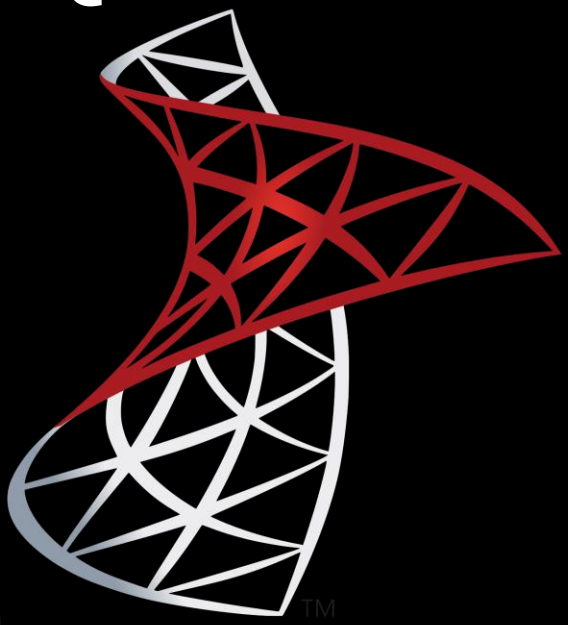
SQL Questions



Microsoft® SQL Server®

Question in SQL Server Syntax

Xhoni Shollaj
EY Business Analyst

Database Used: TSQL 3

1. **Return the total value of orders handled by each employee for each order year**
   (Desired Output Below)

	empid	2013	2014	2015
1	9	9894.52	26310.39	41103.17
2	3	18223.96	108026.17	76562.75
3	6	16642.61	43126.38	14144.16
4	7	15232.16	60471.19	48864.89
5	1	35764.52	93148.11	63195.02
6	4	49945.12	128809.81	54135.94
7	5	18383.92	30716.48	19691.90
8	2	21757.06	70444.14	74336.56
9	8	22240.12	56032.63	48589.54

2. **Return customers for whom all employees in the organization handled orders.**
   (Desired Output Below)

	custid
1	71

3. **Return orders that were placed on the last date of activity of the month**
   (Desired Output Below)

	orderid	orderdate	custid	empid
1	10269	2013-07-31	89	5
2	10294	2013-08-30	65	4
3	10317	2013-09-30	48	6
4	10343	2013-10-31	44	4
5	10368	2013-11-29	20	2
6	10399	2013-12-31	83	8
7	10432	2014-01-31	75	3
8	10460	2014-02-28	24	8
9	10461	2014-02-28	46	1
10	10490	2014-03-31	35	7
11	10491	2014-03-31	28	8
12	10522	2014-04-30	44	4
13	10553	2014-05-30	87	2
14	10554	2014-05-30	56	4

4. Return the orders that were placed on the last date of activity of the customer
(Desired Output Below)

	orderid	orderdate	custid	empid
1	11044	2015-04-23	91	4
2	11005	2015-04-07	90	2
3	11066	2015-05-01	89	7
4	10935	2015-03-09	88	4
5	11025	2015-04-15	87	6
6	11046	2015-04-23	86	8
7	10739	2014-11-12	85	3
8	10850	2015-01-23	84	1
9	10994	2015-04-02	83	2
10	10822	2015-01-08	82	6
11	10839	2015-01-19	81	3
12	11069	2015-05-04	80	1
13	10967	2015-03-23	79	2
14	11003	2015-04-06	78	3

5. Return customers who placed orders
(Desired Output Below)

	custid	companyname
1	1	Customer NRZBB
2	2	Customer MLTDN
3	3	Customer KBUDE
4	4	Customer HFBZG
5	5	Customer HGVLZ
6	6	Customer XHXJV
7	7	Customer QXVLA
8	8	Customer QUHWH
9	9	Customer RTXGC
10	10	Customer EEALV
11	11	Customer UBHAU
12	12	Customer PSNMQ
13	13	Customer VMLOG
14	14	Customer WNMAF

6. **Return order years and the distinct number of customers handled in each year for years that had more than 70 distinct customers handled.**
(Desired Output Below)

	orderyear	numcusts
1	2014	86
2	2015	81

7. **Computes the number of orders that were handled in each year, and the difference from the count of the previous year.**
(Desired Output Below)

	orderyear	numorders	diff
1	2013	152	NULL
2	2014	408	256
3	2015	270	-138

8. **Return customers and their orders, (including customers with no orders).**
(Desired Output Below)

	custid	companyname	country	orderid	shipcountry
1	85	Customer ENQZT	France	10248	France
2	79	Customer FAPSM	Germany	10249	Germany
3	34	Customer IBVRG	Brazil	10250	Brazil
4	84	Customer NRCSK	France	10251	France
5	76	Customer SFOGW	Belgium	10252	Belgium
6	34	Customer IBVRG	Brazil	10253	Brazil
7	14	Customer WNMAF	Switzerland	10254	Switzerland
8	68	Customer CCKOT	Switzerland	10255	Switzerland
9	88	Customer SRQVM	Brazil	10256	Brazil
10	35	Customer UMTLM	Venezuela	10257	Venezuela
11	20	Customer THHDP	Austria	10258	Austria
12	13	Customer VMLOG	Mexico	10259	Mexico
13	56	Customer QNIVZ	Germany	10260	Germany
14	61	Customer WULWD	Brazil	10261	Brazil

9. Returns only customers who didn't place orders

(Desired Output)

	custid	companyname	orderid
1	22	Customer DTDMN	NULL
2	57	Customer WVAXS	NULL

10. Match employees with their managers, who are also employees

(Desired Output)

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

11. Use a non-equi join with a less than (<) operator to identify unique pairs of employees.

(Desired Output)

	empid	lastname	firstname	empid	lastname	firstname
1	1	Davis	Sara	2	Funk	Don
2	1	Davis	Sara	3	Lew	Judy
3	2	Funk	Don	3	Lew	Judy
4	1	Davis	Sara	4	Peled	Yael
5	2	Funk	Don	4	Peled	Yael
6	3	Lew	Judy	4	Peled	Yael
7	1	Davis	Sara	5	Mortensen	Sven
8	2	Funk	Don	5	Mortensen	Sven
9	3	Lew	Judy	5	Mortensen	Sven
10	4	Peled	Yael	5	Mortensen	Sven
11	1	Davis	Sara	6	Suurs	Paul
12	2	Funk	Don	6	Suurs	Paul
13	3	Lew	Judy	6	Suurs	Paul
14	4	Peled	Yael	6	Suurs	Paul

12. Join five tables to return customer-supplier pairs that had activity together.
(Desired Output)

	customer	supplier
1	Customer AHPOP	Supplier BWGYE
2	Customer AHPOP	Supplier ELCRN
3	Customer AHPOP	Supplier EQPNC
4	Customer AHPOP	Supplier ERVYZ
5	Customer AHPOP	Supplier GQRCV
6	Customer AHPOP	Supplier JDNUG
7	Customer AHPOP	Supplier JNNES
8	Customer AHPOP	Supplier KEREV
9	Customer AHPOP	Supplier LVJUA
10	Customer AHPOP	Supplier OAVQT
11	Customer AHPOP	Supplier QOVFD
12	Customer AHPOP	Supplier QQYEU
13	Customer AHPOP	Supplier QZGUF
14	Customer AHPOP	Supplier SVIYA

13. Return the customer ID and company name of customers who placed orders.
(Desired Output)

	custid	companyname
1	1	Customer NRZBB
2	2	Customer MLTDN
3	3	Customer KBUDE
4	4	Customer HFBZG
5	5	Customer HGVLZ
6	6	Customer XHXJV
7	7	Customer QXVLA
8	8	Customer QUHWH
9	9	Customer RTXGC
10	10	Customer EEALV
11	11	Customer UBHAU
12	12	Customer PSNMQ
13	13	Customer VMLOG
14	14	Customer WNMAF

14. Return locations that are both employee locations and customer locations.

	country	region	city
1	UK	NULL	London
2	USA	WA	Kirkland
3	USA	WA	Seattle

15. Return distinct employee locations that are not customer locations.

	country	region	city
1	USA	WA	Redmond
2	USA	WA	Tacoma

16. Compute rank distribution information of student test scores (Percent Rank and CumeDist)

	testid	studentid	score	percentrank	cumedist
1	Test ABC	Student E	50	0.00	11.11
2	Test ABC	Student C	55	12.50	33.33
3	Test ABC	Student D	55	12.50	33.33
4	Test ABC	Student H	65	37.50	44.44
5	Test ABC	Student I	75	50.00	55.56
6	Test ABC	Student B	80	62.50	77.78
7	Test ABC	Student F	80	62.50	77.78
8	Test ABC	Student A	95	87.50	100.00
9	Test ABC	Student G	95	87.50	100.00
10	Test XYZ	Student E	50	0.00	10.00
11	Test XYZ	Student C	55	11.11	30.00
12	Test XYZ	Student D	55	11.11	30.00
13	Test XYZ	Student H	65	33.33	40.00
14	Test XYZ	Student I	75	44.44	50.00

17. Compute the median student test scores within the same test using both models (Inverse Distribution Functions)

	testid	studentid	score	mediandisc	mediancont
1	Test ABC	Student E	50	75	75
2	Test ABC	Student C	55	75	75
3	Test ABC	Student D	55	75	75
4	Test ABC	Student H	65	75	75
5	Test ABC	Student I	75	75	75
6	Test ABC	Student B	80	75	75
7	Test ABC	Student F	80	75	75
8	Test ABC	Student A	95	75	75
9	Test ABC	Student G	95	75	75
10	Test XYZ	Student E	50	75	77.5
11	Test XYZ	Student C	55	75	77.5
12	Test XYZ	Student D	55	75	77.5
13	Test XYZ	Student H	65	75	77.5
14	Test XYZ	Student I	75	75	77.5

18. Pivot data so that you return a row per customer, a column per order year, and in the intersection of each customer and year, to return the sum of applicable values (Many to One Pivot).

	custid	2013	2014	2015
1	1	NULL	2022.50	2250.50
2	2	88.80	799.75	514.40
3	3	403.20	5960.78	660.00
4	4	1379.00	6406.90	5604.75
5	5	4324.40	13849.02	6754.16
6	6	NULL	1079.80	2160.00
7	7	9986.20	7817.88	730.00
8	8	982.00	3026.85	224.00
9	9	4074.28	11208.36	6680.61
10	10	1832.80	7630.25	11338.56
11	11	479.40	3179.50	2431.00
12	12	NULL	238.00	1576.80
13	13	100.80	NULL	NULL
14	14	1674.22	6516.40	4158.26

19. Find for each customer the employee who handled the most orders. That's the employee ID that appears the most times in the customer's orders. In case of ties in the count, you're supposed to return the greater employee ID.

	custid	empid	cnt
1	1	4	2
2	2	3	2
3	3	3	3
4	4	4	4
5	5	3	6
6	6	9	3
7	7	4	3
8	8	4	2
9	9	4	4
10	10	3	4
11	11	6	2
12	12	8	2
13	13	4	1
14	14	6	2

20. Get three most recent orders.

	orderid	orderdate	custid	empid
1	11077	2015-05-06	65	1
2	11076	2015-05-06	9	4
3	11075	2015-05-06	68	8

21. Return most recent order for each customer.

	orderid	orderdate	custid	empid
1	11011	2015-04-09	1	3
2	10926	2015-03-04	2	4
3	11016	2015-04-10	4	9
4	10924	2015-03-04	5	3
5	10856	2015-01-28	3	3
6	11058	2015-04-29	6	9
7	10826	2015-01-12	7	6
8	11076	2015-05-06	9	4
9	11048	2015-04-24	10	7
10	11023	2015-04-14	11	1
11	10259	2013-07-18	13	4
12	11041	2015-04-22	14	3
13	11054	2015-04-28	12	8
14	11042	2015-04-22	15	2

22. Create a query in which you skip the first 50 rows and filter the next 25 rows.

	orderid	orderdate	custid	empid
1	11027	2015-04-16	10	1
2	11026	2015-04-15	27	4
3	11025	2015-04-15	87	6
4	11024	2015-04-15	19	4
5	11023	2015-04-14	11	1
6	11022	2015-04-14	34	9
7	11021	2015-04-14	63	3
8	11020	2015-04-14	56	2
9	11019	2015-04-13	64	6
10	11018	2015-04-13	48	4
11	11017	2015-04-13	20	9
12	11016	2015-04-10	4	9
13	11015	2015-04-10	70	2
14	11014	2015-04-10	47	2

Database Used: TSQL 4

23. Filters orders that were placed by customer 71, groups those orders by employee and order year, and filters only groups of employees and years that have more than one order.

	empid	orderyear	numorders
1	1	2015	2
2	1	2016	3
3	2	2016	2
4	3	2015	2
5	4	2016	3
6	5	2015	3
7	6	2015	3
8	7	2016	2
9	8	2015	4

24. Return the total freight and number of orders per employee and order year.

	empid	orderyear	totalfreight	numorders
1	1	2014	126.56	1
2	2	2014	89.16	1
3	9	2014	214.27	1
4	1	2015	711.13	2
5	2	2015	352.69	1
6	3	2015	297.65	2
7	4	2015	86.53	1
8	5	2015	277.14	3
9	6	2015	628.31	3
10	7	2015	388.98	1
11	8	2015	371.07	4
12	1	2016	357.44	3
13	2	2016	672.16	2
14	4	2016	651.83	3

25. Return all customers for whom the region is different than WA.

	custid	country	region	city
1	10	Canada	BC	Tsawassen
2	15	Brazil	SP	Sao Paulo
3	21	Brazil	SP	Sao Paulo
4	31	Brazil	SP	Campinas
5	32	USA	OR	Eugene
6	33	Venezuela	DF	Caracas
7	34	Brazil	RJ	Rio de Janeiro
8	35	Venezuela	Táchira	San Cristóbal
9	36	USA	OR	Elgin
10	37	Ireland	Co. Cork	Cork
11	38	UK	Isle of Wight	Cowes
12	42	Canada	BC	Vancouver
13	45	USA	CA	San Francisco
14	46	Venezuela	Lara	Barquisimeto

26. Produce the fullname result column by concatenating firstname, a space, and lastname.

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

27. Return for each employee, the number of times the character ‘e’ appears in the lastname attribute.

	empid	lastname	numoccur
1	8	Cameron	1
2	1	Davis	0
3	9	Doyle	1
4	2	Funk	0
5	7	King	0
6	3	Lew	1
7	5	Mortensen	2
8	4	Peled	2
9	6	Suurs	0

28. Return employees where the last name starts with D.

	empid	lastname
1	1	Davis
2	9	Doyle

29. Return employees where the second character in the last name is 'e'.

	empid	lastname
1	3	Lew
2	4	Peled

30. Return employees where the first character in the last name is not a letter in the range A through E.

	empid	lastname
1	2	Funk
2	7	King
3	3	Lew
4	5	Mortensen
5	4	Peled
6	6	Suurs

31. Return orders placed on the last day of the month.

	orderid	orderdate	custid	empid
1	10269	2014-07-31	89	5
2	10317	2014-09-30	48	6
3	10343	2014-10-31	44	4
4	10399	2014-12-31	83	8
5	10432	2015-01-31	75	3
6	10460	2015-02-28	24	8
7	10461	2015-02-28	46	1
8	10490	2015-03-31	35	7
9	10491	2015-03-31	28	8
10	10522	2015-04-30	44	4
11	10583	2015-06-30	87	2
12	10584	2015-06-30	7	4
13	10616	2015-07-31	32	1
14	10617	2015-07-31	32	4

32. Write a query against the Sales.Orders table that calculates row numbers for orders based on order date ordering (using the order ID as the tiebreaker) for each customer separately.

	custid	orderdate	orderid	rownum
1	1	2015-08-25	10643	1
2	1	2015-10-03	10692	2
3	1	2015-10-13	10702	3
4	1	2016-01-15	10835	4
5	1	2016-03-16	10952	5
6	1	2016-04-09	11011	6
7	2	2014-09-18	10308	1
8	2	2015-08-08	10625	2
9	2	2015-11-28	10759	3
10	2	2016-03-04	10926	4
11	3	2014-11-27	10365	1
12	3	2015-04-15	10507	2
13	3	2015-05-13	10535	3
14	3	2015-06-19	10573	4

33. Write a query against the Sales.Orders table that returns the three shipped-to countries with the highest average freight in 2015.

	shipcountry	avgfreight
1	Austria	178.3642
2	Switzerland	117.1775
3	Sweden	105.16

34. Join the Customers and Orders tables to match customers with their orders, and then join the result of the first join with the OrderDetails table to match orders with the order lines.

	custid	companyname	orderid	productid	qty
1	85	Customer ENQZT	10248	11	12
2	85	Customer ENQZT	10248	42	10
3	85	Customer ENQZT	10248	72	5
4	79	Customer FAPSM	10249	14	9
5	79	Customer FAPSM	10249	51	40
6	34	Customer IBVRG	10250	41	10
7	34	Customer IBVRG	10250	51	35
8	34	Customer IBVRG	10250	65	15
9	84	Customer NRCSK	10251	22	6
10	84	Customer NRCSK	10251	57	15
11	84	Customer NRCSK	10251	65	20
12	76	Customer SFOGW	10252	20	40
13	76	Customer SFOGW	10252	33	25
14	76	Customer SFOGW	10252	60	40

35. Join the Customers and Orders tables, based on a match between the customer's customer ID and the order 's customer ID, to return customers and their orders.

	custid	companyname	orderid
1	1	Customer NRZBB	10643
2	1	Customer NRZBB	10692
3	1	Customer NRZBB	10702
4	1	Customer NRZBB	10835
5	1	Customer NRZBB	10952
6	1	Customer NRZBB	11011
7	2	Customer MLTDN	10308
8	2	Customer MLTDN	10625
9	2	Customer MLTDN	10759
10	2	Customer MLTDN	10926
11	3	Customer KBUDE	10365
12	3	Customer KBUDE	10507
13	3	Customer KBUDE	10535
14	3	Customer KBUDE	10573

36. Return only customers who did not place any orders.

	custid	companyname
1	22	Customer DTDMN
2	57	Customer WVAXS

37. Return the count of orders for each customer

	custid	numorders
1	1	6
2	2	4
3	3	7
4	4	13
5	5	18
6	6	7
7	7	11
8	8	3
9	9	17
10	10	14
11	11	10
12	12	6
13	13	1
14	14	8

38. Return US customers, and for each customer return the total number of orders and total quantities.

	custid	numorders	totalqty
1	32	11	345
2	36	5	122
3	43	2	20
4	45	4	181
5	48	8	134
6	55	10	603
7	65	18	1383
8	71	31	4958
9	75	9	327
10	77	4	46
11	78	3	59
12	82	3	89
13	89	14	1063

39. Return customers and their orders, including customers who placed no orders.

	custid	companyname	orderid	orderdate
1	85	Customer ENQZT	10248	2014-07-04
2	79	Customer FAPSM	10249	2014-07-05
3	34	Customer IBVRG	10250	2014-07-08
4	84	Customer NRCSK	10251	2014-07-08
5	76	Customer SFOGW	10252	2014-07-09
6	34	Customer IBVRG	10253	2014-07-10
7	14	Customer WNMAF	10254	2014-07-11
8	68	Customer CCKOT	10255	2014-07-12
9	88	Customer SRQVM	10256	2014-07-15
10	35	Customer UMTLM	10257	2014-07-16
11	20	Customer THHDP	10258	2014-07-17
12	13	Customer VMLOG	10259	2014-07-18
13	56	Customer QNIVZ	10260	2014-07-19
14	61	Customer WULWD	10261	2014-07-19

40. Return customers with orders placed on February 12, 2016, along with their orders.

	custid	companyname	orderid	orderdate
1	48	Customer DVFMB	10883	2016-02-12
2	45	Customer QXPPT	10884	2016-02-12
3	76	Customer SFOGW	10885	2016-02-12

41. Return all customers, and for each return a Yes/No value depending on whether the customer placed orders on February 12, 2016.

	custid	companyname	HasOrderOn20160212
1	1	Customer NRZBB	No
2	2	Customer MLTDN	No
3	3	Customer KBUDE	No
4	4	Customer HFBZG	No
5	5	Customer HGVLZ	No
6	6	Customer XHXJV	No
7	7	Customer QXVLA	No
8	8	Customer QUHWH	No
9	9	Customer RTXGC	No
10	10	Customer EEALV	No
11	11	Customer UBHAU	No
12	12	Customer PSNMQ	No
13	13	Customer VMLOG	No
14	14	Customer WNMAF	No

42. Return orders placed by employees with a last name starting with D.

	orderid
1	10258
2	10270
3	10275
4	10285
5	10292
6	10293
7	10304
8	10306
9	10311
10	10314
11	10316
12	10325
13	10340
14	10351

43. Write a query that returns orders placed by customers from the United States

	custid	orderid	orderdate	empid
1	65	10262	2014-07-22	8
2	89	10269	2014-07-31	5
3	75	10271	2014-08-01	6
4	65	10272	2014-08-02	6
5	65	10294	2014-08-30	4
6	55	10305	2014-09-13	8
7	48	10307	2014-09-17	2
8	77	10310	2014-09-20	8
9	65	10314	2014-09-25	1
10	65	10316	2014-09-27	1
11	48	10317	2014-09-30	6
12	71	10324	2014-10-08	9
13	75	10329	2014-10-15	4
14	55	10338	2014-10-25	4

44. Returns orders with the maximum order ID for each customer.

	custid	orderid	orderdate	empid
1	91	11044	2016-04-23	4
2	90	11005	2016-04-07	2
3	89	11066	2016-05-01	7
4	88	10935	2016-03-09	4
5	87	11025	2016-04-15	6
6	86	11046	2016-04-23	8
7	85	10739	2015-11-12	3
8	84	10850	2016-01-23	1
9	83	10994	2016-04-02	2
10	82	10822	2016-01-08	6
11	81	10839	2016-01-19	3
12	80	11069	2016-05-04	1
13	79	10967	2016-03-23	2
14	78	11003	2016-04-06	3

45. Return customers from Spain who did not place orders.

	custid	companyname
1	22	Customer DTDMN

46. query the Orders table in the TSQLV4 database and return, for each order, information about the current order and also the previous order ID

	orderid	orderdate	empid	custid	prevorderid
1	10248	2014-07-04	5	85	NULL
2	10249	2014-07-05	6	79	10248
3	10250	2014-07-08	4	34	10249
4	10251	2014-07-08	3	84	10250
5	10252	2014-07-09	4	76	10251
6	10253	2014-07-10	3	34	10252
7	10254	2014-07-11	5	14	10253
8	10255	2014-07-12	9	68	10254
9	10256	2014-07-15	3	88	10255
10	10257	2014-07-16	4	35	10256
11	10258	2014-07-17	1	20	10257
12	10259	2014-07-18	4	13	10258
13	10260	2014-07-19	4	56	10259
14	10261	2014-07-19	4	61	10260

47. Compute for each year the running total quantity up to and including that year's. For the earliest year recorded in the view (2014), the running total is equal to that year's quantity. For the second year (2015), the running total is the sum of the first year plus the second year, and so on.

	orderyear	qty	runqty
1	2014	9581	9581
2	2015	25489	35070
3	2016	16247	51317

48. Write a query that returns all orders placed on the last day of activity that can be found in the Orders table.

	orderid	orderdate	custid	empid
1	11077	2016-05-06	65	1
2	11076	2016-05-06	9	4
3	11075	2016-05-06	68	8
4	11074	2016-05-06	73	7

49. Write a query that returns countries where there are customers but not employees.

	country
1	Argentina
2	Austria
3	Belgium
4	Brazil
5	Canada
6	Denmark
7	Finland
8	France
9	Germany
10	Ireland
11	Italy
12	Mexico
13	Norway
14	Poland

50. Write a query that calculates a running-total quantity for each customer and month.

	custid	ordermonth	qty	runqty
1	1	2015-08-01	38	38
2	1	2015-10-01	41	79
3	1	2016-01-01	17	96
4	1	2016-03-01	18	114
5	1	2016-04-01	60	174
6	2	2014-09-01	6	6
7	2	2015-08-01	18	24
8	2	2015-11-01	10	34
9	2	2016-03-01	29	63
10	3	2014-11-01	24	24
11	3	2015-04-01	30	54
12	3	2015-05-01	80	134
13	3	2015-06-01	83	217
14	3	2015-09-01	102	319

51. Write a query that returns for each order the number of days that passed since the same customer's previous order. To determine recency among orders, use orderdate as the primary sort element and orderid as the tiebreaker:

	custid	orderdate	orderid	prevdate
1	1	2015-08-25	10643	NULL
2	1	2015-10-03	10692	2015-08-25
3	1	2015-10-13	10702	2015-10-03
4	1	2016-01-15	10835	2015-10-13
5	1	2016-03-16	10952	2016-01-15
6	1	2016-04-09	11011	2016-03-16
7	2	2014-09-18	10308	NULL
8	2	2015-08-08	10625	2014-09-18
9	2	2015-11-28	10759	2015-08-08
10	2	2016-03-04	10926	2015-11-28
11	3	2014-11-27	10365	NULL
12	3	2015-04-15	10507	2014-11-27
13	3	2015-05-13	10535	2015-04-15
14	3	2015-06-19	10573	2015-05-13

52. Write a query that returns the maximum value in the orderdate column for each employee. Encapsulate the query in a derived table. Write a join query between the derived table and the Orders table to return the orders with the maximum order date for each employee.

	empid	orderdate	orderid	custid
1	9	2016-04-29	11058	6
2	8	2016-05-06	11075	68
3	7	2016-05-06	11074	73
4	6	2016-04-23	11045	10
5	5	2016-04-22	11043	74
6	4	2016-05-06	11076	9
7	3	2016-04-30	11063	37
8	2	2016-05-05	11073	58
9	2	2016-05-05	11070	44
10	1	2016-05-06	11077	65

53. Write a query that calculates a row number for each order based on orderdate, orderid ordering. After that, write a query that returns rows with row numbers 11 through 20 based on the row-number definition above. Use a CTE to encapsulate the code previously used.

	orderid	orderdate	custid	empid	rownum
1	10258	2014-07-17	20	1	11
2	10259	2014-07-18	13	4	12
3	10260	2014-07-19	56	4	13
4	10261	2014-07-19	61	4	14
5	10262	2014-07-22	65	8	15
6	10263	2014-07-23	20	9	16
7	10264	2014-07-24	24	6	17
8	10265	2014-07-25	7	2	18
9	10266	2014-07-26	87	3	19
10	10267	2014-07-29	25	4	20

54. Write a solution using a recursive CTE that returns the management chain leading to Patricia Doyle (employee ID 9):

	empid	mgrid	firstname	lastname
1	9	5	Patricia	Doyle
2	5	2	Sven	Mortensen
3	2	1	Don	Funk
4	1	NULL	Sara	Davis

55. Return distinct locations that are both employee locations and customer locations.

	country	region	city
1	UK	NULL	London
2	USA	WA	Kirkland
3	USA	WA	Seattle

56. Return the number of distinct locations that are either employee or customer locations in each country.

	country	numlocations
1	Argentina	1
2	Austria	2
3	Belgium	2
4	Brazil	4
5	Canada	3
6	Denmark	2
7	Finland	2
8	France	9
9	Germany	11
10	Ireland	1
11	Italy	3
12	Mexico	1
13	Norway	1
14	Poland	1

57. Use TOP queries to return the two most recent orders for employees 3 and 5.

	empid	orderid	orderdate
1	3	11063	2016-04-30
2	3	11057	2016-04-29
3	5	11043	2016-04-22
4	5	10954	2016-03-17

58. Write a query that returns customer and employee pairs that had order activity in both January 2016 and February 2016.

	custid	empid
1	20	3
2	39	9
3	46	5
4	67	1
5	71	4

59. Use ranking window functions to rank each row with respect to others in the window. (ROW_NUMBER, RANK, DENSE_RANK, and NTILE)
(Desired Output Below)

	orderid	custid	val	rownum	rank	dense_rank	ntile
1	10782	12	12.50	1	1	1	1
2	10807	27	18.40	2	2	2	1
3	10586	66	23.80	3	3	3	1
4	10767	76	28.00	4	4	4	1
5	10898	54	30.00	5	5	5	1
6	10900	88	33.75	6	6	6	1
7	10883	48	36.00	7	7	7	1
8	11051	41	36.00	8	7	7	1
9	10815	71	40.00	9	9	8	1
10	10674	38	45.00	10	10	9	2
11	11057	53	45.00	11	10	9	2
12	10271	75	48.00	12	12	10	2
13	10602	83	48.75	13	13	11	2
14	10422	27	49.80	14	14	12	2

60. Use a window aggregate function to compute the running-total values for each employee and month.
(Desired Output Below)

	empid	ordermonth	val	runval
1	1	2014-07-01	1614.88	1614.88
2	1	2014-08-01	5555.90	7170.78
3	1	2014-09-01	6651.00	13821.78
4	1	2014-10-01	3933.18	17754.96
5	1	2014-11-01	9562.65	27317.61
6	1	2014-12-01	8446.91	35764.52
7	1	2015-01-01	7331.60	43096.12
8	1	2015-02-01	1946.40	45042.52
9	1	2015-03-01	5124.08	50166.60
10	1	2015-04-01	240.00	50406.60
11	1	2015-05-01	9115.96	59522.56
12	1	2015-06-01	5468.35	64990.91
13	1	2015-07-01	19530.93	84521.84
14	1	2015-08-01	5104.70	89626.54

Database Used: TSQL 5

(Window Functions)

61. Calculate the percent of the current order value out of the grand total, as well as out of the customer total.

	orderid	custid	val	pctall	pctcust
1	10643	1	814.50	0.06	19.06
2	10692	1	878.00	0.07	20.55
3	10702	1	330.00	0.03	7.72
4	10835	1	845.80	0.07	19.79
5	10952	1	471.20	0.04	11.03
6	11011	1	933.50	0.07	21.85
7	10926	2	514.40	0.04	36.67
8	10759	2	320.00	0.03	22.81
9	10625	2	479.75	0.04	34.20
10	10308	2	88.80	0.01	6.33
11	10365	3	403.20	0.03	5.74
12	10507	3	749.06	0.06	10.66
13	10535	3	1940.85	0.15	27.63
14	10677	3	813.37	0.06	11.58

62. Calculates the difference between the current quantity and the employee monthly average up to three months before the present date (not the current row's month).

	empid	ordermonth	qty	diff
1	1	2017-07-01	121	-218
2	1	2017-08-01	247	-92
3	1	2017-09-01	255	-84
4	1	2017-10-01	143	-196
5	1	2017-11-01	318	-21
6	1	2017-12-01	536	197
7	1	2018-01-01	304	-35
8	1	2018-02-01	168	-171
9	1	2018-03-01	275	-64
10	1	2018-04-01	20	-319
11	1	2018-05-01	337	-2
12	1	2018-06-01	260	-79
13	1	2018-07-01	652	313
14	1	2018-08-01	204	-135

63. Return with each order the number of distinct customers that were handled by the current employee up to, and including, the current date.

	empid	orderdate	orderid	custid	val	distinct_custid
1	1	2019-01-15	10835	1	845.80	1
2	1	2019-03-16	10952	1	471.20	NULL
3	1	2018-09-22	10677	3	813.37	3
4	1	2018-02-21	10453	4	407.70	4
5	1	2018-06-04	10558	4	2142.90	NULL
6	1	2018-11-17	10743	4	319.20	NULL
7	1	2018-05-01	10524	5	3192.65	5
8	1	2018-08-11	10626	5	1503.60	NULL
9	1	2018-10-01	10689	5	472.50	NULL
10	1	2018-11-07	10733	5	1459.00	NULL
11	1	2017-10-29	10340	9	2436.18	9
12	1	2018-05-02	10525	9	818.40	NULL
13	1	2019-01-12	10827	9	843.00	NULL
14	1	2019-03-25	10975	10	717.50	10

64. Use the NTILE function to arrange the rows from the OrderValues view in 10 equally sized tiles based on val ordering.

	orderid	val	rownum	tile
1	10782	12.50	1	1
2	10807	18.40	2	1
3	10586	23.80	3	1
4	10767	28.00	4	1
5	10898	30.00	5	1
6	10900	33.75	6	1
7	10883	36.00	7	1
8	11051	36.00	8	1
9	10815	40.00	9	1
10	10674	45.00	10	1
11	11057	45.00	11	1
12	10271	48.00	12	1
13	10602	48.75	13	1
14	10422	49.80	14	1

65. Computes both the percentile rank and cumulative distribution of student test scores, partitioned by testid and ordered by score.

	testid	studentid	score	percentrank	cumedist
1	Test ABC	Student E	50	0	0.111111111111111
2	Test ABC	Student C	55	0.125	0.333333333333333
3	Test ABC	Student D	55	0.125	0.333333333333333
4	Test ABC	Student H	65	0.375	0.444444444444444
5	Test ABC	Student I	75	0.5	0.555555555555556
6	Test ABC	Student B	80	0.625	0.777777777777778
7	Test ABC	Student F	80	0.625	0.777777777777778
8	Test ABC	Student A	95	0.875	1
9	Test ABC	Student G	95	0.875	1
10	Test XYZ	Student E	50	0	0.1
11	Test XYZ	Student C	55	0.111111111111111	0.3
12	Test XYZ	Student D	55	0.111111111111111	0.3
13	Test XYZ	Student H	65	0.333333333333333	0.4
14	Test XYZ	Student I	75	0.444444444444444	0.5

66. Return the current order value for each customer order, as well as the values of the previous and next orders by the same customer.

	custid	orderdate	orderid	val	prevval	nextval
1	1	2018-08-25	10643	814.50	NULL	878.00
2	1	2018-10-03	10692	878.00	814.50	330.00
3	1	2018-10-13	10702	330.00	878.00	845.80
4	1	2019-01-15	10835	845.80	330.00	471.20
5	1	2019-03-16	10952	471.20	845.80	933.50
6	1	2019-04-09	11011	933.50	471.20	NULL
7	2	2017-09-18	10308	88.80	NULL	479.75
8	2	2018-08-08	10625	479.75	88.80	320.00
9	2	2018-11-28	10759	320.00	479.75	514.40
10	2	2019-03-04	10926	514.40	320.00	NULL
11	3	2017-11-27	10365	403.20	NULL	749.06
12	3	2018-04-15	10507	749.06	403.20	1940.85
13	3	2018-05-13	10535	1940.85	749.06	2082.00
14	3	2018-06-19	10573	2082.00	1940.85	813.37

67. Return, along with each customer's order, the current order value as well as the order values from the customer's first and last orders.

	custid	orderdate	orderid	val	val_firstorder	val_lastorder
1	1	2018-08-25	10643	814.50	814.50	933.50
2	1	2018-10-03	10692	878.00	814.50	933.50
3	1	2018-10-13	10702	330.00	814.50	933.50
4	1	2019-01-15	10835	845.80	814.50	933.50
5	1	2019-03-16	10952	471.20	814.50	933.50
6	1	2019-04-09	11011	933.50	814.50	933.50
7	2	2017-09-18	10308	88.80	88.80	514.40
8	2	2018-08-08	10625	479.75	88.80	514.40
9	2	2018-11-28	10759	320.00	88.80	514.40
10	2	2019-03-04	10926	514.40	88.80	514.40
11	3	2017-11-27	10365	403.20	403.20	660.00
12	3	2018-04-15	10507	749.06	403.20	660.00
13	3	2018-05-13	10535	1940.85	403.20	660.00
14	3	2018-06-19	10573	2082.00	403.20	660.00

68. Return, along with each customer's order, the current order value as well as the difference between the current value and the values of the customer's first and last orders.

	custid	orderdate	orderid	val	difffirst	difflast
1	1	2018-08-25	10643	814.50	0.00	-119.00
2	1	2018-10-03	10692	878.00	63.50	-55.50
3	1	2018-10-13	10702	330.00	-484.50	-603.50
4	1	2019-01-15	10835	845.80	31.30	-87.70
5	1	2019-03-16	10952	471.20	-343.30	-462.30
6	1	2019-04-09	11011	933.50	119.00	0.00
7	2	2017-09-18	10308	88.80	0.00	-425.60
8	2	2018-08-08	10625	479.75	390.95	-34.65
9	2	2018-11-28	10759	320.00	231.20	-194.40
10	2	2019-03-04	10926	514.40	425.60	0.00
11	3	2017-11-27	10365	403.20	0.00	-256.80
12	3	2018-04-15	10507	749.06	345.86	89.06
13	3	2018-05-13	10535	1940.85	1537.65	1280.85
14	3	2018-06-19	10573	2082.00	1678.80	1422.00