

## Ejercicios SQL Bootcamp Data Engineer - EDVAI Consigna:

escribir la query/consulta necesaria para llegar al resultado (print).

```
select distinct category_name  
from categories c
```

### SELECT DISTINCT

1. Obtener una lista de todas las categorías distintas:

Print:

	ABC category_name
1	Beverages
2	Produce
3	Condiments
4	Grains/Cereals
5	Meat/Poultry
6	Dairy Products
7	Confections
8	Seafood

2. Obtener una lista de todas las regiones distintas para los clientes: Print:

```
select distinct region  
from customers c |
```

	ABC region
1	Lara
2	DF
3	ID
4	Táchira
5	SP
6	[NULL]
7	Nueva Esparta
8	CA
9	AK
10	Isle of Wight
11	WY
12	OR
13	MT
14	RJ
15	WA
16	NM
17	Co. Cork
18	BC
19	Québec

3. Obtener una lista de todos los títulos de contacto distintos:

```
select distinct contact_title  
from customers c
```

Print:

	ABC contact_title
1	Sales Manager
2	Owner
3	Owner/Marketing Assistant
4	Accounting Manager
5	Sales Associate
6	Sales Agent
7	Assistant Sales Agent
8	Assistant Sales Representative
9	Order Administrator
10	Marketing Manager
11	Sales Representative
12	Marketing Assistant

## ORDER BY

4. Obtener una lista de todos los clientes, ordenados por país:

Print:

```
select *
from customers c
order by country
```

customers 1 x

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

customer id	company name	contact name	contact title
OCEAN	Océano Atlántico Ltda.	Yvonne Moncada	Sales Agent
CACTU	Cactus Comidas para llevar	Patricio Simpson	Sales Agent
RANCH	Rancho grande	Sergio Gutiérrez	Sales Representative

	customer_id	company	contact_name	contact_title	address	city	region	postal_code	country
1	OCEAN	Océano Atlántico	Yvonne Moncada	Sales Agent	Ing. Gustavo Moncada 8581	Buenos Aires	(NULL)	1010	Argentina
2	CACTU	Cactus Comidas	Patricio Simpson	Sales Agent	Cerrito 333	Buenos Aires	(NULL)	1010	Argentina
3	RANCH	Rancho grande	Sergio Gutiérrez	Sales Representative	Av. del Libertador 900	Buenos Aires	(NULL)	1010	Argentina
4	ERNSH	Ernst Handel	Roland Mendel	Sales Manager	Kirchgasse 6	Graz	(NULL)	8010	Austria
5	PICCO	Piccolo und mehr	Georg Pipps	Sales Manager	Geislweg 14	Salzburg	(NULL)	5020	Austria
6	MAISD	Maison Dewey	Catherine Dewey	Sales Agent	Rue Joseph-Bens 532	Bruxelles	(NULL)	B-1180	Belgium
7	SUPRD	Suprêmes délices	Pascale Cartrain	Accounting Manager	Boulevard Tirou, 255	Charleroi	(NULL)	B-6000	Belgium
8	WELLI	Wellington Importers	Paula Parente	Sales Manager	Rua do Mercado, 12	Resende	SP	08737-363	Brazil
9	HANAR	Hanari Carnes	Mario Pontes	Accounting Manager	Rua do Paço, 67	Rio de Janeiro	RJ	05454-876	Brazil
10	GOURL	Gourmet Lanchonetes	André Fonseca	Sales Associate	Av. Brasil, 442	Campinas	SP	04876-786	Brazil
11	QUEDE	Que Pasa	Bernardo Batista	Accounting Manager	Rua da Panificadora, 12	Rio de Janeiro	RJ	02389-673	Brazil
12	QUEEN	Queen Cozinha	Lúcia Carvalho	Marketing Assistant	Alameda dos Canários, 891	Sao Paulo	SP	05487-020	Brazil
13	COMMI	Comércio Mineiro	Pedro Afonso	Sales Associate	Av. dos Lusíadas, 23	Sao Paulo	SP	05432-043	Brazil
14	TRADH	Tradição Hipermercado	Anabela Domingues	Sales Representative	Av. Inês de Castro, 414	Sao Paulo	SP	05634-030	Brazil
15	FAMIA	Família Argulhal	Arla Cruz	Marketing Assistant	Rua Orós, 92	Sao Paulo	SP	05442-030	Brazil
16	RICAR	Ricardo Adocica	Janete Limeira	Assistant Sales	Av. Copacabana, 267	Rio de Janeiro	RJ	02389-890	Brazil
17	BOTTM	Bottom-Dollar Products	Elizabeth Lincoln	Accounting Manager	23 Tsawassen Blvd.	Tsawassen	BC	T2F 8M4	Canada
18	LAUGB	Laughing Bacchus Wine	Yoshi Tannamuri	Marketing Assistant	1900 Oak St.	Vancouver	BC	V3F 2K1	Canada
19	MEREP	Mère Pâillarde	Jean Fresnière	Marketing Assistant	43 rue St. Laurent	Montréal	Québec	H1J 1C3	Canada
20	SIMOB	Simons bistro	Jytte Petersen	Owner	Vinbæltet 34	København	(NULL)	1734	Denmark
21	VAFFE	Vaffelbageriet	Palle Ibsen	Sales Manager	Smaalsloot 45	Århus	(NULL)	8200	Denmark

5. Obtener una lista de todos los pedidos, ordenados por id del empleado y fecha del pedido:

```
select *
from orders o
order by employee_id, order_date
```

1 X

\* From orders o order b | Enter a SQL expression to filter results (use Ctrl+Space)

order id	customer id	employee id	order date
10,258	ERNSH	1	1996-07-17
10,270	WARTH	1	1996-08-01
10,275	MAGAA	1	1996-08-07
10,285	QUICK	1	1996-08-20
10,292	TRADH	1	1996-08-28

Print:

	customer_id	employee_id	order_date	required_date	shipped_date	ship_via	freight
1	10,258 ERNSH	1	1996-07-17	1996-08-14	1996-07-23	1	140.5
2	10,270 WARTH	1	1996-08-01	1996-08-29	1996-08-02	1	136.5
3	10,275 MAGAA	1	1996-08-07	1996-09-04	1996-08-09	1	26.9
4	10,285 QUICK	1	1996-08-20	1996-09-17	1996-08-26	2	76.8
5	10,292 TRADH	1	1996-08-28	1996-09-25	1996-09-02	2	1.3
6	10,293 TORTU	1	1996-08-29	1996-09-26	1996-09-11	3	21.1
7	10,304 TORTU	1	1996-09-12	1996-10-10	1996-09-17	2	83.7
8	10,306 ROMEY	1	1996-09-18	1996-10-14	1996-09-23	3	7.5
9	10,311 DUMON	1	1996-09-20	1996-10-04	1996-09-26	3	24.6
10	10,314 RATTG	1	1996-09-25	1996-10-23	1996-10-04	2	74.1
11	10,316 RATTG	1	1996-09-27	1996-10-25	1996-10-08	3	150.1
12	10,325 KOENE	1	1996-10-09	1996-10-23	1996-10-14	3	64.
13	10,340 BONAP	1	1996-10-29	1996-11-26	1996-11-08	3	166.3
14	10,351 ERNSH	1	1996-11-11	1996-12-09	1996-11-20	1	162.3
15	10,357 LILAS	1	1996-11-19	1996-12-17	1996-12-02	3	34.8
16	10,361 QUICK	1	1996-11-22	1996-12-20	1996-12-03	2	183.1
17	10,364 EASTC	1	1996-11-26	1997-01-07	1996-12-04	1	71.1
18	10,371 LAMAI	1	1996-12-03	1996-12-31	1996-12-24	1	0.4
19	10,374 WOLZA	1	1996-12-05	1997-01-02	1996-12-09	3	3.9
20	10,376 MEREP	1	1996-12-09	1997-01-06	1996-12-13	2	20.1
21	10,377 SEVES	1	1996-12-09	1997-01-06	1996-12-13	3	22.21

# INSERT INTO

6. Insertar un nuevo cliente en la tabla Customers:

Print:

```
INSERT INTO customers (customer_id, company_name, contact_name, contact_title, address, city, region,
VALUES ('99256', 'blalba', 'roner', 'ale', 'pire 34', 'vancouver', 'south', '12345', 'allgun', null, '4
```

Name	Value
Updated Rows	1
Query	INSERT INTO customers (customer_id, company_name, contact_name, contact_title, address, city, region, VALUES ('99256', 'blalba', 'roner', 'ale', 'pire 34', 'vancouver', 'south', '12345', 'allgun', null, '42542')
Start time	Mon Apr 08 21:41:13 ART 2024
Finish time	Mon Apr 08 21:41:13 ART 2024

Updated Rows 1

7. Insertar una nueva región en la tabla Región:

Print:

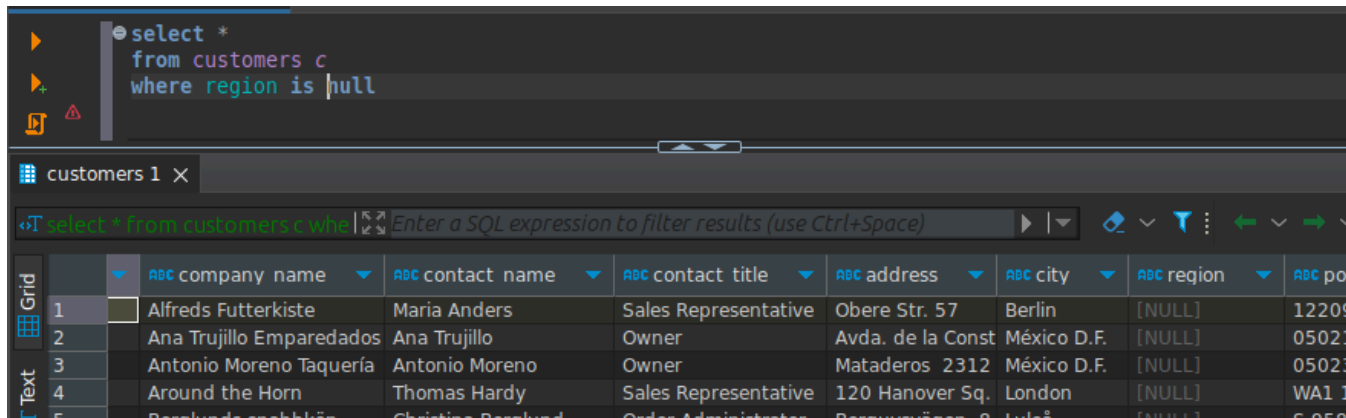
```
INSERT INTO region (region_id, region_description)
VALUES (99, 'gracias');
```

Name	Value
Updated Rows	1
Query	INSERT INTO region (region_id, region_description) VALUES (99, 'gracias')
Start time	Mon Apr 08 21:42:45 ART 2024
Finish time	Mon Apr 08 21:42:45 ART 2024

Updated Rows 1

## NULL -COALESCE

8. Obtener todos los clientes de la tabla Customers donde el campo Región es NULL:



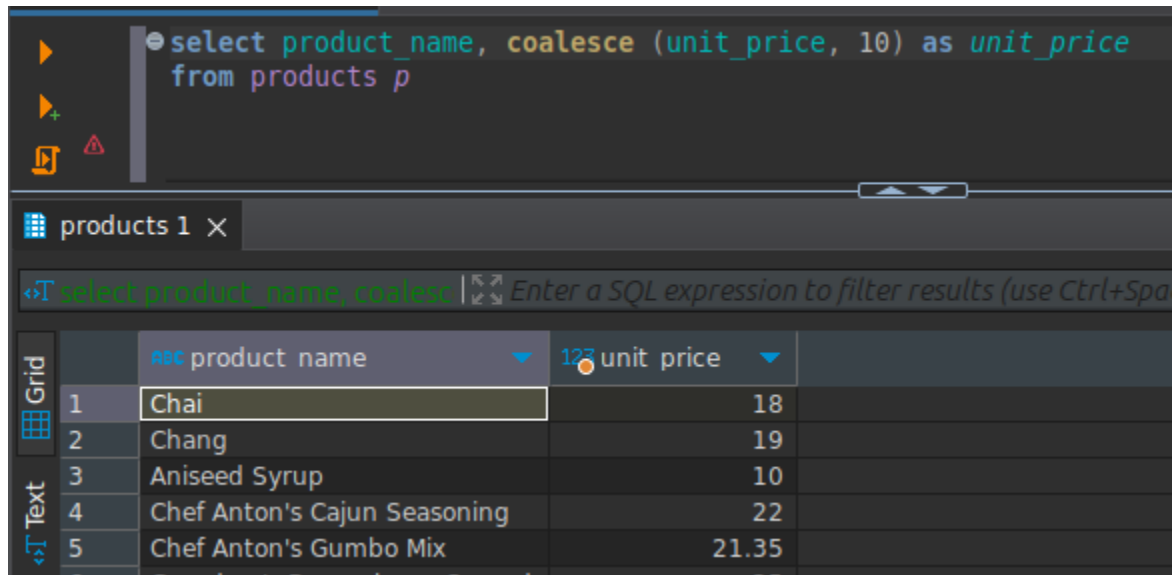
```
select *
from customers c
where region is null
```

	ABC company name	ABC contact name	ABC contact title	ABC address	ABC city	ABC region	ABC po
1	Alfreds Futterkiste	Maria Anders	Sales Representative	Obere Str. 57	Berlin	[NULL]	12209
2	Ana Trujillo Emparedados	Ana Trujillo	Owner	Avda. de la Constit	México D.F.	[NULL]	05021
3	Antonio Moreno Taquería	Antonio Moreno	Owner	Mataderos 2312	México D.F.	[NULL]	05021
4	Around the Horn	Thomas Hardy	Sales Representative	120 Hanover Sq.	London	[NULL]	WA1 1HP
5	Berglunds snabbköp	Christina Berglund	Order Administrator	Berguvsvägen 8	Luleå	[NULL]	S-951 88

Print:

	customer_id	company_name	contact_name	contact_title	address	city	region	postal_code
1	ALFKI	Alfreds Futterkiste	Maria Anders	Sales Representative	Obere Str. 57	Berlin	[NULL]	12209
2	ANATR	Ana Trujillo Emparedados	Ana Trujillo	Owner	Avda. de la Constit	México D.F.	[NULL]	05021
3	ANTON	Antonio Moreno Taquería	Antonio Moreno	Owner	Mataderos 2312	México D.F.	[NULL]	05021
4	AROUT	Around the Horn	Thomas Hardy	Sales Representative	120 Hanover Sq.	London	[NULL]	WA1 1HP
5	BERGS	Berglunds snabbköp	Christina Berglund	Order Administrator	Berguvsvägen 8	Luleå	[NULL]	S-951 88
6	BLAUS	Blauer See Delikatessen	Hanna Moos	Sales Representative	Forsterstr. 57	Mannheim	[NULL]	68305
7	BLONP	Blondesddel père et fils	Frédérique Citeaux	Marketing Manager	24, place Kléber	Strasbourg	[NULL]	67000
8	BOLID	Bólido Comidas preparadas	Martín Sommer	Owner	C/ Araquil, 67	Madrid	[NULL]	28015
9	BONAP	Bon app'	Laurence Lablanc	Owner	12, rue des Bouchers	Marseille	[NULL]	13008
10	BSBEV	B's Beverages	Victoria Ashworth	Sales Representative	Fauntleroy Circus	London	[NULL]	EC2 1FA
11	CACTU	Cactus Comidas para llevar	Patricio Simpson	Sales Agent	Cerrito 333	Buenos Aires	[NULL]	1010
12	CENTC	Centro comercial Moctezuma	Francisco Chang	Marketing Manager	Sierras de Granada	México D.F.	[NULL]	05021
13	CHOPS	Chop-suey Chinese	Yang Wang	Owner	Hauptstr. 29	Bern	[NULL]	3012
14	CONSH	Consolidated Holdings	Elizabeth Brown	Sales Representative	Berkeley Gardens	London	[NULL]	WX1 1HP
15	DRACD	Drachenblut Delikatessen	Sven Ottlieb	Order Administrator	Walserweg 21	Aachen	[NULL]	52064
16	DUMON	Du monde entier	Janine Labrune	Owner	67, rue des Cinqu	Nantes	[NULL]	44001
17	EASTC	Eastern Connection	Ann Devon	Sales Agent	35 King George	London	[NULL]	WX3 1HP
18	ERNSH	Ernst Handel	Roland Mendel	Sales Manager	Kirchgasse 6	Graz	[NULL]	8010
19	FISSA	FISSA Fabrica Inter. Salchik	Diego Roel	Accounting Manager	C/ Moralzarzal, 86	Madrid	[NULL]	28001
20	FOLIG	Foies gourmandes	Martine Rancé	Assistant Sales Agent	184, chaussée de	Lille	[NULL]	59001
21	FOLKO	Folk och få HB	Maria Larsson	Owner	Åkeroatan 24	Bräcke	[NULL]	S-84

9. Obtener Product\_Name y Unit\_Price de la tabla Products, y si Unit\_Price es NULL, use el precio estándar de \$10 en su lugar:



The screenshot shows a SQL IDE interface. At the top, a query editor contains the following SQL statement:

```
select product_name, coalesce (unit_price, 10) as unit_price
from products p
```

Below the query editor, a tab labeled "products 1" is active. Below the tab, a filter bar shows the query: `select product_name, coalesce`. Below the filter bar, a table grid displays the results of the query. The grid has two columns: "product name" and "unit price". The results are as follows:

	product name	unit price
1	Chai	18
2	Chang	19
3	Aniseed Syrup	10
4	Chef Anton's Cajun Seasoning	22
5	Chef Anton's Gumbo Mix	21.35

Print:



	ABC product_name	123 price
1	Chai	18
2	Chang	19
3	Aniseed Syrup	10
4	Chef Anton's Cajun Seasoning	22
5	Chef Anton's Gumbo Mix	21.35000038
6	Grandma's Boysenberry Spread	25
7	Uncle Bob's Organic Dried Pears	30
8	Northwoods Cranberry Sauce	40
9	Mishi Kobe Niku	97
10	Ikura	31
11	Queso Cabrales	21
12	Queso Manchego La Pastora	38
13	Konbu	6
14	Tofu	23.25
15	Genen Shouyu	13
16	Pavlova	17.45000076
17	Alice Mutton	39
18	Carnarvon Tigers	62.5
19	Teatime Chocolate Biscuits	9.19999981
20	Sir Rodney's Marmalade	81
21	Sir Rodnev's Scones	10

## INNER JOIN

10. Obtener el nombre de la empresa, el nombre del contacto y la fecha del pedido de todos los pedidos:

The screenshot shows a SQL IDE with a query editor at the top and a results grid below. The query is:

```
select c.company_name, c.contact_name, o.order_date
from customers c inner join orders o
on c.customer_id = o.customer_id
```

The results grid shows the first five rows of the query results:

	ABC company name	ABC company name	order date
1	Vins et alcools Chevalier	Vins et alcools Chevalier	1996-07-04
2	Toms Spezialitäten	Toms Spezialitäten	1996-07-05
3	Hanari Carnes	Hanari Carnes	1996-07-08
4	Victuailles en stock	Victuailles en stock	1996-07-08
5	Suprêmes délices	Suprêmes délices	1996-07-09

Print:

	ABC company_name	ABC contact_name	order_date
1	Vins et alcools Chevalier	Paul Henriot	1996-07-04
2	Toms Spezialitäten	Karin Josephs	1996-07-05
3	Hanari Carnes	Mario Pontes	1996-07-08
4	Victuailles en stock	Mary Saveley	1996-07-08
5	Suprêmes délices	Pascale Cartrain	1996-07-09
6	Hanari Carnes	Mario Pontes	1996-07-10
7	Chop-suey Chinese	Yang Wang	1996-07-11
8	Richter Supermarkt	Michael Holz	1996-07-12
9	Wellington Importadora	Paula Parente	1996-07-15
10	HILARION-Abastos	Carlos Hernández	1996-07-16
11	Ernst Handel	Roland Mendel	1996-07-17
12	Centro comercial Moctezuma	Francisco Chang	1996-07-18
13	Ottillies Käseladen	Henriette Pfalzheim	1996-07-19
14	Que Delícia	Bernardo Batista	1996-07-19
15	Rattlesnake Canyon Grocery	Paula Wilson	1996-07-22
16	Ernst Handel	Roland Mendel	1996-07-23
17	Folk och fä HB	Maria Larsson	1996-07-24
18	Blondesddsl père et fils	Frédérique Citeaux	1996-07-25
19	Wartian Herkku	Pirkko Koskitalo	1996-07-26
20	Frankenversand	Peter Franken	1996-07-29
21	GROSELLA-Restaurante	Manuel Pereira	1996-07-30

11. Obtener la identificación del pedido, el nombre del producto y el descuento de todos los detalles del pedido y productos:

```
select od.order_id, p.product_name , od.discount
from order_details od inner join products p
on od.product_id = p.product_id
```

order\_details(+) 1 x

select od.order\_id, p.product\_name, od.discount

	order id	product name	discount
1	10,248	Queso Cabrales	0
2	10,248	Singaporean Hokkien Fried Mee	0
3	10,248	Mozzarella di Giovanni	0
4	10,249	Tofu	0
5	10,249	Manjimup Dried Apples	0

Print:

order_id	product_name	discount
10,248	Queso Cabrales	0
10,248	Singaporean Hokkien Fried Mee	0
10,248	Mozzarella di Giovanni	0
10,249	Tofu	0
10,249	Manjimup Dried Apples	0
10,250	Jack's New England Clam Chowder	0
10,250	Manjimup Dried Apples	0.150000006
10,250	Louisiana Fiery Hot Pepper Sauce	0.150000006
10,251	Gustaf's Knäckebröd	0.050000001
10,251	Ravioli Angelo	0.050000001
10,251	Louisiana Fiery Hot Pepper Sauce	0
10,252	Sir Rodney's Marmalade	0.050000001
10,252	Geitost	0.050000001
10,252	Camembert Pierrot	0
10,253	Gorgonzola Telino	0
10,253	Chartreuse verte	0
10,253	Maxilaku	0
10,254	Guaraná Fantástica	0.150000006
10,254	Pâté chinois	0.150000006
10,254	Longlife Tofu	0
10,255	Chana	0

## LEFT JOIN

12. Obtener el identificador del cliente, el nombre de la compañía, el identificador y la fecha de la orden de todas las órdenes y aquellos clientes que hagan match :

```
select c.customer_id , c.company_name , o.order_id , o.order_date |
from customers c left join orders o
on c.customer_id = o.customer_id
```

customers(+) 1 x

select c.customer\_id , c.comp | Enter a SQL expression to filter results (use Ctrl+Space)

	customer id	company name	order id	order date
1	VINET	Vins et alcools Chevalier	10,248	1996-07-04
2	TOMSP	Toms Spezialitäten	10,249	1996-07-05
3	HANAR	Hanari Carnes	10,250	1996-07-08
4	VICTE	Victuailles en stock	10,251	1996-07-08
5	SUPRD	Suprêmes délices	10,252	1996-07-09

Print:

customer_id	company_name	order_id	order_date
VINET	Vins et alcools Chevalier	10,248	1996-07-04
TOMSP	Toms Spezialitäten	10,249	1996-07-05
HANAR	Hanari Carnes	10,250	1996-07-08
VICTE	Victuailles en stock	10,251	1996-07-08
SUPRD	Suprêmes délices	10,252	1996-07-09
HANAR	Hanari Carnes	10,253	1996-07-10
CHOPS	Chop-suey Chinese	10,254	1996-07-11
RICSU	Richter Supermarkt	10,255	1996-07-12
WELLI	Wellington Importadora	10,256	1996-07-15
HILAA	HILARION-Abastos	10,257	1996-07-16
ERNSH	Ernst Handel	10,258	1996-07-17
CENTC	Centro comercial Moctezuma	10,259	1996-07-18
OTTIK	Ottilies Käseladen	10,260	1996-07-19
QUEDE	Que Delícia	10,261	1996-07-19
RATTC	Rattlesnake Canyon Grocery	10,262	1996-07-22
ERNSH	Ernst Handel	10,263	1996-07-23
FOLKO	Folk och fä HB	10,264	1996-07-24
BLONP	Blondesddsl père et fils	10,265	1996-07-25
WARTH	Wartian Herkku	10,266	1996-07-26
FRANK	Frankenversand	10,267	1996-07-29
GROSR	GROSELLA-Restaurante	10,268	1996-07-30

13. Obtener el identificador del empleados, apellido, identificador de territorio y descripción del territorio de todos los empleados y aquellos que hagan match en territorios: Print:

```
select e.employee_id , e.last_name , t.territory_id , t.territory_description
from employees e
LEFT JOIN employee_territories et
ON e.employee_id = et.employee_id
LEFT JOIN territories t
ON et.territory_id = t.territory_id;
```

employees(+) 1 x

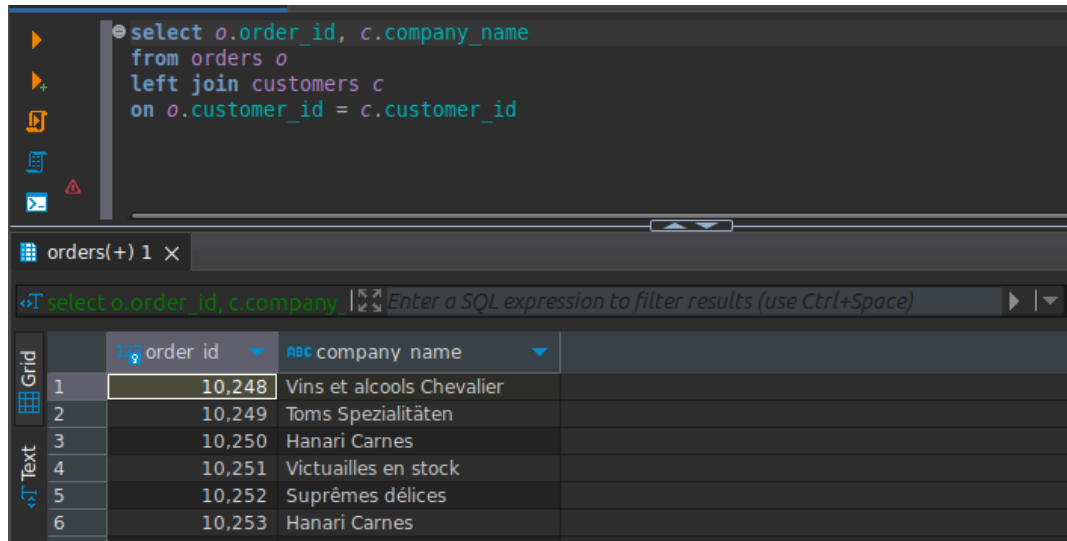
select e.employee\_id , e.last\_name , t.territory\_id , t.territory\_description

	employee id	last name	territory id	territory description
1	1	Davolio	06897	Wilton
2	1	Davolio	19713	Neward
3	2	Fuller	01581	Westboro
4	2	Fuller	01730	Bedford
5	2	Fuller	01833	Georgetow
6	2	Fuller	02116	Boston

123 emplo	Ctrl+click to open SQL console	ABC territory_id	ABC territory_description
1	Davolio	06897	Wilton
1	Davolio	19713	Neward
2	Fuller	01581	Westboro
2	Fuller	01730	Bedford
2	Fuller	01833	Georgetow
2	Fuller	02116	Boston
2	Fuller	02139	Cambridge
2	Fuller	02184	Braintree
2	Fuller	40222	Louisville
3	Leverling	30346	Atlanta
3	Leverling	31406	Savannah
3	Leverling	32859	Orlando
3	Leverling	33607	Tampa
4	Peacock	20852	Rockville
4	Peacock	27403	Greensboro
4	Peacock	27511	Cary
5	Buchanan	02903	Providence
5	Buchanan	07960	Morristown
5	Buchanan	08837	Edison
5	Buchanan	10019	New York
5	Buchanan	10038	New York

14. Obtener el identificador de la orden y el nombre de la empresa de todos las órdenes y aquellos clientes que hagan match:

Print:



The screenshot shows a SQL IDE interface. The top pane contains a SQL query:

```
select o.order_id, c.company_name
from orders o
left join customers c
on o.customer_id = c.customer_id
```

The bottom pane shows the results in a grid view. The first row is highlighted, showing order\_id 10,248 and company\_name Vins et alcools Chevalier.

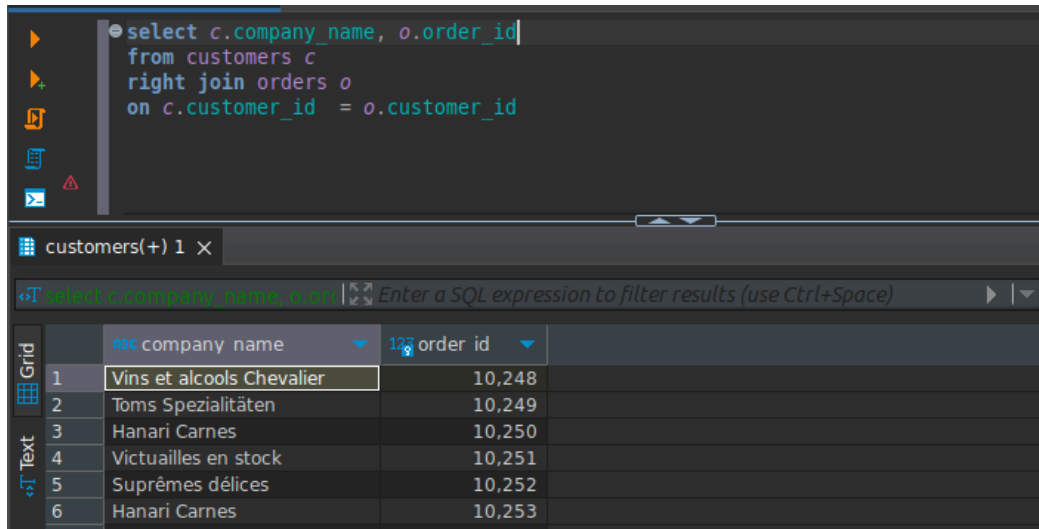
order id	company name
10,248	Vins et alcools Chevalier
10,249	Toms Spezialitäten
10,250	Hanari Carnes
10,251	Victuailles en stock
10,252	Suprêmes délices
10,253	Hanari Carnes

order id	company name
10,248	Vins et alcools Chevalier
10,249	Toms Spezialitäten
10,250	Hanari Carnes
10,251	Victuailles en stock
10,252	Suprêmes délices
10,253	Hanari Carnes
10,254	Chop-suey Chinese
10,255	Richter Supermarkt
10,256	Wellington Importadora
10,257	HILARION-Abastos
10,258	Ernst Handel
10,259	Centro comercial Moctezuma
10,260	Otilies Käseladen
10,261	Que Delícia
10,262	Rattlesnake Canyon Grocery
10,263	Ernst Handel
10,264	Folk och fä HB
10,265	Blondesddsl père et fils
10,266	Wartian Herkku
10,267	Frankenversand
10,268	GROSELLA-Restaurante

## RIGHT JOIN

15. Obtener el identificador de la orden, y el nombre de la compañía de todas las órdenes y aquellos clientes que hagan match:

Print:



The screenshot shows a SQL IDE with a query editor at the top and a results grid at the bottom. The query is a right join between the 'customers' and 'orders' tables on the 'customer\_id' column. The results grid displays the 'company name' and 'order id' for the first six rows of the result set.

```
select c.company_name, o.order_id
from customers c
right join orders o
on c.customer_id = o.customer_id
```

	company name	order id
1	Vins et alcools Chevalier	10,248
2	Toms Spezialitäten	10,249
3	Hanari Carnes	10,250
4	Victuailles en stock	10,251
5	Suprêmes délices	10,252
6	Hanari Carnes	10,253

16. Obtener el nombre de la compañía, y la fecha de la orden de todas las órdenes y aquellos transportistas que hagan match. Solamente para aquellas ordenes del año 1996:

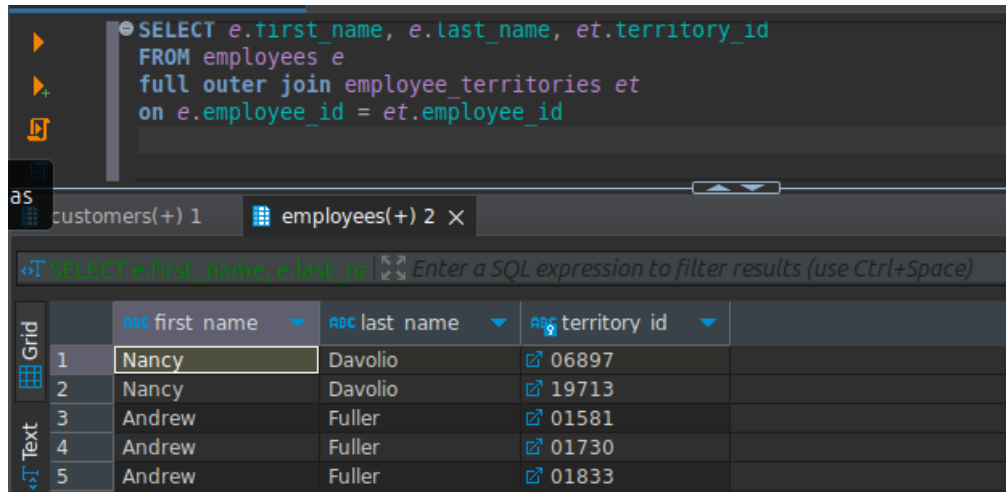
Print:

ESTE AUN NO LO LOGRE...

Transportistas = SHIPPERS?

## FULL OUTER JOIN

17. Obtener nombre y apellido del empleados y el identificador de territorio, de todos los empleados y aquellos que hagan match o no de employee\_territories:



The screenshot shows a SQL query in the SQL Editor window:

```
SELECT e.first_name, e.last_name, et.territory_id
FROM employees e
full outer join employee_territories et
on e.employee_id = et.employee_id
```

Below the query, the Results window displays the results in a grid format. The columns are 'first\_name', 'last\_name', and 'territory\_id'. The results show 5 rows of data:

first_name	last_name	territory_id
Nancy	Davolio	06897
Nancy	Davolio	19713
Andrew	Fuller	01581
Andrew	Fuller	01730
Andrew	Fuller	01833

Print:

first_name	last_name	territory_id
Nancy	Davolio	06897
Nancy	Davolio	19713
Andrew	Fuller	01581
Andrew	Fuller	01730
Andrew	Fuller	01833
Andrew	Fuller	02116
Andrew	Fuller	02139
Andrew	Fuller	02184
Andrew	Fuller	40222
Janet	Leverling	30346
Janet	Leverling	31406
Janet	Leverling	32859
Janet	Leverling	33607
Margaret	Peacock	20852
Margaret	Peacock	27403
Margaret	Peacock	27511
Steven	Buchanan	02903
Steven	Buchanan	07960
Steven	Buchanan	08837
Steven	Buchanan	10019
Steven	Buchanan	10038



18. Obtener el identificador de la orden, precio unitario, cantidad y total de todas las órdenes y aquellas órdenes detalles que hagan match o no:

```
SELECT o.order_id , od.unit_price, od.quantity, od.quantity * od.unit_price as total
FROM orders o
full outer join order_details od
on o.order_id = od.order_id
```

order id	unit price	quantity	total
10,248	14	12	168
10,248	9.8	10	98.0000019073
10,248	34.8	5	173.9999961853
10,249	18.6	9	167.4000034332
10,249	42.4	40	1,696.0000610352

Print:



## UNION

19. Obtener la lista de todos los nombres de los clientes y los nombres de los proveedores: Print:

```
select company_name
from customers c
union
select company_name
from suppliers s
```

Results 1 X

select company\_name from c | Enter

	company name
1	Heli Süßwaren GmbH & Co. KG
2	Wilman Kala
3	Magazzini Alimentari Riuniti
4	Escargots Nouveaux
5	Wellington Importadora
6	The Cracker Box
7	Toms Spezialitäten
8	Furia Bacalhau e Frutos do Mar
9	Lehmanns Marktstand
10	Mère Paillard
11	Produit de la

20. Obtener la lista de los nombres de todos los empleados y los nombres de los gerentes de departamento:

No logro encontrar los nombres de los gerentes de departamento

```
select e.first_name
from employees e
```

employees 1 X

```
select e.first_name from empl
```

asc first name	
Nancy	
Andrew	
Janet	
Margaret	
Steven	
Michael	
Robert	
Laura	
Anne	

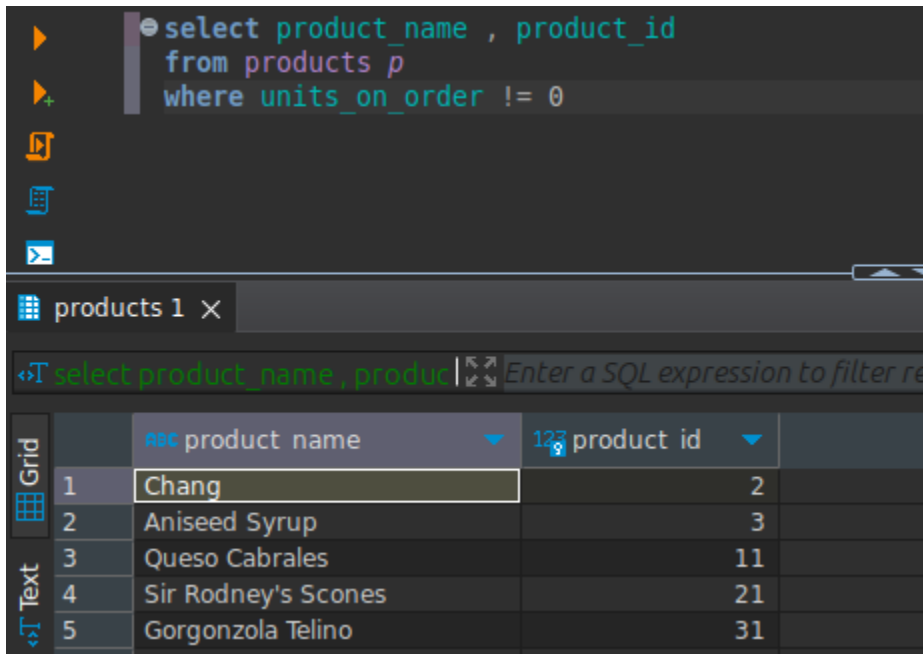
Print:



## SUBQUERIES

21. Obtener los productos del stock que han sido vendidos:

Creo debería ser esto



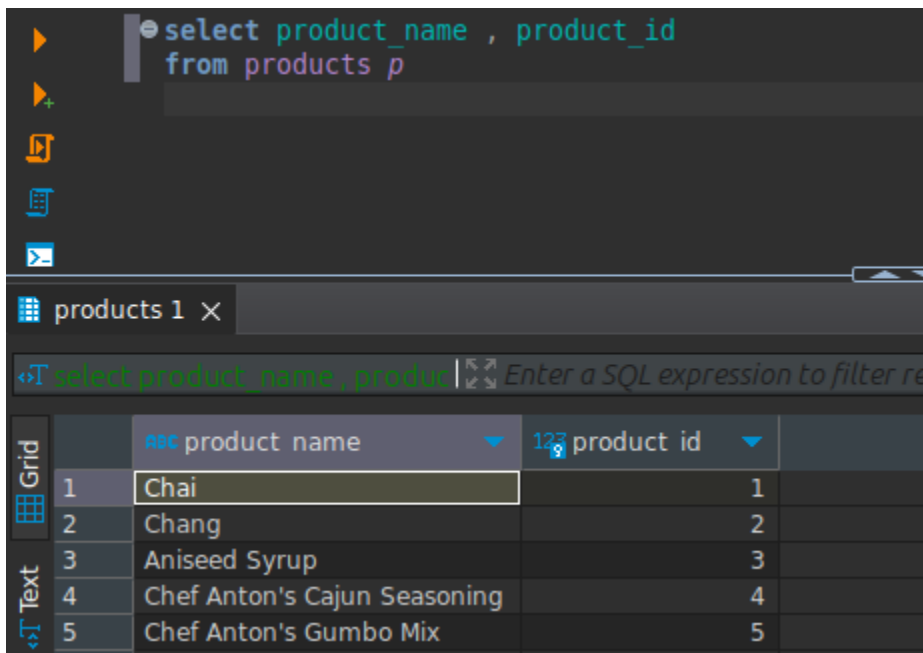
The screenshot shows a SQL IDE with a query editor at the top containing the following SQL code:

```
select product_name , product_id  
from products p  
where units_on_order != 0
```

Below the query editor, a tab labeled "products 1" is active. A filter bar shows the query: `select product_name , product_id`. Below the filter bar, a table displays the results of the query:

	product name	product id
1	Chang	2
2	Aniseed Syrup	3
3	Queso Cabrales	11
4	Sir Rodney's Scones	21
5	Gorgonzola Telino	31

Pero el ejemplo es esto



The screenshot shows a SQL IDE with a query editor at the top containing the following SQL code:

```
select product_name , product_id  
from products p
```

Below the query editor, a tab labeled "products 1" is active. A filter bar shows the query: `select product_name , product_id`. Below the filter bar, a table displays the results of the query:

	product name	product id
1	Chai	1
2	Chang	2
3	Aniseed Syrup	3
4	Chef Anton's Cajun Seasoning	4
5	Chef Anton's Gumbo Mix	5

22. Obtener los clientes que han realizado un pedido con destino a Argentina: Print:

```
select company_name
from customers c
where country = 'Argentina'
```

customers 1 X

select company\_name from c | Enter a SQL expression to filter

	asc company name	
1	Cactus Comidas para llevar	
2	Océano Atlántico Ltda.	
3	Rancho grande	

23. Obtener el nombre de los productos que nunca han sido pedidos por clientes de Francia:

GPT me sugirió esta posibilidad pero no coincide

Entiendo que selecciona los customer de francia, vincula su id con la orden, la orden con el producto y me daría como resultado los match. Pero no confío.

```
SELECT p.product_name
FROM products p
WHERE p.product_id IN (
  SELECT od.product_id
  FROM order_details od
  WHERE od.order_id IN (
    SELECT o.order_id
    FROM orders o
    WHERE o.customer_id IN (
      SELECT c.customer_id
      FROM customers c
      WHERE c.country != 'France'
    )
  )
);
```

products 1 X

SELECT p.product\_name FROM | Enter a SQL expression to filter

	asc product name	
1	Chai	
2	Chang	
3	Aniseed Syrup	
4	Chef Anton's Cajun Seasoning	
5	Chef Anton's Gumbo Mix	
6	Grandma's Boysenberry Spread	
7	Uncle Bob's Organic Dried Pears	

Refresh Save Cancel

## GROUP BY

24. Obtener la cantidad de productos vendidos por identificador de orden: Print:

```
select od.order_id, sum(od.quantity) as sum
from order_details od
group by od.order_id
```

order\_details 1 X

select od.order\_id, sum(od.quantity) as sum

	123 order id	123 sum
1	11,038	37
2	10,782	1
3	10,725	22
4	10,422	24

25. Obtener el promedio de productos en stock por producto

```
select p.product_name , avg(p.units_in_stock) as avg
from products p
group by p.product_name
```

products 1 X

select p.product\_name , avg(p.units\_in\_stock) as avg

	asc product name	123 avg
1	Pâté chinois	115
2	Flotemysost	26
3	Camembert Pierrot	19
4	Inlagd Sill	112
5	Thüringer Rostbratwurst	0
6	Gustaf's Knäckebröd	104
7	Carnarvon Tigers	42
8	Tofu	35

Print:

## HAVING

26. Cantidad de productos en stock por producto, donde haya más de 100 productos en stock

```
select p.product_name, sum(p.units_in_stock) as sum
from products p
group by p.product_name
having sum(p.units_in_stock) > 100
```

products 1 x

select p.product\_name, sum(p | Enter a SQL expression to filter results (use Ctrl+Sp

	product name	sum
1	Pâté chinois	115
2	Inlagd Sill	112
3	Gustaf's Knäckebröd	104
4	Röd Kaviar	101
5	Rhönbräu Klosterbier	125
6	Sirop d'érable	113
7	Grandma's Boysenberry Spread	120
8	Sasquatch Ale	111

Print:



27. Obtener el promedio de pedidos por cada compañía y solo mostrar aquellas con un promedio de pedidos superior a 10: No lo entendí... :(

```

SELECT c.company_name, averageorders
FROM customers c
JOIN (
    SELECT o.customer_id, AVG(od.quantity) AS averageorders
    FROM orders o
    INNER JOIN order_details od ON o.order_id = od.order_id
    GROUP BY o.customer_id
) AS subquery ON c.customer_id = subquery.customer_id
WHERE averageorders >= 10;

```

customers 1 X

SELECT c.company\_name, ave | Enter a SQL expression to filter results (use Ctrl+Spa

	ABC company name	123 averageorders
1	Alfreds Futterkiste	14.5
2	Antonio Moreno Taquería	21.1176470588
3	Around the Horn	21.6666666667
4	Berglunds snabbköp	19.25
5	Blauer See Delikatessen	10
6	Blondesddsl père et fils	25.6153846154
7	Bólido Comidas preparadas	31.6666666667
8	Bon app'	22.2727272727
9	Bottom-Dollar Markets	27.3142857143
10	B's Beverages	13.3181818182
11	Cactus Comidas para llevar	10.4545454545

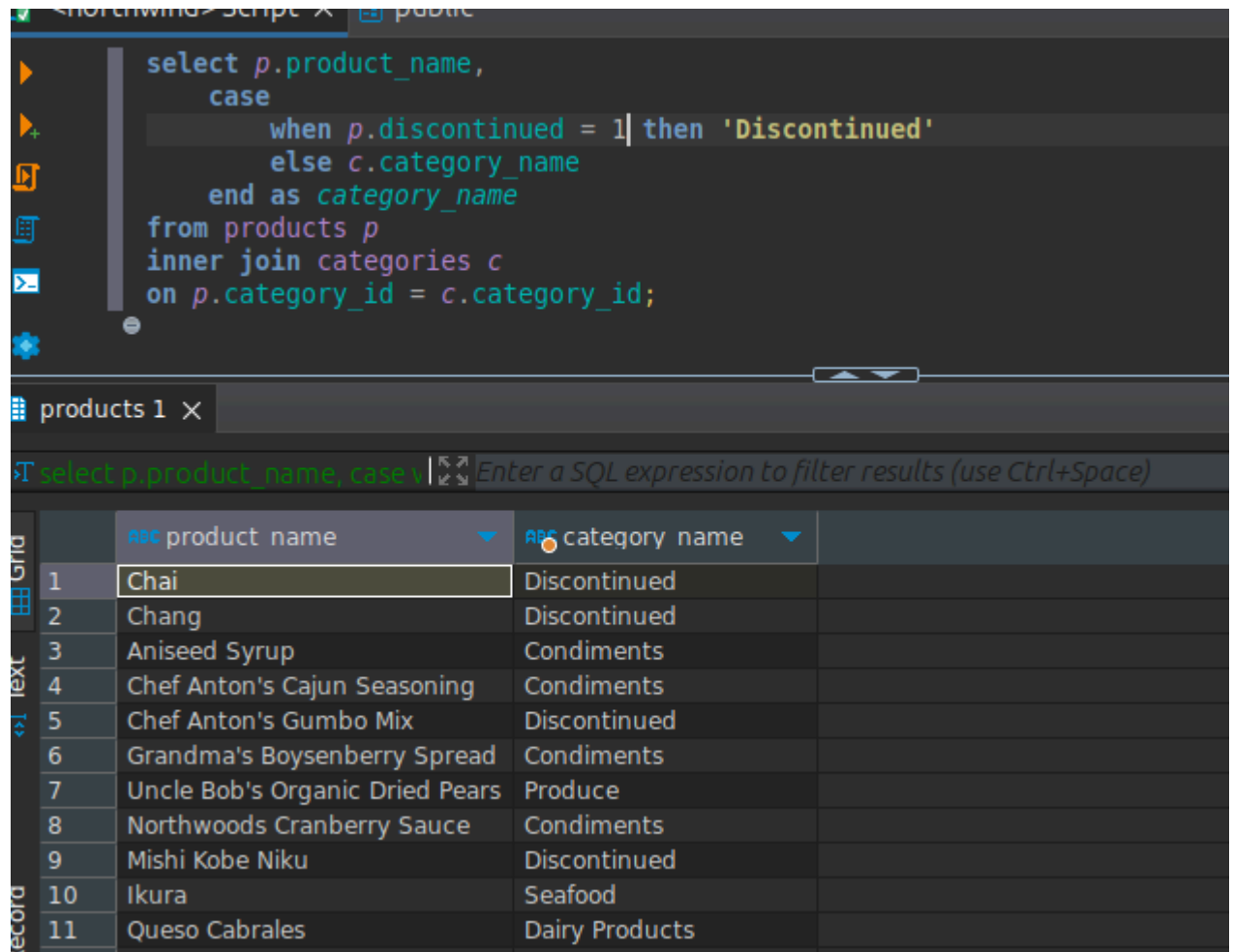
Print:

	ABC company_name	123 averageorders
1	Wilman Kala	10,807.1428571429
2	Magazzini Alimentari Riuniti	10,632.6
3	The Cracker Box	10,800.6666666667
4	Wellington Importadora	10,695.2222222222
5	Let's Stop N Shop	10,729.25
6	Furia Bacalhau e Frutos do Mar	10,552.125
7	Toms Spezialitäten	10,542.6666666667
8	Lehmanns Marktstand	10,617.7333333333
9	Mère Paillard	10,515.8461538462
10	Rattlesnake Canyon Grocery	10,600.1111111111
11	Que Delícia	10,565.4444444444
12	The Big Cheese	10,703.75
13	Hungry Owl All-Night Grocers	10,616.3157894737
14	Königlich Essen	10,629.2571428571



## CASE

28. Obtener el nombre del producto y su categoría, pero muestre "Discontinued" en lugar del nombre de la categoría si el producto ha sido descontinuado:



The screenshot shows a SQL IDE with a query editor and a results grid. The query in the editor is:

```
select p.product_name,  
       case  
         when p.discontinued = 1 then 'Discontinued'  
         else c.category_name  
       end as category_name  
from products p  
inner join categories c  
on p.category_id = c.category_id;
```

The results grid displays the following data:

	product name	category name
1	Chai	Discontinued
2	Chang	Discontinued
3	Aniseed Syrup	Condiments
4	Chef Anton's Cajun Seasoning	Condiments
5	Chef Anton's Gumbo Mix	Discontinued
6	Grandma's Boysenberry Spread	Condiments
7	Uncle Bob's Organic Dried Pears	Produce
8	Northwoods Cranberry Sauce	Condiments
9	Mishi Kobe Niku	Discontinued
10	Ikura	Seafood
11	Queso Cabrales	Dairy Products

Print:

29. Obtener el nombre del empleado y su título, pero muestre "Gerente de Ventas" en lugar del título si el empleado es un gerente de ventas (Sales Manager):

Print:

```
select e.first_name, e.last_name,
       case
         when e.title = 'Sales Manager' then 'Gerente de Ventas'
         else e.title
       end as job_title
from employees e
```

employees 1 X

Select e.first\_name, e.last\_name | Enter a SQL expression to filter results (use Ctrl+Space)

first name	last name	job title
Nancy	Davolio	Sales Representative
Andrew	Fuller	Vice President, Sales
Janet	Leverling	Sales Representative
Margaret	Peacock	Sales Representative
Steven	Buchanan	Gerente de Ventas
Michael	Suyama	Sales Representative
Robert	King	Sales Representative
Laura	Callahan	Inside Sales Coordinator
Anne	Dodsworth	Sales Representative