

Sprint_2

Nivell 1

Ejercicio 1

aggiungi screenshot tablas y descripcion columnas

En la base de datos “transactions” hay dos tablas:

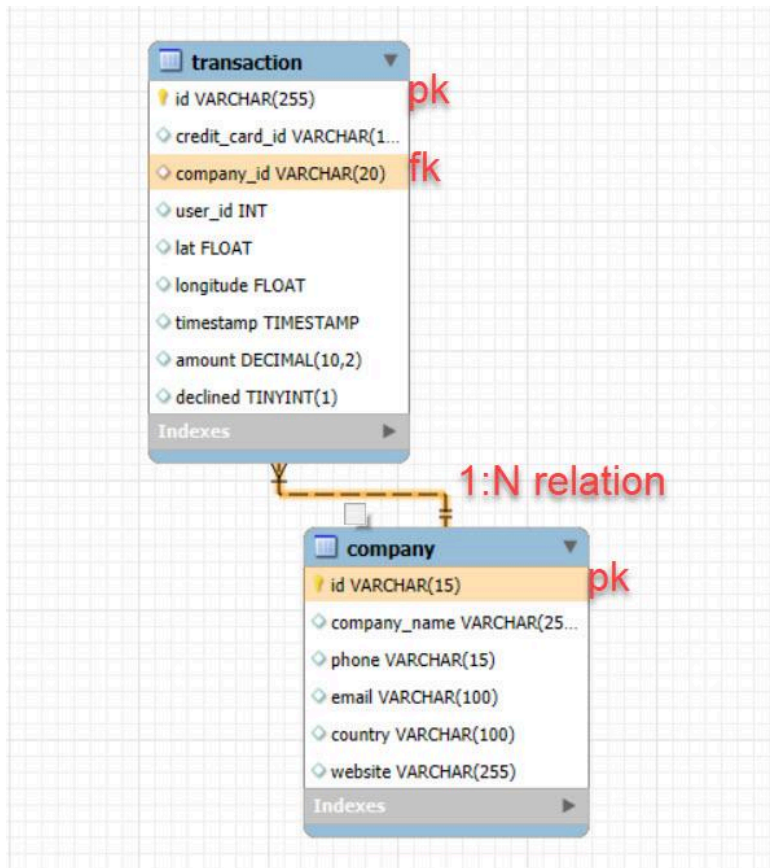
- company
- transaction

Relación

Son conectadas entre ellas con una relación 1:N (uno a muchos, a través de la primary key “id” contenida en la tabla “company” que se conecte a la foreign key “company_id” contenida en la tabla “transaction”.

O sea: una empresa puede tener más transacciones; cada transacción está conectada a una sola empresa.

Schema ER:



La tabla “company” funciona como tabla de dimensión, mientras la tabla “transaction” funciona como tabla de hechos.

Estructura de los datos

Tabla “company”

estructura datos datos_introduir_sprint2 sprint_2* Ejercizi_subqueries_3_conAI

```

1 • USE transactions;
2
3 • SELECT *
4 FROM company
5 LIMIT 5;
  
```

Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content: Result Grid Form Editor

id	company_name	phone	email	country	website
b-2222	Ac Fermentum Incorporated	06 85 56 52 33	donec.porttitor.tellus@yahoo.net	Germany	https://instagram.com/site
b-2226	Magna A Neque Industries	04 14 44 64 62	risus.donec.nibh@icloud.org	Australia	https://whatsapp.com/group/9
b-2230	Fusce Corp.	08 14 97 58 85	risus@protonmail.edu	United States	https://pinterest.com/sub/cars
b-2234	Convallis In Incorporated	06 66 57 29 50	mauris.ut@aol.co.uk	Germany	https://cnn.com/user/110
b-2238	Ante Iaculis Nec Foundation	08 23 04 99 53	sed.dictum.proin@outlook.ca	New Zealand	https://netflix.com/settings
NULL	NULL	NULL	NULL	NULL	NULL

company 1 x Apply Revert

Column count: 6

Row count: 100

Columns:

id	varchar(15)	PK
company_name	varchar(255)	
phone	varchar(15)	
email	varchar(100)	
country	varchar(100)	
website	varchar(255)	

Tabla "transaction"

The screenshot shows a database management interface with a SQL query editor and a result grid. The query is: `SELECT * FROM transaction LIMIT 5;`. The result grid displays the following data:

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
00043A49-2949-494B-A5DD-ASBAE3BB19DD	CcS-9294	b-2458	4713	46.1999	1.43554	2024-08-28 07:16:46	395.43	0
000447FE-B650-4DCF-8SDE-C7ED0EE1CAAD	CcS-5019	b-2370	438	41.5972	12.2218	2016-12-21 20:07:18	155.63	0
00045D68-ED2E-4F2F-8186-CEE074D875D0	CcS-6699	b-2390	2118	29.7573	-95.3796	2020-07-14 15:37:45	326.01	0
000481C3-1C26-4FEF-83A0-4CD0EB004BBD	CcS-6696	b-2230	2115	53.5489	-113.503	2017-09-04 19:44:53	161.60	0
00051AA4-9CBE-4268-B070-C38062A1B3E2	CcS-7606	b-2266	3025	52.2084	5.69081	2017-01-05 18:19:25	148.91	0
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Column count: 9
Row count: 100283

Columns:

id	varchar(255)	PK
credit_card_id	varchar(15)	
company_id	varchar(20)	FK por la tabla company (company_id -> id)
user_id	int	
lat	float	
longitude	float	
timestamp	timestamp	
amount	decimal(10,2)	

declined

tinyint(1)

Tipo de Variables

Reconozco también algunos tipos de variables que contienen la dos tablas:

- VARCHAR > es una cadena de texto (el número entre paréntesis nos dice cuantos caracteres puede contener como máximo).
- FLOAT > es un número con coma flotante
- INT > es un número entero
- TIMESTAMP > es una fecha y ora

Ejercicio 2

Utilizando JOIN realizarás las siguientes consultas:

- a) Listado de los países que están generando ventas.

The screenshot shows a database management tool interface. The top pane displays a SQL query for Exercise 2, part a, which lists countries generating sales using an INNER JOIN between the 'company' and 'transaction' tables, filtering for 'declined = 0'. The bottom pane shows the 'Result Grid' with a list of countries: Australia, Belgium, Canada, China, France, Germany, Ireland, Italy, Netherlands, New Zealand, Norway, and Spain. The 'Output' pane at the bottom shows the execution log, indicating that the query returned 15 rows.

```
10
11 -- Ejercicio 2
12 -- CON JOIN
13 -- a) Listado de los países que están generando ventas
14 • SELECT c.country
15 FROM company AS c
16 INNER JOIN transaction AS t
17 ON c.id = t.company_id
18 WHERE declined = 0 -- que están generando ventas (tienen que ser "declined" = 0 = false)
19 GROUP BY c.country
20 ORDER BY country ASC; -- añadido para una mayor legibilidad
```

country
Australia
Belgium
Canada
China
France
Germany
Ireland
Italy
Netherlands
New Zealand
Norway
Spain

#	Time	Action	Message
1	11:09:32	USE transactions	0 row(s) affected
2	11:09:43	SELECT c.country FROM company AS c INNER JOIN transaction AS t ON c.id = t.company_id WHERE...	15 row(s) returned

- b) Desde cuántos países se generan las ventas.

estructura dades dades_introduir_sprint2 sprint_2 x

21
22 -- b) Desde cuántos países se generan las ventas
23 • SELECT COUNT(DISTINCT country) AS n_paises
24 FROM company AS c
25 INNER JOIN transaction AS t
26 ON c.id = t.company_id
27 WHERE t.declined = 0; -- países que generan ventas
28

Result Grid

n_paises
15

Result 5 x Read Only

Output

Action Output

#	Time	Action	Message
1	11:13:06	SELECT COUNT(DISTINCT country) AS n_paises FROM company AS c INNER JOIN transaction AS t ...	1 row(s) returned

c) Identifica a la compañía con la mayor media de ventas.

estructura dades dades_introduir_sprint2 sprint_2 x

29 -- c) Identifica a la compañía con la mayor media de ventas
30 • SELECT
31 c.id,
32 c.company_name,
33 AVG(t.amount) AS avg_amount
34 FROM company AS c
35 INNER JOIN transaction AS t
36 ON c.id = t.company_id
37 GROUP BY
38 c.id,
39 c.company_name
40 ORDER BY avg_amount DESC
41 LIMIT 1;
42

Result Grid

id	company_name	avg_amount
b-2222	Ac Fermentum Incorporated	284.867160

Result 6 x Read Only

Output

Action Output

#	Time	Action	Message
1	11:14:12	SELECT c.id, c.company_name, AVG(t.amount) AS avg_amount FROM company AS c INNER JOIN tra...	1 row(s) returned

Ejercicio 3

Utilizando sólo subconsultas (sin utilizar JOIN):

- a) Muestra todas las transacciones realizadas por empresas de Alemania.

Opcion 1 - sin WHEN EXISTS

```
-- Ejercicio 3
-- SIN JOIN
-- a) Muestra todas las transacciones realizadas por empresas de Alemania
SELECT *
FROM transaction
WHERE company_id IN (
    SELECT
        id
    FROM company
    WHERE country = "Germany"
)
```

The screenshot shows a database IDE with a SQL query editor and a result grid. The query is a SELECT statement filtering transactions by company_id, using a subquery with IN to find companies in Germany. The result grid displays columns: id, credit_card_id, company_id, user_id, lat, longitude, timestamp, amount, and declined. The output shows 13291 rows returned.

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
0013803B-206D-4C03-94B7-63A2676EB984	CcS-4899	b-2222	318	41.3781	12.447	2020-03-25 10:43:43	426.36	0
0013C1B6-3B84-4D6C-8154-E2B3FEBCA8E9	CcS-5070	b-2222	489	41.3814	2.18176	2020-12-17 18:15:37	316.90	0
00201A11-2E62-44C4-941D-198FC8D877F0	CcU-3512	b-2222	193	55.5704	-3.65129	2021-01-22 23:44:27	453.04	0
00235618-0A5C-4D49-9DCB-83A940SD8923	CcS-8137	b-2222	3556	59.8421	18.729	2020-09-09 15:43:19	263.14	0
005ASA7B-1F1A-4B6C-9B15-1625A78C9C38	CcS-8998	b-2222	4417	41.1591	-8.63905	2024-05-15 09:10:11	442.01	0
00687139-4882-4FFA-8E73-820376F04AB4	CcS-4870	b-2222	289	51.1966	10.4669	2019-03-09 19:37:49	524.84	0

Opcion 2 - con WHEN EXISTS

Con IN: la subquery hace primero una lista y después comprueba si hay id iguales.

Con EXISTS: comprueba fila por fila de la tabla transaction. Es más eficiente.

```
-- con WHEN EXISTS - a) Muestra todas las transacciones realizadas por empresas de Alemania
SELECT *
FROM transaction
WHERE EXISTS (SELECT 1
    FROM company AS c
    WHERE c.id = transaction.company_id
    AND c.country = "Germany"
)
```

The screenshot shows a database IDE with a SQL query editor and a result grid. The query is a SELECT statement filtering transactions by company_id, using a subquery with EXISTS to find companies in Germany. The result grid displays columns: id, credit_card_id, company_id, user_id, lat, longitude, timestamp, amount, and declined. The output shows 13291 rows returned.

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
0013803B-206D-4C03-94B7-63A2676EB984	CcS-4899	b-2222	318	41.3781	12.447	2020-03-25 10:43:43	426.36	0
0013C1B6-3B84-4D6C-8154-E2B3FEBCA8E9	CcS-5070	b-2222	489	41.3814	2.18176	2020-12-17 18:15:37	316.90	0
00201A11-2E62-44C4-941D-198FC8D877F0	CcU-3512	b-2222	193	55.5704	-3.65129	2021-01-22 23:44:27	453.04	0
00235618-0A5C-4D49-9DCB-83A940SD8923	CcS-8137	b-2222	3556	59.8421	18.729	2020-09-09 15:43:19	263.14	0
005ASA7B-1F1A-4B6C-9B15-1625A78C9C38	CcS-8998	b-2222	4417	41.1591	-8.63905	2024-05-15 09:10:11	442.01	0

- b) Lista las empresas que han realizado transacciones por un amount superior a la media de todas las transacciones.

The screenshot shows a SQL IDE with a query editor and a result grid. The query is as follows:

```
66 WHERE c.country = "Germany";
67 */
68
69 -- b) lista las empresas que han realizado transacciones por un amount superior a la media de todas las transacciones
70 • SELECT
71     DISTINCT company_id
72 FROM transaction
73 WHERE amount > (
74     SELECT AVG(amount) AS avg_amount -- calculo la media de todas las transacciones
75     FROM transaction
76 )
77
78
79 -- c) Eliminarán del sistema las emopras que carecen de transacciones registradas. entrega el listado de estas emopras
```

The result grid shows the following data:

company_id
b-2222
b-2226
b-2230
b-2234
b-2238
b-2242

The output pane shows the following message:

```
1 11:17:31 SELECT DISTINCT company_id FROM transaction WHERE amount > ( SELECT AVG(amount) AS avg... 100 row(s) returned
```

- c) Eliminarán del sistema las empresas que carecen de transacciones registradas, entrega el listado de estas empresas.

Opción 1 - Con IN

The screenshot shows a SQL IDE with a query editor and a result grid. The query is as follows:

```
75 FROM transaction
76 )
77 ;
78
79 -- c) Eliminarán del sistema las empresas que carecen de transacciones registradas, entrega el listado de estas empresas
80 • SELECT id AS empresas_sin_transacciones
81 FROM company
82 WHERE id NOT IN (
83     SELECT
84     DISTINCT company_id
85     FROM transaction
86 )
87
```

The result grid shows the following data:

empresas_sin_transacciones

The output pane shows the following message:

```
1 11:18:57 SELECT id AS empresas_sin_transacciones FROM company WHERE id NOT IN ( SELECT DISTINCT ... 0 row(s) returned
```

(no hay empresas sin transacciones)

Opción 2 - con WHEN EXISTS

The screenshot shows a database IDE with a SQL query editor and a results pane. The query is as follows:

```
98
99 -- CON WHEN EXISTS - c) Eliminarán del sistema las empresas que carecen de transacciones registradas,
100 -- entrega el listado de estas empresas
101 SELECT id
102 FROM company AS c
103 WHERE NOT EXISTS (SELECT 1
104                  FROM transaction AS t
105                  WHERE t.company_id = c.id
106                  )
107
108
109
110
```

The results pane shows a table with one column, 'id', and one row with the value 'NULL'. Below the results pane, the 'Output' section shows the execution log:

#	Time	Action	Message
1	12:39:30	SELECT id AS empresas_sin_transacciones FROM company WHERE id NOT IN (SELECT company_i...	0 row(s) returned
2	12:39:46	SELECT id FROM company AS c WHERE NOT EXISTS (SELECT 1 FROM transaction AS t	0 row(s) returned

(no hay empresas sin transacciones)

Nivel 2

Ejercicio 1

Identifica los cinco días que se generó la mayor cantidad de ingresos en la empresa por ventas. Muestra la fecha de cada transacción junto con el total de las ventas.

estructura datos | datos_introduir_sprint2 | sprint_2 | Ejercizi_subqueries_3_conAI | transactions.company | transactions.transaction

Find | Done

```

112 -- Identifica los cinco días que se generó la mayor cantidad de ingresos en la empresa por ventas.
113 -- Muestra la fecha de cada transacción junto con el total de las ventas.
114 • SELECT
115     CAST(timestamp AS DATE) AS date, -- opcion sin CAST: DATE(timestamp) AS date,
116     SUM(amount) AS ingresos
117 FROM transaction
118 GROUP BY date
119 ORDER BY ingresos DESC
120 LIMIT 5;
121
122 -- Ejercicio 2
123 -- ¿Cuál es la media de ventas por país? Presenta los resultados ordenados de mayor a menor medio
124 • SELECT

```

Result Grid | Filter Rows: | Exports: | Wrap Cell Content: | Read Only

date	ingresos
2022-12-13	14337.44
2019-11-18	13591.32
2023-02-20	13332.59
2017-12-20	13318.43
2019-03-18	12680.95

Result 22 x

Output

Action Output

#	Time	Action	Message
1	12:50:59	SELECT CAST(timestamp AS DATE) AS date, -- opcion sin CAST: DATE(timestamp) AS date, SUM(...	5 row(s) returned

Ejercicio 2

¿Cuál es la media de ventas por país? Presenta los resultados ordenados de mayor a menor medio.

estructura dades dades_introduir_sprint2 sprint_2*

Don't Limit

```

99 GROUP BY t.timestamp
100 ORDER BY ingresos DESC
101 LIMIT 5;
102
103 -- Ejercicio 2
104 -- ¿Cuál es la media de ventas por país? Presenta los resultados ordenados de mayor a menor medio
105 • SELECT
106     c.country,
107     AVG(t.amount) AS avg_ingresos
108 FROM company AS c
109 INNER JOIN transaction AS t
110     ON c.id = t.company_id
111 GROUP BY country
112 ORDER BY avg_ingresos DESC;

```

Set limit for number of rows returned by queries.
Workbench will automatically add the LIMIT clause with the configured number of rows to SELECT queries.

Result Grid

country	avg_ingresos
Australia	265.190742
United States	264.977877
Belgium	261.153042
Germany	260.841391
Ireland	260.644761
Spain	260.468125
France	259.979185

Result 26 x

Output

Action Output

#	Time	Action	Message
1	12:51:51	SELECT c.country, AVG(t.amount) AS avg_ingresos FROM company AS c INNER JOIN transaction ...	15 row(s) returned

Ejercicio 3

En tu empresa, se plantea un nuevo proyecto para lanzar algunas campañas publicitarias para hacer competencia a la compañía “Non Institute”. Para ello, te piden la lista de todas las transacciones realizadas por empresas que están ubicadas en el mismo país que esta compañía.

- a) Muestra el listado aplicando JOIN y subconsultas.

estructura dades dades_introduir_sprint2 sprint_2*

```

114 /*Ejercicio 3
115 En tu empresa, se plantea un nuevo proyecto para lanzar algunas campañas publicitarias para hacer competencia
116 a la compañía "Non Institute".
117 Para ello, te piden la lista de todas las transacciones realizadas por empresas
118 que están ubicadas en el mismo país que esta compañía.
119 */
120
121 -- Muestra el listado aplicando JOIN y subconsultas
122 SELECT t.*
123 FROM transaction AS t
124 INNER JOIN company AS c
125     ON c.id = t.company_id
126 WHERE c.country IN (SELECT country
127                     FROM company
128                     WHERE company_name = "Non Institute"
129                     )
130
131

```

Result Grid

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
008629B4-C9A9-406C-A3D2-71FDA47BC546	CcS-7063	b-2246	2482	45.7666	4.83048	2015-07-30 12:12:42	486.44	0
00B72BA4-54A3-4B8E-813F-2D57535AA17A	CcS-8475	b-2246	3894	55.6212	-3.7546	2017-10-26 22:08:26	414.06	0
01F075B1-D7AE-4D02-AAD9-5FFD72A43F3C	CcS-8700	b-2246	4119	55.856	-3.15783	2018-01-27 13:44:36	103.73	0
023FFCE8-E618-4938-BF56-C8DF80540ADD	CcS-7816	b-2246	3235	46.3568	1.82755	2016-12-19 11:53:45	219.28	0
02683BEB-EF91-4564-957B-D6F1662AB7C5	CcS-9471	b-2246	4890	42.1332	12.396	2017-01-10 21:09:29	326.87	0

Result 37 x Read Only

Output

Action Output

#	Time	Action	Message
1	13:23:25	SELECT t.* FROM transaction AS t INNER JOIN company AS c ON c.id = t.company_id WHERE c.cou...	13776 row(s) returned

b) Muestra el listado aplicando solo subconsultas.

estructura dades dades_introduir_sprint2 sprint_2*

```

131
132 -- Muestra el listado aplicando solo subconsultas
133 • SELECT t.*
134 FROM transaction AS t
135 WHERE t.company_id IN (SELECT c.id
136 FROM company AS c
137 WHERE c.country IN (SELECT c.country
138 FROM company AS c
139 WHERE c.company_name = "Non Institute"
140 )
141 )
142
143
144
145
146
147

```

Result Grid

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
008629B4-C9A9-406C-A3D2-71FDA47BC546	CcS-7063	b-2246	2482	45.7666	4.83048	2015-07-30 12:12:42	486.44	0
00B72BA4-54A3-4B8E-B13F-2D57535AA17A	CcS-8475	b-2246	3894	55.6212	-3.7546	2017-10-26 22:08:26	414.06	0
01F075B1-D7AE-4D02-AAD9-5FFD72A43F3C	CcS-8700	b-2246	4119	55.856	-3.15783	2018-01-27 13:44:36	103.73	0
023FFCE8-E618-4938-BF56-C8DF80540ADD	CcS-7816	b-2246	3235	46.3568	1.82755	2016-12-19 11:53:45	219.28	0
02683BEB-EF91-4564-957B-D6F1662AB7C5	CcS-9471	b-2246	4890	42.1332	12.396	2017-01-10 21:09:29	326.87	0

Result 38 x Read Only

Output

Action Output

#	Time	Action	Message
1	13:23:25	SELECT t.* FROM transaction AS t INNER JOIN company AS c ON c.id = t.company_id WHERE c.cou...	13776 row(s) returned
2	13:24:10	SELECT t.* FROM transaction AS t INNER JOIN company AS c ON c.id = t.company_id WHERE c.cou...	13776 row(s) returned

Nivel 3

Ejercicio 1

Presenta el nombre, teléfono, país, fecha y amount, de aquellas empresas que realizaron transacciones con un valor comprendido entre 350 y 400 euros y en alguna de estas fechas: 29 de abril de 2015, 20 de julio de 2018 y 13 de marzo de 2024.

Ordena los resultados de mayor a menor cantidad.

estructura dades dades_introduir_sprint2 sprint_2* Ejercizi_subqueries_3_conAl transactions.company transactions.transaction

Find

```

166 -- Ejercicio 1
167 -- Presenta el nombre, teléfono, país, fecha y amount, de aquellas empresas que realizaron transacciones
168 -- con un valor comprendido entre 350 y 400 euros y en alguna de estas fechas:
169 -- 29 de abril de 2015, 20 de julio de 2018 y 13 de marzo de 2024.
170 -- Ordena los resultados de mayor a menor cantidad.
171 • SELECT
172     c.company_name,
173     c.phone,
174     c.country,
175     CAST(t.timestamp AS DATE) AS date,
176     t.amount
177 FROM company AS c
178 INNER JOIN transaction AS t
179     ON c.id = t.company_id
180 WHERE t.amount BETWEEN 350 AND 400
181 HAVING date IN ("2015-04-29", "2018-07-20", "2024-03-13")
182 ORDER BY amount DESC;

```

Result Grid

company_name	phone	country	date	amount
Aliquam PC	01 45 73 52 16	Germany	2024-03-13	399.84
Auctor Mauris Vel LLP	08 09 28 74 14	United States	2018-07-20	399.51
At Pedo Corp.	06 14 48 33 15	Italy	2015-04-29	390.69
Aliquam PC	01 45 73 52 16	Germany	2024-03-13	388.29
Ordi Adipiscing Limited	03 18 00 77 81	United Kingdom	2018-07-20	373.71

Result 27 x

Output

Action Output

#	Time	Action	Message
1	13:15:21	SELECT c.company_name, c.phone, c.country, CAST(t.timestamp AS DATE) AS date, t.amou...	8 row(s) returned

Ejercicio 2

Necesitamos optimizar la asignación de los recursos y dependerá de la capacidad operativa que se requiera, por lo que te piden la información sobre la cantidad de transacciones que realizan las empresas, pero el departamento de recursos humanos es exigente y quiere un listado de las empresas en las que especifiques si tienen más de 400 transacciones o menos.

```
estructura dades      dades_introduir_sprint2      sprint_2* x
[Icons] Don't Limit

164 -- Ejercicio 2
165 -- Necesitamos optimizar la asignación de los recursos y dependerá de la capacidad operativa que se requiera,
166 -- por lo que te piden la información sobre la cantidad de transacciones que realizan las empresas,
167 -- pero el departamento de recursos humanos es exigente y quiere un listado de las empresas en las que especifiques
168 -- si tienen más de 400 transacciones o menos.
169 SELECT
170     c.id,
171     c.company_name,
172     (CASE WHEN COUNT(t.id) >= 400 THEN "400 transacciones o mas"
173     ELSE "menos de 400 transacciones"
174     END) AS n_transacciones
175 FROM company AS c
176 INNER JOIN transaction AS t
177     ON c.id = t.company_id
178 GROUP BY c.id, c.company_name
179 ORDER BY n_transacciones DESC; -- no obligatorio

Result Grid
[Icons] Filter Rows: [ ] Export: [ ] Wrap Cell Content: [ ]

| id | company_name | n_transacciones |
|---|---|---|
| b-2470 | Nec Luctus LLC | menos de 400 transacciones |
| b-2286 | Fringilla LLC | menos de 400 transacciones |
| b-2362 | Lorem Eu Incorporated | menos de 400 transacciones |
| b-2222 | Ac Fermentum Incorporated | 400 transacciones o mas |
| b-2226 | Magna A Neque Industries | 400 transacciones o mas |
| b-2230 | Fusce Corp. | 400 transacciones o mas |
| b-2224 | Consectetur Inc Incorporated | 400 transacciones o mas |

Result 46 x [ ] Read Only

Output
[Icons] Action Output [ ]

# | Time | Action | Message
1 | 13:59:57 | SELECT c.id, c.company_name, (CASE WHEN COUNT(t.id) >= 400 THEN "400 transacciones o ... | 100 row(s) returned
```