CROSS JOINS

```
1   USE Northwinds2022TSQLV7
2
3   SELECT C.CustomerId, E.EmployeeId
4   FROM Sales.Customer AS C
5       CROSS JOIN HumanResources.Employee AS E;
```

(819 rows affected)

Total execution time: 00:00:00.012

| | CustomerId | EmployeeId |
|---|---|---|
| 81 | 81 | 1 |
| 82 | 82 | 1 |
| 83 | 83 | 1 |
| 84 | 84 | 1 |
| 85 | 85 | 1 |
| 86 | 86 | 1 |
| 87 | 87 | 1 |
| 88 | 88 | 1 |
| 89 | 89 | 1 |
| 90 | 90 | 1 |
| 91 | 91 | 1 |
| 92 | 1 | 2 |
| 93 | 2 | 2 |
| 94 | 3 | 2 |
| 95 | 4 | 2 |
| 96 | 5 | 2 |
| 97 | 6 | 2 |

SELF CROSS JOINS

```
1   USE Northwinds2022TSQLV7
2   SELECT E1.EmployeeId, E2.EmployeeId
3   FROM HumanResources.Employee AS E1
4       CROSS JOIN HumanResources.Employee AS E2;
```

(81 rows affected)

Total execution time: 00:00:00.011

| | EmployeeId | EmployeeId |
|---|---|---|
| 1 | 1 | 1 |
| 2 | 2 | 1 |
| 3 | 3 | 1 |
| 4 | 4 | 1 |
| 5 | 5 | 1 |
| 6 | 6 | 1 |
| 7 | 7 | 1 |
| 8 | 8 | 1 |
| 9 | 9 | 1 |
| 10 | 1 | 2 |
| 11 | 2 | 2 |
| 12 | 3 | 2 |
| 13 | 4 | 2 |
| 14 | 5 | 2 |
| 15 | 6 | 2 |
| 16 | 7 | 2 |

Producing Tables of Numbers (Application of self cross join)

```
[7]
1  USE Northwinds2022TSQLV7
2  DROP TABLE IF EXISTS dbo.Digit;
3  CREATE TABLE dbo.Digit (digit INT NOT NULL PRIMARY KEY);
4  INSERT INTO dbo.Digit (digit)
5      VALUES (0), (1), (2), (3), (4), (5), (6), (7), (8), (9);
6
7  SELECT digit FROM dbo.Digit;
8
9  SELECT D3.digit * 100 + D2.digit * 10 + D1.digit + 1 AS n -- multiply by power of 10. 100, 10, digit table itself (ones). +1 is to discard 0.
10 FROM dbo.Digit AS D1
11     CROSS JOIN dbo.Digit AS D2
12     CROSS JOIN dbo.digit AS D3
13 ORDER BY n;
14
```

```
(10 rows affected)
(10 rows affected)
(1000 rows affected)
Total execution time: 00:00:00.041
```

| | digit |
|---|---|
| 1 | 0 |
| 2 | 1 |
| 3 | 2 |
| 4 | 3 |
| 5 | 4 |
| 6 | 5 |
| 7 | 6 |
| 8 | 7 |

| | |
|---|---|
| 982 | 982 |
| 983 | 983 |
| 984 | 984 |
| 985 | 985 |
| 986 | 986 |
| 987 | 987 |
| 988 | 988 |
| 989 | 989 |
| 990 | 990 |
| 991 | 991 |
| 992 | 992 |
| 993 | 993 |
| 994 | 994 |
| 995 | 995 |
| 996 | 996 |
| 997 | 997 |
| 998 | 998 |
| 999 | 999 |
| 10… | 1000 |

INNER JOINS OR JOINS

```sql
USE Northwinds2022TSQLV7
SELECT E.EmployeeId, E.EmployeeFirstName, E.EmployeeLastName, O.OrderId
FROM HumanResources.Employee AS E
    INNER JOIN SALES.[Order] AS O
        ON E.EmployeeId = O.EmployeeId
```

(830 rows affected)

Total execution time: 00:00:00.018

| | EmployeeId | EmployeeFirstName | EmployeeLastName | OrderId |
|---|---|---|---|---|
| 1 | 5 | Sven | Mortensen | 10248 |
| 2 | 6 | Paul | Suurs | 10249 |
| 3 | 4 | Yael | Peled | 10250 |
| 4 | 3 | Judy | Lew | 10251 |
| 5 | 4 | Yael | Peled | 10252 |
| 6 | 3 | Judy | Lew | 10253 |
| 7 | 5 | Sven | Mortensen | 10254 |
| 8 | 9 | Patricia | Doyle | 10255 |

COMPOSITE JOINS

```sql
--In the following table creation, there are two columns in Sales.ORderDetailAudit that are also present in Sales.OrderDetail: orderid
USE Northwinds2022TSQLV7
DROP TABLE IF EXIstS Sales.OrderDetailsAudit;

CREATE TABLE Sales.OrderDetailsAudit(
    lsn INT NOT NULL IDENTITY,
    orderid INT NOT NULL,
    productid INT NOT NULL,
    dt DATETIME NOT NULL,
    loginname sysname NOT NULL,
    columnname sysname NOT NULL,
    oldval SQL_VARIANT,
    newval SQL_VARIANT,
    CONSTRAINT PK_OrderDetailsAudit PRIMARY KEY (lsn),
    CONSTRAINT FK_OrderDetailsAudit_OrderDetails
        FOREIGN KEY(orderid, productid)
        REFERENCES Sales.OrderDetail (OrderId, ProductId)
);

SELECT OD.OrderId, OD.ProductId, OD.Quantity,
    ODA.dt, ODA.loginname, ODA.oldval, ODA.newval
FROM Sales.OrderDetail AS OD
    INNER JOIN Sales.OrderDetailsAudit AS ODA
        ON OD.orderid = ODA.orderid
        AND OD.productid = ODA.productid
WHERE ODA.columnname = N'qty';
```

(0 rows affected)

Total execution time: 00:00:00.029

| OrderId | ProductId | Quantity | dt | loginname | oldval | newval |
|---|---|---|---|---|---|---|

NON-EQUI JOINS

```sql
USE Northwinds2022TSQLV7
SELECT
    E1.EmployeeId, E1.EmployeeFirstName, E1.EmployeeLastName,
    E2.EmployeeId, E2.EmployeeFirstName, E2.EmployeeLastname
FROM HumanResources.Employee AS E1
    INNER JOIN HumanResources.Employee AS E2
        ON E1.EmployeeId < E2.EmployeeId
ORDER BY E1.EmployeeId;
```

(36 rows affected)

Total execution time: 00:00:00.007

| | EmployeeId | EmployeeFirstName | EmployeeLastName | EmployeeId | EmployeeFirstName | EmployeeLastname |
|---|---|---|---|---|---|---|
| 1 | 1 | Sara | Davis | 2 | Don | Funk |
| 2 | 1 | Sara | Davis | 3 | Judy | Lew |
| 3 | 1 | Sara | Davis | 4 | Yael | Peled |
| 4 | 1 | Sara | Davis | 5 | Sven | Mortensen |
| 5 | 1 | Sara | Davis | 6 | Paul | Suurs |
| 6 | 1 | Sara | Davis | 7 | Russell | King |
| 7 | 1 | Sara | Davis | 8 | Maria | Cameron |
| 8 | 1 | Sara | Davis | 9 | Patricia | Doyle |
| 9 | 2 | Don | Funk | 3 | Judy | Lew |
| 10 | 2 | Don | Funk | 4 | Yael | Peled |
| 11 | 2 | Don | Funk | 5 | Sven | Mortensen |
| 12 | 2 | Don | Funk | 6 | Paul | Suurs |
| 13 | 2 | Don | Funk | 7 | Russell | King |
| 14 | 2 | Don | Funk | 8 | Maria | Cameron |

MULTI- JOIN QUERIES

```sql
USE Northwinds2022TSQLV7
SELECT
    C.CustomerId, C.CustomerCompanyName,
    O.OrderId,
    OD.ProductId, OD.Quantity
FROM Sales.Customer AS C
    INNER JOIN Sales.[Order] AS O
        ON C.CustomerId = O.CustomerId
    INNER JOIN Sales.OrderDetail AS OD
        ON O.OrderId = OD.OrderId;
```

(2155 rows affected)

Total execution time: 00:00:00.027

| | CustomerId | CustomerCompanyName | OrderId | ProductId | Quantity |
|---|---|---|---|---|---|
| 16 | 34 | Customer IBVRG | 10253 | 39 | 42 |
| 17 | 34 | Customer IBVRG | 10253 | 49 | 40 |
| 18 | 14 | Customer WNMAF | 10254 | 24 | 15 |
| 19 | 14 | Customer WNMAF | 10254 | 55 | 21 |
| 20 | 14 | Customer WNMAF | 10254 | 74 | 21 |
| 21 | 68 | Customer CCKOT | 10255 | 2 | 20 |

0

OUTER JOIN

```sql
USE Northwinds2022TSQLV7
SELECT C.CustomerId,C.CustomerCompanyName,
    O.orderid
FROM Sales.Customer AS C
    LEFT OUTER JOIN Sales.[Order] as O
        ON C.CustomerId = O.CustomerId
ORDER BY C.CustomerID;

--Notice that the output includes NULL under OrderId.

--The following only identifies the customers with no orders:
SELECT C.CustomerId,C.CustomerCompanyName,
    O.orderid
FROM Sales.Customer AS C
    LEFT OUTER JOIN Sales.[Order] as O
        ON C.CustomerId = O.CustomerId
WHERE O.OrderId IS NULL;

```

(832 rows affected)

(2 rows affected)

Total execution time: 00:00:00.041

| | CustomerId | CustomerCompanyName | orderid |
|---|---|---|---|
| 185 | 21 | Customer KIDPX | 10512 |
| 186 | 21 | Customer KIDPX | 10650 |
| 187 | 21 | Customer KIDPX | 10581 |
| 188 | 21 | Customer KIDPX | 10725 |
| 189 | 22 | Customer DTDMN | *NULL* |
| 190 | 23 | Customer WVFAF | 10763 |
| 191 | 23 | Customer WVFAF | 10789 |
| 192 | 23 | Customer WVFAF | 10634 |
| 193 | 23 | Customer WVFAF | 10480 |
| 194 | 23 | Customer WVFAF | 10408 |
| 195 | 24 | Customer CYZTN | 10434 |
| 196 | 24 | Customer CYZTN | 10460 |
| 197 | 24 | Customer CYZTN | 10533 |
| 198 | 24 | Customer CYZTN | 10378 |
| 199 | 24 | Customer CYZTN | 10327 |

| | CustomerId | CustomerCompanyName | orderid |
|---|---|---|---|
| 1 | 22 | Customer DTDMN | *NULL* |
| 2 | 57 | Customer WVAXS | *NULL* |

Using outer joins to identify and include the missing values

```sql
--This query returns a sequence of all dates in the range Jan 1, 2014 through Dec
USE Northwinds2022TSQLV7
SELECT DATEADD (day, n-1, CAST('20140101' AS DATE)) AS orderdate -- From Jan 1st 2
FROM dbo.Nums
WHERE n <= DATEDIFF (day, '20140101', '20161231') +1--this calculates how many tim
ORDER BY orderdate;
--this is like a forloop. n is the number of iterations.

--Then outer join between the nums and the orders.

SELECT DATEADD (day, Nums.n-1, CAST('20140101' AS DATE)) AS orderdate,
    O.orderId, O.CustomerId, O.EmployeeId
FROM dbo.Nums
    LEFT OUTER JOIN Sales.[Order] AS O
        ON DATEADD (day, Nums.n-1, CAST('20140101' AS DATE)) = O.orderdate;
```

(1096 rows affected)

(100350 rows affected)

Displaying Top 5000 rows.

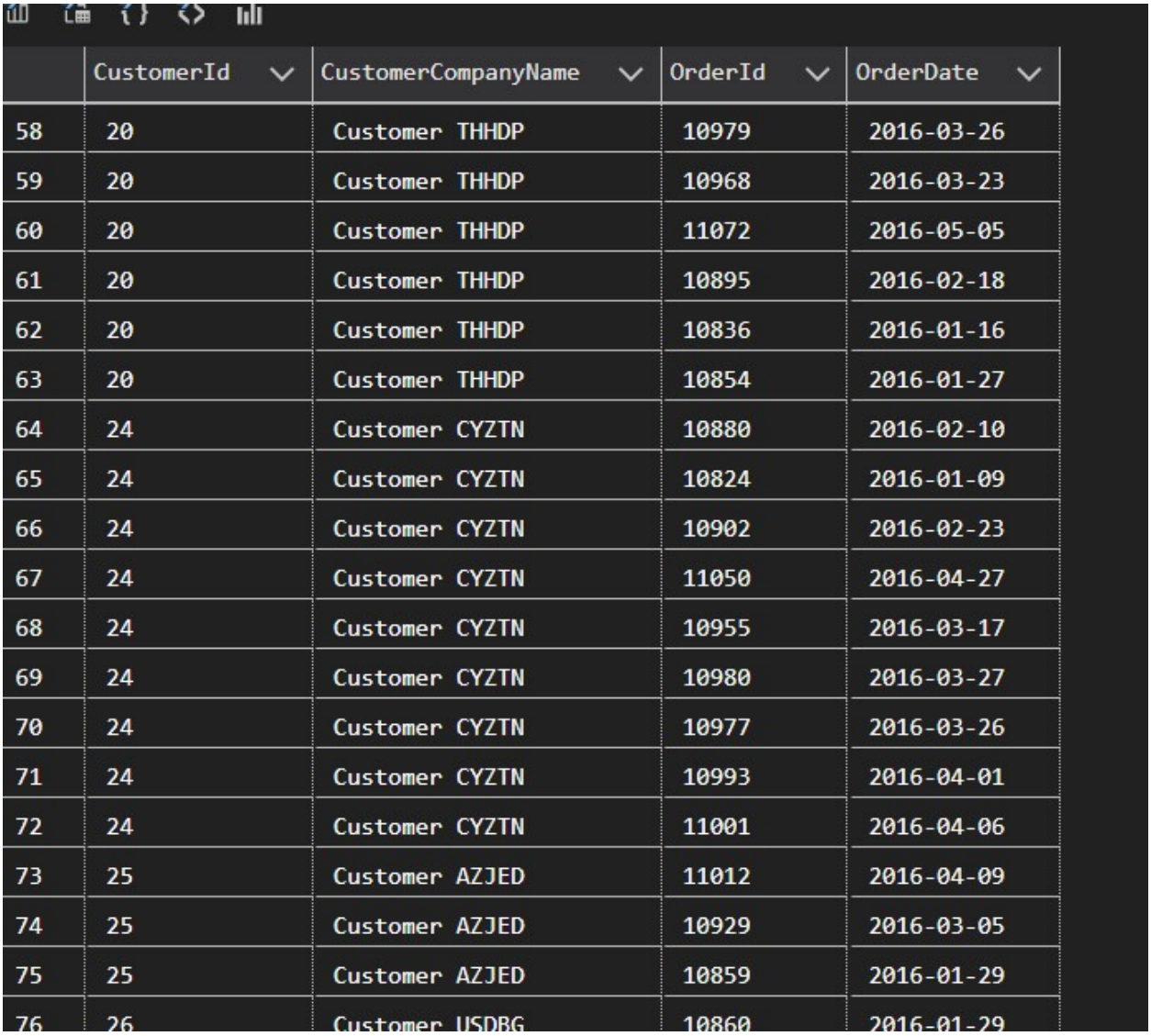
Total execution time: 00:00:00.365

| | orderdate |
|---|---|
| 1 | 2014-01-01 |
| 2 | 2014-01-02 |
| 3 | 2014-01-03 |
| 4 | 2014-01-04 |
| 5 | 2014-01-05 |
| 6 | 2014-01-06 |
| 7 | 2014-01-07 |
| 8 | 2014-01-08 |
| 9 | 2014-01-09 |
| 10 | 2014-01-10 |
| 11 | 2014-01-11 |
| 12 | 2014-01-12 |
| 13 | 2014-01-13 |

| | orderdate | orderId | CustomerId | EmployeeId |
|---|---|---|---|---|
| 1 | 2014-01-01 | NULL | NULL | NULL |
| 2 | 2014-01-02 | NULL | NULL | NULL |
| 3 | 2014-01-03 | NULL | NULL | NULL |
| 4 | 2014-01-04 | NULL | NULL | NULL |
| 5 | 2014-01-05 | NULL | NULL | NULL |
| 6 | 2014-01-06 | NULL | NULL | NULL |
| 7 | 2014-01-07 | NULL | NULL | NULL |
| 8 | 2014-01-08 | NULL | NULL | NULL |
| 9 | 2014-01-09 | NULL | NULL | NULL |
| 10 | 2014-01-10 | NULL | NULL | NULL |
| 11 | 2014-01-11 | NULL | NULL | NULL |
| 12 | 2014-01-12 | NULL | NULL | NULL |
| 13 | 2014-01-13 | NULL | NULL | NULL |
| 14 | 2014-01-14 | NULL | NULL | NULL |
| 15 | 2014-01-15 | NULL | NULL | NULL |
| 16 | 2014-01-16 | NULL | NULL | NULL |
| 17 | 2014-01-17 | NULL | NULL | NULL |

OUTER JOINS: MISTAKE TO AVOID

```
1   USE Northwinds2022TSQLV7
2   SELECT C.CustomerId, C.CustomerCompanyName, O.OrderId, O.OrderDate
3   FROM Sales.Customer AS C
4       LEFT Outer JOIN Sales.[Order] AS O
5           ON C.CustomerId = O.CustomerId
6   WHERE O.OrderDate >= '20160101';
```



| | CustomerId | CustomerCompanyName | OrderId | OrderDate |
|---|---|---|---|---|
| 58 | 20 | Customer THHDP | 10979 | 2016-03-26 |
| 59 | 20 | Customer THHDP | 10968 | 2016-03-23 |
| 60 | 20 | Customer THHDP | 11072 | 2016-05-05 |
| 61 | 20 | Customer THHDP | 10895 | 2016-02-18 |
| 62 | 20 | Customer THHDP | 10836 | 2016-01-16 |
| 63 | 20 | Customer THHDP | 10854 | 2016-01-27 |
| 64 | 24 | Customer CYZTN | 10880 | 2016-02-10 |
| 65 | 24 | Customer CYZTN | 10824 | 2016-01-09 |
| 66 | 24 | Customer CYZTN | 10902 | 2016-02-23 |
| 67 | 24 | Customer CYZTN | 11050 | 2016-04-27 |
| 68 | 24 | Customer CYZTN | 10955 | 2016-03-17 |
| 69 | 24 | Customer CYZTN | 10980 | 2016-03-27 |
| 70 | 24 | Customer CYZTN | 10977 | 2016-03-26 |
| 71 | 24 | Customer CYZTN | 10993 | 2016-04-01 |
| 72 | 24 | Customer CYZTN | 11001 | 2016-04-06 |
| 73 | 25 | Customer AZJED | 11012 | 2016-04-09 |
| 74 | 25 | Customer AZJED | 10929 | 2016-03-05 |
| 75 | 25 | Customer AZJED | 10859 | 2016-01-29 |
| 76 | 26 | Customer USDBG | 10860 | 2016-01-29 |

USING OUTER JOINS IN A MULTI-JOIN QUERY

```
1   --The following query is bad becaues of the reason above.
2   USE Northwinds2022TSQLV7
3   SELECT C.CustomerId, O.OrderId, OD.ProductId, OD.Quantity
4   FROM Sales.Customer AS C
5       LEFT OUTER JOIN Sales.[Order] AS O
6           ON C.CustomerId = O.CustomerId
7       INNER JOIN Sales.OrderDetail AS OD
8           ON O.OrderId = Od.OrderId;
9
```

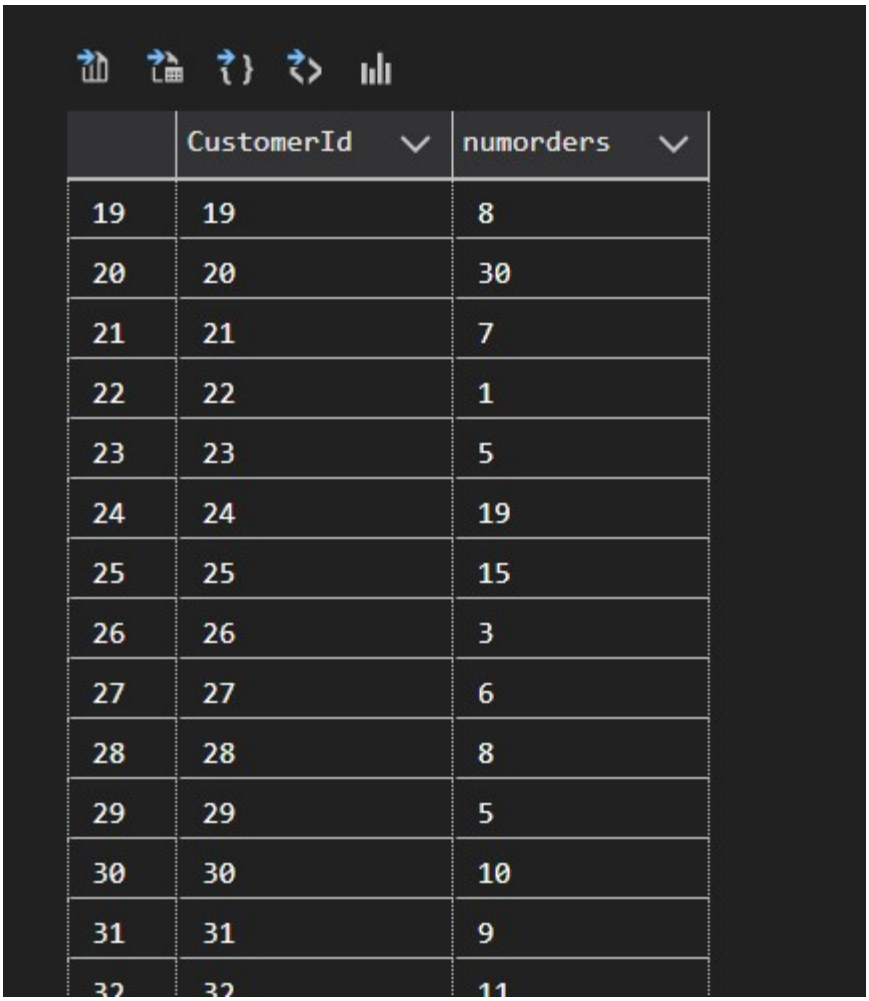
```
--Work- around 1 (both left outer joins): This might not be good.
SELECT C.CustomerId, O.OrderId, OD.ProductId, OD.Quantity
FROM Sales.Customer AS C
    LEFT OUTER JOIN Sales.[Order] AS O
        ON C.CustomerId = O.CustomerId
    LEFT OUTER JOIN Sales.OrderDetail AS OD
        ON O.OrderId = Od.OrderId;

--Work- around 2: inner join before the outer join. Inner join be
SELECT C.CustomerId, O.OrderId, OD.ProductId, OD.Quantity
FROM Sales.[Order] AS O
    INNER JOIN Sales.OrderDetail AS OD
        ON O.OrderId = Od.OrderId
    RIGHT OUTER JOIN Sales.[Order] AS C
        ON C.CustomerId = O.CustomerId;
--Work- around 3: Use parenthesis. Similar to the work-around 2.
SELECT C.CustomerId, O.OrderId, OD.ProductId, OD.Quantity
FROM Sales.Customer AS C
    LEFT OUTER JOIN
        (Sales.[Order] AS O
            INNER JOIN Sales.OrderDetail AS OD
                ON O.OrderID = OD.OrderId)
    ON O.CustomerId = C.CustomerId;
```

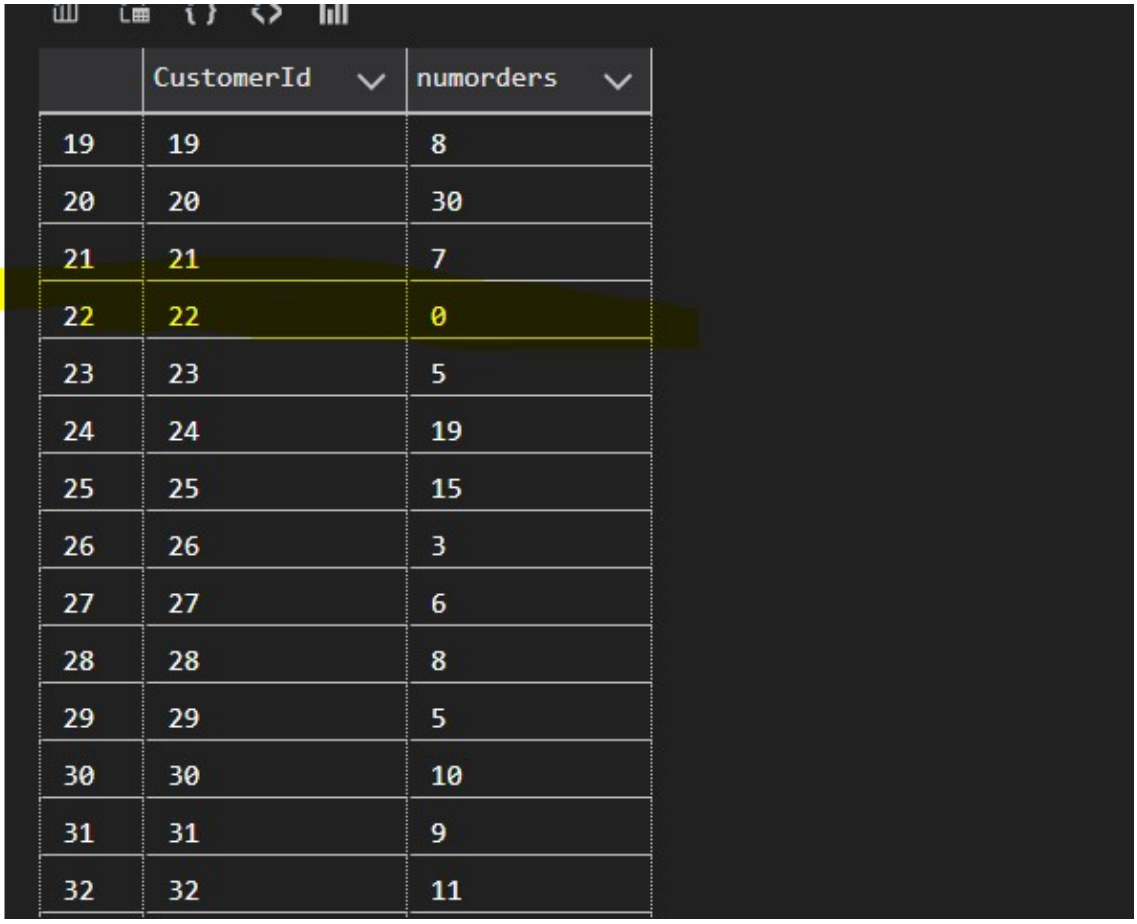
USING THE COUNT AGGREGATE WITH OUTER JOINS

```sql
--The following query is supposed to return the count of
USE Northwinds2022TSQLV7
SELECT C.CustomerId, COUNT(*) AS numorders
FROM Sales.Customer AS C
    LEFT OUTER JOIN Sales.[Order] AS O
        ON C.CustomerId = O.CustomerId
GROUP BY C.CustomerId;

--Rewrite this query like this:
SELECT C.CustomerId, COUNT(O.OrderId) AS numorders
FROM Sales.Customer AS C
    LEFT OUTER JOIN Sales.[Order] AS O
        ON C.CustomerId = O.CustomerId
GROUP BY C.CustomerId;
```



| | CustomerId | numorders |
|---|---|---|
| 19 | 19 | 8 |
| 20 | 20 | 30 |
| 21 | 21 | 7 |
| 22 | 22 | 1 |
| 23 | 23 | 5 |
| 24 | 24 | 19 |
| 25 | 25 | 15 |
| 26 | 26 | 3 |
| 27 | 27 | 6 |
| 28 | 28 | 8 |
| 29 | 29 | 5 |
| 30 | 30 | 10 |
| 31 | 31 | 9 |
| 32 | 32 | 11 |



| | CustomerId | numorders |
|---|---|---|
| 19 | 19 | 8 |
| 20 | 20 | 30 |
| 21 | 21 | 7 |
| 22 | 22 | 0 |
| 23 | 23 | 5 |
| 24 | 24 | 19 |
| 25 | 25 | 15 |
| 26 | 26 | 3 |
| 27 | 27 | 6 |
| 28 | 28 | 8 |
| 29 | 29 | 5 |
| 30 | 30 | 10 |
| 31 | 31 | 9 |
| 32 | 32 | 11 |