

1.

USE AdventureWorks2019

GO

SELECT \* FROM HumanResources.Employee

ORDER BY JobTitle ASC

**Output:**

	BusinessEntityID	NationalIDNumber	LoginID	OrganizationNode	OrganizationLevel	JobTitle	BirthDate	MaritalStatus	Gender	HireDate	SalariedFlag	VacationHours	SickLeaveHours
1	245	36391111	adventure-works\barbara1	0x85B080	3	Accountant	1976-01-06	M	F	2009-02-18	1	58	49
2	248	480951955	adventure-works\mike0	0x85B380	3	Accountant	1979-07-01	S	M	2009-03-08	1	59	49
3	241	30845	adventure-works\david6	0x85A0	2	Accounts Manager	1983-07-08	M	M	2009-01-30	1	57	48
4	246	663843431	adventure-works\dragan0	0x85B180	3	Accounts Payable Specialist	1977-02-14	M	M	2009-02-11	0	63	51
5	247	519756660	adventure-works\janet0	0x85B280	3	Accounts Payable Specialist	1979-03-09	M	F	2009-03-01	0	64	52
6	242	363923697	adventure-works\deborah0	0x85AB	3	Accounts Receivable Specialist	1976-03-06	M	F	2008-12-18	0	60	50
7	243	60517918	adventure-works\candy0	0x85AD	3	Accounts Receivable Specialist	1976-02-23	S	F	2009-01-06	0	61	50
8	244	931190412	adventure-works\bryan1	0x85AF	3	Accounts Receivable Specialist	1984-09-20	S	M	2009-01-24	0	62	51
9	267	58317344	adventure-works\karen1	0x85A0	2	Application Specialist	1978-05-19	S	F	2009-02-16	1	74	57
10	268	31474799	adventure-works\ramesh0	0x8DE0	2	Application Specialist	1988-03-13	S	M	2009-02-03	1	73	56
11	269	671089628	adventure-works\dan0	0x8E10	2	Application Specialist	1987-05-26	M	M	2009-01-11	1	72	56
12	272	525932996	adventure-works\janaina0	0x8E70	2	Application Specialist	1985-01-30	M	F	2008-12-23	1	71	55
13	262	231203233	adventure-works\david5	0x8610	2	Assistant to the Chief Financial Officer	1964-06-21	S	M	2009-01-12	0	56	48
14	239	619308550	adventure-works\mindy0	0x857080	3	Benefits Specialist	1984-11-20	M	F	2008-12-25	0	51	45
15	251	603686790	adventure-works\mikael0	0x85EB88	4	Buyer	1984-08-17	S	M	2009-02-10	0	59	49
16	252	792847334	adventure-works\lind0	0x85EB88	4	Buyer	1974-08-21	M	M	2009-02-28	0	60	50
17	253	407505660	adventure-works\linda2	0x85EB78	4	Buyer	1970-11-30	M	F	2009-12-17	0	56	48
18	254	482810518	adventure-works\lukko0	0x85EB84	4	Buyer	1970-11-24	M	M	2010-01-04	0	57	48
19	255	466142721	adventure-works\gordon0	0x85EBBC	4	Buyer	1966-11-29	M	M	2010-01-11	0	52	46
20	256	367453993	adventure-works\frank2	0x85EB94	4	Buyer	1952-05-12	M	M	2010-01-23	0	58	49
21	257	3811073001	adventure-works\eric2	0x85EB9C	4	Buyer	1972-09-17	S	M	2010-01-27	0	54	47
22	258	785853949	adventure-works\erin0	0x85EB82	4	Buyer	1971-01-04	S	F	2010-01-31	0	53	46
23	259	20269531	adventure-works\ben0	0x85EB86	4	Buyer	1973-06-03	M	M	2010-03-09	0	55	47
24	1	295847284	adventure-works\ken0	NULL	NULL	Chief Executive Officer	1969-01-29	S	M	2009-01-14	1	99	69
25	234	184188301	adventure-works\aura1	0x84	1	Chief Financial Officer	1976-01-06	M	F	2009-01-31	1	0	20
26	221	260805477	adventure-works\chriss1	0x7B5B08	4	Control Specialist	1987-05-26	M	M	2009-03-06	0	75	57
27	218	540688287	adventure-works\engizi2	0x7B5B00	4	Control Specialist	1990-04-28	S	M	2008-12-16	0	76	58
28	270	643805155	adventure-works\francois0	0x8E30	2	Database Administrator	1975-05-17	S	M	2009-01-17	1	67	53
29	271	929666391	adventure-works\dan1	0x8E50	2	Database Administrator	1976-01-06	M	M	2009-01-22	1	66	53
30	5	695256908	adventure-works\gail0	0x8ADA	3	Design Engineer	1952-09-27	M	F	2008-01-06	1	5	22
31	8	688271802	adventure-works\linda1	0x8A81F	2	Design Engineer	1962-11-21	M	M	2008-11-24	1	6	24

Query executed successfully.

LAPTOP-938SNHB2\MSQLSERVER... LAPTOP-938SNHB2\Ongez... AdventureWorks2019 00:00:00 290 rows

2.

USE AdventureWorks2019

GO

SELECT p.\* FROM Person.Person as p

ORDER BY p.LastName

**Output:**

	BusinessEntityID	PersonType	NameStyle	Title	FirstName	MiddleName	LastName	Suffix	EmailPromotion	AdditionalContactInfo	Demographics	rowguid
1	285	SP	0	Mr.	Syed	E	Abbas	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	FF284881-01C2-4C1
2	293	SC	0	Ms.	Catherine	R.	Abel	NULL	1	<AdditionalContactInfo xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	D54E0552-C226-4C	
3	295	SC	0	Ms.	Kim	NULL	Abercrombie	NULL	0	<AdditionalContactInfo xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	F7CB0B48-0B44-47	
4	2170	GC	0	NULL	Kim	NULL	Abercrombie	NULL	2	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	24F01B54-7A67-4B
5	38	EM	0	NULL	Kim	B	Abercrombie	NULL	2	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	9A2163B3-2F4D-4F
6	211	EM	0	NULL	Hazem	E	Abdulros	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	C2637051-25A5-44E
7	2357	GC	0	NULL	Sam	NULL	Abdulros	NULL	1	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	788BF41E-ED14-4E
8	297	SC	0	Sr.	Humberto	NULL	Acevedo	NULL	2	<AdditionalContactInfo xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	5A41D336-8AC1-44	
9	291	SC	0	Mr.	Gustavo	NULL	Achong	NULL	2	<AdditionalContactInfo xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	D4C132D3-FCB5-42	
10	299	SC	0	Sra.	Pilar	NULL	Ackerman	NULL	0	<AdditionalContactInfo xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	DF1FB8AB-2323-43	
11	121	EM	0	NULL	Pilar	G	Ackerman	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	81F50324-D0B5-4E
12	16867	IN	0	NULL	Aaron	B	Adams	NULL	2	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	B46210F5-AC3C-46E
13	16901	IN	0	NULL	Adam	NULL	Adams	NULL	1	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	16C926AD-B2AE-4C
14	16724	IN	0	NULL	Alex	C	Adams	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	AF998B4C-00EF-42
15	10263	IN	0	NULL	Alexandra	J	Adams	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	90A13D3C-DC8C-44
16	10312	IN	0	NULL	Allison	L	Adams	NULL	2	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	3E1A7BC5-E85A-4B
17	10274	IN	0	NULL	Amanda	P	Adams	NULL	1	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	9650B992-E01E-42
18	10292	IN	0	NULL	Amber	NULL	Adams	NULL	1	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	0850CD17-E142-4F
19	10314	IN	0	NULL	Andrea	M	Adams	NULL	1	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	A6BF9164-29F5-41A
20	16699	IN	0	NULL	Angel	NULL	Adams	NULL	1	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	E8BF898D-0493-4D
21	10299	IN	0	NULL	Bailey	NULL	Adams	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	B3252441-E5B0-44F
22	1770	IN	0	Mr.	Ben	NULL	Adams	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	AD322F5C-D052-4D
23	4194	IN	0	NULL	Blake	L	Adams	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	7944C305-C520-411
24	305	SC	0	Ms.	Carla	J.	Adams	NULL	0	<AdditionalContactInfo xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	5F183B92-1648-494	
25	16691	IN	0	NULL	Carlos	NULL	Adams	NULL	1	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	481FEF73-5084-4F
26	4891	IN	0	NULL	Charles	R	Adams	NULL	2	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	FAA29129-5B94-4F8
27	10251	IN	0	NULL	Chloe	A	Adams	NULL	1	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	5C76DEF0-C793-E48
28	16872	IN	0	NULL	Connor	NULL	Adams	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	A3FAE63D-B957-4A
29	10293	IN	0	NULL	Courtney	C	Adams	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	205564F6-8F29-4D
30	4503	IN	0	NULL	Dalton	NULL	Adams	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microsoft.com/office/2006/08/schemas/contacts">	3346FAF4-0685-4B4

Query executed successfully.

LAPTOP-938SNHB2\MSQLSERVER... | LAPTOP-938SNHB2\Ongez... | AdventureWorks2019 | 00:00:02 | 19 972 rows

3.

USE AdventureWorks2019

GO

```
SELECT FirstName, LastName, BusinessEntityID AS Employee_id FROM Person.Person
ORDER BY LastName ASC
```

**Output:**

Results Messages

	FirstName	LastName	Employee_id
1	Syed	Abbas	285
2	Catherine	Abel	293
3	Kim	Abercrombie	295
4	Kim	Abercrombie	2170
5	Kim	Abercrombie	38
6	Hazem	Abolrous	211
7	Sam	Abolrous	2357
8	Humberto	Acevedo	297
9	Gustavo	Achong	291
10	Pilar	Ackerman	299
11	Pilar	Ackerman	121
12	Aaron	Adams	16867
13	Adam	Adams	16901
14	Alex	Adams	16724
15	Alexandra	Adams	10263
16	Allison	Adams	10312
17	Amanda	Adams	10274
18	Amber	Adams	10292
19	Andrea	Adams	10314
20	Angel	Adams	16699
21	Bailey	Adams	10299
22	Ben	Adams	1770
23	Blake	Adams	4194
24	Carla	Adams	305
25	Carlos	Adams	16691
26	Charles	Adams	4891
27	Chloe	Adams	10251
28	Connor	Adams	16872
29	Courtney	Adams	10293
30	Dalton	Adams	4503

✓ Query executed successfully.

4.

USE AdventureWorks2019

GO

```
SELECT ProductID, ProductNumber, Name FROM Production.Product
WHERE SellStartDate IS NOT NULL AND ProductLine = 'T'
ORDER BY Name ASC
```

**Output:**

Results    Messages

	ProductID	ProductNumber	Name
1	890	FR-T98U-46	HL Touring Frame - Blue, 46
2	891	FR-T98U-50	HL Touring Frame - Blue, 50
3	892	FR-T98U-54	HL Touring Frame - Blue, 54
4	893	FR-T98U-60	HL Touring Frame - Blue, 60
5	887	FR-T98Y-46	HL Touring Frame - Yellow, 46
6	888	FR-T98Y-50	HL Touring Frame - Yellow, 50
7	889	FR-T98Y-54	HL Touring Frame - Yellow, 54
8	885	FR-T98Y-60	HL Touring Frame - Yellow, 60
9	947	HB-T928	HL Touring Handlebars
10	916	SE-T924	HL Touring Seat/Saddle
11	903	FR-T67U-44	LL Touring Frame - Blue, 44
12	895	FR-T67U-50	LL Touring Frame - Blue, 50
13	896	FR-T67U-54	LL Touring Frame - Blue, 54
14	897	FR-T67U-58	LL Touring Frame - Blue, 58
15	898	FR-T67U-62	LL Touring Frame - Blue, 62
16	899	FR-T67Y-44	LL Touring Frame - Yellow, 44
17	900	FR-T67Y-50	LL Touring Frame - Yellow, 50
18	901	FR-T67Y-54	LL Touring Frame - Yellow, 54
19	902	FR-T67Y-58	LL Touring Frame - Yellow, 58
20	886	FR-T67Y-62	LL Touring Frame - Yellow, 62
21	946	HB-T721	LL Touring Handlebars
22	914	SE-T312	LL Touring Seat/Saddle
23	912	SE-R908	ML Road Seat/Saddle
24	915	SE-T762	ML Touring Seat/Saddle
25	821	FW-T905	Touring Front Wheel

5.

USE AdventureWorks2019

GO

```
SELECT SalesOrderID, CustomerID, OrderDate, SubTotal, (TaxAmt/SubTotal)*100 AS
tax_percent FROM Sales.SalesOrderHeader
```

```
ORDER BY SubTotal ASC
```

**Output:**

Results Messages

	SalesOrderID	CustomerID	OrderDate	SubTotal	tax_percent
1	51782	30078	2013-06-30 00:00:00.000	1,374	7,99
2	51885	15640	2013-06-30 00:00:00.000	2,29	8,00
3	51886	18293	2013-06-30 00:00:00.000	2,29	8,00
4	52031	19088	2013-07-03 00:00:00.000	2,29	8,00
5	52371	11527	2013-07-10 00:00:00.000	2,29	8,00
6	52532	12370	2013-07-13 00:00:00.000	2,29	8,00
7	52682	11142	2013-07-16 00:00:00.000	2,29	8,00
8	52786	11637	2013-07-18 00:00:00.000	2,29	8,00
9	53015	19090	2013-07-23 00:00:00.000	2,29	8,00
10	53084	28094	2013-07-24 00:00:00.000	2,29	8,00
11	53088	11226	2013-07-24 00:00:00.000	2,29	8,00
12	53198	11321	2013-07-26 00:00:00.000	2,29	8,00
13	53248	19086	2013-07-27 00:00:00.000	2,29	8,00
14	53629	11218	2013-07-31 00:00:00.000	2,29	8,00
15	53674	18733	2013-08-01 00:00:00.000	2,29	8,00
16	53786	11196	2013-08-03 00:00:00.000	2,29	8,00
17	53787	11279	2013-08-03 00:00:00.000	2,29	8,00
18	53884	11521	2013-08-05 00:00:00.000	2,29	8,00
19	53886	11288	2013-08-05 00:00:00.000	2,29	8,00
20	54206	11632	2013-08-11 00:00:00.000	2,29	8,00
21	54596	12036	2013-08-18 00:00:00.000	2,29	8,00
22	54939	18932	2013-08-24 00:00:00.000	2,29	8,00
23	54987	11209	2013-08-25 00:00:00.000	2,29	8,00

✓ Query executed successfully.

6.

USE AdventureWorks2019

GO

SELECT DISTINCT JobTitle FROM HumanResources.Employee

ORDER BY JobTitle ASC

**Output:**

	JobTitle
1	Accountant
2	Accounts Manager
3	Accounts Payable Specialist
4	Accounts Receivable Specialist
5	Application Specialist
6	Assistant to the Chief Financial Officer
7	Benefits Specialist
8	Buyer
9	Chief Executive Officer
10	Chief Financial Officer
11	Control Specialist
12	Database Administrator
13	Design Engineer
14	Document Control Assistant
15	Document Control Manager
16	Engineering Manager
17	European Sales Manager
18	Facilities Administrative Assistant
19	Facilities Manager
20	Finance Manager
21	Human Resources Administrative Assistant
22	Human Resources Manager
23	Information Services Manager

✓ Query executed successfully.

7.

USE AdventureWorks2019

GO

```
SELECT CustomerID, SUM(Freight) AS total_freight FROM Sales.SalesOrderHeader
GROUP BY CustomerID
ORDER BY CustomerID ASC
```

**Output:**

Results    Messages

	CustomerID	total_freight
1	11000	206,2249
2	11001	159,5971
3	11002	202,8511
4	11003	203,4823
5	11004	204,9003
6	11005	203,0334
7	11006	202,9759
8	11007	205,2751
9	11008	202,6578
10	11009	202,2833
11	11010	202,2011
12	11011	203,3261
13	11012	2,0315
14	11013	2,849
15	11014	3,4613
16	11015	62,5243
17	11016	58,307
18	11017	160,8579
19	11018	163,3321
20	11019	22,068
21	11020	57,9243
22	11021	59,299
23	11022	58,057

Query executed successfully.

8.

USE AdventureWorks2019

GO

```
SELECT CustomerID,SalesPersonID, AVG(SubTotal) AS avg_subtotal, SUM(SubTotal) AS sum_subtotal FROM Sales.SalesOrderHeader
```

```
GROUP BY CustomerID, SalesPersonID
```

```
ORDER BY CustomerID DESC
```

**Output:**

Results Messages

	CustomerID	SalesPersonID	avg_subtotal	sum_subtotal
1	30118	275	34638,1521	207828,9131
2	30118	277	35369,8284	70739,6569
3	30117	275	77171,7928	463030,757
4	30117	277	58954,1365	353724,8193
5	30116	276	46778,5502	187114,2011
6	30115	289	1114,6949	8917,5594
7	30114	290	1456,6238	11652,9911
8	30113	282	34148,2363	273185,8908
9	30112	280	93591,6217	93591,6217
10	30112	284	54909,3785	384365,6498
11	30111	277	69131,6007	276526,4031
12	30110	276	406,3188	1625,2755
13	30109	275	59857,2605	119714,521
14	30109	277	36114,5232	216687,1397
15	30108	289	36150,6454	144602,5816
16	30107	276	48090,7976	577089,5723
17	30106	275	6172,4534	12344,9068
18	30106	277	14809,7794	88858,6768
19	30105	280	26523,0027	26523,0027
20	30105	284	20957,8988	146705,2917
21	30104	282	1549,0503	4647,151
22	30104	287	968,4284	968,4284
23	30103	290	80468,2369	643745,8958

✔ Query executed successfully.

9.

USE AdventureWorks2019

GO

```
SELECT ProductID, SUM(Quantity) AS total_quantity FROM
Production.ProductInventory WHERE Shelf IN ('A','C','H')
GROUP BY ProductID HAVING SUM(Quantity) >500
ORDER BY ProductID ASC
```

**Output:**

	ProductID	total_quantity
1	1	761
2	2	791
3	3	909
4	4	900
5	316	532
6	317	593
7	319	797
8	320	1136
9	321	1750
10	322	1684
11	323	1684
12	324	1629
13	325	1210
14	326	1097
15	328	1044
16	329	1025
17	330	1005
18	331	831
19	350	719
20	355	546
21	356	518
22	367	643
23	371	585

✓ Query executed successfully.

10.

USE AdventureWorks2019

GO

```
SELECT SUM(Quantity) *10 AS total_quantity FROM Production.ProductInventory  
GROUP BY LocationID
```

**Output:**

Results Messages

	total_quantity
1	1860
2	831730
3	173190
4	728990
5	954770
6	9580
7	135840
8	1100
9	202950
10	3320
11	55490
12	204190
13	5080
14	51650

Query executed successfully.

11.

USE AdventureWorks2019

GO

```
SELECT p.BusinessEntityID, p.FirstName, p.LastName, ph.PhoneNumber AS
person_phone FROM Person.Person AS p
LEFT JOIN Person.PersonPhone AS ph ON p.BusinessEntityID = ph.BusinessEntityID
WHERE p.LastName LIKE 'L%'
ORDER BY p.LastName ASC, p.FirstName ASC
```

**Output:**

Results Messages

	BusinessEntityID	FirstName	LastName	person_phone
1	5527	Aaron	Lal	605-555-0159
2	5268	Adam	Lal	513-555-0110
3	12539	Alejandro	Lal	1 (11) 500 555-0117
4	19786	Alicia	Lal	1 (11) 500 555-0161
5	12004	Alisha	Lal	1 (11) 500 555-0119
6	16649	Alison	Lal	1 (11) 500 555-0177
7	5005	Alvin	Lal	1 (11) 500 555-0168
8	5070	Andres	Lal	1 (11) 500 555-0127
9	10416	Arturo	Lal	638-555-0164
10	8951	Ashlee	Lal	1 (11) 500 555-0148
11	6283	Austin	Lal	541-555-0141
12	11600	Barbara	Lal	1 (11) 500 555-0176
13	6744	Benjamin	Lal	1 (11) 500 555-0148
14	17275	Bethany	Lal	1 (11) 500 555-0196
15	3694	Bonnie	Lal	1 (11) 500 555-0191
16	9390	Brad	Lal	463-555-0111
17	20292	Bradley	Lal	1 (11) 500 555-0124
18	6943	Brandon	Lal	445-555-0135
19	9340	Brendan	Lal	660-555-0111
20	5581	Caleb	Lal	392-555-0153
21	6334	Cameron	Lal	906-555-0115
22	16953	Carl	Lal	1 (11) 500 555-0125
23	4496	Carly	Lal	1 (11) 500 555-0150
24	16290	Casey	Lal	1 (11) 500 555-0190
25	8890	Cassie	Lal	1 (11) 500 555-0148
26	20334	Chad	Lal	1 (11) 500 555-0147
27	6232	Christian	Lal	423-555-0171

12.

USE AdventureWorks2019

GO

SELECT SalesPersonID, CustomerID, SUM(SubTotal) AS sum\_subtotal FROM Sales.SalesOrderHeader

WHERE SalesPersonID IS NOT NULL

GROUP BY ROLLUP (SalesPersonID, CustomerID)

ORDER BY SalesPersonID, CustomerID

**Output:**

Results    Messages

	SalesPersonID	CustomerID	sum_subtotal
1	NULL	NULL	80487704,1832
2	274	NULL	1092123,8561
3	274	29491	33406,7043
4	274	29493	2146,962
5	274	29514	3405,1668
6	274	29523	34349,2656
7	274	29576	53,994
8	274	29579	35331,66
9	274	29604	647,994
10	274	29605	29482,0603
11	274	29616	138046,3212
12	274	29617	198993,3507
13	274	29623	1946,022
14	274	29650	83,988
15	274	29666	15842,6141
16	274	29669	3962,2441
17	274	29671	11802,564
18	274	29675	4254,45
19	274	29680	2709,6518
20	274	29691	209,256
21	274	29707	79994,1743
22	274	29716	61206,4782
23	274	29719	971,982
24	274	29722	125422,2078
25	274	29755	12206,364

✔ Query executed successfully.

13.

USE AdventureWorks2019

GO

SELECT LocationID, Shelf, SUM(Quantity) AS TotalQuantity FROM  
Production.ProductInventory

GROUP BY ROLLUP (LocationID, Shelf)

ORDER BY LocationID, Shelf

**Output:**

Results Messages

	LocationID	Shelf	TotalQuantity
1	NULL	NULL	335974
2	1	NULL	72899
3	1	A	2727
4	1	C	13777
5	1	D	6551
6	1	E	8032
7	1	F	7614
8	1	G	3954
9	1	H	10905
10	1	J	5051
11	1	K	6751
12	1	L	7537
13	2	NULL	5549
14	2	B	900
15	2	C	1557
16	2	D	3092
17	3	NULL	186
18	3	A	186
19	4	NULL	110
20	4	A	110
21	5	NULL	20295
22	5	A	6572
23	5	B	1281
24	5	D	1215
25	5	E	1279

Query executed successfully.

14.

USE AdventureWorks2019

GO

```
SELECT LocationID, Shelf, SUM(Quantity) AS TotalQuantity FROM
Production.ProductInventory
```

```
GROUP BY ROLLUP (LocationID, Shelf)
```

```
ORDER BY LocationID, Shelf
```

**Output:**

Results | Messages

	LocationID	Shelf	TotalQuantity
1	NULL	NULL	335974
2	1	NULL	72899
3	1	A	2727
4	1	C	13777
5	1	D	6551
6	1	E	8032
7	1	F	7614
8	1	G	3954
9	1	H	10905
10	1	J	5051
11	1	K	6751
12	1	L	7537
13	2	NULL	5549
14	2	B	900
15	2	C	1557
16	2	D	3092
17	3	NULL	186
18	3	A	186
19	4	NULL	110
20	4	A	110

15.

USE AdventureWorks2019

GO

```
SELECT LocationID, SUM(Quantity) AS TotalQuantity FROM  
Production.ProductInventory
```

```
GROUP BY ROLLUP (LocationID)
```

```
ORDER BY LocationID
```

**Output:**

Results Messages

	LocationID	TotalQuantity
1	NULL	335974
2	1	72899
3	2	5549
4	3	186
5	4	110
6	5	20295
7	6	83173
8	7	17319
9	10	13584
10	20	5165
11	30	958
12	40	508
13	45	332
14	50	95477
15	60	20419

Query executed successfully.

16.

USE AdventureWorks2019

GO

SELECT CITY, COUNT(\*) AS noofemployees FROM Person.Address

GROUP BY City

ORDER BY City ASC

**Output:**

Results    Messages

	CITY	noofemployees
1	Abingdon	1
2	Albany	4
3	Alexandria	2
4	Alhambra	1
5	Alpine	1
6	Altadena	2
7	Altamonte Springs	1
8	Anacortes	3
9	Arlington	1
10	Ascheim	1
11	Atlanta	2
12	Auburn	1
13	Augsburg	2
14	Augusta	1
15	Aujan Mournede	1
16	Aurora	1
17	Austell	1
18	Austin	2
19	Bad Soden	1
20	Baldwin Park	1
21	Ballard	69
22	Baltimore	1
23	Barrie	1
24	Barstow	2
25	Basingstoke Hants	3

Query executed successfully.

17.

USE AdventureWorks2019

GO

```
SELECT YEAR(OrderDate) AS YearOfOrderDate, SUM(TotalDue) AS TotalDueOrder FROM
Sales.SalesOrderHeader
```

```
GROUP BY YEAR(OrderDate)
```

```
ORDER BY YEAR(OrderDate) ASC
```

**Output:**

Results Messages

	YearOfOrderDate	TotalDueOrder
1	2011	14155699,525
2	2012	37675700,312
3	2013	48965887,9632
4	2014	22419498,3157

18.

USE AdventureWorks2019

GO

```
SELECT YEAR(OrderDate) AS YearOfOrderDate, SUM(TotalDue) AS TotalDueOrder FROM
Sales.SalesOrderHeader

WHERE YEAR(OrderDate) <= 2016

GROUP BY YEAR(OrderDate)

ORDER BY YEAR(OrderDate) ASC
```

**Output:**

Results Messages

	YearOfOrderDate	TotalDueOrder
1	2011	14155699,525
2	2012	37675700,312
3	2013	48965887,9632
4	2014	22419498,3157

19.

USE AdventureWorks2019

GO

```
SELECT ContactTypeID, Name FROM Person.ContactType

WHERE Name LIKE '%Manager%'

ORDER BY ContactTypeID DESC
```

**Output:**

The screenshot shows a SQL Server Management Studio window with two tabs: 'Results' and 'Messages'. The 'Results' tab is selected and displays a table with six rows of data. The columns are labeled 'ContactTypeID' and 'Name'. The data is as follows:

	ContactTypeID	Name
1	19	Sales Manager
2	15	Purchasing Manager
3	13	Product Manager
4	8	Marketing Manager
5	6	International Marketing Manager
6	1	Accounting Manager

20.

USE AdventureWorks2019

GO

```
SELECT p.BusinessEntityID, p.LastName, p.FirstName FROM Person.Person AS p
JOIN Person.BusinessEntityContact AS bec ON p.BusinessEntityID = bec.PersonID
JOIN Person.ContactType AS ct ON bec.ContactTypeID = ct.ContactTypeID
WHERE ct.Name = 'Purchasing Manager'
ORDER BY p.LastName ASC, p.FirstName ASC
```

**Output:**

Results Messages

	BusinessEntityID	LastName	FirstName
1	1149	Alexander	Mary
2	363	Arakawa	Hannah
3	365	Arbelaez	Kyley
4	377	Ault	John
5	379	Avalos	Robert
6	389	Bailey	James
7	391	Baldwin	Douglas
8	399	Banks	Darrell
9	401	Barbariol	Angela
10	403	Barber	David
11	409	Barlow	Brenda
12	411	Barnhill	Josh
13	413	Barr	Adam
14	423	Bauer	Ciro
15	425	Beanston	Glenna
16	427	Beasley	Shaun
17	447	Ben-Sachar	Ido
18	449	Benson	Edna
19	453	Benson	Max
20	451	Benson	Payton
21	455	Bent	Scot
22	457	Bentley	Richard
23	465	Berger	John
24	467	Bergin	Kris
25	469	Berqlund	Andreas

Query executed successfully.

21.

USE AdventureWorks2019

GO

```
SELECT ROW_NUMBER() OVER (PARTITION BY a.PostalCode ORDER BY sp.SalesYTD
DESC) AS RowNum, p.LastName, sp.SalesYTD, a.PostalCode FROM Sales.SalesPerson
sp
```

```
INNER JOIN Person.Person p ON sp.BusinessEntityID = p.BusinessEntityID
```

```
INNER JOIN Person.Address a ON p.BusinessEntityID = a.AddressID
```

```
INNER JOIN Sales.SalesTerritory st ON sp.TerritoryID = st.TerritoryID
```

```
WHERE sp.SalesYTD <> 0 AND a.PostalCode IS NOT NULL
```

```
ORDER BY a.PostalCode ASC, sp.SalesYTD DESC
```

**Output:**

	RowNum	LastName	SalesYTD	PostalCode
1	1	Mitchell	4251368,5497	98027
2	2	Blythe	3763178,1787	98027
3	3	Carson	3189418,3662	98027
4	4	Reiter	2315185,611	98027
5	5	Vargas	1453719,4653	98027
6	6	Ansman-Wolfe	1352577,1325	98027
7	1	Pak	4116871,2277	98055
8	2	Varkey Chudukatil	3121616,3202	98055
9	3	Saraiva	2604540,7172	98055
10	4	Ito	2458535,6169	98055
11	5	Valdez	1827066,7118	98055
12	6	Mensa-Annan	1576562,1966	98055
13	7	Campbell	1573012,9383	98055
14	8	Tsoflias	1421810,9242	98055

22.

USE AdventureWorks2019

GO

```
SELECT ct.ContactTypeID, ct.Name AS ContactTypeName, COUNT(*) AS NoContacts
FROM Person.ContactType AS ct
JOIN Person.BusinessEntityContact AS bec ON ct.ContactTypeID = bec.ContactTypeID
GROUP BY ct.ContactTypeID, ct.Name HAVING COUNT(*) >= 100
ORDER BY NoContacts DESC
```

**Output:**

	ContactTypeID	ContactTypeName	NoContacts
1	11	Owner	266
2	15	Purchasing Manager	245
3	14	Purchasing Agent	242

23.

USE AdventureWorks2019

GO

```
SELECT CONVERT(date, eph.RateChangeDate) AS fromdate, p.LastName + ', ' +  
p.FirstName + COALESCE(' ' + LEFT(p.MiddleName, 1), '') AS nameinfull,
```

```
CAST(eph.Rate * 40 AS decimal(18,4)) AS salaryinaweek FROM  
HumanResources.EmployeePayHistory AS eph
```

```
INNER JOIN HumanResources.Employee AS e ON e.BusinessEntityID =  
eph.BusinessEntityID
```

```
INNER JOIN Person.Person AS p ON p.BusinessEntityID = e.BusinessEntityID
```

```
ORDER BY nameinfull ASC
```

**Output:**

Results    Messages

	fromdate	nameinfull	salaryinaweek
1	2013-03-14	Abbas, Syed E	1924.0400
2	2010-01-16	Abercrombie, Kim B	498.0000
3	2009-02-28	Abolrous, Hazem E	1153.8480
4	2009-01-02	Ackerman, Pilar G	769.2320
5	2009-03-05	Adams, Jay G	498.0000
6	2009-01-17	Ajenstat, François P	1538.4600
7	2012-04-16	Alberts, Amy E	1924.0400
8	2008-12-02	Alderson, Greg F	400.0000
9	2008-12-28	Alexander, Sean P	423.0760
10	2009-12-02	Altman, Gary E	961.5400
11	2009-01-02	Anderson, Nancy A	498.0000
12	2011-05-31	Ansman-Wolfe, Pamela O	923.0760
13	2009-01-04	Arifin, Zainal T	711.5400
14	2009-01-11	Bacon, Dan K	1096.1520
15	2009-01-21	Baker, Bryan	498.0000
16	2009-12-25	Baker, Mary R	538.0000
17	2009-01-20	Barbariol, Angela W	440.0000
18	2009-01-12	Barber, David M	538.4600
19	2008-12-06	Barreto de Mattos, Paula M	1085.5760
20	2011-01-07	Benshoof, Wanida M	538.4600
21	2009-02-16	Berg, Karen A	1096.1520
22	2009-02-09	Berge, Karen R	410.0000
23	2009-02-02	Berglund, Andreas T	423.0760
24	2009-01-20	Berndt, Matthias T	380.0000
25	2010-03-07	Berry, Jo L	370.0000

Query executed successfully.

24.

USE AdventureWorks2019

GO

```
SELECT CONVERT(date, eph.RateChangeDate) AS fromdate, p.LastName + ' ' +
p.FirstName + COALESCE(' ' + LEFT(p.MiddleName, 1), '') AS nameinfull,
CAST(eph.Rate * 40 AS decimal(18,4)) AS salaryinaweek FROM
HumanResources.EmployeePayHistory AS eph
INNER JOIN HumanResources.Employee AS e ON e.BusinessEntityID =
eph.BusinessEntityID
INNER JOIN Person.Person AS p ON p.BusinessEntityID = e.BusinessEntityID
```

```

INNER JOIN (SELECT BusinessEntityID, MAX(RateChangeDate) AS LatestChange FROM
HumanResources.EmployeePayHistory

GROUP BY BusinessEntityID

) AS latest ON latest.BusinessEntityID = eph.BusinessEntityID AND latest.LatestChange
= eph.RateChangeDate

ORDER BY nameinfull ASC

```

**Output:**

	fromdate	nameinfull	salaryinaweek
1	2013-03-14	Abbas, Syed E	1924.0400
2	2010-01-16	Abercrombie, Kim B	498.0000
3	2009-02-28	Abolrous, Hazem E	1153.8480
4	2009-01-02	Ackerman, Pilar G	769.2320
5	2009-03-05	Adams, Jay G	498.0000
6	2009-01-17	Ajenstat, François P	1538.4600
7	2012-04-16	Alberts, Amy E	1924.0400
8	2008-12-02	Alderson, Greg F	400.0000
9	2008-12-28	Alexander, Sean P	423.0760
10	2009-12-02	Altman, Gary E	961.5400
11	2009-01-02	Anderson, Nancy A	498.0000
12	2011-05-31	Ansman-Wolfe, Pamela O	923.0760
13	2009-01-04	Arifin, Zainal T	711.5400
14	2009-01-11	Bacon, Dan K	1096.1520
15	2009-01-21	Baker, Bryan	498.0000
16	2009-12-25	Baker, Mary R	538.0000
17	2009-01-20	Barbariol, Angela W	440.0000
18	2009-01-12	Barber, David M	538.4600
19	2008-12-06	Barreto de Mattos, Paula M	1085.5760
20	2011-01-07	Benshoof, Wanida M	538.4600
21	2009-02-16	Berg, Karen A	1096.1520
22	2009-02-09	Berge, Karen R	410.0000
23	2009-02-02	Berglund, Andreas T	423.0760
24	2009-01-20	Berndt, Matthias T	380.0000
25	2010-03-07	Berry, Jo L	370.0000

Query executed successfully.

25.

USE AdventureWorks2019

GO

```

SELECT d.SalesOrderID, d.ProductID, d.OrderQty, agg.TotalQty AS [Total Quantity],
agg.AvgQty AS [Avg Quantity], agg.OrderCount AS [No of Orders],
agg.MinQty AS [Min Quantity], agg.MaxQty AS [Max Quantity] FROM
Sales.SalesOrderDetail AS d

CROSS JOIN (SELECT SUM(d2.OrderQty) AS TotalQty, AVG(CAST(d2.OrderQty AS
decimal(18,6))) AS AvgQty, COUNT(*) AS OrderCount, MIN(d2.OrderQty) AS MinQty,
MAX(d2.OrderQty) AS MaxQty FROM Sales.SalesOrderDetail AS d2

WHERE d2.SalesOrderID IN (43659, 43664)) AS agg

WHERE d.SalesOrderID IN (43659, 43664)

ORDER BY d.SalesOrderID, d.ProductID

```

**Output:**

	SalesOrderID	ProductID	OrderQty	Total Quantity	Avg Quantity	No of Orders	Min Quantity	Max Quantity
1	43659	709	6	40	2.000000	20	1	6
2	43659	711	4	40	2.000000	20	1	6
3	43659	712	2	40	2.000000	20	1	6
4	43659	714	3	40	2.000000	20	1	6
5	43659	716	1	40	2.000000	20	1	6
6	43659	771	1	40	2.000000	20	1	6
7	43659	772	1	40	2.000000	20	1	6
8	43659	773	2	40	2.000000	20	1	6
9	43659	774	1	40	2.000000	20	1	6
10	43659	776	1	40	2.000000	20	1	6
11	43659	777	3	40	2.000000	20	1	6
12	43659	778	1	40	2.000000	20	1	6
13	43664	714	1	40	2.000000	20	1	6
14	43664	716	1	40	2.000000	20	1	6
15	43664	771	3	40	2.000000	20	1	6
16	43664	772	1	40	2.000000	20	1	6
17	43664	773	1	40	2.000000	20	1	6
18	43664	775	4	40	2.000000	20	1	6
19	43664	777	2	40	2.000000	20	1	6
20	43664	778	1	40	2.000000	20	1	6

Query executed successfully.

LAPTOP-938S

26

USE AdventureWorks2019

GO

```

SELECT SalesOrderID AS OrderNumber, ProductID, OrderQty AS Quantity,
SUM(OrderQty) OVER (ORDER BY SalesOrderID, ProductID) AS Total,
AVG(OrderQty) OVER (ORDER BY SalesOrderID, ProductID) AS Avg, COUNT(OrderQty)
OVER (ORDER BY SalesOrderID, ProductID) AS Count FROM Sales.SalesOrderDetail

```

```
WHERE SalesOrderID IN (43659, 43664) AND ProductID LIKE '71%'  
ORDER BY SalesOrderID, ProductID
```

**Output:**

	OrderNumber	ProductID	Quantity	Total	Avg	Count
1	43659	711	4	4	4	1
2	43659	712	2	6	3	2
3	43659	714	3	9	3	3
4	43659	716	1	10	2	4
5	43664	714	1	11	2	5
6	43664	716	1	12	2	6

27

```
USE AdventureWorks2019
```

```
GO
```

```
SELECT SalesOrderID, SUM(OrderQty * UnitPrice) AS TotalCost FROM  
Sales.SalesOrderDetail
```

```
GROUP BY SalesOrderID HAVING SUM(OrderQty * UnitPrice) > 100000
```

**Output:**

	SalesOrderID	TotalCost
1	43875	122744.1667
2	43884	116248.4897
3	44518	127099.9957
4	44528	108783.5872
5	44530	104960.0101
6	44795	104588.713
7	46066	100378.9078
8	46067	101857.213
9	46607	121037.4456
10	46616	153432.0611
11	46643	110830.3608
12	46645	101373.1246
13	46660	119415.1966
14	46981	149533.7021
15	47018	107991.0123
16	47027	105375.2251
17	47255	120195.6757

Query executed successfully.

28

USE AdventureWorks2019

GO

```
SELECT ProductID, Name FROM Production.Product
```

```
WHERE Name LIKE 'Lock Washer%'
```

```
ORDER BY ProductID ASC
```

**Output:**

The screenshot shows the SQL Server Management Studio interface with the 'Results' tab selected. The results grid displays 13 rows of data from the Production.Product table, filtered by name starting with 'Lock Washer'. The columns are 'ProductID' and 'Name'. The first row, where ProductID is 463 and Name is 'Lock Washer 4', is highlighted with a blue border.

	ProductID	Name
1	463	Lock Washer 4
2	464	Lock Washer 5
3	465	Lock Washer 10
4	466	Lock Washer 6
5	467	Lock Washer 13
6	468	Lock Washer 8
7	469	Lock Washer 1
8	470	Lock Washer 7
9	471	Lock Washer 12
10	472	Lock Washer 2
11	473	Lock Washer 9
12	474	Lock Washer 3
13	475	Lock Washer 11

29

USE AdventureWorks2019

GO

```
SELECT ProductID, Name, Color FROM Production.Product
```

```
ORDER BY ListPrice
```

**Output:**

Results Messages

	ProductID	Name	Color
1	1	Adjustable Race	NULL
2	2	Bearing Ball	NULL
3	3	BB Ball Bearing	NULL
4	4	Headset Ball Bearings	NULL
5	316	Blade	NULL
6	317	LL Crankarm	Black
7	318	ML Crankarm	Black
8	319	HL Crankarm	Black
9	320	Chainring Bolts	Silver
10	321	Chainring Nut	Silver
11	322	Chainring	Black
12	323	Crown Race	NULL
13	324	Chain Stays	NULL
14	325	Decal 1	NULL
15	326	Decal 2	NULL
16	327	Down Tube	NULL
17	229	Mountain End Caps	NULL

✓ Query executed successfully.

30

USE AdventureWorks2019

GO

```
SELECT BusinessEntityID, JobTitle, HireDate FROM HumanResources.Employee  
ORDER BY YEAR(HireDate)
```

**Output:**

Results Messages

	BusinessEntityID	JobTitle	HireDate
1	28	Production Technician - WC60	2006-06-30
2	40	Production Supervisor - WC60	2007-12-26
3	3	Engineering Manager	2007-11-11
4	4	Senior Tool Designer	2007-12-05
5	12	Tool Designer	2007-12-11
6	16	Marketing Manager	2007-12-20
7	17	Marketing Assistant	2007-01-26
8	5	Design Engineer	2008-01-06
9	6	Design Engineer	2008-01-24
10	8	Research and Development Engineer	2008-12-29
11	22	Marketing Specialist	2008-12-12
12	2	Vice President of Engineering	2008-01-31
13	26	Production Control Manager	2008-12-01
14	27	Production Supervisor - WC60	2008-02-27
15	32	Production Technician - WC60	2008-12-29
16	33	Production Technician - WC60	2008-12-17
17	42	Production Technician - WC60	2008-12-27

✔ Query executed successfully.

31

USE AdventureWorks2019

GO

SELECT LastName, FirstName FROM Person.Person

WHERE LastName LIKE 'R%'

ORDER BY FirstName ASC, LastName DESC

**Output:**

Results Messages

	LastName	FirstName
1	Russell	Aaron
2	Ross	Aaron
3	Roberts	Aaron
4	Rana	Abby
5	Raman	Abby
6	Russell	Abigail
7	Ross	Abigail
8	Ross	Abigail
9	Rogers	Abigail
10	Rodriguez	Abigail
11	Robinson	Abigail
12	Rivera	Abigail
13	Richardson	Abigail
14	Reed	Abigail
15	Ramirez	Abigail
16	Russell	Adam
17	Ross	Adam

Query executed successfully.

32

USE AdventureWorks2019

GO

SELECT BusinessEntityID, SalariedFlag FROM HumanResources.Employee

ORDER BY

CASE WHEN SalariedFlag = 1 THEN 0 ELSE 1 END,

CASE WHEN SalariedFlag = 1 THEN BusinessEntityID END DESC,

CASE WHEN SalariedFlag = 0 THEN BusinessEntityID END ASC

**Output:**

Results    Messages

	BusinessEntityID	SalariedFlag
1	290	1
2	289	1
3	288	1
4	287	1
5	286	1
6	285	1
7	284	1
8	283	1
9	282	1
10	281	1
11	280	1
12	279	1
13	278	1
14	277	1
15	276	1
16	275	1
17	274	1

Query executed successfully.

33

USE AdventureWorks2019

GO

```

SELECT sp.BusinessEntityID AS businessentityid, p.LastName AS lastname, st.Name
AS territoryname, cr.Name AS countryregionname FROM Sales.SalesPerson AS sp
JOIN Person.Person AS p ON p.BusinessEntityID = sp.BusinessEntityID
LEFT JOIN Sales.SalesTerritory AS st ON st.TerritoryID = sp.TerritoryID
LEFT JOIN Person.CountryRegion AS cr ON cr.CountryRegionCode =
st.CountryRegionCode
ORDER BY
CASE WHEN cr.Name = 'United States' THEN 1 ELSE 0 END,
CASE WHEN cr.Name = 'United States' THEN st.Name END ASC,
CASE WHEN cr.Name <> 'United States' THEN cr.Name END ASC

```

**Output:**

Results    Messages

	businessentityid	lastname	territoryname	countryregionname
1	274	Jiang	NULL	NULL
2	285	Abbas	NULL	NULL
3	287	Alberts	NULL	NULL
4	286	Tsoflias	Australia	Australia
5	278	Vargas	Canada	Canada
6	282	Saraiva	Canada	Canada
7	290	Varkey Chudukatil	France	France
8	288	Valdez	Germany	Germany
9	289	Pak	United Kingdom	United Kingdom
10	277	Carson	Central	United States
11	275	Blythe	Northeast	United States
12	283	Campbell	Northwest	United States
13	284	Mensa-Annan	Northwest	United States
14	280	Ansman-Wolfe	Northwest	United States
15	279	Reiter	Southeast	United States
16	276	Mitchell	Southwest	United States
17	291	Ito	Southwest	United States

Query executed successfully.

34

USE AdventureWorks2019

GO

```

SELECT p.FirstName, p.LastName, ROW_NUMBER() OVER (ORDER BY a.PostalCode) AS [Row Number], RANK() OVER (ORDER BY a.PostalCode) AS [Rank], DENSE_RANK()
OVER (ORDER BY a.PostalCode) AS [Dense Rank],  

NTILE(4) OVER (ORDER BY a.PostalCode) AS [Quartile], e.SalesYTD, a.PostalCode
FROM Sales.SalesPerson AS e  

JOIN HumanResources.Employee AS hr ON e.BusinessEntityID = hr.BusinessEntityID
JOIN Person.Person AS p ON e.BusinessEntityID = p.BusinessEntityID
JOIN Person.Address AS a ON e.BusinessEntityID = a.AddressID
WHERE e.TerritoryID IS NOT NULL AND e.SalesYTD <> 0
ORDER BY a.PostalCode

```

**Output:**

Results Messages

	FirstName	LastName	Row Number	Rank	Dense Rank	Quartile	SalesYTD	PostalCode
1	Michael	Blythe	1	1	1	1	3763178,1787	98027
2	Linda	Mitchell	2	1	1	1	4251368,5497	98027
3	Jillian	Carson	3	1	1	1	3189418,3662	98027
4	Garrett	Vargas	4	1	1	1	1453719,4653	98027
5	Tsvi	Reiter	5	1	1	2	2315185,611	98027
6	Pamela	Anzman-Wolfe	6	1	1	2	1352577,1325	98027
7	Shu	Ito	7	7	2	2	2458535,6169	98055
8	José	Saraiva	8	7	2	2	2604540,7172	98055
9	David	Campbell	9	7	2	3	1573012,9383	98055
10	Tete	Mensa-Annan	10	7	2	3	1576562,1966	98055
11	Lynn	Tsoflias	11	7	2	3	1421810,9242	98055
12	Rachel	Valdez	12	7	2	4	1827066,7118	98055
13	Jae	Pak	13	7	2	4	4116871,2277	98055
14	Ranjit	Varkey Chudukatil	14	7	2	4	3121616,3202	98055

Query executed successfully.

35

USE AdventureWorks2019

GO

SELECT DepartmentID, Name, GroupName FROM HumanResources.Department

ORDER BY DepartmentID

OFFSET 10 ROWS

**Output:**

Results Messages

	DepartmentID	Name	GroupName
1	11	Information Services	Executive General and Administration
2	Click to select all grid cells	Control	Quality Assurance
3	13	Quality Assurance	Quality Assurance
4	14	Facilities and Maintenance	Executive General and Administration
5	15	Shipping and Receiving	Inventory Management
6	16	Executive	Executive General and Administration

36

USE AdventureWorks2019

GO

SELECT DepartmentID, Name, GroupName FROM HumanResources.Department

ORDER BY DepartmentID

OFFSET 5 ROWS

FETCH NEXT 5 ROWS ONLY

**Output:**

The screenshot shows a results grid from a SQL query. The grid has three columns: DepartmentID, Name, and GroupName. The DepartmentID column is sorted in descending order. The first five rows are highlighted with a dotted border, indicating they are the result of the 'FETCH NEXT 5 ROWS ONLY' clause. The last five rows are not highlighted.

	DepartmentID	Name	GroupName
1	6	Research and Development	Research and Development
2	7	Production	Manufacturing
3	8	Production Control	Manufacturing
4	9	Human Resources	Executive General and Administration
5	10	Finance	Executive General and Administration

37

USE AdventureWorks2019

GO

SELECT Name, Color, ListPrice FROM Production.Product

WHERE Color IN ('Red', 'Blue')

ORDER BY ListPrice

**Output:**

	Name	Color	ListPrice
1	Sport-100 Helmet, Red	Red	34,99
2	Sport-100 Helmet, Blue	Blue	34,99
3	Classic Vest, S	Blue	63,50
4	Classic Vest, M	Blue	63,50
5	Classic Vest, L	Blue	63,50
6	LL Touring Frame - Blue, 50	Blue	333,42
7	LL Touring Frame - Blue, 54	Blue	333,42
8	LL Touring Frame - Blue, 58	Blue	333,42
9	LL Touring Frame - Blue, 62	Blue	333,42
10	LL Touring Frame - Blue, 44	Blue	333,42
11	LL Road Frame - Red, 44	Red	337,22
12	LL Road Frame - Red, 48	Red	337,22
13	LL Road Frame - Red, 52	Red	337,22
14	LL Road Frame - Red, 58	Red	337,22
15	LL Road Frame - Red, 60	Red	337,22
16	LL Road Frame - Red, 62	Red	337,22
17	ML Road Frame - Red, 44	Red	504,92

✓ Query executed successfully.

38

USE AdventureWorks2019

GO

```
SELECT p.Name AS ProductName, sod.SalesOrderID FROM Production.Product AS p
FULL OUTER JOIN Sales.SalesOrderDetail AS sod ON p.ProductID = sod.ProductID
ORDER BY p.Name
```

**Output:**

Results | Messages

	ProductName	SalesOrderID
1	Adjustable Race	NULL
2	All-Purpose Bike Stand	51179
3	All-Purpose Bike Stand	51488
4	All-Purpose Bike Stand	51520
5	All-Purpose Bike Stand	51558
6	All-Purpose Bike Stand	51882
7	All-Purpose Bike Stand	51903
8	All-Purpose Bike Stand	51970
9	All-Purpose Bike Stand	52010
10	All-Purpose Bike Stand	52032
11	All-Purpose Bike Stand	52095
12	All-Purpose Bike Stand	52309
13	All-Purpose Bike Stand	52357
14	All-Purpose Bike Stand	52372
15	All-Purpose Bike Stand	52380
16	All-Purpose Bike Stand	52417
17	All-Purpose Bike Stand	52559

Query executed successfully.

39

USE AdventureWorks2019

GO

```
SELECT p.Name, soh.SalesOrderID FROM Production.Product AS p
LEFT JOIN Sales.SalesOrderDetail AS sod ON p.ProductID = sod.ProductID
LEFT JOIN Sales.SalesOrderHeader AS soh ON sod.SalesOrderID = soh.SalesOrderID
ORDER BY p.Name
```

**Output:**

Results Messages

	Name	SalesOrderID
1	Adjustable Race	NULL
2	All-Purpose Bike Stand	52357
3	All-Purpose Bike Stand	53295
4	All-Purpose Bike Stand	53997
5	All-Purpose Bike Stand	54397
6	All-Purpose Bike Stand	54482
7	All-Purpose Bike Stand	54597
8	All-Purpose Bike Stand	54780
9	All-Purpose Bike Stand	55484
10	All-Purpose Bike Stand	55533
11	All-Purpose Bike Stand	55984
12	All-Purpose Bike Stand	57473
13	All-Purpose Bike Stand	58492
14	All-Purpose Bike Stand	59094
15	All-Purpose Bike Stand	60332
16	All-Purpose Bike Stand	60679
17	All-Purpose Bike Stand	60912

Query executed successfully.

40

USE AdventureWorks2019

GO

```
SELECT p.Name, soh.SalesOrderID FROM Production.Product AS p
JOIN Sales.SalesOrderDetail AS sod ON sod.ProductID = p.ProductID
JOIN Sales.SalesOrderHeader AS soh ON soh.SalesOrderID = sod.SalesOrderID
ORDER BY p.Name ASC
```

**Output:**

Results Messages

	name	salesorderid
1	All-Purpose Bike Stand	51903
2	All-Purpose Bike Stand	52010
3	All-Purpose Bike Stand	53046
4	All-Purpose Bike Stand	54286
5	All-Purpose Bike Stand	55629
6	All-Purpose Bike Stand	56347
7	All-Purpose Bike Stand	56447
8	All-Purpose Bike Stand	57876
9	All-Purpose Bike Stand	58505
10	All-Purpose Bike Stand	59916
11	All-Purpose Bike Stand	59991
12	All-Purpose Bike Stand	60377
13	All-Purpose Bike Stand	61145
14	All-Purpose Bike Stand	61570
15	All-Purpose Bike Stand	62924
16	All-Purpose Bike Stand	64464
17	All Purpose Bike Stand	66600

✓ Query executed successfully.

41

USE AdventureWorks2019

GO

```
SELECT st.Name AS Territory, sp.BusinessEntityID FROM Sales.SalesPerson AS sp
```

```
LEFT JOIN Sales.SalesTerritory AS st ON sp.TerritoryID = st.TerritoryID
```

**Output:**

Results Messages

	TerritoryName	BusinessEntityID
1	NULL	274
2	Northeast	275
3	Southwest	276
4	Central	277
5	Canada	278
6	Southeast	279
7	Northwest	280
8	Southwest	281
9	Canada	282
10	Northwest	283
11	Northwest	284
12	NULL	285
13	Australia	286
14	NULL	287
15	Germany	288
16	United Kingdom	289
17	France	290

Query executed successfully.

42

USE AdventureWorks2019

GO

```
SELECT p.FirstName + ' ' + p.LastName AS name, a.City FROM
HumanResources.Employee AS e
JOIN Person.Person AS p ON e.BusinessEntityID = p.BusinessEntityID
JOIN Person.BusinessEntityAddress AS bea ON e.BusinessEntityID =
bea.BusinessEntityID
JOIN Person.Address AS a ON bea.AddressID = a.AddressID
ORDER BY p.LastName, p.FirstName
```

**Output:**

Results Messages

	name	City
1	Syed Abbas	Bothell
2	Kim Abercrombie	Carnation
3	Hazem Abolrous	Kenmore
4	Pilar Ackerman	Seattle
5	Jay Adams	Monroe
6	François Ajenstat	Issaquah
7	Amy Alberts	Renton
8	Greg Alderson	Bellevue
9	Sean Alexander	Renton
10	Gary Altman	Renton
11	Nancy Anderson	Sammamish
12	Pamela Ansman-Wolfe	Portland
13	Zainal Arifin	Issaquah
14	Dan Bacon	Issaquah
15	Bryan Baker	Monroe
16	Mary Baker	Seattle
17	Angela Barbario	Edmonds

Query executed successfully.

43

USE AdventureWorks2019

GO

SELECT d.BusinessEntityID, d.FirstName, d.LastName FROM

(

SELECT BusinessEntityID, FirstName, LastName, PersonType FROM Person.Person

) AS d

WHERE d.PersonType = 'IN' AND d.LastName = 'Adams'

ORDER BY d.FirstName ASC

**Output:**

Results Messages

	BusinessEntityID	FirstName	LastName
1	16867	Aaron	Adams
2	16901	Adam	Adams
3	16724	Alex	Adams
4	10263	Alexandra	Adams
5	10312	Allison	Adams
6	10274	Amanda	Adams
7	10292	Amber	Adams
8	10314	Andrea	Adams
9	16699	Angel	Adams
10	10299	Bailey	Adams
11	1770	Ben	Adams
12	4194	Blake	Adams
13	16691	Carlos	Adams
14	4891	Charles	Adams
15	10251	Chloe	Adams
16	16872	Connor	Adams
17	10202	Courtney	Adams

Query executed successfully.

44

USE AdventureWorks2019

GO

```
SELECT BusinessEntityID, FirstName, LastName FROM Person.Person
WHERE BusinessEntityID <= 1500 AND LastName LIKE 'Al%' AND FirstName LIKE 'M%'
ORDER BY BusinessEntityID
```

**Output:**

Results Messages

	BusinessEntityID	FirstName	LastName
1	327	Milton	Albury
2	335	Michelle	Alexander
3	341	Marvin	Allen
4	343	Michael	Allen
5	1149	Mary	Alexander

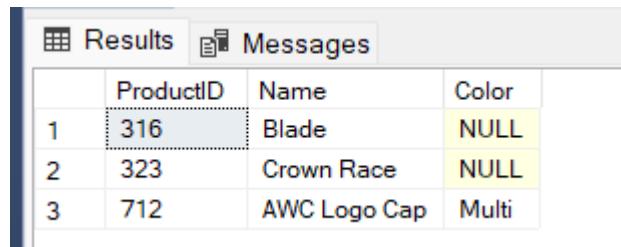
45

USE AdventureWorks2019

GO

```
SELECT p.ProductID, p.Name, p.Color FROM Production.Product AS p
JOIN
(
    SELECT 'Blade' AS Name
UNION ALL
    SELECT 'Crown Race'
UNION ALL
    SELECT 'AWC Logo Cap'
) AS t ON p.Name = t.Name
```

**Output:**



The screenshot shows the SQL Server Management Studio interface with the 'Results' tab selected. The results window displays a table with three rows. The columns are labeled 'ProductID', 'Name', and 'Color'. The data is as follows:

	ProductID	Name	Color
1	316	Blade	NULL
2	323	Crown Race	NULL
3	712	AWC Logo Cap	Multi

46

```
USE AdventureWorks2019
```

```
GO
```

```
SELECT SalesPersonID, COUNT(SalesOrderID) AS TotalSales, YEAR(OrderDate) AS
SalesYear FROM Sales.SalesOrderHeader
```

```
WHERE SalesPersonID IS NOT NULL
```

```
GROUP BY YEAR(OrderDate), SalesPersonID
```

```
ORDER BY SalesPersonID, YEAR(OrderDate)
```

**Output:**

Results Messages

	SalesPersonID	TotalSales	SalesYear
1	274	4	2011
2	274	22	2012
3	274	14	2013
4	274	8	2014
5	275	65	2011
6	275	148	2012
7	275	175	2013
8	275	62	2014
9	276	46	2011
10	276	151	2012
11	276	162	2013
12	276	59	2014
13	277	59	2011
14	277	166	2012
15	277	185	2013
16	277	63	2014
17	279	20	2011

Query executed successfully.

47

USE AdventureWorks2019

GO

```
SELECT AVG(OrderCount) AS [Average Sales Per Person] FROM
(
    SELECT sp.BusinessEntityID, COUNT(soh.SalesOrderID) AS OrderCount FROM
        Sales.SalesPerson AS sp
        JOIN Sales.SalesOrderHeader AS soh ON sp.BusinessEntityID = soh.SalesPersonID
        GROUP BY sp.BusinessEntityID
) AS SalesPerPerson
```

**Output:**

Results Messages

	Average Sales Per Person
1	223

48

USE AdventureWorks2019

GO

SELECT \* FROM Production.ProductPhoto

WHERE LargePhotoFileName LIKE '%green\_%'

**Output:**

	ProductPhotoID	ThumbnailPhoto	ThumbnailPhotoFileName	LargePhoto	LargePhotoFileName	ModifiedDate
1	76	0x47494638396150003100F700007698B3B1D5A475777BECE...	racer02_green_f_small.gif	0x474946333961F0009500F70000CACBF8DE4EFA3D19EECE...	racer02_green_f_large.gif	2012-10-19 09:56:38.273
2	77	0x47494638396150003100F700008BD3C4E2E3F397A5A9455...	racer02_green_small.gif	0x474946333961F0009500F7000444549309407CACAF8D6E4...	racer02_green_large.gif	2012-10-19 09:56:38.273
3	91	0x47494638396150003100F7000000000000000000000000800...	julianax_r_02_f_green_small.gif	0x474946333961F0009500F7000000020500810040C101018...	julianax_r_02_f_green_large.gif	2012-10-19 09:56:38.273
4	95	0x47494638396150003100F700000000000000000000000800...	julianax_r_02_green_small.gif	0x474946333961F0009500F700000002020080800001000081...	julianax_r_02_green_large.gif	2012-10-19 09:56:38.273
5	105	0x47494638396150003100F70000200050D0512151012181...	roadster_green_f_small.gif	0x474946333961F0009500F700000000000000000000000800...	roadster_green_f_large.gif	2012-10-19 09:56:38.273
6	106	0x47494638396150003100F70000200050B0810100018120...	roadster_green_small.gif	0x474946333961F0009500F700000000000000000000000800...	roadster_green_large.gif	2012-10-19 09:56:38.273
7	123	0x47494638396150003100F700001008101810182110292110...	superlight_metalicgreen_f_small.gif	0x474946333961F0009500F70000150E11B1221221E262921...	superlight_metalicgreen_f_large.gif	2012-10-19 09:56:38.273
8	124	0x47494638396150003100F7000010081018101818182118...	superlight_metalicgreen_small.gif	0x474946333961F0009500F70000150E131C1821221D252921...	superlight_metalicgreen_large.gif	2012-10-19 09:56:38.273

49

USE AdventureWorks2019

GO

SELECT a.AddressLine1, a.AddressLine2, a.City, a.PostalCode, cr.CountryRegionCode  
FROM Person.Address AS a

JOIN Person.StateProvince AS sp ON a.StateProvinceID = sp.StateProvinceID

JOIN Person.CountryRegion AS cr ON sp.CountryRegionCode = cr.CountryRegionCode

WHERE cr.CountryRegionCode <> 'US' AND a.City LIKE 'Pa%'

**Output:**

Results Messages

	AddressLine1	AddressLine2	City	PostalCode	CountryRegionCode
1	1, avenue des Champs-Elysées	NULL	Paris	75017	FR
2	1, place Beaubernard	NULL	Paris	75003	FR
3	1, place Beaubernard	NULL	Paris	75009	FR
4	1, place de la République	NULL	Paris	75017	FR
5	1, rue de Fontfroide	NULL	Paris	75003	FR
6	1, rue de la Cavalerie	NULL	Paris La Défense	92081	FR
7	1, rue de Maubeuge	NULL	Paris	75007	FR
8	1, rue de Maubeuge	NULL	Paris	75009	FR
9	1, rue Georges-Clémenceau	NULL	Paris	75003	FR
10	1, rue Pierre-Demoulin	NULL	Paris	75007	FR
11	1, rue Pierre-Demoulin	NULL	Paris	75012	FR
12	10, avenue de Norvege	NULL	Paris	75005	FR
13	10, avenue du Président-Kennedy	NULL	Paris	75019	FR
14	10, route de Marseille	NULL	Pantin	93500	FR
15	10, rue Lafayette	NULL	Paris	75007	FR
16	10, rue Philibert-Delorme	NULL	Paris	75007	FR
17	100, boulevard d'Albi	NULL	Paris	75006	FR

✓ Query executed successfully.

50

USE AdventureWorks2019

GO

SELECT TOP 20 JobTitle, HireDate FROM HumanResources.Employee

ORDER BY HireDate DESC

**Output:**

Results Messages

	JobTitle	HireDate
1	Sales Representative	2013-05-30
2	Sales Representative	2013-05-30
3	Pacific Sales Manager	2013-03-14
4	Sales Representative	2012-09-30
5	Sales Representative	2012-05-30
6	Sales Representative	2012-05-30
7	European Sales Manager	2012-04-16
8	Sales Representative	2011-05-31
9	Sales Representative	2011-05-31
10	Sales Representative	2011-05-31
11	Sales Representative	2011-05-31
12	Sales Representative	2011-05-31
13	Sales Representative	2011-05-31
14	Sales Representative	2011-05-31
15	Sales Representative	2011-05-31
16	Sales Representative	2011-05-31
17	Purchasing Manager	2011-02-25

✓ Query executed successfully.