

Name : Evan Diantha Fafian  
Class : SIB 2G  
Absent : 09  
NIM : 2341760163

# **JOBSHEET**

## **PRAKTIKUM BASIS DATA LANJUT**

**Jurusan Teknologi Informasi  
POLITEKNIK NEGERI MALANG  
2024**



**Week 2**

**SQL SERVER - SELECT, JOIN SORTING DAN FILTERING DATA**

**Team Teaching:**

Habibie Ed Dien, S.Kom., M.T.

Irsyad Arif Mashudi, S.Kom M.Kom

Vit Zuraida, S.Kom., M.Kom.

Rokhimatul Wakhidah, S.Pd., M.T.

Annisa Taufika Firdausi, ST., MT.

Elok Nur Hamdana, S.T., M.T



Information Technology Department, Malang State Polytechnic

## **Jobsheet- 1 : Introduction to Transact-SQL and Statements**

### **SELECT, Join, Sorting, and Filtering data**

### **Advanced Database Course**

**Supervisor:** Advanced Database Teaching Team

*September 2024*

#### **Topics**

1. Introduction to T-SQL and *Query Select*
2. Querying Multiple Tables
3. Sorting and Filtering Data

#### **Objective**

Students are expected to be able to:

1. Understanding the basic differences between Transact-SQL (T-SQL) and ANSI SQL.
2. Understanding how to create *a database* from an existing SQL file
3. Understand how to execute part or all of a SQL *script* from an existing file.
4. Understanding the concept of using '*comments*' in T-SQL.
5. Understand the concept of using the SELECT statement to analyze existing tables in *a database* .
6. Understanding how to display data in a *unique / distinct manner* .
7. Understand how to use *ALIAS* for table names and column names.
8. Understand the concept of *CASE* expressions and how to use them.
9. Students understand how to query multiple tables in a SELECT clause using JOIN.
10. Students understand how to write INNER JOIN , OUTER JOIN , SELF-JOIN and CROSS JOIN queries
- .
11. Students understand how to do Data Sorting , Data Filtering with predicates , Data Filtering with TOP and OFFSET-FETCH
12. Students understand how to handle missing and unknown values in real data.

#### **General Instructions**

1. Follow the steps in the practical sections in the order given.
2. Answer all questions marked **[Question-X]** that are found in certain steps in each part of the practicum.
3. In each step of the practicum, there is an explanation that will help you answer the questions in instruction number 3, so read and do all the practicum parts in this jobsheet.
4. Write the answers to the questions in the instructions number 3 in a report that is done using a word processing application (Word, OpenOffice, or other similar). Export as a **PDF file** with the following name format:
  - **BDL\_Class\_03\_YourFullName .pdf**
  - Collect the PDF files as a practical report to the supervising lecturer.
  - In addition to the file name, also include your identity on the first page of the report.



## Practical Preparation: Creating a Database from Existing SQL

| Step | Information   |
|------|---|
| 1    | Create a TSQL database<br>  |
| 2    | On the <b>File menu</b> , click <b>Open</b> and click <b>Project/Solution</b> .<br>   |
| 3    | In the <b>Open Project window</b> , open the given project file.<br>  |
| 4    | Next, the Solution Explorer window will display the following display. Then please open the "Setup" file. This file contains the sql script to create the tables needed for this practicum.<br> |
| 5    | After the setup file is opened, a display like the image below will appear. Then click <i>Execute</i> and please wait until the process is complete.  |

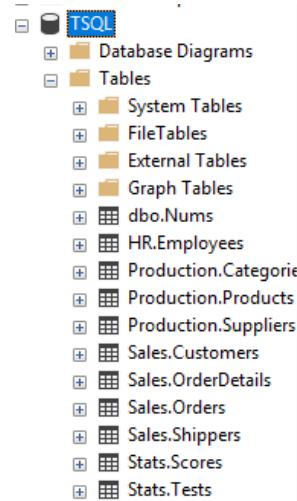


```

File Edit View Project Debug Tools Window Help
New Query Execute Debug Default Default Generic Debugger
Object Explorer Solution Explorer
Connect PUSPA\SQLEXPRESS (SQL Server 14.0.2)
Databases Solutions
System Databases Queries
Database Snapshots
TSQL
Security
Server Objects Miscellaneous
Replication
PolyBase
Management
XEvent Profiler
setup.sql - PUSPA\ASUS(52) - X
setup.sql - PUSPA\ASUS(52) - X
SET NOCOUNT ON;
GO
-- Insert data into Stats.Scores
INSERT INTO Stats.Scores(testid, studentid, score) VALUES
('Test ABC', 'Student A', 95);
INSERT INTO Stats.Scores(testid, studentid, score) VALUES
('Test ABC', 'Student B', 65);
INSERT INTO Stats.Scores(testid, studentid, score) VALUES
('Test ABC', 'Student C', 75);
INSERT INTO Stats.Scores(testid, studentid, score) VALUES
('Test XYZ', 'Student A', 95);
INSERT INTO Stats.Scores(testid, studentid, score) VALUES
('Test XYZ', 'Student B', 80);
INSERT INTO Stats.Scores(testid, studentid, score) VALUES
('Test XYZ', 'Student C', 55);
INSERT INTO Stats.Scores(testid, studentid, score) VALUES
('Test XYZ', 'Student D', 55);
INSERT INTO Stats.Scores(testid, studentid, score) VALUES
('Test XYZ', 'Student E', 50);
INSERT INTO Stats.Scores(testid, studentid, score) VALUES
('Test XYZ', 'Student F', 80);
GO
100 % 4
Messages Properties
Commands completed successfully.

```

After the process is successful, several tables will be formed, as shown in the image below.



6

For example, to check *records* in the Sales.Customers table, please execute the command below:

7

```

USE [TSQL]
GO

SELECT [custid],
       [companyname],
       [contactname],
       [contacttitle],
       [address],
       [city],
       [region],
       [postalcode],
       [country],
       [phone],
       [fax]
  FROM [Sales].[Customers]
GO

```

8

The results of the SQL command above are as follows

|   | custid | companyname    | contactname       | contacttitle         | address                       | city        | region |
|---|--------|----------------|-------------------|----------------------|-------------------------------|-------------|--------|
| 1 | 1      | Customer NRZBB | Allen, Michael    | Sales Representative | Obere Str. 0123               | Berlin      | NULL   |
| 2 | 2      | Customer MLTDN | Hassall, Mark     | Owner                | Avda. de la Constitución 5678 | México D.F. | NULL   |
| 3 | 3      | Customer KBUDE | Peoples, John     | Owner                | Mataderos 7890                | México D.F. | NULL   |
| 4 | 4      | Customer HFBZG | Amdt, Torsten     | Sales Representative | 7890 Hanover Sq.              | London      | NULL   |
| 5 | 5      | Customer HGVLZ | Higginbotham, Tom | Order Administrator  | Berguvsvägen 5678             | Luleå       | NULL   |

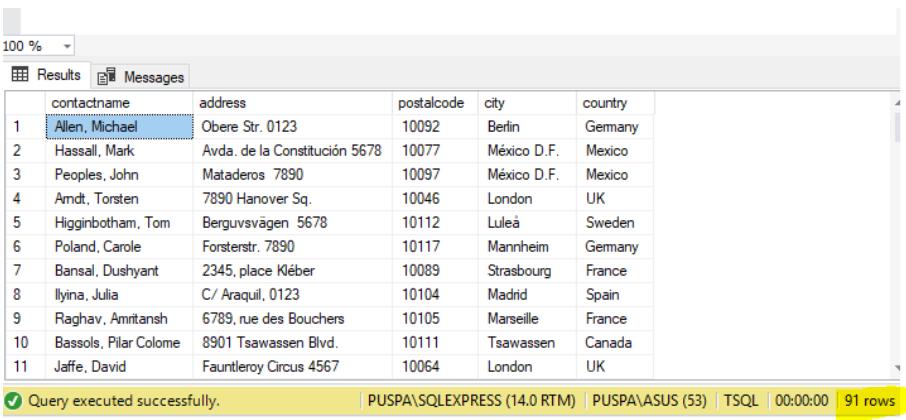
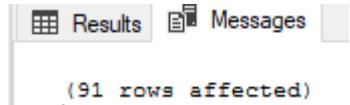


## Practical – Part 1: Executing part or all of a SQL script

| Step | Information  |
|------|--|
| 1    | <p>Please type the following <i>query</i> in your <i>query panel</i> then click <i>execute</i>. Note the results displayed.</p> <pre>SELECT * FROM Sales.Customers;</pre>  |
| 2    | <p>Please add the following <i>query</i> to your <i>query panel</i> then click <i>execute</i>. Note the results displayed</p> <pre>SELECT * FROM Sales.Customers;  SELECT custid, companyname, contactname, contacttitle, address, city, region, postalcode, country, phone, fax FROM Sales.Customers;</pre>   |
| 3    | <p>Make a selection on one of the existing <i>queries</i> then click <i>execute</i>. Note the results displayed. <b>What is the difference with the results in the second step above? (Question 1)</b></p> <p>My Answer :</p> <ul style="list-style-type: none"><li>- it's the same, because * means to select all columns in the Sales.Customers table</li></ul> <pre>SELECT * FROM Sales.Customers;  SELECT custid, companyname, contactname, contacttitle, address, city, region, postalcode, country, phone, fax FROM Sales.Customers;</pre> |
| 4    | <p>In the <i>query panel</i> please type</p> <pre>SELECT * FROM</pre>  |
| 5    | <p>then on the Object Explorer tab – Tables please find the Sales.Customers table. Click the table and drag it to the query pane l . The result is as shown below, after that add a semicolon after the name of the table in question and click execute.</p> <pre>SELECT * FROM [Sales].[Customers];</pre>   |



## Practical – Part 2: Using the SELECT statement for specific columns

| Step | Information   |                               |             |             |            |      |         |   |                |                 |       |        |         |   |               |                               |       |             |        |   |               |                |       |             |        |   |               |                  |       |        |    |   |                  |                   |       |       |        |   |                |                  |       |          |         |   |                  |                    |       |            |        |   |               |                  |       |        |       |   |                  |                        |       |           |        |    |                       |                      |       |           |        |    |              |                        |       |        |    |
|------|---|-------------------------------|-------------|-------------|------------|------|---------|---|----------------|-----------------|-------|--------|---------|---|---------------|-------------------------------|-------|-------------|--------|---|---------------|----------------|-------|-------------|--------|---|---------------|------------------|-------|--------|----|---|------------------|-------------------|-------|-------|--------|---|----------------|------------------|-------|----------|---------|---|------------------|--------------------|-------|------------|--------|---|---------------|------------------|-------|--------|-------|---|------------------|------------------------|-------|-----------|--------|----|-----------------------|----------------------|-------|-----------|--------|----|--------------|------------------------|-------|--------|----|
| 1    | In the query panel, please type the script below<br><pre>SELECT     contactname, address, postalcode, city, country FROM Sales.Customers;</pre>   |                               |             |             |            |      |         |   |                |                 |       |        |         |   |               |                               |       |             |        |   |               |                |       |             |        |   |               |                  |       |        |    |   |                  |                   |       |       |        |   |                |                  |       |          |         |   |                  |                    |       |            |        |   |               |                  |       |        |       |   |                  |                        |       |           |        |    |                       |                      |       |           |        |    |              |                        |       |        |    |
| 2    | <i>Highlights query above and click execute</i>   |                               |             |             |            |      |         |   |                |                 |       |        |         |   |               |                               |       |             |        |   |               |                |       |             |        |   |               |                  |       |        |    |   |                  |                   |       |       |        |   |                |                  |       |          |         |   |                  |                    |       |            |        |   |               |                  |       |        |       |   |                  |                        |       |           |        |    |                       |                      |       |           |        |    |              |                        |       |        |    |
| 3    | Please observe the results. How many <i>rows</i> are produced? To find out, you can do it on the results tab as shown in the image below<br> <p>The screenshot shows the SSMS interface with the 'Results' tab selected. A table is displayed with the following data:</p> <table border="1"><thead><tr><th></th><th>contactname</th><th>address</th><th>postalcode</th><th>city</th><th>country</th></tr></thead><tbody><tr><td>1</td><td>Allen, Michael</td><td>Obere Str. 0123</td><td>10092</td><td>Berlin</td><td>Germany</td></tr><tr><td>2</td><td>Hassall, Mark</td><td>Avda. de la Constitución 5678</td><td>10077</td><td>México D.F.</td><td>Mexico</td></tr><tr><td>3</td><td>Peoples, John</td><td>Mataderos 7890</td><td>10097</td><td>México D.F.</td><td>Mexico</td></tr><tr><td>4</td><td>Amdt, Torsten</td><td>7890 Hanover Sq.</td><td>10046</td><td>London</td><td>UK</td></tr><tr><td>5</td><td>Higinbotham, Tom</td><td>Berguvsvägen 5678</td><td>10112</td><td>Luleå</td><td>Sweden</td></tr><tr><td>6</td><td>Poland, Carole</td><td>Forsterstr. 7890</td><td>10117</td><td>Mannheim</td><td>Germany</td></tr><tr><td>7</td><td>Bansal, Dushyant</td><td>2345, place Kléber</td><td>10089</td><td>Strasbourg</td><td>France</td></tr><tr><td>8</td><td>Ilyina, Julia</td><td>C/ Araquil, 0123</td><td>10104</td><td>Madrid</td><td>Spain</td></tr><tr><td>9</td><td>Raghav, Amitansh</td><td>6789, rue des Bouchers</td><td>10105</td><td>Marseille</td><td>France</td></tr><tr><td>10</td><td>Bassols, Pilar Colome</td><td>8901 Tsawassen Blvd.</td><td>10111</td><td>Tsawassen</td><td>Canada</td></tr><tr><td>11</td><td>Jaffe, David</td><td>Fauntleroy Circus 4567</td><td>10064</td><td>London</td><td>UK</td></tr></tbody></table> <p>At the bottom of the results window, a status bar indicates: <b>Query executed successfully.</b>   PUSPA\SQLEXPRESS (14.0 RTM)   PUSPA\ASUS (53)   TSQL   00:00:00   91 rows</p> <p>Or you can also go to the messages tab as shown in the image below.</p>  <p>The screenshot shows the SSMS interface with the 'Messages' tab selected. The message '(91 rows affected)' is displayed.</p> |                               | contactname | address     | postalcode | city | country | 1 | Allen, Michael | Obere Str. 0123 | 10092 | Berlin | Germany | 2 | Hassall, Mark | Avda. de la Constitución 5678 | 10077 | México D.F. | Mexico | 3 | Peoples, John | Mataderos 7890 | 10097 | México D.F. | Mexico | 4 | Amdt, Torsten | 7890 Hanover Sq. | 10046 | London | UK | 5 | Higinbotham, Tom | Berguvsvägen 5678 | 10112 | Luleå | Sweden | 6 | Poland, Carole | Forsterstr. 7890 | 10117 | Mannheim | Germany | 7 | Bansal, Dushyant | 2345, place Kléber | 10089 | Strasbourg | France | 8 | Ilyina, Julia | C/ Araquil, 0123 | 10104 | Madrid | Spain | 9 | Raghav, Amitansh | 6789, rue des Bouchers | 10105 | Marseille | France | 10 | Bassols, Pilar Colome | 8901 Tsawassen Blvd. | 10111 | Tsawassen | Canada | 11 | Jaffe, David | Fauntleroy Circus 4567 | 10064 | London | UK |
|      | contactname   | address                       | postalcode  | city        | country    |      |         |   |                |                 |       |        |         |   |               |                               |       |             |        |   |               |                |       |             |        |   |               |                  |       |        |    |   |                  |                   |       |       |        |   |                |                  |       |          |         |   |                  |                    |       |            |        |   |               |                  |       |        |       |   |                  |                        |       |           |        |    |                       |                      |       |           |        |    |              |                        |       |        |    |
| 1    | Allen, Michael  | Obere Str. 0123               | 10092       | Berlin      | Germany    |      |         |   |                |                 |       |        |         |   |               |                               |       |             |        |   |               |                |       |             |        |   |               |                  |       |        |    |   |                  |                   |       |       |        |   |                |                  |       |          |         |   |                  |                    |       |            |        |   |               |                  |       |        |       |   |                  |                        |       |           |        |    |                       |                      |       |           |        |    |              |                        |       |        |    |
| 2    | Hassall, Mark   | Avda. de la Constitución 5678 | 10077       | México D.F. | Mexico     |      |         |   |                |                 |       |        |         |   |               |                               |       |             |        |   |               |                |       |             |        |   |               |                  |       |        |    |   |                  |                   |       |       |        |   |                |                  |       |          |         |   |                  |                    |       |            |        |   |               |                  |       |        |       |   |                  |                        |       |           |        |    |                       |                      |       |           |        |    |              |                        |       |        |    |
| 3    | Peoples, John   | Mataderos 7890                | 10097       | México D.F. | Mexico     |      |         |   |                |                 |       |        |         |   |               |                               |       |             |        |   |               |                |       |             |        |   |               |                  |       |        |    |   |                  |                   |       |       |        |   |                |                  |       |          |         |   |                  |                    |       |            |        |   |               |                  |       |        |       |   |                  |                        |       |           |        |    |                       |                      |       |           |        |    |              |                        |       |        |    |
| 4    | Amdt, Torsten   | 7890 Hanover Sq.              | 10046       | London      | UK         |      |         |   |                |                 |       |        |         |   |               |                               |       |             |        |   |               |                |       |             |        |   |               |                  |       |        |    |   |                  |                   |       |       |        |   |                |                  |       |          |         |   |                  |                    |       |            |        |   |               |                  |       |        |       |   |                  |                        |       |           |        |    |                       |                      |       |           |        |    |              |                        |       |        |    |
| 5    | Higinbotham, Tom  | Berguvsvägen 5678             | 10112       | Luleå       | Sweden     |      |         |   |                |                 |       |        |         |   |               |                               |       |             |        |   |               |                |       |             |        |   |               |                  |       |        |    |   |                  |                   |       |       |        |   |                |                  |       |          |         |   |                  |                    |       |            |        |   |               |                  |       |        |       |   |                  |                        |       |           |        |    |                       |                      |       |           |        |    |              |                        |       |        |    |
| 6    | Poland, Carole  | Forsterstr. 7890              | 10117       | Mannheim    | Germany    |      |         |   |                |                 |       |        |         |   |               |                               |       |             |        |   |               |                |       |             |        |   |               |                  |       |        |    |   |                  |                   |       |       |        |   |                |                  |       |          |         |   |                  |                    |       |            |        |   |               |                  |       |        |       |   |                  |                        |       |           |        |    |                       |                      |       |           |        |    |              |                        |       |        |    |
| 7    | Bansal, Dushyant  | 2345, place Kléber            | 10089       | Strasbourg  | France     |      |         |   |                |                 |       |        |         |   |               |                               |       |             |        |   |               |                |       |             |        |   |               |                  |       |        |    |   |                  |                   |       |       |        |   |                |                  |       |          |         |   |                  |                    |       |            |        |   |               |                  |       |        |       |   |                  |                        |       |           |        |    |                       |                      |       |           |        |    |              |                        |       |        |    |
| 8    | Ilyina, Julia   | C/ Araquil, 0123              | 10104       | Madrid      | Spain      |      |         |   |                |                 |       |        |         |   |               |                               |       |             |        |   |               |                |       |             |        |   |               |                  |       |        |    |   |                  |                   |       |       |        |   |                |                  |       |          |         |   |                  |                    |       |            |        |   |               |                  |       |        |       |   |                  |                        |       |           |        |    |                       |                      |       |           |        |    |              |                        |       |        |    |
| 9    | Raghav, Amitansh  | 6789, rue des Bouchers        | 10105       | Marseille   | France     |      |         |   |                |                 |       |        |         |   |               |                               |       |             |        |   |               |                |       |             |        |   |               |                  |       |        |    |   |                  |                   |       |       |        |   |                |                  |       |          |         |   |                  |                    |       |            |        |   |               |                  |       |        |       |   |                  |                        |       |           |        |    |                       |                      |       |           |        |    |              |                        |       |        |    |
| 10   | Bassols, Pilar Colome   | 8901 Tsawassen Blvd.          | 10111       | Tsawassen   | Canada     |      |         |   |                |                 |       |        |         |   |               |                               |       |             |        |   |               |                |       |             |        |   |               |                  |       |        |    |   |                  |                   |       |       |        |   |                |                  |       |          |         |   |                  |                    |       |            |        |   |               |                  |       |        |       |   |                  |                        |       |           |        |    |                       |                      |       |           |        |    |              |                        |       |        |    |
| 11   | Jaffe, David  | Fauntleroy Circus 4567        | 10064       | London      | UK         |      |         |   |                |                 |       |        |         |   |               |                               |       |             |        |   |               |                |       |             |        |   |               |                  |       |        |    |   |                  |                   |       |       |        |   |                |                  |       |          |         |   |                  |                    |       |            |        |   |               |                  |       |        |       |   |                  |                        |       |           |        |    |                       |                      |       |           |        |    |              |                        |       |        |    |

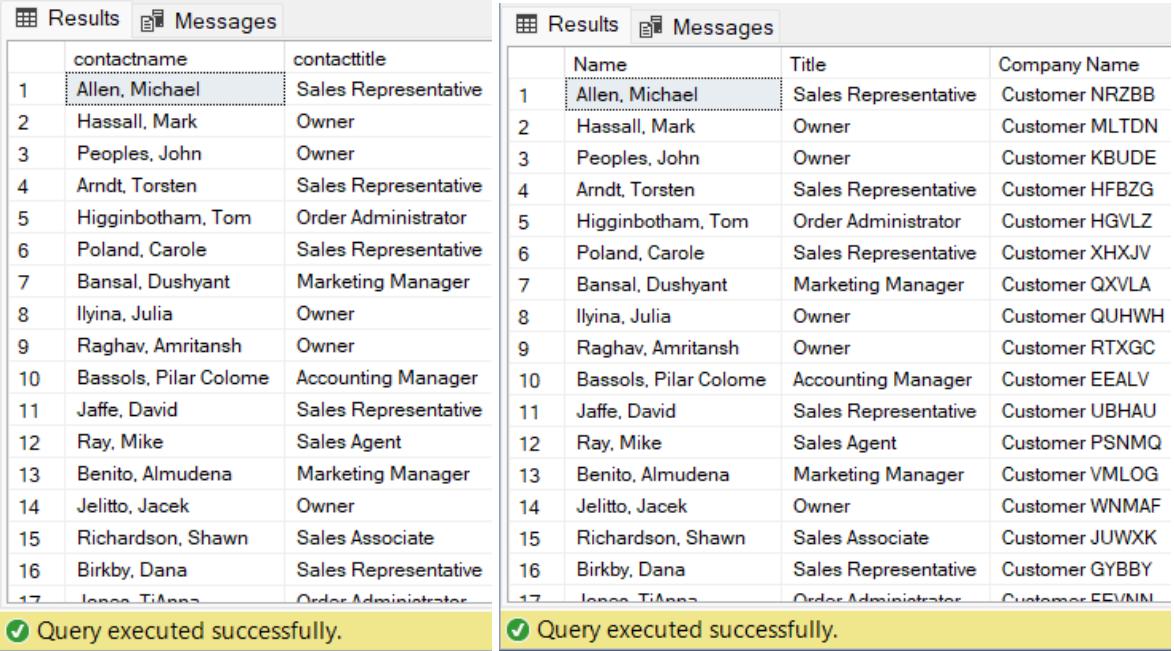


## Practical – Part 3: Using the SELECT statement to display data *uniquely / DISTINCT*

| Step | Information   |
|------|---|
| 1    | In the query panel, please type the script below<br><pre>SELECT     country FROM Sales.Customers;</pre>   |
| 2    | Highlights query above and click execute  |
| 3    | Please observe the results. <b>Is there any duplicate data? If YES, why? Capture the results of executing the SQL script above (Question 2)</b><br>My Answer :<br>- Yes, because select takes all rows in a specified column in the table.  |
| 4    | In the query pane, please type the script below.<br><pre>SELECT DISTINCT     country FROM Sales.Customers;</pre><br>Please click execute and observe the results.   |
| 5    | <b>Is there any duplicate data? Explain the difference in results in step 4 and step 3!? What are the benefits of the DISTINCT command? Capture the results of executing the SQL script above (Question 3)</b><br>My Answer :<br>- No, because distinct will remove duplicates and return only unique rows from the column. |



## Practical – Part 4: Using *ALIAS* for table names and column names

| Step | Information  |
|------|--|
| 1    | In the query panel, please type the script below<br><pre>SELECT     c.contactname, c.contacttitle FROM Sales.Customers AS c;</pre>   |
| 2    | Highlights query above and click execute. Observe the results  |
| 3    | In the query panel, please type the script below.<br><pre>SELECT     c.contactname AS Name, c.contacttitle AS Title, c.companyname AS [Company Name] FROM Sales.Customers AS c;</pre>  |
| 4    | Highlights query above and click execute . Observe the results.  |
| 5    | <b>What is the difference between the execution results of the query stage 1 and stage 3 above? What are the benefits of the AS command? Please explain! Capture the results of the SQL script execution above (Question 4)</b><br><br> <p>The screenshot shows two separate SQL queries run in the SSMS 'Results' tab. Both queries return 17 rows of data from the 'Customers' table, with columns 'contactname', 'contacttitle', and 'companyname'. The first query uses column aliases 'Name', 'Title', and 'Company Name'. The second query uses the original column names 'contactname', 'contacttitle', and 'companyname'. Both queries end with a message: 'Query executed successfully.'</p> <p><b>My Answer :</b></p> <ul style="list-style-type: none"><li>- The difference is in the use of AS during column selection, the function of AS is to shorten and rename the column only temporarily when the query is run.</li></ul> |



## Practicum – Part 5: Use of CASE

| Step | Information  |
|------|--|
| 1    | In the query panel, please type the script below<br><pre>SELECT     p.categoryid, p.productname FROM Production.Products AS p;</pre>   |
| 2    | Highlights query above and click execute. Observe the results  |
| 3    | In the query panel, please type the script below.<br><pre>SELECT     p.categoryid, p.productname,     CASE         WHEN p.categoryid = 1 THEN 'Beverages'         WHEN p.categoryid = 2 THEN 'Condiments'         WHEN p.categoryid = 3 THEN 'Confections'         WHEN p.categoryid = 4 THEN 'Dairy Products'         WHEN p.categoryid = 5 THEN 'Grains/Cereals'         WHEN p.categoryid = 6 THEN 'Meat/Poultry'         WHEN p.categoryid = 7 THEN 'Produce'         WHEN p.categoryid = 8 THEN 'Seafood'         ELSE 'Other'     END AS categoryname FROM Production.Products AS p;</pre> |
| 4    | Highlights query above and click execute . Observe the results.  |
| 5    | <b>What is the difference between the execution results of the query stage 1 and stage 3 above? What are the benefits of the CASE command? Please explain! Capture the results of the SQL script execution above (Question 5)</b>  |



|    | Results    |               |                | Messages |
|----|------------|---------------|----------------|----------|
|    | categoryid | productname   | categoryname   |          |
| 1  | 1          | Product HHYDP | Beverages      |          |
| 2  | 1          | Product RECZE | Beverages      |          |
| 3  | 2          | Product IMEHJ | Condiments     |          |
| 4  | 2          | Product KSBRM | Condiments     |          |
| 5  | 2          | Product EPEIM | Condiments     |          |
| 6  | 2          | Product VAIIV | Condiments     |          |
| 7  | 7          | Product HMLNI | Produce        |          |
| 8  | 2          | Product WVJFP | Condiments     |          |
| 9  | 6          | Product AOZBW | Meat/Poultry   |          |
| 10 | 8          | Product YHXGE | Seafood        |          |
| 11 | 4          | Product QMVUN | Dairy Products |          |
| 12 | 4          | Product OSFNS | Dairy Products |          |
| 13 | 8          | Product POXFU | Seafood        |          |
| 14 | 7          | Product PWCJB | Produce        |          |
| 15 | 2          | Product KSZOI | Condiments     |          |
| 16 | 3          | Product PAFRH | Confections    |          |
| 17 | 6          | Product PLQAY | Meat/Poultry   |          |

Query executed successfully.

My Answer :

- The CASE expression goes through conditions and returns a value when the first condition is met (like an if-then-else statement). So, once a condition is true, it will stop reading and return the result. If no conditions are true, it returns the value in the ELSE clause.

| 6 | In the query panel, please type the script below. |



```
SELECT
    p.categoryid, p.productname,
    CASE
        WHEN p.categoryid = 1 THEN 'Beverages'
        WHEN p.categoryid = 2 THEN 'Condiments'
        WHEN p.categoryid = 3 THEN 'Confections'
        WHEN p.categoryid = 4 THEN 'Dairy Products'
        WHEN p.categoryid = 5 THEN 'Grains/Cereals'
        WHEN p.categoryid = 6 THEN 'Meat/Poultry'
        WHEN p.categoryid = 7 THEN 'Produce'
        WHEN p.categoryid = 8 THEN 'Seafood'
        ELSE 'Other'
    END AS categoryname,
    CASE
        WHEN p.categoryid IN (1, 7, 8) THEN 'Campaign Products'
        ELSE 'Non-Campaign Products'
    END AS iscampaig
FROM Production.Products AS p;
```

7

**Please capture the results, what data is obtained from the query command above? Explain (Question 6)**

|    | categoryid | productname   | categoryname   | iscampaig             |
|----|------------|---------------|----------------|-----------------------|
| 1  | 1          | Product HHYDP | Beverages      | Campaign Products     |
| 2  | 1          | Product RECZE | Beverages      | Campaign Products     |
| 3  | 2          | Product IMEHJ | Condiments     | Non-Campaign Products |
| 4  | 2          | Product KSBRM | Condiments     | Non-Campaign Products |
| 5  | 2          | Product EPEIM | Condiments     | Non-Campaign Products |
| 6  | 2          | Product VAIIV | Condiments     | Non-Campaign Products |
| 7  | 7          | Product HMLNI | Produce        | Campaign Products     |
| 8  | 2          | Product WVJFP | Condiments     | Non-Campaign Products |
| 9  | 6          | Product AOZBW | Meat/Poultry   | Non-Campaign Products |
| 10 | 8          | Product YHXGE | Seafood        | Campaign Products     |
| 11 | 4          | Product QMVUN | Dairy Products | Non-Campaign Products |
| 12 | 4          | Product OSFNS | Dairy Products | Non-Campaign Products |
| 13 | 8          | Product POXFU | Seafood        | Campaign Products     |
| 14 | 7          | Product PWCJB | Produce        | Campaign Products     |
| 15 | 2          | Product KSZOI | Condiments     | Non-Campaign Products |
| 16 | 3          | Product PAFRH | Confections    | Non-Campaign Products |
| 17 | 6          | Product PLCAY | Meat/Poultry   | Non-Campaign Products |

Query executed successfully.

My Answer :

- a condition with value (1, 7, 8) will be justified and this expression will stop reading and return the result by printing whether the product is a campaign product or non-campaign product.



Based on question number 6, please display data that is in the 'seafood' category only and use the *ALIAS command* to change the column name as shown in the image below.

**Capture your SQL command and how many rows are produced (Question 7)**

8

|   | ID_KATEGORI | NAMA_PRODUK   | NAMA_KATEGORI | STATUS            |
|---|-------------|---------------|---------------|-------------------|
| 1 | 8           | Product ACRVI | Seafood       | Campaign Products |
| 2 | 8           | Product AQOKR | Seafood       | Campaign Products |
| 3 | 8           | Product CBRRL | Seafood       | Campaign Products |
| 4 | 8           | Product CKEDC | Seafood       | Campaign Products |
| 5 | 8           | Product EVFFA | Seafood       | Campaign Products |
| 6 | 8           | Product GMKIJ | Seafood       | Campaign Products |
| 7 | 8           | Product LYERX | Seafood       | Campaign Products |
| 8 | 8           | Product POXFU | Seafood       | Campaign Products |
| 9 | 8           | Product TTEEX | Seafood       | Campaign Products |

```
SQLQuery2.sql - LA...van Diantha F (52)*  □ X
SELECT
    p.categoryid,
    p.productname,
    CASE
        WHEN p.categoryid = 1 THEN 'Beverages'
        WHEN p.categoryid = 2 THEN 'Condiments'
        WHEN p.categoryid = 3 THEN 'Confections'
        WHEN p.categoryid = 4 THEN 'Dairy Products'
        WHEN p.categoryid = 5 THEN 'Grains/Cereals'
        WHEN p.categoryid = 6 THEN 'Meat/Poultry'
        WHEN p.categoryid = 7 THEN 'Produce'
        WHEN p.categoryid = 8 THEN 'Seafood'
        ELSE 'Other'
    END AS categoryname,
    CASE
        WHEN p.categoryid IN (1, 7, 8) THEN 'Campaign Products'
        ELSE 'Non-Campaign Products'
    END AS iscampaingn
FROM
    Production.Products AS p
WHERE
    p.categoryid = 8;
```

My Answer :

- With use syntax where filtered to show only category 8

9

Display employee data from HR.Employees table that comes from country 'USA' and city 'Seattle', use ALIAS command to change column name as shown below. **Capture your SQL command (Question 8)**

|   | FIRST_NAME | LAST_NAME | CITY    | COUNTRY |
|---|------------|-----------|---------|---------|
| 1 | Sara       | Davis     | Seattle | USA     |
| 2 | Maria      | Cameron   | Seattle | USA     |

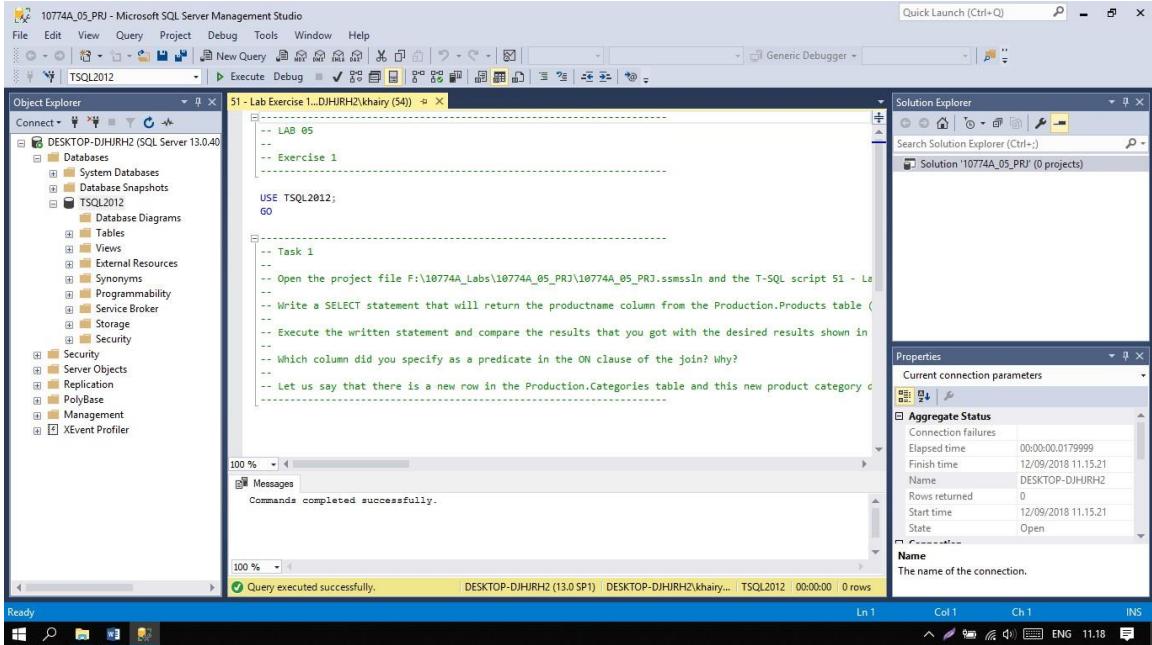
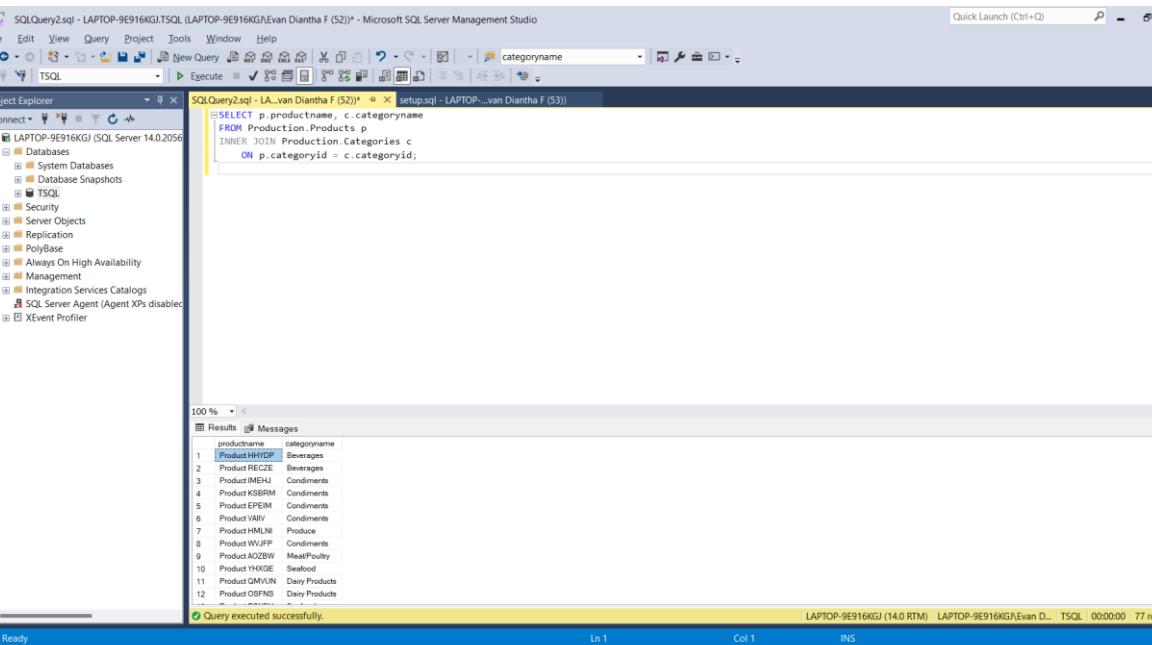
My Answer :



```
SQLQuery2.sql - LA...van Diantha F (52)*  X  setup.sql - LAPTOP-...van Diantha F (53))  
[=]SELECT  
    firstname AS FIRST_NAME,  
    lastname AS LAST_NAME,  
    city AS CITY,  
    country AS COUNTRY  
FROM  
    HR.Employees  
WHERE  
    city = 'Seattle' AND country = 'USA';
```



## Practical – Part 6 : Creating an Inner Join Query

| Step           | Information   |             |              |               |           |               |           |               |            |                |            |              |            |               |            |               |         |               |            |               |              |               |         |                |                |               |                |
|----------------|---|-------------|--------------|---------------|-----------|---------------|-----------|---------------|------------|----------------|------------|--------------|------------|---------------|------------|---------------|---------|---------------|------------|---------------|--------------|---------------|---------|----------------|----------------|---------------|----------------|
| 1              | <p>To experiment on this jobsheet, first log in to SQL Server Management Studio (SSMS). Then open the project \10774A_Labs\10774A_05_PRJ\10774A_05_PRJ.ssmssln and the T-SQL script 51 - Lab Exercise 1.sql. Make sure the database is connected to “ TSQL ”.</p>    |             |              |               |           |               |           |               |            |                |            |              |            |               |            |               |         |               |            |               |              |               |         |                |                |               |                |
| 2              | <p><b>[Question- 9 ]</b> Write a T-SQL SELECT that will display the productname column from the Production.Products table (use the alias table "p") and the categoryname column from the Production.Categories table (use the alias table "c") using inner join.</p> <p>My Answer :</p>  <pre> SELECT p.productname, c.categoryname FROM Production.Products p INNER JOIN Production.Categories c ON p.categoryid = c.categoryid; </pre> <table border="1"> <thead> <tr> <th>productname</th> <th>categoryname</th> </tr> </thead> <tbody> <tr><td>Product HHYDP</td><td>Beverages</td></tr> <tr><td>Product RECZE</td><td>Beverages</td></tr> <tr><td>Product IMEHJ</td><td>Condiments</td></tr> <tr><td>Product PIZZAM</td><td>Condiments</td></tr> <tr><td>Product EPEM</td><td>Condiments</td></tr> <tr><td>Product WWHII</td><td>Condiments</td></tr> <tr><td>Product HMLNI</td><td>Produce</td></tr> <tr><td>Product WJFJP</td><td>Condiments</td></tr> <tr><td>Product AJOZW</td><td>Meat/Poultry</td></tr> <tr><td>Product YHKGE</td><td>Seafood</td></tr> <tr><td>Product OMVJUN</td><td>Dairy Products</td></tr> <tr><td>Product QSFNB</td><td>Dairy Products</td></tr> </tbody> </table> | productname | categoryname | Product HHYDP | Beverages | Product RECZE | Beverages | Product IMEHJ | Condiments | Product PIZZAM | Condiments | Product EPEM | Condiments | Product WWHII | Condiments | Product HMLNI | Produce | Product WJFJP | Condiments | Product AJOZW | Meat/Poultry | Product YHKGE | Seafood | Product OMVJUN | Dairy Products | Product QSFNB | Dairy Products |
| productname    | categoryname  |             |              |               |           |               |           |               |            |                |            |              |            |               |            |               |         |               |            |               |              |               |         |                |                |               |                |
| Product HHYDP  | Beverages   |             |              |               |           |               |           |               |            |                |            |              |            |               |            |               |         |               |            |               |              |               |         |                |                |               |                |
| Product RECZE  | Beverages   |             |              |               |           |               |           |               |            |                |            |              |            |               |            |               |         |               |            |               |              |               |         |                |                |               |                |
| Product IMEHJ  | Condiments  |             |              |               |           |               |           |               |            |                |            |              |            |               |            |               |         |               |            |               |              |               |         |                |                |               |                |
| Product PIZZAM | Condiments  |             |              |               |           |               |           |               |            |                |            |              |            |               |            |               |         |               |            |               |              |               |         |                |                |               |                |
| Product EPEM   | Condiments  |             |              |               |           |               |           |               |            |                |            |              |            |               |            |               |         |               |            |               |              |               |         |                |                |               |                |
| Product WWHII  | Condiments  |             |              |               |           |               |           |               |            |                |            |              |            |               |            |               |         |               |            |               |              |               |         |                |                |               |                |
| Product HMLNI  | Produce   |             |              |               |           |               |           |               |            |                |            |              |            |               |            |               |         |               |            |               |              |               |         |                |                |               |                |
| Product WJFJP  | Condiments  |             |              |               |           |               |           |               |            |                |            |              |            |               |            |               |         |               |            |               |              |               |         |                |                |               |                |
| Product AJOZW  | Meat/Poultry  |             |              |               |           |               |           |               |            |                |            |              |            |               |            |               |         |               |            |               |              |               |         |                |                |               |                |
| Product YHKGE  | Seafood   |             |              |               |           |               |           |               |            |                |            |              |            |               |            |               |         |               |            |               |              |               |         |                |                |               |                |
| Product OMVJUN | Dairy Products  |             |              |               |           |               |           |               |            |                |            |              |            |               |            |               |         |               |            |               |              |               |         |                |                |               |                |
| Product QSFNB  | Dairy Products  |             |              |               |           |               |           |               |            |                |            |              |            |               |            |               |         |               |            |               |              |               |         |                |                |               |                |

**3**

Compare the results in step 2 with the file 52 - Lab Exercise 1 - Task 1 Result.txt. If they are the same then the T-SQL you wrote is correct.

| productname          | categoryname |
|----------------------|--------------|
| Product BWRLG        | Beverages    |
| Product HHYDP        | Beverages    |
| Product RECZE        | Beverages    |
| Product IMEHJ        | Condiments   |
| ...                  |              |
| ...                  |              |
| Product JYGFEE       | Beverages    |
| Product LUNZZ        | Condiments   |
| (77 row(s) affected) |              |

| productname          | categoryname |
|----------------------|--------------|
| Product BWRLG        | Beverages    |
| Product HHYDP        | Beverages    |
| Product RECZE        | Beverages    |
| Product IMEHJ        | Condiments   |
| ...                  |              |
| ...                  |              |
| Product JYGFEE       | Beverages    |
| Product LUNZZ        | Condiments   |
| (77 row(s) affected) |              |

**4**

**[Question- 10 ]** Which column is specified as a predicate in the ON join clause? Why?

**My Answer :**

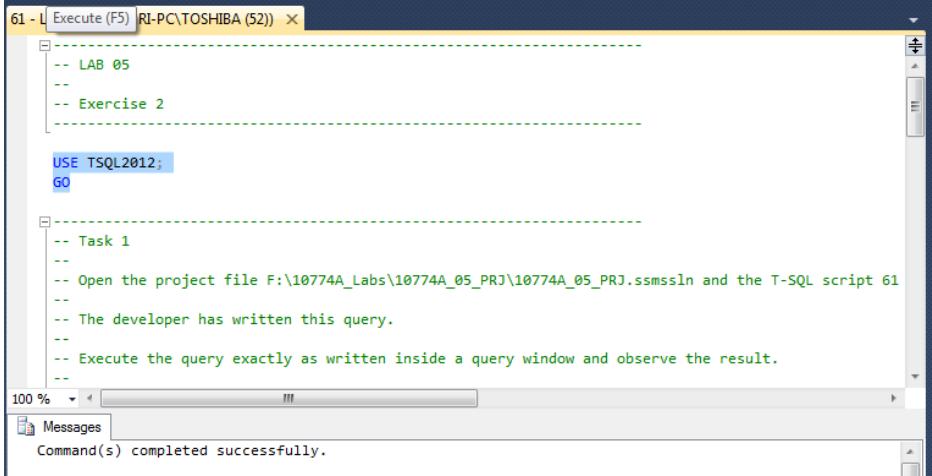
- The columns specified as predicates in the ON clause of the join are the categoryid of the Products table (p.categoryid) and the categoryid of the Categories table (c.categoryid). Because of the Relationship Between Tables: The categoryid column is used to join the two tables because it is a FOREIGN KEY in the Products table that refers to the PRIMARY KEY in the Categories table. This means that each product in the Products table has a category listed in the Categories table.

**5**

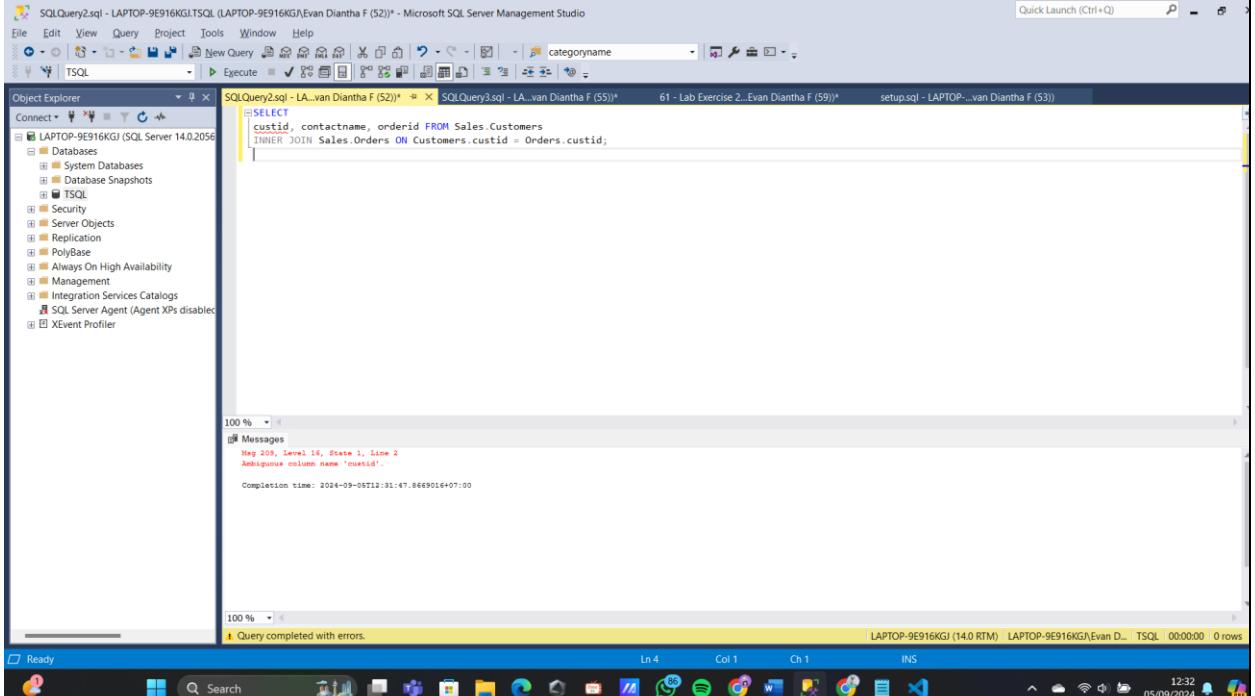
**Conclusion :** After carrying out this part of the practicum, students know and understand how to perform an INNER JOIN on two tables.



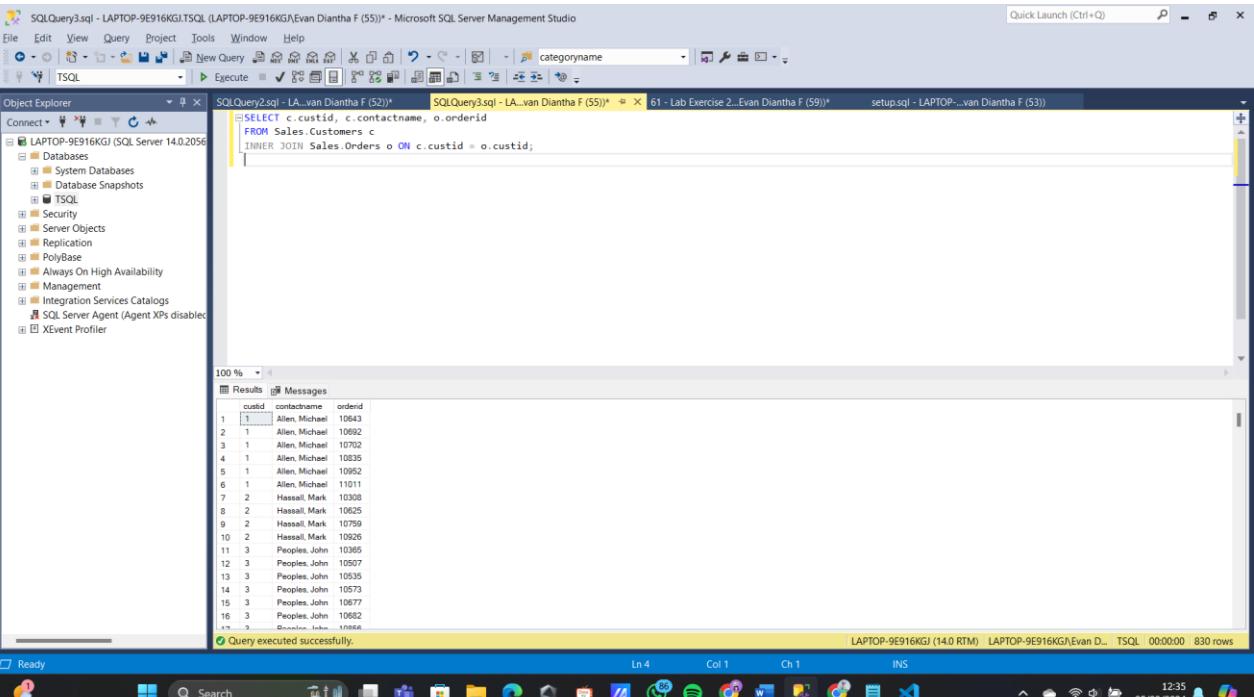
## Practical – Part 7 : Creating an Inner Join Query on Multiple Tables

| Step | Information   |
|------|---|
| 1    | <p>A developer will often be asked to run T-SQL files obtained from various departments . For example, the sales department wants a sales report of all customers for at least one order , with detailed information about each order. Then the developer will prepare the initialization of the SELECT statement to retrieve the custid and contactname columns in the Sales.Orders table. In accordance with the case study, this part 2 practicum will be carried out.</p> <p>Open the project \10774A Labs\10774A_05_PRJ\10774A_05_PRJ.ssmssln and the T-SQL script 61 - Lab Exercise 2.sql. Make sure the database is connected with “TSQL”.</p>  |
| 2    | <p>The developer will write T-SQL:</p> <pre>SELECT     custid , contactname , orderid FROM Sales . Customers INNER JOIN Sales . Orders ON Customers . custid = Orders . custid ;</pre> <p>Execute the T-SQL , and observe the results!</p>  |



|   |  |
|---|--|
|   |  A screenshot of Microsoft SQL Server Management Studio (SSMS) showing a query execution window. The query is: <pre>SELECT custid, contactname, orderid FROM Sales.Customers<br/>INNER JOIN Sales.Orders ON Customers.custid = Orders.custid;</pre> The results pane shows an error message: <p>Msg 209, Level 16, State 1, Line 2<br/>Ambiguous column name 'custid'.<br/>Completion time: 2024-09-05T12:31:47.8669016+07:00</p> Below the results pane, the status bar indicates "Query completed with errors." The taskbar at the bottom shows various application icons. |
| 3 | <p><b>[Question- 11 ]</b> After the 2nd stage of the experiment is carried out, an error will appear. What is the content of the error message? Why can this error occur? Explain!</p> <p>My Answer :</p> <ul style="list-style-type: none"><li>- The query is incorrect because of the mismatch in writing table aliases. In the query, the Sales.Customers and Sales.Orders tables are not given aliases, but their columns can be accessed directly by table name. Due to the inconsistent writing of tables and columns in the ON clause, SQL Server does not know which table is meant.</li></ul>   |
| 4 | <p><b>[Question- 12 ]</b> In this 4th trial, fix the error that occurred in the 3rd stage trial which explains that all table names have their own table identities.</p> <p>My Answer :</p>  |



|   |   |
|---|---|
|   |  <p>The Sales.Customers table is given the alias c and Sales.Orders is given the alias o. For clarity and consistency, the columns accessed in the SELECT and ON clauses are used with the aliases c.custid and o.custid.</p> |
| 5 | Observe and compare the results of the 4th stage trial with the file 62 - Lab Exercise 2 - Task 2 Result.txt. If the results are the same, then your answer is correct.   |



|   |   |
|---|---|
|   | <p>The screenshot shows the SQL Server Management Studio interface. At the top, there are two tabs: '62 - Lab Exercise 2 - Task 2 Result.txt' and '61 - Lab Exercise 2...RI-PC\TOSHIBA (52)*'. The results window displays a table with columns 'custid', 'contactname', and 'orderid'. The data includes multiple entries for customer ID 1 (Allen, Michael) and customer ID 91 (Conn, Steve). Below the table, it says '(830 row(s) affected)'. In the bottom-left pane, the Object Explorer shows the database structure. In the bottom-right pane, the Results tab shows the same data as the results window.</p>   |
| 6 | <p><b>[Question- 13 ]</b> Copy the T-SQL in the 4th stage of the test and modify it by using the alias table " c " to Sales.Custumers table and " o " for Sales.Orders table.</p> <p><b>My Answer :</b></p> <pre>SQLQuery2.sql - LA...van Diantha F (52)*          SQLQuery3.sql - LA...var SELECT c.custid, c.contactname, o.orderid FROM Sales.Customers c INNER JOIN Sales.Orders o ON c.custid = o.custid;</pre> <ul style="list-style-type: none"> <li>- The Sales.Customers table uses the alias c, while the Sales.Orders table uses the alias o. To maintain clarity and code consistency, the ON clause has been changed to support the use of aliases.</li> </ul> |
| 7 | Execute T-SQL on stage-6 test and compare the result with the result of stage 4 execution! If the result is the same then your T-SQL is correct.  |
|   | Change the column prefix in the SELECT clause to the full name, then execute the T-SQL!   |



8

62 - Lab Exercise 2 - Task 2 Result.txt 61 - Lab Exercise 2...RI-PC\TOSHIBA (52)\* X

```
-- Notice that there are full source table names written as table aliases.
-- Apply the needed changes to the SELECT statement so that it will run without an error. Test the code.
-- Observe and compare the results that you got with the recommended result shown in the file 62 - Lab Exercise 2 - Task 2 Result.txt

SELECT
    Customers.custid, Customers.contactname, Orders.orderid
FROM Sales.Customers AS c
INNER JOIN Sales.Orders AS o ON c.custid = o.custid;

-- Task 3
-- Copy the T-SQL statement from task 2 and modify it to use the table aliases "C" for the Sales.Customers
-- and "O" for the Sales.Orders tables.
```

100 % !!!

Messages

```
Msg 209, Level 16, State 1, Line 23
Ambiguous column name 'custid'.
Msg 4104, Level 16, State 1, Line 41
The multi-part identifier "Customers.custid" could not be bound.
Msg 4104, Level 16, State 1, Line 41
The multi-part identifier "Customers.contactname" could not be bound.
Msg 4104, Level 16, State 1, Line 41
The multi-part identifier "Orders.orderid" could not be bound.
```

9

**[Question- 14 ]** Why does the execution result of T-SQL stage 8 produce an error?

My Answer :

- causes errors because unclear column references (such as custid, contactname, and orderid) are used in the SELECT clause without mentioning the original table. Since the query uses INNER JOIN, SQL Server needs to know which table each column comes from. SQL Server cannot find the desired column if it does not have the table alias or full table name.

10

**[Question- 15 ]** Change the column name prefix in the T-SQL test step 8 with its alias name, then display the execution results!

My Answer :

SQLQuery2.sql - LA...van Diantha F (52)\* X SQLQuery3.s

```
SELECT
    c.custid, c.contactname, o.orderid
FROM Sales.Customers c
INNER JOIN Sales.Orders o
    ON c.custid = o.custid;
```

Results Messages

|    | custid | contactname    | orderid |
|----|--------|----------------|---------|
| 1  | 1      | Allen, Michael | 10643   |
| 2  | 1      | Allen, Michael | 10692   |
| 3  | 1      | Allen, Michael | 10702   |
| 4  | 1      | Allen, Michael | 10835   |
| 5  | 1      | Allen, Michael | 10952   |
| 6  | 1      | Allen, Michael | 11011   |
| 7  | 2      | Hassall, Mark  | 10308   |
| 8  | 2      | Hassall, Mark  | 10625   |
| 9  | 2      | Hassall, Mark  | 10759   |
| 10 | 2      | Hassall, Mark  | 10926   |
| 11 | 3      | Peoples, John  | 10365   |
| 12 | 3      | Peoples, John  | 10507   |
| 13 | 3      | Peoples, John  | 10555   |

Query executed successfully.

11

**Conclusion :** After carrying out this part of the practicum, you should now know and understand the importance of using table alias names and how to JOIN multiple tables (more than two tables).



## Practical – Part 8 : Creating a Self-Join Query

| Step | Information  |              |           |                       |           |       |       |   |   |       |      |     |      |   |   |      |     |                       |   |   |   |     |      |               |   |   |   |       |      |                      |   |   |   |      |      |               |   |   |   |       |      |                      |   |   |   |      |         |                      |   |   |   |         |       |                      |   |   |   |              |      |                      |   |
|------|--|--------------|-----------|-----------------------|-----------|-------|-------|---|---|-------|------|-----|------|---|---|------|-----|-----------------------|---|---|---|-----|------|---------------|---|---|---|-------|------|----------------------|---|---|---|------|------|---------------|---|---|---|-------|------|----------------------|---|---|---|------|---------|----------------------|---|---|---|---------|-------|----------------------|---|---|---|--------------|------|----------------------|---|
| 1    | <p>This practicum uses a case study in an HR department that wants to display reports on employees and managers. Some of the things that want to be displayed are the lastname, firstname, and title columns of the HR.Employees table for employees and managers.</p> <p>Open the project \10774A Labs\10774A_05_PRJ\10774A_05_PRJ.ssmssln and the T-SQL script 71 - Lab Exercise 3.sql. Make sure the database is connected with “TSQL”.</p> <pre>USE TSQL2012; GO  -- Task 1 -- Open the project file F:\10774A_Labs\10774A_05_PRJ\10774A_05_PRJ.ssmssln and -- In order to better understand the needed tasks, you will first write a SELECT -- Execute the written statement and compare the results that you got with the</pre>  |              |           |                       |           |       |       |   |   |       |      |     |      |   |   |      |     |                       |   |   |   |     |      |               |   |   |   |       |      |                      |   |   |   |      |      |               |   |   |   |       |      |                      |   |   |   |      |         |                      |   |   |   |         |       |                      |   |   |   |              |      |                      |   |
| 2    | <p><b>[Question- 16 ]</b> Write T-SQL using SELECT clause to display empid, lastname, firstname, title, and mgrid columns. on the table HR.Employees by giving the alias name “e” for the HR.Employees table.</p> <p><b>My Answer :</b></p> <table border="1"><thead><tr><th></th><th>empid</th><th>lastname</th><th>firstname</th><th>title</th><th>mgrid</th></tr></thead><tbody><tr><td>1</td><td>1</td><td>Davis</td><td>Sara</td><td>CEO</td><td>NULL</td></tr><tr><td>2</td><td>2</td><td>Funk</td><td>Don</td><td>Vice President, Sales</td><td>1</td></tr><tr><td>3</td><td>3</td><td>Lew</td><td>Judy</td><td>Sales Manager</td><td>2</td></tr><tr><td>4</td><td>4</td><td>Peled</td><td>Yael</td><td>Sales Representative</td><td>3</td></tr><tr><td>5</td><td>5</td><td>Buck</td><td>Sven</td><td>Sales Manager</td><td>2</td></tr><tr><td>6</td><td>6</td><td>Suurs</td><td>Paul</td><td>Sales Representative</td><td>5</td></tr><tr><td>7</td><td>7</td><td>King</td><td>Russell</td><td>Sales Representative</td><td>5</td></tr><tr><td>8</td><td>8</td><td>Cameron</td><td>Maria</td><td>Sales Representative</td><td>3</td></tr><tr><td>9</td><td>9</td><td>Dolgopyatova</td><td>Zoya</td><td>Sales Representative</td><td>5</td></tr></tbody></table> |              | empid     | lastname              | firstname | title | mgrid | 1 | 1 | Davis | Sara | CEO | NULL | 2 | 2 | Funk | Don | Vice President, Sales | 1 | 3 | 3 | Lew | Judy | Sales Manager | 2 | 4 | 4 | Peled | Yael | Sales Representative | 3 | 5 | 5 | Buck | Sven | Sales Manager | 2 | 6 | 6 | Suurs | Paul | Sales Representative | 5 | 7 | 7 | King | Russell | Sales Representative | 5 | 8 | 8 | Cameron | Maria | Sales Representative | 3 | 9 | 9 | Dolgopyatova | Zoya | Sales Representative | 5 |
|      | empid  | lastname     | firstname | title                 | mgrid     |       |       |   |   |       |      |     |      |   |   |      |     |                       |   |   |   |     |      |               |   |   |   |       |      |                      |   |   |   |      |      |               |   |   |   |       |      |                      |   |   |   |      |         |                      |   |   |   |         |       |                      |   |   |   |              |      |                      |   |
| 1    | 1  | Davis        | Sara      | CEO                   | NULL      |       |       |   |   |       |      |     |      |   |   |      |     |                       |   |   |   |     |      |               |   |   |   |       |      |                      |   |   |   |      |      |               |   |   |   |       |      |                      |   |   |   |      |         |                      |   |   |   |         |       |                      |   |   |   |              |      |                      |   |
| 2    | 2  | Funk         | Don       | Vice President, Sales | 1         |       |       |   |   |       |      |     |      |   |   |      |     |                       |   |   |   |     |      |               |   |   |   |       |      |                      |   |   |   |      |      |               |   |   |   |       |      |                      |   |   |   |      |         |                      |   |   |   |         |       |                      |   |   |   |              |      |                      |   |
| 3    | 3  | Lew          | Judy      | Sales Manager         | 2         |       |       |   |   |       |      |     |      |   |   |      |     |                       |   |   |   |     |      |               |   |   |   |       |      |                      |   |   |   |      |      |               |   |   |   |       |      |                      |   |   |   |      |         |                      |   |   |   |         |       |                      |   |   |   |              |      |                      |   |
| 4    | 4  | Peled        | Yael      | Sales Representative  | 3         |       |       |   |   |       |      |     |      |   |   |      |     |                       |   |   |   |     |      |               |   |   |   |       |      |                      |   |   |   |      |      |               |   |   |   |       |      |                      |   |   |   |      |         |                      |   |   |   |         |       |                      |   |   |   |              |      |                      |   |
| 5    | 5  | Buck         | Sven      | Sales Manager         | 2         |       |       |   |   |       |      |     |      |   |   |      |     |                       |   |   |   |     |      |               |   |   |   |       |      |                      |   |   |   |      |      |               |   |   |   |       |      |                      |   |   |   |      |         |                      |   |   |   |         |       |                      |   |   |   |              |      |                      |   |
| 6    | 6  | Suurs        | Paul      | Sales Representative  | 5         |       |       |   |   |       |      |     |      |   |   |      |     |                       |   |   |   |     |      |               |   |   |   |       |      |                      |   |   |   |      |      |               |   |   |   |       |      |                      |   |   |   |      |         |                      |   |   |   |         |       |                      |   |   |   |              |      |                      |   |
| 7    | 7  | King         | Russell   | Sales Representative  | 5         |       |       |   |   |       |      |     |      |   |   |      |     |                       |   |   |   |     |      |               |   |   |   |       |      |                      |   |   |   |      |      |               |   |   |   |       |      |                      |   |   |   |      |         |                      |   |   |   |         |       |                      |   |   |   |              |      |                      |   |
| 8    | 8  | Cameron      | Maria     | Sales Representative  | 3         |       |       |   |   |       |      |     |      |   |   |      |     |                       |   |   |   |     |      |               |   |   |   |       |      |                      |   |   |   |      |      |               |   |   |   |       |      |                      |   |   |   |      |         |                      |   |   |   |         |       |                      |   |   |   |              |      |                      |   |
| 9    | 9  | Dolgopyatova | Zoya      | Sales Representative  | 5         |       |       |   |   |       |      |     |      |   |   |      |     |                       |   |   |   |     |      |               |   |   |   |       |      |                      |   |   |   |      |      |               |   |   |   |       |      |                      |   |   |   |      |         |                      |   |   |   |         |       |                      |   |   |   |              |      |                      |   |
| 3    | <p><b>[Question- 17 ]</b> Execute the 2nd stage of the test and compare it with 72 - Lab Exercise 3 - Task 1 Result.txt . If the results are the same, then your test is correct.</p> <p><b>My Answer :</b></p>  |              |           |                       |           |       |       |   |   |       |      |     |      |   |   |      |     |                       |   |   |   |     |      |               |   |   |   |       |      |                      |   |   |   |      |      |               |   |   |   |       |      |                      |   |   |   |      |         |                      |   |   |   |         |       |                      |   |   |   |              |      |                      |   |



The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, the connection is to 'LAPTOP-9E916KG1 (SQL Server 14.0.2056)'. The 'HR.Employees' database is selected. In the center pane, a query window displays the following T-SQL code:

```
SELECT
    e.empid, e.lastname, e.firstname, e.title, e.mgrid
FROM HR.Employees e;
```

The results pane shows a table with 9 rows of employee data:

|   | empid | lastname     | firstname | title                 | mgrid |
|---|-------|--------------|-----------|-----------------------|-------|
| 1 | 1     | Davis        | Sara      | CEO                   | NULL  |
| 2 | 2     | Funk         | Don       | Vice President, Sales | 1     |
| 3 | 3     | Lew          | Judy      | Sales Manager         | 2     |
| 4 | 4     | Peled        | Yael      | Sales Representative  | 3     |
| 5 | 5     | Buck         | Sven      | Sales Manager         | 2     |
| 6 | 6     | Suurs        | Paul      | Sales Representative  | 5     |
| 7 | 7     | King         | Russell   | Sales Representative  | 5     |
| 8 | 8     | Cameron      | Maria     | Sales Representative  | 3     |
| 9 | 9     | Dolgopyatova | Zoya      | Sales Representative  | 5     |

At the bottom of the results pane, it says '(9 rows affected)'. The status bar at the bottom right shows the completion time as 'Completion time: 2024-09-05T15:03:05.8369094+07:00'.

- 4 [Question- 18 ] Copy the T-SQL in step 2 then modify it by adding columns about manager information, namely lastname, firstname using SELF-JOIN. Use the aliases mgrlastname and mgrfirstname to distinguish the names of managers and employees.

My Answer :

The screenshot shows the Microsoft SQL Server Management Studio interface. In the center pane, a query window displays the following T-SQL code, which includes a self-join on the 'HR.Employees' table:

```
SELECT
    e.empid, e.lastname, e.firstname, e.title, e.mgrid, mgr.lastname AS mgrlastname, mgr.firstname AS mgrfirstname
FROM HR.Employees e
LEFT JOIN HR.Employees mgr
    ON e.mgrid = mgr.empid;
```

The results pane shows a table with 9 rows of employee data, including manager information:

|   | empid | lastname     | firstname | title                 | mgrid | mgrlastname | mgrfirstname |
|---|-------|--------------|-----------|-----------------------|-------|-------------|--------------|
| 1 | 1     | Davis        | Sara      | CEO                   | NULL  | NULL        | NULL         |
| 2 | 2     | Funk         | Don       | Vice President, Sales | 1     | Davis       | Sara         |
| 3 | 3     | Lew          | Judy      | Sales Manager         | 2     | Funk        | Don          |
| 4 | 4     | Peled        | Yael      | Sales Representative  | 3     | Lew         | Judy         |
| 5 | 5     | Buck         | Sven      | Sales Manager         | 2     | Funk        | Don          |
| 6 | 6     | Suurs        | Paul      | Sales Representative  | 5     | Buck        | Sven         |
| 7 | 7     | King         | Russell   | Sales Representative  | 5     | Buck        | Sven         |
| 8 | 8     | Cameron      | Maria     | Sales Representative  | 3     | Lew         | Judy         |
| 9 | 9     | Dolgopyatova | Zoya      | Sales Representative  | 5     | Buck        | Sven         |

At the bottom of the results pane, it says 'Query executed successfully.'



5

**[Question- 19 ]** Execute the 2nd stage of the test and compare it with 7 3 - Lab Exercise 3 - Task 2 Result.txt . If the results are the same, then your test is correct.

**My Answer :**

```
SELECT
    e.empid, e.lastname, e.firstname, e.title, e.mgrid, mgr.lastname AS mgrlastname, mgr.firstname AS mgrfirstname
FROM HR.Employees e
LEFT JOIN HR.Employees mgr
ON e.mgrid = mgr.empid;
```

| empid | lastname   | firstname | title                 | mgrid | mgrlastname | mgrfirstname |
|-------|------------|-----------|-----------------------|-------|-------------|--------------|
| 1     | Davis      | Sara      | CEO                   | NULL  | NULL        | NULL         |
| 2     | Funk       | Don       | Vice President, Sales | 1     | Davis       | Sara         |
| 3     | Lew        | Judy      | Sales Manager         | 2     | Funk        | Don          |
| 4     | Peled      | Yael      | Sales Representative  | 3     | Lew         | Judy         |
| 5     | Buck       | Sven      | Sales Manager         | 2     | Funk        | Don          |
| 6     | Suurs      | Paul      | Sales Representative  | 5     | Buck        | Sven         |
| 7     | King       | Russell   | Sales Representative  | 5     | Buck        | Sven         |
| 8     | Cameron    | Maria     | Sales Representative  | 3     | Lew         | Judy         |
| 9     | Dolgopolov | Zoya      | Sales Representative  | 5     | Buck        | Sven         |

Query executed successfully.

Completion time: 2024-09-05T15:04:01.5103261+07:00

6

**[Question- 20 ]** Is it mandatory to write the table alias name when executing the SELF-JOIN command? Can the original table name be used as an alias name? Explain!

**My Answer :**

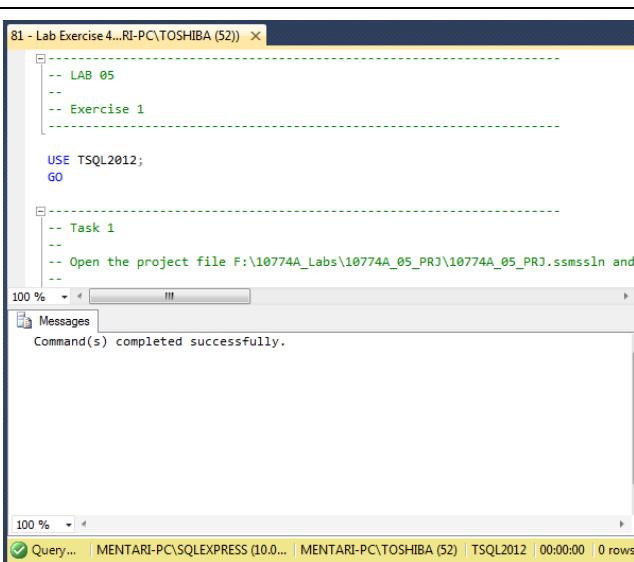
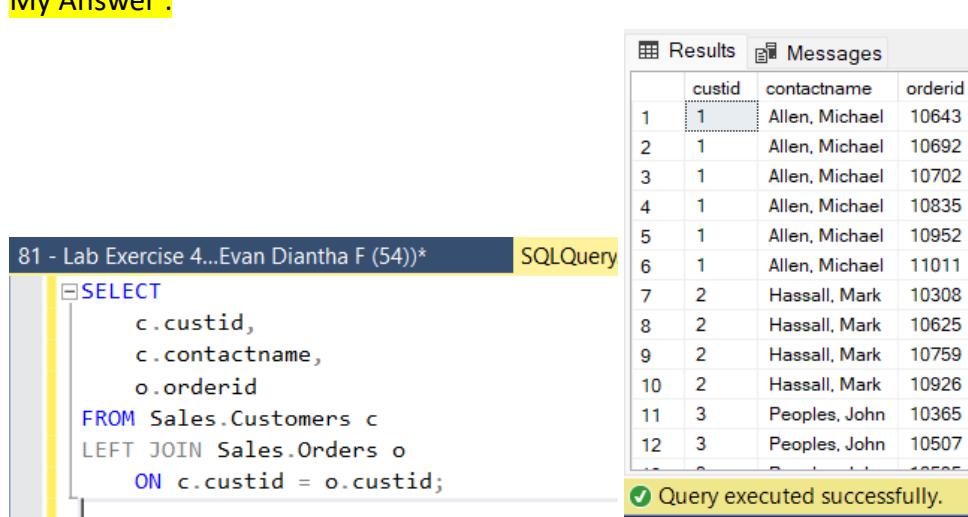
- When performing a self-join, table aliases are highly recommended as they help clarify the code, avoid ambiguity, and make the code more concise and readable. While the original table name can be used as an alias, a better practice to keep the code clear and maintainable is to use a shorter, more descriptive alias.

7

**Conclusion :** After doing this part of the practicum, you should understand how to write a T-SQL SELF-JOIN statement.



## Practical – Part 9 : Creating Outer-Join Query

| Step | Information  |
|------|--|
| 1    | <p>The case study used in this practicum part 4 continues the practicum in part 3. The sales department is quite satisfied with the report that has been made. Then the sales department wants to change the report to show all customers, even though the customer does not have an order history or customers who have an order history. Therefore, a SELECT clause is needed to retrieve all rows from the Sales.Customers table (custid and contactname columns) and the orderid column From the Sales.Orders table.</p> <p>Open the project \10774A Labs\10774A_05_PRJ\10774A_05_PRJ.ssmssln and the T-SQL script 81 - Lab Exercise 4.sql. Make sure the database is connected with “TSQL”.</p>  |
| 2    | <p><b>[Question- 21 ]</b> Write a T-SQL command with a SELECT clause to retrieve the custid and contactname columns from the table Sales.Customers and the orderid column from the Sales.Orders table . The command created must retrieve all rows from the Sales.Customers table .</p> <p><b>My Answer :</b></p>    |



3

**[Question- 22 ]** Execute the 2nd stage of the test and compare it with 82 - Lab Exercise 4 - Task 1 Result.txt . If the results are the same, then your test is correct.

**My Answer :**

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, a database named 'LAPTOP-9E916KGJ.TSQL' is selected. In the center pane, a query window displays the following T-SQL code:

```
SELECT c.custid, c.contactname, o.orderid
FROM Sales.Customers c
LEFT JOIN Sales.Orders o
ON c.custid = o.custid;
```

The results pane shows a table with three columns: custid, contactname, and orderid. The data consists of 832 rows, with the first few rows being:

| custid | contactname    | orderid |
|--------|----------------|---------|
| 1      | Allen, Michael | 10643   |
| 2      | Allen, Michael | 10692   |
| 3      | Allen, Michael | 10702   |
| 4      | Allen, Michael | 10835   |
| 5      | Allen, Michael | 10952   |
| 6      | Allen, Michael | 11011   |
| 7      | Hassall, Mark  | 10308   |
| 8      | Hassall, Mark  | 10625   |
| 9      | Hassall, Mark  | 10759   |
| 10     | Hassall, Mark  | 10926   |
| 11     | Peoples, John  | 10690   |
| 12     | Peoples, John  | 10697   |
| 13     | Peoples, John  | 10536   |
| 14     | Peoples, John  | 10573   |
| 15     | Peoples, John  | 10677   |
| 16     | Peoples, John  | 10682   |
| 17     | Peoples, John  | 10884   |

At the bottom of the results pane, it says '(832 rows affected)'. Below the results pane, the system tray shows the date and time as '05/09/2024 13:27'.

4

**[ Question- 23 ]** Pay attention to the values in the orderid column . Are there any missing values (NULL)? Why?

**My Answer :**

- If the orderid column in the query result shows a NULL value, this means that some customers do not have an associated order in the Sales.Orders table. The NULL value appears because using LEFT JOIN ensures that all rows from the Sales.Customers table are displayed, including customers who do not have orders. If a customer has no orders, the orderid column will contain NULL. To verify the number of customers without orders, you can use a query that counts the number of rows with NULL values in the orderid column.

5

**Conclusion :** After doing this part of the practicum, you should understand how to write the T-SQL OUTER-JOIN statement .



## Practical – Part 10 : Creating a Cross-Join Query

| Step | Information   |
|------|---|
| 1    | <p>This case study begins with the HR department wanting to set up a personal calendar for each employee. The IT department will provide a T-SQL code that generates all days in the past year. Therefore, <i>the developer</i> will use the SELECT clause to return all rows from the calendar table for each row in the HR.Employees table.</p> <p>Open the project \10774A Labs\10774A_05_PRJ\10774A_05_PRJ.ssmssln and the T-SQL script 91 - Lab Exercise 5.sql. Make sure the database is connected with “TSQL”.</p> <pre>-- LAB 05 --  -- Exercise 1  USE TSQL2012; GO  -- Task 1 -- Open the project file F:\10774A_Labs\10774A_05_PRJ\10774A_05_PRJ.ssmssln and the T-SQL script 91 - Lab Exercise 5.sql. Make sure the database is connected with “TSQL”. -- Execute the T-SQL code under Task 1. Do not worry if you do not understand the provided T-SQL code.  SET NOCOUNT ON;  Messages Command(s) completed successfully.</pre>   |
| 2    | <p><b>[ Question- 24 ]</b> Run the T-SQL code under task 1. Display the output! (Don't worry if you don't understand the T-SQL code. The next step will provide a more concrete example of how CROSS-JOIN is implemented.)</p> <p><b>My Answer :</b></p> <pre>91 - Lab Exercise 5.sql - LAPTOP-9E916KGJ.TSQL (LAPTOP-9E916KGJ\ Evan Diantha F (54)) - Microsoft SQL Server Management Studio File Edit View Query Project Tools Window Help File Edit View Project Tools Window Help New Query Object Explorer Task List Object Explorer Task List Object Explorer Object Explorer Connect Connect LAPTOP-9E916KGJ (SQL Server 14.0.2056) Databases Security System Databases Database Snapshots Replication PolyBase Always On High Availability Management Integration Services Catalogs SQL Server Agent (Agent XPs disabled) XEvent Profiler  91 - Lab Exercise 5...Evan Diantha F (54)* - SQLQuery1.sql - LA_van Diantha F (52)* SQLQuery2.sql - LA_van Diantha F (55)* setup.sql - LAPTOP-...van Diantha F (53)  91 - Lab Exercise 5...Evan Diantha F (54)* - SQLQuery2.sql - LA_van Diantha F (52)* SQLQuery3.sql - LA_van Diantha F (55)* setup.sql - LAPTOP-...van Diantha F (53)  Results Results calendardate 1 2024-01-01 2 2024-01-02 3 2024-01-03 4 2024-01-04 5 2024-01-05 6 2024-01-06 7 2024-01-07 8 2024-01-08 9 2024-01-09 10 2024-01-10 11 2024-01-11 12 2024-01-12 13 2024-01-13 14 2024-01-14 15 2024-01-15 16 2024-01-16 17 2024-01-17 18 2024-01-18 19 2024-01-19 20 2024-01-20 21 2024-01-21 22 2024-01-22 23 2024-01-23 24 2024-01-24 25 2024-01-25 26 2024-01-26 27 2024-01-27 28 2024-01-28 29 2024-01-29 30 2024-01-30 31 2024-01-31 32 2024-02-01 33 2024-02-02 34 2024-02-03 35 2024-02-04 36 2024-02-05 37 2024-02-06 38 2024-02-07 39 2024-02-08 40 2024-02-09 41 2024-02-10 42 2024-02-11 43 2024-02-12 44 2024-02-13 45 2024-02-14 46 2024-02-15 47 2024-02-16 48 2024-02-17 49 2024-02-18 50 2024-02-19 51 2024-02-20 52 2024-02-21 53 2024-02-22 54 2024-02-23 55 2024-02-24 56 2024-02-25 57 2024-02-26 58 2024-02-27 59 2024-02-28 60 2024-02-29 61 2024-03-01 62 2024-03-02 63 2024-03-03 64 2024-03-04 65 2024-03-05 66 2024-03-06 67 2024-03-07 68 2024-03-08 69 2024-03-09 70 2024-03-10 71 2024-03-11 72 2024-03-12 73 2024-03-13 74 2024-03-14 75 2024-03-15 76 2024-03-16 77 2024-03-17 78 2024-03-18 79 2024-03-19 80 2024-03-20 81 2024-03-21 82 2024-03-22 83 2024-03-23 84 2024-03-24 85 2024-03-25 86 2024-03-26 87 2024-03-27 88 2024-03-28 89 2024-03-29 90 2024-03-30 91 2024-03-31 92 2024-04-01 93 2024-04-02 94 2024-04-03 95 2024-04-04 96 2024-04-05 97 2024-04-06 98 2024-04-07 99 2024-04-08 100 2024-04-09 101 2024-04-10 102 2024-04-11 103 2024-04-12 104 2024-04-13 105 2024-04-14 106 2024-04-15 107 2024-04-16 108 2024-04-17 109 2024-04-18 110 2024-04-19 111 2024-04-20 112 2024-04-21 113 2024-04-22 114 2024-04-23 115 2024-04-24 116 2024-04-25 117 2024-04-26 118 2024-04-27 119 2024-04-28 120 2024-04-29 121 2024-04-30 122 2024-04-31 123 2024-05-01 124 2024-05-02 125 2024-05-03 126 2024-05-04 127 2024-05-05 128 2024-05-06 129 2024-05-07 130 2024-05-08 131 2024-05-09 132 2024-05-10 133 2024-05-11 134 2024-05-12 135 2024-05-13 136 2024-05-14 137 2024-05-15 138 2024-05-16 139 2024-05-17 140 2024-05-18 141 2024-05-19 142 2024-05-20 143 2024-05-21 144 2024-05-22 145 2024-05-23 146 2024-05-24 147 2024-05-25 148 2024-05-26 149 2024-05-27 150 2024-05-28 151 2024-05-29 152 2024-05-30 153 2024-05-31 154 2024-06-01 155 2024-06-02 156 2024-06-03 157 2024-06-04 158 2024-06-05 159 2024-06-06 160 2024-06-07 161 2024-06-08 162 2024-06-09 163 2024-06-10 164 2024-06-11 165 2024-06-12 166 2024-06-13 167 2024-06-14 168 2024-06-15 169 2024-06-16 170 2024-06-17 171 2024-06-18 172 2024-06-19 173 2024-06-20 174 2024-06-21 175 2024-06-22 176 2024-06-23 177 2024-06-24 178 2024-06-25 179 2024-06-26 180 2024-06-27 181 2024-06-28 182 2024-06-29 183 2024-06-30 184 2024-06-31 185 2024-07-01 186 2024-07-02 187 2024-07-03 188 2024-07-04 189 2024-07-05 190 2024-07-06 191 2024-07-07 192 2024-07-08 193 2024-07-09 194 2024-07-10 195 2024-07-11 196 2024-07-12 197 2024-07-13 198 2024-07-14 199 2024-07-15 200 2024-07-16 201 2024-07-17 202 2024-07-18 203 2024-07-19 204 2024-07-20 205 2024-07-21 206 2024-07-22 207 2024-07-23 208 2024-07-24 209 2024-07-25 210 2024-07-26 211 2024-07-27 212 2024-07-28 213 2024-07-29 214 2024-07-30 215 2024-07-31 216 2024-08-01 217 2024-08-02 218 2024-08-03 219 2024-08-04 220 2024-08-05 221 2024-08-06 222 2024-08-07 223 2024-08-08 224 2024-08-09 225 2024-08-10 226 2024-08-11 227 2024-08-12 228 2024-08-13 229 2024-08-14 230 2024-08-15 231 2024-08-16 232 2024-08-17 233 2024-08-18 234 2024-08-19 235 2024-08-20 236 2024-08-21 237 2024-08-22 238 2024-08-23 239 2024-08-24 240 2024-08-25 241 2024-08-26 242 2024-08-27 243 2024-08-28 244 2024-08-29 245 2024-08-30 246 2024-08-31 247 2024-09-01 248 2024-09-02 249 2024-09-03 250 2024-09-04 251 2024-09-05 252 2024-09-06 253 2024-09-07 254 2024-09-08 255 2024-09-09 256 2024-09-10 257 2024-09-11 258 2024-09-12 259 2024-09-13 260 2024-09-14 261 2024-09-15 262 2024-09-16 263 2024-09-17 264 2024-09-18 265 2024-09-19 266 2024-09-20 267 2024-09-21 268 2024-09-22 269 2024-09-23 270 2024-09-24 271 2024-09-25 272 2024-09-26 273 2024-09-27 274 2024-09-28 275 2024-09-29 276 2024-09-30 277 2024-10-01 278 2024-10-02 279 2024-10-03 280 2024-10-04 281 2024-10-05 282 2024-10-06 283 2024-10-07 284 2024-10-08 285 2024-10-09 286 2024-10-10 287 2024-10-11 288 2024-10-12 289 2024-10-13 290 2024-10-14 291 2024-10-15 292 2024-10-16 293 2024-10-17 294 2024-10-18 295 2024-10-19 296 2024-10-20 297 2024-10-21 298 2024-10-22 299 2024-10-23 300 2024-10-24 301 2024-10-25 302 2024-10-26 303 2024-10-27 304 2024-10-28 305 2024-10-29 306 2024-10-30 307 2024-10-31 308 2024-11-01 309 2024-11-02 310 2024-11-03 311 2024-11-04 312 2024-11-05 313 2024-11-06 314 2024-11-07 315 2024-11-08 316 2024-11-09 317 2024-11-10 318 2024-11-11 319 2024-11-12 320 2024-11-13 321 2024-11-14 322 2024-11-15 323 2024-11-16 324 2024-11-17 325 2024-11-18 326 2024-11-19 327 2024-11-20 328 2024-11-21 329 2024-11-22 330 2024-11-23 331 2024-11-24 332 2024-11-25 333 2024-11-26 334 2024-11-27 335 2024-11-28 336 2024-11-29 337 2024-11-30 338 2024-12-01 339 2024-12-02 340 2024-12-03 341 2024-12-04 342 2024-12-05 343 2024-12-06 344 2024-12-07 345 2024-12-08 346 2024-12-09 347 2024-12-10 348 2024-12-11 349 2024-12-12 350 2024-12-13 351 2024-12-14 352 2024-12-15 353 2024-12-16 354 2024-12-17 355 2024-12-18 356 2024-12-19 357 2024-12-20 358 2024-12-21 359 2024-12-22 360 2024-12-23 361 2024-12-24 362 2024-12-25 363 2024-12-26 364 2024-12-27 365 2024-12-28 366 2024-12-29 367 2024-12-30 368 2024-12-31 369 2025-01-01 370 2025-01-02 371 2025-01-03 372 2025-01-04 373 2025-01-05 374 2025-01-06 375 2025-01-07 376 2025-01-08 377 2025-01-09 378 2025-01-10 379 2025-01-11 380 2025-01-12 381 2025-01-13 382 2025-01-14 383 2025-01-15 384 2025-01-16 385 2025-01-17 386 2025-01-18 387 2025-01-19 388 2025-01-20 389 2025-01-21 390 2025-01-22 391 2025-01-23 392 2025-01-24 393 2025-01-25 394 2025-01-26 395 2025-01-27 396 2025-01-28 397 2025-01-29 398 2025-01-30 399 2025-01-31 400 2025-02-01 401 2025-02-02 402 2025-02-03 403 2025-02-04 404 2025-02-05 405 2025-02-06 406 2025-02-07 407 2025-02-08 408 2025-02-09 409 2025-02-10 410 2025-02-11 411 2025-02-12 412 2025-02-13 413 2025-02-14 414 2025-02-15 415 2025-02-16 416 2025-02-17 417 2025-02-18 418 2025-02-19 419 2025-02-20 420 2025-02-21 421 2025-02-22 422 2025-02-23 423 2025-02-24 424 2025-02-25 425 2025-02-26 426 2025-02-27 427 2025-02-28 428 2025-02-29 429 2025-02-30 430 2025-02-31 431 2025-03-01 432 2025-03-02 433 2025-03-03 434 2025-03-04 435 2025-03-05 436 2025-03-06 437 2025-03-07 438 2025-03-08 439 2025-03-09 440 2025-03-10 441 2025-03-11 442 2025-03-12 443 2025-03-13 444 2025-03-14 445 2025-03-15 446 2025-03-16 447 2025-03-17 448 2025-03-18 449 2025-03-19 450 2025-03-20 451 2025-03-21 452 2025-03-22 453 2025-03-23 454 2025-03-24 455 2025-03-25 456 2025-03-26 457 2025-03-27 458 2025-03-28 459 2025-03-29 460 2025-03-30 461 2025-03-31 462 2025-04-01 463 2025-04-02 464 2025-04-03 465 2025-04-04 466 2025-04-05 467 2025-04-06 468 2025-04-07 469 2025-04-08 470 2025-04-09 471 2025-04-10 472 2025-04-11 473 2025-04-12 474 2025-04-13 475 2025-04-14 476 2025-04-15 477 2025-04-16 478 2025-04-17 479 2025-04-18 480 2025-04-19 481 2025-04-20 482 2025-04-21 483 2025-04-22 484 2025-04-23 485 2025-04-24 486 2025-04-25 487 2025-04-26 488 2025-04-27 489 2025-04-28 490 2025-04-29 491 2025-04-30 492 2025-04-31 493 2025-05-01 494 2025-05-02 495 2025-05-03 496 2025-05-04 497 2025-05-05 498 2025-05-06 499 2025-05-07 500 2025-05-08 501 2025-05-09 502 2025-05-10 503 2025-05-11 504 2025-05-12 505 2025-05-13 506 2025-05-14 507 2025-05-15 508 2025-05-16 509 2025-05-17 510 2025-05-18 511 2025-05-19 512 2025-05-20 513 2025-05-21 514 2025-05-22 515 2025-05-23 516 2025-05-24 517 2025-05-25 518 2025-05-26 519 2025-05-27 520 2025-05-28 521 2025-05-29 522 2025-05-30 523 2025-05-31 524 2025-06-01 525 2025-06-02 526 2025-06-03 527 2025-06-04 528 2025-06-05 529 2025-06-06 530 2025-06-07 531 2025-06-08 532 2025-06-09 533 2025-06-10 534 2025-06-11 535 2025-06-12 536 2025-06-13 537 2025-06-14 538 2025-06-15 539 2025-06-16 540 2025-06-17 541 2025-06-18 542 2025-06-19 543 2025-06-20 544 2025-06-21 545 2025-06-22 546 2025-06-23 547 2025-06-24 548 2025-06-25 549 2025-06-26 550 2025-06-27 551 2025-06-28 552 2025-06-29 553 2025-06-30 554 2025-06-31 555 2025-07-01 556 2025-07-02 557 2025-07-03 558 2025-07-04 559 2025-07-05 560 2025-07-06 561 2025-07-07 562 2025-07-08 563 2025-07-09 564 2025-07-10 565 2025-07-11 566 2025-07-12 567 2025-07-13 568 2025-07-14 569 2025-07-15 570 2025-07-16 571 2025-07-17 572 2025-07-18 573 2025-07-19 574 2025-07-20 575 2025-07-21 576 2025-07-22 577 2025-07-23 578 2025-07-24 579 2025-07-25 580 2025-07-26 581 2025-07-27 582 2025-07-28 583 2025-07-29 584 2025-07-30 585 2025-07-31 586 2025-08-01 587 2025-08-02 588 2025-08-03 589 2025-08-04 590 2025-08-05 591 2025-08-06 592 2025-08-07 593 2025-08-08 594 2025-08-09 595 2025-08-10 596 2025-08-11 597 2025-08-12 598 2025-08-13 599 2025-08-14 600 2025-08-15 601 2025-08-16 602 2025-08-17 603 2025-08-18 604 2025-08-19 605 2025-08-20 606 2025-08-21 607 2025-08-22 608 2025-08-23 609 2025-08-24 610 2025-08-25 611 2025-08-26 612 2025-08-27 613 2025-08-28 614 2025-08-29 615 2025-08-30 616 2025-08-31 617 2025-09-01 618 2025-09-02 619 2025-09-03 620 2025-09-04 621 2025-09-05 622 2025-09-06 623 2025-09-07 624 2025-09-08 625 2025-09-09 626 2025-09-10 627 2025-09-11 628 2025-09-12 629 2025-09-13 630 2025-09-14 631 2025-09-15 632 2025-09-16 633 2025-09-17 634 2025-09-18 635 2025-09-19 636 2025-09-20 637 2025-09-21 638 2025-09-22 639 2025-09-23 640 2025-09-24 641 2025-09-25 642 2025-09-26 643 2025-09-27 644 2025-09-28 645 2025-09-29 646 2025-09-30 647 2025-09-31 648 2025-10-01 649 2025-10-02 650 2025-10-03 651 2025-10-04 652 2025-10-05 653 2025-10-06 654 2025-10-07 655 2025-10-08 656 2025-10-09 657 2025-10-10 658 2025-10-11 659 2025-10-12 660 2025-10-13 661 2025-10-14 662 2025-10-15 663 2025-10-16 664 2025-10-17 665 2025-10-18 666 2025-10-19 667 2025-10-20 668 2025-10-21 669 2025-10-22 670 2025-10-23 671 2025-10-24 672 2025-10-25 673 2025-10-26 674 2025-10-27 675 2025-10-28 676 2</pre> |



3

**[Question- 25 ]** Write a SELECT command to retrieve values from the empid, firstname, and lastname columns from the HR.Employees table and the calendardate column from the HR.Calendar table.

**My Answer :**

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, the connection is to LAPTOP-9E916KGJ\SQL Server 14.0.2056. In the center pane, a query window contains the following T-SQL code:

```
SELECT e.empid, e.firstname, e.lastname, c.calendardate
FROM HR.Employees AS e
CROSS JOIN
HR.Calendar AS c;
```

The results pane displays a table with 3294 rows, showing data for employee Sara Davis from January 1, 2024, to January 17, 2024. The columns are empid, firstname, lastname, and calendardate.

| empid | firstname | lastname | calendardate |
|-------|-----------|----------|--------------|
| 1     | Sara      | Davis    | 2024-01-01   |
| 2     | Sara      | Davis    | 2024-01-02   |
| 3     | Sara      | Davis    | 2024-01-03   |
| 4     | Sara      | Davis    | 2024-01-04   |
| 5     | Sara      | Davis    | 2024-01-05   |
| 6     | Sara      | Davis    | 2024-01-06   |
| 7     | Sara      | Davis    | 2024-01-07   |
| 8     | Sara      | Davis    | 2024-01-08   |
| 9     | Sara      | Davis    | 2024-01-09   |
| 10    | Sara      | Davis    | 2024-01-10   |
| 11    | Sara      | Davis    | 2024-01-11   |
| 12    | Sara      | Davis    | 2024-01-12   |
| 13    | Sara      | Davis    | 2024-01-13   |
| 14    | Sara      | Davis    | 2024-01-14   |
| 15    | Sara      | Davis    | 2024-01-15   |
| 16    | Sara      | Davis    | 2024-01-16   |
| 17    | Sara      | Davis    | 2024-01-17   |

At the bottom, it says "Query executed successfully."

4

**[Question- 26 ]** Execute the 3rd stage test and compare it with the file 92 - Lab Exercise 5 - Task 2 Result.txt . If the results are the same, then your test is correct.

**My Answer :**

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, the connection is to LAPTOP-9E916KGJ\SQL Server 14.0.2056. In the center pane, a query window contains the following T-SQL code:

```
SELECT e.empid, e.firstname, e.lastname, c.calendardate
FROM HR.Employees AS e
CROSS JOIN
HR.Calendar AS c;
```

The results pane displays a table with 3294 rows, showing data for employee Sara Davis from January 1, 2024, to January 17, 2024. The columns are empid, firstname, lastname, and calendardate.

| empid | firstname | lastname | calendardate |
|-------|-----------|----------|--------------|
| 1     | Sara      | Davis    | 2024-01-01   |
| 2     | Sara      | Davis    | 2024-01-02   |
| 3     | Sara      | Davis    | 2024-01-03   |
| 4     | Sara      | Davis    | 2024-01-04   |
| 5     | Sara      | Davis    | 2024-01-05   |
| 6     | Sara      | Davis    | 2024-01-06   |
| 7     | Sara      | Davis    | 2024-01-07   |
| 8     | Sara      | Davis    | 2024-01-08   |
| 9     | Sara      | Davis    | 2024-01-09   |
| 10    | Sara      | Davis    | 2024-01-10   |
| 11    | Sara      | Davis    | 2024-01-11   |
| 12    | Sara      | Davis    | 2024-01-12   |
| 13    | Sara      | Davis    | 2024-01-13   |
| 14    | Sara      | Davis    | 2024-01-14   |
| 15    | Sara      | Davis    | 2024-01-15   |
| 16    | Sara      | Davis    | 2024-01-16   |
| 17    | Sara      | Davis    | 2024-01-17   |

At the bottom, it says "Query executed successfully."



|   |  |
|---|--|
|   | <p>Results Messages</p> <p>(3294 rows affected)</p> <p>Completion time: 2024-09-05T15:00:40.7898355+07:00</p>                |
| 5 | <p>Drop the HR.Calendar table by executing the T-SQL code below task 3.</p>  |
| 6 | <p><b>Conclusion :</b> After completing this practical section, you will understand how to write T-SQL CROSS-JOIN code .</p> |



## Practical – Part 11 : Writing Queries Who Will Filter Data with WHERE clause

| Step | Information  |
|------|--|
| 1    | <p>The scenario in this practicum uses the problems in the marketing department. The marketing department is working on several campaigns for old customers. The marketing staff needs a different customer list according to several business rules. Therefore, <i>the developer</i> will write a SELECT command to retrieve the desired rows from the Sales.Customers table.</p> <p>Open the project \10774A Labs\10774A_06_PRJ\10774A_06_PRJ.ssmssln and the T-SQL script 51 - Lab Exercise 1.sql. Make sure the database is connected with "TSQL".</p>           |
| 2    | <p>Write a SELECT statement that will return the column values from a table, Then filter the results to only customers who are from "Brazil"!</p> <pre>SELECT     custid , companyname , contactname , address , city , country , telephone FROM Sales . Customers WHERE     country = Brazil' ;</pre> <p>Use of the N prefix for literal characters ( <code>N'Brazil'</code> ). This prefix is used because the country column is a Unicode data type. When expressing Unicode characters literally, the N character (for National) is specified as the prefix.</p> |
| 3    | <p><b>[Question- 27 ]</b> Execute the 2nd stage of the test and compare it with the file 52 - Lab Exercise 1 - Task 1 Result.txt . If the results are the same, then your test is correct.</p> <p><b>My Answer :</b></p>   |



The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, a connection to 'LAPTOP-9E916KGJ (SQL Server 14.0.2056)' is selected, with the 'Databases' node expanded. The 'TSQL' tab is active. In the main pane, a query window displays the following T-SQL code:

```
SELECT
    custid, companyname, contactname, address, city, country, phone
FROM Sales.Customers
WHERE
    country = N'Brazil';
```

The results pane shows a table with 9 rows of customer data from the 'Sales.Customers' table, filtered by country 'Brazil'. The columns are: custid, companyname, contactname, address, city, country, and phone. The data includes entries like Customer JLN00K (Richardson, Shaw) from São Paulo, Brazil, and Customer YQWVW (Nagel, Jean-Philippe) from Rio de Janeiro, Brazil.

At the bottom of the results pane, it says '(9 rows affected)' and 'Completion time: 2024-09-05T15:06:33.6943604+07:00'.



4

**[Question- 28 ]** Write a SELECT command that will return values in the custid, companyname, contactname, address, city, columns. country, and phone in the Sales.Customers table , then filter the results only for “Brazil, UK and USA” (Use the IN predicate in the WHERE clause).

**My Answer :**

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure. The main window contains a T-SQL query editor with the following code:

```
SELECT
    custid, companyname, contactname, address, city, country, phone
FROM Sales.Customers
WHERE
    country IN ('N'Brazil', 'N'UK', 'N'USA')
```

The results grid displays customer data for Brazil, UK, and USA. The results are as follows:

| custid | companyname    | contactname           | address                         | city                      | country | phone          |
|--------|----------------|-----------------------|---------------------------------|---------------------------|---------|----------------|
| 4      | Customer HRBZQ | Annett Torsten        | 7990 Hanover Sq                 | London                    | UK      | (171) 456-7890 |
| 11     | Customer UBHAU | Jaffe, David          | Faunlero Cicus 4567             | London                    | UK      | (171) 789-0123 |
| 15     | Customer JUNWK | Richardson, Shaw      | Avenida Lusitânia, 6789         | Sao Paulo                 | Brazil  | (11)012-3456   |
| 16     | Customer GYBBY | Birkby, Dana          | Berkley Gardens 0123 Brewery    | London                    | UK      | (171) 234-5678 |
| 19     | Customer RFNOC | Boseman, Randall      | 5678 King George                | London                    | UK      | (171) 345-6789 |
| 21     | Customer KIDPK | Russo, Giuseppe       | Rua Orla, 3456                  | Sao Paulo                 | Brazil  | (11)456-7890   |
| 31     | Customer YJCBX | Cheng, Yao-Qiang      | Av. Brasil, 5678                | Campinas                  | Brazil  | (11)567-8901   |
| 32     | Customer YSQX  | Krishnan, Venky       | 6789 Baker Blvd.                | Eugene                    | USA     | (503) 555-0122 |
| 34     | Customer IBVRO | Cohen, Shy            | Rua do Papo, 7890               | Rio de Janeiro            | Brazil  | (21)789-0123   |
| 36     | Customer LVJSO | Smith, Denise         | City Center Plaza 2345 Main St. | Elgin                     | USA     | (503) 555-0126 |
| 38     | Customer LIJUC | Lee, Frank            | Garden House Crowther Way 3458  | Coves                     | UK      | (198) 567-8901 |
| 43     | Customer UISUJ | Deshpande, Anu        | 8901 Orchestra Terreno          | Walla Walla               | USA     | (509) 555-0119 |
| 45     | Customer QXPPT | Sunkamurrali, Krishna | 1234 Polk St. Suite 5           | San Francisco             | USA     | (415) 555-0118 |
| 48     | Customer DVFBM | Szymczak, Radoslaw    | 9012 Chiaroscuro Rd.            | Portland                  | USA     | (503) 555-0117 |
| 53     | Customer GCJSQ | Mallit, Ken           | South House 1234 Queensbridge   | London                    | UK      | (171) 890-1234 |
| 55     | Customer KZQZT | Egelund-Muller, Arja  | 7890 Bering St                  | Anchorage                 | USA     | (907) 555-0115 |
| 56     | Customer KZQZT | Egelund-Muller, Arja  | Blvd. das Glorietas, 1234       | Blvd. das Glorietas, 1234 | Brasil  | (171) 789-0111 |

Below the results, a message indicates "Query executed successfully."

5

**[Question- 29 ]** Execute the 3rd stage test and compare it with file 53 - Lab Exercise 1 - Task 2 Result.txt . If the results are the same, then your test is correct.

**My Answer :**

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure. The main window contains a T-SQL query editor with the following code:

```
SELECT
    custid, companyname, contactname, address, city, country, phone
FROM Sales.Customers
WHERE
    country IN ('N'Brazil', 'N'UK', 'N'USA')
```

The results grid displays customer data for Brazil, UK, and USA. The results are as follows:

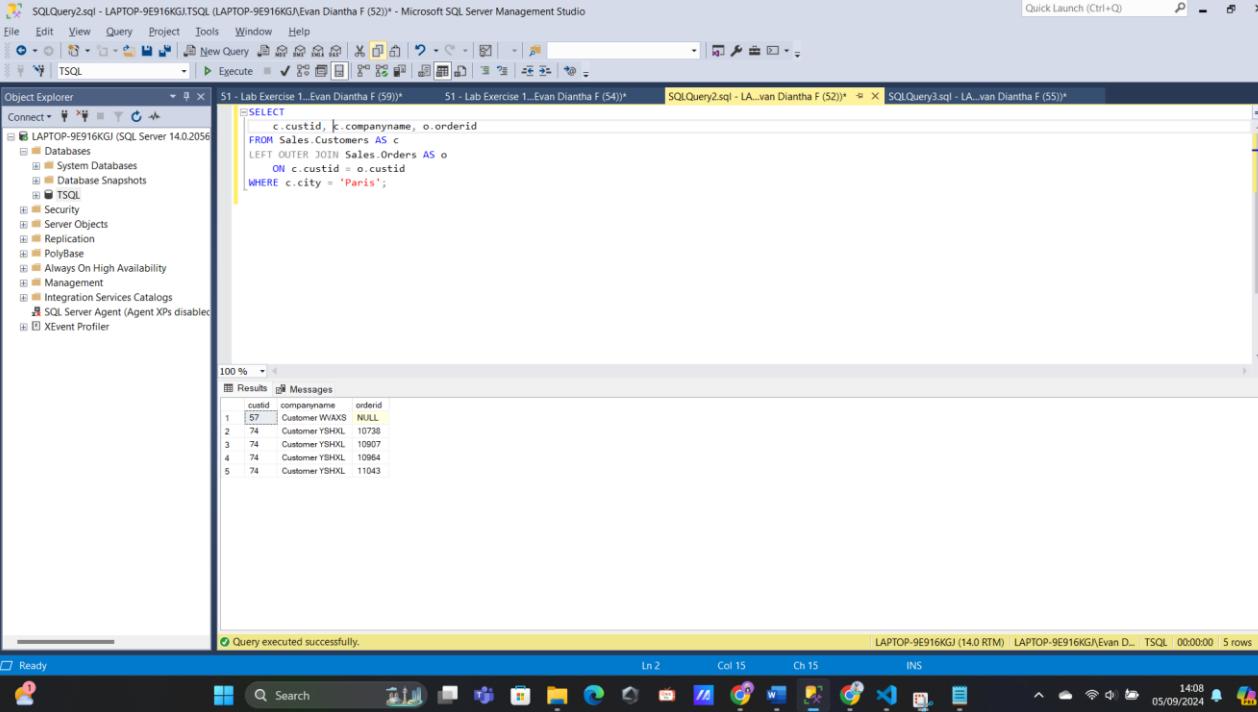
| custid | companyname    | contactname           | address                         | city                      | country | phone          |
|--------|----------------|-----------------------|---------------------------------|---------------------------|---------|----------------|
| 4      | Customer HRBZQ | Annett Torsten        | 7990 Hanover Sq                 | London                    | UK      | (171) 456-7890 |
| 11     | Customer UBHAU | Jaffe, David          | Faunlero Cicus 4567             | London                    | UK      | (171) 789-0123 |
| 15     | Customer JUNWK | Richardson, Shaw      | Avenida Lusitânia, 6789         | Sao Paulo                 | Brazil  | (11)012-3456   |
| 16     | Customer GYBBY | Birkby, Dana          | Berkley Gardens 0123 Brewery    | London                    | UK      | (171) 234-5678 |
| 19     | Customer RFNOC | Boseman, Randall      | 5678 King George                | London                    | UK      | (171) 345-6789 |
| 21     | Customer KIDPK | Russo, Giuseppe       | Rua Orla, 3456                  | Sao Paulo                 | Brazil  | (11)456-7890   |
| 31     | Customer YJCBX | Cheng, Yao-Qiang      | Av. Brasil, 5678                | Campinas                  | Brazil  | (11)567-8901   |
| 32     | Customer YSQX  | Krishnan, Venky       | 6789 Baker Blvd.                | Eugene                    | USA     | (503) 555-0122 |
| 34     | Customer IBVRO | Cohen, Shy            | Rua do Papo, 7890               | Rio de Janeiro            | Brazil  | (21)789-0123   |
| 36     | Customer LVJSO | Smith, Denise         | City Center Plaza 2345 Main St. | Elgin                     | USA     | (503) 555-0126 |
| 38     | Customer LIJUC | Lee, Frank            | Garden House Crowther Way 3458  | Coves                     | UK      | (198) 567-8901 |
| 43     | Customer UISUJ | Deshpande, Anu        | 8901 Orchestra Terreno          | Walla Walla               | USA     | (509) 555-0119 |
| 45     | Customer QXPPT | Sunkamurrali, Krishna | 1234 Polk St. Suite 5           | San Francisco             | USA     | (415) 555-0118 |
| 48     | Customer DVFBM | Szymczak, Radoslaw    | 9012 Chiaroscuro Rd.            | Portland                  | USA     | (503) 555-0117 |
| 53     | Customer GCJSQ | Mallit, Ken           | South House 1234 Queensbridge   | London                    | UK      | (171) 890-1234 |
| 55     | Customer KZQZT | Egelund-Muller, Arja  | 7890 Bering St                  | Anchorage                 | USA     | (907) 555-0115 |
| 56     | Customer KZQZT | Egelund-Muller, Arja  | Blvd. das Glorietas, 1234       | Blvd. das Glorietas, 1234 | Brasil  | (171) 789-0111 |

Below the results, a message indicates "Query executed successfully."



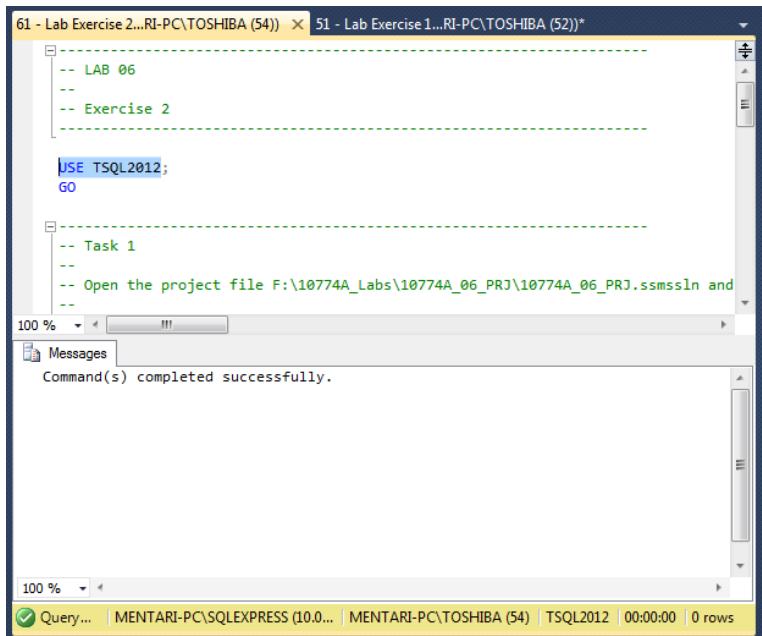
|   | <p>(29 rows affected)</p> <p>Completion time: 2024-09-05T15:08:09.5958652+07:00</p>   |                |         |             |         |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |
|---|---|----------------|---------|-------------|---------|---|---|----------------|------|---|---|----------------|------|---|---|----------------|------|---|---|----------------|------|---|---|----------------|------|---|---|----------------|------|---|---|----------------|------|---|---|----------------|------|---|---|----------------|------|
| 6 | <p>The IT department has written T-SQL code to return values in the custid, companyname columns in the Sales.Customers table and the orderid column. in the Sales.Orders table as below:</p> <pre>SELECT     c . custid , c . companyname , o . orderid FROM Sales . Customers AS c LEFT OUTER JOIN Sales . Orders AS o ON c . custid = o . custid AND c . city = 'Paris' ;</pre>   |                |         |             |         |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |
| 7 | <p>Query execution in the 7th stage of the trial. Note two things, first the query will retrieve all rows in the Sales.Customers table . Second, the use of the comparison operator with the ON clause makes the city column more specific, namely the same as the value "Paris".</p> <p>The screenshot shows the SQL Server Management Studio interface. A query window titled '51 - Lab Exercise 1...RI-PC\TOSHIBA (52)'' contains the following T-SQL code:</p> <pre>-- Is the result the same as in the first T-SQL statement? Why? What is the difference? SELECT     c.custid, c.companyname, o.orderid FROM Sales.Customers AS c LEFT OUTER JOIN Sales.Orders AS o ON c.custid = o.custid AND c.city = 'Paris';</pre> <p>Below the code, a task list shows 'Task 5' with the instruction 'Write a T-SQL statement to retrieve customers from the Sales.Customers table'. The results pane shows a table with 9 rows:</p> <table border="1"><thead><tr><th></th><th>custid</th><th>companyname</th><th>orderid</th></tr></thead><tbody><tr><td>1</td><td>1</td><td>Customer NRZBB</td><td>NULL</td></tr><tr><td>2</td><td>2</td><td>Customer MLTDN</td><td>NULL</td></tr><tr><td>3</td><td>3</td><td>Customer KBUDE</td><td>NULL</td></tr><tr><td>4</td><td>4</td><td>Customer HFBZG</td><td>NULL</td></tr><tr><td>5</td><td>5</td><td>Customer HGVLZ</td><td>NULL</td></tr><tr><td>6</td><td>6</td><td>Customer XHXJV</td><td>NULL</td></tr><tr><td>7</td><td>7</td><td>Customer QXVLA</td><td>NULL</td></tr><tr><td>8</td><td>8</td><td>Customer QUHWH</td><td>NULL</td></tr><tr><td>9</td><td>9</td><td>Customer RTXGC</td><td>NULL</td></tr></tbody></table> <p>The status bar at the bottom indicates: 100%   Query...   MENTARI-PC\SQLEXPRESS (10.0...)   MENTARI-PC\TOSHIBA (52)   TSQL2012   00:00:00   94 rows</p> |                | custid  | companyname | orderid | 1 | 1 | Customer NRZBB | NULL | 2 | 2 | Customer MLTDN | NULL | 3 | 3 | Customer KBUDE | NULL | 4 | 4 | Customer HFBZG | NULL | 5 | 5 | Customer HGVLZ | NULL | 6 | 6 | Customer XHXJV | NULL | 7 | 7 | Customer QXVLA | NULL | 8 | 8 | Customer QUHWH | NULL | 9 | 9 | Customer RTXGC | NULL |
|   | custid  | companyname    | orderid |             |         |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |
| 1 | 1   | Customer NRZBB | NULL    |             |         |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |
| 2 | 2   | Customer MLTDN | NULL    |             |         |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |
| 3 | 3   | Customer KBUDE | NULL    |             |         |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |
| 4 | 4   | Customer HFBZG | NULL    |             |         |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |
| 5 | 5   | Customer HGVLZ | NULL    |             |         |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |
| 6 | 6   | Customer XHXJV | NULL    |             |         |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |
| 7 | 7   | Customer QXVLA | NULL    |             |         |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |
| 8 | 8   | Customer QUHWH | NULL    |             |         |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |
| 9 | 9   | Customer RTXGC | NULL    |             |         |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |
| 8 | <p>[Question-30] Copy the T-SQL Code in step 7 then modify it with the comparison operator for the city column in the WHERE clause. After that execute the code, show the result!</p> <p>My Answer :</p>  |                |         |             |         |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |   |   |                |      |



|        |  <p>The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, the connection to 'LAPTOP-9E916KGJ (SQL Server 14.0.2056)' is selected. In the center pane, a query window displays the following T-SQL code:</p> <pre>SELECT c.custid, c.companyname, o.orderid FROM Sales.Customers AS c LEFT OUTER JOIN Sales.Orders AS o ON c.custid = o.custid WHERE c.city = 'Paris';</pre> <p>The results grid shows five rows of data:</p> <table border="1"><thead><tr><th>custid</th><th>companyname</th><th>orderid</th></tr></thead><tbody><tr><td>57</td><td>Customer WAXXS</td><td>NULL</td></tr><tr><td>74</td><td>Customer YSHXL</td><td>10738</td></tr><tr><td>74</td><td>Customer YSHXL</td><td>10907</td></tr><tr><td>74</td><td>Customer YSHXL</td><td>10964</td></tr><tr><td>74</td><td>Customer YSHXL</td><td>11043</td></tr></tbody></table> <p>At the bottom of the results grid, it says '(5 row(s) affected)'. The status bar at the bottom right indicates 'LAPTOP-9E916KGJ (14.0 RTM) LAPTOP-9E916KGJ\Evan D... TSQL 00:00:00 5 rows'.</p> | custid  | companyname | orderid | 57 | Customer WAXXS | NULL | 74 | Customer YSHXL | 10738 | 74 | Customer YSHXL | 10907 | 74 | Customer YSHXL | 10964 | 74 | Customer YSHXL | 11043 |
|--------|---|---------|-------------|---------|----|----------------|------|----|----------------|-------|----|----------------|-------|----|----------------|-------|----|----------------|-------|
| custid | companyname   | orderid |             |         |    |                |      |    |                |       |    |                |       |    |                |       |    |                |       |
| 57     | Customer WAXXS  | NULL    |             |         |    |                |      |    |                |       |    |                |       |    |                |       |    |                |       |
| 74     | Customer YSHXL  | 10738   |             |         |    |                |      |    |                |       |    |                |       |    |                |       |    |                |       |
| 74     | Customer YSHXL  | 10907   |             |         |    |                |      |    |                |       |    |                |       |    |                |       |    |                |       |
| 74     | Customer YSHXL  | 10964   |             |         |    |                |      |    |                |       |    |                |       |    |                |       |    |                |       |
| 74     | Customer YSHXL  | 11043   |             |         |    |                |      |    |                |       |    |                |       |    |                |       |    |                |       |
| 9      | Compare the results of step 9 with file 55 - Lab Exercise 1 - Task 4 Result.txt . If the results are the same, then your test is correct.   |         |             |         |    |                |      |    |                |       |    |                |       |    |                |       |    |                |       |
| 10     | <b>Conclusion :</b> After completing the practicum and answering the questions in this section, you should understand how to filter data rows from one or more tables using the WHERE clause with logical operator predicates.  |         |             |         |    |                |      |    |                |       |    |                |       |    |                |       |    |                |       |



## Practical – Part 11 : Writing Queries Which Will Sort Data with clause ORDER BY

| Step | Information   |
|------|---|
| 1    | <p>The case study in this lab is based on a problem in the sales department. The sales department wants to create a report that shows all orders with some customer information. In addition, there is an additional request to sort the data based on order dates and the customer IDs. The order rows in the previous lab were displayed without using the ORDER BY clause, therefore specifically for this lab section the WHERE command will be followed by the ORDER BY clause.</p> <p>Open the project \10774A Labs\10774A_06_PRJ\10774A_06_PRJ.ssmssln and the T-SQL script 61 - Lab Exercise 2.sql . Make sure the database is connected with "TSQL".</p>  |
| 2    | <p>[ Question- 31 ] Write a SELECT command to retrieve the custid, custname columns from the Sales.Customers table and the orderid, orderdate columns from the Sales.Orders table ! Filter the results only for orders on or after April 1, 2008. Then sort the results based on orderdate in descending order and custid in ascending order!</p> <p>My Answer :</p>  |



SQLQuery2.sql - LAPTOP-9E916KGJ.TSQL (LAPTOP-9E916KGJ\ Evan Diantha F (52)) - Microsoft SQL Server Management Studio

```

SELECT
    c.custid, c.companyname AS custname, o.orderid, o.orderdate
FROM Sales.Customers c
INNER JOIN Sales.Orders o
    ON c.custid = o.custid
WHERE o.orderdate > '2008-04-01'
ORDER BY
    o.orderdate DESC, c.custid ASC;

```

| custid | custname        | orderid | orderdate               |
|--------|-----------------|---------|-------------------------|
| 1      | Customer RTXGC  | 11076   | 2008-05-06 00:00:00.000 |
| 2      | Customer NYUHS  | 11077   | 2008-05-06 00:00:00.000 |
| 3      | Customer CCKOT  | 11075   | 2008-05-06 00:00:00.000 |
| 4      | Customer JMJKW  | 11074   | 2008-05-06 00:00:00.000 |
| 5      | Customer THHDOP | 11072   | 2008-05-05 00:00:00.000 |
| 6      | Customer OXFRU  | 11076   | 2008-05-05 00:00:00.000 |
| 7      | Customer XPNRN  | 11071   | 2008-05-05 00:00:00.000 |
| 8      | Customer AHQHT  | 11073   | 2008-05-05 00:00:00.000 |
| 9      | Customer FEVNN  | 11067   | 2008-05-04 00:00:00.000 |
| 10     | Customer WRFZJ  | 11068   | 2008-05-04 00:00:00.000 |
| 11     | Customer VNTK   | 11069   | 2008-05-04 00:00:00.000 |
| 12     | Customer YRQZL  | 11064   | 2008-05-01 00:00:00.000 |
| 13     | Customer LCOUJ  | 11064   | 2008-05-01 00:00:00.000 |
| 14     | Customer VBQTI  | 11066   | 2008-05-31 00:00:00.000 |
| 15     | Customer WMFEA  | 11069   | 2008-05-30 00:00:00.000 |
| 16     | Customer YSIQX  | 11061   | 2008-04-30 00:00:00.000 |
| 17     | Customer ERVZL  | 11063   | 2008-04-30 00:00:00.000 |

Query executed successfully.

[ Question- 32 ] Execute the 2nd stage of the test and compare it with the file 62 - Lab Exercise 2 - Task 1 Result.txt . If the results are the same, then your test is correct.

My Answer :

SQLQuery3.sql - LAPTOP-9E916KGJ.TSQL (LAPTOP-9E916KGJ\ Evan Diantha F (55)) - Microsoft SQL Server Management Studio

```

SELECT
    c.custid, c.companyname AS custname, o.orderid, o.orderdate
FROM Sales.Customers c
INNER JOIN Sales.Orders o
    ON c.custid = o.custid
WHERE o.orderdate > '2008-04-01'
ORDER BY
    o.orderdate DESC, c.custid ASC;

```

| custid | custname        | orderid | orderdate               |
|--------|-----------------|---------|-------------------------|
| 1      | Customer RTXGC  | 11076   | 2008-05-06 00:00:00.000 |
| 2      | Customer NYUHS  | 11077   | 2008-05-06 00:00:00.000 |
| 3      | Customer CCKOT  | 11075   | 2008-05-06 00:00:00.000 |
| 4      | Customer JMJKW  | 11074   | 2008-05-06 00:00:00.000 |
| 5      | Customer THHDOP | 11072   | 2008-05-05 00:00:00.000 |
| 6      | Customer OXFRU  | 11076   | 2008-05-05 00:00:00.000 |
| 7      | Customer XPNRN  | 11071   | 2008-05-05 00:00:00.000 |
| 8      | Customer AHQHT  | 11073   | 2008-05-05 00:00:00.000 |
| 9      | Customer FEVNN  | 11067   | 2008-05-04 00:00:00.000 |
| 10     | Customer WRFZJ  | 11068   | 2008-05-04 00:00:00.000 |
| 11     | Customer VNTK   | 11069   | 2008-05-04 00:00:00.000 |
| 12     | Customer YRQZL  | 11064   | 2008-05-01 00:00:00.000 |
| 13     | Customer LCOUJ  | 11064   | 2008-05-01 00:00:00.000 |
| 14     | Customer VBQTI  | 11066   | 2008-05-01 00:00:00.000 |
| 15     | Customer WMFEA  | 11069   | 2008-04-30 00:00:00.000 |
| 16     | Customer YSIQX  | 11061   | 2008-04-30 00:00:00.000 |
| 17     | Customer ERVZL  | 11063   | 2008-04-30 00:00:00.000 |

Query executed successfully.

Results Messages

(88 rows affected)

Completion time: 2024-09-05T14:15:53.2896494+07:00

The T-SQL command from the previous practicum followed by the WHERE command is as



follows:

**3**

```
SELECT
e . empid , e . lastname , e . firstname , e . title , e . mgrid ,
m . lastname AS mgrlastname , m . firstname AS mgrfirstname
FROM HR . Employees AS e
INNER JOIN HR . Employees AS m ON e . mgrid = m . empid
WHERE
mgrlastname = N'Buck' ;
```



[ Question- 33 ] Execute the T-SQL command at stage 3. Did an error occur? What is the error message? What do you think is the cause?

My Answer :

- The use of a column alias (mgrlastname) in the WHERE clause causes errors in T-SQL queries. This is due to the fact that this clause is processed before the application of the alias. To ensure the query can be executed correctly, you should add the original column name, for example, m.lastname, to the WHERE clause.

4

[Question-3 4] Make changes to the T-SQL command to fix the error in the 3rd trial, then execute it! Compare the execution results with the file 63 - Lab Exercise 2 - Task 2 Result.txt. If the same, then the test result is correct.

| empid | lastname     | firstname | title                | mgrid | mgrlastname |
|-------|--------------|-----------|----------------------|-------|-------------|
| 6     | Suurs        | Paul      | Sales Representative | 5     | Buck        |
| 7     | King         | Russell   | Sales Representative | 5     | Buck        |
| 9     | Dolgopyatova | Zoya      | Sales Representative | 5     | Buck        |

(3 row(s) affected)

My Answer :

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer pane on the left lists databases, server objects, and other system components. The Query Editor pane contains a T-SQL script:

```
SELECT
    e.empid, e.lastname, e.firstname, e.title, e.mgrid, m.lastname AS mgrlastname, m.firstname AS mgrfirstname
FROM HR.Employees AS e
INNER JOIN HR.Employees AS m ON e.mgrid = m.empid
WHERE
    m.lastname = 'Buck';
```

The Results pane displays the query results:

| empid | lastname     | firstname | title                | mgrid | mgrlastname | mgrfirstname |
|-------|--------------|-----------|----------------------|-------|-------------|--------------|
| 6     | Suurs        | Paul      | Sales Representative | 5     | Buck        | Sven         |
| 7     | King         | Russell   | Sales Representative | 5     | Buck        | Sven         |
| 9     | Dolgopyatova | Zoya      | Sales Representative | 5     | Buck        | Sven         |

Below the Results pane, a message states "Query executed successfully."



**[ Question- 35 ]** Copy the T-SQL command in experiment 4, and modify it to produce all employees ORDER BY manager's first name. Initially test using the table's original name, then test using the table's alias name! Execute the T-SQL and compare the results to the 64 - Lab Exercise 2 - Task 3 Result.txt file . If the results are the same, then the experiment was correct.

5

|   | empid        | lastname | firstname             | title | mgrid | mgrlastname |
|---|--------------|----------|-----------------------|-------|-------|-------------|
| 3 | Lew          | Judy     | Sales Manager         | 2     | Funk  |             |
| 5 | Buck         | Sven     | Sales Manager         | 2     | Funk  |             |
| 4 | Peled        | Yael     | Sales Representative  | 3     | Lew   |             |
| 8 | Cameron      | Maria    | Sales Representative  | 3     | Lew   |             |
| 2 | Funk         | Don      | Vice President, Sales | 1     | Davis |             |
| 6 | Suurs        | Paul     | Sales Representative  | 5     | Buck  |             |
| 7 | King         | Russell  | Sales Representative  | 5     | Buck  |             |
| 9 | Dolgopyatova | Zoya     | Sales Representative  | 5     | Buck  |             |

(8 row(s) affected)

My Answer :

The screenshot shows the Microsoft SQL Server Management Studio interface. In the center, there is a results grid displaying employee data. At the top, there is a query editor window containing the following T-SQL code:

```

SELECT
    e.empid, e.lastname, e.firstname, e.title, e.mgrid, m.lastname AS mgrlastname, m.firstname AS mgrfirstname
FROM HR.Employees AS e
INNER JOIN HR.Employees AS m ON e.mgrid = m.empid
ORDER BY
    m.firstname;
  
```

The results grid below the query editor shows the following data:

|   | empid | lastname     | firstname | title                 | mgrid | mgrlastname | mgrfirstname |
|---|-------|--------------|-----------|-----------------------|-------|-------------|--------------|
| 1 | 3     | Lew          | Judy      | Sales Manager         | 2     | Funk        | Don          |
| 2 | 5     | Buck         | Sven      | Sales Manager         | 2     | Funk        | Don          |
| 3 | 4     | Peled        | Yael      | Sales Representative  | 3     | Lew         | Judy         |
| 4 | 8     | Cameron      | Maria     | Sales Representative  | 3     | Lew         | Judy         |
| 5 | 2     | Funk         | Don       | Vice President, Sales | 1     | Davis       | Sara         |
| 6 | 6     | Suurs        | Paul      | Sales Representative  | 5     | Buck        | Sven         |
| 7 | 7     | King         | Russell   | Sales Representative  | 5     | Buck        | Sven         |
| 8 | 9     | Dolgopyatova | Zoya      | Sales Representative  | 5     | Buck        | Sven         |

At the bottom of the screen, the taskbar shows the date and time as 05/09/2024 14:28.

6

**[Question-3 6 ]** Why can we use column names according to the original table name or use table alias names?

My Answer :

- Since SQL is processed before the application of column aliases, the original column names of the table are used in clauses such as WHERE. The original column names are necessary for conditions and calculations that occur before the aliases are applied, but table aliases make it easier to reference tables, especially when performing joins or when the same table is used more than once in a query. Table aliases also help simplify and clarify complex queries.

7

**Conclusion :** After working on the practical work and questions in this section, you should now understand how to use the ORDER BY clause .



## Practical – Part 12 : Writing Queries Who Will Do Data Filtering with clauses TOP

| Step           | Information   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
|----------------|---|-------------|-----------|----------------|--------|---------------|--------|----------------|-------|---------------|-------|---------------|-------|---------------|-------|---------------|-------|---------------|-------|---------------|-------|---------------|-------|--------------|-------|---------------|-------|---------------|-------|---------------|-------|---------------|-------|---------------|-------|---------------|-------|
| 1              | <p>Part 8 of the lab uses a case study on the sales department. The sales department wants to create an additional report that shows the order invoices and the 10 percent of the most expensive products that have been sold.</p> <p>Open the project \10774A Labs\10774A_06_PRJ\10774A_06_PRJ.ssmssln and the T-SQL script 71 - Lab Exercise 3.sql . Make sure the database is connected with “TSQL”.</p>   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| 2              | <p><b>[ Question- 37 ]</b> Write a SELECT command to display the productname and unitprice columns in the Production.Products table sorted descending by unitprice! Show the execution results!</p> <p><b>My Answer :</b></p> <table border="1"><thead><tr><th>productname</th><th>unitprice</th></tr></thead><tbody><tr><td>Product QDDOMO</td><td>263.50</td></tr><tr><td>Product VJXYN</td><td>123.79</td></tr><tr><td>Product AJOZBW</td><td>97.00</td></tr><tr><td>Product GHFPF</td><td>81.00</td></tr><tr><td>Product CKEDC</td><td>62.50</td></tr><tr><td>Product UKURJ</td><td>55.00</td></tr><tr><td>Product APTIJ</td><td>53.00</td></tr><tr><td>Product WUXYK</td><td>49.30</td></tr><tr><td>Product KZVQD</td><td>49.00</td></tr><tr><td>Product OFPNT</td><td>45.60</td></tr><tr><td>Product SMOH</td><td>43.90</td></tr><tr><td>Product HKHKK</td><td>43.90</td></tr><tr><td>Product WUJPF</td><td>40.00</td></tr><tr><td>Product BLCLX</td><td>39.00</td></tr><tr><td>Product OSFNS</td><td>38.00</td></tr><tr><td>Product VKCMF</td><td>38.00</td></tr><tr><td>Product DPAVX</td><td>36.00</td></tr></tbody></table> | productname | unitprice | Product QDDOMO | 263.50 | Product VJXYN | 123.79 | Product AJOZBW | 97.00 | Product GHFPF | 81.00 | Product CKEDC | 62.50 | Product UKURJ | 55.00 | Product APTIJ | 53.00 | Product WUXYK | 49.30 | Product KZVQD | 49.00 | Product OFPNT | 45.60 | Product SMOH | 43.90 | Product HKHKK | 43.90 | Product WUJPF | 40.00 | Product BLCLX | 39.00 | Product OSFNS | 38.00 | Product VKCMF | 38.00 | Product DPAVX | 36.00 |
| productname    | unitprice   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product QDDOMO | 263.50  |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product VJXYN  | 123.79  |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product AJOZBW | 97.00   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product GHFPF  | 81.00   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product CKEDC  | 62.50   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product UKURJ  | 55.00   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product APTIJ  | 53.00   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product WUXYK  | 49.30   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product KZVQD  | 49.00   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product OFPNT  | 45.60   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product SMOH   | 43.90   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product HKHKK  | 43.90   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product WUJPF  | 40.00   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product BLCLX  | 39.00   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product OSFNS  | 38.00   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product VKCMF  | 38.00   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |
| Product DPAVX  | 36.00   |             |           |                |        |               |        |                |       |               |       |               |       |               |       |               |       |               |       |               |       |               |       |              |       |               |       |               |       |               |       |               |       |               |       |               |       |

**[ Question- 38 ]** Copy and modify the T-SQL command in trial 2 with the limitation that only 10 percent of the child products are displayed based on unitprice ordering! Execute the command, and compare whether it is in accordance with the file 73 - Lab Exercise 3 - Task 2 Result.txt.

3

| productname         | unitprice |
|---------------------|-----------|
| <hr/>               |           |
| Product QDOME       | 263.50    |
| Product VJXVN       | 123.79    |
| Product AOZBW       | 97.00     |
| Product QHFFP       | 81.00     |
| Product CKEDC       | 62.50     |
| Product UKXRI       | 55.00     |
| Product APITJ       | 53.00     |
| Product WUXYK       | 49.30     |
| (8 row(s) affected) |           |

My Answer :

The screenshot shows the Microsoft SQL Server Management Studio interface. The top window displays the results of a query from '73 - Lab Exercise 3 - Task 2 Result.txt', showing eight rows of product names and unit prices. Below this, the Object Explorer shows the database structure. In the center, three query panes are open: '61 - Lab Exercise 2...Evan Diantha F (60)\*' contains the original T-SQL code using ROW\_NUMBER(), 'SQLQuery2.sql - LA...van Diantha F (52)\*' shows the modified code using a CTE and WHERE clause, and 'SQLQuery3.sql - LA...van Diantha F (55)\*' shows the result of the modified query with only 8 rows. The bottom status bar indicates the query was executed successfully.

```
;WITH SortedProducts AS (
    SELECT
        productname, unitprice,
        ROW_NUMBER() OVER (ORDER BY unitprice DESC) AS RowNum,
        COUNT(*) OVER () AS TotalRows
    FROM Production.Products
)
SELECT
    productname, unitprice
FROM SortedProducts
WHERE RowNum <= (TotalRows * 0.10);
```

4

**[ Question- 39 ]** Is it possible to implement the 5 trial T-SQL command using the OFFSET-FETCH clause?

My Answer :

- In T-SQL queries, you can use the OFFSET-FETCH clause instead of ROW\_NUMBER(). Using this clause, you can specify a subset of data based on the order and number of rows retrieved; for example, you can count the total rows, specify 10% of that number, and then use OFFSET-FETCH to retrieve the calculated number of rows. More advanced methods of paging and retrieving subsets of data are supported by SQL Server 2012 and later versions.

5

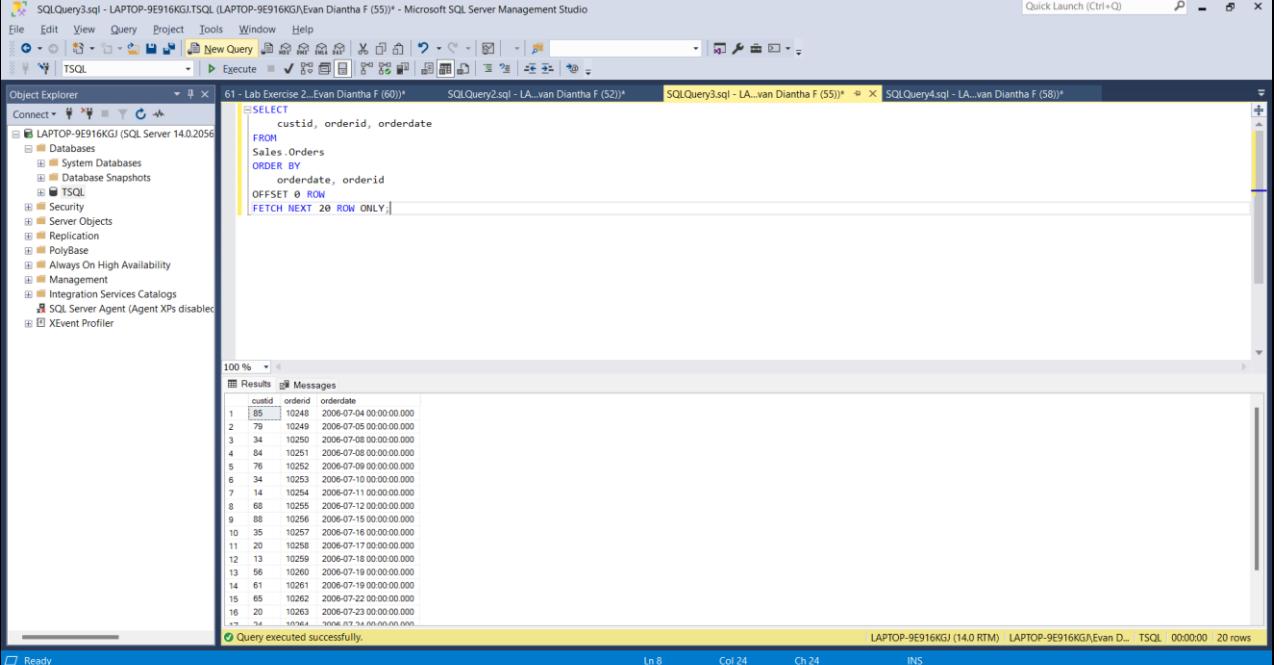
**Conclusion :** After completing the practical work and questions in this section, you should now understand how to apply the TOP option to the SELECT clause of the T-SQL command.



## Practical – Part 13 : Writing Queries Who Will Filter Data with OFFSET-FETCH clause

| Step                 | Information   |                         |         |           |    |       |                         |    |       |                         |    |       |                         |     |  |  |     |  |  |     |  |  |   |       |                         |    |       |                         |    |       |                         |                      |  |  |
|----------------------|---|-------------------------|---------|-----------|----|-------|-------------------------|----|-------|-------------------------|----|-------|-------------------------|-----|--|--|-----|--|--|-----|--|--|---|-------|-------------------------|----|-------|-------------------------|----|-------|-------------------------|----------------------|--|--|
| 1                    | <p>Practical part 9 will implement paging solution to display rows from Sales.Orders table , because the number of rows is too many. On each report page, user can only see 20 rows.</p> <p>Open the project \10774A Labs\10774A_06_PRJ\10774A_06_PRJ.ssmssln and the T-SQL script 81 - Lab Exercise 4.sql . Make sure the database is connected with “TSQL”.</p>   |                         |         |           |    |       |                         |    |       |                         |    |       |                         |     |  |  |     |  |  |     |  |  |   |       |                         |    |       |                         |    |       |                         |                      |  |  |
| 2                    | <p><b>[Question- 40 ]</b> Write a SELECT command to display the custid, orderid, and orderdate columns in the Sales.Orders table . Sort the rows by orderdate and orderid. Take the first 20 rows. Execute the command and compare the results with the file 82 - Lab Exercise 4 - Task 1 Result.txt. If the results are the same, then your test is correct.</p> <table border="1"><thead><tr><th>custid</th><th>orderid</th><th>orderdate</th></tr></thead><tbody><tr><td>85</td><td>10248</td><td>2006-07-04 00:00:00.000</td></tr><tr><td>79</td><td>10249</td><td>2006-07-05 00:00:00.000</td></tr><tr><td>34</td><td>10250</td><td>2006-07-08 00:00:00.000</td></tr><tr><td>...</td><td></td><td></td></tr><tr><td>...</td><td></td><td></td></tr><tr><td>...</td><td></td><td></td></tr><tr><td>7</td><td>10265</td><td>2006-07-25 00:00:00.000</td></tr><tr><td>87</td><td>10266</td><td>2006-07-26 00:00:00.000</td></tr><tr><td>25</td><td>10267</td><td>2006-07-29 00:00:00.000</td></tr><tr><td colspan="3">(20 row(s) affected)</td></tr></tbody></table> <p>My Answer :</p> | custid                  | orderid | orderdate | 85 | 10248 | 2006-07-04 00:00:00.000 | 79 | 10249 | 2006-07-05 00:00:00.000 | 34 | 10250 | 2006-07-08 00:00:00.000 | ... |  |  | ... |  |  | ... |  |  | 7 | 10265 | 2006-07-25 00:00:00.000 | 87 | 10266 | 2006-07-26 00:00:00.000 | 25 | 10267 | 2006-07-29 00:00:00.000 | (20 row(s) affected) |  |  |
| custid               | orderid   | orderdate               |         |           |    |       |                         |    |       |                         |    |       |                         |     |  |  |     |  |  |     |  |  |   |       |                         |    |       |                         |    |       |                         |                      |  |  |
| 85                   | 10248   | 2006-07-04 00:00:00.000 |         |           |    |       |                         |    |       |                         |    |       |                         |     |  |  |     |  |  |     |  |  |   |       |                         |    |       |                         |    |       |                         |                      |  |  |
| 79                   | 10249   | 2006-07-05 00:00:00.000 |         |           |    |       |                         |    |       |                         |    |       |                         |     |  |  |     |  |  |     |  |  |   |       |                         |    |       |                         |    |       |                         |                      |  |  |
| 34                   | 10250   | 2006-07-08 00:00:00.000 |         |           |    |       |                         |    |       |                         |    |       |                         |     |  |  |     |  |  |     |  |  |   |       |                         |    |       |                         |    |       |                         |                      |  |  |
| ...                  |   |                         |         |           |    |       |                         |    |       |                         |    |       |                         |     |  |  |     |  |  |     |  |  |   |       |                         |    |       |                         |    |       |                         |                      |  |  |
| ...                  |   |                         |         |           |    |       |                         |    |       |                         |    |       |                         |     |  |  |     |  |  |     |  |  |   |       |                         |    |       |                         |    |       |                         |                      |  |  |
| ...                  |   |                         |         |           |    |       |                         |    |       |                         |    |       |                         |     |  |  |     |  |  |     |  |  |   |       |                         |    |       |                         |    |       |                         |                      |  |  |
| 7                    | 10265   | 2006-07-25 00:00:00.000 |         |           |    |       |                         |    |       |                         |    |       |                         |     |  |  |     |  |  |     |  |  |   |       |                         |    |       |                         |    |       |                         |                      |  |  |
| 87                   | 10266   | 2006-07-26 00:00:00.000 |         |           |    |       |                         |    |       |                         |    |       |                         |     |  |  |     |  |  |     |  |  |   |       |                         |    |       |                         |    |       |                         |                      |  |  |
| 25                   | 10267   | 2006-07-29 00:00:00.000 |         |           |    |       |                         |    |       |                         |    |       |                         |     |  |  |     |  |  |     |  |  |   |       |                         |    |       |                         |    |       |                         |                      |  |  |
| (20 row(s) affected) |   |                         |         |           |    |       |                         |    |       |                         |    |       |                         |     |  |  |     |  |  |     |  |  |   |       |                         |    |       |                         |    |       |                         |                      |  |  |



|   |  |
|---|--|
|   |    |
| 3 | <p><b>[ Question- 41 ]</b> Write a SELECT statement to display the same results as question no. 43, skip the first 20 rows, and continue with the next 20 rows using the OFFSET-FETCH clause! Execute the statement and compare 83 - Lab Exercise 4 - Task 2 Result.txt. If the results are the same, then your test is correct.</p> <p><b>My Answer :</b></p> |



The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure for 'LAPTOP-9E916KGJ\SQL'. The main window displays a T-SQL query:

```
SELECT
    custid, orderid, orderdate
FROM
    Sales.Orders
ORDER BY
    orderdate, orderid
OFFSET 0 ROWS
FETCH NEXT 20 ROW ONLY;
```

The results pane shows the output of the query:

|    | custid | orderid | orderdate               |
|----|--------|---------|-------------------------|
| 1  | 85     | 10248   | 2006-07-04 00:00:00.000 |
| 2  | 79     | 10249   | 2006-07-05 00:00:00.000 |
| 3  | 34     | 10250   | 2006-07-08 00:00:00.000 |
| 4  | 84     | 10251   | 2006-07-08 00:00:00.000 |
| 5  | 76     | 10252   | 2006-07-09 00:00:00.000 |
| 6  | 34     | 10253   | 2006-07-10 00:00:00.000 |
| 7  | 14     | 10254   | 2006-07-11 00:00:00.000 |
| 8  | 68     | 10255   | 2006-07-12 00:00:00.000 |
| 9  | 88     | 10256   | 2006-07-15 00:00:00.000 |
| 10 | 35     | 10257   | 2006-07-16 00:00:00.000 |
| 11 | 29     | 10258   | 2006-07-17 00:00:00.000 |
| 12 | 13     | 10259   | 2006-07-18 00:00:00.000 |
| 13 | 56     | 10260   | 2006-07-19 00:00:00.000 |
| 14 | 61     | 10261   | 2006-07-19 00:00:00.000 |
| 15 | 65     | 10262   | 2006-07-22 00:00:00.000 |
| 16 | 20     | 10263   | 2006-07-23 00:00:00.000 |
| 17 | 54     | 10264   | 2006-07-24 00:00:00.000 |

Below the results, a message bar indicates: "Query executed successfully." The status bar at the bottom right shows: LAPTOP-9E916KGJ (14.0 RTM) | LAPTOP-9E916KGJ\Evan D... | TSQL | 00:00:00 | 20 rows.



|  | 83 - Lab Exercise 4 - Task 2 Result.txt |         |                         |
|--|---|---------|-------------------------|
|  | custid                                  | orderid | orderdate               |
|  | 33                                      | 10268   | 2006-07-30 00:00:00.000 |
|  | 89                                      | 10269   | 2006-07-31 00:00:00.000 |
|  | 87                                      | 10270   | 2006-08-01 00:00:00.000 |
|  | ...                                     |         |                         |
|  | ...                                     |         |                         |
|  | ...                                     |         |                         |
|  | 63                                      | 10285   | 2006-08-20 00:00:00.000 |
|  | 63                                      | 10286   | 2006-08-21 00:00:00.000 |
|  | 67                                      | 10287   | 2006-08-22 00:00:00.000 |
|  | (20 row(s) affected)                    |         |                         |

4

**Conclusion :** After working on the practical work and questions in this section, you should now understand how to use the OFFSET-FETCH clause in T-SQL commands.

-- *Have a great time doing it -*