

# **JOBSHEET**

## **PRAKTIKUM BASIS DATA LANJUT**

**Jurusan Teknologi Informasi**  
**POLITEKNIK NEGERI MALANG**



### **PERTEMUAN 4**

**SQL SERVER - SUBQUERY, GROUPING, DAN  
AGGREGATING**

#### **Team Teaching:**

Dwi Puspitasari, S.Kom., M.Kom.

Yan Watequlis Syaifudin, ST., MMT., PhD.

Ariadi Retno Ririd, S.Kom., M.Kom

Eka Larasati Amalia, S.ST., MT.

Annisa Puspa Kirana, S. Kom, M.Kom

Milyun Ni'ma Shoumi, S.Kom., M.Kom

Dika Rizky Yuniato, S.Kom, M.Kom

Irsyad Arif Mashudi, S.Kom M.Kom

Muhammad Shulhan Khairy, S.Kom, M.Kom

Yoppy Yunhasnawa, S.ST., M.Sc.

Alvionitha Sari Agstriningtyas, S.Kom.,M.Tr.T



Jurusan Teknologi Informasi Politeknik Negeri Malang

### **Jobsheet 3: Subquery, Grouping, dan Aggregating**

**Mata Kuliah Basis Data Lanjut**

**Pengampu:** Tim Ajar Basis Data Lanjut

*September 2022*

---

#### **Tujuan**

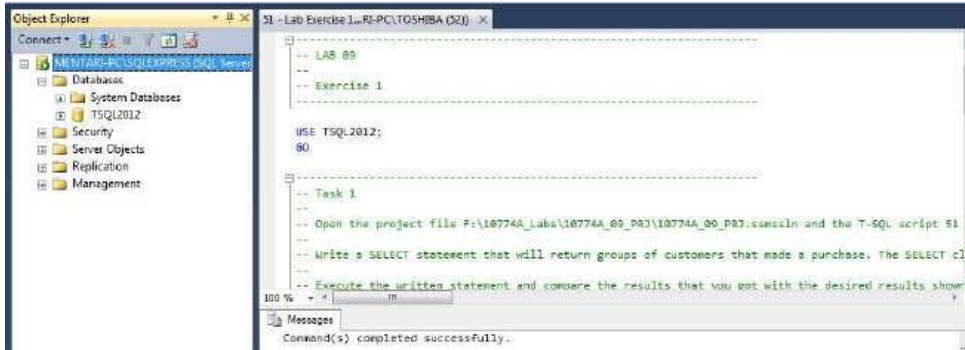
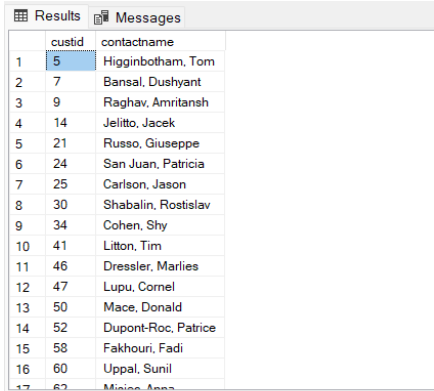
Mahasiswa diharapkan dapat:

1. Memahami perbedaan mendasar antara Transact-SQL (T-SQL) dengan SQL ANSI.
2. Memahami cara membuat *database* dari file SQL yang sudah ada
3. Memahami cara mengeksekusi sebagian maupun keseluruhan *script* SQL dari file yang sudah ada.
4. Memahami konsep penggunaan '*comment*' pada T-SQL.
5. Memahami konsep penggunaan statement SELECT untuk menganalisa tabel yang sudah ada pada *database*.
6. Memahami cara menampilkan data secara *unique/distinct*.
7. Memahami cara menggunakan *ALIAS* untuk nama tabel dan nama kolom.
8. Memahami konsep ekspresi *CASE* dan cara penggunaannya.
9. Mahasiswa memahami cara querying banyak tabel dalam klausa SELECT menggunakan JOIN
10. Mahasiswa memahami cara menuliskan query INNER JOIN, OUTER JOIN, SELF-JOIN dan CROSS JOIN
11. Mahasiswa memahami cara melakukan Sorting Data, Filtering Data dengan predikat, Filtering Data dengan TOP dan OFFSET-FETCH
12. Mahasiswa memahami cara melakukan Penanganan nilai yang hilang dan tidak diketahui pada data real.

#### **Petunjuk Umum**

1. Ikuti langkah-langkah pada bagian-bagian praktikum sesuai dengan urutan yang diberikan.
2. Jawablah semua pertanyaan bertanda [Soal-X] yang terdapat pada langkah-langkah tertentu di setiap bagian praktikum.
3. 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.
4. 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:
  - **BDL\_Kelas\_03\_NamaLengkapAnda.pdf**
  - 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: Menulis Query Menggunakan Klausu GROUP BY

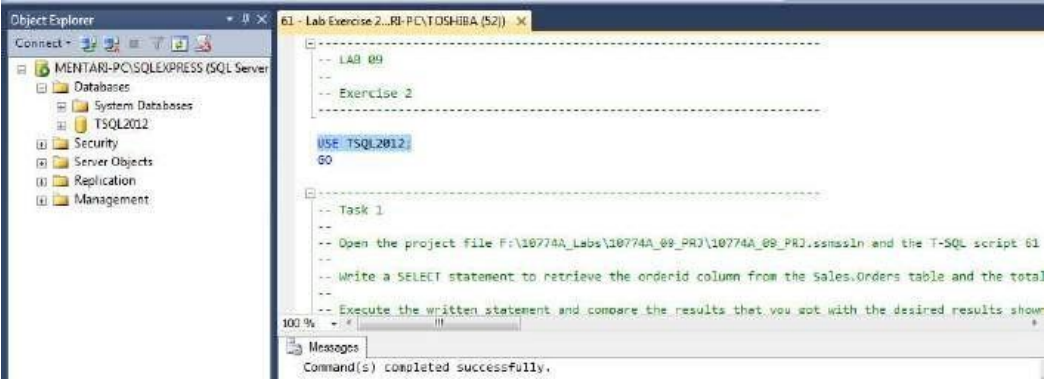
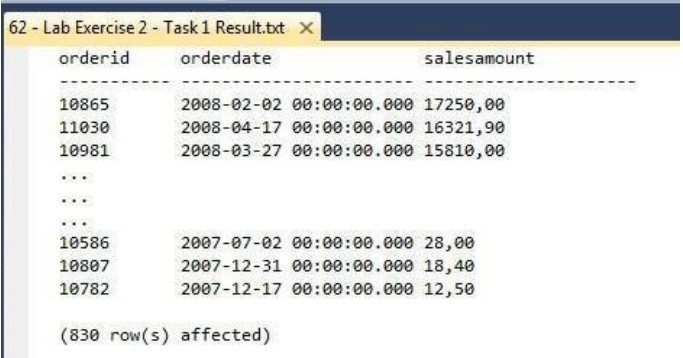
Langkah	Keterangan
1	<p>Skenario :</p> <p>Departemen penjualan suatu perusahaan ingin menciptakan peluang <i>up-sell</i> tambahan dari para pelanggan. Untuk itu karyawan butuh melakukan analisis mengenai berbagai kelompok pelanggan dan kategori produk berdasar pada beberapa peraturan bisnis. Dengan adanya skenario tersebut, maka diperlukan pernyataan T-SQL menggunakan klausa SELECT untuk mengambil baris yang dibutuhkan dari tabel Sales.Customers.</p> <p>Untuk melakukan percobaan pada praktikum bagian 1 ini terlebih dahulu lakukan login pada SQL Server Management Studio (SSMS). Kemudian buka folder Minggu6/1, dan buka script 51 - Lab Exercise 1.sql. Pastikan database terhubung dengan “TSQL2012”.</p>
	
2	<p><b>[Soal-1]</b> Tuliskan T-SQL SELECT yang akan menampilkan kelompok pelanggan yang melakukan pembelian. Klausu SELECT harus mencakup kolom custid dari tabel Sales.Orders dan kolom contactname dari tabel Sales.Customers. Kelompokkan kedua kolom tersebut, dan filter hanya pesanan dari sales employee yang memiliki empid sama dengan 5!</p> <p><b>Jawab:</b></p> <div>  <pre> SELECT     o.[custid], c.[contactname] FROM [Sales].[Orders] AS o INNER JOIN [Sales].[Customers] AS c ON o.[custid] = c.[custid] WHERE o.[empid] = 5 GROUP BY o.custid, c.contactname; </pre> </div>



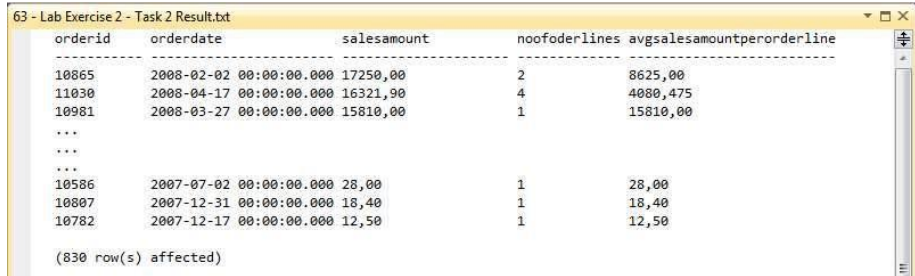
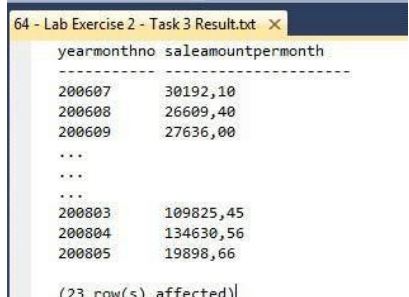
3	<p>Bandingkan hasil pada tahap 2 dengan file 52 - Lab Exercise 1 - Task 1 Result.txt. Jika sama maka T-SQL yang Anda tuliskan sudah benar.</p> <pre>52 - Lab Exercise 1 - Task 1 Result.txt X custid    contactname ----- 5         Higginbotham, Tom 7         Bansal, Dushyant 9         Raghav, Amritansh ... ... ... 87        Ludwig, Michael 89        Smith Jr., Ronaldo 91        Conn, Steve (29 row(s) affected)</pre>
4	<p>[Soal-2] Salin T-SQL jawaban soal-1. Kemudian modifikasi untuk menampilkan informasi tambahan kolom city dari tabel Sales.Customers pada klausa SELECT!</p> <p>Jawab:</p> <pre>SELECT     o.[custid], c.[contactname], c.[city] FROM [Sales].[Orders] AS o INNER JOIN [Sales].[Customers] AS c ON o.[custid] = c.[custid] WHERE o.[empid] = 5 GROUP BY o.custid, c.contactname;</pre>
5	<p>[Soal-3] Apakah terdapat pesan error pada jawaban soal-2? Apakah pesan errornya? Kenapa pesan itu bisa terjadi?</p> <p>Jawaban: ya, karena jika c.city tidak di panggil pada klausa group by</p>
6	<p>[Soal-4] Perbaiki error yang terjadi pada jawaban soal-2! Jika hasil eksekusinya sama dengan file 53 - Lab Exercise 1 - Task 2 Result.txt, maka T-SQL yang dibuat sudah tepat.</p> <pre>53 - Lab Exercise 1 - Task 2 Result.txt X custid    contactname    city ----- 5         Higginbotham, Tom    Luleå 7         Bansal, Dushyant    Strasbourg 9         Raghav, Amritansh    Marseille ... ... ... 87        Ludwig, Michael    Oulu 89        Smith Jr., Ronaldo    Seattle 91        Conn, Steve        Warszawa (29 row(s) affected)</pre>
7	<p>[Soal-5] Tuliskan pernyataan SELECT yang akan menampilkan kelompok baris berdasarkan kolom custid dan akan dihitung oleh kolom orderyear mewakili tahun pesanan berdasarkan kolom orderdate dari tabel Sales.Orders. Kemudian filter hasilnya untuk memasukkan hanya pesanan dari karyawan penjualan yang empid nya sama dengan 5!</p>

	<pre>SELECT     custid, YEAR(orderdate) AS orderyear FROM [Sales].[Orders] WHERE empid = 5 GROUP BY custid, YEAR(orderdate) ORDER BY custid, orderyear;</pre>																																																																										
8	<p>Bandingkan hasil pada soal-5 dengan file 54 - Lab Exercise 1 - Task 3 Result.txt. Jika sama maka T-SQL yang Anda tuliskan sudah benar</p> <div><div>54 - Lab Exercise1 - Task 3 Result.txt X</div><div><table><thead><tr><th>custid</th><th>orderyear</th></tr></thead><tbody><tr><td>5</td><td>2007</td></tr><tr><td>5</td><td>2008</td></tr><tr><td>7</td><td>2006</td></tr><tr><td>...</td><td></td></tr><tr><td>...</td><td></td></tr><tr><td>...</td><td></td></tr><tr><td>87</td><td>2006</td></tr><tr><td>89</td><td>2006</td></tr><tr><td>91</td><td>2008</td></tr></tbody></table><p>(34 row(s) affected)</p></div><div><div>ResultsMessages</div><table><thead><tr><th></th><th>custid</th><th>orderyear</th></tr></thead><tbody><tr><td>1</td><td>5</td><td>2007</td></tr><tr><td>2</td><td>5</td><td>2008</td></tr><tr><td>3</td><td>7</td><td>2006</td></tr><tr><td>4</td><td>9</td><td>2007</td></tr><tr><td>5</td><td>14</td><td>2006</td></tr><tr><td>6</td><td>21</td><td>2007</td></tr><tr><td>7</td><td>24</td><td>2006</td></tr><tr><td>8</td><td>25</td><td>2007</td></tr><tr><td>9</td><td>30</td><td>2008</td></tr><tr><td>10</td><td>34</td><td>2008</td></tr><tr><td>11</td><td>41</td><td>2006</td></tr><tr><td>12</td><td>46</td><td>2008</td></tr><tr><td>13</td><td>47</td><td>2008</td></tr><tr><td>14</td><td>50</td><td>2007</td></tr><tr><td>15</td><td>52</td><td>2007</td></tr><tr><td>16</td><td>58</td><td>2007</td></tr><tr><td>17</td><td>60</td><td>2006</td></tr></tbody></table><p>LAPTOP-CEEMFUHE (16.0 RTM) LAPTOP-CEEMFUHE\JAG (66)</p></div></div>	custid	orderyear	5	2007	5	2008	7	2006	...		...		...		87	2006	89	2006	91	2008		custid	orderyear	1	5	2007	2	5	2008	3	7	2006	4	9	2007	5	14	2006	6	21	2007	7	24	2006	8	25	2007	9	30	2008	10	34	2008	11	41	2006	12	46	2008	13	47	2008	14	50	2007	15	52	2007	16	58	2007	17	60	2006
custid	orderyear																																																																										
5	2007																																																																										
5	2008																																																																										
7	2006																																																																										
...																																																																											
...																																																																											
...																																																																											
87	2006																																																																										
89	2006																																																																										
91	2008																																																																										
	custid	orderyear																																																																									
1	5	2007																																																																									
2	5	2008																																																																									
3	7	2006																																																																									
4	9	2007																																																																									
5	14	2006																																																																									
6	21	2007																																																																									
7	24	2006																																																																									
8	25	2007																																																																									
9	30	2008																																																																									
10	34	2008																																																																									
11	41	2006																																																																									
12	46	2008																																																																									
13	47	2008																																																																									
14	50	2007																																																																									
15	52	2007																																																																									
16	58	2007																																																																									
17	60	2006																																																																									
9	<p>[Soal-6] Tuliskan pernyataan SELECT yang akan mengembalikan kelompok baris berdasarkan kolom categoryname di tabel Production.Categories. Kemudian filter hasilnya hanya untuk product categories yang dipesan pada tahun 2008!</p> <pre>select c.categoryid,c.categoryname from Sales.orders o INNER JOIN Sales.OrderDetails d ON o.orderid=d.orderid INNER JOIN Production.Products p ON p.productid=p.productid INNER JOIN Production.Categories c ON c.categoryid=p.categoryid WHERE o.orderdate &gt;='20080101' AND o.orderdate &lt;'20090101' GROUP BY c.categoryid, c.categoryname order by c.categoryid ASC;</pre>																																																																										
10	<p>Bandingkan hasil pada soal-6 dengan file 55 - Lab Exercise 1 - Task 4 Result.txt. Jika sama maka T-SQL yang Anda tuliskan sudah benar</p> <div><div>55 - Lab Exercise1 - Task 4 Result.txt X</div><div><table><thead><tr><th>categoryid</th><th>categoryname</th></tr></thead><tbody><tr><td>1</td><td>Beverages</td></tr><tr><td>2</td><td>Condiments</td></tr><tr><td>3</td><td>Confections</td></tr><tr><td>4</td><td>Dairy Products</td></tr><tr><td>5</td><td>Grains/Cereals</td></tr><tr><td>6</td><td>Meat/Poultry</td></tr><tr><td>7</td><td>Produce</td></tr><tr><td>8</td><td>Seafood</td></tr></tbody></table><p>(8 row(s) affected)</p></div><div><div>ResultsMessages</div><table><thead><tr><th></th><th>categoryid</th><th>categoryname</th></tr></thead><tbody><tr><td>1</td><td>1</td><td>Beverages</td></tr><tr><td>2</td><td>2</td><td>Condiments</td></tr><tr><td>3</td><td>3</td><td>Confections</td></tr><tr><td>4</td><td>4</td><td>Dairy Products</td></tr><tr><td>5</td><td>5</td><td>Grains/Cereals</td></tr><tr><td>6</td><td>6</td><td>Meat/Poultry</td></tr><tr><td>7</td><td>7</td><td>Produce</td></tr><tr><td>8</td><td>8</td><td>Seafood</td></tr></tbody></table></div></div>	categoryid	categoryname	1	Beverages	2	Condiments	3	Confections	4	Dairy Products	5	Grains/Cereals	6	Meat/Poultry	7	Produce	8	Seafood		categoryid	categoryname	1	1	Beverages	2	2	Condiments	3	3	Confections	4	4	Dairy Products	5	5	Grains/Cereals	6	6	Meat/Poultry	7	7	Produce	8	8	Seafood																													
categoryid	categoryname																																																																										
1	Beverages																																																																										
2	Condiments																																																																										
3	Confections																																																																										
4	Dairy Products																																																																										
5	Grains/Cereals																																																																										
6	Meat/Poultry																																																																										
7	Produce																																																																										
8	Seafood																																																																										
	categoryid	categoryname																																																																									
1	1	Beverages																																																																									
2	2	Condiments																																																																									
3	3	Confections																																																																									
4	4	Dairy Products																																																																									
5	5	Grains/Cereals																																																																									
6	6	Meat/Poultry																																																																									
7	7	Produce																																																																									
8	8	Seafood																																																																									
11	<p><b>Kesimpulan :</b> Setelah menjalankan praktikum bagian ini, mahasiswa bisa menggunakan klausa GROUP BY dalam pernyataan T-SQL.</p>																																																																										

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

Langkah	Keterangan
1	<p>Skenario :</p> <p>Bagian pemasaran ingin meluncurkan kampanye baru, sehingga karyawan perlu mendapatkan wawasan yang lebih baik mengenai perilaku pembelian para pelanggan. Oleh karena itu, harus dibuat laporan penjualan yang berbeda yang didasarkan pada jumlah penjualan rata-rata per tahun per pelanggan</p> <p>Untuk melakukan percobaan pada praktikum bagian 2 ini, buka folder Minggu6/1, dan buka script 61 - Lab Exercise 2.sql. Pastikan database terhubung dengan “TSQL2012”.</p>
	
2	<p>[Soal-7] Tuliskan pernyataan SELECT yang akan mengembalikan kolom orderid, orderdate dari tabel Sales.Orders dan total sales amount per orderid (Petunjuk : Kalikan kolom qty dan unitprice dari tabel Sales.OrderDetails) Gunakan alias salesamount untuk kolom yang dihitung. Kemudian urutkan hasilnya dengan total sales amount dalam urutan menurun!</p> <pre> SELECT     o.[orderid],     o.[orderdate],     SUM(d.[qty] * d.[unitprice]) AS salesamount FROM [Sales].[Orders] AS o JOIN [Sales].[OrderDetails] AS d     ON o.orderid=d.orderid GROUP BY     o.[orderid], o.[orderdate] ORDER BY     salesamount DESC; </pre>
3	<p>Bandingkan hasil pada soal-7 dengan file 62 - Lab Exercise 2 - Task 1 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar</p> 

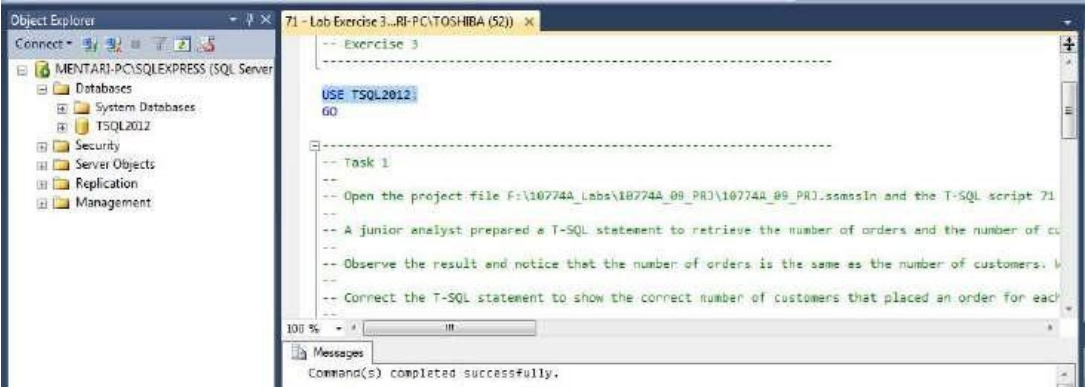


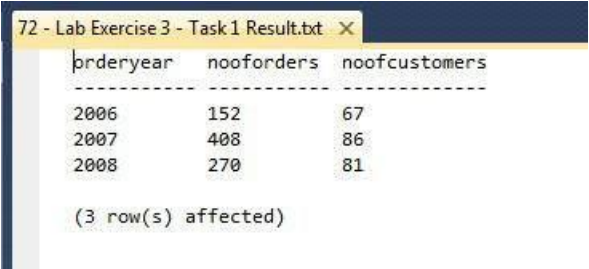
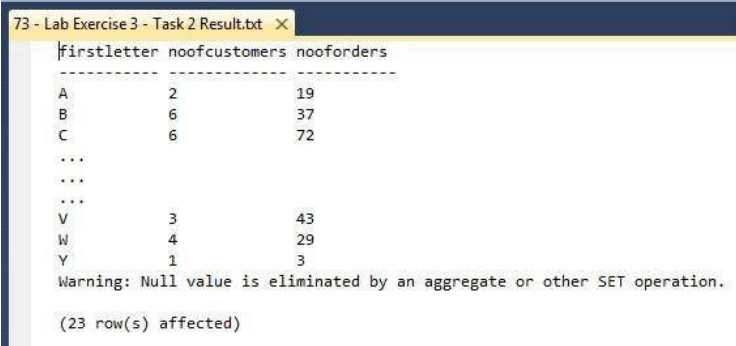
4	<p><b>[Soal-8]</b> Salin pernyataan T-SQL pada jawaban soal-7 dan modifikasi dengan memasukkan jumlah order lines untuk setiap order dan nilai rata-rata sales amount per orderid sesuai pesanan. Gunakan nama alias masing-masing nooforderlines dan avgsalesamountperorderlines!</p> <pre>SELECT     o.[orderid],     o.[orderdate],     SUM(d.[qty] * d.[unitprice]) AS salesamount,     COUNT(d.orderid) AS nooforderlines,     SUM(d.qty*d.unitprice) / COUNT(DISTINCT o.orderid) salesamountperorderlines FROM [Sales].[Orders] AS o JOIN [Sales].[OrderDetails] AS d     ON o.orderid=d.orderid GROUP BY     o.orderid, o.orderdate ORDER BY     salesamountperorderlines DESC;</pre>																																								
5	<p>Bandingkan hasil pada soal-8 dengan file 63 - Lab Exercise 2 - Task 2 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p>																																								
	 <table><tr><th>orderid</th><th>orderdate</th><th>salesamount</th><th>nooforderlines</th><th>avgsalesamountperorderline</th></tr><tr><td>10865</td><td>2008-02-02 00:00:00.000</td><td>17250,00</td><td>2</td><td>8625,00</td></tr><tr><td>11030</td><td>2008-04-17 00:00:00.000</td><td>16321,90</td><td>4</td><td>4080,475</td></tr><tr><td>10981</td><td>2008-03-27 00:00:00.000</td><td>15810,00</td><td>1</td><td>15810,00</td></tr><tr><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td></tr><tr><td>10586</td><td>2007-07-02 00:00:00.000</td><td>28,00</td><td>1</td><td>28,00</td></tr><tr><td>10807</td><td>2007-12-31 00:00:00.000</td><td>18,40</td><td>1</td><td>18,40</td></tr><tr><td>10782</td><td>2007-12-17 00:00:00.000</td><td>12,50</td><td>1</td><td>12,50</td></tr></table> <p>(830 row(s) affected)</p>	orderid	orderdate	salesamount	nooforderlines	avgsalesamountperorderline	10865	2008-02-02 00:00:00.000	17250,00	2	8625,00	11030	2008-04-17 00:00:00.000	16321,90	4	4080,475	10981	2008-03-27 00:00:00.000	15810,00	1	15810,00	...	...	...	...	...	10586	2007-07-02 00:00:00.000	28,00	1	28,00	10807	2007-12-31 00:00:00.000	18,40	1	18,40	10782	2007-12-17 00:00:00.000	12,50	1	12,50
orderid	orderdate	salesamount	nooforderlines	avgsalesamountperorderline																																					
10865	2008-02-02 00:00:00.000	17250,00	2	8625,00																																					
11030	2008-04-17 00:00:00.000	16321,90	4	4080,475																																					
10981	2008-03-27 00:00:00.000	15810,00	1	15810,00																																					
...	...	...	...	...																																					
10586	2007-07-02 00:00:00.000	28,00	1	28,00																																					
10807	2007-12-31 00:00:00.000	18,40	1	18,40																																					
10782	2007-12-17 00:00:00.000	12,50	1	12,50																																					
6	<p><b>[Soal-9]</b> Tuliskan pernyataan SELECT untuk mengambil jumlah penjualan total untuk setiap bulannya! Penggunaan klausa SELECT seharusnya menyertakan perhitungan kolom yearmonthno (notasi YYYYMM) berdasarkan kolom orderdate pada tabel Sales.Orders dan total jumlah penjualan (Perkalian kolom qty dengan unitprice dari tabel Sales.OrderDetils) yang diberi alias saleamountpermonth. Urutan hasilnya didasarkan pada perhitungan kolom yearmonthno.</p> <pre>SELECT     FORMAT(o.[orderdate], 'yyyyMM') AS yearmonthno,     SUM(d.[qty] * d.[unitprice]) AS saleamountpermonth FROM [Sales].[Orders] AS o JOIN [Sales].[OrderDetails] AS d     ON o.orderid=d.orderid GROUP BY     FORMAT(o.[orderdate], 'yyyyMM') ORDER BY yearmonthno;</pre>																																								
7	<p>Bandingkan hasil pada soal-9 dengan file 64 - Lab Exercise 2 - Task 3 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p>  <table><tr><th>yearmonthno</th><th>saleamountpermonth</th></tr><tr><td>200607</td><td>30192,10</td></tr><tr><td>200608</td><td>26609,40</td></tr><tr><td>200609</td><td>27636,00</td></tr><tr><td>...</td><td>...</td></tr><tr><td>200803</td><td>109825,45</td></tr><tr><td>200804</td><td>134630,56</td></tr><tr><td>200805</td><td>19898,66</td></tr></table> <p>(23 row(s) affected)</p>	yearmonthno	saleamountpermonth	200607	30192,10	200608	26609,40	200609	27636,00	...	...	200803	109825,45	200804	134630,56	200805	19898,66																								
yearmonthno	saleamountpermonth																																								
200607	30192,10																																								
200608	26609,40																																								
200609	27636,00																																								
...	...																																								
200803	109825,45																																								
200804	134630,56																																								
200805	19898,66																																								

8	<p>[Soal-10] Tulislah perintah SELECT yang akan mengambil semua pelanggan/customer (termasuk yang tidak memiliki pesanan) dan jumlah penjualan, jumlah pesanan maksimum per baris, dan jumlah pesanan! Klausula SELECT harus memasukkan kolom custid dan contactname dari tabel Sales.Customers dan 4 (empat) kolom yang dikalkulasi berdasarkan fungsi agregasi sebagai berikut :</p> <ul style="list-style-type: none"><li>1) totalsalesamount, adalah alias untuk jumlah penjualan total per pesanan</li><li>2) maxsalesamountperorderline, adalah alias untuk jumlah penjualan maksimum per baris pesanan</li><li>3) numberofrows, adalah alias untuk jumlah baris (gunakan * dalam fungsi COUNT)</li><li>4) numberoforderlines, adalah alias untuk jumlah baris pesanan (gunakan kolom orderid di kolom fungsi COUNT)</li></ul> <p>Urutkan hasilnya berdasarkan kolom totalsalesamount.</p> <pre>SELECT     c.[custid],     c.[contactname],     COALESCE(SUM(d.[qty] * d.[unitprice]), NULL) AS totalsalesamount,     COALESCE(MAX(d.[qty] * d.[unitprice]), NULL) AS maxsalesamountperorderline,     COUNT(*) AS numberofrows,     COUNT(od.[orderid]) AS numberoforderlines FROM     [Sales].[Customers] AS c LEFT JOIN     [Sales].[Orders] AS o ON c.[custid] = o.[custid] LEFT JOIN     [Sales].[OrderDetails] AS od ON o.[orderid] = od.[orderid] GROUP BY     c.[custid], c.[contactname] ORDER BY     totalsalesamount;</pre>																																																																						
9	<p>Bandingkan hasil pada soal-10 dengan file 65 - Lab Exercise 2 - Task 4 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p>																																																																						
	<div>65 - Lab Exercise 2 - Task 4 Result.txt</div> <table><tr><th></th><th>custid</th><th>contactname</th><th>totalsalesamount</th><th>maxsalesamountperorderline</th><th>numberofrows</th><th>numberoforderlines</th></tr><tr><td>22</td><td></td><td>Bueno, Janaina Burdan, Neville</td><td>NULL</td><td>NULL</td><td>1</td><td>0</td></tr><tr><td>57</td><td></td><td>Tollefsen, Bjørn</td><td>NULL</td><td>NULL</td><td>1</td><td>0</td></tr><tr><td>13</td><td></td><td>Benito, Almudena</td><td>100,80</td><td>80,00</td><td>2</td><td>2</td></tr><tr><td>...</td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>...</td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>...</td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>20</td><td></td><td>Kane, John</td><td>113236,68</td><td>6360,00</td><td>102</td><td>102</td></tr><tr><td>71</td><td></td><td>Navarro, Tomás</td><td>115673,39</td><td>7427,40</td><td>116</td><td>116</td></tr><tr><td>63</td><td></td><td>Veronesi, Giorgio</td><td>117483,39</td><td>15810,00</td><td>86</td><td>86</td></tr></table> <p>Warning: Null value is eliminated by an aggregate or other SET operation.</p> <p>(91 row(s) affected)</p>		custid	contactname	totalsalesamount	maxsalesamountperorderline	numberofrows	numberoforderlines	22		Bueno, Janaina Burdan, Neville	NULL	NULL	1	0	57		Tollefsen, Bjørn	NULL	NULL	1	0	13		Benito, Almudena	100,80	80,00	2	2	...							...							...							20		Kane, John	113236,68	6360,00	102	102	71		Navarro, Tomás	115673,39	7427,40	116	116	63		Veronesi, Giorgio	117483,39	15810,00	86	86
	custid	contactname	totalsalesamount	maxsalesamountperorderline	numberofrows	numberoforderlines																																																																	
22		Bueno, Janaina Burdan, Neville	NULL	NULL	1	0																																																																	
57		Tollefsen, Bjørn	NULL	NULL	1	0																																																																	
13		Benito, Almudena	100,80	80,00	2	2																																																																	
...																																																																							
...																																																																							
...																																																																							
20		Kane, John	113236,68	6360,00	102	102																																																																	
71		Navarro, Tomás	115673,39	7427,40	116	116																																																																	
63		Veronesi, Giorgio	117483,39	15810,00	86	86																																																																	
10	<p><b>Kesimpulan:</b> Setelah menjalankan praktikum bagian ini, maka seharusnya sekarang mahasiswa sudah mengetahui bagaimana menggunakan fungsi agregasi.</p>																																																																						



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

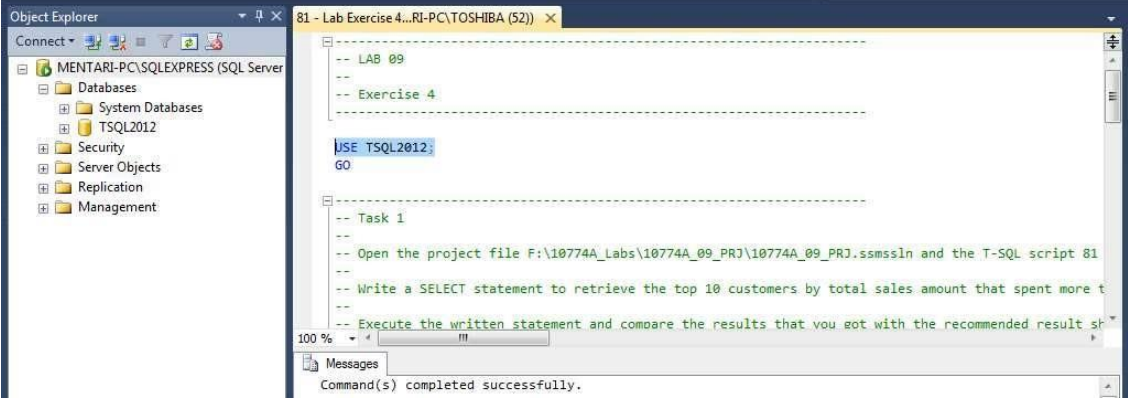
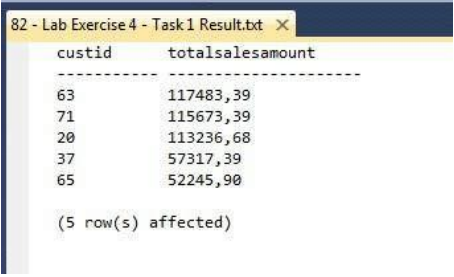
Langkah	Keterangan
1	<p>Skenario :</p> <p>Departemen pemasaran ingin memiliki beberapa laporan tambahan yang menunjukkan jumlah pelanggan yang memiliki pemesanan dalam jangka waktu tertentu dan jumlah pelanggan berdasarkan huruf pertama dan <i>contact name</i>.</p> <p>Untuk melakukan percobaan pada praktikum bagian 3 ini, buka folder Minggu6/1, dan buka script 71 - Lab Exercise 3.sql. Pastikan database terhubung dengan "TSQL2012".</p>
	
2	<p><b>[Soal-11]</b> Berdasarkan hasil eksekusi T-SQL di bawah ini, Kenapa jumlah pesanan (nooforders) sama dengan jumlah pelanggan (noofcustomers)?</p> <pre> SELECT YEAR(orderdate) AS orderyear, COUNT(orderid) AS nooforders, COUNT(custid) AS noofcustomers FROM Sales.Orders GROUP BY YEAR(orderdate); </pre> <p>Jawaban: karena setiap pelanggan hanya melakukan satu pesanan dalam setahun, maka jika setiap custid terhubung dengan orderid maka hasilnya akan sama</p>
3	<p><b>[Soal-12]</b> Perbaiki T-SQL pada soal-12 untuk menunjukkan jumlah pelanggan yang benar yang melakukan pemesanan setiap tahun!</p> <pre> SELECT YEAR(orderdate) AS orderyear, COUNT(orderid) AS nooforders, COUNT(DISTINCT custid) AS noofcustomers FROM Sales.Orders GROUP BY YEAR(orderdate); </pre>

4	<p>Bandungkan hasil pada soal-12 dengan file 72 - Lab Exercise 3 - Task 1 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p> 
5	<p>[Soal-13] Tuliskan pernyataan SELECT untuk mengambil jumlah pelanggan berdasarkan huruf pertama dari nilai pada kolom contactname dari tabel Sales.Customers. Tambahkan kolom yang menunjukkan jumlah pesanan yang dilakukan setiap grup pelanggan/customer. Gunakan alias masing-masing firstletter, noofcustomers dan nooforders. Urutkan hasilnya berdasarkan kolom firstletter!</p> <pre> SELECT     SUBSTRING(c.contactname,1,1) firstletter,     COUNT(DISTINCT c.custid) AS noofcustomers,     COUNT(o.orderid) AS nooforders     FROM Sales.Customers c     LEFT JOIN SALES.ORDERS o     on c.custid=o.custid group by SUBSTRING(c.contactname,1,1) order by firstletter; </pre>
6	<p>Bandungkan hasil pada soal-13 dengan file 73 - Lab Exercise 3 - Task 2 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p> 
7	<p>[Soal-14] Salin T-SQL pada jawaban soal-6 kemudian modifikasi dengan memasukkan informasi tentang setiap category produk : jumlah penjualan, jumlah pesanan, dan jumlah penjualan rata-rata setiap pemesanan. Gunakan nama alias masing-masing, nooforders, dan avgsalesamountperorder.</p>

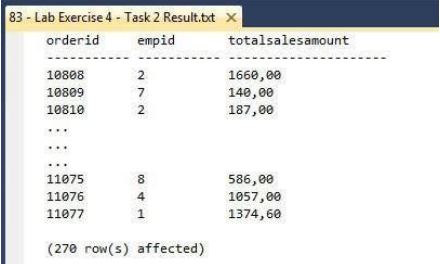
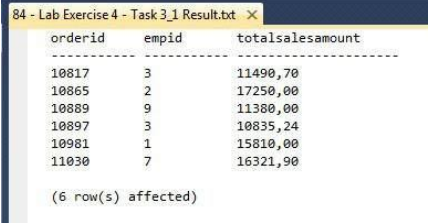


	<pre>select c.categoryid,        c.categoryname,        SUM(d.qty*d.unitprice) totalsalesamount,        COUNT(DISTINCT o.orderid) noofder,        SUM(d.qty*d.unitprice) / COUNT(DISTINCT o.orderid) avgsalesamount from Sales.orders o       INNER JOIN Sales.OrderDetails d ON o.orderid=d.orderid       INNER JOIN Production.Products p ON p.productid=d.productid       INNER JOIN Production.Categories c ON c.categoryid=p.categoryid WHERE o.orderdate &gt;='20080101' AND o.orderdate &lt;'20090101' GROUP BY c.categoryid, c.categoryname order by c.categoryid ASC;</pre>																																													
8	<p>Bandingkan hasil pada soal-13 dengan file 74 - Lab Exercise 3 - Task 3 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p> <div><div>74 - Lab Exercise 3 - Task 3 Result.txt</div><table><thead><tr><th>categoryid</th><th>categoryname</th><th>totalsalesamount</th><th>nooforders</th><th>avgsalesamountperorder</th></tr></thead><tbody><tr><td>1</td><td>Beverages</td><td>122223,75</td><td>128</td><td>954,873</td></tr><tr><td>2</td><td>Condiments</td><td>34557,45</td><td>62</td><td>557,3782</td></tr><tr><td>3</td><td>Confections</td><td>58359,73</td><td>89</td><td>655,7273</td></tr><tr><td>4</td><td>Dairy Products</td><td>82803,90</td><td>90</td><td>920,0433</td></tr><tr><td>5</td><td>Grains/Cereals</td><td>30422,25</td><td>55</td><td>553,1318</td></tr><tr><td>6</td><td>Meat/Poultry</td><td>60275,57</td><td>43</td><td>1401,7574</td></tr><tr><td>7</td><td>Produce</td><td>32415,85</td><td>42</td><td>771,8059</td></tr><tr><td>8</td><td>Seafood</td><td>48712,84</td><td>101</td><td>482,3053</td></tr></tbody></table><p>(8 row(s) affected)</p></div>	categoryid	categoryname	totalsalesamount	nooforders	avgsalesamountperorder	1	Beverages	122223,75	128	954,873	2	Condiments	34557,45	62	557,3782	3	Confections	58359,73	89	655,7273	4	Dairy Products	82803,90	90	920,0433	5	Grains/Cereals	30422,25	55	553,1318	6	Meat/Poultry	60275,57	43	1401,7574	7	Produce	32415,85	42	771,8059	8	Seafood	48712,84	101	482,3053
categoryid	categoryname	totalsalesamount	nooforders	avgsalesamountperorder																																										
1	Beverages	122223,75	128	954,873																																										
2	Condiments	34557,45	62	557,3782																																										
3	Confections	58359,73	89	655,7273																																										
4	Dairy Products	82803,90	90	920,0433																																										
5	Grains/Cereals	30422,25	55	553,1318																																										
6	Meat/Poultry	60275,57	43	1401,7574																																										
7	Produce	32415,85	42	771,8059																																										
8	Seafood	48712,84	101	482,3053																																										
9	<p><b>Kesimpulan:</b> Setelah melakukan praktikum bagian ini mahasiswa seharusnya memahami bagaimana menerapkan fungsi agregasi DISTINCT.</p>																																													

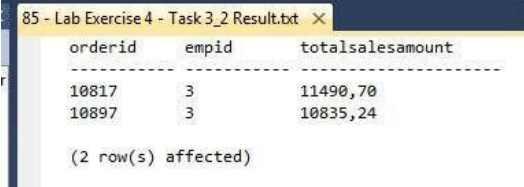
## Praktikum – Bagian 4: Menulis Query Yang Melakukan Filter Group Dengan Klausu HAVING

Langkah	Keterangan
1	<p>Skenario :</p> <p>Laporan tentang analisis tingkah laku pelanggan yang telah dibuat pada percobaan sebelumnya, telah memenuhi kebutuhan departemen penjualan dan pemasaran. Sekarang departemen tersebut membutuhkan laporan tersebut difilter berdasarkan total jumlah penjualan dan jumlah pesanan. Jadi skenario bagian ini akan membahas tata cara filter hasil uji coba sebelumnya berdasarkan fungsi agregasi dan mempelajari penggunaan klausa WHERE dan HAVING.</p> <p>Untuk melakukan percobaan pada praktikum bagian 4 ini, buka folder Minggu6/1, dan buka script 81 - Lab Exercise 4.sql. Pastikan database terhubung dengan "TSQL2012".</p>
	
2	<p><b>[Soal-15]</b> Tuliskan perintah T-SQL dengan klausa SELECT untuk mengambil 5 pelanggan teratas dengan penjualan total lebih dari \$10.000. Tampilkan kolom custid dari tabel order dan hitung kolom yang berisi jumlah penjualan berdasarkan kolom qty dan unitprice dari tabel Sales.OrderDetails. Gunakan alias o.Gr.</p> <pre> SELECT     o.custid,     SUM(d.qty * d.unitprice) AS totalsalesamount FROM [Sales].[Orders] AS o JOIN [Sales].[OrderDetails] AS d     ON o.orderid = d.orderid GROUP BY     o.custid HAVING     SUM(d.qty * d.unitprice) &gt; 10000 ORDER BY     totalsalesamount OFFSET 0 ROWS FETCH NEXT 5 ROWS ONLY; </pre>
3	<p>Bandingkan hasil pada soal-15 dengan file 82 - Lab Exercise 4 - Task 1 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p> 



4	<p>[Soal-16] Tuliskan perintah T-SQL dengan klausa SELECT untuk mengambil kolom empid,orderid dan kolom yang mempresentasikan perhitungan total penjualan (total sales amount) berdasarkan tabel Sales.Orders dan Sales.OrderDetails. Filter hasilnya menjadi grup baris data hanya untuk pesanan di tahun 2008!</p> <pre>SELECT     o.orderid,     o.empid,     SUM(d.qty * d.unitprice) AS totalsalesamount FROM [Sales].[Orders] AS o JOIN [Sales].[OrderDetails] AS d     ON o.orderid = d.orderid WHERE YEAR(o.orderdate) = 2008 GROUP BY     o.orderid,o.empid;</pre>
5	<p>Bandingkan hasil pada soal-16 dengan file 83 - Lab Exercise 4 - Task 2 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p>
	
6	<p>[Soal-17] Salin perintah T-SQL jawaban soal-16 dan modifikasi untuk menambahkan filter yang hanya mengambil baris yang memiliki jumlah penjualan lebih dari \$10.000!</p> <pre>SELECT     o.orderid,     o.empid,     SUM(d.qty * d.unitprice) AS totalsalesamount FROM [Sales].[Orders] AS o JOIN [Sales].[OrderDetails] AS d     ON o.orderid = d.orderid WHERE YEAR(o.orderdate) = 2008 GROUP BY     o.orderid,o.empid HAVING     SUM(d.qty * d.unitprice) &gt; 10000;</pre>
7	<p>Bandingkan hasil pada soal-17 dengan file 84 - Lab Exercise 4 - Task 3_1 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p> 



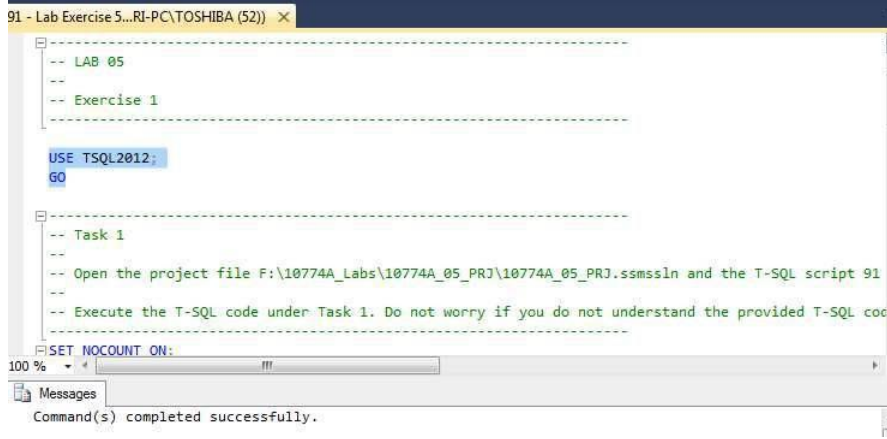
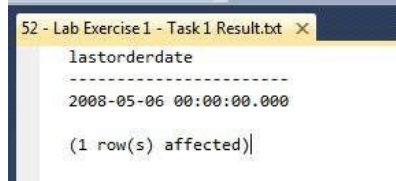
8	<p>[Soal-18] Salin perintah T-SQL jawaban soal-17 dan modifikasi untuk menambahkan filter yang hanya menampilkan pegawai dengan empid sama dengan 3(tiga)!</p> <pre>SELECT     o.orderid,     o.empid,     SUM(d.qty * d.unitprice) AS totalsalesamount FROM [Sales].[Orders] AS o JOIN [Sales].[OrderDetails] AS d     ON o.orderid = d.orderid WHERE     YEAR(o.orderdate) = 2008     AND o.empid = 3 GROUP BY     o.orderid,o.empid HAVING     SUM(d.qty * d.unitprice) &gt; 10000;</pre>
9	<p>Bandingkan hasil pada soal-18 dengan file 85 - Lab Exercise 4 - Task 3_2 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p> 
10	<p>[Soal-19] Tuliskan perintah T-SQL dengan klausa SELECT untuk mengambil semua pelanggan yang memiliki lebih dari 25 order, dan tambahkan informasi mengenai tanggal pesanan terakhir dan jumlah penjualan. Tampilkan kolom custid dari tabel Sales.Orders table dan dua kolom perhitungan (lastorderdate berdasarkan kolom orderdate dan totalsalesamount berdasarkan kolom qty dan unitprice dari tabel Sales.OrderDetails!</p> <pre>SELECT TOP 3     o.custid,     MAX(o.orderdate) AS lastorderdate,     SUM(d.qty * d.unitprice) AS totalsalesamount FROM [Sales].[Orders] AS o JOIN [Sales].[OrderDetails] AS d     ON o.orderid = d.orderid GROUP BY     o.custid HAVING     COUNT(o.orderid) &gt; 25 ORDER BY     totalsalesamount DESC;</pre>

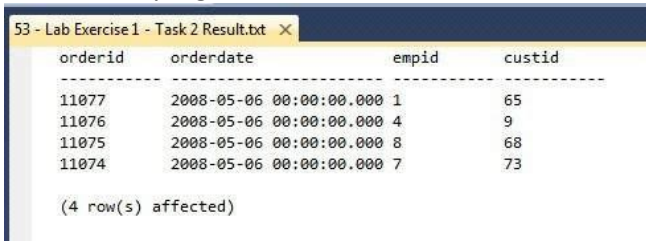
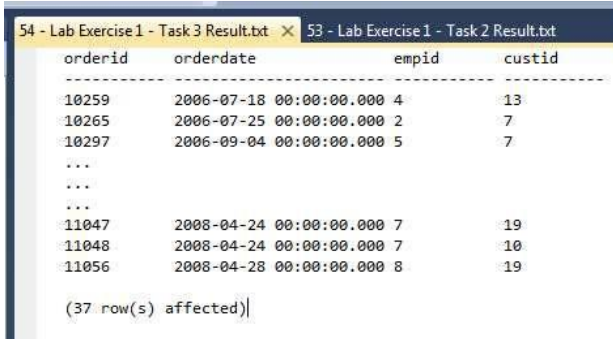




11	<p>Bandingkan hasil pada soal-19 dengan file 86 - Lab Exercise 4 - Task 4 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p> <pre>86 - Lab Exercise 4 - Task 4 Result.txt X custid      lastorderdate      totalsalesamount ----- 63          2008-04-14 00:00:00.000 117483,39 71          2008-05-01 00:00:00.000 115673,39 20          2008-05-05 00:00:00.000 113236,68  (3 row(s) affected)</pre>
12	<p><b>Kesimpulan:</b> Setelah melakukan praktikum bagian ini mahasiswa seharusnya memahami bagaimana menggunakan klausa HAVING.</p>

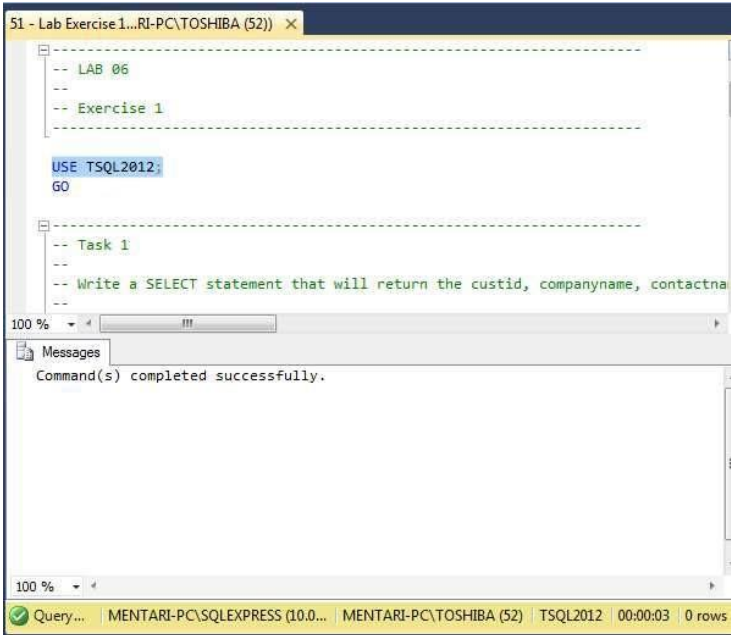
### Praktikum – Bagian 5: Menulis Query Menggunakan Self-Contained Sub-query

Langkah	Keterangan
1	<p>Skenario :</p> <p>Departemen penjualan memerlukan beberapa laporan lanjutan untuk menganalisis pesanan penjualan. Untuk itu dibutuhkan pernyataan SELECT yang menggunakan self-contained sub-query.</p> <p>Untuk melakukan percobaan pada praktikum bagian 5 ini, buka folder Minggu6/2, dan buka script 51 - Lab Exercise 1.sql. Pastikan database terhubung dengan "TSQL2012".</p>
	
2	<p><b>[Soal-20]</b> Tulislah pernyataan SELECT untuk menampilkan pemesanan orderdate maksimum dari tabel Sales.Orders.</p> <pre> SELECT     MAX([orderdate]) AS lastorderdate FROM [Sales].[Orders]; </pre>
3	<p>Bandingkan hasil pada soal-20 dengan file 52 - Lab Exercise 1 - Task 1 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p> 
4	<p><b>[Soal-21]</b> Tulislah pernyataan SELECT untuk menampilkan kolom orderid, orderdate, empid, dan custid dari tabel Sales.Orders. Kemudian saring hasilnya dengan menyertakan hanya pesanan yang sesuai dengan waktu pesan paling akhir (Gunakan query pada jawaban soal-20 sebagai sub-query self-contained subquery)!</p> <pre> SELECT     [orderid],     [orderdate],     [empid],     [custid] FROM [Sales].[Orders] WHERE     [orderdate] = (SELECT MAX([orderdate]) FROM [Sales].[Orders]); </pre>

5	<p>Bandungkan hasil pada soal-21 dengan file 53 - Lab Exercise 1 - Task 2 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p> 
6	<p><b>[Soal-22]</b> Eksekusi T-SQL di bawah ini, kemudian modifikasi dengan filter pelanggan berdasarkan contact name yang diawali dengan huruf B!</p> <pre> SELECT orderid, orderdate, empid, custid FROM Sales.Orders WHERE custid = (     SELECT custid     FROM Sales.Customers     WHERE contactname LIKE N'I%' ); FROM Sales.Customers WHERE contactname LIKE N'B%' </pre>
7	<p><b>[Soal-23]</b> Apakah terjadi error pada hasil eksekusi soal-22?kenapa?</p> <p>Jawaban: karena operator “=” hanya digunakan untuk membandingkan 1 nilai, sedangkan jika dari i% langsung diganti menjadi B% maka akan mengembalikan lebih satu nilai</p>
8	<p><b>[Soal-24]</b> Perbaiki jawaban soal-23 sehingga hasilnya tidak error!</p> <pre> SELECT orderid, orderdate, empid, custid FROM Sales.Orders WHERE custid IN (     SELECT custid     FROM Sales.Customers     WHERE contactname LIKE N'B%' ); </pre>
9	<p>Bandungkan hasil pada soal-24 dengan file 54 - Lab Exercise 1 - Task 3 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p> 

10	<p>[Soal-25] Tulislah pernyataan SELECT untuk mengambil kolom orderid dari tabel Sales.Orders dan juga kolom hasil perhitungan :</p> <ol style="list-style-type: none"> <li>1) totalsalesamount (berdasarkan kolom qty dan unitprice dari tabel Sales.OrderDetails)</li> <li>2) salespctoftotal (presentase total jumlah penjualan setiap pesanan dibagi jumlah total penjualan untuk semua pesanan dalam periode tertentu)</li> </ol> <p>Filter hasilnya hanya untuk pemesanan pada bulan mei 2008.</p> <pre> SELECT     o.orderid,     SUM(od.qty * od.unitprice) AS totalsalesamount, -- Menghitung total penjualan p     (SUM(od.qty * od.unitprice) * 100.0 /      (SELECT SUM(od2.qty * od2.unitprice)       FROM Sales.Orders AS o2       JOIN Sales.OrderDetails AS od2 ON o2.orderid = od2.orderid       WHERE o2.orderdate &gt;= '2008-05-01' AND o2.orderdate &lt; '2008-06-01'))     ) AS salespctoftotal -- Menghitung persentase penjualan per pesanan terhadap to FROM     Sales.Orders AS o JOIN     Sales.OrderDetails AS od ON o.orderid = od.orderid -- Menggabungkan tabel Order WHERE     o.orderdate &gt;= '2008-05-01' AND o.orderdate &lt; '2008-06-01' -- Memfilter untuk b GROUP BY     o.orderid; </pre>
11	<p>Bandingkan hasil pada soal-25 dengan file 55 - Lab Exercise 1 - Task 4 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p> <pre> 55 - Lab Exercise 1 - Task 4 Result.txt orderid    totalsalesamount    salespctoftotal ----- 11064      4722,30             23.7300 11065      252,56              1.2600 11066      928,75              4.6600 ... ... 11075      586,00              2.9400 11076      1057,00             5.3100 11077      1374,60             6.9000 (14 row(s) affected) </pre>
12	<p><b>Kesimpulan:</b> Setelah mengerjakan praktikum bagian ini mahasiswa seharusnya bisa menggunakan self-contained sub-query pada pernyataan T-SQL.</p>

## Praktikum – Bagian 6: Menulis Query Yang Menggunakan Sub-Query Skalar Dan Multi Nilai

Langkah	Keterangan
1	<p>Skenario :</p> <p>Departemen pemasaran ingin menyiapkan bahan untuk berbagai kelompok produk dan pelanggan berdasarkan riwayat informasi penjualan. Untuk itu diperlukan pernyataan SELECT menggunakan Sub-Query dalam klausa WHERE.</p> <p>Untuk melakukan percobaan pada praktikum bagian 6 ini, buka folder Minggu6/2, dan buka script 61 - Lab Exercise 2.sql. Pastikan database terhubung dengan “TSQL2012”.</p>
	
2	<p><b>[Soal-26]</b> Tulislah pernyataan SELECT untuk mengambil kolom productid dan productname dari tabel Production.Products. Kemudian filter hasilnya untuk menampilkan produk yang terjual dalam jumlah yang banyak (lebih dari 100 produk) untuk baris pesana tertentu!</p> <pre> SELECT     p.[productid],     p.[productname] FROM [Production].[Products] AS p JOIN [Sales].[OrderDetails] AS d     ON p.productid = d.productid WHERE     d.qty &gt; 100 GROUP BY     p.[productid],     p.[productname]; </pre>



3	<p>Bandungkan hasil pada soal-26 dengan file 62 - Lab Exercise 2 - Task 1 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p> <pre>62 - Lab Exercise 2 - Task 1 Result.txt X productid  productname ----- 24         Product QOGNU 27         Product SMIOH 39         Product LSOFL ... ... 61         Product XYZPE 64         Product HCQDE 75         Product BWRLG  (12 row(s) affected)</pre>
4	<p>[Soal-27] Tulislah pernyataan SELECT untuk mengambil kolom custid dan contactname dari tabel Sales.Customers. Kemudian lakukan filter hanya untuk pelanggan/customer yang tidak memiliki pesanan apapun!</p> <pre>SELECT     c.custid,     c.contactname FROM     Sales.Customers AS c LEFT JOIN     Sales.Orders AS o ON c.custid = o.custid WHERE     o.orderid IS NULL;</pre>
5	<p>Bandungkan hasil pada soal-27 dengan file 63 - Lab Exercise 2 - Task 2 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p> <pre>63 - Lab Exercise 2 - Task 2 Result.txt X custid      contactname ----- 22          Bueno, Janaina Burdan, Neville 57          Tollefsen, Bjørn  (2 row(s) affected)</pre>
6	<p>[Soal-28] Terdapat tambahan satu baris data pada tabel Sales.Orders dengan T-SQL sebagai berikut :</p> <pre>INSERT INTO Sales.Orders (     custid, empid, orderdate, requireddate, shippeddate, shipperid, freight,     shipname, shipaddress, shipcity, shipregion, shippostalcode, shipcountry) VALUES     (NULL, 1, '20111231', '20111231', '20111231', 1, 0,     'ShipOne', 'ShipAddress', 'ShipCity', 'RA', '1000', 'USA');</pre> <p>Eksekusi perintah tersebut! Kemudian salin jawaban pada soal-27. Bagaimana perbandingan hasil kedua T-SQL tersebut? Kenapa?</p> <p>Jawaban: hasil tetap sama, karena orderid tidak null</p>
7	<p>[Soal-29] Modifikasi jawaban soal-27 (cara yang berbeda dengan output yang sama), dengan cara menghapus baris dengan nilai yang tidak diketahui pada kolom custid!</p>





	<pre>DELETE FROM [Sales].[Orders] WHERE [custid] IS NULL;  SELECT     c.custid,     c.contactname FROM     Sales.Customers AS c LEFT JOIN     Sales.Orders AS o ON c.custid = o.custid WHERE     o.orderid IS NULL;  </pre>
8	<p>Bandingkan hasil pada soal-29 dengan file 64 - Lab Exercise 2 - Task 3 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p> <pre>64 - Lab Exercise 2 - Task 3 Result.txt X custid      contactname ----- 22          Bueno, Janaina Burdan, Neville 57          Tollefsen, Bjørn  (2 row(s) affected)</pre>
9	<p><b>Kesimpulan:</b> Setelah mengerjakan praktikum dan menjawab soal-soal pada bagian ini mahasiswa seharusnya memahami cara menggunakan multi-hasil dalam pernyataan T-SQL.</p>

## **Praktikum – Bagian 7: Menulis Query Yang Menggunakan Sub-Query Yang Berkorelasi Dan Predikat EXISTS**

Langkah	Keterangan
1	<p>Skenario :</p> <p>Departemen penjualan ingin memiliki beberapa laporan tambahan untuk menampilkan berbagai analisis untuk para pelanggan. Karena permintaan departemen penjualan ini rumit, maka diperlukan penggunaan Sub-Query yang berkorelasi.</p> <p>Untuk melakukan percobaan pada praktikum bagian 7 ini, buka folder Minggu6/2, dan buka script 71 - Lab Exercise 3.sql. Pastikan database terhubung dengan “TSQL2012”.</p>
	
2	<p><b>[Soal-30]</b> Tulislah pernyataan SELECT untuk mengambil kolom custid dan contactname dari tabel Sales.Customers. Tambahkan kolom lastorderdate yang berisi tanggal terakhir dari tabel Sales.Orders untuk setiap pelanggan (Gunakan sub-query yang berkorelasi).</p> <pre> SELECT     c.custid,     c.contactname,     (         SELECT MAX(o.[orderdate])         FROM [Sales].[Orders] AS o     ) AS lastorderdate FROM [Sales].[Customers] AS c; </pre>

3

4

5

6

Bandingkan hasil pada soal-30 dengan file 72 - Lab Exercise 3 - Task 1 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.

72 - Lab Exercise 3 - Task 1 Result.txt

custid	contactname	lastorderdate
1	Allen, Michael	2008-04-09 00:00:00.000
2	Hassall, Mark	2008-03-04 00:00:00.000
3	Peoples, John	2008-01-28 00:00:00.000
...		
...		
...		
89	Smith Jr., Ronaldo	2008-05-01 00:00:00.000
90	Larsson, Katarina	2008-04-07 00:00:00.000
91	Conn, Steve	2008-04-23 00:00:00.000

(91 row(s) affected)

[Soal-31] Tuliskan pernyataan SELECT untuk mengambil semua pelanggan yang tidak memiliki pesanan di tabel Sales.Orders. Gunakan predikat EXISTS untuk melakukan filter yang mencakup pelanggan yang tidak memiliki pesanan! (Tidak diperlukan pemeriksaan eksplisit kolom custid dari tabel Sales.Orders table berstatus not NULL)

```
SELECT
    c.custid,
    c.contactname
FROM [Sales].[Customers] AS c
WHERE
    NOT EXISTS(
        SELECT 1
        FROM [Sales].[Orders] AS o
        WHERE o.custid = c.custid
    );
```

Bandingkan hasil pada soal-31 dengan file 73 - Lab Exercise 3 - Task 2 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.

73 - Lab Exercise 3 - Task 2 Result.txt

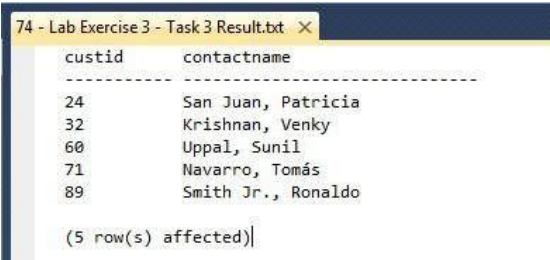
custid	contactname
22	Bueno, Janaina Burdan, Neville
57	Tollefsen, Bjørn

(2 row(s) affected)

[Soal-32] Tuliskan pernyataan SELECT untuk mengambil kolom custid dan contactname dari tabel Sales.Customers. Kemudian filter hasilnya hanya untuk pelanggan yang memesan pada atau setelah 1 april 2008, dan melakukan pemesanan dengan harga tinggi di atas \$100!

```
SELECT
    c.custid,
    c.contactname
FROM [Sales].[Customers] AS c
JOIN [Sales].[Orders] AS o ON c.custid = o.custid
JOIN [Sales].[OrderDetails] AS d ON o.orderid = d.orderid
WHERE o.orderdate >= '2008-04-01'
GROUP BY
    c.custid, c.contactname
HAVING
    SUM(d.qty * d.unitprice) > 100
ORDER BY c.custid;
```



7	<p>Bandungkan hasil pada soal-32 dengan file 74 - Lab Exercise 3 - Task 3 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p> 
8	<p>[Soal-33] Tulislah pernyataan SELECT yang akan mengambil informasi setiap tahun sebagai berikut :</p> <ol style="list-style-type: none"><li>1) Tahun pesanan</li><li>2) Jumlah total penjualan</li><li>3) Jumlah total penjualan yang terjual selama bertahun-tahun (setiap tahun dikembalikan jumlah total penjualan sampai tahun tertentu, misalkan awal tahun 2006 mengembalikan jumlah total penjualan untuk tahun selanjutnya 2007)</li><li>4) Pernyataan SELECT harus memiliki 3 kolom :<ul style="list-style-type: none"><li>• orderyear, berasal dari kolom orderyear dari tabel Sales.Orders</li><li>• totalsales, berasal dari kolom qty dan unitprice dari tabel Sales.OrderDetails</li><li>• runsales, mewakili jumlah penjualan yang sedang terjadi. Kolom ini menggunakan sub-query yang berkorelasi</li></ul></li></ol> <pre>SELECT YEAR(o.orderDate) AS orderyear, SUM(d.qty * d.unitPrice) AS totalsales, (     SELECT SUM(d.qty * d.unitPrice)     FROM Sales.Orders AS o     JOIN Sales.OrderDetails AS d ON o.orderId = d.orderId     WHERE YEAR(o.orderDate) &lt;= YEAR(o.orderDate) ) AS runsales FROM Sales.Orders o JOIN Sales.OrderDetails d ON o.orderId = d.orderId GROUP BY YEAR(o.orderDate) ORDER BY orderyear;</pre>
9	<p>Bandungkan hasil pada soal-33 dengan file 75 - Lab Exercise 3 - Task 4 Result.txt. Jika sama, maka T-SQL yang Anda tuliskan sudah benar.</p>



	<div>75 - Lab Exercise 3 - Task 4 Result.txt X</div> <table><tr><th>orderyear</th><th>totalsales</th><th>runsales</th></tr><tr><td>2006</td><td>226298,50</td><td>226298,50</td></tr><tr><td>2007</td><td>658388,75</td><td>884687,25</td></tr><tr><td>2008</td><td>469771,34</td><td>1354458,59</td></tr></table> <p>(3 row(s) affected)</p>	orderyear	totalsales	runsales	2006	226298,50	226298,50	2007	658388,75	884687,25	2008	469771,34	1354458,59
orderyear	totalsales	runsales											
2006	226298,50	226298,50											
2007	658388,75	884687,25											
2008	469771,34	1354458,59											
10	<p><b>Kesimpulan:</b> Setelah mengerjakan praktikum dan soal pada bagian ini, seharusnya sekarang Anda paham bagaimana menggunakan Sub-Query yang berkorelasi di SQL.</p>												

--- Selamat Mengerjakan ---