**FOR PRESENTATION:**

**-** video of code of exercises running on slide

- keep explanation simple as possible

- solutions are in textbook

**Code in notebook:**

Northwind and Tsql

**Explanations:**

Use northwinds only

WRITE PROPOSITIONS using northwinds

---------------------------------------------------------------------

-- Microsoft SQL Server T-SQL Fundamentals

-- Chapter 03 - Joins

-- Exercises

-- © Itzik Ben-Gan

---------------------------------------------------------------------

-- 1

-- 1-1

-- Write a query that generates 5 copies out of each employee row

-- Tables involved: TSQLV4 database, Employees and Nums tables

USE TSQLV4;

SELECT empid, firstname, lastname, n

FROM HR.Employees, dbo.Nums

where n < 6

ORDER BY

n,

empid;

USE Northwinds2022TSQLV7;

SELECT EmployeeId, EmployeeFirstName, EmployeeLastName, N

FROM HumanResources.Employee, dbo.Nums

where N < 6

ORDER BY

N,

EmployeeId;

--Desired output

empid firstname lastname n

----------- ---------- -------------------- -----------

1 Sara Davis 1

2 Don Funk 1

3 Judy Lew 1

4 Yael Peled 1

5 Sven Mortensen 1

6 Paul Suurs 1

7 Russell King 1

8 Maria Cameron 1

9 Patricia Doyle 1

1 Sara Davis 2

2 Don Funk 2

3 Judy Lew 2

4 Yael Peled 2

5 Sven Mortensen 2

6 Paul Suurs 2

7 Russell King 2

8 Maria Cameron 2

9 Patricia Doyle 2

1 Sara Davis 3

2 Don Funk 3

3 Judy Lew 3

4 Yael Peled 3

5 Sven Mortensen 3

6 Paul Suurs 3

7 Russell King 3

8 Maria Cameron 3

9 Patricia Doyle 3

1 Sara Davis 4

2 Don Funk 4

3 Judy Lew 4

4 Yael Peled 4

5 Sven Mortensen 4

6 Paul Suurs 4

7 Russell King 4

8 Maria Cameron 4

9 Patricia Doyle 4

1 Sara Davis 5

2 Don Funk 5

3 Judy Lew 5

4 Yael Peled 5

5 Sven Mortensen 5

6 Paul Suurs 5

7 Russell King 5

8 Maria Cameron 5

9 Patricia Doyle 5

(45 row(s) affected)

-- 1-2 (Optional, Advanced)

-- Write a query that returns a row for each employee and day

-- in the range June 12, 2016 – June 16 2016.

-- Tables involved: TSQLV4 database, Employees and Nums tables

USE TSQLV4;

SELECT e.empid, CONVERT(DATE, DATEADD(DAY, n.n, '2016-06-12')-1) AS dt

FROM HR.Employees e

CROSS JOIN dbo.Nums n

WHERE n.n BETWEEN 0 AND DATEDIFF(DAY, '2016-06-12', '2016-06-16')+1

ORDER BY e.empid, dt;

USE Northwinds2022TSQLV7;

SELECT e.EmployeeId, CONVERT(DATE, DATEADD(DAY, n.N, '2016-06-12')-1) AS dt

FROM HumanResources.[Employee] e

CROSS JOIN dbo.[Nums] n

WHERE n.N BETWEEN 0 AND DATEDIFF(DAY, '2016-06-12', '2016-06-16')+1

ORDER BY e.EmployeeId, dt;

--Desired output

empid dt

----------- -----------

1 2016-06-12

1 2016-06-13

1 2016-06-14

1 2016-06-15

1 2016-06-16

2 2016-06-12

2 2016-06-13

2 2016-06-14

2 2016-06-15

2 2016-06-16

3 2016-06-12

3 2016-06-13

3 2016-06-14

3 2016-06-15

3 2016-06-16

4 2016-06-12

4 2016-06-13

4 2016-06-14

4 2016-06-15

4 2016-06-16

5 2016-06-12

5 2016-06-13

5 2016-06-14

5 2016-06-15

5 2016-06-16

6 2016-06-12

6 2016-06-13

6 2016-06-14

6 2016-06-15

6 2016-06-16

7 2016-06-12

7 2016-06-13

7 2016-06-14

7 2016-06-15

7 2016-06-16

8 2016-06-12

8 2016-06-13

8 2016-06-14

8 2016-06-15

8 2016-06-16

9 2016-06-12

9 2016-06-13

9 2016-06-14

9 2016-06-15

9 2016-06-16

(45 row(s) affected)

-- 2

-- Explain what’s wrong in the following query and provide a correct alternative

SELECT Customers.custid, Customers.companyname, Orders.orderid, Orders.orderdate

FROM Sales.Customers AS C

INNER JOIN Sales.Orders AS O

ON Customers.custid = Orders.custid;

ANSWER: This issue with this table is that the query provided alias for the tables 'Sales Customer' and 'Sales.Order' which are 'C' and 'O' respectively in the 'ON' clause of the 'JOIN', however instead of using the alias was not used in the 'FROM' clause resulting in a syntax error.

Use TSQLV4

SELECT C.custid, C.companyname, O.orderid, O.orderdate

FROM Sales.Customers AS C

INNER JOIN Sales.Orders AS O

ON C.custid = O.custid;

Use Northwinds2022TSQLV7

SELECT C.CustomerId, C.CustomerCompanyName, O.OrderId, O.OrderDate

FROM Sales.Customer AS C

INNER JOIN Sales.[Order] AS O

ON C.customerId = O.customerId;

-- 3

-- Return US customers, and for each customer the total number of orders

-- and total quantities.

-- Tables involved: TSQLV4 database, Customers, Orders and OrderDetails tables

Use TSQLV4

SELECT C.custid, COUNT( DISTINCT O.orderid) AS numorders, SUM(OD.qty) AS totalqty

FROM Sales.Customers AS C

INNER JOIN Sales.Orders AS O

ON O.custid = C.custid

INNER JOIN Sales.OrderDetails AS OD

ON OD.orderid = O.orderid

WHERE C.country = N'USA'

GROUP BY C.custid;

Use Northwinds2022TSQLV7

SELECT C.customerId, COUNT( DISTINCT O.orderid) AS numorders, SUM(OD.Quantity) AS totalquantity

FROM Sales.Customer AS C

INNER JOIN Sales.[Order] AS O

ON O.CustomerId = C.CustomerId

INNER JOIN Sales.OrderDetail AS OD

ON OD.orderid = O.orderid

WHERE C.CustomerCountry = N'USA'

GROUP BY C.CustomerId;

--Desired output

custid numorders totalqty

----------- ----------- -----------

32 11 345

36 5 122

43 2 20

45 4 181

48 8 134

55 10 603

65 18 1383

71 31 4958

75 9 327

77 4 46

78 3 59

82 3 89

89 14 1063

(13 row(s) affected)

-- 4

-- Return customers and their orders including customers who placed no orders

-- Tables involved: TSQLV4 database, Customers and Orders tables

use TSQLV4;

SELECT c.custid, c.companyname, o.orderid, o.orderdate

FROM Sales.Customers c

LEFT JOIN Orders o ON c.custid = o.custid;

USE Northwinds2022TSQLV7;

SELECT c.CustomerId, c.CustomerCompanyName, o.orderid, o.orderdate

FROM Sales.Customer c

LEFT JOIN Sales.[Order] o ON c.CustomerId = o.CustomerId;

-- Desired output

custid companyname orderid orderdate

----------- --------------- ----------- -----------

85 Customer ENQZT 10248 2014-07-04

79 Customer FAPSM 10249 2014-07-05

34 Customer IBVRG 10250 2014-07-08

84 Customer NRCSK 10251 2014-07-08

...

73 Customer JMIKW 11074 2016-05-06

68 Customer CCKOT 11075 2016-05-06

9 Customer RTXGC 11076 2016-05-06

65 Customer NYUHS 11077 2016-05-06

22 Customer DTDMN NULL NULL

57 Customer WVAXS NULL NULL

(832 row(s) affected)

-- 5

-- Return customers who placed no orders

-- Tables involved: TSQLV4 database, Customers and Orders tables

* Identify customers who have not placed any orders
* Display Customer ID and Company Name for such customers
* Utilize tables: Sales.Customer (referred to as c) and Sales.Order (referred to as o)
* You want to find out which customers haven't placed any orders yet. This query helps you identify those customers by displaying their Customer ID and Company Name. It involves looking at information from two tables: Customer and Order.

use TSQLV4;

SELECT c.custid, c.companyname

FROM Sales.Customers c

LEFT JOIN Orders o ON c.custid = o.custid

WHERE o.orderid IS NULL;

USE Northwinds2022TSQLV7;

SELECT c.CustomerId, c.CustomerCompanyName

FROM Sales.Customer c

LEFT JOIN Sales.[Order] o ON c.CustomerId = o.CustomerId

WHERE o.orderid IS NULL;

-- Desired output

custid companyname

----------- ---------------

22 Customer DTDMN

57 Customer WVAXS

(2 row(s) affected)

-- 6

-- Return customers with orders placed on Feb 12, 2016 along with their orders

-- Tables involved: TSQLV4 database, Customers and Orders tables

use TSQLV4;

SELECT c.custid, c.companyname, o.orderid, o.orderdate

FROM Sales.Customers c

JOIN Orders o ON c.custid = o.custid AND CONVERT(DATE, o.orderdate) = '2016-02-12';

USE Northwinds2022TSQLV7;

SELECT c.CustomerId, c.CustomerCompanyName, o.OrderId, o.OrderDate

FROM Sales.Customer c

JOIN Sales.[Order] o ON c.CustomerId = o.CustomerId AND CONVERT(DATE, o.orderdate) = '2016-02-12';

-- Desired output

custid companyname orderid orderdate

----------- --------------- ----------- ----------

48 Customer DVFMB 10883 2016-02-12

45 Customer QXPPT 10884 2016-02-12

76 Customer SFOGW 10885 2016-02-12

(3 row(s) affected)

-- 7 (Optional, Advanced)

-- Write a query that returns all customers in the output, but matches

-- them with their respective orders only if they were placed on February 12, 2016

-- Tables involved: TSQLV4 database, Customers and Orders tables

use TSQLV4;

SELECT c.custid, c.companyname, o.orderid, o.orderdate

FROM Sales.Customers c

LEFT JOIN Orders o ON c.custid = o.custid AND CONVERT(DATE, o.orderdate) = '2016-02-12';

USE Northwinds2022TSQLV7;

SELECT c.CustomerId, c.CustomerCompanyName, o.OrderId, o.OrderDate

FROM Sales.Customer c

LEFT JOIN Sales.[Order] o ON c.CustomerId = o.CustomerId AND CONVERT(DATE, o.OrderDate) = '2016-02-12';

-- Desired output

custid companyname orderid orderdate

----------- --------------- ----------- ----------

72 Customer AHPOP NULL NULL

58 Customer AHXHT NULL NULL

25 Customer AZJED NULL NULL

18 Customer BSVAR NULL NULL

91 Customer CCFIZ NULL NULL

68 Customer CCKOT NULL NULL

49 Customer CQRAA NULL NULL

24 Customer CYZTN NULL NULL

22 Customer DTDMN NULL NULL

48 Customer DVFMB 10883 2016-02-12

10 Customer EEALV NULL NULL

40 Customer EFFTC NULL NULL

85 Customer ENQZT NULL NULL

82 Customer EYHKM NULL NULL

79 Customer FAPSM NULL NULL

...

51 Customer PVDZC NULL NULL

52 Customer PZNLA NULL NULL

56 Customer QNIVZ NULL NULL

8 Customer QUHWH NULL NULL

67 Customer QVEPD NULL NULL

45 Customer QXPPT 10884 2016-02-12

7 Customer QXVLA NULL NULL

60 Customer QZURI NULL NULL

19 Customer RFNQC NULL NULL

9 Customer RTXGC NULL NULL

76 Customer SFOGW 10885 2016-02-12

69 Customer SIUIH NULL NULL

86 Customer SNXOJ NULL NULL

88 Customer SRQVM NULL NULL

54 Customer TDKEG NULL NULL

20 Customer THHDP NULL NULL

...

(91 row(s) affected)

-- 8 (Optional, Advanced)

-- Explain why the following query isn’t a correct solution query for exercise 7.

USE TSQLV4

SELECT C.custid, C.companyname, O.orderid, O.orderdate

FROM Sales.Customers AS C

LEFT OUTER JOIN Sales.Orders AS O

ON O.custid = C.custid

WHERE O.orderdate = '20160212'

OR O.orderid IS NULL;

USE Northwinds2022TSQLV7

SELECT C.CustomerId, C.CustomerCompanyName, O.OrderId, O.OrderDate

FROM Sales.Customer AS C

LEFT OUTER JOIN Sales.[Order] AS O

ON O.CustomerId = C.CustomerId

WHERE O.OrderDate = '20160212'

OR O.OrderId IS NULL;

--ANSWER: The use of where clause here will not display customers who have not

--made any orders

-- 9 (Optional, Advanced)

-- Return all customers, and for each return a Yes/No value

-- depending on whether the customer placed an order on Feb 12, 2016

-- Tables involved: TSQLV4 database, Customers and Orders tables

* Determine if customers made orders on a specific date
* Display Customer ID, Company Name, and indicate whether they placed orders on '2016-02-12'
* Utilize tables: Sales.Customer (referred to as c) and Sales.Order (referred to as o)
* You want to find out if customers made orders on a particular date. This query helps you achieve that by displaying Customer ID, Company Name, and a clear indication of whether they placed orders on '2016-02-12'. It looks at data from two tables: Customer and Order.

use TSQLV4;

SELECT c.custid, c.companyname,

CASE WHEN EXISTS (

SELECT 1

FROM Orders o

WHERE c.custid = o.custid

AND CONVERT(DATE, o.orderdate) = '2016-02-12'

) THEN 'Yes' ELSE 'No' END AS HasOrderOn20160212

FROM Sales.Customers c;

USE Northwinds2022TSQLV7;

SELECT c.CustomerId, c.CustomerCompanyName,

CASE WHEN EXISTS (

SELECT 1

FROM Sales.[Order] o

WHERE c.CustomerId = o.CustomerId

AND CONVERT(DATE, o.orderdate) = '2016-02-12'

) THEN 'Yes' ELSE 'No' END AS HasOrderOn20160212

FROM Sales.Customer c;

-- Desired output

custid companyname HasOrderOn20160212

----------- --------------- ------------------

...

40 Customer EFFTC No

41 Customer XIIWM No

42 Customer IAIJK No

43 Customer UISOJ No

44 Customer OXFRU No

45 Customer QXPPT Yes

46 Customer XPNIK No

47 Customer PSQUZ No

48 Customer DVFMB Yes

49 Customer CQRAA No

50 Customer JYPSC No

51 Customer PVDZC No

52 Customer PZNLA No

53 Customer GCJSG No

...

(91 row(s) affected)