

JOBSHEET

PRAKTIKUM BASIS DATA LANJUT

Jurusan Teknologi Informasi
POLITEKNIK NEGERI MALANG



PERTEMUAN 6

SQL SERVER – TABLE EXPRESSION

Topik

1. Table Expressions

Tujuan

1. Mahasiswa memahami cara menggunakan VIEWS
2. Mahasiswa memahami cara menggunakan derived table/ tabel turunan
3. Mahasiswa memahami cara menggunakan common table-expression (CTE)
4. Mahasiswa memahami cara menggunakan inline table-valued function (TVF)

Petunjuk Umum

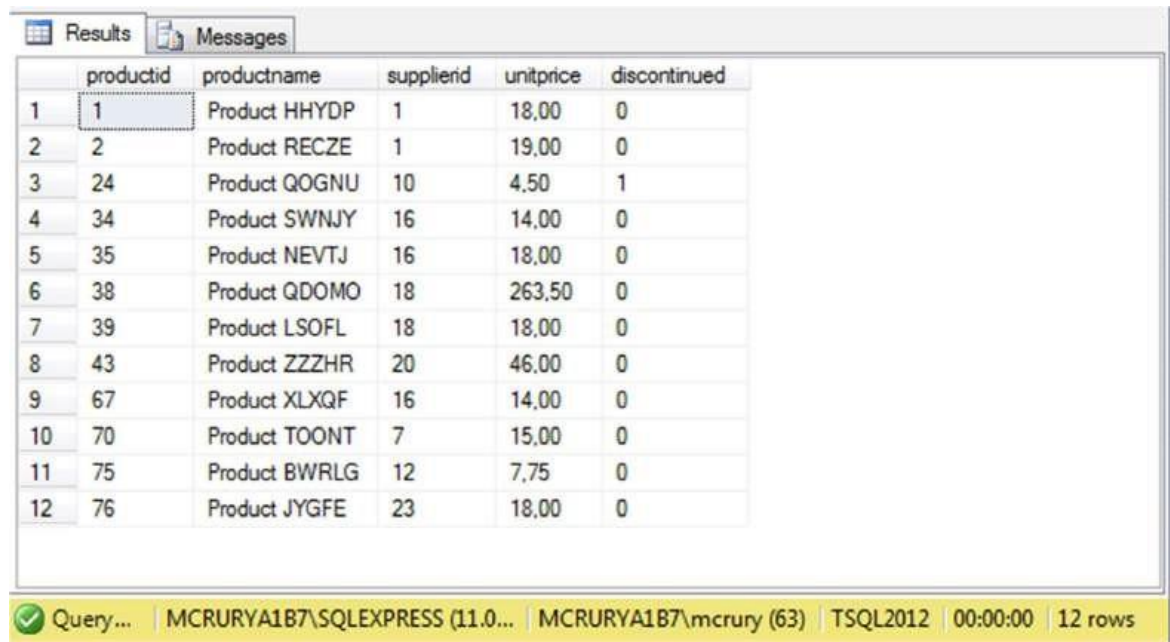
1. Ikuti langkah-langkah pada bagian-bagian praktikum sesuai dengan urutan yang diberikan.
2. Anda dapat menggunakan SQL Server 2012 Standard Edition untuk mencoba praktikum pada jobsheet ini. Sesuaikan dengan kondisi komputer Anda.
3. Jawablah semua pertanyaan bertanda [Soal-X] yang terdapat pada langkah-langkah tertentu di setiap bagian praktikum.
4. Dalam setiap langkah pada praktikum terdapat penjelasan yang akan membantu Anda dalam menjawab pertanyaan-pertanyaan pada petunjuk nomor 3, maka baca dan kerjakanlah semua bagian praktikum dalam jobsheet ini.
5. Tulis jawaban dari soal-soal pada petunjuk nomor 3 pada sebuah laporan yang dikerjakan menggunakan aplikasi word processing (Word, OpenOffice, atau yang lain yang sejenis). Ekspor sebagai file **PDF** dengan format nama sebagai berikut:
 - **SMBD_Tugas7_Kelas_2DigitNomorAbsen_NamaLengkapAnda.pdf**
 - Contoh:
 - o **SMBD_Tugas7_TI2Q_99_DonaldTrump.pdf**
 - Perhatikan baik-baik format penamaanya.
 - Kumpulkan file PDF tersebut sebagai laporan praktikum kepada dosen pengampu.
 - Selain pada nama file, cantumkan juga identitas Anda pada halaman pertama laporan tersebut.

Praktikum – Bagian 1: View - Menulis sebuah query SELECT untuk mendapatkan semua produk dalam kategori tertentu

Langkah	Keterangan
1	Pastikan database Anda terkoneksi ke 'TSQL2012'
2	<p>[Soal-1] Tulislah sebuah query SELECT untuk menampilkan kolom <i>productid</i>, <i>productname</i>, <i>supplierid</i>, <i>unitprice</i> dan kolom <i>discontinued</i> dari tabel Productions.Product. Kemudian filter hasilnya agar hanya menampilkan produk yang ada di dalam kategori Beverages saja (<i>categoryid</i> = 1)</p> <pre>SELECT [productid],[productname],[supplierid],[unitprice],[discontinued] FROM [Production].[Products] WHERE categoryid=1;</pre>

3

Eksekusi query pada langkah 2 di atas dan bandingkan dengan hasil yang ditunjukkan pada tampilan berikut ini:



	productid	productname	supplierid	unitprice	discontinued
1	1	Product HHYDP	1	18,00	0
2	2	Product RECZE	1	19,00	0
3	24	Product QOGNU	10	4,50	1
4	34	Product SWNJY	16	14,00	0
5	35	Product NEVTJ	16	18,00	0
6	38	Product QDOMO	18	263,50	0
7	39	Product LSOFL	18	18,00	0
8	43	Product ZZZHR	20	46,00	0
9	67	Product XLXQF	16	14,00	0
10	70	Product TOONT	7	15,00	0
11	75	Product BWRLG	12	7,75	0
12	76	Product JYGFE	23	18,00	0

Query... MCRURYA1B7\SQLEXPRESS (11.0... MCRURYA1B7\mcrury (63) TSQL2012 00:00:00 12 rows

4

[Soal-2] Modifikasilah kode T-SQL dari no 2 di atas dengan menambahkan kode T-SQL berikut (letakkan sebelum query SELECT)

CREATE VIEW Production.ProductsBeverages AS

```
CREATE VIEW Production.ProductsBeverages AS
SELECT
    productid, productname, supplierid, unitprice, discontinued
FROM Production.Products
WHERE categoryid = 1;
```

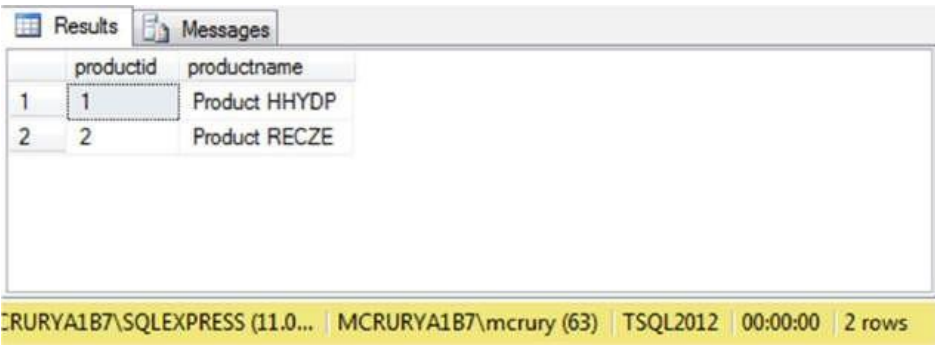
5

Eksekusilah query pada langkah 4 di atas, sehingga akan menghasilkan sebuah objek VIEW bernama **ProductsBeverages** di dalam skema **Production**



Praktikum – Bagian 2: View - Menulis query SELECT terhadap VIEW yang sudah dibuat

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

1	<p>[Soal-3] Buatlah sebuah query SELECT yang terdiri dari kolom <i>productid</i> dan <i>productname</i> dari VIEW Production.ProductsBeverages. Lalu filterlah hasilnya agar hanya menampilkan produk dengan <i>supplierid</i> = 1.</p>
2	<p>Eksekusi query pada langkah 1 di atas dan bandingkan dengan hasil yang ditunjukkan pada tampilan berikut ini:</p> 

Praktikum – Bagian 3: View - Menambahkan klausa ORDER BY pada VIEW

Langkah	Keterangan
1	<p>Perhatikan skrip T-SQL berikut:</p> <pre>ALTER VIEW Production.ProductsBeverages AS SELECT productid, productname, supplierid, unitprice, discontinued FROM Production.Products WHERE categoryid = 1 ORDER BY productname;</pre>
2	<p>[Soal-4] Setelah mengeksekusi T-SQL di atas, apakah yang terjadi? Tuliskan pesan error yang ada dan jelaskan penyebab mengapa error tersebut muncul!</p> <pre>ALTER VIEW Production.ProductsBeverages AS SELECT productid, productname, supplierid, unitprice, discontinued FROM Production.Products WHERE categoryid = 1 ORDER BY productname;</pre> <p>Karena ada orderby tidak bisa digunakan pada view, kecuali menggunakan offset atau top</p>

3	<p>Modifikasilah T-SQL pada langkah 1 di atas dengan menambahkan TOP(100) PERCENT sehingga sekarang query menjadi:</p> <pre>ALTER VIEW Production.ProductsBeverages AS SELECT TOP(100) PERCENT productid, productname, supplierid, unitprice, discontinued FROM Production.Products WHERE categoryid = 1 ORDER BY productname;</pre>
4	<p>Eksekusi T-SQL pada langkah 3 di atas dan perhatikan bahwa query tersebut telah berhasil mengubah VIEW Production.ProductsBeverages meskipun masih terdapat klausa ORDER BY dalam query tersebut.</p>
5	<p>[Soal-5] Jika sebuah query dijalankan terhadap VIEW Production.ProductsBeverages yang telah dimodifikasi, apakah baris yang dihasilkan dari VIEW tersebut akan selaluurut berdasarkan <i>productname</i>? Jelaskan!</p> <pre>ALTER VIEW Production.ProductsBeverages AS SELECT TOP(100) PERCENT productid, productname, supplierid, unitprice, discontinued FROM Production.Products WHERE categoryid = 1 ORDER BY productname;</pre> <pre>SELECT * FROM Production.ProductsBeverages ORDER BY productname asc;</pre> <p>Tidak, meskipun sudah diatur pada syntax tersebut, tetap tidak akan melakukan perubahan pada hasil, kecuali jika pada saat menampilkan views baru diurutkan maka akan bisa berubah</p>

Praktikum – Bagian 4: View - Menambahkan kolom ke dalam VIEW

Langkah	Keterangan
1	<p>Perhatikan statement T-SQL berikut ini yang menambahkan sebuah kolom tambahan ke dalam VIEW Production.ProductsBeverages yang sudah dibuat di <u>Praktikum - Bagian 1</u> dengan perintah ALTER VIEW</p> <pre>ALTER VIEW Production.ProductsBeverages AS SELECT productid, productname, supplierid, unitprice, discontinued, CASE WHEN unitprice > 100. THEN N'high' ELSE N'normal' END FROM Production.Products WHERE categoryid = 1;</pre>

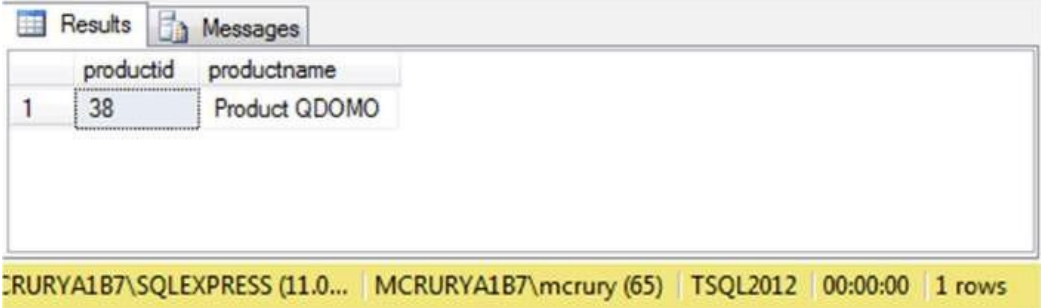
2	[Soal-6] Setelah mengeksekusi T-SQL di atas, apakah yang terjadi? Tuliskan pesan error yang ada dan jelaskan penyebab mengapa error tersebut muncul!
3	[Soal-7] Perbaiki skrip T-SQL di atas sehingga dapat dijalankan dengan benar.

Praktikum – Bagian 5: View - Menghapus VIEW

Langkah	Keterangan
1	<p>Untuk menghapus VIEW Production.ProductsBeverages, eksekusi perintah T-SQL berikut:</p> <pre>IF OBJECT_ID(N'Production.ProductsBeverages', N'V') IS NOT NULL DROP VIEW Production.ProductsBeverages;</pre>

Praktikum – Bagian 6: Derived Table - Membuat query SELECT dalam derived table

Langkah	Keterangan
1	<p>[Soal-8] Dengan menggunakan database TSQL2012, buatlah sebuah query SELECT terhadap <i>derived table</i> (tabel turunan) yang berisi kolom <i>productid</i> dan <i>productname</i>, dengan filter hanya menampilkan data yang '<i>pricetype</i>'-nya adalah 'high'.</p> <p>Gunakan query SELECT yang ada pada <u>Praktikum - Bagian 4 – Langkah 1</u> sebagai <i>derived table</i>-nya. Beri nama alias <i>p</i> terhadap <i>derived table</i> tersebut.</p> <pre>SELECT p.productid, p.productname FROM (SELECT productid, productname, supplierid, unitprice, discontinued, CASE WHEN unitprice > 100 THEN N'high' ELSE N'normal' END AS pricetype FROM Production.Products WHERE categoryid = 1) AS p WHERE p.pricetype = N'high';</pre>

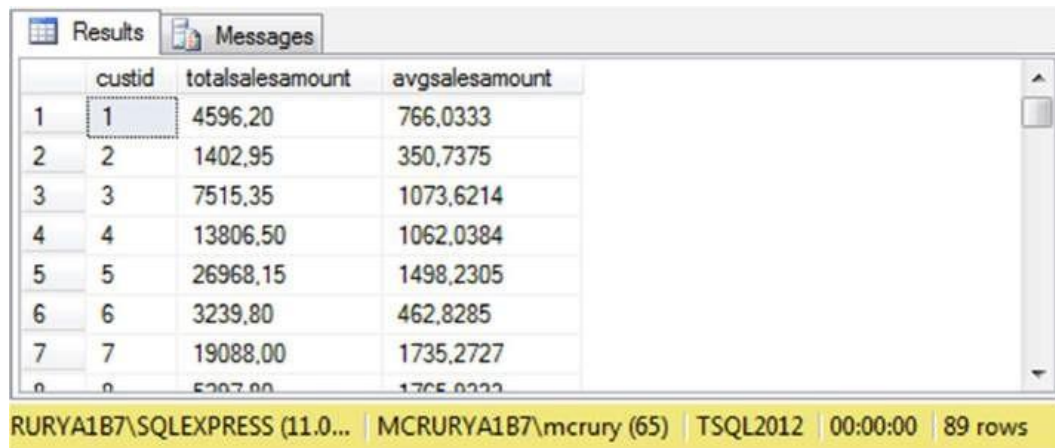
2	<p>Eksekusi query pada langkah 1 di atas dan bandingkan dengan hasil yang ditunjukkan pada tampilan berikut ini:</p> 
---	---

Praktikum – Bagian 7: Derived Table - Membuat query SELECT untuk mengetahui total dan rata-rata jumlah order (nominal)

Langkah	Keterangan
1	<p>[Soal-9] Buatlah sebuah query SELECT untuk mendapatkan kolom <i>custid</i> dan 2 (dua) kolom kalkulasi, yakni <i>totalsalesamount</i> (total jumlah nominal order per customer) dan <i>avgsalesamount</i> (rata-rata jumlah nominal order per customer).</p> <p>Untuk mengetahui rata-rata nominal order per customer, harus dicari dahulu jumlah total nominal per order. Caranya adalah dengan membuat sebuah <i>derived table</i> yang berisi query JOIN antara tabel Sales.Orders dan Sales.OrderDetails. Setelah itu, Anda bisa gunakan kolom <i>custid</i> dan <i>orderid</i> dari tabel Sales.Orders, serta kolom <i>qty</i> dan <i>unitprice</i> dari tabel Sales.OrderDetails.</p> <pre> SELECT c.custid, SUM(c.totalsalesamountperorder) AS totalsalesamount, AVG(c.totalsalesamountperorder) AS avgsalesamount FROM (SELECT o.custid, o.orderid, SUM(d.unitprice * d.qty) AS totalsalesamountperorder FROM Sales.Orders AS o INNER JOIN Sales.OrderDetails d ON d.orderid = o.orderid GROUP BY o.custid, o.orderid) AS c GROUP BY c.custid; </pre>

2

Eksekusi query pada langkah 1 di atas dan bandingkan dengan hasil yang ditunjukkan pada tampilan berikut ini:



	custid	totalsalesamount	avgsalesamount
1	1	4596,20	766,0333
2	2	1402,95	350,7375
3	3	7515,35	1073,6214
4	4	13806,50	1062,0384
5	5	26968,15	1498,2305
6	6	3239,80	462,8285
7	7	19088,00	1735,2727
8	8	5207,00	1765,0333

RURYA1B7\SQLEXPRESS (11.0... | MCRURYA1B7\mcrury (65) | TSQL2012 | 00:00:00 | 89 rows

Praktikum – Bagian 8: Derived Table - Membuat query SELECT untuk mendapatkan prosentase perkembangan penjualan

Langkah	Keterangan
1	<p>[Soal-10] Tulislah sebuah query SELECT yang berisi kolom-kolom berikut ini:</p> <ul style="list-style-type: none"> - <i>orderyear</i>: tahun dari tanggal order - <i>curtotalsales</i>: total jumlah penjualan pada tahun tersebut - <i>prevtotalsales</i>: total jumlah penjualan pada tahun sebelumnya - <i>percentgrowth</i>: prosentase perkembangan penjualan dari tahun yang sedang berjalan dibanding tahun sebelumnya
2	<p>Anda harus membuat query T-SQL dengan menggunakan 2 (dua) <i>derived table</i>. Untuk mendapatkan tahun dan total penjualan untuk setiap query SELECT, Anda bisa gunakan VIEW yang sudah ada bernama Sales.OrderValues. Dalam view tersebut, kolom <i>val</i> merepresentasikan jumlah penjualan.</p>
3	<p>Perlu diperhatikan bahwa dalam database TSQL2012, tahun 2006 adalah tahun order paling awal (tidak ada lagi tahun sebelumnya), akan tetapi query masih bisa dijalankan.</p>

4

Eksekusi query pada langkah 1 di atas dan bandingkan dengan hasil yang ditunjukkan pada tampilan berikut ini:

Results

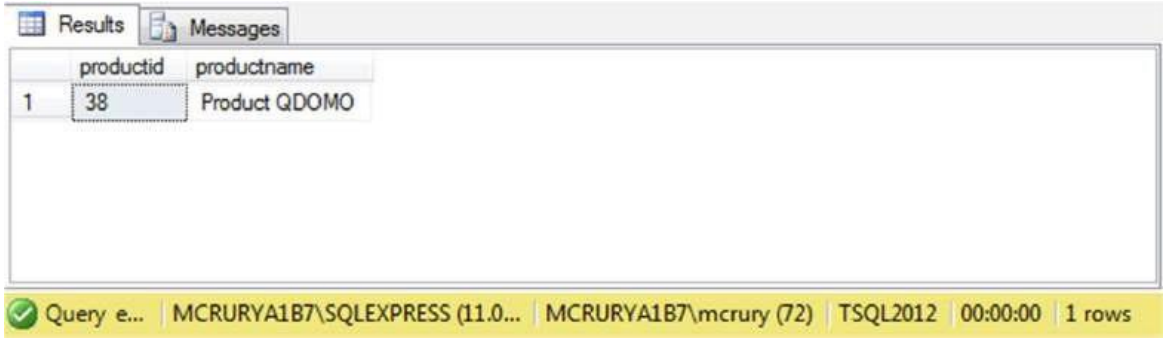
Messages

	orderyear	curtotalsales	prevtotalsales	percentgrowth
1	2006	208083.99	NULL	NULL
2	2007	617085.30	208083.99	196.555800
3	2008	440623.93	617085.30	-28.595900

CRURYA1B7\SQLEXPRESS (11.0... | MCRURYA1B7\mcrury (65) | TSQL2012 | 00:00:00 | 3 rows

Praktikum – Bagian 9: CTE - Membuat query SELECT yang menggunakan CTE

Langkah	Keterangan
1	<p>[Soal-11] Dengan tetap menggunakan database TSQL2012, buatlah query SELECT seperti di <u>Praktikum - Bagian 6</u>, akan tetapi dengan menggunakan Common Table Expressions (CTE). Beri nama alias query CTE tersebut sebagai ProductBeverages.</p> <pre> WITH ProductsBeverages AS (SELECT productid, productname, supplierid, unitprice, discontinued, CASE WHEN unitprice > 100 THEN N'high' ELSE N'normal' END AS pricetype FROM Production.Products WHERE categoryid = 1) SELECT productid, productname FROM ProductsBeverages WHERE pricetype = N'high'; </pre>

2	<p>Eksekusi query pada langkah 1 dan bandingan dengan hasil yang ditunjukkan pada tampilan berikut ini:</p> 
---	--

Praktikum – Bagian 10: CTE - Membuat query SELECT untuk mendapatkan total jumlah penjualan (nominalnya) untuk setiap customer

Langkah	Keterangan
1	<p>[Soal-12] Buatlah sebuah query SELECT terhadap view Sales.OrderValues untuk mendapatkan ID customer dan total jumlah penjualan pada tahun 2008. Beri nama CTE ini sebagai c2008, yang terdiri dari kolom <i>custid</i> dan <i>salesamt2008</i>.</p> <p>Kemudian, lakukan operasi JOIN antara table Sales.Customers dan CTE c2008, sehingga menghasilkan kolom <i>custid</i> dan <i>contactname</i> dari table Sales.Customer dan kolom <i>salesamt2008</i> dari CTE c2008.</p>
2	<p>Eksekusi query pada langkah 1 di atas dan bandingkan dengan hasil yang ditunjukkan pada tampilan berikut ini:</p>

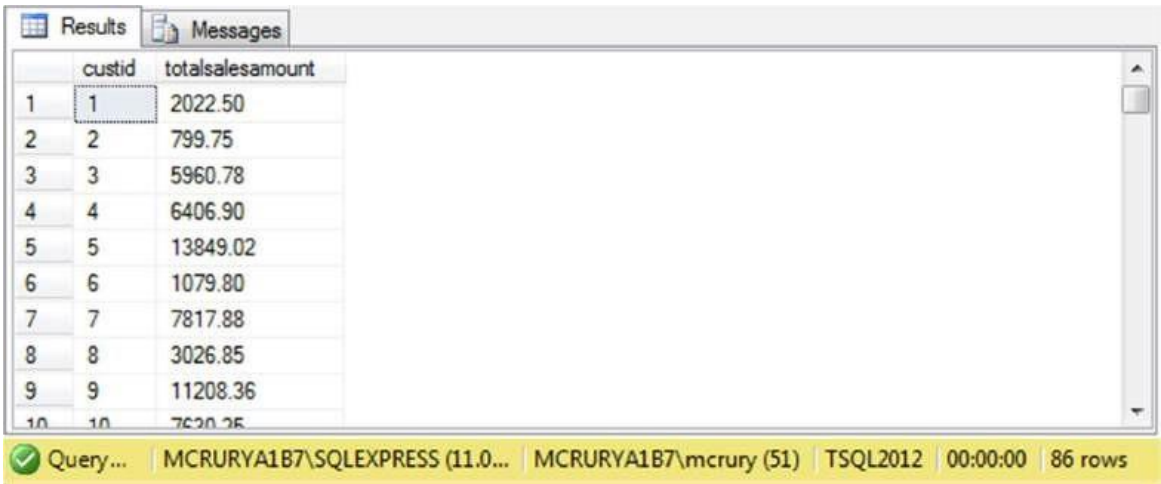
Results				Messages
	custid	contactname	salesamt2008	
1	1	Allen, Michael	2250.50	
2	2	Hassall, Mark	514.40	
3	3	Peoples, John	660.00	
4	4	Amdt, Torsten	5604.75	
5	5	Higginbotham, Tom	6754.16	
6	6	Poland, Carole	2160.00	
7	7	Bansal, Dushyant	730.00	
8	8	Ilyina, Julia	224.00	

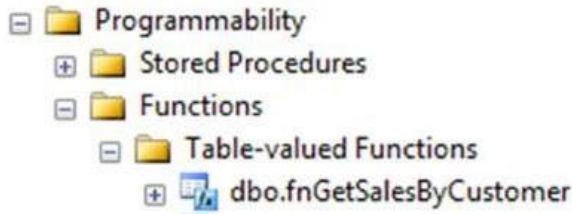
Query... MCRURYA1B7\SQLEXPRESS (11.0... MCRURYA1B7\mcrury (72) TSQL2012 00:00:00 91 rows

Praktikum – Bagian 11: CTE - Membuat query SELECT untuk membandingkan jumlah total penjualan untuk setiap customer dengan tahun sebelumnya

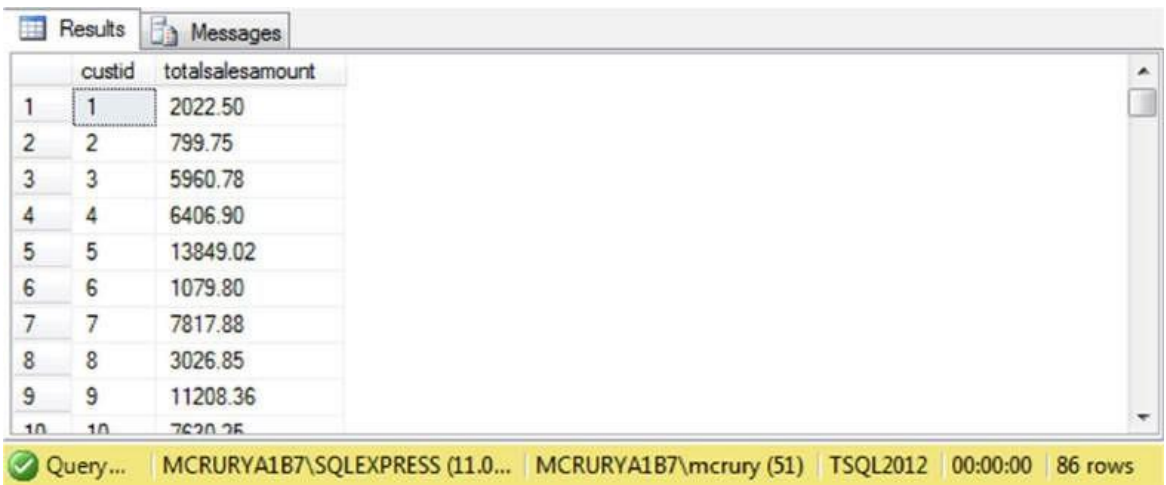
Langkah	Keterangan																																																						
1	<p>[Soal-13] Buatlah sebuah query SELECT yang berisi kolom <i>custid</i> dan <i>contactname</i> terhadap table Sales.Customers. Selain itu, dapatkan juga nilai untuk beberapa kolom berikut:</p> <ul style="list-style-type: none">- <i>salesamt2008</i>: total jumlah penjualan tahun 2008- <i>salesamt2007</i>: total jumlah penjualan tahun 2007- <i>percentgrowth</i>: prosentase perkembangan penjualan antara tahun 2007 dan 2008 <p>Apabila <i>percentgrowth</i> menghasilkan NULL, tampilkan sebagai 0.</p> <p>Anda bisa gunakan CTE dari <u>Praktikum Bagian 10</u> dan buatlah satu CTE lagi untuk tahun 2007. Kemudian, lakukan operasi JOIN antara kedua CTE tersebut dengan tabel Sales.Customers. Urutkan hasilnya berdasarkan kolom <i>percentgrowth</i>.</p>																																																						
2	<p>Eksekusi query pada langkah 1 di atas dan bandingkan dengan hasil yang ditunjukkan pada tampilan berikut ini:</p> <div><div>ResultsMessages</div><table><tr><th></th><th>custid</th><th>contactname</th><th>salesamt2008</th><th>salesamt2007</th><th>percentgrowth</th></tr><tr><td>1</td><td>74</td><td>O'Brien, Dave</td><td>2371.00</td><td>52.35</td><td>4429.130800</td></tr><tr><td>2</td><td>54</td><td>Tiano, Mike</td><td>3031.00</td><td>429.20</td><td>606.197500</td></tr><tr><td>3</td><td>17</td><td>Jones, TiAnna</td><td>2809.61</td><td>420.00</td><td>568.954700</td></tr><tr><td>4</td><td>12</td><td>Ray, Mike</td><td>1576.80</td><td>238.00</td><td>562.521000</td></tr><tr><td>5</td><td>70</td><td>Ginters, Kaspars</td><td>3976.75</td><td>700.00</td><td>468.107100</td></tr><tr><td>6</td><td>27</td><td>Schmöller, Martin</td><td>1296.00</td><td>249.70</td><td>419.022800</td></tr><tr><td>7</td><td>34</td><td>Cohen, Shy</td><td>23821.20</td><td>6022.77</td><td>295.519000</td></tr><tr><td>8</td><td>81</td><td>Nagel, Jean-Philippe</td><td>4234.26</td><td>1320.40</td><td>220.680000</td></tr></table><div>Query...MCRURYA1B7\SQLEXPRESS (11.0...MCRURYA1B7\mcrury (72)TSQL201200:00:0091 rows</div></div>		custid	contactname	salesamt2008	salesamt2007	percentgrowth	1	74	O'Brien, Dave	2371.00	52.35	4429.130800	2	54	Tiano, Mike	3031.00	429.20	606.197500	3	17	Jones, TiAnna	2809.61	420.00	568.954700	4	12	Ray, Mike	1576.80	238.00	562.521000	5	70	Ginters, Kaspars	3976.75	700.00	468.107100	6	27	Schmöller, Martin	1296.00	249.70	419.022800	7	34	Cohen, Shy	23821.20	6022.77	295.519000	8	81	Nagel, Jean-Philippe	4234.26	1320.40	220.680000
	custid	contactname	salesamt2008	salesamt2007	percentgrowth																																																		
1	74	O'Brien, Dave	2371.00	52.35	4429.130800																																																		
2	54	Tiano, Mike	3031.00	429.20	606.197500																																																		
3	17	Jones, TiAnna	2809.61	420.00	568.954700																																																		
4	12	Ray, Mike	1576.80	238.00	562.521000																																																		
5	70	Ginters, Kaspars	3976.75	700.00	468.107100																																																		
6	27	Schmöller, Martin	1296.00	249.70	419.022800																																																		
7	34	Cohen, Shy	23821.20	6022.77	295.519000																																																		
8	81	Nagel, Jean-Philippe	4234.26	1320.40	220.680000																																																		

Praktikum – Bagian 12: Inline TVF - Membuat query SELECT untuk mendapatkan total jumlah penjualan (nominal) untuk setiap customer

Langkah	Keterangan
1	<p>[Soal-14] Dengan menggunakan database TSQL2012, buatlah query SELECT terhadap view Sales.OrderValues yang berisi kolom <i>custid</i> dan kolom <i>totalsalesamount</i> (total dari kolom <i>val</i>). Filter hasilnya agar hanya menampilkan order pada tahun 2017 saja.</p>
2	<p>Eksekusi query pada langkah 1 di atas dan bandingkan dengan hasil yang ditunjukkan pada tampilan berikut ini:</p> 
3	<p>[Soal-15] Buatlah sebuah inline TVF/ Table-Valued Function dengan menambahkan baris berikut ini dan diletakkan sebelum query SELECT pada Langkah 1 di atas</p> <pre> CREATE FUNCTION dbo.fnGetSalesByCustomer (@orderyear AS INT) RETURNS TABLE AS RETURN CREATE FUNCTION dbo.fnGetSalesByCustomer (@orderyear AS INT) RETURNS TABLE AS RETURN SELECT custid, SUM(val) AS totalsalesamount FROM Sales.OrderValues WHERE YEAR(orderdate) = 2007 GROUP BY custid; </pre>
4	<p>[Soal-16] Modifikasilah query tersebut dengan mengganti nilai konstanta tahun 2007 pada klausa WHERE, dengan parameter @orderyear.</p>

	<pre>FROM Sales.OrderValues WHERE YEAR(orderdate) = @orderyear GROUP BY custid;</pre>	
5	Jalankan skrip pada langkah 4 di atas sehingga akan terbentuk sebuah inline TVF bernama dbo.fnGetSalesByCustomer	
		

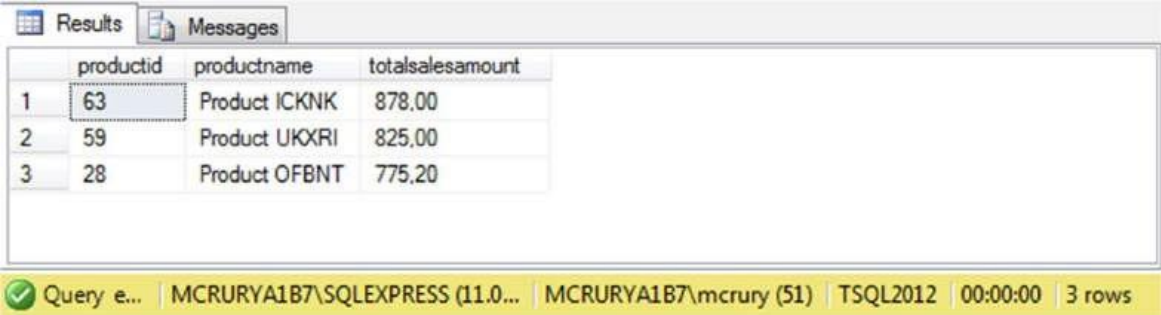
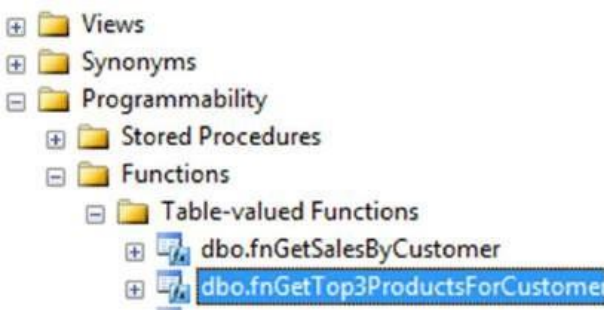
Praktikum – Bagian 12: Inline ITF - Membuat query SELECT yang beroperasi pada inline table-valued function

Langkah	Keterangan
1	<p>[Soal-17] Buatlah sebuah query SELECT yang berisi kolom <i>custid</i> dan <i>totalsalesamount</i> terhadap inline TVF dbo.fnGetSalesByCustomer. Masukkan nilai 2007 sebagai parameternya.</p> <pre>SELECT custid, totalsalesamount FROM dbo.fnGetSalesByCustomer(2007);</pre>
2	<p>Eksekusi query pada langkah 1 di atas dan bandingkan dengan hasil yang ditunjukkan pada tampilan berikut ini:</p> 

Praktikum – Bagian 13: Inline ITF - Membuat query SELECT untuk mendapatkan 3 produk terlaris untuk customer tertentu

Langkah	Keterangan
1	[Soal-18] Buatlah query SELECT yang menampilkan 3 produk terlaris untuk customer dengan ID = 1. Dapatkan kolom <i>productid</i> dan <i>productname</i> dari tabel Production.Products . Gunakan kolom <i>qty</i> dan <i>unitprice</i> dari tabel <i>Sales.OrderDetails</i> untuk menghitung nominal setiap baris

	<p>order, yang kemudian nominal tersebut dijumlahkan untuk setiap produk sehingga menghasilkan kolom <i>totalsalesamount</i>. Filter hasil tersebut agar hanya menampilkan data dengan nilai custid = 1.</p>
--	--

2	<p>Eksekusi query pada langkah 1 di atas dan bandingkan dengan hasil yang ditunjukkan pada tampilan berikut ini:</p> 
3	<p>[Soal-19] Dengan menggunakan query SELECT pada langkah 1 di atas, buatlah sebuah inline TVF dengan menambahkan beberapa baris fungsi sebelum query SELECT dan set nilai konstanta <i>custid</i> dalam query dengan parameter @custid, seperti berikut ini:</p> <pre>CREATE FUNCTION dbo.fnGetTop3ProductsForCustomer (@custid AS INT) RETURNS TABLE AS RETURN</pre>
4	<p>Jalankan skrip tersebut sehingga akan terbentuk sebuah inline TVF bernama dbo.fnGetTop3ProductsForCustomer yang memiliki parameter customer ID.</p> 
5	<p>[Soal-20] Lakukan ujicoba dengan membuat sebuah query SELECT pada inline TVF tersebut dan masukkan nilai 1 sebagai parameter customer ID-nya. Tampilkan kolom <i>productid</i>, <i>productname</i>, <i>totalsalesamount</i>, serta beri nama alias <i>p</i> untuk inline TVF-nya.</p>
6	<p>Eksekusi query pada langkah 1 di atas dan bandingkan dengan hasil yang ditunjukkan pada tampilan berikut ini:</p>

Results Messages			
	productid	productname	totalsalesamount
1	63	Product ICKNK	878,00
2	59	Product UKXRI	825,00
3	28	Product OFBNT	775,20

Query e... | MCRURYA1B7\SQLEXPRESS (11.0... | MCRURYA1B7\mcrury (51) | TSQL2012 | 00:00:00 | 3 rows

Praktikum – Bagian 14: Inline TVF - Menghapus inline Table-valued function

Langkah	Keterangan
1	<p>Hapuslah inline TVF yang sudah dibuat dengan menjalankan skrip berikut:</p> <pre>IF OBJECT_ID('dbo.fnGetSalesByCustomer') IS NOT NULL DROP FUNCTION dbo.fnGetSalesByCustomer; IF OBJECT_ID('dbo.fnGetTop3ProductsForCustomer') IS NOT NULL DROP FUNCTION dbo.fnGetTop3ProductsForCustomer;</pre>

--- Selamat Mengerjakan ---