

Xavier Bravo

SPRINT 2

Nociones básicas de SQL



07/11/2024

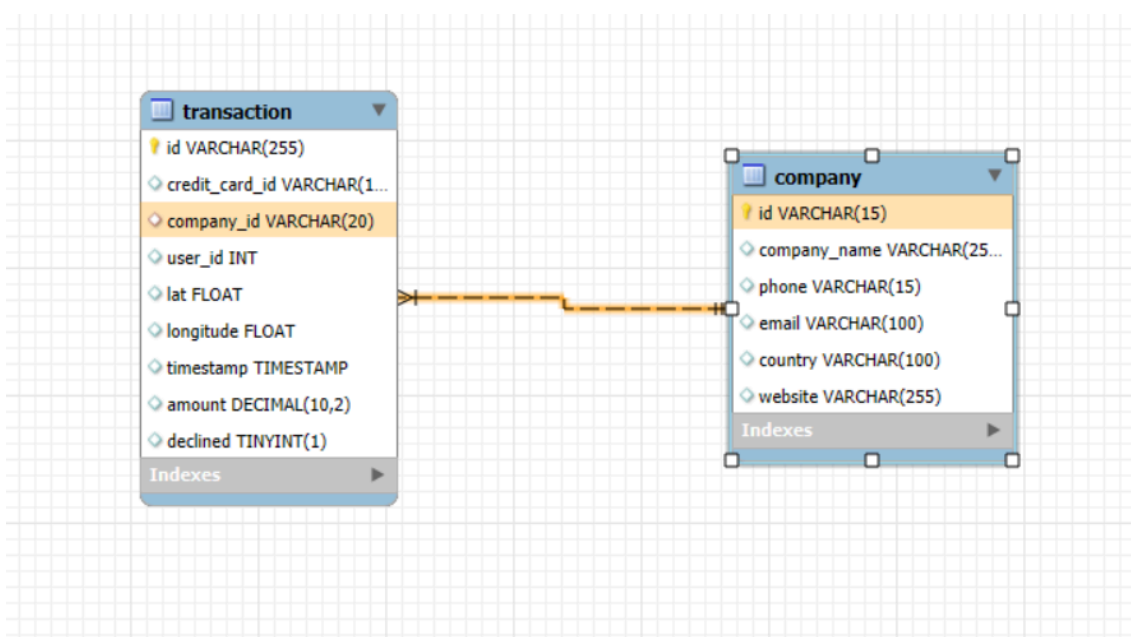
NIVEL 1

Ejercicio 1

El diagrama ER (Entidad -Relación) ayuda a visualizar la estructura de la base de datos y sus relaciones, proporcionando una comprensión clara de cómo se organizan y almacenan los datos. Además, su representación gráfica facilita el análisis relacional, indicando si las relaciones son de uno a uno, uno a muchos, o viceversa. También nos muestra el tipo de variable de cada columna.

La base de datos transactions contiene dos tablas principales: **company** (tabla principal) y **transaction** (tabla relación) . La tabla **transaction** almacena información detallada sobre las transacciones realizadas, incluyendo campos como el usuario (**user_id**), el monto (**amount**), la ubicación (**lat** y **longitude**), la fecha y hora (**timestamp**), y un indicador de rechazo (**declined**). Por otro lado, la tabla **company** contiene información sobre las empresas, como el nombre (**company_name**), el teléfono (**phone**), el correo electrónico (**email**), el país (**country**), y el sitio web (**website**).

La relación entre ambas tablas se establece a través del campo **company_id** en la tabla **transaction**, que actúa como clave foránea apuntando al campo **id** en la tabla **company**. Esto crea una relación de uno a muchos, en la cual una empresa puede tener múltiples transacciones asociadas. Gracias a esta conexión, cada transacción puede vincularse fácilmente con la empresa que la realizó.



Ejercicio 2

2.1) Listado de países que hacen compras:

USE transactions;

SELECT

 DISTINCT country

FROM company AS c

LEFT JOIN transaction AS t

ON c.id = t.company_id;

Establecemos la cláusula **USE** para definir el uso de las consultas sobre la base de datos especificada después de **USE**. El código obtiene una lista única de países (**country**) de la tabla **company**, incluyendo aquellos sin transacciones en la tabla **transaction**.

```
37 • SELECT DISTINCT country
38 FROM transaction AS t
39 LEFT JOIN company AS c
40 ON t.company_id = c.id;
```

Result Grid				Filter Rows:	Export:	Wrap Cell Content:
country						
▶	Germany					
	Australia					
	United States					
	New Zealand					
	Norway					
	United Kingdom					
	Italy					
	Belgium					
	Sweden					
	Ireland					
	China					
	...					
#	Time	Action		Message		
63	20:34:09	SELECT DISTINCT country FROM transaction AS t LEFT JOIN company AS c ON t.company_id = c.id LIMIT ...		16 row(s) returned		

2.2) Desde cuántos países se realizan compras:

SELECT

COUNT(DISTINCT country) AS num_paises_compradores

FROM company AS c

LEFT JOIN transaction AS t

ON c.id = t.company_id;

Esta consulta cuenta la cantidad de países distintos (**num_purchasing_countries**) en los que hay empresas de la tabla `company`, incluyendo aquellos sin transacciones en la tabla `transaction`. **COUNT DISTINCT** country devuelve un conteo de países únicos en los que hay empresas que han realizado compras. Podemos obtener los países de estas empresas mediante el **LEFT JOIN** (unión a la izquierda) de valores coincidentes, donde la unión se realiza con la cláusula **ON**, especificando las columnas de unión.

```
45 • SELECT COUNT(DISTINCT country) AS num_purchasing_countries
46 FROM transaction AS t
47 LEFT JOIN company AS c
48 ON t.company_id = c.id;
```

Result Grid | | Filter Rows: | Export: | Wrap Cell Content:

	num_purchasing_countries
▶	15

#	Time	Action	Message
✓ 66	20:38:35	SELECT COUNT(DISTINCT country) AS num_paises_compradores FROM transaction AS t LEFT JOIN compa...	1 row(s) returned

2.3) Identificar la compañía con la media de ventas más alta:

```
SELECT
    company_name,
    AVG(amount)AS max_avg_sale
FROM company AS c
LEFT JOIN transaction AS t
ON c.id = t.company_id
WHERE declined = 0
GROUP BY company_name
ORDER BY max_avg_saleDESC
LIMIT 1;
```

Esta consulta obtiene el nombre de la empresa (**company_name**) junto con la cantidad media de transacciones (**max_avg_sale**) realizadas por cada una. Se utiliza **AVG(amount)** para calcular el promedio del monto de las transacciones por empresa. La unión **LEFT JOIN** asegura que se incluyan todas las empresas, incluso aquellas sin transacciones. Los resultados se agrupan por el nombre de la empresa (**GROUP BY company_name**), se ordenan de mayor a menor según la cantidad media de las transacciones (**ORDER BY max_avg_sale DESC**) y se limita el resultado a la empresa con el mayor promedio de transacciones (**LIMIT 1**).

```
53 • SELECT
54     company_name,
55     ROUND(AVG(amount), 2) max_avg_sale
56 FROM company AS c
57 LEFT JOIN transaction AS t
58 ON c.id = t.company_id
59 WHERE declined = 0
60 GROUP BY company_name
61 ORDER BY max_avg_sale DESC
62 LIMIT 1;
```

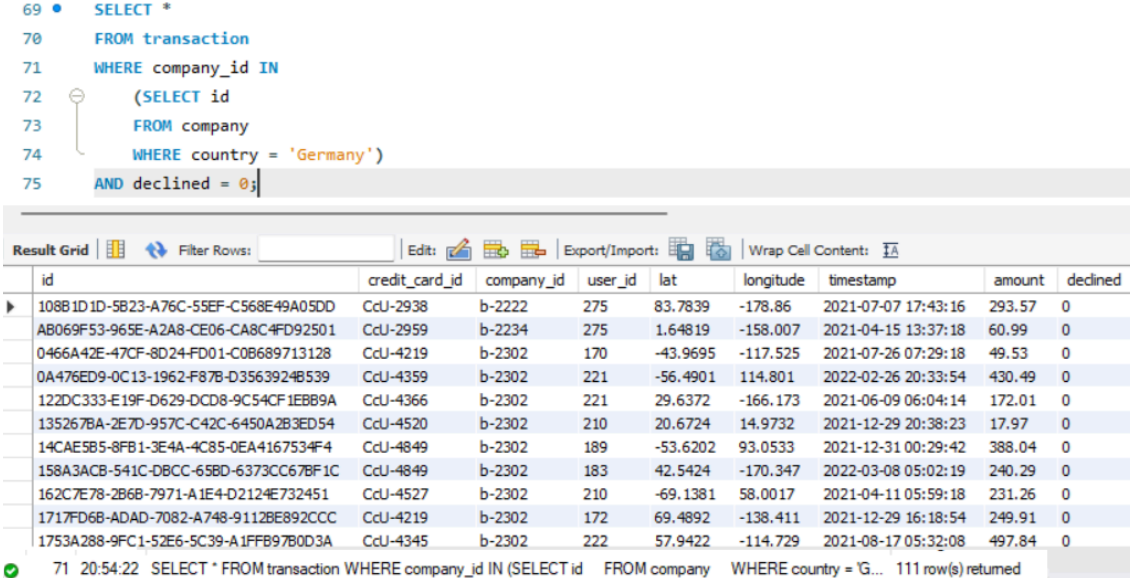
Result Grid				Filter Rows:	Export:	Wrap Cell Content:	Fetch rows:
company_name		max_avg_sale					
Eget Ipsum Ltd		481.86					
#	Time	Action		Message			
69	20:46:28	SELECT company_name, ROUND(AVG(amount), 2) max_avg_sale FROM company AS c LEFT JOIN tra...		1 row(s) returned			

Ejercicio 3

3.1) Mostrar todas las transacciones realizadas por empresas de Alemania:

```
SELECT *  
FROM transaction  
WHERE company_id IN  
    (SELECT id  
     FROM company  
     WHERE country = 'Germany')  
AND declined = 0;
```

Esta consulta selecciona todas las columnas de la tabla transaction para aquellas transacciones cuyo campo company_id coincide con los id de las empresas que tienen el valor de la columna country igual a 'Germany'. Utiliza la cláusula **WHERE** con un operador **IN**, que filtra las transacciones cuyo company_id está presente en el conjunto de ids que se obtienen a través de la subconsulta. La subconsulta, a su vez, selecciona los ids de la tabla company donde el valor de la columna country es igual a 'Germany'. Además, la consulta principal incluye la condición **AND declined = 0**, que asegura que sólo se seleccionen transacciones que no hayan sido rechazadas (declined = 0).



The screenshot shows a SQL query editor with the following query:

```
69 • SELECT *  
70 FROM transaction  
71 WHERE company_id IN  
72     (SELECT id  
73      FROM company  
74      WHERE country = 'Germany')  
75 AND declined = 0;
```

Below the query editor is a table with the following columns: id, credit_card_id, company_id, user_id, lat, longitude, timestamp, amount, and declined. The table contains 11 rows of data.

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
108B1D1D-5B23-A76C-55EF-C568E49A05DD	CdU-2938	b-2222	275	83.7839	-178.86	2021-07-07 17:43:16	293.57	0
AB069F53-965E-A2A8-CE06-CA8C4FD92501	CdU-2959	b-2234	275	1.64819	-158.007	2021-04-15 13:37:18	60.99	0
0466A42E-47CF-8D24-FD01-C0B689713128	CdU-4219	b-2302	170	-43.9695	-117.525	2021-07-26 07:29:18	49.53	0
0A476ED9-0C13-1962-F87B-D3563924B539	CdU-4359	b-2302	221	-56.4901	114.801	2022-02-26 20:33:54	430.49	0
122DC333-E19F-D629-DCD8-9C54CF1EBB9A	CdU-4366	b-2302	221	29.6372	-166.173	2021-06-09 06:04:14	172.01	0
135267BA-2E7D-957C-C42C-6450A2B3ED54	CdU-4520	b-2302	210	20.6724	14.9732	2021-12-29 20:38:23	17.97	0
14CAE5B5-8FB1-3E4A-4C85-0EA4167534F4	CdU-4849	b-2302	189	-53.6202	93.0533	2021-12-31 00:29:42	388.04	0
158A3ACB-541C-DBCC-65BD-6373CC67BF1C	CdU-4849	b-2302	183	42.5424	-170.347	2022-03-08 05:02:19	240.29	0
162C7E78-2B6B-7971-A1E4-D2124E732451	CdU-4527	b-2302	210	-69.1381	58.0017	2021-04-11 05:59:18	231.26	0
1717FD6B-ADAD-7082-A748-9112BE892CCC	CdU-4219	b-2302	172	69.4892	-138.411	2021-12-29 16:18:54	249.91	0
1753A288-9FC1-52E6-5C39-A1FFB97B0D3A	CdU-4345	b-2302	222	57.9422	-114.729	2021-08-17 05:32:08	497.84	0

At the bottom of the screenshot, a status bar shows the query execution time as 20:54:22 and the number of rows returned as 111 row(s) returned.

3.2) Listado de las empresas que han realizado transacciones por una cantidad superior a la media de todas las transacciones:

```
SELECT  (SELECT company_name
          FROM company
          WHERE id = transaction.company_id) AS company_name,
        amount
FROM transaction
WHERE amount >
      (SELECT AVG(amount)
        FROM transaction)
AND declined = 0
ORDER BY amount DESC;
```

Esta consulta selecciona el nombre de la empresa y el monto de las transacciones cuyo monto es mayor que el promedio de todas las transacciones. Utiliza una subconsulta en el **SELECT** para obtener el nombre de la empresa a partir del **company_id** y otra subconsulta en la cláusula **WHERE** para calcular el promedio de los montos en la tabla transaction, filtrando aquellas transacciones con monto superior al promedio calculado.

```

82 SELECT (SELECT company_name
83         FROM company
84         WHERE id = transaction.company_id) AS company_name,
85        amount
86 FROM transaction
87 WHERE amount > (
88                SELECT AVG(amount)
89                FROM transaction)
90 AND declined = 0
91 ORDER BY amount DESC;

```

company_name	amount
Nunc Interdum Incorporated	499.23
Nunc Interdum Incorporated	497.84
Enim Condimentum Ltd	494.82
Arcu LLP	494.17
Non Institute	492.42
Malesuada PC	492.25
Ut Semper Foundation	492.19
Nunc Interdum Incorporated	487.64
Arcu LLP	486.83
Malesuada PC	486.54
Nunc Interdum Incorporated	485.31

79 21:12:08 SELECT (SELECT company_name FROM company WHERE id = transaction.company_id) AS company... 256 row(s) returned

3.3) Eliminarán del sistema las empresas que no tienen transacciones registradas, entrega el listado de estas empresas:

```

SELECT id,
       (SELECT company_name
        FROM company) AS company_name
FROM company AS c
WHERE NOT EXISTS
      (SELECT company_id
       FROM transaction AS t
       WHERE t.company_id = c.id);

```

Esta consulta selecciona el **id** y el **company_name** de la tabla **company**, pero solo para aquellas empresas que no tienen transacciones asociadas en la tabla **transaction**. La subconsulta dentro del **SELECT** obtiene el nombre de la empresa para cada **id**. La cláusula **WHERE NOT EXISTS** filtra aquellas empresas que no tienen ningún **company_id** correspondiente en la tabla `transaction`, es decir, aquellas empresas sin transacciones.


```

60 • SELECT id,
61      (SELECT company_name
62       FROM company
63       ) AS company_name
64 FROM company
65 WHERE id NOT IN
66      (SELECT company_id
67       FROM transaction);
68

```

Result Grid | Filter Rows: | Export: | Wrap Cell Contents: |

id	company_name
----	--------------

Output

Action Output

#	Time	Action	Message
✓ 11	22:55:28	SELECT id, (SELECT company_name FROM company) AS company_name FROM company WHERE i...	0 row(s) returned

NIVEL 2

EJERCICIO 1

Identifica los 5 días en los que se generaron la cantidad más grande de ingresos por ventas. Muestra la fecha de cada transacción juntamente con el total de las ventas:

Usando Joins

SELECT

DATE_FORMAT(t.timestamp, '%Y-%m-%d') AS date,
SUM(t.amount) AS sales_by_date

FROM transaction t

JOIN company c

ON t.company_id = c.id

WHERE t.declined = 0

GROUP BY DATE_FORMAT(t.timestamp, '%Y-%m-%d')

ORDER BY sales_by_date **DESC**

LIMIT 5;

Esta consulta selecciona la fecha (**date**) y la cantidad (**sales_by_date**) de las transacciones, agrupadas por día. La función **DATE_FORMAT(t.timestamp, '%Y-%m-%d')** se utiliza para extraer solo la fecha (sin la hora) de la columna timestamp de la tabla transaction. Luego, los resultados se filtran para incluir solo aquellas transacciones donde **declined = 0** (es decir, aquellas que no han sido rechazadas). La suma de los **montos** se realiza con **SUM(t.amount)**. Los resultados se agrupan por fecha, ya que la consulta calcula el total de ventas por día. Además, se realiza un **JOIN** entre las tablas **transaction** y **company** usando **company_id**, para asociar cada transacción con su empresa correspondiente. Finalmente, los resultados se ordenan de mayor a menor según el total de ventas por día (**sales_by_date**), y se limita a los 5 primeros registros mediante **LIMIT 5**.

```
111 • SELECT
112     DATE_FORMAT(t.timestamp, '%Y-%m-%d') AS date,
113     SUM(t.amount) AS sales_by_date
114
115 FROM transaction t
116 JOIN company c
117 ON t.company_id = c.id
118 WHERE t.declined = 0
119 GROUP BY DATE_FORMAT(t.timestamp, '%Y-%m-%d')
120 ORDER BY sales_by_date DESC
121 LIMIT 5;
```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:	Fetch rows:
	date	sales_by_date				
▶	2021-12-20	1532.36				
	2021-04-22	1397.96				
	2021-05-09	1344.37				
	2022-02-26	1337.62				
	2021-03-29	1325.12				

81 21:41:46 SELECT DATE_FORMAT(t.timestamp, '%Y-%m-%d') AS date, SUM(t.amount) AS sales_by_date ... 5 row(s) returned

```
DATE_FORMAT(timestamp, '%Y-%m-%d') AS date,
SUM(amount) AS sales_by_date
```

```
FROM transaction
```

```
WHERE declined = 0
```

```
GROUP BY DATE_FORMAT(timestamp, '%Y-%m-%d'),
```

```
ORDER BY sales_by_date DESC
```

```
LIMIT 5;
```

Esta consulta selecciona la **fecha (date)**, el **company_id**, y la **suma del monto (sales_by_date)** de las transacciones, agrupadas por **día y empresa**. La función **DATE_FORMAT(timestamp, '%Y-%m-%d')** se utiliza para extraer solo la **fecha** (sin la hora) de la columna timestamp. Luego, los resultados se agrupan por **fecha y company_id**, ya que la consulta calcula las ventas totales por **empresa y día**. La **suma de los montos** se realiza con **SUM(amount)**. Los resultados se **ordenan de mayor a menor** según el total de ventas por día (**sales_by_date**), y se limita a los **5 primeros registros** mediante **LIMIT 5**.

```

111 • SELECT
112     DATE_FORMAT(t.timestamp, '%Y-%m-%d') AS date,
113     SUM(t.amount) AS sales_by_date
114
115 FROM transaction t
116 JOIN company c
117 ON t.company_id = c.id
118 WHERE t.declined = 0
119 GROUP BY DATE_FORMAT(t.timestamp, '%Y-%m-%d')
120 ORDER BY sales_by_date DESC
121 LIMIT 5;

```

date	sales_by_date
2021-12-20	1532.36
2021-04-22	1397.96
2021-05-09	1344.37
2022-02-26	1337.62
2021-03-29	1325.12

81 21:41:46 SELECT DATE_FORMAT(t.timestamp, '%Y-%m-%d') AS date, SUM(t.amount) AS sales_by_date ... 5 row(s) returned

EJERCICIO 2

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

Usando Joins:

```

SELECT c.country,
       ROUND(AVG(amount),2) AS avg_by_country
FROM transaction AS t
LEFT JOIN company AS c
WHERE declined = 0
ON t.company_id = c.id
GROUP BY c.country
ORDER BY avg_by_country DESC;

```

Esta consulta obtiene el país (**country**) y el promedio de las transacciones (**avg_by_country**) por país. Se utiliza un **LEFT JOIN** para unir las tablas **transaction** y **company** basándose en la coincidencia del **company_id** de **transaction** con el **id** de **company**. El promedio del monto se calcula con **AVG(amount)** y los resultados se agrupan por país (**GROUP BY c.country**).

Finalmente, los países se ordenan de mayor a menor según el promedio de las transacciones (**ORDER BY *media_por_pais* DESC**).

```
137 • SELECT c.country,  
138         ROUND(AVG(amount),2) AS avg_by_country  
139     FROM transaction AS t  
140     LEFT JOIN company AS c  
141     ON t.company_id = c.id  
142     WHERE declined = 0  
143     GROUP BY c.country  
144     ORDER BY avg_by_country DESC;
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	country	avg_by_country
▶	United States	287.53
	Ireland	285.83
	Sweden	276.67
	United Kingdom	271.77
	Canada	258.28
	Belgium	255.22
	Norway	251.11
	Italy	243.34
	Germany	242.24
	Netherlands	240.94
	China	222.24

✓

84

21:45:38

SELECT c.country, ROUND(AVG(amount),2) AS avg_by_country FROM transaction AS t LEFT JOIN compan...

16 row(s) returned

Usando subconsultas:

```
SELECT    country,  
          (SELECT AVG(amount) AS media  
            FROM transaction AS t  
            WHERE company_id IN  
                  (SELECT id  
                    FROM company  
                    WHERE country = c.country)  
          ) AS media_por_pais  
FROM company AS c  
GROUP BY country  
ORDER BY media_por_pais DESC;
```

La consulta selecciona el país (**country**) y el promedio de las transacciones (**media_por_pais**) para cada país. La subconsulta calcula el promedio de los montos (**AVG(amount)**) de la tabla **transaction**, pero solo para aquellas transacciones cuyos **company_id** coinciden con los id de las empresas de ese país. La subconsulta filtra las

empresas por el **country** de la tabla company, utilizando la relación entre ambas tablas. Luego, los resultados se agrupan por país (**GROUP BY country**) y se ordenan de mayor a menor según el promedio de las transacciones (**ORDER BY media_por_país DESC**).

```
148 • SELECT country,
149       (SELECT ROUND(AVG(amount),2) AS average
150        FROM transaction AS t
151        WHERE declined = 0
152        AND company_id IN
153          (SELECT id
154           FROM company
155           WHERE country = c.country)
156        ) AS avg_by_country
157 FROM company AS c
158 GROUP BY country
159 ORDER BY avg_by_country DESC;
```

country	avg_by_country
United States	287.53
Ireland	285.83
Sweden	276.67
United Kingdom	271.77
Canada	258.28
Belgium	255.22
Norway	251.11
Italy	243.34
Germany	242.24
Netherlands	240.94
China	222.24

86 21:49:22 SELECT country, (SELECT ROUND(AVG(amount),2) AS average FROM transaction AS t WHERE ... 16 row(s) returned

EJERCICIO 3

Usando joins

Se pide la lista de todas las transacciones realizadas por empresas que están situadas en el mismo país que esta compañía.

```
SELECT t.*
FROM transaction t
LEFT JOIN company c
ON t.company_id = c.id
WHERE country = 'United Kingdom'
      AND c.id != 'b-2618'
      AND declined = 0;
```

Esta consulta selecciona **todos los campos** de la tabla transaction (t), como id, credit_card_id, company_id, user_id, lat, longitude, timestamp, amount, y declined. La tabla transaction se une con la tabla company usando un **LEFT JOIN**, vinculando el company_id de transaction con el id de company. La consulta filtra los resultados para incluir solo las transacciones de empresas ubicadas en el país 'United Kingdom' (**WHERE country = 'United Kingdom'**). Esto asegura que solo se muestren las

transacciones de las empresas en ese país. Además, se excluye la transacción cuyo company_id sea 'b-2618' (**AND** c.id != 'b-2618') y se filtran sólo las transacciones que no han sido rechazadas (**AND** declined = 0), es decir, las transacciones aceptadas.

```

171 • SELECT
172     t.*
173 FROM transaction t
174 LEFT JOIN company c
175 ON t.company_id = c.id
176 WHERE country = 'United Kingdom'
177     AND c.id != 'b-2618'
178     AND declined = 0;

```

Result Grid									
Filter Rows:									
Export:									
Wrap Cell Content:									
	id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
▶	2B928E1C-EC14-A760-0A75-871477649D6A	CcU-2980	b-2246	275	-41.0496	161.685	2021-08-10 08:14:49	383.73	0
	4334349E-CEB0-3D68-A4D4-FEB7718A1ACE	CcU-3092	b-2310	275	-20.4859	150.87	2021-05-03 22:37:23	458.74	0
	1479B3D2-B7BA-C7BB-4CE3-8D7C2DE85ABB	CcU-2994	b-2326	133	66.2672	172.399	2021-08-09 00:58:07	309.45	0
	152598C2-029D-D684-4B66-91EDF393EBFF	CcU-2994	b-2326	126	-67.0189	-141.672	2021-07-05 03:10:00	395.43	0
	1B636B58-A2E8-7C69-D9C9-C54535DAFD3B	CcU-2994	b-2326	131	70.2543	-13.1336	2021-07-06 08:48:46	195.06	0
	20418DE5-B804-BE9B-8D7A-A95C1BFDBF5C	CcU-2994	b-2326	126	-79.1145	1.51481	2022-01-03 15:59:29	479.52	0
	239B8576-6C0E-137A-C2F6-3180A188A2D3	CcU-2994	b-2326	126	23.6174	137.222	2021-08-26 06:04:05	43.90	0
	267C4A86-7BA7-1C5E-0718-2824983C87DD	CcU-2994	b-2326	126	-17.5259	104.915	2021-10-01 21:08:53	122.63	0
	3142C93E-B3B7-49E4-EE2D-29CA834B198D	CcU-2994	b-2326	126	-67.8476	-119.978	2021-04-06 17:24:44	91.59	0
	3578688E-7B1D-B887-3BC7-20B8673AA31E	CcU-2994	b-2326	126	87.0665	-22.7339	2021-07-26 22:59:24	303.60	0
	360C7814-F7AF-B43A-0946-AB38D2683C86	CcU-2994	b-2326	116	-7.93005	-79.0733	2021-08-21 10:19:58	494.82	0
✓	88 22:31:01 SELECT t.* FROM transaction t LEFT JOIN company c ON t.company_id = c.id WHERE country = 'United Ki... 63 row(s) returned								

Usando subconsultas

```

SELECT *
FROM transaction
WHERE company_id IN
    (SELECT id
    FROM company
    WHERE country = 'United Kingdom')
    AND id != 'b-2618'
    AND declined = 0);

```

La consulta selecciona todas las columnas de la tabla transaction y filtra las filas en las que el company_id coincide con el id de la tabla company. Además, utiliza un **WHERE** en la subconsulta para filtrar aquellas empresas cuyo país (**country**) sea igual a 'United Kingdom', asegurando que solo se muestren las transacciones correspondientes a empresas ubicadas en ese país. Dentro de la subconsulta, también se aplica una

condición `id != 'b-2618'` para excluir una empresa específica, y se filtran las empresas cuyo campo `declined` es igual a 0, es decir, las empresas cuyas transacciones no han sido rechazadas. Así, la consulta devuelve sólo las transacciones de empresas ubicadas en el Reino Unido, excepto la empresa con el id específico y con transacciones aceptadas.

```

182 • SELECT *
183 FROM transaction
184 WHERE company_id IN
185     (SELECT id
186 FROM company
187 WHERE country = 'United Kingdom'
188 AND id != 'b-2618'
189 AND declined = 0);

```

	id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
▶	2B928E1C-EC14-A760-0A75-871477649D6A	CcU-2980	b-2246	275	-41.0496	161.685	2021-08-10 08:14:49	383.73	0
	4334349E-CEB0-3D68-A4D4-FEB7718A1ACE	CcU-3092	b-2310	275	-20.4859	150.87	2021-05-03 22:37:23	458.74	0
	1479B3D2-87BA-C7BB-4CE3-8D7C2DE85ABB	CcU-2994	b-2326	133	66.2672	172.399	2021-08-09 00:58:07	309.45	0
	152598C2-029D-D684-4B66-91EDF393EBFF	CcU-2994	b-2326	126	-67.0189	-141.672	2021-07-05 03:10:00	395.43	0
	1B636B58-A2E8-7C69-D9C9-C54535DAFD3B	CcU-2994	b-2326	131	70.2543	-13.1336	2021-07-06 08:48:46	195.06	0
	20418DE5-B804-8E9B-8D7A-A95C1BFD8F5C	CcU-2994	b-2326	126	-79.1145	1.51481	2022-01-03 15:59:29	479.52	0
	239B8576-6C0E-137A-C2F6-3180A188A2D3	CcU-2994	b-2326	126	23.6174	137.222	2021-08-26 06:04:05	43.90	0
	267C4A86-7BA7-1C5E-0718-2824983C87DD	CcU-2994	b-2326	126	-17.5259	104.915	2021-10-01 21:08:53	122.63	0
	3142C93E-83B7-49E4-EE2D-29CA834B198D	CcU-2994	b-2326	126	-67.8476	-119.978	2021-04-06 17:24:44	91.59	0
	3578688E-7B1D-B887-8BC7-20B8673AA31E	CcU-2994	b-2326	126	87.0665	-22.7339	2021-07-26 22:59:24	303.60	0
	360C7814F7AF-B43A-0946-AB38D2683C86	CcU-2994	b-2326	116	-7.93005	-79.0733	2021-08-21 10:19:58	494.82	0

91 22:44:48 SELECT * FROM transaction WHERE company_id IN (SELECT id FROM company WHERE country = 'United... 63 row(s) returned

NIVEL 3

Ejercicio 1

Muestra el `company_name`, `phone`, `country`, fecha (`timestamp`) y cantidad (`amount`) de aquellas empresas que realizaron transacciones con un valor comprendido entre 100 y 200 euros y en alguna de estas fechas: 29 de abril de 2021, 20 de julio de 2021 y 13 de marzo de 2022. Ordena los resultados de mayor a menor cantidad.

```

SELECT company_name,
       phone,
       country,
       DATE_FORMAT(timestamp, '%Y-%m-%d') AS date,
       amount
FROM company AS c
LEFT JOIN transaction AS t
ON c.id = t.company_id
WHERE DATE_FORMAT(timestamp, '%Y-%m-%d')
      IN ('2021-04-29', '2021-07-20', '2022-03-13')
AND amount BETWEEN 100 AND 200

```

ORDER BY date;

La consulta selecciona el nombre de la empresa (**company_name**), teléfono (**phone**), país (**country**), fecha (formateada a 'YYYY-MM-DD') y cantidad (**amount**) de la tabla company, uniéndose con la tabla transaction mediante un **LEFT JOIN**. Se filtran los resultados para incluir sólo las transacciones que ocurrieron en fechas específicas: el 29 de abril de 2021, 20 de julio de 2021 y 13 de marzo de 2022. Además, se seleccionan sólo aquellas transacciones cuyo monto se encuentra entre 100 y 200. Finalmente, los resultados se ordenan por fecha.

```
199 • SELECT company_name,  
200         phone,  
201         country,  
202         DATE_FORMAT(timestamp, '%Y-%m-%d') AS date,  
203         amount  
204 FROM company AS c  
205 LEFT JOIN transaction AS t  
206 ON c.id = t.company_id  
207 WHERE DATE_FORMAT(timestamp, '%Y-%m-%d') IN ('2021-04-29', '2021-07-20', '2022-03-13')  
208        AND amount BETWEEN 100 AND 200  
209 ORDER BY date;
```

	company_name	phone	country	date	amount
▶	Enim Condimentum Ltd	09 55 51 66 25	United Kingdom	2021-04-29	149.89
	Nunc Interdum Incorporated	05 18 15 48 13	Germany	2021-04-29	111.51
	Lorem Eu Incorporated	01 83 66 62 07	Canada	2021-07-20	133.39
	Interdum Feugiat Sed Associates	04 88 40 32 52	United Kingdom	2021-07-20	164.86
	Nunc Interdum Incorporated	05 18 15 48 13	Germany	2022-03-13	164.32

✓ 93 22:54:40 SELECT company_name, phone, country, DATE_FORMAT(timestamp, '%Y-%m-%d') AS date, a... 5 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 una lista de las empresas donde especifiques si tienen más de 4 transacciones o menos.

```
SELECT company_id,  
       COUNT(id) AS num_transactions,  
       CASE WHEN COUNT(id) > 4 THEN 'más de 4' ELSE 'menos de 4' END AS  
performance  
FROM transaction  
GROUP BY company_id  
ORDER BY num_transactions DESC;
```

La consulta cuenta el número de transacciones por empresa, utilizando el `company_id` como clave de agrupación. Además, agrega una columna que clasifica el **performance** como 'más de 4' o 'menos de 4' según el número de transacciones realizadas, utilizando la cláusula **CASE**. La cláusula **GROUP BY** agrupa los resultados por **company_id** y la cláusula **COUNT** se utiliza para contar las transacciones. Finalmente, la cláusula **ORDER BY** ordena los resultados de forma descendente según el conteo de transacciones (**num_transactions**).

```
42 • SELECT company_id,  
43         COUNT(id) AS num_transactions,  
44         CASE WHEN COUNT(id) > 4 THEN 'más de 4' ELSE 'menos de 4' END AS rendimiento  
45 FROM transaction  
46 GROUP BY company_id  
47 ORDER BY num_transactions DESC;
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
company_id	num_transactions	rendimiento	
b-2302	105	más de 4	
b-2346	59	más de 4	
b-2326	57	más de 4	
b-2278	56	más de 4	
b-2362	54	más de 4	