000	91	399.85	6
10870	2008-02-04 00:00:00.000	91	160.00	7

(830 row(s) affected)

	orderid	orderdate	custid	val	orderrankno
1	11011	2008-04-09 00:00:00.000	1	933.50	1
2	10692	2007-10-03 00:00:00.000	1	878.00	2
3	10835	2008-01-15 00:00:00.000	1	845.80	3
4	10643	2007-08-25 00:00:00.000	1	814.50	4
5	10952	2008-03-16 00:00:00.000	1	471.20	5
6	10702	2007-10-13 00:00:00.000	1	330.00	6
7	10926	2008-03-04 00:00:00.000	2	514.40	1
8	10625	2007-08-08 00:00:00.000	2	479.75	2
9	10759	2007-11-28 00:00:00.000	2	320.00	3
10	10308	2006-09-18 00:00:00.000	2	88.80	4
11	10573	2007-06-19 00:00:00.000	3	2082.00	1

Query executed successfully. | MENTARI-PC\MENTARI (11.0 SP2) | MENTARI-PC\TOSHIBA (52) | TSQL2012 | 00:00:00 | 830 rows

```
SELECT orderid, orderdate, custid, val,  
       RANK() OVER (PARTITION BY custid ORDER BY val DESC) AS orderrankno  
FROM Sales.OrderValues;
```

100 %

	orderid	orderdate	custid	val	orderrankno
1	11011	2008-04-09 00:00:00.000	1	933.50	1
2	10692	2007-10-03 00:00:00.000	1	878.00	2
3	10835	2008-01-15 00:00:00.000	1	845.80	3
4	10643	2007-08-25 00:00:00.000	1	814.50	4
5	10952	2008-03-16 00:00:00.000	1	471.20	5
6	10702	2007-10-13 00:00:00.000	1	330.00	6
7	10926	2008-03-04 00:00:00.000	2	514.40	1
8	10625	2007-08-08 00:00:00.000	2	479.75	2
9	10759	2007-11-28 00:00:00.000	2	320.00	3
10	10308	2006-09-18 00:00:00.000	2	88.80	4
11	10573	2007-06-19 00:00:00.000	3	2082.00	1

Nama : Maulana Arya Putra Nugraha

Kelas : TI 2D

NIM : 2241720199

6

[Soal-5] Tuliskan pernyataan SELECT untuk mengambil kolom custid dan val dari view Sales.OrderValues. Tambahkan dua kolom berikut:

- 1) orderyear sebagai tahun dari kolom orderdate
- 2) orderrankno sebagai nomor urut, dipartisi berdasarkan pelanggan dan tahun pesanan, dan diurutkan berdasarkan nilai pesanan dalam urutan menurun!

55 - Lab Exercise1 - Task 4 Result.txt

custid	val	orderyear	orderrankno
1	878.00	2007	1
1	814.50	2007	2
1	330.00	2007	3
...			
...			
...			
91	591.60	2008	2
91	427.50	2008	3
91	160.00	2008	4

(830 row(s) affected)

Results

Messages

	custid	val	orderyear	orderrankno
1	1	878.00	2007	1
2	1	814.50	2007	2
3	1	330.00	2007	3
4	1	933.50	2008	1
5	1	845.80	2008	2
6	1	471.20	2008	3
7	2	88.80	2006	1
8	2	479.75	2007	1
9	2	320.00	2007	2
10	2	514.40	2008	1
11	3	403.20	2006	1

Query executed successfully.

MENTARI-PC\MENTARI (11.0 SP2)

MENTARI-PC\TOSHIBA (52)

TSQL2012

00:00:00

830 rows

```
SELECT custid, val,  
       YEAR(orderdate) AS orderyear,  
       ROW_NUMBER() OVER (PARTITION BY custid, YEAR(orderdate) ORDER BY val DESC) AS orderrankno  
FROM Sales.OrderValues;
```

10 %

Results Messages

	custid	val	orderyear	orderrankno
1	1	878.00	2007	1
2	1	814.50	2007	2
3	1	330.00	2007	3
4	1	933.50	2008	1
5	1	845.80	2008	2
6	1	471.20	2008	3
7	2	88.80	2006	1
8	2	479.75	2007	1
9	2	320.00	2007	2
10	2	514.40	2008	1
11	3	403.20	2006	1

Nama : Maulana Arya Putra Nugraha

Kelas : TI 2D

NIM : 2241720199

**[Soal-6]** Salin query jawaban soal nomor 6 dan modifikasi untuk memfilter hanya pesanan dengan dua peringkat paling awal berdasarkan kolom orderrankno!

56 - Lab Exercise 1 - Task 5 Result.txt

custid	orderyear	orderrankno	val
1	2007	1	878.00
1	2007	2	814.50
1	2008	1	933.50
...			
...			
...			
91	2007	2	399.85
91	2008	1	686.00
91	2008	2	591.60

(418 row(s) affected)

Results

Messages

	custid	orderyear	orderrankno	val
1	1	2007	1	878.00
2	1	2007	2	814.50
3	1	2008	1	933.50
4	1	2008	2	845.80
5	2	2006	1	88.80
6	2	2007	1	479.75
7	2	2007	2	320.00
8	2	2008	1	514.40
9	3	2006	1	403.20
10	3	2007	1	2082.00
11	3	2007	2	1940.85

Query executed successfully.

MENTARI-PC\MENTARI (11.0 SP2)

MENTARI-PC\TOSHIBA (52)

TSQL2012

00:00:00

418 rows

```
SELECT custid, orderyear, orderrankno, val
FROM (
    SELECT custid, val,
           YEAR(orderyear) AS orderyear,
           ROW_NUMBER() OVER (PARTITION BY custid, YEAR(orderyear) ORDER BY val DESC) AS orderrankno
    FROM Sales.OrderValues
) AS RankedOrders
WHERE orderrankno <= 2;
```

100 % ▾

Results		Messages		
	custid	orderyear	orderrankno	val
1	1	2007	1	878.00
2	1	2007	2	814.50
3	1	2008	1	933.50
4	1	2008	2	845.80
5	2	2006	1	88.80
6	2	2007	1	479.75
7	2	2007	2	320.00
8	2	2008	1	514.40
9	3	2006	1	403.20
10	3	2007	1	2082.00
11	3	2007	2	1940.85

**8 Kesimpulan :** Setelah menjalankan praktikum bagian ini, mahasiswa mengetahui bagaimana menggunakan fungsi ranking pada pernyataan T-SQL.

## Praktikum – Bagian 2: Menulis Query Menggunakan Fungsi OFFSET

Langkah	Keterangan
---------	------------

Nama : Maulana Arya Putra Nugraha

Kelas : TI 2D

NIM : 2241720199

Skenario :

Laporan lainnya diperlukan untuk menganalisis perbedaan antara dua baris secara berturut- turut. Hal ini akan mempermudah *business user* untuk menganalisa pertumbuhan dan tren.

Untuk melakukan percobaan pada praktikum bagian 2 ini, pastikan database terhubung dengan “TSQL”.

[Soal-7] Buatlah (*common table expression*) CTE dengan nama OrderRows berdasarkan query yang mengambil kolom orderid, orderdate, and val dari view Sales.OrderValues. Tambahkan kolom hasil perhitungan dengan nama rowno menggunakan fungsi ROW\_NUMBER yang diurutkan berdasarkan kolom orderdate dan orderid!

```
WITH OrderRows AS (  
    SELECT orderid, orderdate, val,  
           ROW_NUMBER() OVER (ORDER BY orderdate, orderid) AS rowno  
    FROM Sales.OrderValues  
)  
SELECT orderid, orderdate, val, rowno  
FROM OrderRows;
```

%

Results Messages

orderid	orderdate	val	rowno
10248	2006-07-04 00:00:00.000	440.00	1
10249	2006-07-05 00:00:00.000	1863.40	2
10250	2006-07-08 00:00:00.000	1552.60	3
10251	2006-07-08 00:00:00.000	654.06	4
10252	2006-07-09 00:00:00.000	3597.90	5
10253	2006-07-10 00:00:00.000	1444.80	6
10254	2006-07-11 00:00:00.000	556.62	7
10255	2006-07-12 00:00:00.000	2490.50	8
10256	2006-07-15 00:00:00.000	517.80	9
10257	2006-07-16 00:00:00.000	1119.90	10
10258	2006-07-17 00:00:00.000	1614.88	11



Nama : Maulana Arya Putra Nugraha

Kelas : TI 2D

NIM : 2241720199

**[Soal-8]** Tuliskan pernyataan SELECT terhadap CTE dan gunakan LEFT JOIN dengan CTE yang sama untuk mengambil baris saat ini (*current row*) dan baris sebelumnya (*previous row*) berdasarkan kolom rowno. Kembalikan kolomorderid, orderdate, and val untuk baris saat ini dan kolom val untuk baris sebelumnya sebagai prevval. Tambahkan kolom hasil perhitungan dengan nama diffprev yang menunjukkan perbedaan antara val saat ini dengan sebelumnya!

3

62 - Lab Exercise 2 - Task 1 Result.txt

orderid	orderdate	val	prevval	diffprev
10248	2006-07-04 00:00:00.000	440.00	NULL	NULL
10249	2006-07-05 00:00:00.000	1863.40	440.00	1423.40
10250	2006-07-08 00:00:00.000	1552.60	1863.40	-310.80
...				
...				
11075	2008-05-06 00:00:00.000	498.10	232.09	266.01
11076	2008-05-06 00:00:00.000	792.75	498.10	294.65
11077	2008-05-06 00:00:00.000	1255.72	792.75	462.97

(830 row(s) affected)

Results Messages

	orderid	orderdate	val	prevval	diffprev
1	10248	2006-07-04 00:00:00.000	440.00	NULL	NULL
2	10249	2006-07-05 00:00:00.000	1863.40	440.00	1423.40
3	10250	2006-07-08 00:00:00.000	1552.60	1863.40	-310.80
4	10251	2006-07-08 00:00:00.000	654.06	1552.60	-898.54
5	10252	2006-07-09 00:00:00.000	3597.90	654.06	2943.84
6	10253	2006-07-10 00:00:00.000	1444.80	3597.90	-2153.10
7	10254	2006-07-11 00:00:00.000	556.62	1444.80	-888.18
8	10255	2006-07-12 00:00:00.000	2490.50	556.62	1933.88
9	10256	2006-07-15 00:00:00.000	517.80	2490.50	-1972.70
10	10257	2006-07-16 00:00:00.000	1119.90	517.80	602.10
11	10258	2006-07-17 00:00:00.000	1614.88	1119.90	494.98

Query executed successfully. MENTARI-PC\MENTARI (11.0 SP2) MENTARI-PC\TOSHIBA (52) TSQL2012 00:00:00 830 rows

```
WITH OrderRows AS (
    SELECT orderid, orderdate, val,
           ROW_NUMBER() OVER (ORDER BY orderdate, orderid) AS rowno
    FROM Sales.OrderValues
)
SELECT
    CR.orderid,
    CR.orderdate,
    CR.val,
    PR.val AS prevval,
    CR.val - PR.val AS diffprev
FROM OrderRows CR
LEFT JOIN OrderRows PR ON CR.rowno = PR.rowno + 1;
```

0 %

Results

Messages

	orderid	orderdate	val	prevval	diffprev
	10248	2006-07-04 00:00:00.000	440.00	NULL	NULL
!	10249	2006-07-05 00:00:00.000	1863.40	440.00	1423.40
!	10250	2006-07-08 00:00:00.000	1552.60	1863.40	-310.80
!	10251	2006-07-08 00:00:00.000	654.06	1552.60	-898.54
!	10252	2006-07-09 00:00:00.000	3597.90	654.06	2943.84
!	10253	2006-07-10 00:00:00.000	1444.80	3597.90	-2153.10
!	10254	2006-07-11 00:00:00.000	556.62	1444.80	-888.18
!	10255	2006-07-12 00:00:00.000	2490.50	556.62	1933.88
!	10256	2006-07-15 00:00:00.000	517.80	2490.50	-1972.70
0	10257	2006-07-16 00:00:00.000	1119.90	517.80	602.10
1	10258	2006-07-17 00:00:00.000	1614.88	1119.90	494.98



Nama : Maulana Arya Putra Nugraha

Kelas : TI 2D

NIM : 2241720199

4

[Soal-9] Tuliskan pernyataan SELECT menggunakan fungsi LAG untuk mendapat hasil yang sama dengan query pada soal no.2! Query tidak yang dibuat pada soal ini tidak menggunakan CTE.

63 - Lab Exercise 2 - Task 2 Result.txt\*

orderid	orderdate	val	prevval	diffprev
10248	2006-07-04 00:00:00.000	440.00	NULL	NULL
10249	2006-07-05 00:00:00.000	1863.40	440.00	1423.40
10250	2006-07-08 00:00:00.000	1552.60	1863.40	-310.80
...				
...				
11075	2008-05-06 00:00:00.000	498.10	232.09	266.01
11076	2008-05-06 00:00:00.000	792.75	498.10	294.65
11077	2008-05-06 00:00:00.000	1255.72	792.75	462.97

(830 row(s) affected)

Results Messages

	orderid	orderdate	val	prevval	diffprev
1	10248	2006-07-04 00:00:00.000	440.00	NULL	NULL
2	10249	2006-07-05 00:00:00.000	1863.40	440.00	1423.40
3	10250	2006-07-08 00:00:00.000	1552.60	1863.40	-310.80
4	10251	2006-07-08 00:00:00.000	654.06	1552.60	-898.54
5	10252	2006-07-09 00:00:00.000	3597.90	654.06	2943.84
6	10253	2006-07-10 00:00:00.000	1444.80	3597.90	-2153.10
7	10254	2006-07-11 00:00:00.000	556.62	1444.80	-888.18
8	10255	2006-07-12 00:00:00.000	2490.50	556.62	1933.88
9	10256	2006-07-15 00:00:00.000	517.80	2490.50	-1972.70
10	10257	2006-07-16 00:00:00.000	1119.90	517.80	602.10
11	10258	2006-07-17 00:00:00.000	1614.88	1119.90	494.98

Query executed successfully.

MENTARI-PC\MENTARI (11.0 SP2) | MENTARI-PC\TOSHIBA (52) | TSQL2012 | 00:00:00 | 830 rows

```
SELECT
    orderid,
    orderdate,
    val,
    LAG(val) OVER (ORDER BY orderdate, orderid) AS prevval,
    val - LAG(val) OVER (ORDER BY orderdate, orderid) AS diffprev
FROM Sales.OrderValues;
```

orderid	orderdate	val	prevval	diffprev
10248	2006-07-04 00:00:00.000	440.00	NULL	NULL
10249	2006-07-05 00:00:00.000	1863.40	440.00	1423.40
10250	2006-07-08 00:00:00.000	1552.60	1863.40	-310.80
10251	2006-07-08 00:00:00.000	654.06	1552.60	-898.54
10252	2006-07-09 00:00:00.000	3597.90	654.06	2943.84
10253	2006-07-10 00:00:00.000	1444.80	3597.90	-2153.10
10254	2006-07-11 00:00:00.000	556.62	1444.80	-888.18
10255	2006-07-12 00:00:00.000	2490.50	556.62	1933.88
10256	2006-07-15 00:00:00.000	517.80	2490.50	-1972.70
10257	2006-07-16 00:00:00.000	1119.90	517.80	602.10
10258	2006-07-17 00:00:00.000	1614.88	1119.90	494.98

5

[Soal-10] Buatlah sebuah CTE bernama SalesMonth2007 yang membuat dua kolom yaitu, monthno (jumlah bulan dari kolom orderdate) dan val (agregat dari kolom val)! Kemudian filter hasilnya hanya untuk tahun pesanan 2007 dan dikelompokkan berdasarkan monthno!

Nama : Maulana Arya Putra Nugraha

Kelas : TI 2D

NIM : 2241720199

```
WITH SalesMonth2007 AS (  
SELECT  
MONTH (orderdate) AS monthno,  
SUM(val) AS val  
FROM Sales.OrderValues  
WHERE YEAR (orderdate) = 2007  
GROUP BY MONTH(orderdate))  
SELECT * FROM SalesMonth2007;
```

100 %

Results Messages

	monthno	val
1	1	61258.08
2	2	38483.64
3	3	38547.23
4	4	53032.95
5	5	53781.30
6	6	36362.82
7	7	51020.86
8	8	47287.68
9	9	55629.27
10	10	66749.23
11	11	43533.80
12	12	71398.44

[Soal-11] Tuliskan pernyataan SELECT yang akan mengambil kolom monthno dan val dari CTE dan tambahkan 3 kolom untuk ditampilkan, yaitu :

- 1) avglast3months (jumlah penjualan rata-rata tiga bulan terakhir)
- 2) diffjanuary (perbedaan antara val saat ini dengan val pada bulan januari, gunakan fungsi FIRST\_VALUE)
- 3) nextval (nilai dari kolom val pada bulan selanjutnya)

Informasi : Jumlah rata-rata untuk tiga bulan terakhir tidak dihitung dengan benar karena jumlah total 2 bulan pertama dibagi dengan 3.

6

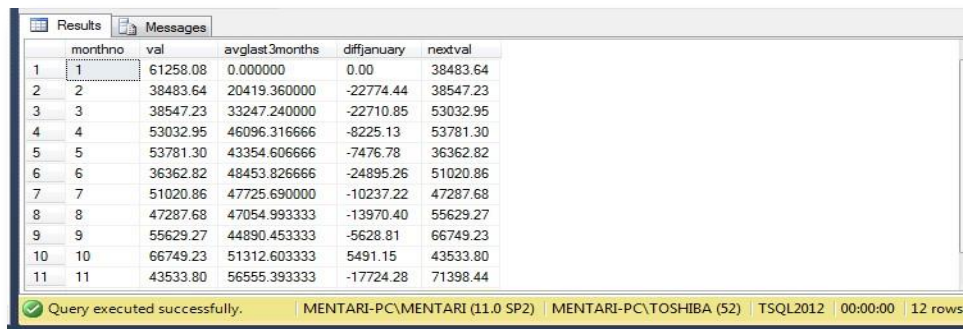
monthno	val	avglast3months	diffjanuary	nextval
1	61258.08	0.000000	0.00	38483.64
2	38483.64	20419.360000	-22774.44	38547.23
3	38547.23	33247.240000	-22710.85	53032.95
4	53032.95	46096.316666	-8225.13	53781.30
5	53781.30	43354.606666	-7476.78	36362.82
6	36362.82	48453.826666	-24895.26	51020.86
7	51020.86	47725.690000	-10237.22	47287.68
8	47287.68	47054.993333	-13970.40	55629.27
9	55629.27	44890.453333	-5628.81	66749.23
10	66749.23	51312.603333	5491.15	43533.80
11	43533.80	56555.393333	-17724.28	71398.44
12	71398.44	55304.100000	10140.36	NULL

(12 row(s) affected)

Nama : Maulana Arya Putra Nugraha

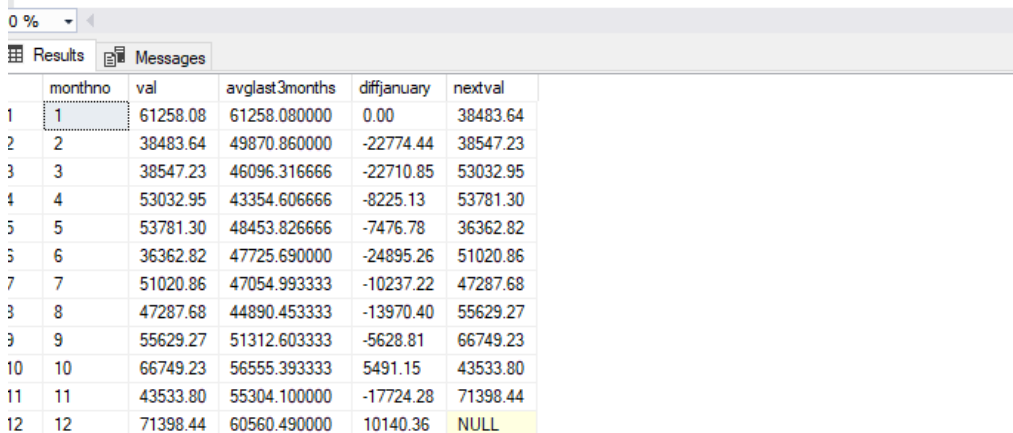
Kelas : TI 2D

NIM : 2241720199



	monthno	val	avglast3months	diffjanuary	nextval
1	1	61258.08	0.000000	0.00	38483.64
2	2	38483.64	20419.360000	-22774.44	38547.23
3	3	38547.23	33247.240000	-22710.85	53032.95
4	4	53032.95	46096.316666	-8225.13	53781.30
5	5	53781.30	43354.606666	-7476.78	36362.82
6	6	36362.82	48453.826666	-24895.26	51020.86
7	7	51020.86	47725.690000	-10237.22	47287.68
8	8	47287.68	47054.993333	-13970.40	55629.27
9	9	55629.27	44890.453333	-5628.81	66749.23
10	10	66749.23	51312.603333	5491.15	43533.80
11	11	43533.80	56555.393333	-17724.28	71398.44

```
WITH SalesMonth2007 AS (  
SELECT MONTH (orderdate) AS monthno, SUM(val) AS val  
FROM Sales.OrderValues  
WHERE YEAR (orderdate) = 2007  
GROUP BY MONTH(orderdate)  
)  
SELECT monthno, val,  
AVG(val) OVER (ORDER BY monthno ROWS BETWEEN 2 PRECEDING AND CURRENT ROW) AS avglast3months,  
val - FIRST_VALUE(val) OVER (ORDER BY monthno) AS diffjanuary,  
LEAD(val) OVER (ORDER BY monthno) AS nextval  
FROM SalesMonth2007;
```



	monthno	val	avglast3months	diffjanuary	nextval
1	1	61258.08	61258.080000	0.00	38483.64
2	2	38483.64	49870.860000	-22774.44	38547.23
3	3	38547.23	46096.316666	-22710.85	53032.95
4	4	53032.95	43354.606666	-8225.13	53781.30
5	5	53781.30	48453.826666	-7476.78	36362.82
6	6	36362.82	47725.690000	-24895.26	51020.86
7	7	51020.86	47054.993333	-10237.22	47287.68
8	8	47287.68	44890.453333	-13970.40	55629.27
9	9	55629.27	51312.603333	-5628.81	66749.23
10	10	66749.23	56555.393333	5491.15	43533.80
11	11	43533.80	55304.100000	-17724.28	71398.44
12	12	71398.44	60560.490000	10140.36	NULL

7

**Kesimpulan:** Setelah menjalankan praktikum bagian ini, mahasiswa bisa menggunakan fungsi OFFSET pada pernyataan T-SQL.

### Praktikum – Bagian 3: Menulis Query Menggunakan Fungsi Agregasi Window

Langkah	Keterangan
1	<p>Skenario :</p> <p>Untuk lebih memahami nilai penjualan kumulatif pelanggan melalui waktu dan untuk menyediakan analisis penjualan dengan analisis selama setahun ini diperlukan pernyataan SELECT yang berbeda menggunakan fungsi agregasi window.</p> <p>Untuk melakukan percobaan pada praktikum bagian 3 ini, pastikan database terhubung dengan "TSQL".</p>

Nama : Maulana Arya Putra Nugraha

Kelas : TI 2D

NIM : 2241720199

2

**[Soal-12]** Tuliskan pernyataan SELECT untuk mengambil kolom custid, orderid, orderdate, dan val dari view Sales.OrderValues. Tambahkan kolom bernama percoftotalcust yang berisi persentase nilai masing-masing jumlah pesanan penjualan dibandingkan dengan jumlah penjualan untuk pelanggan tersebut!

72 - Lab Exercise 3 - Task 1 Result.txt

custid	orderid	orderdate	val	percoftotalcust
1	11011	2008-04-09 00:00:00.000	933.50	21.8464778843903580622513
1	10692	2007-10-03 00:00:00.000	878.00	20.5476246197051252047741
1	10835	2008-01-15 00:00:00.000	845.80	19.7940556985724315469225
...				
...				
...				
91	10906	2008-02-25 00:00:00.000	427.50	12.1037953538413624201928
91	10792	2007-12-23 00:00:00.000	399.85	11.3209416894350146519627
91	10870	2008-02-04 00:00:00.000	160.00	4.5300754540692818414756

(830 row(s) affected)

Results		Messages			
	custid	orderid	orderdate	val	percoftotalcust
1	1	11011	2008-04-09 00:00:00.000	933.50	21.8464778843903580622513
2	1	10692	2007-10-03 00:00:00.000	878.00	20.5476246197051252047741
3	1	10835	2008-01-15 00:00:00.000	845.80	19.7940556985724315469225
4	1	10643	2007-08-25 00:00:00.000	814.50	19.0615492628130119354083
5	1	10952	2008-03-16 00:00:00.000	471.20	11.0273812309852562602387
6	1	10702	2007-10-13 00:00:00.000	330.00	7.7229113035338169904048
7	2	10926	2008-03-04 00:00:00.000	514.40	36.6655974910011048148544
8	2	10625	2007-08-08 00:00:00.000	479.75	34.1958017035532271285505
9	2	10759	2007-11-28 00:00:00.000	320.00	22.8090808653195053280587
10	2	10308	2006-09-18 00:00:00.000	88.80	6.3295199401261627285362
11	3	10573	2007-06-19 00:00:00.000	2082.00	29.6413144684352745879116

Query executed successfully.

MENTARI-PC\MENTARI (11.0 SP2)

MENTARI-PC\TOSHIBA (52)

TSQL2012

00:00:00

830 rows

```
SELECT o.custid, o.orderid, o.orderdate, o.val, (o.val / ct.total_val * 100) as percoftotalcust
FROM Sales.OrderValues o
JOIN (
    SELECT custid,
    SUM(val) as total_val
FROM Sales.OrderValues
GROUP BY custid
) as ct ON o.custid = ct.custid
ORDER BY o.custid, o.val DESC;
```

10 %

Results	Messages					
	custid	orderid	orderdate	val	percoftotalcust	
1	1	11011	2008-04-09 00:00:00.000	933.50	21.8464778843903580622513	
2	1	10692	2007-10-03 00:00:00.000	878.00	20.5476246197051252047742	
3	1	10835	2008-01-15 00:00:00.000	845.80	19.7940556985724315469225	
4	1	10643	2007-08-25 00:00:00.000	814.50	19.0615492628130119354084	
5	1	10952	2008-03-16 00:00:00.000	471.20	11.0273812309852562602387	
6	1	10702	2007-10-13 00:00:00.000	330.00	7.7229113035338169904049	
7	2	10926	2008-03-04 00:00:00.000	514.40	36.6655974910011048148544	
8	2	10625	2007-08-08 00:00:00.000	479.75	34.1958017035532271285506	
9	2	10759	2007-11-28 00:00:00.000	320.00	22.8090808653195053280587	
10	2	10308	2006-09-18 00:00:00.000	88.80	6.3295199401261627285363	
11	3	10573	2007-06-19 00:00:00.000	2082.00	29.6413144684352745879117	
12	3	10535	2007-05-13 00:00:00.000	1940.85	27.6317700221242087819157	
13	3	10677	2007-09-22 00:00:00.000	813.37	11.5799019928872234829826	
14	3	10507	2007-04-15 00:00:00.000	749.06	10.6643242150461704048132	
15	3	10856	2008-01-28 00:00:00.000	660.00	9.3963821081495106762832	

Nama : Maulana Arya Putra Nugraha

Kelas : TI 2D

NIM : 2241720199

**[Soal-13]** Salin pernyataan SELECT sebelumnya dan modifikasi dengan menambahkan kolom yang baru dihitung bernama runval! Kolom ini harus berisi total penjualan yang sedang terjadi untuk setiap pelanggan berdasarkan tanggal pemesanan, menggunakan orderid sebagai tiebreaker.

73 - Lab Exercise 3 - Task 2 Result.txt

custid	orderid	orderdate	val	percoftotalcust	runval
1	10643	2007-08-25 00:00:00.000	814.50	19.0615492628130119354083	814.50
1	10692	2007-10-03 00:00:00.000	878.00	20.5476246197051252047741	1692.50
1	10702	2007-10-13 00:00:00.000	330.00	7.7229113035338169904048	2022.50
...					
...					
91	10906	2008-02-25 00:00:00.000	427.50	12.1037953538413624201928	2254.35
91	10998	2008-04-03 00:00:00.000	686.00	19.4226985093220458953269	2940.35
91	11044	2008-04-23 00:00:00.000	591.60	16.7499539914211696088562	3531.95

(830 row(s) affected)

3

Results Messages

	custid	orderid	orderdate	val	percoftotalcust	runval
1	1	10643	2007-08-25 00:00:00.000	814.50	19.0615492628130119354083	814.50
2	1	10692	2007-10-03 00:00:00.000	878.00	20.5476246197051252047741	1692.50
3	1	10702	2007-10-13 00:00:00.000	330.00	7.7229113035338169904048	2022.50
4	1	10835	2008-01-15 00:00:00.000	845.80	19.7940556985724315469225	2868.30
5	1	10952	2008-03-16 00:00:00.000	471.20	11.0273812309852562602387	3339.50
6	1	11011	2008-04-09 00:00:00.000	933.50	21.8464778843903580622513	4273.00
7	2	10308	2006-09-18 00:00:00.000	88.80	6.3295199401261627285362	88.80
8	2	10625	2007-08-08 00:00:00.000	479.75	34.1958017035532271285505	568.55
9	2	10759	2007-11-28 00:00:00.000	320.00	22.8090808653195053280587	888.55
10	2	10926	2008-03-04 00:00:00.000	514.40	36.6655974910011048148544	1402.95
11	3	10365	2006-11-27 00:00:00.000	403.20	5.7403352515240647040566	403.20

Query executed successfully. MENTARI-PC\MENTARI (11.0 SP2) | MENTARI-PC\TOSHIBA (52) | TSQL2012 | 00:00:00 | 830 rows

```
SELECT o.custid, o.orderid, o.orderdate, o.val, (o.val / ct.total_val * 100) as percoftotalcust,
SUM(o.val) OVER (PARTITION BY o.custid ORDER BY o.orderdate, o.orderid) as runval
FROM Sales.OrderValues o
JOIN (
SELECT custid,
SUM(val) as total_val
FROM Sales.OrderValues
GROUP BY custid
) as ct ON o.custid = ct.custid
ORDER BY o.custid, o.orderdate, o.orderid;
```

Results Messages

	custid	orderid	orderdate	val	percoftotalcust	runval
1	1	10643	2007-08-25 00:00:00.000	814.50	19.0615492628130119354084	814.50
2	1	10692	2007-10-03 00:00:00.000	878.00	20.5476246197051252047742	1692.50
3	1	10702	2007-10-13 00:00:00.000	330.00	7.7229113035338169904049	2022.50
4	1	10835	2008-01-15 00:00:00.000	845.80	19.7940556985724315469225	2868.30
5	1	10952	2008-03-16 00:00:00.000	471.20	11.0273812309852562602387	3339.50
6	1	11011	2008-04-09 00:00:00.000	933.50	21.8464778843903580622513	4273.00
7	2	10308	2006-09-18 00:00:00.000	88.80	6.3295199401261627285363	88.80
8	2	10625	2007-08-08 00:00:00.000	479.75	34.1958017035532271285506	568.55
9	2	10759	2007-11-28 00:00:00.000	320.00	22.8090808653195053280587	888.55
10	2	10926	2008-03-04 00:00:00.000	514.40	36.6655974910011048148544	1402.95
11	3	10365	2006-11-27 00:00:00.000	403.20	5.7403352515240647040567	403.20
12	3	10507	2007-04-15 00:00:00.000	749.06	10.6643242150461704048132	1152.26
13	3	10535	2007-05-13 00:00:00.000	1940.85	27.6317700221242087819157	3093.11
14	3	10573	2007-06-19 00:00:00.000	2082.00	29.6413144684352745879117	5175.11
15	3	10677	2007-09-22 00:00:00.000	813.37	11.5799019928872234879826	5988.48



Nama : Maulana Arya Putra Nugraha

Kelas : TI 2D

NIM : 2241720199

4

**[Soal-14]** Salin CTE SalesMonth2007 dalam percobaan 2. Tuliskan pernyataan SELECT untuk mengambil kolom monthno dan val. Tambahkan dua kolom yang dihitung:

- 1) avglast3months. Kolom ini harus berisi jumlah penjualan rata-rata untuk tiga bulan terakhir sebelum bulan saat ini menggunakan fungsi window agregat. Asumsikan bahwa tidak ada *missing months*.
- 2) ytdval Kolom ini harus berisi nilai penjualan kumulatif sampai dengan bulan saat ini.

74 - Lab Exercise 3 - Task 3 Result.txt

monthno	val	avglast3months	ytdval
1	61258.08	61258.080000	61258.08
2	38483.64	49870.860000	99741.72
3	38547.23	46096.316666	138288.95
4	53032.95	47830.475000	191321.90
5	53781.30	45961.280000	245103.20
6	36362.82	45431.075000	281466.02
7	51020.86	48549.482500	332486.88
8	47287.68	47113.165000	379774.56
9	55629.27	47575.157500	435403.83
10	66749.23	55171.760000	502153.06
11	43533.80	53299.995000	545686.86
12	71398.44	59327.685000	617085.30

(12 row(s) affected)

Results		Messages		
	monthno	val	avglast3months	ytdval
1	1	61258.08	61258.080000	61258.08
2	2	38483.64	49870.860000	99741.72
3	3	38547.23	46096.316666	138288.95
4	4	53032.95	47830.475000	191321.90
5	5	53781.30	45961.280000	245103.20
6	6	36362.82	45431.075000	281466.02
7	7	51020.86	48549.482500	332486.88
8	8	47287.68	47113.165000	379774.56
9	9	55629.27	47575.157500	435403.83
10	10	66749.23	55171.760000	502153.06
11	11	43533.80	53299.995000	545686.86

Query executed successfully.

MENTARI-PC\MENTARI (11.0 SP2)

MENTARI-PC\TOSHIBA (52)

TSQL2012

00:00:00

12 rows

Nama : Maulana Arya Putra Nugraha

Kelas : TI 2D

NIM : 2241720199

```
WITH SalesMonth2007 AS (  
    SELECT MONTH (orderdate) AS monthno,  
           SUM(val) AS val  
    FROM Sales.OrderValues  
    WHERE YEAR(orderdate) = 2007  
    GROUP BY MONTH(orderdate)  
)  
  
SELECT sm.monthno, sm.val,  
       AVG(sm.val) OVER (ORDER BY sm.monthno ROWS BETWEEN 3 PRECEDING AND 1 PRECEDING) AS avglast3months,  
       SUM(sm.val) OVER (ORDER BY sm.monthno) AS ytdval  
FROM SalesMonth2007 sm;
```

100 %

Results Messages

	monthno	val	avglast3months	ytdval
1	1	61258.08	NULL	61258.08
2	2	38483.64	61258.080000	99741.72
3	3	38547.23	49870.860000	138288.95
4	4	53032.95	46096.316666	191321.90
5	5	53781.30	43354.606666	245103.20
6	6	36362.82	48453.826666	281466.02
7	7	51020.86	47725.690000	332486.88
8	8	47287.68	47054.993333	379774.56
9	9	55629.27	44890.453333	435403.83
10	10	66749.23	51312.603333	502153.06
11	11	43533.80	56555.393333	545686.86
12	12	71398.44	55304.100000	617085.30

5

**Kesimpulan:** Setelah melakukan praktikum bagian ini mendapatkan dasar pemahaman bagaimana menggunakan fungsi agregasi window pada pernyataan T-SQL.

--- Selamat Mengerjakan ----