```sql
---- AVG ----
-- 1. Obtener el promedio de precios por cada categoría de producto. La cláusula
-- OVER(PARTITION BY CategoryID) específica que se debe calcular el promedio de
-- precios por cada valor único de CategoryID en la tabla.

select
    c.category_name ,
    p.product_name ,
    avg(p.unit_price) over (partition by p.category_id) avg_prices
from
    products p
inner join categories c
on
    p.category_id = c.category_id;
```

categories(+) 1 ×

Enter a SQL expression to filter results (use Ctrl+Space)

	category_name	product_name	avg_prices
1	Beverages	Guaraná Fantástica	37.9791666667
2	Beverages	Ipoh Coffee	37.9791666667
3	Beverages	Chartreuse verte	37.9791666667
4	Beverages	Côte de Blaye	37.9791666667
5	Beverages	Steeleye Stout	37.9791666667
6	Beverages	Sasquatch Ale	37.9791666667
7	Beverages	Lakkalikööri	37.9791666667
8	Beverages	Rhönbräu Klosterbier	37.9791666667
9	Beverages	Outback Lager	37.9791666667
10	Beverages	Chai	37.9791666667
11	Beverages	Laughing Lumberjack Lager	37.9791666667
12	Beverages	Chang	37.9791666667
13	Condiments	Gula Malacca	22.8541668256
14	Condiments	Original Frankfurter grüne Soße	22.8541668256
15	Condiments	Northwoods Cranberry Sauce	22.8541668256
16	Condiments	Louisiana Hot Spiced Okra	22.8541668256
17	Condiments	Genen Shouyu	22.8541668256

```sql
-- 2. Obtener el promedio de venta de cada cliente

select
    c.customer_id ,
    c. company_name ,
    avg(od.unit_price * od.quantity) over (partition by o.customer_id) as avgsales
from
    customers c ,
    order_details od
inner join orders o on
    o.order_id = od.order_id
where
    c.customer_id = o.customer_id;
```

customers 1 ×

Enter a SQL expression to filter results (use Ctrl+Space)

	customer_id	company_name	avgsales
1	ALFKI	Alfreds Futterkiste	383.0166670481
2	ALFKI	Alfreds Futterkiste	383.0166670481
3	ALFKI	Alfreds Futterkiste	383.0166670481
4	ALFKI	Alfreds Futterkiste	383.0166670481
5	ALFKI	Alfreds Futterkiste	383.0166670481
6	ALFKI	Alfreds Futterkiste	383.0166670481
7	ALFKI	Alfreds Futterkiste	383.0166670481
8	ALFKI	Alfreds Futterkiste	383.0166670481
9	ALFKI	Alfreds Futterkiste	383.0166670481
10	ALFKI	Alfreds Futterkiste	383.0166670481
11	ALFKI	Alfreds Futterkiste	383.0166670481
12	ALFKI	Alfreds Futterkiste	383.0166670481
13	ANATR	Ana Trujillo Emparedados y helados	140.2949990273
14	ANATR	Ana Trujillo Emparedados y helados	140.2949990273
15	ANATR	Ana Trujillo Emparedados y helados	140.2949990273
16	ANATR	Ana Trujillo Emparedados y helados	140.2949990273
17	ANATR	Ana Trujillo Emparedados y helados	140.2949990273

```sql
-- 3. Obtener el promedio de cantidad de productos vendidos por categoría (product_n
-- quantity_per_unit, unit_price, quantity, avgquantity) y ordenarlo por nombre de l
-- categoría y nombre del producto

select
    p.product_name ,
    c.category_name ,
    p.quantity_per_unit ,
    od.unit_price ,
    od.quantity ,
    avg(od.quantity) over (partition by c.category_id) as avgqty
from
    categories c ,
    order_details od
inner join products p on
    od.product_id = p.product_id
where
    c.category_id = p.category_id
order by
    c.category_name ,
    p.product_name;
```

products(+) 1

Enter a SQL expression to filter results (use Ctrl+Space)

	product_name	category_name	quantity_per_unit	unit_price	quantity	avgqty
1	Chai	Beverages	10 boxes x 30 bags	14.4	10	23.5940594059
2	Chai	Beverages	10 boxes x 30 bags	18	25	23.5940594059
3	Chai	Beverages	10 boxes x 30 bags	18	21	23.5940594059
4	Chai	Beverages	10 boxes x 30 bags	18	60	23.5940594059
5	Chai	Beverages	10 boxes x 30 bags	18	20	23.5940594059
6	Chai	Beverages	10 boxes x 30 bags	18	4	23.5940594059
7	Chai	Beverages	10 boxes x 30 bags	18	10	23.5940594059

```sql
---- MIN ----
-- 4. Selecciona el ID del cliente, la fecha de la orden y la fecha más antigua de l
-- orden para cada cliente de la tabla 'Orders'.

select
    o.customer_id ,
    o.order_date ,
    min(o.order_date) over (partition by o.customer_id)
from
    orders o;
```

orders 1

Enter a SQL expression to filter results (use Ctrl+Space)

	customer_id	order_date	min
1	ALFKI	1998-01-15	1997-08-25
2	ALFKI	1997-10-03	1997-08-25
3	ALFKI	1998-04-09	1997-08-25
4	ALFKI	1997-10-13	1997-08-25
5	ALFKI	1997-08-25	1997-08-25
6	ALFKI	1998-03-16	1997-08-25
7	ANATR	1997-08-08	1996-09-18
8	ANATR	1998-03-04	1996-09-18
9	ANATR	1996-09-18	1996-09-18
10	ANATR	1997-11-28	1996-09-18
11	ANTON	1997-09-22	1996-11-27
12	ANTON	1997-05-13	1996-11-27
13	ANTON	1998-01-28	199
14	ANTON	1997-09-25	199
15	ANTON	1997-04-15	1996-11-27
16	ANTON	1997-06-19	1996-11-27
17	ANTON	1996-11-27	1996-11-27
18	AROUT	1997-02-21	1996-11-15
19	AROUT	1997-11-17	1996-11-15
20	AROUT	1996-12-16	1996-11-15
21	AROUT	1998-03-16	1996-11-15

Show attribute filters

```sql
---- MAX ----
 -- 5. Seleccione el id de producto, el nombre de producto, el precio unitario, el id
 -- categoría y el precio unitario máximo para cada categoría de la tabla Products

select
    p.product_id ,
    p.product_name ,
    p.unit_price ,
    p.category_id ,
    max(p.unit_price) over (partition by p.category_id)
 from
    products p;
```

products 1

Enter a SQL expression to filter results (use Ctrl+Space)

	product_id	product_name	unit_price	category_id	max
1	24	Guaraná Fantástica	4.5	1	263.5
2	43	Ipoh Coffee	46	1	263.5
3	39	Chartreuse verte	18	1	263.5
4	38	Côte de Blaye	263.5	1	263.5
5	35	Steeleye Stout	18	1	263.5
6	34	Sasquatch Ale	14	1	263.5
7	76	Lakkalikööri	18	1	263.5
8	75	Rhönbräu Klosterbier	7.75	1	263.5
9	70	Outback Lager	15	1	263.5
10	1	Chai	18	1	263.5
11	67	Laughing Lumberjack Lager	14	1	263.5
12	2	Chang	19	1	263.5
13	44	Gula Malacca	19.45	2	43.9
14	77	Original Frankfurter grüne Soße	13	2	43.9
15	8	Northwoods Cranberry Sauce	40	2	43.9
16	66	Louisiana Hot Spiced Okra	17	2	43.9
17	15	Genen Shouyu	13	2	43.9
18	6	Grandma's Boysenberry Spread	25	2	43.9

```sql
 -- 6. Obtener el ranking de los productos más vendidos.

select
    od.product_id ,
    p.product_name ,
    sum(od.quantity) qty_sum ,
    row_number() over (
    order by sum(od.quantity)desc) as ranking
 from
    order_details od
 join products p
 on
    od.product_id = p.product_id
 group by
    od.product_id ,
    p.product_name;
```

order_details(+) 1

Enter a SQL expression to filter results (use Ctrl+Space)

	product_id	product_name	qty_sum	ranking
1	60	Camembert Pierrot	1,577	1
2	59	Raclette Courdavault	1,496	2
3	31	Gorgonzola Telino	1,397	3
4	56	Gnocchi di nonna Alice	1,263	4
5	16	Pavlova	1,158	5
6	75	Rhönbräu Klosterbier	1,155	6
7	24	Guaraná Fantástica	1,125	7
8	40	Boston Crab Meat	1,103	8
9	62	Tarte au sucre	1,083	9
10	71	Flotemysost	1,057	10
11	2	Chang	1,057	11
12	21	Sir Rodney's Scones	1,016	12
13	41	Jack's New England Clam Chowder	981	13

Show attribute filters

```sql
-- 7. Asignar numeros de fila para cada cliente, ordenados por customer_id

select
    row_number () over (
    order by c.customer_id),
    c.*
 from
    customers c;
```

customers 1 ×

select row_number () over (order by c.customer Enter a SQL expression to filter results (use Ctrl+Space)

	row_number	customer_id	company_name	contact_name	contact_title	address
1	1	ALFKI	Alfreds Futterkiste	Maria Anders	Sales Representative	Obere Str. 57
2	2	ANATR	Ana Trujillo Emparedados y helados	Ana Trujillo	Owner	Avda. de la Constitució
3	3	ANTON	Antonio Moreno Taquería	Antonio Moreno	Owner	Mataderos 2312
4	4	AROUT	Around the Horn	Thomas Hardy	Sales Representative	120 Hanover Sq.
5	5	BERGS	Berglunds snabbköp	Christina Berglund	Order Administrator	Berguvsvägen 8
6	6	BLAUS	Blauer See Delikatessen	Hanna Moos	Sales Representative	Forsterstr. 57
7	7	BLONP	Blondesddsl père et fils	Frédérique Citeaux	Marketing Manager	24, place Kléber
8	8	BOLID	Bólido Comidas preparadas	Martín Sommer	Owner	C/ Araquil, 67
9	9	BONAP	Bon app'	Laurence Lebihan	Owner	12, rue des Bouchers
10	10	BOTTM	Bottom-Dollar Markets	Elizabeth Lincoln	Accounting Manager	23 Tsawassen Blvd.
11	11	BSBEV	B's Beverages	Victoria Ashworth	Sales Representative	Fauntleroy Circus
12	12	CACTU	Cactus Comidas para llevar	Patricio Simpson	Sales Agent	Cerrito 333
13	13	CENTC	Centro comercial Moctezuma	Francisco Chang	Marketing Manager	Sierras de Granada 99
14	14	CHOPS	Chop-suey Chinese	Yang Wang	Owner	Hauptstr. 29
15	15	COMMI	Comércio Mineiro	Pedro Afonso	Sales Associate	Av. dos Lusíadas, 23
16	16	CONSH	Consolidated Holdings	Elizabeth Brown	Sales Representative	Berkeley Gardens 12
17	17	DRACD	Drachenblut Delikatessen	Sven Ottlieb	Order Administrator	Walserweg 21
18	18	DUMON	Du monde entier	Janine Labrune	Owner	67, rue des Cinquante
19	19	EASTC	Eastern Connection	Ann Devon	Sales Agent	35 King George
20	20	ERNSH	Ernst Handel	Roland Mendel	Sales Manager	Kirchgasse 6
21	21	FAMIA	Familia Arquibaldo	Aria Cruz	Marketing Assistant	Rua Orós, 92
22	22	FISSA	FISSA Fabrica Inter. Salchichas S.A.	Diego Roel	Accounting Manager	C/ Moralzarzal, 86

```sql
-- 8. Obtener el ranking de los empleados más jóvenes () ranking, nombre y apellido
 -- empleado, fecha de nacimiento)

select
    row_number () over (
    order by e.birth_date desc) ,
    concat(e.first_name ,
    ' ' ,
    e.last_name) employees ,
    e.birth_date
 from
    employees e;
```

employees 1 ×

select row_number () over (order by e.birth_date Enter a SQL expression to filter results (use Ctrl+Space)

	row_number	employees	birth_date
1	1	Anne Dodsworth	1966-01-27
2	2	Janet Leverling	1963-08-30
3	3	Michael Suyama	1963-07-02
4	4	Robert King	1960-05-29
5	5	Laura Callahan	1958-01-09
6	6	Steven Buchanan	1955-03-04
7	7	Andrew Fuller	1952-02-19
8	8	Nancy Davolio	1948-12-08
9	9	Margaret Peacock	1937-09-19

```sql
---- SUM
 -- 9. Obtener la suma de venta de cada cliente

select
    sum(od.unit_price * od.quantity) over (partition by o.customer_id),
    od.order_id ,
    c.customer_id ,
    c.company_name ,
    o.*
 from
    customers c ,
    order_details od
 inner join orders o
 on
    o.order_id = od.order_id
 where
    c.customer_id = o.customer_id;
```

order_details(+) 1

Enter a SQL expression to filter results (use Ctrl+Space)

	sum	order_id	customer_id	company_name	order_id	customer_id	employee
1	4,596.2000045776	10,702	ALFKI	Alfreds Futterkiste	10,702	ALFKI	
2	4,596.2000045776	10,643	ALFKI	Alfreds Futterkiste	10,643	ALFKI	
3	4,596.2000045776	10,952	ALFKI	Alfreds Futterkiste	10,952	ALFKI	
4	4,596.2000045776	11,011	ALFKI	Alfreds Futterkiste	11,011	ALFKI	
5	4,596.2000045776	11,011	ALFKI	Alfreds Futterkiste	11,011	ALFKI	
6	4,596.2000045776	10,692	ALFKI	Alfreds Futterkiste	10,692	ALFKI	
7	4,596.2000045776	10,643	ALFKI	Alfreds Futterkiste	10,643	ALFKI	
8	4,596.2000045776	10,643	ALFKI	Alfreds Futterkiste	10,643	ALFKI	
9	4,596.2000045776	10,835	ALFKI	Alfreds Futterkiste	10,835	ALFKI	
10	4,596.2000045776	10,952	ALFKI	Alfreds Futterkiste	10,952	ALFKI	
11	4,596.2000045776	10,702	ALFKI	Alfreds Futterkiste	10,702	ALFKI	
12	4,596.2000045776	10,835	ALFKI	Alfreds Futterkiste	10,835	ALFKI	
13	1,402.9499902725	10,308	ANATR	Ana Trujillo Emparedados y helados	10,308	ANATR	

```sql
 -- 10. Obtener la suma total de ventas por categoría de producto

select
    c.category_name ,
    sum(od.unit_price * od.quantity) over (partition by p.category_id),
    p.product_name ,
    od.unit_price ,
    od.quantity
 from
    categories c ,
    order_details od
 inner join products p
 on
    od.product_id = p.product_id
 where
    c.category_id = p.category_id;
-- 11. Calcular la suma total de gastos de envío por país de destino,
```

categories(+) 1

Enter a SQL expression to filter results (use Ctrl+Space)

	category_name	sum	product_name	unit_price	quantity
1	Beverages	286,526.9500956535	Rhönbräu Klosterbier	7.75	20
2	Beverages	286,526.9500956535	Chartreuse verte	18	5
3	Beverages	286,526.9500956535	Sasquatch Ale	11.2	35
4	Beverages	286,526.9500956535	Sasquatch Ale	11.2	10
5	Beverages	286,526.9500956535	Guaraná Fantástica	4.5	8
6	Beverages	286,526.9500956535	Outback Lager	15	50
7	Beverages	286,526.9500956535	Lakkalikööri	18	10
8	Beverages	286,526.9500956535	Côte de Blaye	210.8	10
9	Beverages	286,526.9500956535	Chang	15.2	35
10	Beverages	286,526.9500956535	Chartreuse verte	14.4	54
11	Beverages	286,526.9500956535	Outback Lager	15	3
12	Beverages	286,526.9500956535	Steeleye Stout	18	3
13	Beverages	286,526.9500956535	Chai	18	40

```sql
-- 11. Calcular la suma total de gastos de envío por país de destino,
-- luego ordenarlo por país y por orden de manera ascendente
-- freight es flete, costo economico del transporte.

select
    o.ship_country,
    o.order_id ,
    sum(o.freight) over (partition by o.ship_country)
from
    orders o
order by
    ship_country ,
    o.order_id;
```

orders 1 ×

```
select o.ship_country, o.order_id , sum(o.freight  Enter a SQL expression to filter results (use Ctrl+Space)
```

	ship_country	order_id	sum
1	Argentina		
2	Argentina	10,448	598.58
3	Argentina	10,521	598.58
4	Argentina	10,531	598.58
5	Argentina	10,716	598.58
6	Argentina	10,782	598.58
7	Argentina	10,819	598.58
8	Argentina	10,828	598.58
9	Argentina	10,881	598.58
10	Argentina	10,898	598.58
11	Argentina	10,916	598.58
12	Argentina	10,937	598.58
13	Argentina	10,958	598.58
14	Argentina	10,986	598.58
15	Argentina	11,019	598.58
16	Argentina	11,054	598.58
17	Austria	10,258	7,391.501

Ctrl+click to open SQL console

```sql
-- 12. Ranking de ventas por cliente

select
    o.customer_id,
    sum(od.quantity * od.unit_price) sales ,
    rank() over (
    order by sum(od.quantity * od.unit_price) desc)
from
    order_details od
inner join orders o
on
    o.order_id = od.order_id
group by
    o.customer_id;
```

orders 1 ×

```
select o.customer_id, sum(od.quantity * od.unit  Enter a SQL expression to filter results (use Ctrl+Space)
```

	customer_id	sales	rank
1	QUICK	117,483.390147686	1
2	SAVEA	115,673.3896427154	2
3	ERNSH	113,236.6797819138	3
4	HUNGO	57,317.390162468	4
5	RATTC	52,245.900346756	5
6	HANAR	34,101.1499738693	6
7	FOLKO	32,555.5500192642	7
8	MEREP	32,203.9002342224	8
9	KOENE	31,745.7498931885	9
10	QUEEN	30,226.1001796722	10
11	WHITC	29,073.44992733	11
12	FRANK	28,722.709939003	12
13	BERGS	26,968.1499304771	13
14	PICCO	26,259.9500846863	14
15	SUPRD	24,704.4003038406	15
16	BONAP	23,850.949985981	16

```sql
-- 13. Ranking de empleados por fecha de contratacion

select
    rank() over (
    order by e.hire_date) hire_date_rank,
    e.*
 from
    employees e;
```

employees 1

Enter a SQL expression to filter results (use Ctrl+Space)

	hire_date_rank	employee_id	last_name	first_name	title	title_of_courtesy	birth_da
1	1	3	Leverling	Janet	Sales Representative	Ms.	196
2	2	1	Davolio	Nancy	Sales Representative	Ms.	194
3	3	2	Fuller	Andrew	Vice President, Sales	Dr.	195
4	4	4	Peacock	Margaret	Sales Representative	Mrs.	193
5	5	5	Buchanan	Steven	Sales Manager	Mr.	195
6	5	6	Suyama	Michael	Sales Representative	Mr.	196
7	7	7	King	Robert	Sales Representative	Mr.	196
8	8	8	Callahan	Laura	Inside Sales Coordinator	Ms.	195
9	9	9	Dodsworth	Anne	Sales Representative	Ms.	196

```sql
-- 14. Ranking de productos por precio unitario

select
    p.product_name ,
    p.product_id ,
    p.unit_price ,
    rank() over (
    order by unit_price desc) as rank
 from
    products p;
```

products 1

Enter a SQL expression to filter results (use Ctrl+Space)

	product_name	product_id	unit_price	rank
1	Côte de Blaye	38	263.5	1
2	Thüringer Rostbratwurst	29	123.79	2
3	Mishi Kobe Niku	9	97	3
4	Sir Rodney's Marmalade	20	81	4
5	Carnarvon Tigers	18	62.5	5
6	Raclette Courdavault	59	55	6
7	Manjimup Dried Apples	51	53	7
8	Tarte au sucre	62	49.3	8
9	Ipoh Coffee	43	46	9
10	Rössle Sauerkraut	28	45.6	10
11	Schoggi Schokolade	27	43.9	11
12	Vegie-spread	63	43.9	11
13	Northwoods Cranberry Sauce	8	40	13
14	Alice Mutton	17	39	14
15	Gnocchi di nonna Alice	56	38	15
16	Queso Manchego La Pastora	12	38	15
17	Gudbrandsdalsost	69	36	17
18	Mozzarella di Giovanni	72	34.8	18
19	Camembert Pierrot	60	34	19
20	Wimmers gute Semmelknödel	64	33.25	20
21	Perth Pasties	53	32.8	21

```sql
-- 15. Mostrar por cada producto de una orden, la cantidad vendida y la cantidad
-- vendida del producto previo.

select
    od.order_id ,
    od.product_id ,
    od.quantity ,
    lag(od.quantity) over (
    order by od.order_id asc) as previous_qty
from
    order_details od;
```

order_details 1

```
select od.order_id , od.product_id , od.quantity
```

Enter a SQL expression to filter results (use Ctrl+Space)

	order_id	product_id	quantity	previous_qty
1	10,248	11	12	[NULL]
2	10,248	42	10	12
3	10,248	72	5	10
4	10,249	14	9	5
5	10,249	51	40	9
6	10,250	41	10	40
7	10,250	51	35	10
8	10,250	65	15	35
9	10,251	22	6	15
10	10,251	57	15	6
11	10,251	65	20	15
12	10,252	20	40	20
13	10,252	33	25	40
14	10,252	60	40	25
15	10,253	31	20	40
16	10,253	39	42	20
17	10,253	49	40	42
18	10,254	24	15	40
19	10,254	55	21	15
20	10,254	74	21	21

```sql
-- 16. Obtener un listado de ordenes mostrando el id de la orden, fecha de orden, id
-- y última fecha de orden.

select
    o.order_id ,
    o.order_date ,
    o.customer_id ,
    lag(o.order_date) over (
    order by o.customer_id ,
    o.order_date  asc)
from
    orders o;
```

orders 1

```
select o.order_id , o.order_date , o.customer_id
```

Enter a SQL expression to filter results (use Ctrl+Space)

	order_id	order_date	customer_id	lag
1	10,643	1997-08-25	ALFKI	[NULL]
2	10,692	1997-10-03	ALFKI	1997-08-25
3	10,702	1997-10-13	ALFKI	1997-10-03
4	10,835	1998-01-15	ALFKI	1997-10-13
5	10,952	1998-03-16	ALFKI	1998-01-15
6	11,011	1998-04-09	ALFKI	1998-03-16
7	10,308	1996-09-18	ANATR	1998-04-09
8	10,625	1997-08-08	ANATR	1996-09-18
9	10,759	1997-11-28	ANATR	1997-08-08
10	10,926	1998-03-04	ANATR	1997-11-28
11	10,365	1996-11-27	ANTON	1998-03-04
12	10,507	1997-04-15	ANTON	1996-11-27
13	10,535	1997-05-13	ANTON	1997-04-15
14	10,573	1997-06-19	ANTON	1997-05-13
15	10,677	1997-09-22	ANTON	1997-06-19
16	10,682	1997-09-25	ANTON	1997-09-22
17	10,856	1998-01-28	ANTON	1997-09-25
18	10,355	1996-11-15	AROUT	1998-01-28

```sql
-- 17. Obtener un listado de productos que contengan: id de producto, nombre del pro
-- precio unitario, precio del producto anterior, diferencia entre el precio del pro
-- precio del producto anterior.

select
    p.product_name ,
    p.product_id ,
    p.unit_price ,
    lag(p.unit_price) over (
    order by p.product_id) previous_price,
    (p.unit_price - lag(p.unit_price) over (
    order by p.product_id) ) diff_prices
from
    products p
```

products 1

Enter a SQL expression to filter results (use Ctrl+Space)

	product_name	product_id	unit_price	previous_price	diff_prices
1	Chai	1	18	[NULL]	[NULL]
2	Chang	2	19	18	1
3	Aniseed Syrup	3	10	19	-9
4	Chef Anton's Cajun Seasoning	4	22	10	12
5	Chef Anton's Gumbo Mix	5	21.35	22	-0.6499996
6	Grandma's Boysenberry Spread	6	25	21.35	3.6499996
7	Uncle Bob's Organic Dried Pears	7	30	25	5
8	Northwoods Cranberry Sauce	8	40	30	10
9	Mishi Kobe Niku	9	97	40	57
10	Ikura	10	31	97	-66
11	Queso Cabrales	11	21	31	-10
12	Queso Manchego La Pastora	12	38	21	17
13	Konbu	13	6	38	-32
14	Tofu	14	23.25	6	17.25
15	Genen Shouyu	15	13	23.25	-10.25
16	Pavlova	16	17.45	13	4.450001
17	Alice Mutton	17	39	17.45	21.55

```sql
---- LEAD ----
-- 18 Obtener un listado que muestra el precio de un producto junto con el precio de
-- siguiente:

select
    p.product_name ,
    p.unit_price ,
    lead (p.unit_price) over (
    order by p.product_id asc)
from
    products p
```

products 1

Enter a SQL expression to filter results (use Ctrl+Space)

	product_name	unit_price	lead
1	Chai	18	19
2	Chang	19	10
3	Aniseed Syrup	10	22
4	Chef Anton's Cajun Seasoning	22	21.35
5	Chef Anton's Gumbo Mix	21.35	25
6	Grandma's Boysenberry Spread	25	30
7	Uncle Bob's Organic Dried Pears	30	40
8	Northwoods Cranberry Sauce	40	97
9	Mishi Kobe Niku	97	31
10	Ikura	31	21
11	Queso Cabrales	21	38
12	Queso Manchego La Pastora	38	6
13	Konbu	6	23.25
14	Tofu	23.25	13
15	Genen Shouyu	13	17.45
16	Pavlova	17.45	39
17	Alice Mutton	39	62.5
18	Carnarvon Tigers	62.5	9.2
19	Teatime Chocolate Biscuits	9.2	81

```sql
-- 19. Obtener un listado que muestra el total de ventas por categoría de producto j
 -- total de ventas de la categoría siguiente

select
     sq2.*,
     lead(sq2.sum_category) over (
     order by sq2.category_name)
 from
     (
     select
         distinct on
         (c.category_name) c.category_name,
         sum(sq.sum_product_id) over (partition by c.category_name
     order by
         category_name) as sum_category
     from
         (
         select
             distinct on
             (od.product_id) od.product_id,
             sum(od.unit_price * od.quantity) over (partition by od.product_id) sum_p
         from
             order_details od ) sq,
         categories c,
         products p
     where
         sq.product_id = p.product_id
         and p.category_id = c.category_id
 ) sq2
```

```sql
-- 19. Obtener un listado que muestra el total de ventas por categoría de producto j
 -- total de ventas de la categoría siguiente

select
     sq2.*,
     lead(sq2.sum_category) over (
     order by sq2.category_name)
 from
     (
     select
         distinct on
         (c.category_name) c.category_name,
         sum(sq.sum_product_id) over (partition by c.category_name
     order by
         category name) as sum category
```

categories 1 ×

select sq2.*, lead(sq2.sum_category) over (orde Enter a SQL expression to filter results (use Ctrl+Space)

	category_name	sum_category	lead
1	Beverages	286,526.9500956535	113,694.7496814728
2	Condiments	113,694.7496814728	177,099.1006007194
3	Confections	177,099.1006007194	251,330.4997959137
4	Dairy Products	251,330.4997959137	100,726.7999253273
5	Grains/Cereals	100,726.7999253273	178,188.8009858131
6	Meat/Poultry	178,188.8009858131	105,268.6001739502
7	Produce	105,268.6001739502	141,623.0891823769
8	Seafood	141,623.0891823769	[NULL]

Z