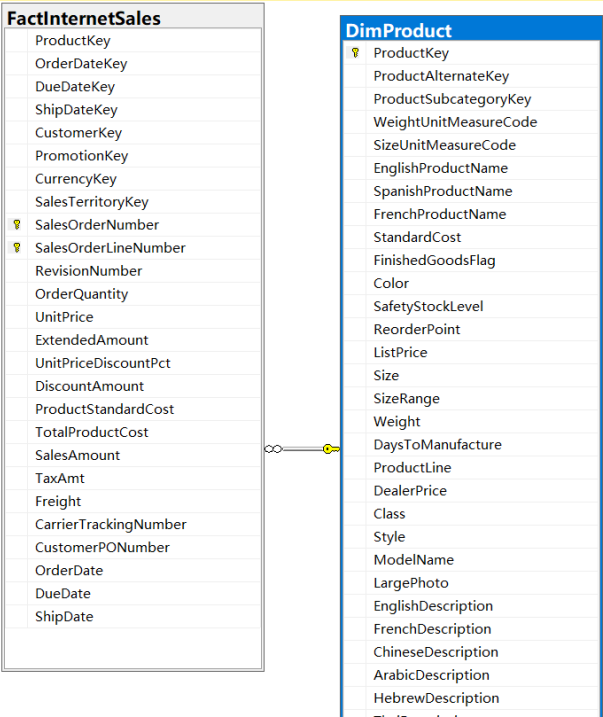```sql
/*
USE AdventureWorksDW2017
return the information of the fact internet sales that after 20110101
*/
USE AdventureWorksDW2017;
SELECT FIS.SalesOrderNumber,
    DP.EnglishProductName,
    FIS.UnitPrice,
    FIS.OrderQuantity
FROM dbo.FactInternetSales AS FIS
    LEFT OUTER JOIN dbo.DimProduct AS DP
        ON DP.ProductKey = FIS.ProductKey
WHERE FIS.OrderDate >= '20110101'
ORDER BY FIS.SalesOrderNumber;
```

Results | Messages

| | SalesOrderNumber | EnglishProductName | UnitPrice | OrderQuantity |
|---|---|---|---|---|
| 52 | SO43762 | Road-150 Red, 56 | 3578.27 | 1 |
| 53 | SO43763 | Road-150 Red, 52 | 3578.27 | 1 |
| 54 | SO43764 | Road-150 Red, 62 | 3578.27 | 1 |
| 55 | SO43765 | Mountain-100 Silve... | 3399.99 | 1 |
| 56 | SO43766 | Road-150 Red, 52 | 3578.27 | 1 |
| 57 | SO43767 | Mountain-100 Blac... | 3374.99 | 1 |
| 58 | SO43768 | Road-150 Red, 48 | 3578.27 | 1 |
| 59 | SO43769 | Road-150 Red, 48 | 3578.27 | 1 |
| 60 | SO43770 | Road-150 Red, 48 | 3578.27 | 1 |
| 61 | SO43771 | Road-650 Red, 52 | 699.0982 | 1 |
| 62 | SO43772 | Road-150 Red, 48 | 3578.27 | 1 |
| 63 | SO43773 | Road-150 Red, 56 | 3578.27 | 1 |
| 64 | SO43774 | Road-150 Red, 52 | 3578.27 | 1 |
| 65 | SO43775 | Mountain-100 Silve... | 3399.99 | 1 |
| 66 | SO43776 | Road-150 Red, 56 | 3578.27 | 1 |
| 67 | SO43777 | Road-150 Red, 44 | 3578.27 | 1 |

```
(60384 rows affected)

Completion time: 2021-10-11T13:10:09.1786552-04:00
```

FactInternetSales

- ProductKey
- OrderDateKey
- DueDateKey
- ShipDateKey
- CustomerKey
- PromotionKey
- CurrencyKey
- SalesTerritoryKey
- 🔑 SalesOrderNumber
- 🔑 SalesOrderLineNumber
- RevisionNumber
- OrderQuantity
- UnitPrice
- ExtendedAmount
- UnitPriceDiscountPct
- DiscountAmount
- ProductStandardCost
- TotalProductCost
- SalesAmount
- TaxAmt
- Freight
- CarrierTrackingNumber
- CustomerPONumber
- OrderDate
- DueDate
- ShipDate

DimProduct

- 🔑 ProductKey
- ProductAlternateKey
- ProductSubcategoryKey
- WeightUnitMeasureCode
- SizeUnitMeasureCode
- EnglishProductName
- SpanishProductName
- FrenchProductName
- StandardCost
- FinishedGoodsFlag
- Color
- SafetyStockLevel
- ReorderPoint
- ListPrice
- Size
- SizeRange
- Weight
- DaysToManufacture
- ProductLine
- DealerPrice
- Class
- Style
- ModelName
- LargePhoto
- EnglishDescription
- FrenchDescription
- ChineseDescription
- ArabicDescription
- HebrewDescription

```
/*
Return UK customers, and for each customer return the total number of orders and total
quantities
*/
USE Northw;
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'UK'
GROUP BY C.CustomerId;
```
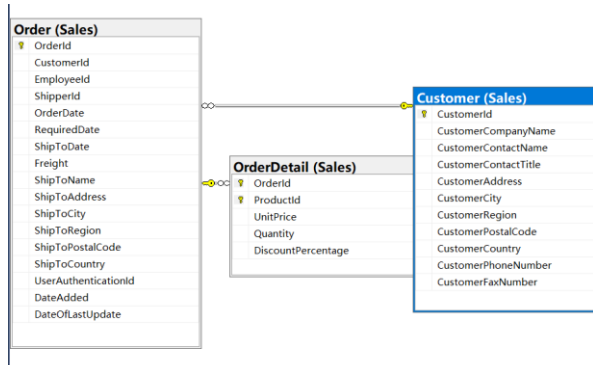
100 %

▦ Results | ▤ Messages

	CustomerId	numorders	totalQuantity
1	4	13	650
2	11	10	293
3	16	3	87
4	19	8	569
5	38	10	295
6	53	3	30
7	72	9	818

```
(7 rows affected)

Completion time: 2021-10-11T11:04:50.4696681-04:00
```


Order (Sales)
- OrderId
- CustomerId
- EmployeeId
- ShipperId
- OrderDate
- RequiredDate
- ShipToDate
- Freight
- ShipToName
- ShipToAddress
- ShipToCity
- ShipToRegion
- ShipToPostalCode
- ShipToCountry
- UserAuthenticationId
- DateAdded
- DateOfLastUpdate

OrderDetail (Sales)
- OrderId
- ProductId
- UnitPrice
- Quantity
- DiscountPercentage

Customer (Sales)
- CustomerId
- CustomerCompanyName
- CustomerContactName
- CustomerContactTitle
- CustomerAddress
- CustomerCity
- CustomerRegion
- CustomerPostalCode
- CustomerCountry
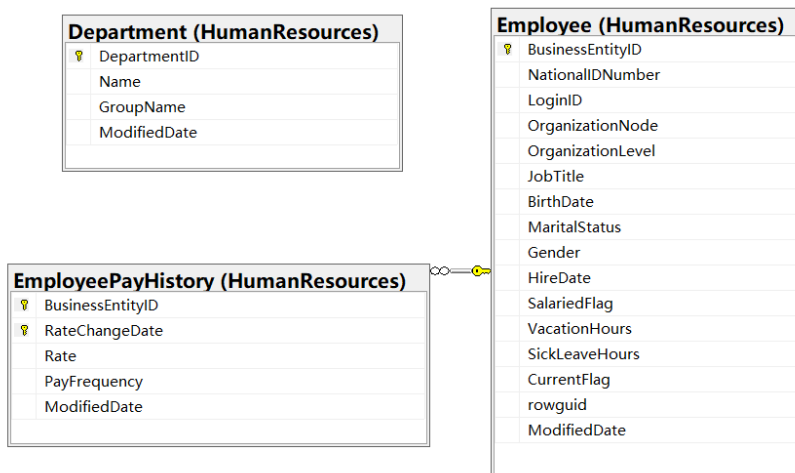- CustomerPhoneNumber
- CustomerFaxNumber

```sql
/*
USE AdventureWorks2017
return the information of  the employees including the BusinessEntityID,BirthDate,Gender,HireDate, and DepartmentID
*/
USE AdventureWorks2017;
SELECT E.BusinessEntityID,
       E.BirthDate,
       E.Gender,
       E.HireDate,
       HD.DepartmentID
FROM HumanResources.Employee AS E
    INNER JOIN HumanResources.EmployeeDepartmentHistory AS ED
        ON E.BusinessEntityID = ED.BusinessEntityID
    INNER JOIN HumanResources.Department AS HD
        ON ED.DepartmentID = HD.DepartmentID;
```

Results | Messages

	BusinessEntityID	BirthDate	Gender	HireDate	DepartmentID
1	217	1976-01-30	M	2009-01-04	12
2	218	1990-04-28	M	2008-12-16	12
3	219	1987-03-12	M	2009-01-22	12
4	220	1975-12-25	F	2009-02-09	12
5	221	1987-05-26	M	2009-03-06	12
6	2	1971-08-01	F	2008-01-31	1
7	3	1974-11-12	M	2007-11-11	1
8	4	1974-12-23	M	2007-12-05	1
9	5	1952-09-27	F	2008-01-06	1
10	6	1959-03-11	M	2008-01-24	1
11	14	1979-06-16	M	2010-12-30	1
12	15	1961-05-02	F	2011-01-18	1
13	1	1969-01-29	M	2009-01-14	16

```
(296 rows affected)

Completion time: 2021-10-11T11:39:25.8815357-04:00
```

Department (HumanResources)
- DepartmentID
- Name
- GroupName
- ModifiedDate

Employee (HumanResources)
- BusinessEntityID
- NationalIDNumber
- LoginID
- OrganizationNode
- OrganizationLevel
- JobTitle
- BirthDate
- MaritalStatus
- Gender
- HireDate
- SalariedFlag
- VacationHours
- SickLeaveHours
- CurrentFlag
- rowguid
- ModifiedDate

EmployeePayHistory (HumanResources)
- BusinessEntityID
- RateChangeDate
- Rate
- PayFrequency
- ModifiedDate

```sql
/*
looking for the costomers who have orders and their name has c a and d
*/
USE WideWorldImporters;
SELECT DISTINCT
    C.CustomerID,
    C.CustomerName
FROM Sales.Customers AS C
    LEFT OUTER JOIN Sales.Orders AS o
        ON C.CustomerID = o.CustomerID
WHERE C.CustomerName LIKE '%c%a%d%'
ORDER BY C.CustomerID;
```
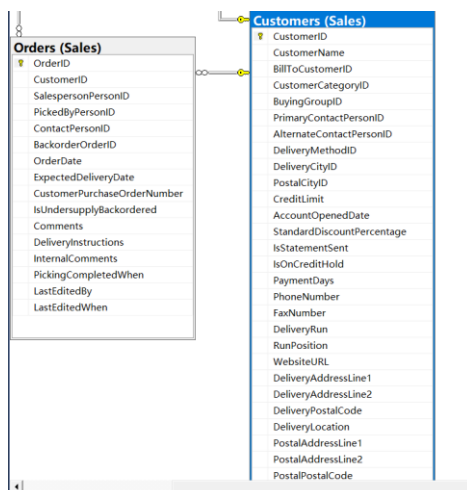
100 %

Results | Messages

	CustomerID	CustomerName
1	16	Tailspin Toys (Coney Island, MO)
2	89	Tailspin Toys (Victory Gardens, NJ)
3	160	Tailspin Toys (Crary, ND)
4	191	Tailspin Toys (Orchard Hill, GA)
5	437	Wingtip Toys (Cadogan, PA)
6	454	Wingtip Toys (Corcovado, PR)
7	462	Wingtip Toys (Crossroads, NM)
8	501	Wingtip Toys (Cavendish, VT)
9	504	Wingtip Toys (Cape Neddick, ME)
10	547	Wingtip Toys (Chaseley, ND)
11	590	Wingtip Toys (McAdenville, NC)
12	802	Cosmina Vlad

Results | Messages

```
(17 rows affected)

Completion time: 2021-10-11T13:53:09.3882801-04:00
```

Orders (Sales)
- OrderID
- CustomerID
- SalespersonPersonID
- PickedByPersonID
- ContactPersonID
- BackorderOrderID
- OrderDate
- ExpectedDeliveryDate
- CustomerPurchaseOrderNumber
- IsUndersupplyBackordered
- Comments
- DeliveryInstructions
- InternalComments
- PickingCompletedWhen
- LastEditedBy
- LastEditedWhen

Customers (Sales)
- CustomerID
- CustomerName
- BillToCustomerID
- CustomerCategoryID
- BuyingGroupID
- PrimaryContactPersonID
- AlternateContactPersonID
- DeliveryMethodID
- DeliveryCityID
- PostalCityID
- CreditLimit
- AccountOpenedDate
- StandardDiscountPercentage
- IsStatementSent
- IsOnCreditHold
- PaymentDays
- PhoneNumber
- FaxNumber
- DeliveryRun
- RunPosition
- WebsiteURL
- DeliveryAddressLine1
- DeliveryAddressLine2
- DeliveryPostalCode
- DeliveryLocation
- PostalAddressLine1
- PostalAddressLine2
- PostalPostalCode

```sql
USE WideWorldImportersDW;
SELECT s.Description,
    s.[Sale Key],
    s.[Tax Amount],
    s.Quantity
FROM Fact.Sale AS s
    LEFT OUTER JOIN Dimension.Customer AS c
        ON c.[Customer Key] = s.[Customer Key]
WHERE s.Quantity >= 5
ORDER BY s.Quantity;
--FOR JSON PATH ,ROOT('5')
----------------------------------------------------------------
```

	Description	Sale Key	Tax Amount	Quantity
1	USB missile launcher (Green)	216407	18.75	5
2	RC toy sedan car with remote control (Black) 1/50 sc...	216563	18.75	5
3	Developer joke mug - when your hammer is C++ (Bl...	106952	9.75	5
4	Superhero action jacket (Blue) XS	216822	18.75	5
5	RC toy sedan car with remote control (Yellow) 1/50 s...	217191	18.75	5
6	Superhero action jacket (Blue) XXS	217572	18.75	5
7	DBA joke mug - I will get you in order (White)	131149	9.75	5
8	DBA joke mug - SELECT caffeine FROM mug (White)	106777	9.75	5
9	USB missile launcher (Green)	217592	18.75	5
10	RC toy sedan car with remote control (Black) 1/50 sc...	218199	18.75	5
11	Superhero action jacket (Blue) 3XS	218633	18.75	5
12	USB rocket launcher (Gray)	218640	18.75	5

Sale (Fact)
- [Sale Key]
- [City Key]
- [Customer Key]
- [Bill To Customer Key]
- [Stock Item Key]
- [Invoice Date Key]
- [Delivery Date Key]
- [Salesperson Key]
- [WWI Invoice ID]
- Description
- Package
- Quantity
- [Unit Price]
- [Tax Rate]
- [Total Excluding Tax]
- [Tax Amount]
- Profit
- [Total Including Tax]
- [Total Dry Items]
- [Total Chiller Items]
- [Lineage Key]

Customer (Dimension)
- [Customer Key]
- [WWI Customer ID]
- Customer
- [Bill To Customer]
- Category
- [Buying Group]
- [Primary Contact]
- [Postal Code]
- [Valid From]
- [Valid To]
- [Lineage Key]
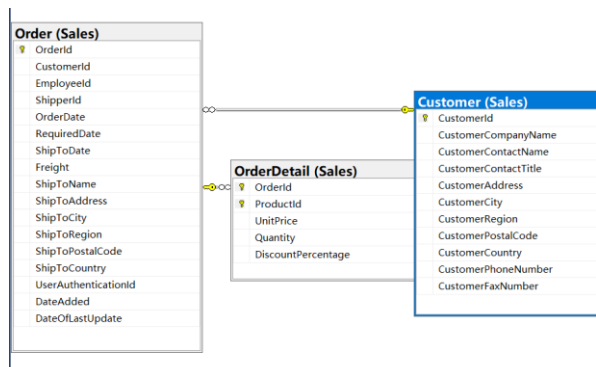
```sql
USE Northw;
DECLARE @chose AS INT =
    (
        SELECT DISTINCT
            O.CustomerId
        FROM Sales.[Order] AS O
        WHERE O.CustomerId = 55
    );
SELECT O.EmployeeId,
    YEAR(O.OrderDate) AS yyear
FROM Sales.[Order] AS O
    JOIN Sales.OrderDetail AS OD
        ON OD.OrderId = O.OrderId
    JOIN Sales.Customer AS C
        ON C.CustomerId = O.CustomerId
WHERE O.CustomerId = @chose
GROUP BY O.EmployeeId,
    YEAR(O.OrderDate);
```

Results | Messages

	EmployeeId	yyear
1	4	2014
2	8	2014
3	1	2015
4	3	2015
5	8	2015
6	2	2016
7	3	2016

(9 rows affected)

Completion time: 2021-10-11T16:59:06.8738407-04:00

Order (Sales)
- 🔑 OrderId
- CustomerId
- EmployeeId
- ShipperId
- OrderDate
- RequiredDate
- ShipToDate
- Freight
- ShipToName
- ShipToAddress
- ShipToCity
- ShipToRegion
- ShipToPostalCode
- ShipToCountry
- UserAuthenticationId
- DateAdded
- DateOfLastUpdate

OrderDetail (Sales)
- 🔑 OrderId
- 🔑 ProductId
- UnitPrice
- Quantity
- DiscountPercentage

Customer (Sales)
- 🔑 CustomerId
- CustomerCompanyName
- CustomerContactName
- CustomerContactTitle
- CustomerAddress
- CustomerCity
- CustomerRegion
- CustomerPostalCode
- CustomerCountry
- CustomerPhoneNumber
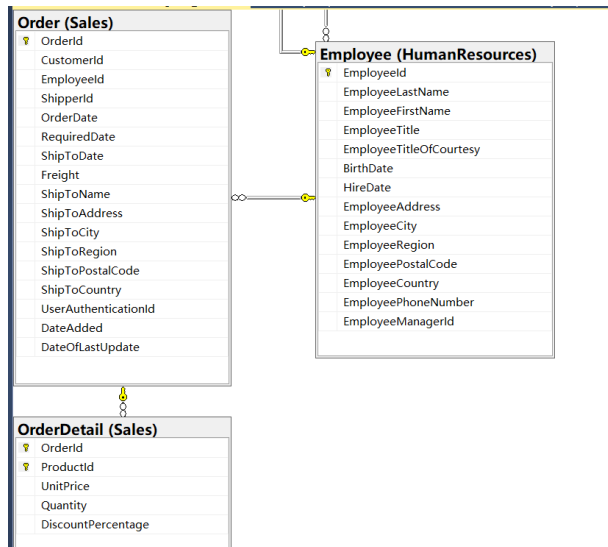- CustomerFaxNumber

```sql
USE Northw;
SELECT O.EmployeeId,
    YEAR(O.OrderDate) AS YYear,
    COUNT(O.OrderId) AS num,
    SUM(OD.UnitPrice * OD.Quantity) AS totalprice
FROM Sales.[Order] AS O
    LEFT OUTER JOIN Sales.OrderDetail AS OD
        ON OD.OrderId = O.OrderId
    LEFT OUTER JOIN HumanResources.Employee AS E
        ON E.EmployeeId = O.EmployeeId
GROUP BY O.EmployeeId,
    YEAR(O.OrderDate),
    O.OrderId,
    OD.UnitPrice * OD.Quantity
ORDER BY O.EmployeeId,
    YYear,
    totalprice;
--FOR JSON PATH ,ROOT('7')
```

	EmployeeId	YYear	num	totalprice
1	1	2014	1	29.50
2	1	2014	1	34.40
3	1	2014	1	36.00
4	1	2014	1	37.20
5	1	2014	1	40.00
6	1	2014	1	43.20
7	1	2014	1	48.00
8	1	2014	1	54.00
9	1	2014	1	57.60
10	1	2014	1	67.20
11	1	2014	1	77.00
12	1	2014	1	78.40
13	1	2014	1	80.00
14	1	2014	1	86.00
15	1	2014	1	91.20
16	1	2014	1	92.40
17	1	2014	1	100.10
18	1	2014	1	120.00
19	1	2014	1	153.60

```
(2150 rows affected)

Completion time: 2021-10-11T20:58:18.9584124-04:00
```

```sql
SELECT Employeeinformation.OrderYear,
    NumberOfUniqueCustomers = COUNT(DISTINCT Employeeinformation.CustomerId),
    Employeeinformation.EmployeeId
FROM
(
    SELECT YEAR(o.OrderDate) AS OrderYear,
        o.CustomerId,
        o.EmployeeId
    FROM Sales.[Order] AS o
) AS Employeeinformation
    INNER JOIN Sales.[Order] AS o1
        ON o1.CustomerId = Employeeinformation.CustomerId
    INNER JOIN Sales.[Order] AS o2
        ON o2.CustomerId = Employeeinformation.CustomerId
GROUP BY Employeeinformation.OrderYear,
        Employeeinformation.EmployeeId
ORDER BY Employeeinformation.EmployeeId,
        Employeeinformation.OrderYear;
FOR JSON PATH, ROOT('8')
```

	OrderYear	NumberOfUniqueCustomers	EmployeeId
1	2014	22	1
2	2015	40	1
3	2016	32	1
4	2014	15	2
5	2015	35	2
6	2016	34	2
7	2014	16	3
8	2015	46	3
9	2016	30	3
10	2014	26	4
11	2015	57	4
12	2016	33	4
13	2014	10	5
14	2015	13	5
15	2016	11	5
16	2014	15	6
17	2015	24	6
18	2016	17	6
19	2014	11	7
20	2015	30	7
21	2016	21	7
22	2014	19	8
23	2015	36	8
24	2016	23	8
25	2014	5	9
26	2015	16	9
27	2016	16	9

```
(27 rows affected)

Completion time: 2021-10-11T21:04:15.2311381-04:00
```

Order (Sales)
- OrderId
- CustomerId
- EmployeeId
- ShipperId
- OrderDate
- RequiredDate
- ShipToDate
- Freight
- ShipToName
- ShipToAddress
- ShipToCity
- ShipToRegion
- ShipToPostalCode
- ShipToCountry
- UserAuthenticationId
- DateAdded
- DateOfLastUpdate

Results

	orderyear	numcusts
1	2014	1
2	2014	1
3	2014	1
4	2014	1
5	2014	1
6	2014	1
7	2014	1
8	2014	1
9	2014	1
10	2014	1
11	2014	1
12	2014	1
13	2014	1
14	2014	1
15	2014	1
16	2014	1
17	2014	1
18	2014	1
19	2014	1
20	2014	1
21	2014	1
22	2014	1
23	2014	1

Results   Messages

(125 rows affected)

Completion time: 2021-10-11T21:49:08.4169939-04:00

```sql
USE Northw;
DECLARE @empid AS INT = 5;
SELECT D.orderyear,
    COUNT(DISTINCT D.CustomerId) AS numcusts
FROM
(
    SELECT YEAR(OrderDate) AS orderyear,
        CustomerId,
        EmployeeId
    FROM Sales.[Order]
    WHERE EmployeeId IN ( @empid, 6, 7 )
) AS D
    INNER JOIN Sales.Customer AS C
        ON C.CustomerId = D.CustomerId
    INNER JOIN HumanResources.Employee AS E
        ON E.EmployeeId = D.EmployeeId
WHERE D.orderyear >= 2013
GROUP BY D.orderyear,
        C.CustomerId
ORDER BY D.orderyear;
```

	orderyear	numcusts
1	2014	1
2	2014	1
3	2014	1
4	2014	1
5	2014	1
6	2014	1
7	2014	1
8	2014	1
9	2014	1
10	2014	1
11	2014	1
12	2014	1
13	2014	1
14	2014	1
15	2014	1
16	2014	1
17	2014	1
18	2014	1
19	2014	1

(125 rows affected)

Completion time: 2021-10-11T21:49:08.4169939-04:00

```sql
SELECT Cur.orderyear,
    Cur.numcusts AS curnumcusts,
    Prv.numcusts AS prvnumcusts,
    Cur.numcusts - Prv.numcusts AS growth
FROM
(
    SELECT YEAR(OrderDate) AS orderyear,
        COUNT(DISTINCT CustomerId) AS numcusts
    FROM Sales.[Order]
    GROUP BY YEAR(OrderDate)
) AS Cur
    LEFT OUTER JOIN
    (
        SELECT YEAR(OrderDate) AS orderyear,
            COUNT(DISTINCT CustomerId) AS numcusts
        FROM Sales.[Order]
        GROUP BY YEAR(OrderDate)
    ) AS Prv
        ON Cur.orderyear = Prv.orderyear + 1
    LEFT OUTER JOIN
    (
        SELECT YEAR(OrderDate) AS orderyear,
            COUNT(DISTINCT CustomerId) AS numcusts
        FROM Sales.[Order]
        GROUP BY YEAR(OrderDate)
    ) AS nex
        ON Cur.orderyear = nex.orderyear - 1
```

⊞ Results   📄 Messages

	orderyear	curnumcusts	prvnumcusts	growth
1	2014	67	NULL	NULL
2	2015	86	67	19
3	2016	81	86	-5

```
(3 rows affected)

Completion time: 2021-10-11T22:21:43.5609032-04:00
```