

Jamil Kocacal

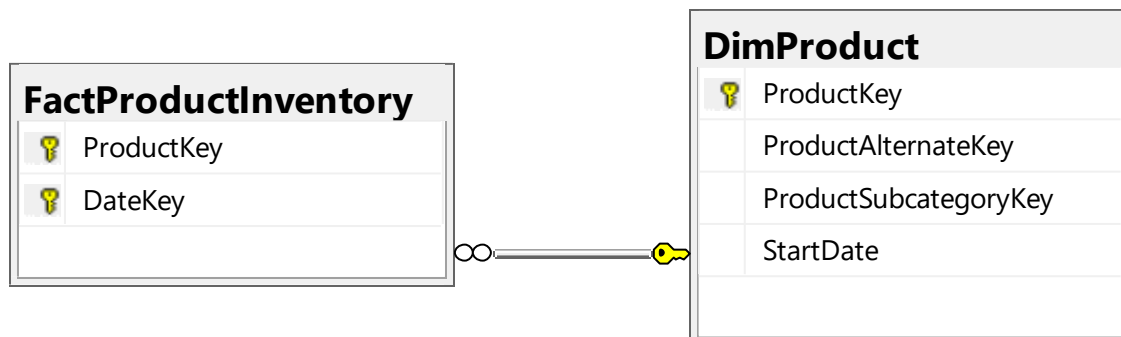
CSCI331

Professor Heller

Group 4 – Project 1

Proposition #1 (Simple)

Using AdventureWorksDW2017 make a table that joins the dbo.DimProduct table and FactProductInventory table such that ProductKey, ProductAlternateKey, UnitCost, UnitsBalance are shown where product key < 50



Column Name	Data Type	Allow Null
ProductKey	int	<input type="checkbox"/>
DateKey	int	<input type="checkbox"/>
MovementDate	date	<input type="checkbox"/>
UnitCost	money	<input type="checkbox"/>
UnitsIn	int	<input type="checkbox"/>
UnitsOut	int	<input type="checkbox"/>
UnitsBalance	int	<input type="checkbox"/>

FK_FactProductInventory_DimProduct

Column Name	Data Type	Allow Null
ProductKey	int	<input type="checkbox"/>
ProductAlternateKey	nvarchar(25)	<input checked="" type="checkbox"/>
ProductSubcategoryKey	int	<input checked="" type="checkbox"/>
WeightUnitMeasureCode	nchar(3)	<input checked="" type="checkbox"/>
SizeUnitMeasureCode	nchar(3)	<input checked="" type="checkbox"/>
EnglishProductName	nvarchar(50)	<input type="checkbox"/>
SpanishProductName	nvarchar(50)	<input type="checkbox"/>
FrenchProductName	nvarchar(50)	<input type="checkbox"/>
StandardCost	money	<input checked="" type="checkbox"/>
FinishedGoodsFlag	bit	<input type="checkbox"/>
Color	nvarchar(15)	<input type="checkbox"/>
SafetyStockLevel	smallint	<input checked="" type="checkbox"/>
ReorderPoint	smallint	<input checked="" type="checkbox"/>
ListPrice	money	<input checked="" type="checkbox"/>
Size	nvarchar(50)	<input checked="" type="checkbox"/>
SizeRange	nvarchar(50)	<input checked="" type="checkbox"/>
Weight	float	<input checked="" type="checkbox"/>
DaysToManufacture	int	<input checked="" type="checkbox"/>
ProductLine	nchar(2)	<input checked="" type="checkbox"/>
DealerPrice	money	<input checked="" type="checkbox"/>
Class	nchar(2)	<input checked="" type="checkbox"/>
Style	nchar(2)	<input checked="" type="checkbox"/>
ModelName	nvarchar(50)	<input checked="" type="checkbox"/>
LargePhoto	varbinary(MAX)	<input checked="" type="checkbox"/>
EnglishDescription	nvarchar(400)	<input checked="" type="checkbox"/>
FrenchDescription	nvarchar(400)	<input checked="" type="checkbox"/>
ChineseDescription	nvarchar(400)	<input checked="" type="checkbox"/>
ArabicDescription	nvarchar(400)	<input checked="" type="checkbox"/>
HebrewDescription	nvarchar(400)	<input checked="" type="checkbox"/>
ThaiDescription	nvarchar(400)	<input checked="" type="checkbox"/>
GermanDescription	nvarchar(400)	<input checked="" type="checkbox"/>
JapaneseDescription	nvarchar(400)	<input checked="" type="checkbox"/>
TurkishDescription	nvarchar(400)	<input checked="" type="checkbox"/>
StartDate	datetime	<input checked="" type="checkbox"/>
EndDate	datetime	<input checked="" type="checkbox"/>

Columns from Tables

Table Name	Column Name
DimProduct	ProductKey, ProductAlternateKey
FactProductInventory	UnitCost, UnitBalance

USE AdventureWorksDW2017

```

SELECT E.ProductKey
        ,E.ProductAlternateKey
        ,O.UnitCost
        ,O.UnitsBalance
FROM dbo.DimProduct AS E

```

```
INNER JOIN dbo.FactProductInventory AS O ON E.ProductKey = O.ProductKey
WHERE E.ProductKey < 10
```

OUTPUT

```
--use AdventureWorksDW2017
--select E.ProductKey, E.ProductAlternateKey, O.UnitCost, O.UnitsBalance
--from dbo.DimProduct as E
--    inner join dbo.FactProductInventory as O
--        on E.ProductKey = O.ProductKey
--    WHERE E.ProductKey < 10
--FOR JSON PATH, ROOT ('makeitup'), INCLUDE_NULL_VALUES;
```

195 %

Results Messages

	ProductKey	ProductAlternateKey	UnitCost	UnitsBalance
1	1	AR-5381	0.32	875
2	1	AR-5381	0.32	875
3	1	AR-5381	0.32	875
4	1	AR-5381	0.32	875
5	1	AR-5381	0.32	875
6	1	AR-5381	0.32	875
7	1	AR-5381	0.32	875
8	1	AR-5381	0.32	875
9	1	AR-5381	0.32	875
10	1	AR-5381	0.32	875
11	1	AR-5381	0.32	875
12	1	AR-5381	0.32	875
13	1	AR-5381	0.32	875
14	1	AR-5381	0.32	875
15	1	AR-5381	0.32	875
16	1	AR-5381	0.32	875
17	1	AR-5381	0.32	875
18	1	AR-5381	0.32	875
19	1	AR-5381	0.32	875

Refresh

Search

ROOT

makeup: [Array]

[0]: [Object]

[1]: [Object]

[2]: [Object]

[3]: [Object]

[4]: [Object]

[5]: [Object]

[6]: [Object]

[7]: [Object]

[8]: [Object]

[9]: [Object]

[10]: [Object]

[11]: [Object]

[12]: [Object]

[13]: [Object]

[14]: [Object]

[15]: [Object]

[16]: [Object]

[17]: [Object]

[18]: [Object]

[19]: [Object]

[20]: [Object]

[21]: [Object]

[22]: [Object]

[23]: [Object]

[24]: [Object]

[25]: [Object]

[26]: [Object]

[27]: [Object]

[28]: [Object]

[29]: [Object]

57620

"UnitCost": 0.2200,

57621

"UnitsBalance": 875

57622

}, {

57623

"ProductKey": 9,

57624

"ProductAlternateKey": "CB-2903",

57625

"UnitCost": 0.2200,

57626

"UnitsBalance": 875

57627

}, {

57628

"ProductKey": 9,

57629

"ProductAlternateKey": "CB-2903",

57630

"UnitCost": 0.2200,

57631

"UnitsBalance": 875

57632

}, {

57633

"ProductKey": 9,

57634

"ProductAlternateKey": "CB-2903",

57635

"UnitCost": 0.2200,

57636

"UnitsBalance": 875

57637

}, {

57638

"ProductKey": 9,

57639

"ProductAlternateKey": "CB-2903",

57640

"UnitCost": 0.2200,

57641

"UnitsBalance": 875

57642

}, {

57643

"ProductKey": 9,

57644

"ProductAlternateKey": "CB-2903",

57645

"UnitCost": 0.2200,

57646

"UnitsBalance": 875

57647

}

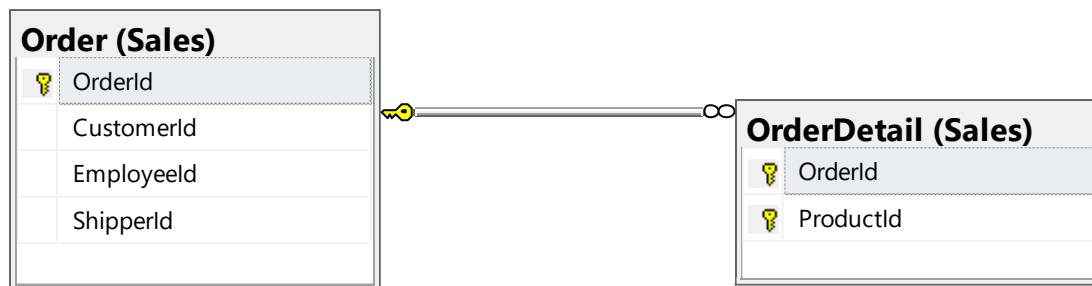
57648

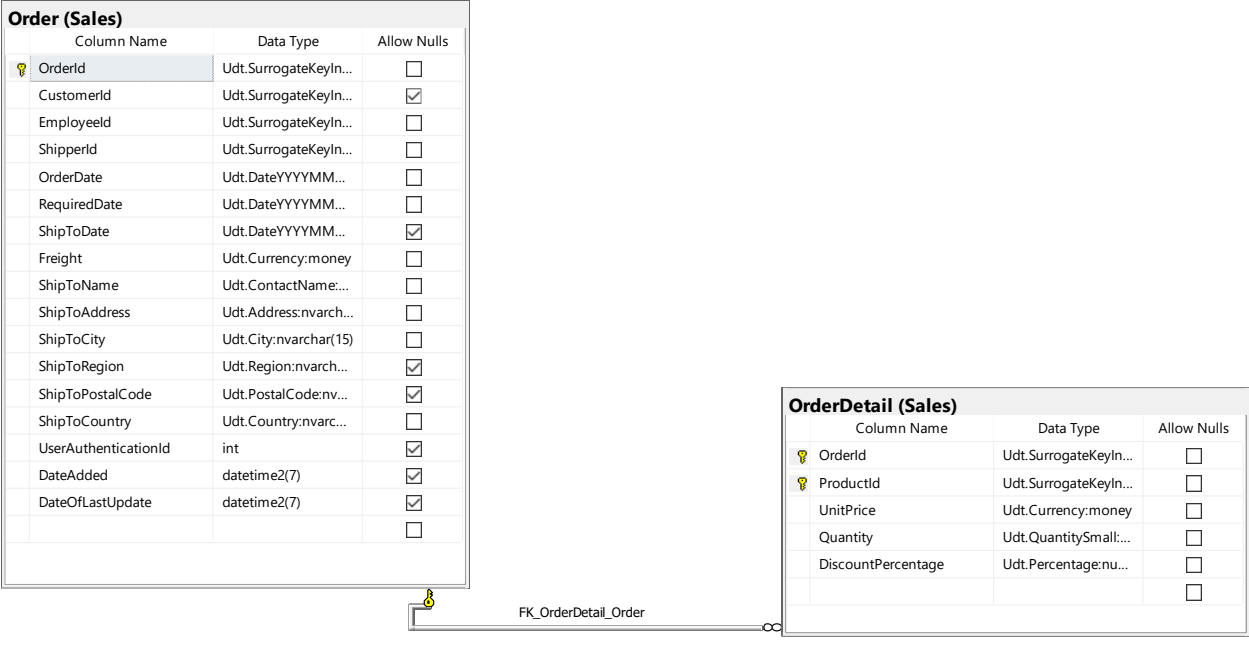
]

57649

Proposition #2 (Simple)

using Northwinds2020TSQLV6 make a table that joins the sales order table and Sales.OrderDetail table such that each row returns the OrderId, CustomerId, EmployeeId, ShipToCountry, ProductId, UnitPrice





Columns from Tables

Table Name	Column Name
Order	OrderId, CustomerId, EmployeeId, ShipToCountry
OrderDetail	UnitCost, UnitBalance

```
USE Northwinds2020TSQVLV6;

SELECT E.OrderId
      ,E.CustomerId
      ,E.EmployeeId
      ,E.ShipToCountry
      ,O.ProductId
      ,O.UnitPrice
FROM Sales.[Order] AS E
INNER JOIN Sales.OrderDetail AS O ON E.OrderId = O.OrderId
```

```

USE Northwinds2020TSQVLV6;

SELECT E.OrderId
      ,E.CustomerId
      ,E.EmployeeId
      ,E.ShipToCountry
      ,O.ProductId
      ,O.UnitPrice
FROM Sales.[Order] AS E
INNER JOIN Sales.OrderDetail AS O ON E.OrderId = O.OrderId
--FOR JSON PATH, ROOT ('makeitup'), INCLUDE_NULL_VALUES;

```

195 %

Results Messages

	OrderId	CustomerId	EmployeeId	ShipToCountry	ProductId	UnitPrice
1	10248	85	5	France	11	14.00
2	10248	85	5	France	42	9.80
3	10248	85	5	France	72	34.80
4	10249	79	6	Germany	14	18.60
5	10249	79	6	Germany	51	42.40
6	10250	34	4	Brazil	41	7.70
7	10250	34	4	Brazil	51	42.40
8	10250	34	4	Brazil	65	16.80
9	10251	84	3	France	22	16.80
10	10251	84	3	France	57	15.60
11	10251	84	3	France	65	16.80
12	10252	76	4	Belgium	20	64.80
13	10252	76	4	Belgium	33	2.00
14	10252	76	4	Belgium	60	27.20
15	10253	34	3	Brazil	31	10.00
16	10253	34	3	Brazil	39	14.40
17	10253	34	3	Brazil	49	16.00
18	10254	14	5	Switzerland	24	3.60
19	10254	14	5	Switzerland	55	19.20

ISToolNpp JSON Viewer

change.log new 1.txt new 1

Refresh Search

ROOT

makeitup: [Array]

[0]: [Object]

[1]: [Object]

[2]: [Object]

[3]: [Object]

[4]: [Object]

[5]: [Object]

[6]: [Object]

[7]: [Object]

[8]: [Object]

[9]: [Object]

[10]: [Object]

[11]: [Object]

[12]: [Object]

[13]: [Object]

[14]: [Object]

[15]: [Object]

[16]: [Object]

[17]: [Object]

[18]: [Object]

[19]: [Object]

[20]: [Object]

[21]: [Object]

[22]: [Object]

[23]: [Object]

[24]: [Object]

[25]: [Object]

[26]: [Object]

[27]: [Object]

[28]: [Object]

[29]: [Object]

15060 "OrderId": 11077,

15061 "CustomerId": 65,

15062 "EmployeeId": 1,

15063 "ShipToCountry": "USA",

15064 "ProductId": 66,

15065 "UnitPrice": 17.0000

15066 }, {

15067 "OrderId": 11077,

15068 "CustomerId": 65,

15069 "EmployeeId": 1,

15070 "ShipToCountry": "USA",

15071 "ProductId": 73,

15072 "UnitPrice": 15.0000

15073 }, {

15074 "OrderId": 11077,

15075 "CustomerId": 65,

15076 "EmployeeId": 1,

15077 "ShipToCountry": "USA",

15078 "ProductId": 75,

15079 "UnitPrice": 7.7500

15080 }, {

15081 "OrderId": 11077,

15082 "CustomerId": 65,

15083 "EmployeeId": 1,

15084 "ShipToCountry": "USA",

15085 "ProductId": 77,

15086 "UnitPrice": 13.0000

15087 }

15088 }

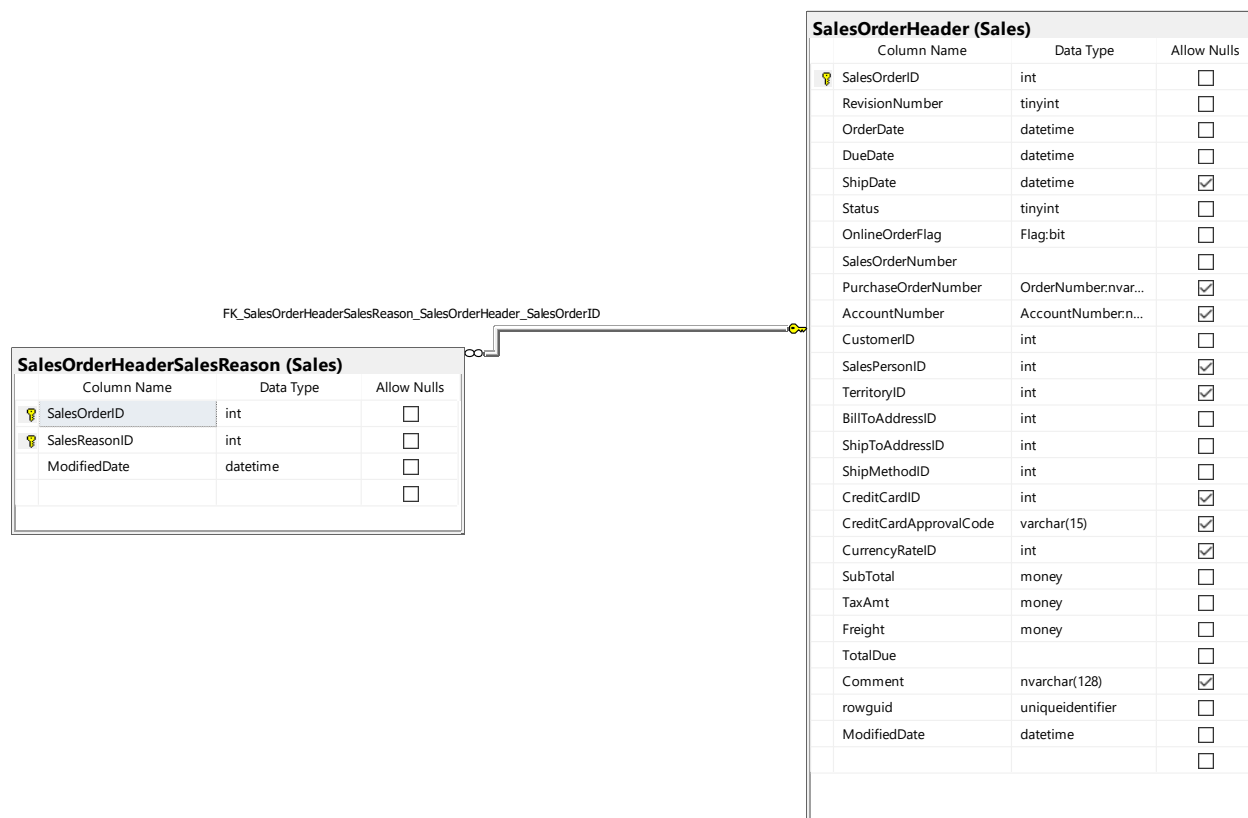
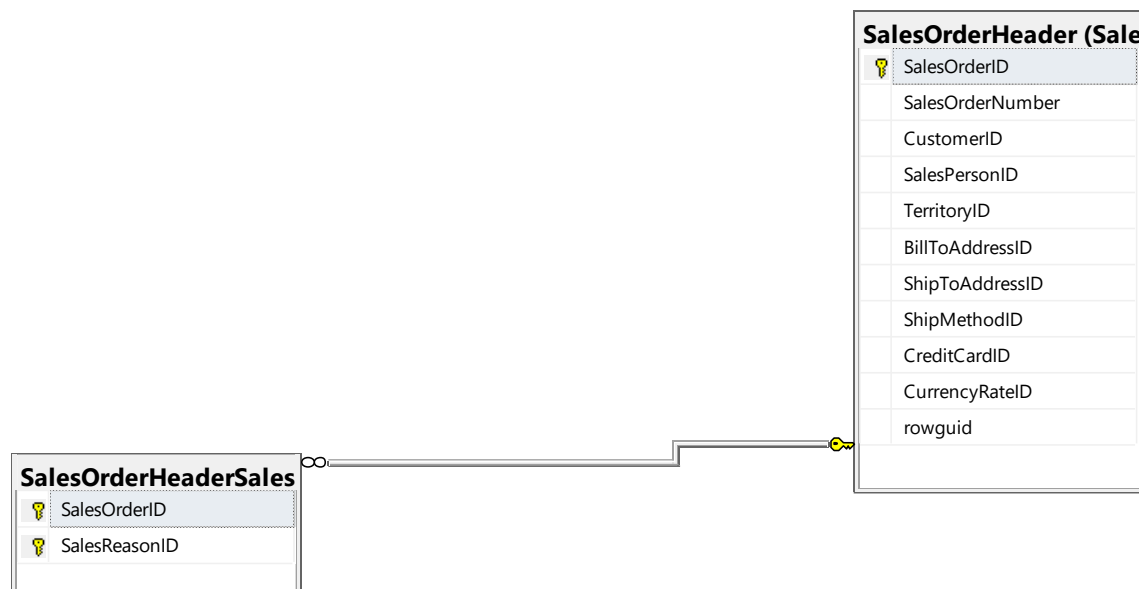
15089 }

Normal text file

length: 451,560 lines: 15,089 Ln: 15,083 Col: 29 Sel: 0|0 Windows (CR LF) UTF-8 INS

Proposition #3 (Simple)

Using AdventureWorks2017 make a table that joins the SalesOrderHeaderSalesReason table along with the SalesOrderHeader such that the table shows the SalesOrderID, SalesReasonID, CustomerID, ModifiedDate, AccountNumber



Columns from Tables

Table Name	Column Name
SalesOrderHeaderSalesReason	SalesOrderId, CustomerId, AccountNumber
SalesOrderHeader	ModifiedDate

USE AdventureWorks2017

```
SELECT O.SalesOrderID
      ,O.CustomerID
      ,O.AccountNumber
FROM Sales.SalesOrderHeaderSalesReason AS E
LEFT JOIN sales.SalesOrderHeader AS O ON E.ModifiedDate = O.ModifiedDate
WHERE O.CustomerID < 12000 and E.SalesOrderID IS NOT NULL
```

```
USE AdventureWorks2017

SELECT O.SalesOrderID
      ,O.CustomerID
      ,O.AccountNumber
FROM Sales.SalesOrderHeaderSalesReason AS E
LEFT JOIN sales.SalesOrderHeader AS O ON E.ModifiedDate = O.ModifiedDate
where O.CustomerID < 15000 and E.SalesOrderID IS NOT NULL
--FOR JSON PATH, ROOT ('temp'), INCLUDE_NULL_VALUES;
```

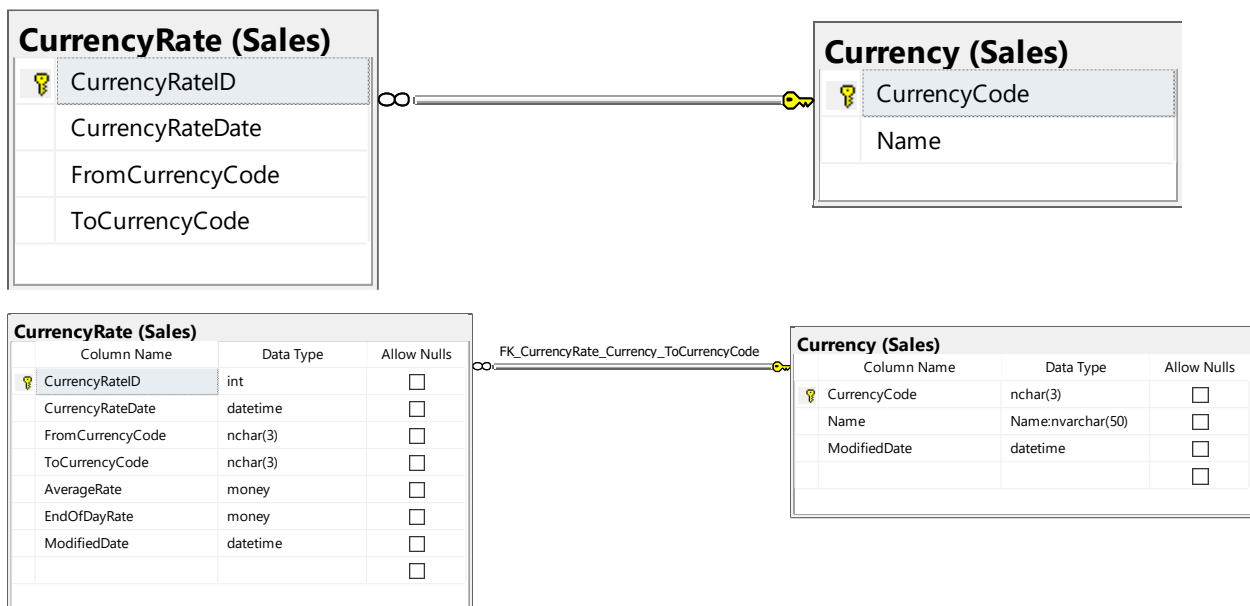
76 %

Results		Messages	
	SalesOrderID	CustomerId	AccountNumber
1	43700	14501	10-4030-014501
2	43701	11003	10-4030-011003
3	43700	14501	10-4030-014501
4	43701	11003	10-4030-011003
5	43700	14501	10-4030-014501
6	43701	11003	10-4030-011003
7	43700	14501	10-4030-014501
8	43701	11003	10-4030-011003
9	43704	11005	10-4030-011005
10	43705	11011	10-4030-011011
11	43704	11005	10-4030-011005
12	43705	11011	10-4030-011011
13	43704	11005	10-4030-011005
14	43705	11011	10-4030-011011
15	43704	11005	10-4030-011005
16	43705	11011	10-4030-011011
17	43704	11005	10-4030-011005
18	43705	11011	10-4030-011011
19	43704	11005	10-4030-011005



Proposition #4 (Simple)

Using AdventureWorks2017 make a table that joins the Currency table along with the CurrencyRate table such that the table shows CurrencyCode, Name, CurrencyRateID, AverageRate, EndOfDayRate such that the currencyrateid is less than 1000



Columns from Tables

Table Name	Column Name
Currency	CurrencyCode, Name
CurrencyRate	CurrencyRateID, AverageRate, EndOfDayRate

```

SELECT E.CurrencyCode
      ,E.Name
      ,O.CurrencyRateID
      ,O.AverageRate
      ,O.EndOfDayRate
FROM sales.Currency AS E
CROSS JOIN Sales.CurrencyRate AS O
WHERE O.CurrencyRateID < 1000

```

```

USE AdventureWorks2017

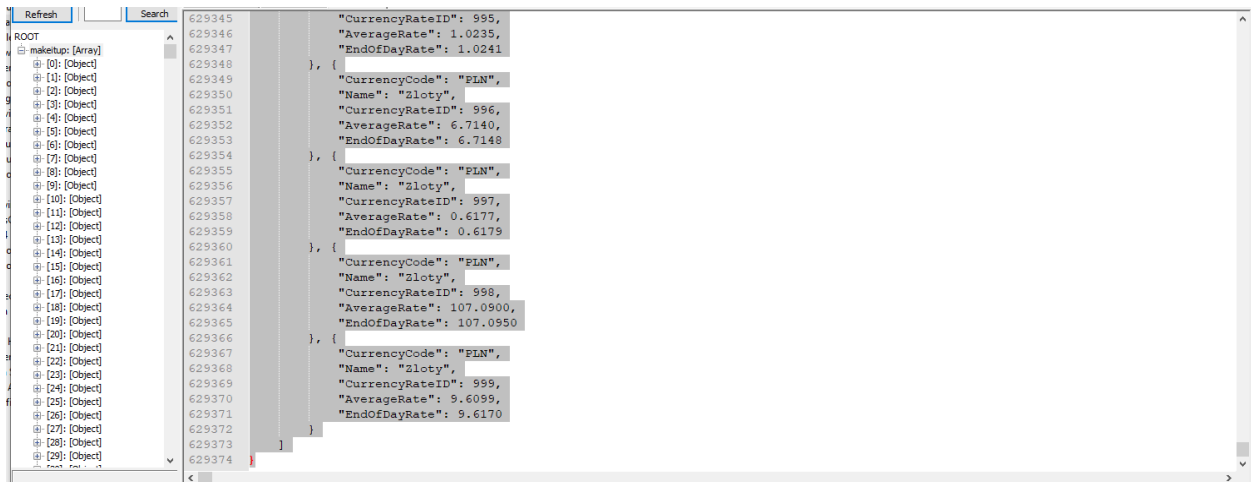
SELECT E.CurrencyCode
      ,E.Name
      ,O.CurrencyRateID
      ,O.AverageRate
      ,O.EndOfDayRate
FROM sales.Currency AS E
CROSS JOIN Sales.CurrencyRate AS O
WHERE O.CurrencyRateID < 1000
--FOR JSON PATH, ROOT('temp2'), INCLUDE_NULL_VALUES;

```

160 %

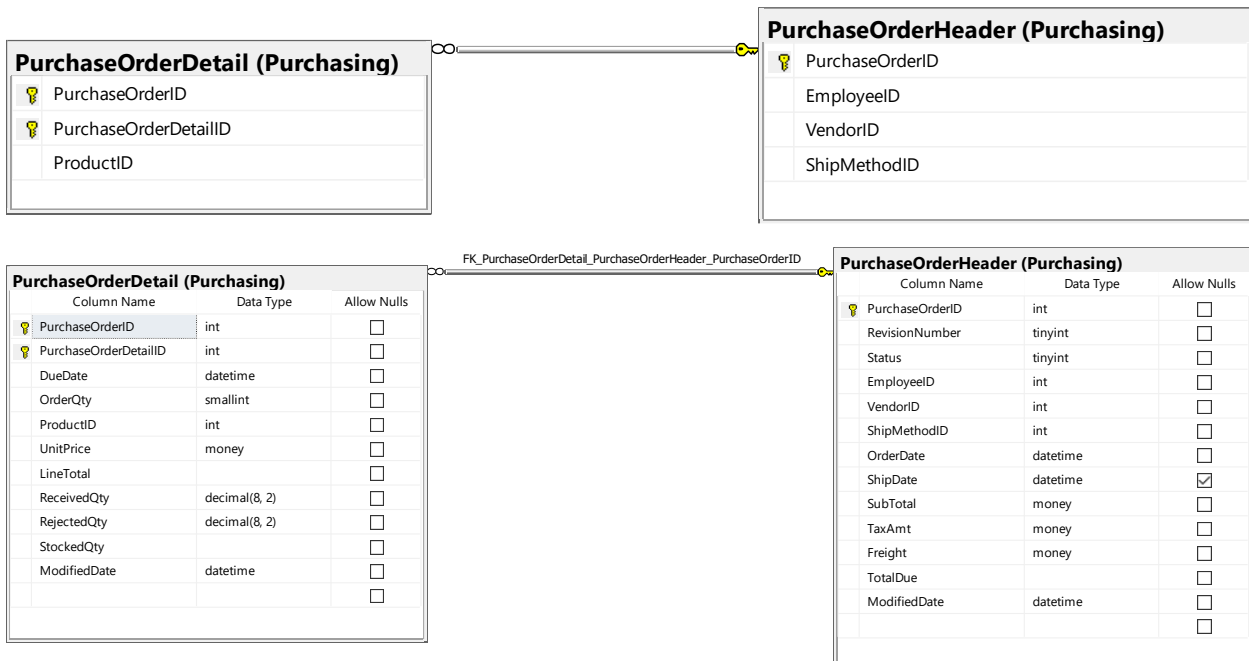
Results Messages

	CurrencyCode	Name	CurrencyRateID	AverageRate	EndOfDayRate
1	AFA	Afghani	1	1.00	1.0002
2	AFA	Afghani	2	1.5491	1.55
3	AFA	Afghani	3	1.9379	1.9419
4	AFA	Afghani	4	1.4641	1.4683
5	AFA	Afghani	5	8.2781	8.2784
6	AFA	Afghani	6	1.8967	1.8976
7	AFA	Afghani	7	0.9697	0.9703
8	AFA	Afghani	8	6.3611	6.3613
9	AFA	Afghani	9	0.6183	0.6183
10	AFA	Afghani	10	104.91	104.958
11	AFA	Afghani	11	9.374	9.384
12	AFA	Afghani	12	3.7507	3.7584
13	AFA	Afghani	13	634.5099	634.60
14	AFA	Afghani	14	1.00	0.9991
15	AFA	Afghani	15	1.5559	1.5558
16	AFA	Afghani	16	1.9339	1.933
17	AFA	Afghani	17	1.4661	1.4637
18	AFA	Afghani	18	8.2781	8.2774
19	AFA	Afghani	19	1.8924	1.8922



Proposition #5 (Simple)

Using AdventureWorks2017 make a table that joins PurchaseOrderDetail and PurchaseOrderHeader such that the table shows the PurchaseOrderID, PurchaseOrderDetailID, TotalDue, TaxAmt



Columns from Tables

Table Name	Column Name
PurchaseOrderDetail	PurchaseOrderID, PurchaseOrderDetailID
PurchaseOrderHeader	TotalDue, TaxAmt

USE AdventureWorks2017

SELECT E.PurchaseOrderID

```

,E.PurchaseOrderDetailID
,O.TotalDue
,O.TaxAmt
FROM Purchasing.PurchaseOrderDetail AS E
RIGHT JOIN Purchasing.PurchaseOrderHeader AS O ON E.PurchaseOrderID =
O.PurchaseOrderID

```

160 %

USE AdventureWorks2017

```

SELECT E.PurchaseOrderID
,E.PurchaseOrderDetailID
,O.TotalDue
,O.TaxAmt
FROM Purchasing.PurchaseOrderDetail AS E
RIGHT JOIN Purchasing.PurchaseOrderHeader AS O ON E.PurchaseOrderID = O.PurchaseOrderID
--FOR JSON PATH, ROOT('temp2'), INCLUDE_NULL_VALUES;

```

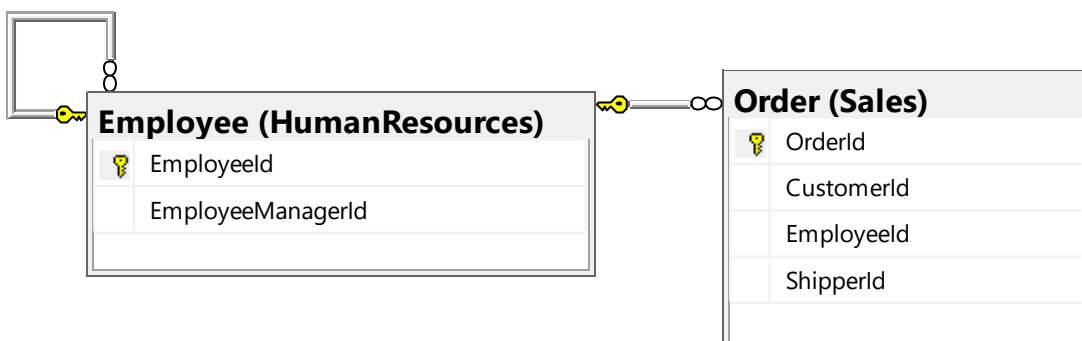
Results Messages

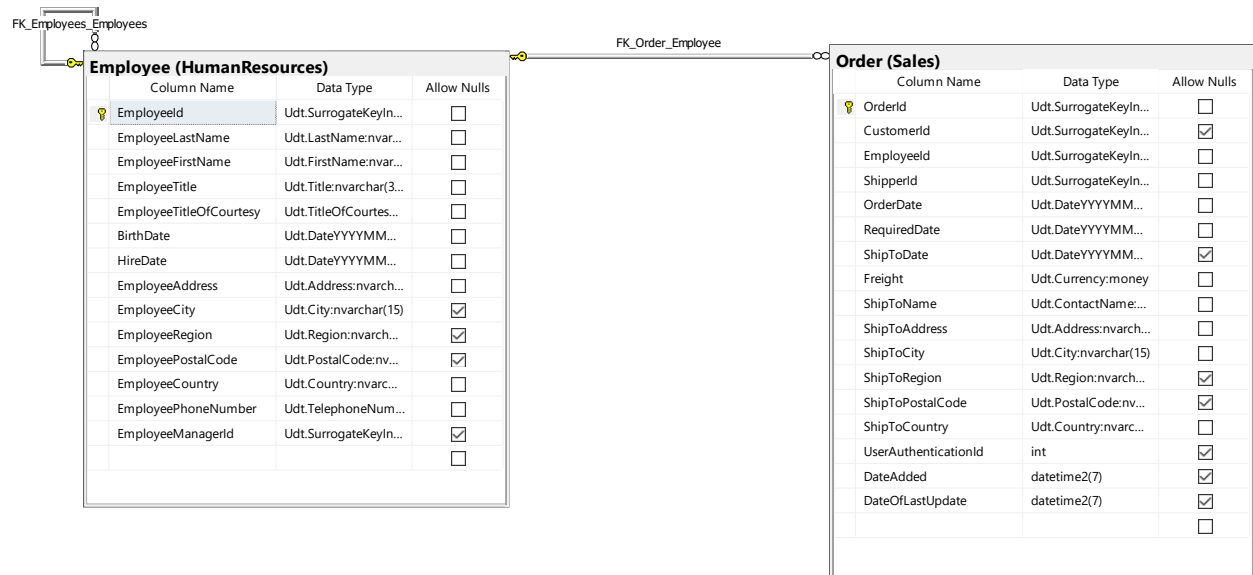
	PurchaseOrderID	PurchaseOrderDetailID	TotalDue	TaxAmt
1	1	1	222.1492	16.0832
2	2	2	300.6721	21.7681
3	2	3	300.6721	21.7681
4	3	4	9776.2665	707.784
5	4	5	189.0395	13.6861
6	5	6	22539.0165	1631.784
7	6	7	16164.0229	1170.246
8	7	8	64847.5328	4694.844
9	7	9	64847.5328	4694.844
10	7	10	64847.5328	4694.844
11	8	11	766.1827	55.4702
12	8	12	766.1827	55.4702
13	8	13	766.1827	55.4702
14	8	14	766.1827	55.4702
15	8	15	766.1827	55.4702
16	9	16	767.0528	55.5332
17	9	17	767.0528	55.5332
18	9	18	767.0528	55.5332
19	9	19	767.0528	55.5332

Refresh	Search	44180	"TotalDue": 41536.0000,
		44181	"TaxAmt": 3020.8000
		44182	}, {
		44183	"PurchaseOrderID": 4011,
		44184	"PurchaseOrderDetailID": 8837,
		44185	"TotalDue": 59941.7500,
		44186	"TaxAmt": 4359.4000
		44187	}, {
		44188	"PurchaseOrderID": 4011,
		44189	"PurchaseOrderDetailID": 8838,
		44190	"TotalDue": 59941.7500,
		44191	"TaxAmt": 4359.4000
		44192	}, {
		44193	"PurchaseOrderID": 4011,
		44194	"PurchaseOrderDetailID": 8839,
		44195	"TotalDue": 59941.7500,
		44196	"TaxAmt": 4359.4000
		44197	}, {
		44198	"PurchaseOrderID": 4011,
		44199	"PurchaseOrderDetailID": 8840,
		44200	"TotalDue": 59941.7500,
		44201	"TaxAmt": 4359.4000
		44202	}, {
		44203	"PurchaseOrderID": 4011,
		44204	"PurchaseOrderDetailID": 8841,
		44205	"TotalDue": 59941.7500,
		44206	"TaxAmt": 4359.4000
		44207	}, {
		44208	"PurchaseOrderID": 4012,
		44209	"PurchaseOrderDetailID": 8842,
		44210	"TotalDue": 1097448.0000,
		44211	"TaxAmt": 79814.4000
		44212	}, {
		44213	"PurchaseOrderID": 4012,
		44214	"PurchaseOrderDetailID": 8843,
		44215	"TotalDue": 1097448.0000,
		44216	"TaxAmt": 79814.4000
		44217	}, {
		44218	"PurchaseOrderID": 4012,
		44219	"PurchaseOrderDetailID": 8844,
		44220	"TotalDue": 1097448.0000,
		44221	"TaxAmt": 79814.4000

Proposition #6 (Medium)

Using northwinds create a table that shows the number of employees in each of their respective city





Columns from Tables

Table Name	Column Name
Order	EmployeeId
Employee	EmployeeCity

USE Northwinds2020TSQVLV6;

```

SELECT O.EmployeeCity
        ,count(E.EmployeeId) AS NumberOfEmployees
FROM Sales.[Order] AS E
INNER JOIN HumanResources.Employee AS O ON E.EmployeeId = O.EmployeeId
GROUP BY O.EmployeeCity

```

```

USE Northwinds2020TSQLV6;

SELECT O.EmployeeCity
      ,count(E.EmployeeId) AS NumberOfEmployees
FROM Sales.[Order] AS E
INNER JOIN HumanResources.Employee AS O ON E.EmployeeId = O.EmployeeId
GROUP BY O.EmployeeCity
--FOR JSON PATH, ROOT('temp2'), INCLUDE_NULL_VALUES;

```

160 %

Results Messages

	EmployeeCity	NumberOfEmployees
1	Kirkland	127
2	London	224
3	Redmond	156
4	Seattle	227
5	Tacoma	96

Refresh

Search

ROOT

```

temp2: [Array]
  [0]: [Object]
  [1]: [Object]
  [2]: [Object]
  [3]: [Object]
  [4]: [Object]

```

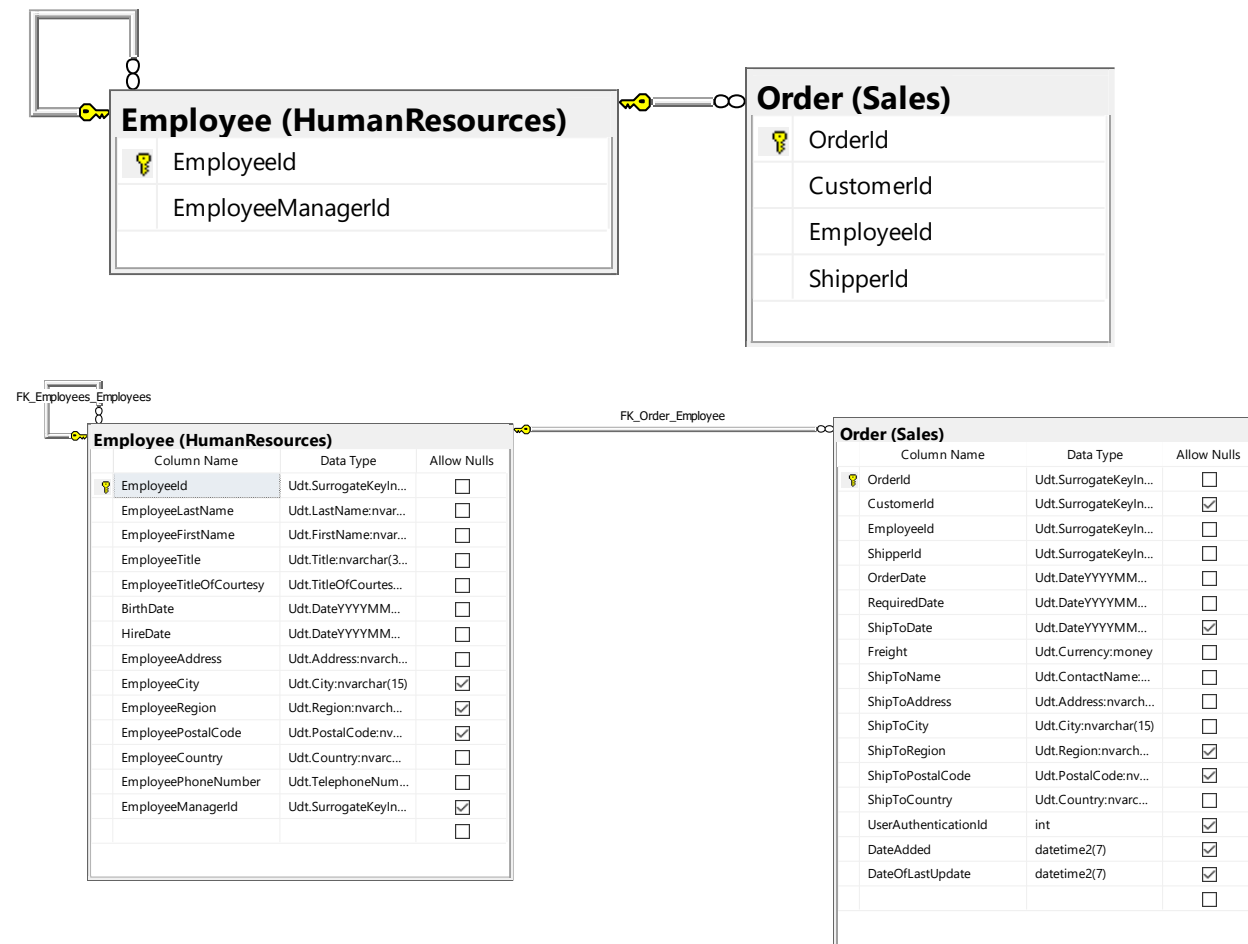
```

1
2  "temp2": [{
3      "EmployeeCity": "Kirkland",
4      "NumberOfEmployees": 127
5    }, {
6      "EmployeeCity": "London",
7      "NumberOfEmployees": 224
8    }, {
9      "EmployeeCity": "Redmond",
10     "NumberOfEmployees": 156
11    }, {
12     "EmployeeCity": "Seattle",
13     "NumberOfEmployees": 227
14    }, {
15     "EmployeeCity": "Tacoma",
16     "NumberOfEmployees": 96
17    }
18  ]
19

```

Proposition #7 (Medium)

Using northwinds Make a table that shows the amount of times an employee's last name appears within the table



Columns from Tables

Table Name	Column Name
Order	EmployeeId, ShipToCity
Employee	EmployeeLastName

USE Northwinds2020TSQLV6;

SELECT count(E.ShipToCity) AS appearances
, O.EmployeeLastName

FROM Sales.[Order] AS E

INNER JOIN HumanResources.Employee AS O ON E.EmployeeId = O.EmployeeId

GROUP BY O.EmployeeLastName


```

USE Northwinds2020TSQLV6;

SELECT count(E.ShipToCity) AS appearances
      ,O.EmployeeLastName
FROM Sales.[Order] AS E
INNER JOIN HumanResources.Employee AS O ON E.EmployeeId = O.EmployeeId
GROUP BY O.EmployeeLastName
--FOR JSON PATH, ROOT('temp2'), INCLUDE_NULL_VALUES;

```

160 %

Results Messages

	appearances	EmployeeLastName
1	104	Cameron
2	123	Davis
3	43	Doyle
4	96	Funk
5	72	King
6	127	Lew
7	42	Mortensen
8	156	Peled
9	67	Suurs

Refresh Search

ROOT
 temp2: [Array]
 [0]: [Object]
 appearances: 104
 EmployeeLastName: "Cameron"
 [1]: [Object]
 appearances: 123
 EmployeeLastName: "Davis"
 [2]: [Object]
 appearances: 43
 EmployeeLastName: "Doyle"
 [3]: [Object]
 appearances: 96
 EmployeeLastName: "Funk"
 [4]: [Object]
 [5]: [Object]
 [6]: [Object]
 [7]: [Object]
 [8]: [Object]

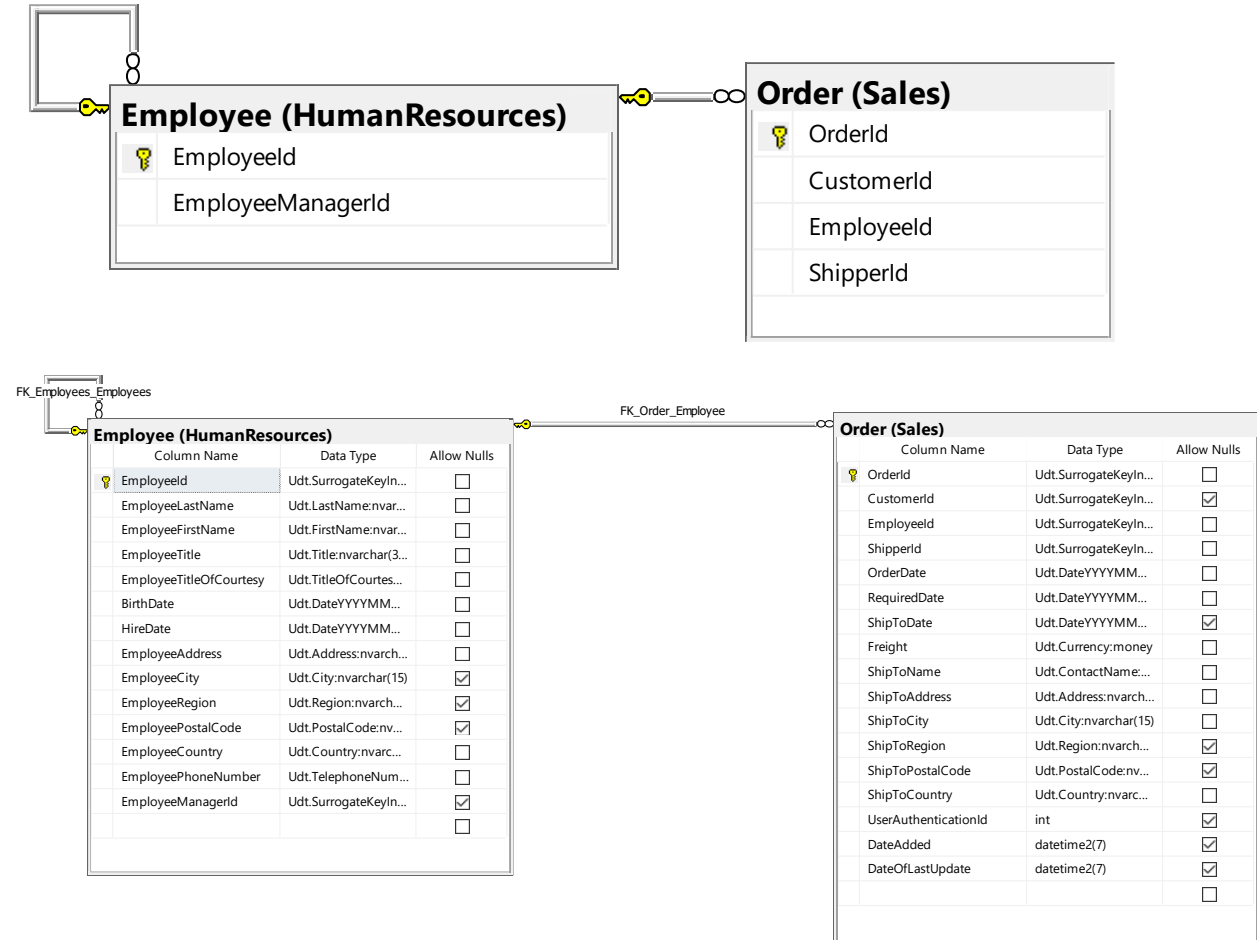
```

1
2      "temp2": [{
3          "appearances": 104,
4          "EmployeeLastName": "Cameron"
5      }, {
6          "appearances": 123,
7          "EmployeeLastName": "Davis"
8      }, {
9          "appearances": 43,
10         "EmployeeLastName": "Doyle"
11     }, {
12         "appearances": 96,
13         "EmployeeLastName": "Funk"
14     }, {
15         "appearances": 72,
16         "EmployeeLastName": "King"
17     }, {
18         "appearances": 127,
19         "EmployeeLastName": "Lew"
20     }, {
21         "appearances": 42,
22         "EmployeeLastName": "Mortensen"
23     }, {
24         "appearances": 156,
25         "EmployeeLastName": "Peled"
26     }, {
27         "appearances": 67,
28         "EmployeeLastName": "Suurs"
29     }
30 ]
31

```

Proposition #8 (Medium)

Using Northwind Create a table that shows the amount of times an Employee's first name appeared in each country.



Columns from Tables

Table Name	Column Name
Order	EmployeeId, ShipToCountry
Employee	EmployeeFirstName

```
USE Northwinds2020TSQLV6;
```

```
SELECT count(E.EmployeeId) AS appearances
      ,O.EmployeeFirstName
      ,E.ShipToCountry
FROM Sales.[Order] AS E
INNER JOIN HumanResources.Employee AS O ON E.EmployeeId = O.EmployeeId
GROUP BY O.EmployeeFirstName
      ,E.ShipToCountry
```

```

USE Northwinds2020TSQLV6;

SELECT count(E.EmployeeId) AS appearances
      ,O.EmployeeFirstName
      ,E.ShipToCountry
FROM Sales.[Order] AS E
INNER JOIN HumanResources.Employee AS O ON E.EmployeeId = O.EmployeeId
GROUP BY O.EmployeeFirstName
      ,E.ShipToCountry
FOR JSON PATH, ROOT('temp2'), INCLUDE_NULL_VALUES;

```

160 %

Results Messages

	appearances	EmployeeFirstName	ShipToCountry
1	1	Don	Argentina
2	1	Judy	Argentina
3	3	Maria	Argentina
4	2	Patricia	Argentina
5	1	Paul	Argentina
6	3	Russell	Argentina
7	1	Sara	Argentina
8	4	Yael	Argentina
9	6	Don	Austria
10	5	Judy	Austria
11	5	Maria	Austria
12	3	Patricia	Austria
13	4	Paul	Austria
14	6	Russell	Austria
15	5	Sara	Austria
16	6	Yael	Austria
17	2	Don	Belgium
18	1	Judy	Belgium
19	2	Patricia	Belgium

Refresh
Search

temp2: [Array]
[0]: [Object]
appearances: 1
EmployeeFirstName: "Dor"
ShipToCountry: "Argentina"
[1]: [Object]
appearances: 1
EmployeeFirstName: "Jud"
ShipToCountry: "Argentina"
[2]: [Object]
appearances: 3
EmployeeFirstName: "Mar"
ShipToCountry: "Argentina"
[3]: [Object]
[4]: [Object]
[5]: [Object]
[6]: [Object]
[7]: [Object]
[8]: [Object]
[9]: [Object]
[10]: [Object]
[11]: [Object]
[12]: [Object]
[13]: [Object]
[14]: [Object]
[15]: [Object]
[16]: [Object]
[17]: [Object]
[18]: [Object]
[19]: [Object]
[20]: [Object]
[21]: [Object]
[22]: [Object]
[23]: [Object]
[24]: [Object]
[25]: [Object]
[26]: [Object]
[27]: [Object]
[28]: [Object]
[29]: [Object]
[30]: [Object]
[31]: [Object]
[32]: [Object]
[33]: [Object]
[34]: [Object]
[35]: [Object]
[36]: [Object]
[37]: [Object]
[38]: [Object]
[39]: [Object]
[40]: [Object]
[41]: [Object]
[42]: [Object]
[43]: [Object]

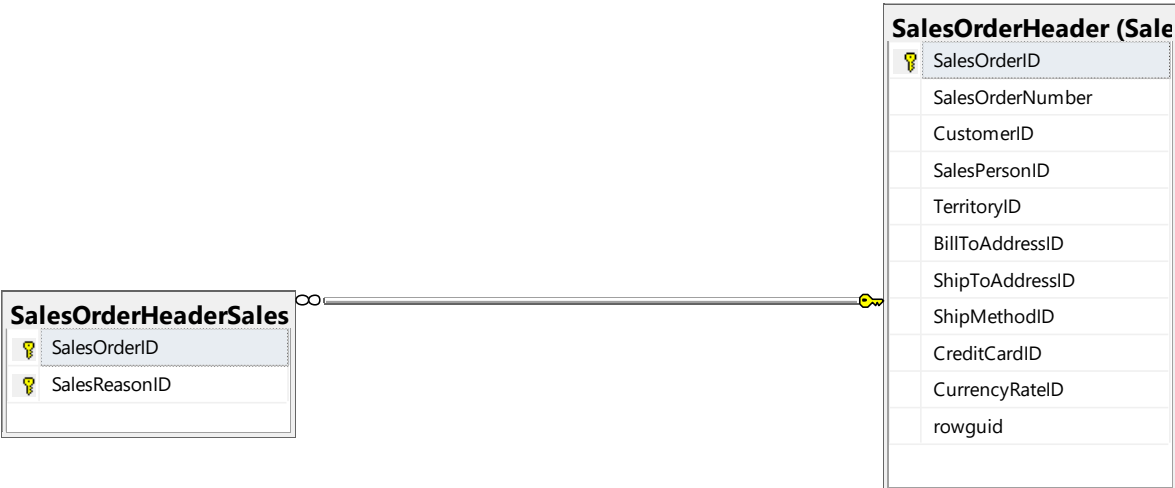
```

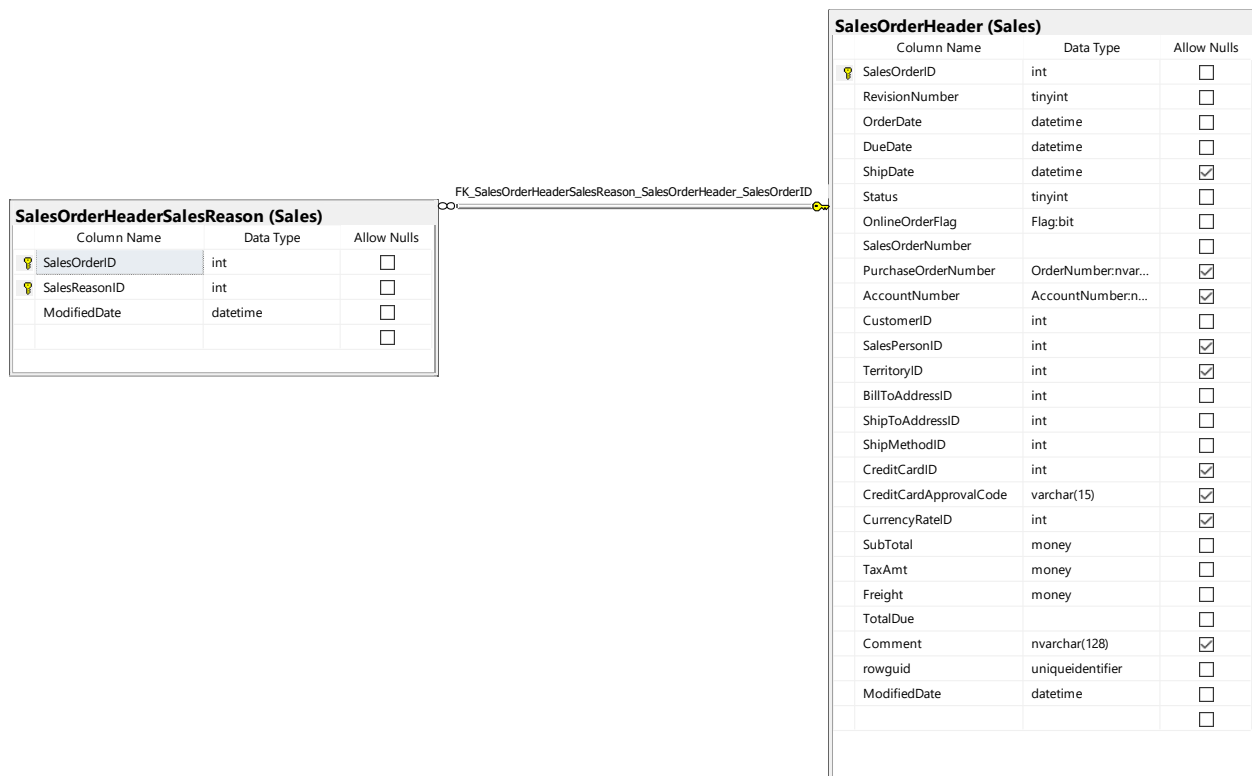
640      "EmployeeFirstName": "Judy",
641      "ShipToCountry": "Venezuela"
642    }, {
643      "appearances": 9,
644      "EmployeeFirstName": "Maria",
645      "ShipToCountry": "Venezuela"
646    }, {
647      "appearances": 1,
648      "EmployeeFirstName": "Patricia",
649      "ShipToCountry": "Venezuela"
650    }, {
651      "appearances": 2,
652      "EmployeeFirstName": "Paul",
653      "ShipToCountry": "Venezuela"
654    }, {
655      "appearances": 3,
656      "EmployeeFirstName": "Russell",
657      "ShipToCountry": "Venezuela"
658    }, {
659      "appearances": 8,
660      "EmployeeFirstName": "Sara",
661      "ShipToCountry": "Venezuela"
662    }, {
663      "appearances": 3,
664      "EmployeeFirstName": "Sven",
665      "ShipToCountry": "Venezuela"
666    }, {
667      "appearances": 8,
668      "EmployeeFirstName": "Yael",
669      "ShipToCountry": "Venezuela"
670    }
671  ]
672

```

Proposition #9 (Medium)

using Adventureworks make a table that returns the number of sales for each ID along with the total due for it





Columns from Tables

Table Name	Column Name
SalesOrderHeaderSalesReason	SalesReasonId
SalesOrderHeader	SalesOrderId, TotalDue

Order By

Table Name	Column Name	Sort Order
SalesOrderHeader	TotalDue	ASCENDING

USE AdventureWorks2017

```

SELECT COUNT(O.SalesOrderID) AS NumberOfSales,
       E.SalesReasonID,
       O.TotalDue
FROM Sales.SalesOrderHeaderSalesReason AS E
LEFT JOIN sales.SalesOrderHeader AS O ON E.ModifiedDate = O.ModifiedDate
WHERE O.ModifiedDate = (
    SELECT MIN(O.ModifiedDate)
    FROM sales.SalesOrderHeader AS O
)
GROUP BY E.SalesReasonID,
         O.TotalDue
ORDER BY O.TotalDue;

```

USE AdventureWorks2017

```
SELECT COUNT(O.SalesOrderID) AS NumberOfSales,
       E.SalesReasonID,
       O.TotalDue
FROM Sales.SalesOrderHeaderSalesReason AS E
LEFT JOIN sales.SalesOrderHeader AS O ON E.ModifiedDate = O.ModifiedDate
WHERE O.ModifiedDate = (
    SELECT MIN(O.ModifiedDate)
    FROM sales.SalesOrderHeader AS O
)
GROUP BY E.SalesReasonID,
         O.TotalDue
ORDER BY O.TotalDue;
```

160 %

Results Messages

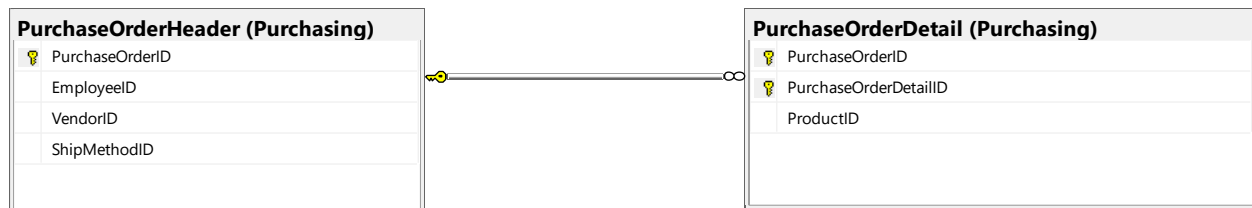
	NumberOfSales	SalesReasonID	TotalDue
1	4	5	472.3108
2	4	9	472.3108
3	2	5	772.5036
4	2	9	772.5036
5	2	5	807.2585
6	2	9	807.2585
7	2	5	985.018
8	2	9	985.018
9	2	5	1416.9322
10	2	9	1416.9322
11	2	5	1457.3288
12	2	9	1457.3288
13	2	5	1460.6061
14	2	9	1460.6061
15	2	5	1481.1742
16	2	9	1481.1742
17	2	5	2955.0542
18	2	9	2955.0542
19	2	5	3082.0191

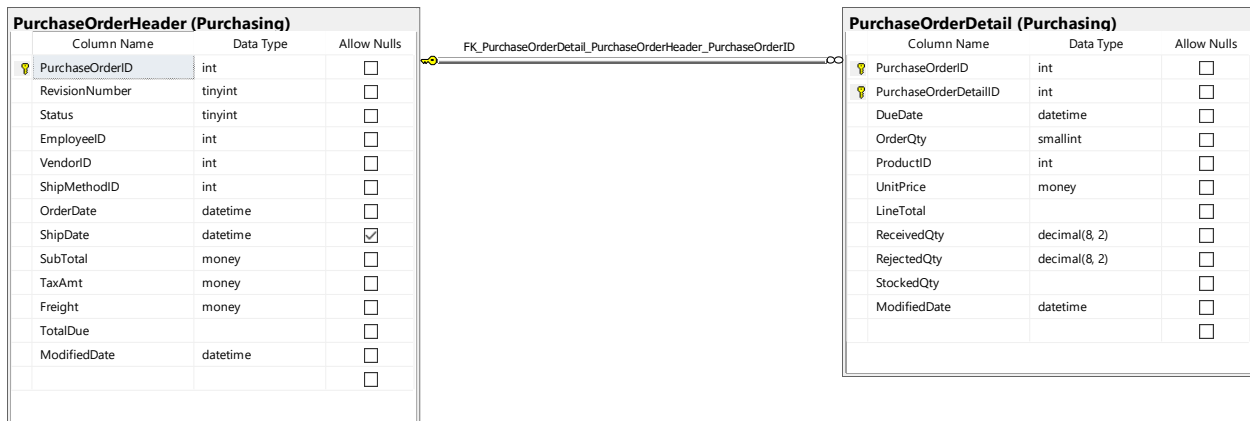
Refresh	Search	
ROOT		
[-] [Array]		
[-] [0]: [Object]		
NumberOfSales: 4		
SalesReasonID: 5		
TotalDue: 472.3108		
[-] [1]: [Object]		
NumberOfSales: 4		
SalesReasonID: 9		
TotalDue: 472.3108		
[-] [2]: [Object]		
[-] [3]: [Object]		
[-] [4]: [Object]		
[-] [5]: [Object]		
[-] [6]: [Object]		
[-] [7]: [Object]		
[-] [8]: [Object]		
[-] [9]: [Object]		
[-] [10]: [Object]		
[-] [11]: [Object]		
[-] [12]: [Object]		
[-] [13]: [Object]		
[-] [14]: [Object]		
[-] [15]: [Object]		
[-] [16]: [Object]		
[-] [17]: [Object]		
[-] [18]: [Object]		
[-] [19]: [Object]		
[-] [20]: [Object]		
[-] [21]: [Object]		
[-] [22]: [Object]		
[-] [23]: [Object]		
[-] [24]: [Object]		
[-] [25]: [Object]		
[-] [26]: [Object]		
[-] [27]: [Object]		
[-] [28]: [Object]		
[-] [29]: [Object]		
[-] [30]: [Object]		
[-] [31]: [Object]		
[-] [32]: [Object]		
[-] [33]: [Object]		
[-] [34]: [Object]		
[-] [35]: [Object]		
[-] [36]: [Object]		
[-] [37]: [Object]		
[-] [38]: [Object]		
[-] [39]: [Object]		
[-] [40]: [Object]		
[-] [41]: [Object]		
[-] [42]: [Object]		
[-] [43]: [Object]		
[-] [44]: [Object]		
[-] [45]: [Object]		
[-] [46]: [Object]		

292	"SalesReasonID": 5,
293	"TotalDue": 40487.7233
294	}, {
295	"NumberOfSales": 2,
296	"SalesReasonID": 9,
297	"TotalDue": 40487.7233
298	}, {
299	"NumberOfSales": 2,
300	"SalesReasonID": 5,
301	"TotalDue": 43362.4196
302	}, {
303	"NumberOfSales": 2,
304	"SalesReasonID": 9,
305	"TotalDue": 43362.4196
306	}, {
307	"NumberOfSales": 2,
308	"SalesReasonID": 5,
309	"TotalDue": 44344.8265
310	}, {
311	"NumberOfSales": 2,
312	"SalesReasonID": 9,
313	"TotalDue": 44344.8265
314	}, {
315	"NumberOfSales": 2,
316	"SalesReasonID": 5,
317	"TotalDue": 48204.0662
318	}, {
319	"NumberOfSales": 2,
320	"SalesReasonID": 9,
321	"TotalDue": 48204.0662
322	}
323	}
324]

Proposition #10 (Medium)

Using AdventureWorks2017 make a table that joins PurchaseOrderDetail and PurchaseOrderHeader such that the table shows the PurchaseOrderID, PurchaseOrderDetailID, TotalDue, TaxAmt order the table by the tax amount





Columns from Tables

Table Name	Column Name
PurchaseOrderDetail	PurchaseOrderId, PurchaseOrderDetailId
PurchaseOrderHeader	TaxAmt, TotalDue

Order By

Table Name	Column Name	Sort Order
PurchaseOrderHeader	TotalDue	ASCENDING

USE AdventureWorks2017

```

SELECT E.PurchaseOrderID,
        E.PurchaseOrderDetailID,
        O.TotalDue,
        O.TaxAmt
FROM Purchasing.PurchaseOrderDetail AS E
RIGHT JOIN Purchasing.PurchaseOrderHeader AS O ON E.PurchaseOrderID =
O.PurchaseOrderID
GROUP BY E.PurchaseOrderID,
        E.PurchaseOrderDetailID,
        O.TotalDue,
        O.TaxAmt
ORDER BY O.TotalDue;
  
```


USE AdventureWorks2017

```
SELECT E.PurchaseOrderID,
       E.PurchaseOrderDetailID,
       O.TotalDue,
       O.TaxAmt
FROM Purchasing.PurchaseOrderDetail AS E
RIGHT JOIN Purchasing.PurchaseOrderHeader AS O ON E.PurchaseOrderID = O.PurchaseOrderID
GROUP BY E.PurchaseOrderID,
         E.PurchaseOrderDetailID,
         O.TotalDue,
         O.TaxAmt
ORDER BY O.TotalDue;
```

FOR JSON PATH, ROOT(''), INCLUDE_NULL_VALUES;

	PurchaseOrderID	PurchaseOrderDetailID	TotalDue	TaxAmt
1	59	134	40.9684	2.966
2	138	333	40.9684	2.966
3	217	517	40.9684	2.966
4	296	676	40.9684	2.966
5	375	842	40.9684	2.966
6	484	1072	40.9684	2.966
7	567	1265	40.9684	2.966
8	654	1460	40.9684	2.966
9	733	1624	40.9684	2.966
10	816	1822	40.9684	2.966
11	899	2016	40.9684	2.966
12	990	2206	40.9684	2.966
13	1018	2295	40.9684	2.966
14	1145	2589	40.9684	2.966
15	1224	2766	40.9684	2.966
16	1303	2941	40.9684	2.966
17	1382	3132	40.9684	2.966
18	1461	3302	40.9684	2.966
19	1540	3464	40.9684	2.966

Refresh Search

ROOT: [Array]

[0]: [Object]

- PurchaseOrderID: 59
- PurchaseOrderDetailID: 1
- TotalDue: 40.9684
- TaxAmt: 2.9660

[1]: [Object]

- PurchaseOrderID: 138
- PurchaseOrderDetailID: 1
- TotalDue: 40.9684
- TaxAmt: 2.9660

[2]: [Object]

- PurchaseOrderID: 217
- PurchaseOrderDetailID: 1
- TotalDue: 40.9684
- TaxAmt: 2.9660

[3]: [Object]

[4]: [Object]

[5]: [Object]

[6]: [Object]

[7]: [Object]

[8]: [Object]

[9]: [Object]

[10]: [Object]

[11]: [Object]

[12]: [Object]

[13]: [Object]

[14]: [Object]

[15]: [Object]

[16]: [Object]

[17]: [Object]

[18]: [Object]

[19]: [Object]

[20]: [Object]

[21]: [Object]

[22]: [Object]

[23]: [Object]

[24]: [Object]

[25]: [Object]

[26]: [Object]

[27]: [Object]

[28]: [Object]

[29]: [Object]

[30]: [Object]

[31]: [Object]

[32]: [Object]

[33]: [Object]

[34]: [Object]

[35]: [Object]

[36]: [Object]

[37]: [Object]

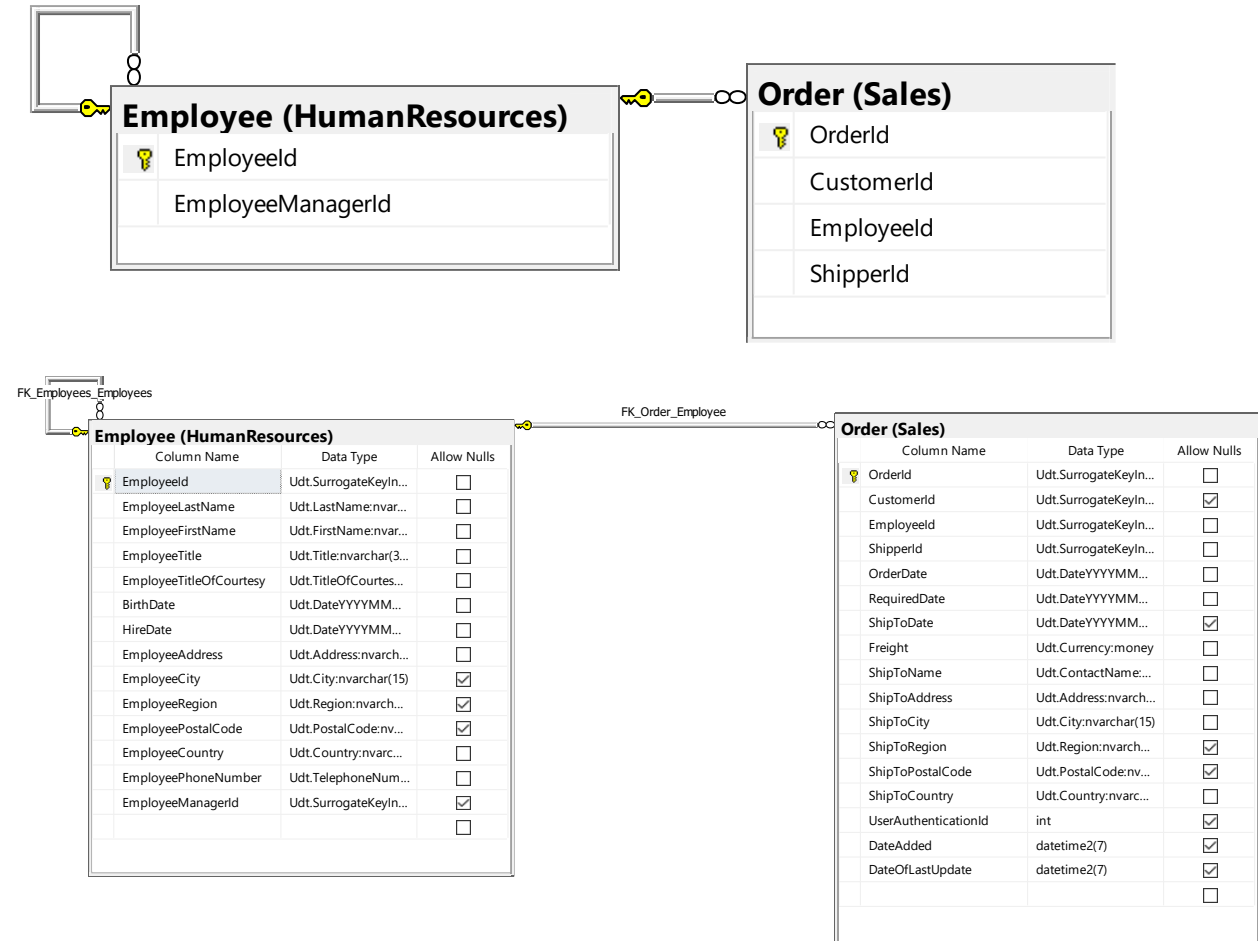
[38]: [Object]

[39]: [Object]

```
44197 }, {
44198   "PurchaseOrderID": 4007,
44199   "PurchaseOrderDetailID": 8814,
44200   "TotalDue": 609422.0000,
44201   "TaxAmt": 44321.6000
44202 }, {
44203   "PurchaseOrderID": 4007,
44204   "PurchaseOrderDetailID": 8815,
44205   "TotalDue": 609422.0000,
44206   "TaxAmt": 44321.6000
44207 }, {
44208   "PurchaseOrderID": 4012,
44209   "PurchaseOrderDetailID": 8842,
44210   "TotalDue": 1097448.0000,
44211   "TaxAmt": 79814.4000
44212 }, {
44213   "PurchaseOrderID": 4012,
44214   "PurchaseOrderDetailID": 8843,
44215   "TotalDue": 1097448.0000,
44216   "TaxAmt": 79814.4000
44217 }, {
44218   "PurchaseOrderID": 4012,
44219   "PurchaseOrderDetailID": 8844,
44220   "TotalDue": 1097448.0000,
44221   "TaxAmt": 79814.4000
44222 }, {
44223   "PurchaseOrderID": 4012,
44224   "PurchaseOrderDetailID": 8845,
44225   "TotalDue": 1097448.0000,
44226   "TaxAmt": 79814.4000
44227 }
44228 ]
44229 }
```

Proposition #11 (Medium)

Using northWinds Make a table that shows the full name of each employee along with their employeeid



Columns from Tables

Table Name	Column Name
Order	EmployeeId
Employee	EmployeeFirstName, EmployeeLastName

Order By

Table Name	Column Name	Sort Order
Employee	EmployeeFirstName	ASCENDING

USE Northwinds2020TSQLV6;

```

SELECT CONCAT (
    O.EmployeeFirstName
    , ' '
    , O.EmployeeLastName
) AS Name
, E.EmployeeId
FROM Sales.[Order] AS E
INNER JOIN HumanResources.Employee AS O ON E.EmployeeId = O.EmployeeId
GROUP BY O.EmployeeFirstName
, E.EmployeeId
, O.EmployeeLastName
ORDER BY O.EmployeeFirstName

```

```

USE Northwinds2020TSQLV6;

SELECT CONCAT (
    O.EmployeeFirstName
    , ' '
    , O.EmployeeLastName
) AS Name
, E.EmployeeId
FROM Sales.[Order] AS E
INNER JOIN HumanResources.Employee AS O ON E.EmployeeId = O.EmployeeId
GROUP BY O.EmployeeFirstName
, E.EmployeeId
, O.EmployeeLastName
ORDER BY O.EmployeeFirstName
FOR JSON PATH, ROOT('temp2'), INCLUDE_NULL_VALUES;

```

160 %

Results Messages

	Name	EmployeeId
1	Don Funk	2
2	Judy Lew	3
3	Maria Cameron	8
4	Patricia Doyle	9
5	Paul Suurs	6
6	Russell King	7
7	Sara Davis	1
8	Sven Mortensen	5
9	Yael Peled	4

Refresh

Search

ROOT

temp2: [Array]

[0]: [Object]

Name: "Don Funk"

EmployeeId: 2

[1]: [Object]

Name: "Judy Lew"

EmployeeId: 3

[2]: [Object]

Name: "Maria Cameron"

EmployeeId: 8

[3]: [Object]

Name: "Patricia Doyle"

EmployeeId: 9

[4]: [Object]

Name: "Paul Suurs"

EmployeeId: 6

[5]: [Object]

Name: "Russell King"

EmployeeId: 7

[6]: [Object]

Name: "Sara Davis"

EmployeeId: 1

[7]: [Object]

Name: "Sven Mortensen"

EmployeeId: 5

[8]: [Object]

Name: "Yael Peled"

EmployeeId: 4

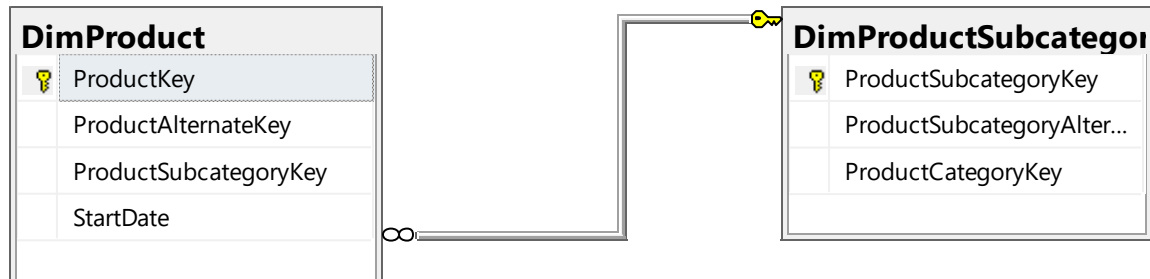
```

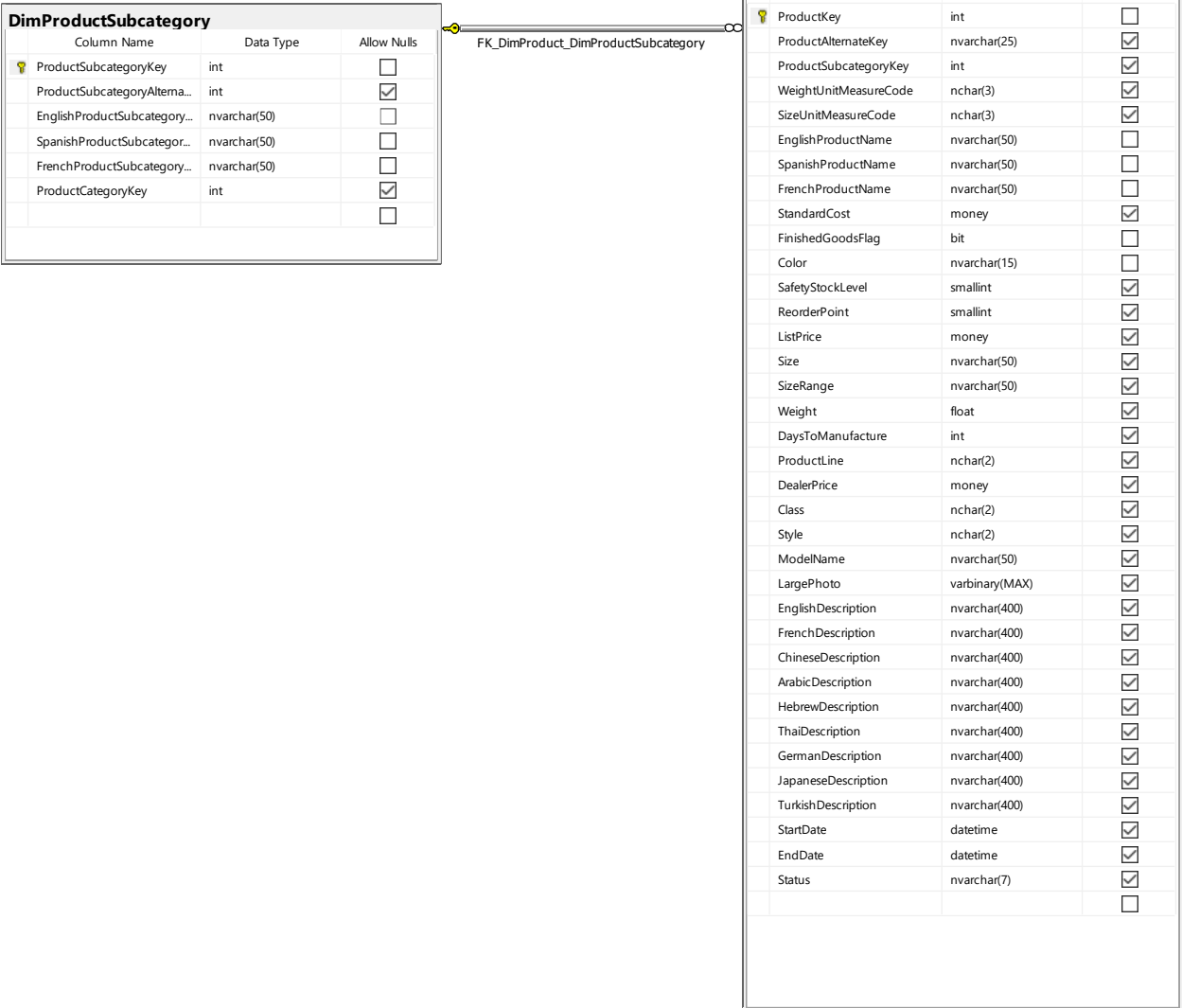
1  "temp2": [{
2
3      "Name": "Don Funk",
4      "EmployeeId": 2
5  }, {
6      "Name": "Judy Lew",
7      "EmployeeId": 3
8  }, {
9      "Name": "Maria Cameron",
10     "EmployeeId": 8
11  }, {
12     "Name": "Patricia Doyle",
13     "EmployeeId": 9
14  }, {
15     "Name": "Paul Suurs",
16     "EmployeeId": 6
17  }, {
18     "Name": "Russell King",
19     "EmployeeId": 7
20  }, {
21     "Name": "Sara Davis",
22     "EmployeeId": 1
23  }, {
24     "Name": "Sven Mortensen",
25     "EmployeeId": 5
26  }, {
27     "Name": "Yael Peled",
28     "EmployeeId": 4
29  }
30  ]
31

```

Proposition #12 (Medium)

Using AdventureWorksDW2017 make a table that outputs the english name of a product, the spanish name and its spanish subcategory name when spanish product name exists





Columns from Tables

Table Name	Column Name
DimProduct	EnglishProductName, SpanishProductName
DimProductSubcategory	SpanishProductSubcategoryName

Order By

Table Name	Column Name	Sort Order
DimProduct	EnglishProductName	ASC

```

SELECT E.EnglishProductName,
       E.SpanishProductName,
       O.SpanishProductSubcategoryName
FROM dbo.DimProduct AS E
INNER JOIN dbo.DimProductSubcategory AS O ON E.ProductSubcategoryKey =
O.ProductSubcategoryKey
WHERE SpanishProductName IS NOT NULL
GROUP BY E.EnglishProductName,
         E.SpanishProductName,
         O.SpanishProductSubcategoryName
ORDER BY E.EnglishProductName

```

SQLQuery1.sql - 10...K5UW2U1 / (sa (04))

USE AdventureWorksDW2017

```

SELECT E.EnglishProductName,
       E.SpanishProductName,
       O.SpanishProductSubcategoryName
FROM dbo.DimProduct AS E
INNER JOIN dbo.DimProductSubcategory AS O ON E.ProductSubcategoryKey = O.ProductSubcategoryKey
WHERE SpanishProductName IS NOT NULL
GROUP BY E.EnglishProductName,
         E.SpanishProductName,
         O.SpanishProductSubcategoryName
ORDER BY E.EnglishProductName

```

FOR JSON PATH, ROOT(''), INCLUDE_NULL_VALUES;

160 %

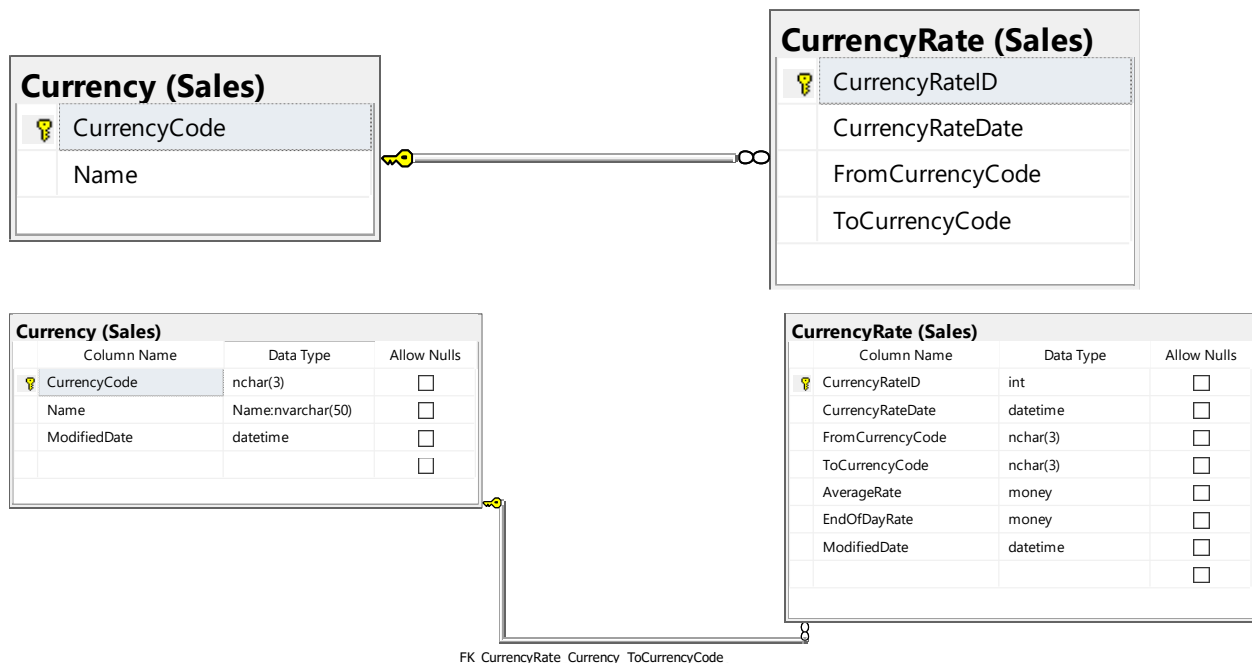
Results Messages

	EnglishProductName	SpanishProductName	SpanishProductSubcategoryName
1	All-Purpose Bike Stand	Soporte multiusos para bicicletas	Soporte para bicicletas
2	AWC Logo Cap		Gorra
3	Bike Wash - Dissolver	Lavado de bicicletas: disolvente	Limpiador
4	Cable Lock	Cable antirrobo	Candado
5	Chain	Cadena	Cadena
6	Classic Vest, L	Camiseta clásica, G	Camiseta
7	Classic Vest, M	Camiseta clásica, M	Camiseta
8	Classic Vest, S	Camiseta clásica, P	Camiseta
9	Fender Set - Mountain	Conjunto de guardabarros: montaña	Guardabarros
10	Front Brakes	Frenos delanteros	Frenos
11	Front Derailleur	Desviador delantero	Desviador
12	Full-Finger Gloves, L	Guantes completos, G	Guantes
13	Full-Finger Gloves, M	Guantes completos, M	Guantes
14	Full-Finger Gloves, S	Guantes completos, P	Guantes
15	Half-Finger Gloves, L		Guantes
16	Half-Finger Gloves, M		Guantes
17	Half-Finger Gloves, S		Guantes
18	Headlights - Dual-Beam	Luces: doble haz	Luz
19	Headlights - Weatherproof	Luces: resistentes al agua	Luz

Refresh	Search	1152	"SpanishProductName": "Cesta de paseo, grande",
		1153	"SpanishProductSubcategoryName": "Cesta"
		1154	}, {
		1155	"EnglishProductName": "Water Bottle - 30 oz.",
		1156	"SpanishProductName": "",
		1157	"SpanishProductSubcategoryName": "Portabotellas y botella"
		1158	}, {
		1159	"EnglishProductName": "Women's Mountain Shorts, L",
		1160	"SpanishProductName": "",
		1161	"SpanishProductSubcategoryName": "Pantalones cortos"
		1162	}, {
		1163	"EnglishProductName": "Women's Mountain Shorts, M",
		1164	"SpanishProductName": "",
		1165	"SpanishProductSubcategoryName": "Pantalones cortos"
		1166	}, {
		1167	"EnglishProductName": "Women's Mountain Shorts, S",
		1168	"SpanishProductName": "",
		1169	"SpanishProductSubcategoryName": "Pantalones cortos"
		1170	}, {
		1171	"EnglishProductName": "Women's Tights, L",
		1172	"SpanishProductName": "Mallas para mujer, G",
		1173	"SpanishProductSubcategoryName": "Mallas"
		1174	}, {
		1175	"EnglishProductName": "Women's Tights, M",
		1176	"SpanishProductName": "Mallas para mujer, M",
		1177	"SpanishProductSubcategoryName": "Mallas"
		1178	}, {
		1179	"EnglishProductName": "Women's Tights, S",
		1180	"SpanishProductName": "Mallas para mujer, P",
		1181	"SpanishProductSubcategoryName": "Mallas"
		1182	}
		1183	}
		1184]

Proposition #13 (Medium)

using adventure works show the amount of times each toCurrency appeared



Columns from Tables

Table Name	Column Name
Currency	Name
CurrencyRate	ToCurrencyCode

```
USE AdventureWorks2017
```

```
SELECT COUNT(E.Name) AS NumberOfSales,  
       O.ToCurrencyCode  
FROM sales.Currency AS E  
CROSS JOIN Sales.CurrencyRate AS O  
GROUP BY O.ToCurrencyCode
```

```
USE AdventureWorks2017  
  
SELECT COUNT(E.Name) AS NumberOfSales,  
       O.ToCurrencyCode  
FROM sales.Currency AS E  
CROSS JOIN Sales.CurrencyRate AS O  
GROUP BY O.ToCurrencyCode  
  
FOR JSON PATH, ROOT(''), INCLUDE_NULL_VALUES;
```

160 %

Results Messages

	NumberOfSales	ToCurrencyCode
1	19320	DEM
2	115185	CAD
3	115185	BRL
4	115185	USD
5	19320	FRF
6	115185	CNY
7	115185	GBP
8	115185	VEB
9	115185	SAR
10	115185	AUD
11	115185	ARS
12	115185	EUR
13	115185	MXN
14	115185	JPY

Refresh

Search

ROOT

[Array]

[0]: [Object]

NumberOfSales: 19320

ToCurrencyCode: "DEM"

[1]: [Object]

NumberOfSales: 115185

ToCurrencyCode: "CAD"

[2]: [Object]

NumberOfSales: 115185

ToCurrencyCode: "BRL"

[3]: [Object]

NumberOfSales: 115185

ToCurrencyCode: "USD"

[4]: [Object]

[5]: [Object]

[6]: [Object]

[7]: [Object]

[8]: [Object]

[9]: [Object]

[10]: [Object]

[11]: [Object]

[12]: [Object]

[13]: [Object]

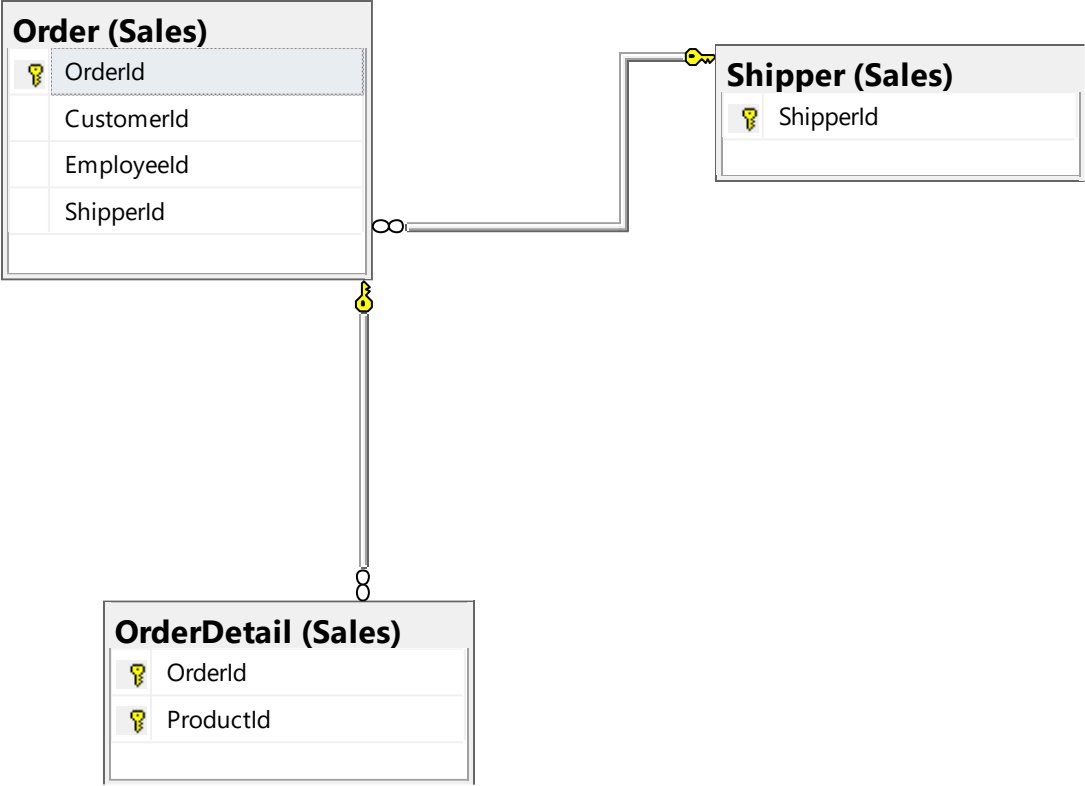
```

14      }, {
15          "NumberOfSales": 19320,
16          "ToCurrencyCode": "FRF"
17      }, {
18          "NumberOfSales": 115185,
19          "ToCurrencyCode": "CNY"
20      }, {
21          "NumberOfSales": 115185,
22          "ToCurrencyCode": "GBP"
23      }, {
24          "NumberOfSales": 115185,
25          "ToCurrencyCode": "VEB"
26      }, {
27          "NumberOfSales": 115185,
28          "ToCurrencyCode": "SAR"
29      }, {
30          "NumberOfSales": 115185,
31          "ToCurrencyCode": "AUD"
32      }, {
33          "NumberOfSales": 115185,
34          "ToCurrencyCode": "ARS"
35      }, {
36          "NumberOfSales": 115185,
37          "ToCurrencyCode": "EUR"
38      }, {
39          "NumberOfSales": 115185,
40          "ToCurrencyCode": "MXN"
41      }, {
42          "NumberOfSales": 115185,
43          "ToCurrencyCode": "JPY"
44      }
45  ]
46

```

Proposition #14 (Complex)

use Northwinds2020TSQLV6 to create a function that returns the total price of each item without the discount, rounded to the nearest hunderth, ordered from cheapest -- total price to most expensive. Return the orderid total price and shipper company name.



Order (Sales)			
Column Name	Data Type	Allow Nulls	
OrderId	Udt.SurrogateKeyIn...	<input type="checkbox"/>	
CustomerId	Udt.SurrogateKeyIn...	<input checked="" type="checkbox"/>	
EmployeeId	Udt.SurrogateKeyIn...	<input type="checkbox"/>	
ShipperId	Udt.SurrogateKeyIn...	<input type="checkbox"/>	
OrderDate	Udt.DateYYYYMM...	<input type="checkbox"/>	
RequiredDate	Udt.DateYYYYMM...	<input type="checkbox"/>	
ShipToDate	Udt.DateYYYYMM...	<input checked="" type="checkbox"/>	
Freight	Udt.Currency:money	<input type="checkbox"/>	
ShipToName	Udt.ContactName:...	<input type="checkbox"/>	
ShipToAddress	Udt.Address:nvarch...	<input type="checkbox"/>	
ShipToCity	Udt.City:nvarchar(15)	<input type="checkbox"/>	
ShipToRegion	Udt.Region:nvarch...	<input checked="" type="checkbox"/>	
ShipToPostalCode	Udt.PostalCode:nv...	<input checked="" type="checkbox"/>	
ShipToCountry	Udt.Country:nvarc...	<input type="checkbox"/>	
UserAuthenticationId	int	<input checked="" type="checkbox"/>	
DateAdded	datetime2(7)	<input checked="" type="checkbox"/>	
DateOfLastUpdate	datetime2(7)	<input checked="" type="checkbox"/>	
		<input type="checkbox"/>	

FK_Order_Shipper

FK_OrderDetail_Order

Shipper (Sales)			
Column Name	Data Type	Allow Nulls	
ShipperId	Udt.SurrogateKeyIn...	<input type="checkbox"/>	
ShipperCompanyName	Udt.CompanyNam...	<input type="checkbox"/>	
PhoneNumber	Udt.TelephoneNum...	<input type="checkbox"/>	
		<input type="checkbox"/>	

OrderDetail (Sales)			
Column Name	Data Type	Allow Nulls	
OrderId	Udt.SurrogateKeyIn...	<input type="checkbox"/>	
ProductId	Udt.SurrogateKeyIn...	<input type="checkbox"/>	
UnitPrice	Udt.Currency:money	<input type="checkbox"/>	
Quantity	Udt.QuantitySmall:...	<input type="checkbox"/>	
DiscountPercentage	Udt.Percentage:nu...	<input type="checkbox"/>	
		<input type="checkbox"/>	

Columns from Tables

Table Name	Column Name
Order	OrderId
OrderDetail	UnitPrice, Quantity
Shipper	ShipperCompanyName

Order By

Table Name	Column Name	Sort Order
OrderDetail	UnitPrice, Quantity	ASC

```
USE Northwinds2020TSQLV6;  
GO
```

```
CREATE  
OR
```

```
ALTER FUNCTION Sales.calculateTotal (  
    @currency MONEY  
    ,@quantity SMALLINT  
)
```

```
RETURNS MONEY
```

```
AS
```

```
BEGIN
```

```
    RETURN @currency * @quantity
```

```
END;
```

```
GO
```

```
SELECT ROUND(Sales.calculateTotal(O.UnitPrice, O.Quantity), 2) AS totalPrice  
    ,E.OrderId  
    ,D.ShipperCompanyName
```

```
FROM Sales.[Order] AS E
```

```
INNER JOIN Sales.OrderDetail AS O ON E.OrderId = O.OrderId
```

```
INNER JOIN Sales.Shipper AS D ON D.ShipperId = E.ShipperId
```

```
GROUP BY E.OrderId
```

```
    ,D.ShipperCompanyName
```

```
    ,O.UnitPrice
```

```
    ,O.Quantity
```

```
ORDER BY totalPrice
```

```

USE Northwinds2020TSQV6;
GO

CREATE
    OR

ALTER FUNCTION Sales.calculateTotal (
    @currency MONEY
    ,@quantity SMALLINT
)
RETURNS MONEY
AS
BEGIN
    RETURN @currency * @quantity
END;
GO

```

160 %

Results Messages

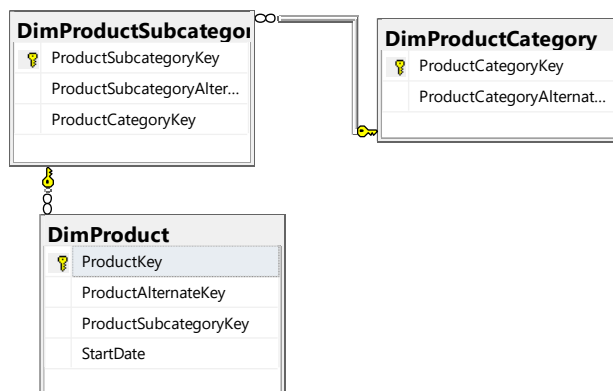
	totalPrice	OrderId	ShipperCompanyName
1	4.80	10462	Shipper GVSUA
2	7.30	10281	Shipper GVSUA
3	9.60	10420	Shipper GVSUA
4	10.00	10850	Shipper GVSUA
5	12.50	10782	Shipper ZHISN
6	13.50	10623	Shipper ETYNR
7	14.00	11038	Shipper ETYNR
8	14.00	11077	Shipper ETYNR
9	15.50	10634	Shipper ZHISN
10	16.00	10341	Shipper ZHISN
11	17.00	11077	Shipper ETYNR
12	17.50	10972	Shipper ETYNR
13	18.00	11077	Shipper ETYNR
14	18.00	10832	Shipper ETYNR
15	18.40	10807	Shipper GVSUA
16	18.40	10831	Shipper ETYNR
17	19.20	10417	Shipper ZHISN
18	20.00	10677	Shipper ZHISN
19	20.00	10528	Shipper ETYNR

✓ Query executed successfully.

Refresh	Search	8588	"OrderId": 10372,
		8589	"ShipperCompanyName": "Shipper ETYNR"
		8590	}, {
		8591	"totalPrice": 9903.2000,
		8592	"OrderId": 10897,
		8593	"ShipperCompanyName": "Shipper ETYNR"
		8594	}, {
		8595	"totalPrice": 10329.2000,
		8596	"OrderId": 10424,
		8597	"ShipperCompanyName": "Shipper ETYNR"
		8598	}, {
		8599	"totalPrice": 10540.0000,
		8600	"OrderId": 10417,
		8601	"ShipperCompanyName": "Shipper ZHISN"
		8602	}, {
		8603	"totalPrice": 10540.0000,
		8604	"OrderId": 10353,
		8605	"ShipperCompanyName": "Shipper ZHISN"
		8606	}, {
		8607	"totalPrice": 10540.0000,
		8608	"OrderId": 10889,
		8609	"ShipperCompanyName": "Shipper ZHISN"
		8610	}, {
		8611	"totalPrice": 15810.0000,
		8612	"OrderId": 10865,
		8613	"ShipperCompanyName": "Shipper GVSUA"
		8614	}, {
		8615	"totalPrice": 15810.0000,
		8616	"OrderId": 10981,
		8617	"ShipperCompanyName": "Shipper ETYNR"
		8618	}
		8619	}
		8620	}

Proposition #15 (Complex)

Create a custom scalar function that combines the products name, category name, and subcategory name and returns them all together and then make a table with two columns that outputs the english and spanish full name



Column Name	Data Type	Allow Null
ProductSubcategoryKey	int	<input type="checkbox"/>
ProductSubcategoryAltern...	int	<input checked="" type="checkbox"/>
EnglishProductSubcategor...	nvarchar(50)	<input type="checkbox"/>
SpanishProductSubcatego...	nvarchar(50)	<input type="checkbox"/>
FrenchProductSubcategory...	nvarchar(50)	<input type="checkbox"/>
ProductCategoryKey	int	<input checked="" type="checkbox"/>

DimProductCategory

Column Name	Data Type	Allow Null
ProductCategoryKey	int	<input type="checkbox"/>
ProductCategoryAlternate...	int	<input checked="" type="checkbox"/>
EnglishProductCategoryN...	nvarchar(50)	<input type="checkbox"/>
SpanishProductCategoryN...	nvarchar(50)	<input type="checkbox"/>
FrenchProductCategoryNa...	nvarchar(50)	<input type="checkbox"/>

FK_DimProduct_DimProductSubcategory

FK_DimProductSubcategory_

DimProduct

Column Name	Data Type	Allow Nulls
ProductKey	int	<input type="checkbox"/>
ProductAlternateKey	nvarchar(25)	<input checked="" type="checkbox"/>
ProductSubcategoryKey	int	<input checked="" type="checkbox"/>
WeightUnitMeasureCode	nchar(3)	<input checked="" type="checkbox"/>
SizeUnitMeasureCode	nchar(3)	<input checked="" type="checkbox"/>
EnglishProductName	nvarchar(50)	<input type="checkbox"/>
SpanishProductName	nvarchar(50)	<input type="checkbox"/>
FrenchProductName	nvarchar(50)	<input type="checkbox"/>
StandardCost	money	<input checked="" type="checkbox"/>
FinishedGoodsFlag	bit	<input type="checkbox"/>
Color	nvarchar(15)	<input type="checkbox"/>
SafetyStockLevel	smallint	<input checked="" type="checkbox"/>
ReorderPoint	smallint	<input checked="" type="checkbox"/>
ListPrice	money	<input checked="" type="checkbox"/>
Size	nvarchar(50)	<input checked="" type="checkbox"/>
SizeRange	nvarchar(50)	<input checked="" type="checkbox"/>
Weight	float	<input checked="" type="checkbox"/>
DaysToManufacture	int	<input checked="" type="checkbox"/>
ProductLine	nchar(2)	<input checked="" type="checkbox"/>
DealerPrice	money	<input checked="" type="checkbox"/>
Class	nchar(2)	<input checked="" type="checkbox"/>
Style	nchar(2)	<input checked="" type="checkbox"/>
ModelName	nvarchar(50)	<input checked="" type="checkbox"/>
LargePhoto	varbinary(MAX)	<input checked="" type="checkbox"/>
EnglishDescription	nvarchar(400)	<input checked="" type="checkbox"/>
FrenchDescription	nvarchar(400)	<input checked="" type="checkbox"/>
ChineseDescription	nvarchar(400)	<input checked="" type="checkbox"/>
ArabicDescription	nvarchar(400)	<input checked="" type="checkbox"/>
HebrewDescription	nvarchar(400)	<input checked="" type="checkbox"/>
ThaiDescription	nvarchar(400)	<input checked="" type="checkbox"/>
GermanDescription	nvarchar(400)	<input checked="" type="checkbox"/>
JapaneseDescription	nvarchar(400)	<input checked="" type="checkbox"/>
TurkishDescription	nvarchar(400)	<input checked="" type="checkbox"/>
StartDate	datetime	<input checked="" type="checkbox"/>
EndDate	datetime	<input checked="" type="checkbox"/>
Status	nvarchar(7)	<input checked="" type="checkbox"/>

Columns from Tables

Table Name	Column Name
DimProduct	EnglishProductName, SpanishProductName
DimProductSubcategory	SpanishProductSubcategoryName, EnglishProductSubcategoryName
DimProductCategory	EnglishProductCategoryName, SpanishProductCategoryName

Order By

Table Name	Column Name	Sort Order
DimProduct	EnglishProductName	ASC

```
USE AdventureWorksDW2017;  
GO
```

```
CREATE  
OR
```

```
ALTER FUNCTION dbo.compProductName (  
    @ProductName NVARCHAR(50) ,  
    @productcategoryname NVARCHAR(50) ,  
    @productsubcategoryname NVARCHAR(50)  
)  
RETURNS NVARCHAR(50)  
AS  
BEGIN  
    RETURN @ProductName + ' ' + @productcategoryname + ' ' +  
@productsubcategoryname;  
END;  
GO
```

```
USE AdventureWorksDW2017
```

```
SELECT dbo.compProductName(E.EnglishProductName ,  
O.EnglishProductSubcategoryName, D.EnglishProductCategoryName) AS  
FullEnglishName ,  
    dbo.compProductName(E.SpanishProductName ,  
O.SpanishProductSubcategoryName, D.SpanishProductCategoryName) AS  
FullSpanishName  
FROM dbo.DimProduct AS E  
INNER JOIN dbo.DimProductSubcategory AS O ON E.ProductSubcategoryKey =  
O.ProductSubcategoryKey  
INNER JOIN dbo.DimProductCategory AS D ON D.ProductCategoryKey =  
O.ProductCategoryKey  
WHERE SpanishProductName IS NOT NULL  
GROUP BY E.EnglishProductName ,  
    E.SpanishProductName ,  
    O.SpanishProductSubcategoryName ,  
    D.SpanishProductCategoryName ,  
    O.EnglishProductSubcategoryName ,
```

D.EnglishProductCategoryName
ORDER BY E.EnglishProductName

```
USE AdventureWorksDW2017;  
GO
```

CREATE

OR

```
ALTER FUNCTION dbo.compProductName (  
    @ProductName NVARCHAR(50)  
    ,@productcategoryname NVARCHAR(50)  
    ,@productsubcategoryname NVARCHAR(50)  
)
```

```
RETURNS NVARCHAR(50)
```

```
AS
```

```
BEGIN
```

```
    RETURN @ProductName + ' ' + @productcategoryname + ' ' + @productsubcategoryname;
```

```
END
```

160 %

Results Messages

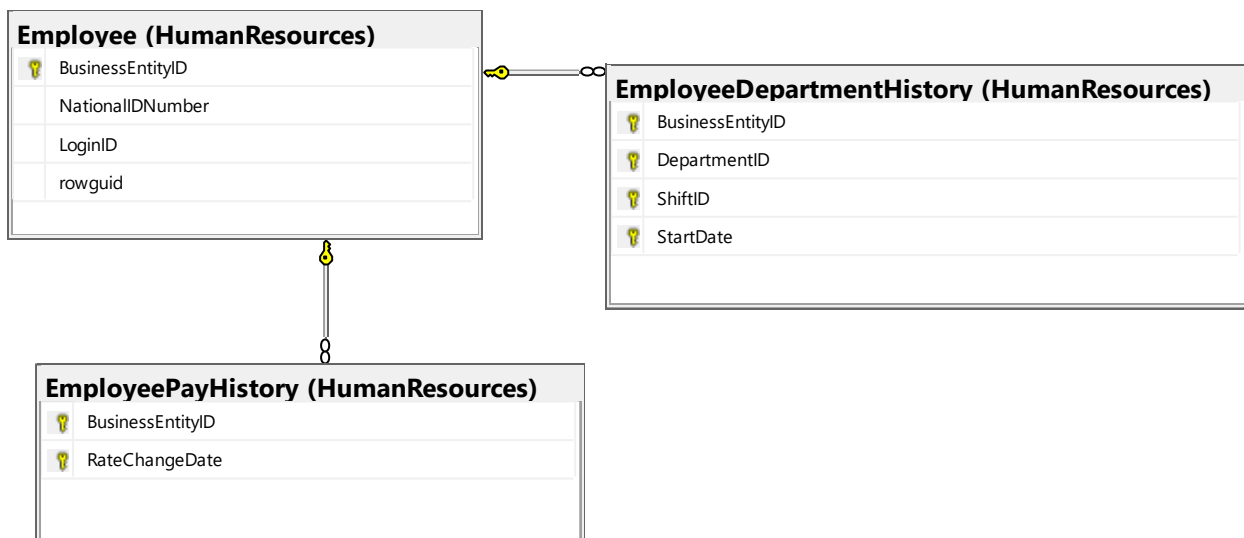
	FullEnglishName	FullSpanishName
1	All-Purpose Bike Stand Bike Stands Accessories	Soporte multiusos para bicicletas Soporte para bic
2	AWC Logo Cap Caps Clothing	Gorra Prenda
3	Bike Wash - Dissolver Cleaners Accessories	Lavado de bicicletas: disolvente Limpiador Accesor
4	Cable Lock Locks Accessories	Cable antirrobo Candado Accesorio
5	Chain Chains Components	Cadena Cadena Componente
6	Classic Vest, L Vests Clothing	Camiseta clásica, G Camiseta Prenda
7	Classic Vest, M Vests Clothing	Camiseta clásica, M Camiseta Prenda
8	Classic Vest, S Vests Clothing	Camiseta clásica, P Camiseta Prenda
9	Fender Set - Mountain Fenders Accessories	Conjunto de guardabarros: montaña Guardabarros Acc
10	Front Brakes Brakes Components	Frenos delanteros Frenos Componente
11	Front Derailleur Derailleurs Components	Desviador delantero Desviador Componente
12	Full-Finger Gloves, L Gloves Clothing	Guantes completos, G Guantes Prenda
13	Full-Finger Gloves, M Gloves Clothing	Guantes completos, M Guantes Prenda
14	Full-Finger Gloves, S Gloves Clothing	Guantes completos, P Guantes Prenda
15	Half-Finger Gloves, L Gloves Clothing	Guantes Prenda
16	Half-Finger Gloves, M Gloves Clothing	Guantes Prenda
17	Half-Finger Gloves, S Gloves Clothing	Guantes Prenda
18	Headlights - Dual-Beam Lights Accessories	Luces: doble haz Luz Accesorio
19	Headlights - Weatherproof Lights Accessories	Luces: resistentes al agua Luz Accesorio

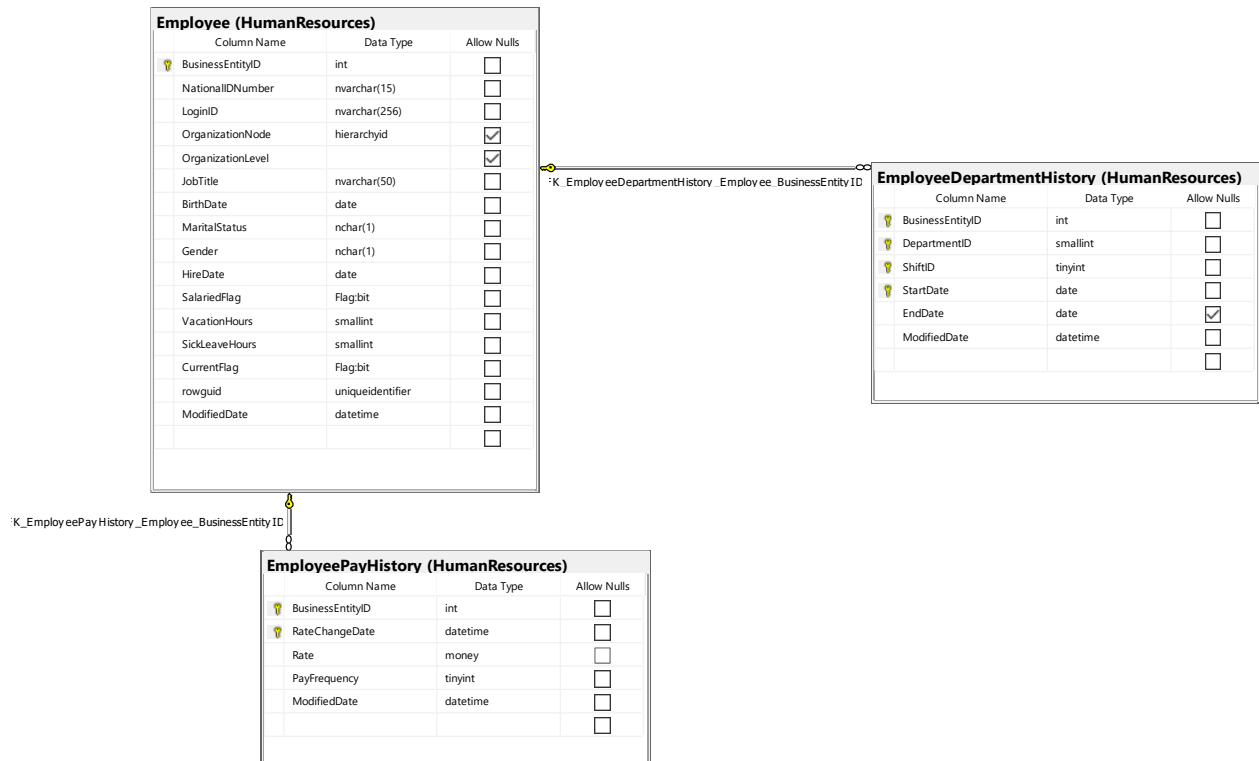
Query executed successfully

Refresh	Search	857	}, {
⊖ [Array]		858	"FullEnglishName": "Touring-3000 Yellow, 58 Touring Bikes Bikes",
⊕ [0]: [Object]	FullEnglishName: "All-Purp"	859	"FullSpanishName": "Paseo: 3000, amarilla, 58 Bicicleta de paseo Bicie"
⊕ [1]: [Object]	FullEnglishName: "AWC Lu"	860	}, {
⊕ [2]: [Object]	FullEnglishName: "Bike W"	861	"FullEnglishName": "Touring-3000 Yellow, 62 Touring Bikes Bikes",
⊕ [3]: [Object]	FullEnglishName: "Lavrad"	862	"FullSpanishName": "Paseo: 3000, amarilla, 62 Bicicleta de paseo Bicie"
⊕ [4]: [Object]	FullEnglishName: "Cable L"	863	}, {
⊕ [5]: [Object]	FullEnglishName: "Chain C"	864	"FullEnglishName": "Touring-Panniers, Large Panniers Accessories",
⊕ [6]: [Object]	FullSpanishName: "Caden"	865	"FullSpanishName": "Cesta de paseo, grande Cesta Accesorio"
⊕ [7]: [Object]		866	}, {
⊕ [8]: [Object]		867	"FullEnglishName": "Water Bottle - 30 oz. Bottles and Cages Accessorie",
⊕ [9]: [Object]		868	"FullSpanishName": " Portabotellas y botella Accesorio"
⊕ [10]: [Object]		869	}, {
⊕ [11]: [Object]		870	"FullEnglishName": "Women's Mountain Shorts, L Shorts Clothing",
⊕ [12]: [Object]		871	"FullSpanishName": " Pantalones cortos Prenda"
⊕ [13]: [Object]		872	}, {
⊕ [14]: [Object]		873	"FullEnglishName": "Women's Mountain Shorts, M Shorts Clothing",
⊕ [15]: [Object]		874	"FullSpanishName": " Pantalones cortos Prenda"
⊕ [16]: [Object]		875	}, {
⊕ [17]: [Object]		876	"FullEnglishName": "Women's Mountain Shorts, S Shorts Clothing",
⊕ [18]: [Object]		877	"FullSpanishName": " Pantalones cortos Prenda"
⊕ [19]: [Object]		878	}, {
⊕ [20]: [Object]		879	"FullEnglishName": "Women's Tights, L Tights Clothing",
⊕ [21]: [Object]		880	"FullSpanishName": "Mallas para mujer, G Mallas Prenda"
⊕ [22]: [Object]		881	}, {
⊕ [23]: [Object]		882	"FullEnglishName": "Women's Tights, M Tights Clothing",
⊕ [24]: [Object]		883	"FullSpanishName": "Mallas para mujer, M Mallas Prenda"
⊕ [25]: [Object]		884	}, {
⊕ [26]: [Object]		885	"FullEnglishName": "Women's Tights, S Tights Clothing",
⊕ [27]: [Object]		886	"FullSpanishName": "Mallas para mujer, P Mallas Prenda"
⊕ [28]: [Object]		887	}, {
⊕ [29]: [Object]		888	"FullEnglishName": "Women's Tights, L Tights Clothing",
⊕ [30]: [Object]		889	"FullSpanishName": "Mallas para mujer, G Mallas Prenda"
⊕ [31]: [Object]			}, {
⊕ [32]: [Object]			"FullEnglishName": "Women's Tights, M Tights Clothing",
⊕ [33]: [Object]			"FullSpanishName": "Mallas para mujer, M Mallas Prenda"
⊕ [34]: [Object]			}, {
⊕ [35]: [Object]			"FullEnglishName": "Women's Tights, S Tights Clothing",
⊕ [36]: [Object]			"FullSpanishName": "Mallas para mujer, P Mallas Prenda"
⊕ [37]: [Object]			}, {
⊕ [38]: [Object]			"FullEnglishName": "Women's Tights, L Tights Clothing",
⊕ [39]: [Object]			"FullSpanishName": "Mallas para mujer, G Mallas Prenda"
⊕ [40]: [Object]			}, {
⊕ [41]: [Object]			"FullEnglishName": "Women's Tights, M Tights Clothing",
⊕ [42]: [Object]			"FullSpanishName": "Mallas para mujer, M Mallas Prenda"
⊕ [43]: [Object]			}, {
⊕ [44]: [Object]			"FullEnglishName": "Women's Tights, S Tights Clothing",
⊕ [45]: [Object]			"FullSpanishName": "Mallas para mujer, P Mallas Prenda"
⊕ [46]: [Object]			}, {
⊕ [47]: [Object]			"FullEnglishName": "Women's Tights, L Tights Clothing",
⊕ [48]: [Object]			"FullSpanishName": "Mallas para mujer, G Mallas Prenda"
⊕ [49]: [Object]			}, {
⊕ [50]: [Object]			"FullEnglishName": "Women's Tights, M Tights Clothing",
⊕ [51]: [Object]			"FullSpanishName": "Mallas para mujer, M Mallas Prenda"
⊕ [52]: [Object]			}, {
⊕ [53]: [Object]			"FullEnglishName": "Women's Tights, S Tights Clothing",
⊕ [54]: [Object]			"FullSpanishName": "Mallas para mujer, P Mallas Prenda"
⊕ [55]: [Object]			}, {
⊕ [56]: [Object]			"FullEnglishName": "Women's Tights, L Tights Clothing",
⊕ [57]: [Object]			"FullSpanishName": "Mallas para mujer, G Mallas Prenda"
⊕ [58]: [Object]			}, {
⊕ [59]: [Object]			"FullEnglishName": "Women's Tights, M Tights Clothing",
⊕ [60]: [Object]			"FullSpanishName": "Mallas para mujer, M Mallas Prenda"
⊕ [61]: [Object]			}, {
⊕ [62]: [Object]			"FullEnglishName": "Women's Tights, S Tights Clothing",
⊕ [63]: [Object]			"FullSpanishName": "Mallas para mujer, P Mallas Prenda"
⊕ [64]: [Object]			}, {
⊕ [65]: [Object]			"FullEnglishName": "Women's Tights, L Tights Clothing",
⊕ [66]: [Object]			"FullSpanishName": "Mallas para mujer, G Mallas Prenda"
⊕ [67]: [Object]			}, {
⊕ [68]: [Object]			"FullEnglishName": "Women's Tights, M Tights Clothing",
⊕ [69]: [Object]			"FullSpanishName": "Mallas para mujer, M Mallas Prenda"
⊕ [70]: [Object]			}, {
⊕ [71]: [Object]			"FullEnglishName": "Women's Tights, S Tights Clothing",
⊕ [72]: [Object]			"FullSpanishName": "Mallas para mujer, P Mallas Prenda"
⊕ [73]: [Object]			}, {
⊕ [74]: [Object]			"FullEnglishName": "Women's Tights, L Tights Clothing",
⊕ [75]: [Object]			"FullSpanishName": "Mallas para mujer, G Mallas Prenda"
⊕ [76]: [Object]			}, {
⊕ [77]: [Object]			"FullEnglishName": "Women's Tights, M Tights Clothing",
⊕ [78]: [Object]			"FullSpanishName": "Mallas para mujer, M Mallas Prenda"
⊕ [79]: [Object]			}, {
⊕ [80]: [Object]			"FullEnglishName": "Women's Tights, S Tights Clothing",
⊕ [81]: [Object]			"FullSpanishName": "Mallas para mujer, P Mallas Prenda"
⊕ [82]: [Object]			}, {
⊕ [83]: [Object]			"FullEnglishName": "Women's Tights, L Tights Clothing",
⊕ [84]: [Object]			"FullSpanishName": "Mallas para mujer, G Mallas Prenda"
⊕ [85]: [Object]			}, {
⊕ [86]: [Object]			"FullEnglishName": "Women's Tights, M Tights Clothing",
⊕ [87]: [Object]			"FullSpanishName": "Mallas para mujer, M Mallas Prenda"
⊕ [88]: [Object]			}, {
⊕ [89]: [Object]			"FullEnglishName": "Women's Tights, S Tights Clothing",
⊕ [90]: [Object]			"FullSpanishName": "Mallas para mujer, P Mallas Prenda"
⊕ [91]: [Object]			}, {
⊕ [92]: [Object]			"FullEnglishName": "Women's Tights, L Tights Clothing",
⊕ [93]: [Object]			"FullSpanishName": "Mallas para mujer, G Mallas Prenda"
⊕ [94]: [Object]			}, {
⊕ [95]: [Object]			"FullEnglishName": "Women's Tights, M Tights Clothing",
⊕ [96]: [Object]			"FullSpanishName": "Mallas para mujer, M Mallas Prenda"
⊕ [97]: [Object]			}, {
⊕ [98]: [Object]			"FullEnglishName": "Women's Tights, S Tights Clothing",
⊕ [99]: [Object]			"FullSpanishName": "Mallas para mujer, P Mallas Prenda"

Proposition #16 (Complex)

make a scalar function that calculates the amount of money paid out for sick leave and vacation days. --
then use that information to construct a table with job column and amount paid for non work days.





Columns from Tables

Table Name	Column Name
Employee	JobTitle, VacationHours, SickLeaveHours
EmployeeDepartmentHistory	BusinessEntityID
EmployeePayHistory	Rate

Order By

Table Name	Column Name	Sort Order
HumanResources	PaidNonWorkHours	ASC

```
USE AdventureWorks2017
GO
```

```
CREATE
```

```
OR
```

```
ALTER FUNCTION HumanResources.PaidNonWorkHours (
    @vacationhours SMALLINT
    ,@sickleavehours SMALLINT
    ,@rate MONEY
)
```

```
RETURNS MONEY
```

```
AS
```

```

BEGIN
    RETURN (@vacationhours * @rate) + (@sickleavehours * @rate)
END;
GO

SELECT E.JobTitle
       ,HumanResources.PaidNonWorkHours(E.VacationHours, E.SickLeaveHours,
D.Rate) paidNonWorkCost
FROM HumanResources.Employee AS E
INNER JOIN HumanResources.EmployeeDepartmentHistory AS O ON
E.BusinessEntityID = O.BusinessEntityID
INNER JOIN HumanResources.EmployeePayHistory AS D ON E.BusinessEntityID =
D.BusinessEntityID
GROUP BY E.JobTitle
       ,E.VacationHours
       ,E.SickLeaveHours
       ,D.Rate
ORDER BY paidNonWorkCost;

```

USE AdventureWorks2017
GO

CREATE
OR

ALTER FUNCTION HumanResources.PaidNonWorkHours (
 @vacationhours SMALLINT
 ,@sickleavehours SMALLINT
 ,@rate MONEY
)
 RETURNS MONEY
 AS
 BEGIN
 RETURN (@vacationhours * @rate) + (@sickleavehours * @rate)
 END;
 GO

SELECT E.JobTitle

132 %

Results Messages


	JobTitle	paidNonWorkCost
1	Production Technician - WC50	220.00
2	Production Technician - WC50	231.00
3	Production Technician - WC50	253.00
4	Production Technician - WC50	264.00
5	Production Technician - WC20	280.00
6	Production Technician - WC50	286.00
7	Production Technician - WC20	294.00
8	Production Technician - WC50	297.00
9	Production Technician - WC50	319.00
10	Production Technician - WC20	322.00
11	Production Technician - WC50	330.00
12	Production Technician - WC20	336.00
13	Production Technician - WC30	344.50
14	Production Technician - WC30	351.00
15	Production Technician - WC50	352.00
16	Production Technician - WC50	363.00
17	Production Technician - WC30	364.00
18	Production Technician - WC20	364.00
19	Production Technician - WC30	370.50


Refresh	Search	
[-] : [Array]		920
[-] [0]: [Object]		921
[-] JobTitle: "Production Ted"		922
[-] paidNonWorkCost: 220.0		923
[-] [1]: [Object]		924
[-] JobTitle: "Production Ted"		925
[-] paidNonWorkCost: 231.0		926
[-] [2]: [Object]		927
[-] JobTitle: "Production Ted"		928
[-] paidNonWorkCost: 253.0		929
[-] [3]: [Object]		930
[-] JobTitle: "Production Ted"		931
[-] paidNonWorkCost: 264.0		932
[-] [4]: [Object]		933
[-] JobTitle: "Production Ted"		934
[-] paidNonWorkCost: 280.0		935
[-] [5]: [Object]		936
[-] [6]: [Object]		937
[-] [7]: [Object]		938
[-] [8]: [Object]		939
[-] [9]: [Object]		940
[-] [10]: [Object]		941
[-] [11]: [Object]		942
[-] [12]: [Object]		943
[-] [13]: [Object]		944
[-] [14]: [Object]		945
[-] [15]: [Object]		946
[-] [16]: [Object]		947
[-] [17]: [Object]		948
[-] [18]: [Object]		949
[-] [19]: [Object]		950
[-] [20]: [Object]		951
[-] [21]: [Object]		952
[-] [22]: [Object]		
[-] [23]: [Object]		
[-] [24]: [Object]		
[-] [25]: [Object]		
[-] [26]: [Object]		
[-] [27]: [Object]		
[-] [28]: [Object]		
[-] [29]: [Object]		
[-] [30]: [Object]		
[-] [31]: [Object]		
[-] [32]: [Object]		
[-] [33]: [Object]		
[-] [34]: [Object]		
[-] [35]: [Object]		
[-] [36]: [Object]		
[-] [37]: [Object]		
[-] [38]: [Object]		
[-] [39]: [Object]		
[-] [40]: [Object]		
[-] [41]: [Object]		
[-] [42]: [Object]		


Proposition #17 (Complex)



Make a scalar function that will add n months to a enter date -- then use this function with the dbo.nums to make a table with sales.order and sales.orderdetail that creates -- new ship dates for each product where each shipment is delayed n number of months where N <= 24


Nums	
	N

OrderDetail (Sales)	
	OrderId
	ProductId

Order (Sales)	
	OrderId
	CustomerId
	EmployeeId
	ShipperId

Nums			
	Column Name	Data Type	Allow Nulls
	N	Udt.SurrogateKeyIn...	<input type="checkbox"/>
			<input type="checkbox"/>

OrderDetail (Sales)			
	Column Name	Data Type	Allow Nulls
	OrderId	Udt.SurrogateKeyIn...	<input type="checkbox"/>
	ProductId	Udt.SurrogateKeyIn...	<input type="checkbox"/>
	UnitPrice	Udt.Currency:money	<input type="checkbox"/>
	Quantity	Udt.QuantitySmall:...	<input type="checkbox"/>
	DiscountPercentage	Udt.Percentage:nu...	<input type="checkbox"/>
			<input type="checkbox"/>

Order (Sales)			
	Column Name	Data Type	Allow Nulls
	OrderId	Udt.SurrogateKeyIn...	<input type="checkbox"/>
	CustomerId	Udt.SurrogateKeyIn...	<input checked="" type="checkbox"/>
	EmployeeId	Udt.SurrogateKeyIn...	<input type="checkbox"/>
	ShipperId	Udt.SurrogateKeyIn...	<input type="checkbox"/>
	OrderDate	Udt.DateYYYYMM...	<input type="checkbox"/>
	RequiredDate	Udt.DateYYYYMM...	<input type="checkbox"/>
	ShipToDate	Udt.DateYYYYMM...	<input checked="" type="checkbox"/>
	Freight	Udt.Currency:money	<input type="checkbox"/>
	ShipToName	Udt.ContactName:...	<input type="checkbox"/>
	ShipToAddress	Udt.Address:nvarch...	<input type="checkbox"/>
	ShipToCity	Udt.City:nvarchar(15)	<input type="checkbox"/>
	ShipToRegion	Udt.Region:nvarch...	<input checked="" type="checkbox"/>
	ShipToPostalCode	Udt.PostalCode:nv...	<input checked="" type="checkbox"/>
	ShipToCountry	Udt.Country:nvarc...	<input type="checkbox"/>
	UserAuthenticationId	int	<input checked="" type="checkbox"/>
	DateAdded	datetime2(7)	<input checked="" type="checkbox"/>
	DateOfLastUpdate	datetime2(7)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

FK_OrderDetail_Order

Columns from Tables

Table Name	Column Name
Order	ProductId
OrderDetail	SpanishProductSubcategoryName
Nums	n

Order By

Table Name	Column Name	Sort Order
Nums	n	ASC

```
USE Northwinds2020TSQLV6
GO
```

```
CREATE
    OR
```

```
ALTER FUNCTION dbo.createDate (
    @numberOfMonths INT
    ,@date DATE
)
RETURNS DATE
AS
BEGIN
    DECLARE @month INT;
    DECLARE @year INT;

    SELECT @year = @numberOfMonths / 12;

    SELECT @month = (@numberOfMonths % 12);

    SELECT @date = DATEADD(year, @year, @date);

    SELECT @date = DATEADD(month, @month, @date)

    RETURN @date;
END;
GO
```

```
SELECT D.n
    ,O.ProductId
    ,E.ShipToDate
    ,dbo.createDate(D.n, E.ShipToDate) AS NewShipToDate
FROM dbo.Nums AS D
CROSS JOIN Sales.[Order] AS E
INNER JOIN Sales.OrderDetail AS O ON E.OrderId = O.OrderId
WHERE n <= 24
    AND ShipToDate IS NOT NULL
GROUP BY D.n
    ,O.ProductId
```

```

,E.ShipToDate
,D.n
,E.ShipToDate
ORDER BY D.n

```

```

USE Northwinds2020TSQLV6
GO

CREATE
    OR

ALTER FUNCTION dbo.createDate (
    @numberOfMonths INT
    ,@date DATE
)
RETURNS DATE
AS
BEGIN
    DECLARE @month INT;
    DECLARE @year INT;

    SELECT @year = @numberOfMonths / 12;

    SELECT @month = (@numberOfMonths % 12);

```

132 %

Results Messages

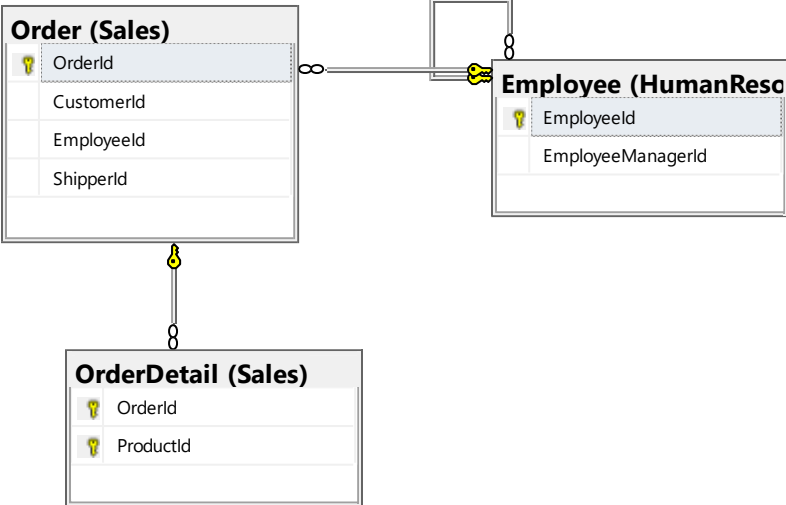
	n	ProductId	ShipToDate	NewShipToDate
1	1	28	2015-02-14	2015-03-14
2	1	55	2015-03-12	2015-04-12
3	1	29	2016-01-19	2016-02-19
4	1	56	2016-01-13	2016-02-13
5	1	11	2015-01-30	2015-02-28
6	1	7	2016-03-06	2016-04-06
7	1	31	2015-09-23	2015-10-23
8	1	77	2015-04-07	2015-05-07
9	1	48	2015-04-22	2015-05-22
10	1	40	2015-03-14	2015-04-14
11	1	60	2015-11-20	2015-12-20
12	1	17	2016-02-18	2016-03-18
13	1	31	2015-04-07	2015-05-07
14	1	44	2016-05-01	2016-06-01
15	1	40	2014-08-12	2014-09-12
16	1	11	2015-04-02	2015-05-02
17	1	5	2015-11-05	2015-12-05
18	1	73	2015-08-21	2015-09-21
19	1	21	2015-09-09	2015-10-09

The screenshot shows a SQL Server Enterprise Manager window with a query result. The left pane displays a tree view of the result set, showing an array of objects with properties like n, ProductId, ShipToDate, and NewShipToDate. The right pane shows the corresponding JSON output for the same data.

OrderID	n	ProductId	ShipToDate	NewShipToDate
243812	24	62	"2016-01-05"	"2018-01-05"
243813	24	16	"2016-03-10"	"2018-03-10"
243814	24	26	"2015-12-04"	"2017-12-04"
243815	24	2	"2014-07-15"	"2016-07-15"
243816	24	42	"2016-04-16"	"2018-04-16"
243817	24	24	"2016-03-31"	"2018-03-31"

Proposition #18 (Complex)

Create a scalar function which returns the gap in days between two dates then use that function to create a table that shows the difference in days between the order date and required date also include the orderId EmployeeID and OrderDate from sales.order sales.orderDetail and HumanResources.Employee



FK_OrderDetail_Order	Order (Sales)			Employee (HumanResources)		
	Column Name	Data Type	Allow Nulls	Column Name	Data Type	Allow Nulls
FK_OrderDetail_Order	OrderId	Udt.SurrogateKeyInt...	<input type="checkbox"/>	EmployeeId	Udt.SurrogateKeyInt...	<input type="checkbox"/>
	CustomerId	Udt.SurrogateKeyInt...	<input checked="" type="checkbox"/>	EmployeeLastName	Udt.LastName:nvarc...	<input type="checkbox"/>
	EmployeeId	Udt.SurrogateKeyInt...	<input type="checkbox"/>	EmployeeFirstName	Udt.FirstName:nvarc...	<input type="checkbox"/>
	ShipperId	Udt.SurrogateKeyInt...	<input type="checkbox"/>	EmployeeTitle	Udt.Title:nvarchar(30)	<input type="checkbox"/>
	OrderDate	Udt.DateYYYYMMD...	<input type="checkbox"/>	EmployeeTitleOfCourtesy	Udt.TitleOfCourtesy...	<input type="checkbox"/>
	RequiredDate	Udt.DateYYYYMMD...	<input type="checkbox"/>	BirthDate	Udt.DateYYYYMMD...	<input type="checkbox"/>
	ShipToDate	Udt.DateYYYYMMD...	<input checked="" type="checkbox"/>	HireDate	Udt.DateYYYYMMD...	<input type="checkbox"/>
	Freight	Udt.Currency:money	<input type="checkbox"/>	EmployeeAddress	Udt.Address:nvarcha...	<input type="checkbox"/>
	ShipToName	Udt.ContactName:n...	<input type="checkbox"/>	EmployeeCity	Udt.City:nvarchar(15)	<input checked="" type="checkbox"/>
	ShipToAddress	Udt.Address:nvarcha...	<input type="checkbox"/>	EmployeeRegion	Udt.Region:nvarchar(...)	<input checked="" type="checkbox"/>
	ShipToCity	Udt.City:nvarchar(15)	<input type="checkbox"/>	EmployeePostalCode	Udt.PostalCode:nvar...	<input checked="" type="checkbox"/>
	ShipToRegion	Udt.Region:nvarchar(...)	<input checked="" type="checkbox"/>	EmployeeCountry	Udt.Country:nvarcha...	<input type="checkbox"/>
	ShipToPostalCode	Udt.PostalCode:nvar...	<input checked="" type="checkbox"/>	EmployeePhoneNumber	Udt.TelephoneNumb...	<input type="checkbox"/>
	ShipToCountry	Udt.Country:nvarcha...	<input type="checkbox"/>	EmployeeManagerId	Udt.SurrogateKeyInt...	<input checked="" type="checkbox"/>
	UserAuthenticationId	int	<input checked="" type="checkbox"/>			<input type="checkbox"/>
	DateAdded	datetime2(7)	<input checked="" type="checkbox"/>			
	DateOfLastUpdate	datetime2(7)	<input checked="" type="checkbox"/>			
			<input type="checkbox"/>			
OrderDetail (Sales)						
	Column Name	Data Type	Allow Nulls			
	OrderId	Udt.SurrogateKeyInt...	<input type="checkbox"/>			
	ProductId	Udt.SurrogateKeyInt...	<input type="checkbox"/>			
	UnitPrice	Udt.Currency:money	<input type="checkbox"/>			
	Quantity	Udt.QuantitySmalls...	<input type="checkbox"/>			
	DiscountPercentage	Udt.Percentage:num...	<input type="checkbox"/>			
			<input type="checkbox"/>			

Columns from Tables

Table Name	Column Name
Order	OrderDate, RequiredDate
OrderDetail	OrderId
Employee	EmployeeId

```
USE Northwinds2020TSQLV6
GO
```

```
CREATE
    OR
```

```
ALTER FUNCTION Sales.dateDiff (
    @StartDate DATE
    ,@EndDate DATE
)
RETURNS INT
AS
BEGIN
    RETURN DATEDIFF(day, @StartDate, @EndDate)
END
GO
```

```
SELECT O.OrderId
    ,D.EmployeeId
    ,E.OrderDate
    ,E.RequiredDate
    ,Sales.dateDiff(E.OrderDate, E.RequiredDate) AS DeliveryTime
FROM Sales.[Order] AS E
INNER JOIN Sales.OrderDetail AS O ON E.OrderId = O.OrderId
INNER JOIN HumanResources.Employee AS D ON D.EmployeeId = E.EmployeeId
GROUP BY O.OrderId
    ,D.EmployeeId
    ,E.OrderDate
    ,E.RequiredDate
    ,Sales.dateDiff(E.OrderDate, E.RequiredDate)
```

```
USE Northwinds2020TSQLV6
```

```
GO
```

```
CREATE
```

```
OR
```

```
ALTER FUNCTION Sales.dateDiff (
```

```
    @StartDate DATE
```

```
    ,@EndDate DATE
```

```
)
```

```
RETURNS INT
```

```
AS
```

```
BEGIN
```

```
    RETURN DATEDIFF(day, @StartDate, @EndDate)
```

```
END
```

```
GO
```

```
SELECT O.OrderId
```

```
    ,D.EmployeeId
```

132 %

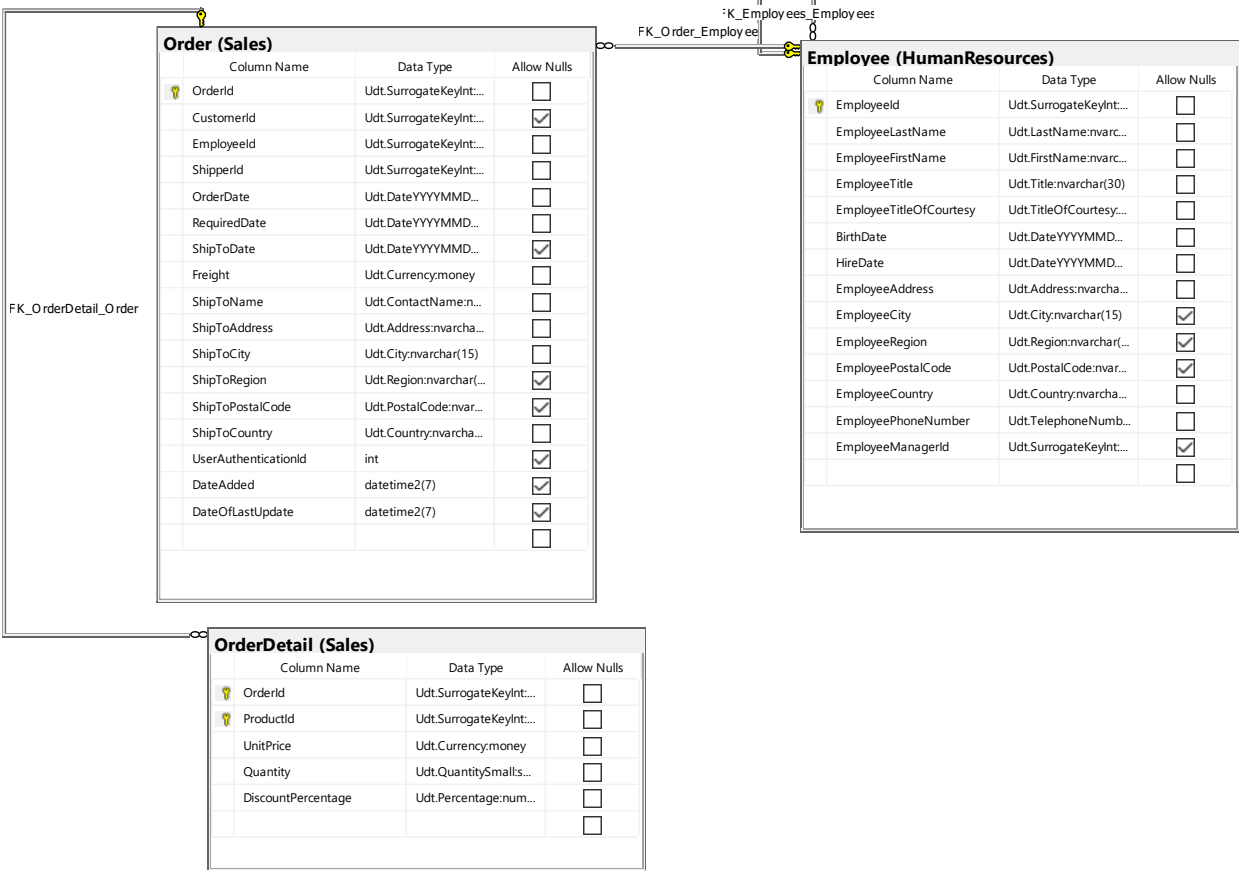
Results Messages

	OrderId	EmployeeId	OrderDate	RequiredDate	DeliveryTime
812	11059	2	2016-04-29	2016-06-10	42
813	11060	2	2016-04-30	2016-05-28	28
814	11061	4	2016-04-30	2016-06-11	42
815	11062	4	2016-04-30	2016-05-28	28
816	11063	3	2016-04-30	2016-05-28	28
817	11064	1	2016-05-01	2016-05-29	28
818	11065	8	2016-05-01	2016-05-29	28
819	11066	7	2016-05-01	2016-05-29	28
820	11067	1	2016-05-04	2016-05-18	14
821	11068	8	2016-05-04	2016-06-01	28
822	11069	1	2016-05-04	2016-06-01	28
823	11070	2	2016-05-05	2016-06-02	28
824	11071	1	2016-05-05	2016-06-02	28
825	11072	4	2016-05-05	2016-06-02	28
826	11073	2	2016-05-05	2016-06-02	28
827	11074	7	2016-05-06	2016-06-03	28
828	11075	8	2016-05-06	2016-06-03	28
829	11076	4	2016-05-06	2016-06-03	28
830	11077	1	2016-05-06	2016-06-03	28

Refresh	Search	
<div> <div>[Array]</div> <div> <div>[0]: [Object]</div> <div> <div>Orderid: 10248</div> <div>Employeeid: 5</div> <div>OrderDate: "2014-07-04"</div> <div>RequiredDate: "2014-08-01"</div> <div>DeliveryTime: 28</div> </div> </div> <div>[1]: [Object]</div> <div> <div>Orderid: 10249</div> <div>Employeeid: 6</div> <div>OrderDate: "2014-07-05"</div> <div>RequiredDate: "2014-08-02"</div> <div>DeliveryTime: 42</div> </div> <div>[2]: [Object]</div> <div> <div>Orderid: 10250</div> <div>Employeeid: 4</div> <div>OrderDate: "2014-07-08"</div> <div>RequiredDate: "2014-08-05"</div> <div>DeliveryTime: 28</div> </div> <div>[3]: [Object]</div> <div> <div>Orderid: 10251</div> <div>Employeeid: 3</div> <div>OrderDate: "2014-07-08"</div> <div>RequiredDate: "2014-08-05"</div> <div>DeliveryTime: 28</div> </div> <div>[4]: [Object]</div> <div> <div>Orderid: 10252</div> <div>Employeeid: 4</div> <div>OrderDate: "2014-07-09"</div> <div>RequiredDate: "2014-08-06"</div> <div>DeliveryTime: 28</div> </div> <div>[5]: [Object]</div> <div>[6]: [Object]</div> <div>[7]: [Object]</div> <div>[8]: [Object]</div> <div>[9]: [Object]</div> <div>[10]: [Object]</div> <div>[11]: [Object]</div> <div>[12]: [Object]</div> <div>[13]: [Object]</div> <div>[14]: [Object]</div> <div>[15]: [Object]</div> <div>[16]: [Object]</div> <div>[17]: [Object]</div> <div>[18]: [Object]</div> <div>[19]: [Object]</div> <div>[20]: [Object]</div> <div>[21]: [Object]</div> <div>[22]: [Object]</div> <div>[23]: [Object]</div> <div>[24]: [Object]</div> <div>[25]: [Object]</div> <div>[26]: [Object]</div> <div>[27]: [Object]</div> </div>	<div> <div>4952</div> <div>4953</div> <div>4954</div> <div>4955</div> <div>4956</div> <div>4957</div> <div>4958</div> <div>4959</div> <div>4960</div> <div>4961</div> <div>4962</div> <div>4963</div> <div>4964</div> <div>4965</div> <div>4966</div> <div>4967</div> <div>4968</div> <div>4969</div> <div>4970</div> <div>4971</div> <div>4972</div> <div>4973</div> <div>4974</div> <div>4975</div> <div>4976</div> <div>4977</div> <div>4978</div> <div>4979</div> <div>4980</div> <div>4981</div> <div>4982</div> <div>4983</div> <div>4984</div> </div>	<div> <div>}, {</div> <div> <div>"OrderId": 11073,</div> <div>"EmployeeId": 2,</div> <div>"OrderDate": "2016-05-05",</div> <div>"RequiredDate": "2016-06-02",</div> <div>"DeliveryTime": 28</div> </div> <div>}, {</div> <div> <div>"OrderId": 11074,</div> <div>"EmployeeId": 7,</div> <div>"OrderDate": "2016-05-06",</div> <div>"RequiredDate": "2016-06-03",</div> <div>"DeliveryTime": 28</div> </div> <div>}, {</div> <div> <div>"OrderId": 11075,</div> <div>"EmployeeId": 8,</div> <div>"OrderDate": "2016-05-06",</div> <div>"RequiredDate": "2016-06-03",</div> <div>"DeliveryTime": 28</div> </div> <div>}, {</div> <div> <div>"OrderId": 11076,</div> <div>"EmployeeId": 4,</div> <div>"OrderDate": "2016-05-06",</div> <div>"RequiredDate": "2016-06-03",</div> <div>"DeliveryTime": 28</div> </div> <div>}, {</div> <div> <div>"OrderId": 11077,</div> <div>"EmployeeId": 1,</div> <div>"OrderDate": "2016-05-06",</div> <div>"RequiredDate": "2016-06-03",</div> <div>"DeliveryTime": 28</div> </div> <div>}</div> </div>

Proposition #19 (Complex)

Create a scalar function which returns the gap in months between two dates then use that function to create a table that shows the difference in months between the birth date and the hire date of an employee also include the orderId EmployeeID birthdate and hire date from sales.order sales.orderDetail and HumanResources.Employee



Columns from Tables

Table Name	Column Name
Order	OrderId
OrderDetail	EmployeeId
Employee	Birthday, HireDay

Order By

Table Name	Column Name	Sort Order
monthDiff	TimeBetweenHiring	ASC

```
USE Northwinds2020TSQVLV6
GO
```

```
CREATE
    OR
```

```
ALTER FUNCTION Sales.monthDiff (
    @StartDate DATE
    ,@EndDate DATE
)
RETURNS INT
AS
BEGIN
    RETURN DATEDIFF(MONTH, @StartDate, @EndDate)
END
GO
```

```
SELECT O.OrderId
    ,D.EmployeeId
    ,D.birthdate
    ,D.HireDate
    ,Sales.monthDiff(D.birthdate, D.HireDate) AS TimeBetweenHiring
FROM Sales.[Order] AS E
INNER JOIN Sales.OrderDetail AS O ON E.OrderId = O.OrderId
INNER JOIN HumanResources.Employee AS D ON D.EmployeeId = E.EmployeeId
WHERE DATEDIFF(DAY, D.birthdate, D.HireDate) > 30
GROUP BY O.OrderId
    ,D.EmployeeId
    ,D.birthdate
    ,D.HireDate
    ,Sales.monthDiff(D.birthdate, D.HireDate)
ORDER BY Sales.monthDiff(D.birthdate, D.HireDate)
```

```

USE Northwinds2020TSQLV6
GO

CREATE
OR

ALTER FUNCTION Sales.monthDiff (
    @StartDate DATE
    ,@EndDate DATE
)
RETURNS INT
AS
BEGIN
    RETURN DATEDIFF(MONTH, @StartDate, @EndDate)
END
GO

SELECT O.OrderId
       ,D.EmployeeId

```

132 %

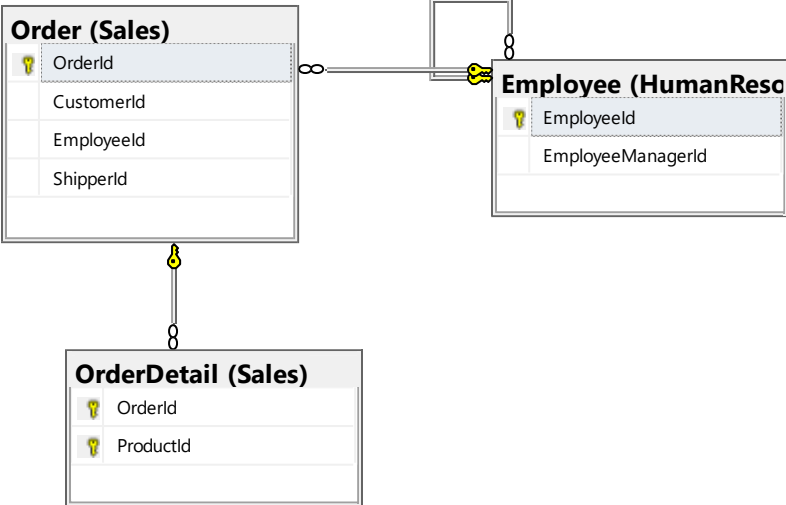
Results Messages

	OrderId	EmployeeId	birthdate	HireDate	TimeBetweenHiring
1	10251	3	1983-08-30	2013-04-01	356
2	10253	3	1983-08-30	2013-04-01	356
3	10256	3	1983-08-30	2013-04-01	356
4	10266	3	1983-08-30	2013-04-01	356
5	10273	3	1983-08-30	2013-04-01	356
6	10283	3	1983-08-30	2013-04-01	356
7	10309	3	1983-08-30	2013-04-01	356
8	10321	3	1983-08-30	2013-04-01	356
9	10330	3	1983-08-30	2013-04-01	356
10	10332	3	1983-08-30	2013-04-01	356
11	10346	3	1983-08-30	2013-04-01	356
12	10352	3	1983-08-30	2013-04-01	356
13	10362	3	1983-08-30	2013-04-01	356
14	10365	3	1983-08-30	2013-04-01	356
15	10375	3	1983-08-30	2013-04-01	356
16	10381	3	1983-08-30	2013-04-01	356
17	10384	3	1983-08-30	2013-04-01	356
18	10391	3	1983-08-30	2013-04-01	356
19	10409	3	1983-08-30	2013-04-01	356

Refresh	Search
<div> <div>[Array]</div> <div> <div>[0]: [Object]</div> <div> <div>OrderId: 10251</div> <div>EmployeeId: 3</div> <div>birthdate: "1983-08-30"</div> <div>HireDate: "2013-04-01"</div> <div>TimeBetweenHiring: 356</div> </div> </div> <div> <div>[1]: [Object]</div> <div> <div>OrderId: 10253</div> <div>EmployeeId: 3</div> <div>birthdate: "1983-08-30"</div> <div>HireDate: "2013-04-01"</div> <div>TimeBetweenHiring: 356</div> </div> </div> <div> <div>[2]: [Object]</div> <div> <div>OrderId: 10256</div> <div>EmployeeId: 3</div> <div>birthdate: "1983-08-30"</div> <div>HireDate: "2013-04-01"</div> <div>TimeBetweenHiring: 356</div> </div> </div> <div> <div>[3]: [Object]</div> <div> <div>OrderId: 10266</div> <div>EmployeeId: 3</div> <div>birthdate: "1983-08-30"</div> <div>HireDate: "2013-04-01"</div> <div>TimeBetweenHiring: 356</div> </div> </div> <div> <div>[4]: [Object]</div> <div>[5]: [Object]</div> <div>[6]: [Object]</div> <div>[7]: [Object]</div> <div>[8]: [Object]</div> <div>[9]: [Object]</div> <div>[10]: [Object]</div> <div>[11]: [Object]</div> <div>[12]: [Object]</div> <div>[13]: [Object]</div> <div>[14]: [Object]</div> <div>[15]: [Object]</div> <div>[16]: [Object]</div> <div>[17]: [Object]</div> <div>[18]: [Object]</div> <div>[19]: [Object]</div> <div>[20]: [Object]</div> <div>[21]: [Object]</div> <div>[22]: [Object]</div> <div>[23]: [Object]</div> <div>[24]: [Object]</div> <div>[25]: [Object]</div> <div>[26]: [Object]</div> <div>[27]: [Object]</div> <div>[28]: [Object]</div> <div>[29]: [Object]</div> <div>[30]: [Object]</div> <div>[31]: [Object]</div> <div>[32]: [Object]</div> <div>[33]: [Object]</div> </div> </div>	<pre> 4952 }, { 4953 "OrderId": 11044, 4954 "EmployeeId": 4, 4955 "birthdate": "1957-09-19", 4956 "HireDate": "2014-05-03", 4957 "TimeBetweenHiring": 680 4958 }, { 4959 "OrderId": 11061, 4960 "EmployeeId": 4, 4961 "birthdate": "1957-09-19", 4962 "HireDate": "2014-05-03", 4963 "TimeBetweenHiring": 680 4964 }, { 4965 "OrderId": 11062, 4966 "EmployeeId": 4, 4967 "birthdate": "1957-09-19", 4968 "HireDate": "2014-05-03", 4969 "TimeBetweenHiring": 680 4970 }, { 4971 "OrderId": 11072, 4972 "EmployeeId": 4, 4973 "birthdate": "1957-09-19", 4974 "HireDate": "2014-05-03", 4975 "TimeBetweenHiring": 680 4976 }, { 4977 "OrderId": 11076, 4978 "EmployeeId": 4, 4979 "birthdate": "1957-09-19", 4980 "HireDate": "2014-05-03", 4981 "TimeBetweenHiring": 680 4982 } 4983] 4984 </pre>

Proposition #20 (Complex)

use Northwinds2020TSQLV6 to make a scalar function to determine which quarter the DATE falls into. Then using that Information create a table that has a orderId, EmployeeID, shiptodate, and quarter from the Sales.order, Sales.OrderDetail, HumanResources.Employee table.



FK_OrderDetail_Order	Order (Sales)			Employee (HumanResources)		
	Column Name	Data Type	Allow Nulls	Column Name	Data Type	Allow Nulls
FK_OrderDetail_Order	OrderId	Udt.SurrogateKeyInt...	<input type="checkbox"/>	EmployeeId	Udt.SurrogateKeyInt...	<input type="checkbox"/>
	CustomerId	Udt.SurrogateKeyInt...	<input checked="" type="checkbox"/>	EmployeeLastName	Udt.LastName:nvarc...	<input type="checkbox"/>
	EmployeeId	Udt.SurrogateKeyInt...	<input type="checkbox"/>	EmployeeFirstName	Udt.FirstName:nvarc...	<input type="checkbox"/>
	ShipperId	Udt.SurrogateKeyInt...	<input type="checkbox"/>	EmployeeTitle	Udt.Title:nvarchar(30)	<input type="checkbox"/>
	OrderDate	Udt.DateYYYYMMD...	<input type="checkbox"/>	EmployeeTitleOfCourtesy	Udt.TitleOfCourtesy...	<input type="checkbox"/>
	RequiredDate	Udt.DateYYYYMMD...	<input type="checkbox"/>	BirthDate	Udt.DateYYYYMMD...	<input type="checkbox"/>
	ShipToDate	Udt.DateYYYYMMD...	<input checked="" type="checkbox"/>	HireDate	Udt.DateYYYYMMD...	<input type="checkbox"/>
	Freight	Udt.Currency:money	<input type="checkbox"/>	EmployeeAddress	Udt.Address:nvarcha...	<input type="checkbox"/>
	ShipToName	Udt.ContactName:n...	<input type="checkbox"/>	EmployeeCity	Udt.City:nvarchar(15)	<input checked="" type="checkbox"/>
	ShipToAddress	Udt.Address:nvarcha...	<input type="checkbox"/>	EmployeeRegion	Udt.Region:nvarchar(...)	<input checked="" type="checkbox"/>
	ShipToCity	Udt.City:nvarchar(15)	<input type="checkbox"/>	EmployeePostalCode	Udt.PostalCode:nvar...	<input checked="" type="checkbox"/>
	ShipToRegion	Udt.Region:nvarchar(...)	<input checked="" type="checkbox"/>	EmployeeCountry	Udt.Country:nvarcha...	<input type="checkbox"/>
	ShipToPostalCode	Udt.PostalCode:nvar...	<input checked="" type="checkbox"/>	EmployeePhoneNumber	Udt.TelephoneNumb...	<input type="checkbox"/>
	ShipToCountry	Udt.Country:nvarcha...	<input type="checkbox"/>	EmployeeManagerId	Udt.SurrogateKeyInt...	<input checked="" type="checkbox"/>
	UserAuthenticationId	int	<input checked="" type="checkbox"/>			<input type="checkbox"/>
	DateAdded	datetime2(7)	<input checked="" type="checkbox"/>			
	DateOfLastUpdate	datetime2(7)	<input checked="" type="checkbox"/>			
			<input type="checkbox"/>			
OrderDetail (Sales)						
	Column Name	Data Type	Allow Nulls			
	OrderId	Udt.SurrogateKeyInt...	<input type="checkbox"/>			
	ProductId	Udt.SurrogateKeyInt...	<input type="checkbox"/>			
	UnitPrice	Udt.Currency:money	<input type="checkbox"/>			
	Quantity	Udt.QuantitySmalls...	<input type="checkbox"/>			
	DiscountPercentage	Udt.Percentage:num...	<input type="checkbox"/>			
			<input type="checkbox"/>			

Columns from Tables

Table Name	Column Name
Order	ShipToDate
OrderDetail	OrderId
Employee	EmployeeId

```
USE Northwinds2020TSQLV6;
GO
```

```
CREATE
    OR
```

```
ALTER FUNCTION Sales.fq (@Date DATE)
RETURNS NVARCHAR(50)
AS
BEGIN
```

```
    -- Declare the return variable here
```

```
    DECLARE @month INT;
```

```
    DECLARE @qtr NVARCHAR(50);
```

```
    SELECT @month = MONTH(@Date);
```

```
    IF (
        @month BETWEEN 1
            AND 3
    )
```

```
        SELECT @qtr = 'QTR I'
```

```
    ELSE IF (
        @month BETWEEN 4
            AND 6
    )
```

```
        SELECT @qtr = 'QTR II'
```

```
    ELSE IF (
        @month BETWEEN 7
            AND 9
    )
```

```
        SELECT @qtr = 'QTR III'
```

```
    ELSE IF (
        @month BETWEEN 10
            AND 12
    )
```

```
        SELECT @qtr = 'QTR IIII'
```

```
    ELSE
        SELECT @qtr = 'Unknown QTR'
```

```
    RETURN @qtr
```

```
END
```

```
GO
```

```
SELECT O.OrderId,
       D.EmployeeId,
       E.ShipToDate,
       Sales.fq(E.ShipToDate) AS Quarter
```

```
FROM Sales.[Order] AS E
```

```
INNER JOIN Sales.OrderDetail AS O ON E.OrderId = O.OrderId
```

```
INNER JOIN HumanResources.Employee AS D ON D.EmployeeId = E.EmployeeId
```

```
GROUP BY O.OrderId,
         D.EmployeeId,
         Sales.fq(E.ShipToDate),
         E.ShipToDate
```

```

USE Northwinds2020TSQV6;
GO

CREATE
OR

ALTER FUNCTION Sales.fq (@Date DATE)
RETURNS NVARCHAR(50)
AS
BEGIN
    -- Declare the return variable here
    DECLARE @month INT;
    DECLARE @qtr NVARCHAR(50);

    SELECT @month = MONTH(@Date);

    IF (
        @month BETWEEN 1
        AND 3

```

132 %

Results Messages

	OrderId	EmployeeId	ShipToDate	Quarter
1	10248	5	2014-07-16	QTR III
2	10249	6	2014-07-10	QTR III
3	10250	4	2014-07-12	QTR III
4	10251	3	2014-07-15	QTR III
5	10252	4	2014-07-11	QTR III
6	10253	3	2014-07-16	QTR III
7	10254	5	2014-07-23	QTR III
8	10255	9	2014-07-15	QTR III
9	10256	3	2014-07-17	QTR III
10	10257	4	2014-07-22	QTR III
11	10258	1	2014-07-23	QTR III
12	10259	4	2014-07-25	QTR III
13	10260	4	2014-07-29	QTR III
14	10261	4	2014-07-30	QTR III
15	10262	8	2014-07-25	QTR III
16	10263	9	2014-07-31	QTR III
17	10264	6	2014-08-23	QTR III
18	10265	2	2014-08-12	QTR III
19	10266	3	2014-07-31	QTR III

Refresh	Search		
: [Array]			
[0]: [Object]			
OrderId: 10248			
EmployeeId: 5			
ShipToDate: "2014-07-16"			
Quarter: "QTR III"			
[1]: [Object]			
OrderId: 10249			
EmployeeId: 6			
ShipToDate: "2014-07-10"			
Quarter: "QTR III"			
[2]: [Object]			
[3]: [Object]			
OrderId: 10251			
EmployeeId: 3			
ShipToDate: "2014-07-15"			
Quarter: "QTR III"			
[4]: [Object]			
[5]: [Object]			
[6]: [Object]			
[7]: [Object]			
[8]: [Object]			
[9]: [Object]			
[10]: [Object]			
[11]: [Object]			
[12]: [Object]			
[13]: [Object]			
[14]: [Object]			
[15]: [Object]			
[16]: [Object]			
[17]: [Object]			
[18]: [Object]			
[19]: [Object]			
[20]: [Object]			
[21]: [Object]			
[22]: [Object]			
[23]: [Object]			
[24]: [Object]			
[25]: [Object]			
[26]: [Object]			
[27]: [Object]			
[28]: [Object]			
[29]: [Object]			
[30]: [Object]			
[31]: [Object]			
[32]: [Object]			
[33]: [Object]			
[34]: [Object]			
[35]: [Object]			
[36]: [Object]			
[37]: [Object]			
[38]: [Object]			
[39]: [Object]			
[40]: [Object]			
[41]: [Object]			
[42]: [Object]			
[43]: [Object]			
[44]: [Object]			
[45]: [Object]			
[46]: [Object]			
[47]: [Object]			
[48]: [Object]			
[49]: [Object]			
[50]: [Object]			
[51]: [Object]			
[52]: [Object]			
[53]: [Object]			
[54]: [Object]			
[55]: [Object]			
[56]: [Object]			
[57]: [Object]			
[58]: [Object]			
[59]: [Object]			
[60]: [Object]			
[61]: [Object]			
[62]: [Object]			
[63]: [Object]			
[64]: [Object]			
[65]: [Object]			
[66]: [Object]			
[67]: [Object]			
[68]: [Object]			
[69]: [Object]			
[70]: [Object]			
[71]: [Object]			
[72]: [Object]			
[73]: [Object]			
[74]: [Object]			
[75]: [Object]			
[76]: [Object]			
[77]: [Object]			
[78]: [Object]			
[79]: [Object]			
[80]: [Object]			
[81]: [Object]			
[82]: [Object]			
[83]: [Object]			
[84]: [Object]			
[85]: [Object]			
[86]: [Object]			
[87]: [Object]			
[88]: [Object]			
[89]: [Object]			
[90]: [Object]			
[91]: [Object]			
[92]: [Object]			
[93]: [Object]			
[94]: [Object]			
[95]: [Object]			
[96]: [Object]			
[97]: [Object]			
[98]: [Object]			
[99]: [Object]			
[100]: [Object]			
[101]: [Object]			
[102]: [Object]			
[103]: [Object]			
[104]: [Object]			
[105]: [Object]			
[106]: [Object]			
[107]: [Object]			
[108]: [Object]			
[109]: [Object]			
[110]: [Object]			
[111]: [Object]			
[112]: [Object]			
[113]: [Object]			
[114]: [Object]			
[115]: [Object]			
[116]: [Object]			
[117]: [Object]			
[118]: [Object]			
[119]: [Object]			
[120]: [Object]			
[121]: [Object]			
[122]: [Object]			
[123]: [Object]			
[124]: [Object]			
[125]: [Object]			
[126]: [Object]			
[127]: [Object]			
[128]: [Object]			
[129]: [Object]			
[130]: [Object]			
[131]: [Object]			
[132]: [Object]			
[133]: [Object]			
[134]: [Object]			
[135]: [Object]			
[136]: [Object]			
[137]: [Object]			
[138]: [Object]			
[139]: [Object]			
[140]: [Object]			
[141]: [Object]			
[142]: [Object]			
[143]: [Object]			
[144]: [Object]			
[145]: [Object]			
[146]: [Object]			
[147]: [Object]			
[148]: [Object]			
[149]: [Object]			
[150]: [Object]			
[151]: [Object]			
[152]: [Object]			
[153]: [Object]			
[154]: [Object]			
[155]: [Object]			
[156]: [Object]			
[157]: [Object]			
[158]: [Object]			
[159]: [Object]			
[160]: [Object]			
[161]: [Object]			
[162]: [Object]			
[163]: [Object]			
[164]: [Object]			
[165]: [Object]			
[166]: [Object]			
[167]: [Object]			
[168]: [Object]			
[169]: [Object]			
[170]: [Object]			
[171]: [Object]			
[172]: [Object]			
[173]: [Object]			
[174]: [Object]			
[175]: [Object]			
[176]: [Object]			
[177]: [Object]			
[178]: [Object]			
[179]: [Object]			
[180]: [Object]			
[181]: [Object]			
[182]: [Object]			
[183]: [Object]			
[184]: [Object]			
[185]: [Object]			
[186]: [Object]			
[187]: [Object]			
[188]: [Object]			
[189]: [Object]			
[190]: [Object]			
[191]: [Object]			
[192]: [Object]			
[193]: [Object]			
[194]: [Object]			
[195]: [Object]			
[196]: [Object]			
[197]: [Object]			
[198]: [Object]			
[199]: [Object]			
[200]: [Object]			
[201]: [Object]			
[202]: [Object]			
[203]: [Object]			
[204]: [Object]			
[205]: [Object]			
[206]: [Object]			
[207]: [Object]			
[208]: [Object]			
[209]: [Object]			
[210]: [Object]			
[211]: [Object]			
[212]: [Object]			
[213]: [Object]			
[214]: [Object]			
[215]: [Object]			
[216]: [Object]			
[217]: [Object]			
[218]: [Object]			
[219]: [Object]			
[220]: [Object]			
[221]: [Object]			
[222]: [Object]			
[223]: [Object]			
[224]: [Object]			
[225]: [Object]			