

```
USE Northwinds2020TSQLV6;
GO
```

## Muhammad Query 1 - good

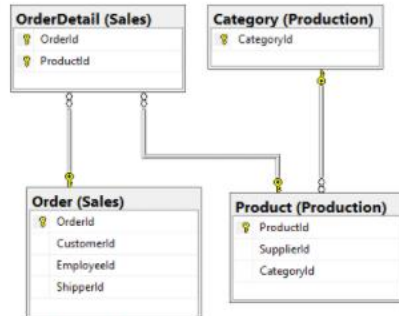
(352 rows affected)

```
CREATE FUNCTION [dbo].[getProductDescription] (@type INT)
RETURNS NVARCHAR(100)
AS
BEGIN
    DECLARE @desc VARCHAR(100)

    SET @desc = (
        SELECT [Description]
        FROM Production.Category
        WHERE CategoryId = @type
    )

    RETURN @desc
END;
```

Select every order and its category  
from a customer in the USA  
show product description using  
custom function  
from Northwinds2020TSQLV6;



```
USE Northwinds2020TSQLV6;
GO
```

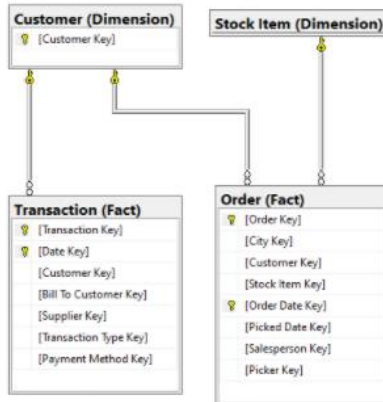
```
SELECT O.OrderId
, CONCAT (
    '$'
    , D.UnitPrice
) AS 'Unit Price'
, P.ProductName
, C.CategoryName
, dbo.getProductDescription(P.CategoryId) AS 'Description'
FROM Sales.[Order] AS O
INNER JOIN Sales.OrderDetail AS D ON O.OrderId = D.OrderId
INNER JOIN Production.Product AS P ON D.ProductId = P.ProductId
INNER JOIN Production.Category AS C ON P.CategoryId = C.CategoryId
WHERE O.CustomerId IN (
    SELECT CustomerId
    FROM Sales.Customer
    WHERE CustomerCountry = 'USA'
)
GROUP BY O.OrderId
, D.UnitPrice
, P.ProductName
, C.CategoryName
, dbo.getProductDescription(P.CategoryId)
```

Results	Messages	OrderId	Unit Price	ProductName	CategoryName	Description
1		10262	\$17.00	Product EPEIM	Condiments	Sweet and savory sauces, relishes, spreads, and ...
2		10262	\$24.00	Product HMLNI	Produce	Dried fruit and bean curd
3		10262	\$30.40	Product VKCMF	Grains/Cereals	Breads, crackers, pasta, and cereal
4		10269	\$2.00	Product ASTMN	Dairy Products	Cheeses
5		10269	\$27.80	Product GEEEO	Dairy Products	Cheeses
6		10271	\$2.00	Product ASTMN	Dairy Products	Cheeses
7		10272	\$10.00	Product XWOXC	Dairy Products	Cheeses
8		10272	\$27.80	Product GEEEO	Dairy Products	Cheeses
9		10272	\$64.80	Product GHFFG	Confections	Desserts, candies, and sweet breads
10		10294	\$6.20	Product BWRLG	Beverages	Soft drinks, coffees, teas, beers, and ales
11		10294	\$14.40	Product HHYDP	Beverages	Soft drinks, coffees, teas, beers, and ales
12		10294	\$27.20	Product WHBYK	Dairy Products	Cheeses
13		10294	\$31.20	Product BLCAH	Meat/Poultry	Prepared meats
14		10294	\$36.80	Product ZZZHumanResources	Beverages	Soft drinks, coffees, teas, beers, and ales
15		10305	\$14.40	Product LSOFL	Beverages	Soft drinks, coffees, teas, beers, and ales
16		10305	\$50.00	Product CKEDC	Seafood	Seaweed and fish
17		10305	\$99.00	Product VUXYN	Meat/Poultry	Prepared meats
18		10307	\$10.00	Product TBTLN	Confections	Desserts, candies, and sweet breads
19		10307	\$39.40	Product WUXYK	Confections	Desserts, candies, and sweet breads
20		10310	\$13.90	Product PAFRH	Confections	Desserts, candies, and sweet breads
21		10310	\$39.40	Product WUXYK	Confections	Desserts, candies, and sweet breads
22		10314	\$10.60	Product ACRVI	Seafood	Seaweed and fish
23		10314	\$25.60	Product NUNAW	Dairy Products	Cheeses
24		10314	\$39.40	Product WUXYK	Confections	Desserts, candies, and sweet breads
25		10316	\$7.70	Product TTEEX	Seafood	Seaweed and fish
26		10316	\$39.40	Product WUXYK	Confections	Desserts, candies, and sweet breads
27		10317	\$14.40	Product HHYDP	Beverages	Soft drinks, coffees, teas, beers, and ales
28		10324	\$9.60	Product CBRRH	Seafood	Seaweed and fish
29		10324	\$13.90	Product PAFRH	Confections	Desserts, candies, and sweet breads
30		10324	\$14.40	Product NEVTJ	Beverages	Soft drinks, coffees, teas, beers, and ales
31		10324	\$35.10	Product KCKNK	Condiments	Sweet and savory sauces, relishes, spreads, and ...
32		10324	\$44.00	Product UKXRI	Dairy Products	Cheeses
33		10329	\$7.30	Product XOXOD	Confections	Desserts, candies, and sweet breads
34		10329	\$20.70	Product LYERX	Seafood	Seaweed and fish
35		10329	\$30.40	Product VKCMF	Grains/Cereals	Breads, crackers, pasta, and cereal
36		10329	\$210.80	Product QDOMO	Beverages	Soft drinks, coffees, teas, beers, and ales
37		10338	\$20.70	Product LYERX	Seafood	Seaweed and fish
38		10338	\$31.20	Product BLCAH	Meat/Poultry	Prepared meats
39		10344	\$17.60	Product KSBRM	Condiments	Sweet and savory sauces, relishes, spreads, and ...
40		10344	\$32.00	Product WVJFP	Condiments	Sweet and savory sauces, relishes, spreads, and ...

```
-- select customer orders where they have ordered a 'joke item'
-- and paid using 'EFT', using custom function
-- from WideWorldImportersDW;
USE WideWorldImportersDW;
GO
```

## Muhammad Query 2 - good

```
CREATE FUNCTION dbo.isJokeItem (@stockItem INT)
RETURNS BIT
AS
BEGIN
    IF @stockItem IN (
        SELECT [Stock Item Key]
        FROM Dimension.[Stock Item]
        WHERE [Stock Item] LIKE '%joke%'
    )
        RETURN 1
    ELSE
        RETURN 0
END;
```



```
SELECT O.[Customer Key]
, SI.[Stock Item]
, C.Category
FROM Fact.[Order] AS O
INNER JOIN Dimension.[Stock Item] AS SI ON O.[Stock Item Key] = SI.[Stock Item Key]
INNER JOIN Dimension.Customer AS C ON O.[Customer Key] = C.[Customer Key]
INNER JOIN Fact.[Transaction] AS T ON O.[Order Key] = T.[Transaction Key]
WHERE T.[Payment Method Key] IN (
    SELECT [Payment Method Key]
    FROM Dimension.[Payment Method]
    WHERE [Payment Method] = 'EFT'
)
AND dbo.isJokeItem(SI.[Stock Item Key]) = 1
GROUP BY O.[Customer Key]
, SI.[Stock Item]
, C.Category
ORDER BY [Customer Key]
```

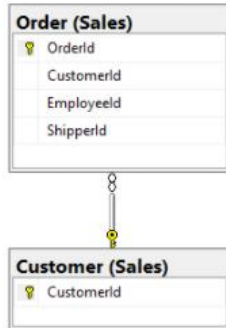
(3302 rows affected)

	Customer Key	Stock Item	Category
898	106	Developer joke mug - a foo walks into a b...	Novelt...
899	106	Developer joke mug - fun was unexpected...	Novelt...
900	106	Developer joke mug - inheritance is the O...	Novelt...
901	106	Developer joke mug - Oct 31 = Dec 25 (Bl...	Novelt...
902	106	Developer joke mug - Oct 31 = Dec 25 (W...	Novelt...
903	106	Developer joke mug - old C developers ne...	Novelt...
904	106	Developer joke mug - understanding recur...	Novelt...
905	106	Developer joke mug - when your hammer i...	Novelt...
906	106	IT joke mug - hardware: part of the comput...	Novelt...
907	106	IT joke mug - keyboard not found - press ...	Novelt...
908	107	DBA joke mug - mind if I join you? (White)	Novelt...
909	107	Developer joke mug - (hip, hip, array) (Black)	Novelt...
910	107	Developer joke mug - Oct 31 = Dec 25 (Bl...	Novelt...
911	107	Developer joke mug - Oct 31 = Dec 25 (W...	Novelt...
912	107	Developer joke mug - that's a hardware pr...	Novelt...
913	107	Developer joke mug - there are 10 types of...	Novelt...
914	107	IT joke mug - keyboard not found - press ...	Novelt...
915	107	IT joke mug - that behavior is by design (Bl...	Novelt...
916	107	IT joke mug - that behavior is by design (...)	Novelt...
917	108	DBA joke mug - daaaaaa-ta (Black)	Novelt...
918	108	DBA joke mug - it depends (White)	Novelt...
919	108	DBA joke mug - you might be a DBA if (Bla...	Novelt...
920	108	Developer joke mug - fun was unexpected...	Novelt...
921	108	Developer joke mug - inheritance is the O...	Novelt...
922	108	Developer joke mug - old C developers ne...	Novelt...
923	108	Developer joke mug - this code was gener...	Novelt...
924	108	Developer joke mug - when your hammer i...	Novelt...
925	108	Developer joke mug - when your hammer i...	Novelt...
926	108	IT joke mug - hardware: part of the comput...	Novelt...
927	109	DBA joke mug - I will get you in order (White)	Novelt...
928	109	DBA joke mug - SELECT caffeine FROM ...	Novelt...
929	109	Developer joke mug - a foo walks into a b...	Novelt...
930	109	Developer joke mug - Oct 31 = Dec 25 (W...	Novelt...
931	109	Developer joke mug - old C developers ne...	Novelt...
932	109	IT joke mug - hardware: part of the comput...	Novelt...
933	109	IT joke mug - that behavior is by design (Bl...	Novelt...
934	110	DBA joke mug - it depends (Black)	Novelt...
935	110	DBA joke mug - mind if I join you? (Black)	Novelt...

## Muhammad Query 3 - good

```
-- Select the most recent order for every company if it was a winter month
-- order by most recent,
-- show the order date with US standard
-- from Northwinds2020TSQLV6;
USE Northwinds2020TSQLV6;
GO
```

```
CREATE FUNCTION dbo.isWinterMonth (@orderdate DATE)
RETURNS BIT
AS
BEGIN
    IF MONTH(@orderdate) IN (
        1
        ,2
        ,12
    )
        RETURN 1
    RETURN 0
END;
USE Northwinds2020TSQLV6;
GO
```



	Company	OrderId	Order Date
1	Customer BSVAR	10890	02/16/2016
2	Customer QXPPT	10884	02/12/2016
3	Customer AHPOP	10869	02/04/2016
4	Customer KBUDE	10856	01/28/2016
5	Customer GYBBY	10848	01/23/2016
6	Customer NRCSK	10850	01/23/2016
7	Customer YQQ...	10839	01/19/2016
8	Customer QXVLA	10826	01/12/2016
9	Customer EYHKM	10822	01/08/2016
10	Customer IAIJK	10810	01/01/2016
11	Customer WVFAF	10789	12/22/2015
12	Customer FVXPQ	10785	12/18/2015

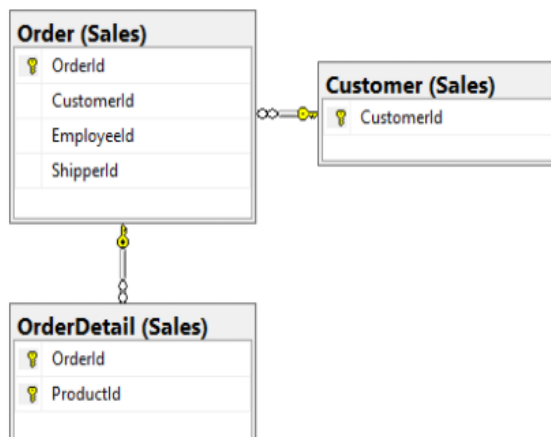
```
SELECT C.CustomerCompanyName AS 'Company'
      ,O1.OrderId
      ,CONVERT(VARCHAR, O1.OrderDate, 101) AS 'Order Date'
FROM Sales.[Order] AS O1
INNER JOIN Sales.Customer AS C ON O1.CustomerId = C.CustomerId
WHERE O1.OrderDate = (
    SELECT MAX(O2.OrderDate)
    FROM Sales.[Order] O2
    WHERE O1.CustomerId = O2.CustomerId
)
AND dbo.isWinterMonth(O1.OrderDate) = 1
ORDER BY O1.OrderDate DESC
```

(12 rows affected)

```
-- 16 - Complex
-- select customer and calculate order id
-- where the shipper's phone number ends in '6'
-- from Northwinds2020TSQLV6;
USE Northwinds2020TSQLV6;
GO
```

## Muhammad Query 1 - not so good

```
CREATE FUNCTION dbo.TotalAmount (
    @quantity INT
    ,@price MONEY
    ,@discount NUMERIC(4, 3)
)
RETURNS DEC(10, 2)
AS
BEGIN
    RETURN @quantity * @price * (1 - @discount)
END;
USE Northwinds2020TSQLV6;
GO
```



```
SELECT C.CustomerID
      ,D.Quantity
      ,D.UnitPrice
      ,D.DiscountPercentage
      ,SUM(dbo.TotalAmount(D.Quantity, D.UnitPrice, D.DiscountPercentage)) AS TotalCost
FROM Sales.[Order] AS O
INNER JOIN Sales.Customer AS C ON O.CustomerId = C.CustomerId
INNER JOIN Sales.OrderDetail AS D ON O.OrderId = D.OrderId
WHERE O.ShipperId = (
    SELECT ShipperId
    FROM Sales.Shipper
    WHERE PhoneNumber LIKE N'%6'
)
GROUP BY C.CustomerID
      ,O.OrderId
      ,O.orderdate
      ,D.Quantity
      ,D.UnitPrice
      ,D.DiscountPercentage
```

(864 rows affected)

	CustomerID	Quantity	UnitPrice	DiscountPercentage
1	34	10	7.70	0.000
2	34	15	16.80	0.150
3	34	35	42.40	0.150
4	76	25	2.00	0.050
5	76	40	27.20	0.000
6	76	40	64.80	0.050
7	34	20	10.00	0.000
8	34	40	16.00	0.000
9	34	42	14.40	0.000
10	14	15	3.60	0.150
11	14	21	8.00	0.000
12	14	21	19.20	0.150
13	88	12	10.40	0.000
14	88	15	26.20	0.000
15	61	20	8.00	0.000
16	61	20	14.40	0.000
17	75	24	2.00	0.000
18	65	6	64.80	0.000
19	65	24	27.80	0.000
20	65	40	10.00	0.000
21	5	8	35.10	0.000
22	5	15	44.00	0.000
23	5	16	15.50	0.000
24	5	25	12.00	0.000
25	44	15	31.20	0.250
26	63	36	26.20	0.200
27	63	40	14.70	0.200
28	63	45	14.40	0.200
29	61	2	42.40	0.100
30	61	20	4.80	0.100
31	61	24	15.50	0.100
32	81	20	64.80	0.000
33	65	6	6.20	0.000
34	65	15	31.20	0.000
35	65	15	36.80	0.000



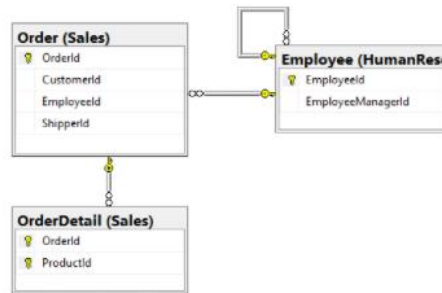
```
-- 15 - Complex
-- get the orderdates where more than 10 items ordered and completed by
-- the most recently hired employee, show age in years and days
-- from Northwinds2020TSQLV6;
USE Northwinds2020TSQLV6
GO
```

## Muhammad Query 2 - not so good

```
CREATE FUNCTION dbo.AgeToDays (@age INT)
RETURNS INT
AS
BEGIN
    RETURN @age * 365
END;

USE Northwinds2020TSQLV6;
GO

SELECT DISTINCT O.OrderDate
, CONCAT (
    E.EmployeeFirstName
    , E.EmployeeLastName
) AS 'Employee'
, DATEDIFF(YEAR, E.birthday, GETDATE()) AS 'Age (Years)'
, dbo.AgeToDays(DATEDIFF(YEAR, E.birthday, GETDATE())) AS 'Age (days)'
FROM Sales.[Order] AS O
INNER JOIN Sales.OrderDetail AS D ON O.OrderId = D.OrderId
INNER JOIN HumanResources.Employee AS E ON O.EmployeeId = E.EmployeeId
WHERE E.EmployeeId IN (
    SELECT EmployeeId
    FROM HumanResources.Employee
    WHERE HireDate IN (
        SELECT MAX(HireDate)
        FROM HumanResources.Employee
    )
)
AND D.Quantity > 10
```



	OrderDate	Employee	Age (Years)	Age (days)
1	2014-07-12	Patricia Doyle	35	12775
2	2014-07-23	Patricia Doyle	35	12775
3	2014-10-08	Patricia Doyle	35	12775
4	2014-10-16	Patricia Doyle	35	12775
5	2014-12-18	Patricia Doyle	35	12775
6	2015-01-10	Patricia Doyle	35	12775
7	2015-03-14	Patricia Doyle	35	12775
8	2015-04-09	Patricia Doyle	35	12775
9	2015-04-15	Patricia Doyle	35	12775
10	2015-06-03	Patricia Doyle	35	12775
11	2015-06-12	Patricia Doyle	35	12775
12	2015-06-23	Patricia Doyle	35	12775
13	2015-08-27	Patricia Doyle	35	12775
14	2015-09-17	Patricia Doyle	35	12775
15	2015-09-30	Patricia Doyle	35	12775
16	2015-10-15	Patricia Doyle	35	12775
17	2015-11-11	Patricia Doyle	35	12775
18	2015-11-18	Patricia Doyle	35	12775
19	2015-11-21	Patricia Doyle	35	12775
20	2015-12-10	Patricia Doyle	35	12775
21	2015-12-26	Patricia Doyle	35	12775
22	2016-01-13	Patricia Doyle	35	12775
23	2016-01-16	Patricia Doyle	35	12775
24	2016-01-23	Patricia Doyle	35	12775
25	2016-02-05	Patricia Doyle	35	12775

(36 rows affected)

## Muhammad Query 3 - not so good

```
-- 1 - Simple
-- Select every order from every customer from AdventureWorks2017
USE AdventureWorks2017;
GO

SELECT *
FROM Sales.Customer C
INNER JOIN Sales.SalesOrderHeader AS O ON C.CustomerID = O.CustomerID;
```

### Customer (Sales)

CustomerID
PersonID
StoreID
TerritoryID
AccountNumber
rowguid
ModifiedDate

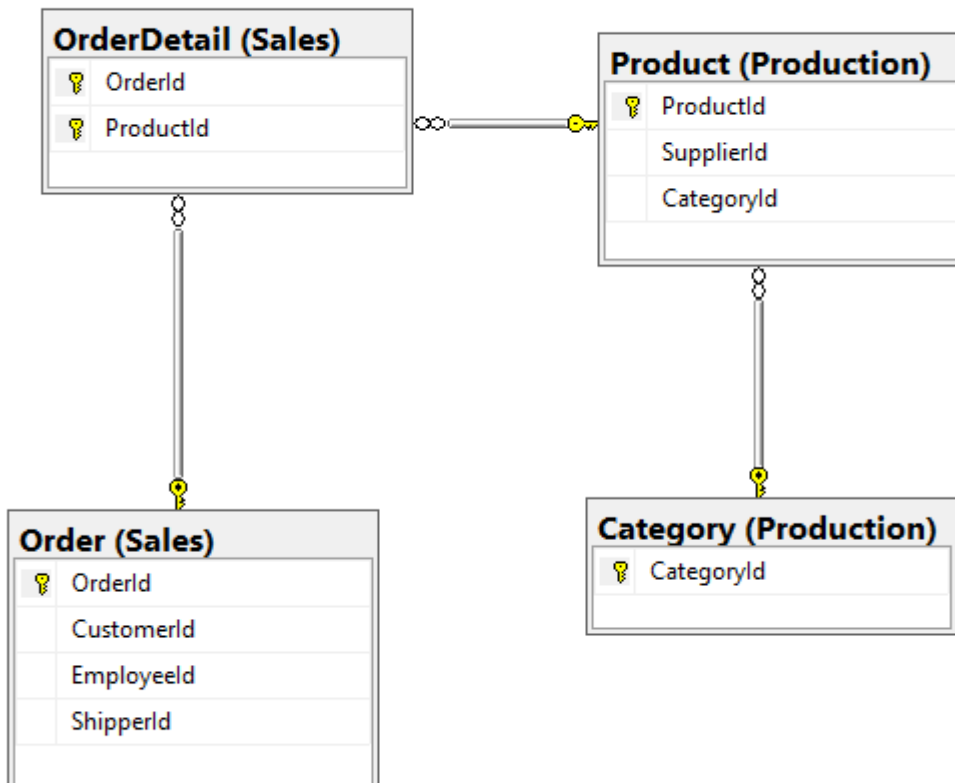
	CustomerID	PersonID	StoreID	TerritoryID	AccountNumber	rowguid	ModifiedDate	SalesOrderID	RevisionNumber	OrderDate	DueDate	ShipDate	Status
415	11044	5244	NULL	9	AW00011044	F7691B2E-2209-4649-A3DC-2BC9A77349EA	2014-09-12 11:15:07.263	44073	8	2011-07-31 00:00:00.000	2011-08-12 00:00:00.000	2011-08-07 00:00:00.000	5
416	29592	543	544	2	AW00029592	57C1E381-05F6-4C55-9639-D513D43D7164	2014-09-12 11:15:07.263	44074	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
417	29632	631	632	1	AW00029632	B162868C-2CA6-4FF6-9DD0-6FA6C03D1B99	2014-09-12 11:15:07.263	44075	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
418	30003	1435	1436	3	AW00030003	A6FE5230-AFE5-4633-AD8B-84176A335960	2014-09-12 11:15:07.263	44076	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
419	29714	811	812	6	AW00029714	DF3FEDCE-E229-4EE7-8F01-402C29B7817F	2014-09-12 11:15:07.263	44077	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
420	29622	611	612	1	AW00029622	EA5EA0F8-67B9-4D47-9D52-220F951FC446	2014-09-12 11:15:07.263	44078	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
421	29967	1357	1358	3	AW00029967	C80B8680-FAEA-4732-8DB2-CD6AD1B5226A	2014-09-12 11:15:07.263	44079	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
422	29733	849	850	4	AW00029733	E8F81644-1D44-4D25-9756-5897E20FBAAE	2014-09-12 11:15:07.263	44080	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
423	29950	1321	1322	6	AW00029950	0631F720-A96D-4505-A07C-11A954A67F68	2014-09-12 11:15:07.263	44081	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
424	29803	997	998	6	AW00029803	DE649D68-0F40-4170-A584-4AF87DF50083	2014-09-12 11:15:07.263	44082	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
425	29969	1363	1364	6	AW00029969	FACB12B2-C8B4-42FE-8641-F753856806B4	2014-09-12 11:15:07.263	44083	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
426	29494	311	312	6	AW00029494	66D2DC13-4220-43B8-B2C7-1311B3CAEC34	2014-09-12 11:15:07.263	44084	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
427	29639	647	648	6	AW00029639	16864FBF-49C0-46CA-AC6C-530EB20CAD1B	2014-09-12 11:15:07.263	44085	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
428	29722	827	828	6	AW00029722	68396E18-907F-4CD5-A5F8-84D7969DAC11	2014-09-12 11:15:07.263	44086	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
429	29944	1307	1308	4	AW00029944	881137FC-AE6A-4FBD-B71A-4AF9C2F65914	2014-09-12 11:15:07.263	44087	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
430	29842	1081	1082	4	AW00029842	5D6433E4-A1A5-43EB-9056-78F234C9150E	2014-09-12 11:15:07.263	44088	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
431	29673	723	724	4	AW00029673	82FF1AAC-744A-447D-87D6-7FAC7C2BAF89	2014-09-12 11:15:07.263	44089	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
432	29659	693	694	4	AW00029659	293F44D5-8949-4FD0-9337-DDACD20FD948	2014-09-12 11:15:07.263	44090	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
433	29914	1243	1244	4	AW00029914	A4ED5B4F-19A2-4C26-A323-E952A84E6DD7	2014-09-12 11:15:07.263	44091	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5
434	29999	1033	1034	4	AW00029999	7E4A1A7B-376C-437E-BD34-3A3086A375EE	2014-09-12 11:15:07.263	44092	8	2011-08-01 00:00:00.000	2011-08-13 00:00:00.000	2011-08-08 00:00:00.000	5

(31465 rows affected)

## Improved Queries

```
-- Medium
-- select every order and its category from a customer in the USA
-- from Northwinds2020TSQVL6;
USE Northwinds2020TSQVL6;
GO

SELECT O.OrderId
      ,CONCAT (
          '$'
          ,D.UnitPrice
        ) AS 'Unit Price'
      ,P.ProductName
      ,C.CategoryName
FROM Sales.[Order] AS O
INNER JOIN Sales.OrderDetail AS D ON O.Orderid = D.OrderId
INNER JOIN Production.Product AS P ON D.ProductId = P.ProductId
INNER JOIN Production.Category AS C ON P.CategoryId = C.CategoryId
WHERE O.CustomerId IN (
    SELECT CustomerId
    FROM Sales.Customer
    WHERE CustomerCountry = N'USA'
)
GROUP BY O.OrderId
      ,D.UnitPrice
      ,P.ProductName
      ,C.CategoryName
```

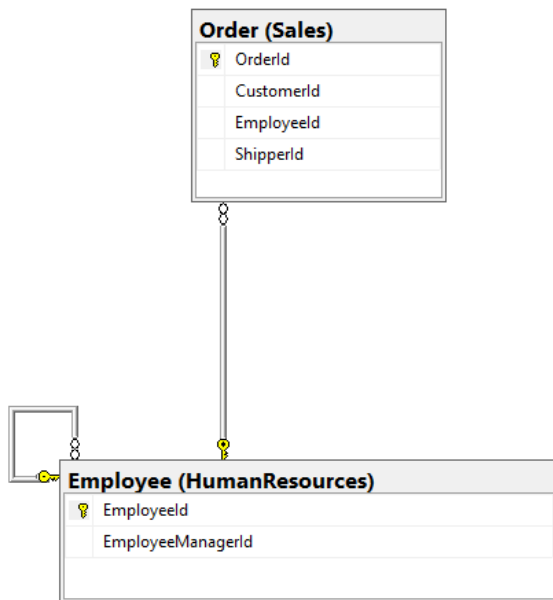


Results		Messages		
	OrderId	Unit Price	ProductName	CategoryName
1	10262	\$17.00	Product EPEIM	Condiments
2	10262	\$24.00	Product HMLNI	Produce
3	10262	\$30.40	Product VKCMF	Grains/Cereals
4	10269	\$2.00	Product ASTMN	Dairy Products
5	10269	\$27.80	Product GEEOO	Dairy Products
6	10271	\$2.00	Product ASTMN	Dairy Products
7	10272	\$10.00	Product XWOXC	Dairy Products
8	10272	\$27.80	Product GEEOO	Dairy Products
9	10272	\$64.80	Product QHFFP	Confections
10	10294	\$6.20	Product BWRLG	Beverages
11	10294	\$14.40	Product HHYDP	Beverages
12	10294	\$27.20	Product WHBYK	Dairy Products
13	10294	\$31.20	Product BLCAX	Meat/Poultry
14	10294	\$36.80	Product ZZZHumanResources	Beverages
15	10305	\$14.40	Product LSOFL	Beverages
16	10305	\$50.00	Product CKEDC	Seafood
17	10305	\$99.00	Product VJXYN	Meat/Poultry
18	10307	\$10.00	Product TBTBL	Confections
19	10307	\$39.40	Product WUXYK	Confections

## Improved Queries

```
--Simple
-- Select every order in 2016 and the employee who completed the order
-- from Northwinds2020TSQLV6
USE Northwinds2020TSQLV6;
GO

SELECT CONCAT (
    E.EmployeeFirstName
    ,', '
    ,E.EmployeeLastName
) AS 'Employee Name'
    ,E.EmployeeTitle
    ,O.orderid
    ,O.OrderDate
FROM HumanResources.Employee AS E
INNER JOIN Sales.[Order] AS O ON E.EmployeeId = O.EmployeeId
WHERE YEAR(O.OrderDate) = 2016
```

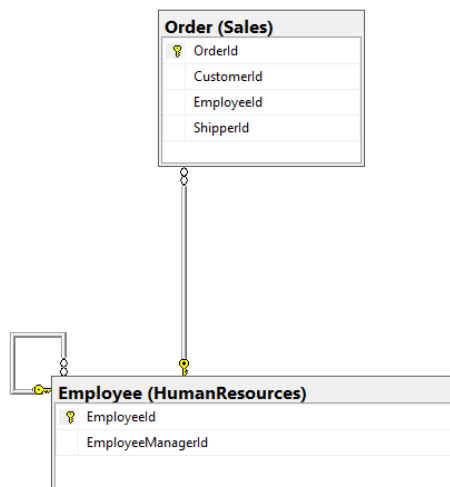


	Employee Name	EmployeeTitle	orderid	OrderDate
1	Don, Funk	Vice President, Sales	10808	2016-01-01
2	Russell, King	Sales Representative	10809	2016-01-01
3	Don, Funk	Vice President, Sales	10810	2016-01-01
4	Maria, Cameron	Sales Representative	10811	2016-01-02
5	Sven, Mortensen	Sales Manager	10812	2016-01-02
6	Sara, Davis	CEO	10813	2016-01-05
7	Judy, Lew	Sales Manager	10814	2016-01-05
8	Don, Funk	Vice President, Sales	10815	2016-01-05
9	Yael, Peled	Sales Representative	10816	2016-01-06
10	Judy, Lew	Sales Manager	10817	2016-01-06
11	Russell, King	Sales Representative	10818	2016-01-07
12	Don, Funk	Vice President, Sales	10819	2016-01-07
13	Judy, Lew	Sales Manager	10820	2016-01-07
14	Sara, Davis	CEO	10821	2016-01-08
15	Paul, Suurs	Sales Representative	10822	2016-01-08
16	Sven, Mortensen	Sales Manager	10823	2016-01-09
17	Maria, Cameron	Sales Representative	10824	2016-01-09
18	Sara, Davis	CEO	10825	2016-01-09
19	Paul, Suurs	Sales Representative	10826	2016-01-12

## Improved Queries

```
- Medium
-- Select all the types of credit cards used
-- along with counts for all orders in 2013, 2014
-- FROM AdventureWorks2017
USE AdventureWorks2017;
GO

SELECT CC.CardType
      ,COUNT(CC.CardType) AS 'Count'
FROM Sales.SalesOrderHeader AS O
INNER JOIN Sales.CreditCard AS CC ON O.CreditCardID = CC.CreditCardID
WHERE YEAR(OrderDate) IN (
    2013
    ,2014
)
GROUP BY CC.CardType
```



	CardType	Count
1	SuperiorCard	6305
2	Vista	6072
3	Distinguish	6308
4	ColonialVoice	6303