

SPRINT 2 – Data Analytics

A continuación, se presentan los resultados del Sprint #2 de la especialización Data Analytics de IT Academy, realizados por Rossemary Castellanos la semana del 15/09/2025. Se realizaron los 3 niveles del Sprint.

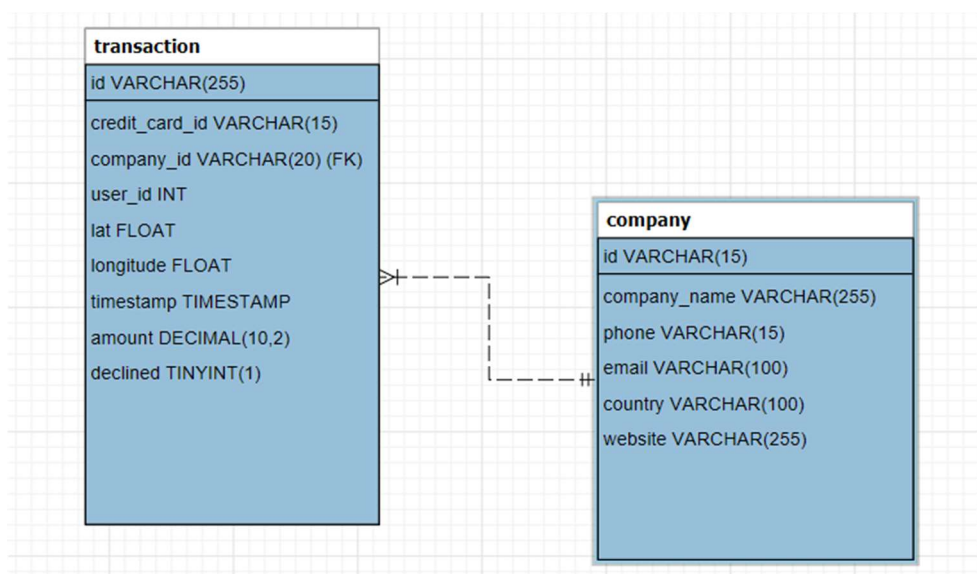
NIVEL 1

Ejercicio 1

A partir de los documentos adjuntos (estructura_datos y datos_introducir), importa las dos tablas. Muestra las principales características del esquema creado y explica las diferentes tablas y variables que existen. Asegúrate de incluir un diagrama que ilustre la relación entre las distintas tablas y variables.

Los archivos referidos en el enunciado contenían: uno la estructura de la base de datos, así como la de sus tablas, incluidas los formatos de las variables y la relaciones entre las tablas; y el otro archivo todos los registros a introducir en ambas tablas.

En la figura anexa se presenta el diagrama entidad-relación generado para la base de datos cargada de nombre **transactions**, la misma contiene 2 tablas: **company** y **transaction**.



11. Diagrama Entidad - Relación para base de datos transactions

La tabla **company** presenta los datos generales de las empresas que venden:

- número identificador del código de la empresa como clave primaria y única (id),
- nombre de la empresa (company_name),
- referencias de contacto como teléfono (phone) y correo electrónico (email), e
- información del país de ubicación (country) y su página web (website).

La tabla **transaction** presenta todas las operaciones (transacciones) de ventas realizadas por las empresas listadas en la tabla company:

- la identificación de la transacción como clave primaria y única (id),
- numero identificador de la tarjeta de crédito utilizada (credit_card_id),
- identificador de la compañía que vende (company_id) como clave foránea,
- identificador de usuario (user_id),
- ubicación referida como latitud y longitud (lat, longitude) respectivamente,
- el momento en que se realiza la operación (fecha y hora = timestamp),
- el importe de la venta (amount), y
- verificación de proceder o no con la venta (declined).

La clave que vincula ambas tablas es el código de empresa siendo clave primaria en la tabla company (id) y clave foránea en la tabla transaction (company_id).

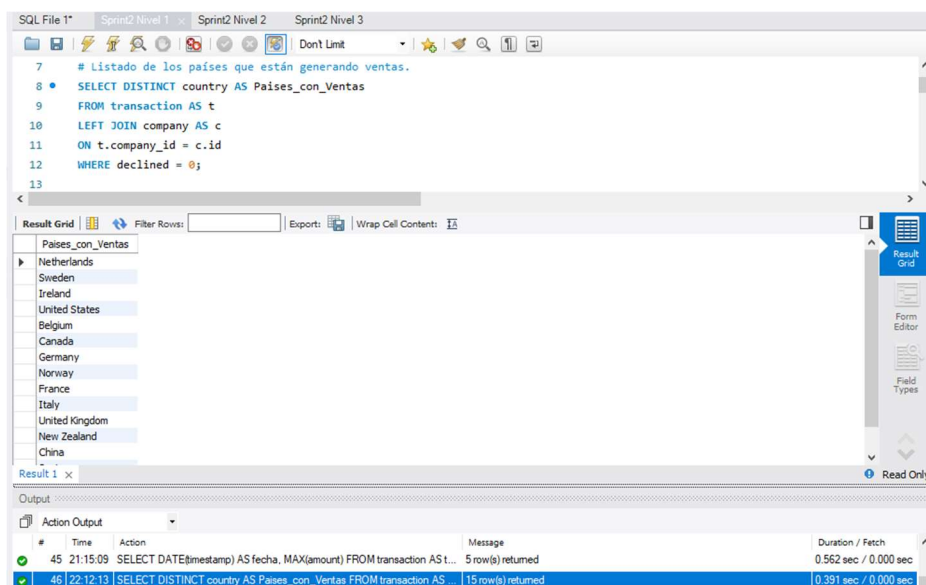
La relación entre las tablas es de 1: N, de 1 a muchos. Una empresa de la tabla company puede realizar muchas ventas (transacciones) registradas en la tabla transaction, sin embargo, una venta o transacción es ejecutada por una única empresa.

Es importante resaltar que para todo el sprint se consideró que la solicitud de información relacionada a ventas implicaba que la misma era positiva, ósea que no había sido declined y se había ejecutado satisfactoriamente con la aprobación del usuario.

Ejercicio 2

Utilizando JOIN realizarás las siguientes consultas:

- Listado de los países que están generando ventas.



En esta consulta como se menciono antes se considera declined = 0 como ejecución positiva de las ventas. Se usa DISTINCT para listar los países solo una vez.

- Desde cuántos países se generan las ventas.

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

```
14 # Desde cuántos países se generan las ventas.
15 • SELECT COUNT(DISTINCT country) AS Cantidad_Paises_con_Ventas
16 FROM transaction AS t
17 LEFT JOIN company AS c
18 ON t.company_id = c.id
19 WHERE declined = 0;
20
```

The results pane shows a single row with the value 15 for the column Cantidad_Paises_con_Ventas.

Cantidad_Paises_con_Ventas
15

The output pane shows the execution details for the query:

#	Time	Action	Message	Duration / Fetch
46	22:12:13	SELECT DISTINCT country AS Paises_con_Ventas FROM transaction AS ...	15 row(s) returned	0.391 sec / 0.000 sec
47	22:16:25	SELECT COUNT(DISTINCT country) AS Cantidad_Paises_con_Ventas FR...	1 row(s) returned	0.406 sec / 0.000 sec

Similar a la consulta anterior agregando el uso de un COUNT a la lista.

- Identifica a la compañía con la mayor media de ventas.

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

```
22 • SELECT company_name, ROUND(AVG(amount), 2) AS avg_ventas
23 FROM transaction AS t
24 LEFT JOIN company AS c
25 ON t.company_id = c.id
26 WHERE declined = 0
27 GROUP BY company_name
28 ORDER BY avg_ventas DESC
29 LIMIT 1;
```

The results pane shows a single row with the company name 'Ac Fermentum Incorporated' and an average sales value of 284.91.

company_name	avg_ventas
Ac Fermentum Incorporated	284.91

The output pane shows the execution details for the query:

#	Time	Action	Message	Duration / Fetch
50	22:17:42	SELECT company_name, ROUND(AVG(amount)) AS avg_ventas FROM tr...	1 row(s) returned	0.500 sec / 0.000 sec
51	22:17:50	SELECT company_name, ROUND(AVG(amount), 2) AS avg_ventas FROM ...	1 row(s) returned	0.547 sec / 0.000 sec

Se agrupa por empresa y ordena de mayor a menor para dar con la que tiene la mayor venta promedio.

Ejercicio 3

Utilizando sólo subconsultas (sin utilizar JOIN):

- Muestra todas las transacciones realizadas por empresas de Alemania.

The screenshot shows a SQL query in a text editor window. The query is as follows:

```
47 # Muestra todas las transacciones realizadas por empresas de Alemania.
48 # RESULTADO
49 SELECT t.id, g.company_name, country
50 FROM transaction AS t,
51 (SELECT company_name, id, country
52 FROM company AS c
53 WHERE country = "Germany") g
54 WHERE declined = 0 AND g.id = t.company_id ;
```

Below the query, the 'Result Grid' shows the results of the query. The results are as follows:

id	company_name	country
00138D3B-206D-4C03-94B7-63A2676E9B64	Ac Fermentum Incorporated	Germany
0013C1B6-3884-4D6C-8154-E2B3FEBCA8E9	Ac Fermentum Incorporated	Germany
00201A11-2E62-44C4-941D-198FC8D677F0	Ac Fermentum Incorporated	Germany
00235618-0A5C-4D49-90CB-83A940D8923	Ac Fermentum Incorporated	Germany
005A5A7B-1F1A-4B6C-9B15-1625A78C9C38	Ac Fermentum Incorporated	Germany
00687139-48B2-4FFA-8E73-B20376F04AB4	Ac Fermentum Incorporated	Germany
0074F4DD-32F1-4827-8758-55896314623A	Ac Fermentum Incorporated	Germany
00AAB9CD-39D6-4DCB-8A1D-13BE73DC90A9	Ac Fermentum Incorporated	Germany
00BE09D4-6920-47D8-ABE8-325E2269829D	Ac Fermentum Incorporated	Germany
00DA0383-E048-4577-8ED1-3C56C258FF2F	Ac Fermentum Incorporated	Germany
00DD11DE-ED01-48BD-93A0-174D183A59DF	Ac Fermentum Incorporated	Germany
01449CE0-98E9-4DE5-9810-728C6A00E6F	Ac Fermentum Incorporated	Germany

The 'Output' window shows the execution of the query. The results are as follows:

#	Time	Action	Message	Duration / Fetch
51	22:17:50	SELECT company_name, ROUND(AVG(amount), 2) AS avg_ventas FROM ...	1 row(s) returned	0.547 sec / 0.000 sec
52	22:38:43	SELECT t.id, g.company_name, country FROM transaction AS t, (SELECT ...	13269 row(s) returned	0.016 sec / 0.093 sec

Se realiza primero una subconsulta para identificar las empresas alemanas (resultando en 8 empresas), esta tabla de nombre g se anexa en el FROM para así generar todas las transacciones.

- Lista las empresas que han realizado transacciones por ventas superior a la media de todas las transacciones.

The screenshot shows a SQL query in a text editor window. The query is as follows:

```
82 # Lista las empresas que han realizado transacciones por un amount superior a la media de todas las transacciones.
83 # RESPUESTA FINAL
84 SELECT DISTINCT company_name AS empresas_ventas_mayor_al_promedio
85 FROM company AS c
86 WHERE id IN (SELECT DISTINCT company_id
87 FROM transaction AS t
88 WHERE declined = 0
89 AND amount > (SELECT ROUND(AVG(amount), 2) # filtro de ventas al AVG
90 FROM transaction AS t));
```

Below the query, the 'Result Grid' shows the results of the query. The results are as follows:

empresas_ventas_mayor_al_promedio
Ac Fermentum Incorporated
Magna A Neque Industries
Fusce Corp.
Conwallis In Incorporated
Ante Iaculis Nec Foundation
Donec Ltd
Sed Nunc Ltd
Amet Nulla Donec Corporation

The 'Output' window shows the execution of the query. The results are as follows:

#	Time	Action	Message	Duration / Fetch
44	11:46:06	SELECT DISTINCT company_name AS empresas_mayor_ventas_al_promedio ...	100 row(s) returned	0.188 sec / 0.000 sec
45	11:46:20	SELECT DISTINCT company_name AS empresas_ventas_mayor_al_promedio ...	100 row(s) returned	0.203 sec / 0.000 sec
46	11:49:44	SELECT DISTINCT company_name AS empresas_ventas_mayor_al_promedio ...	100 row(s) returned	0.187 sec / 0.000 sec

Primero se realizó una subconsulta para definir el valor de las ventas promedio (AVG), luego se generó la subconsulta para identificar los códigos de empresa que cumplan la condición de ventas superiores a ventas promedio calculado anteriormente, para al final aplicar la consulta para obtener el listado de las empresas.

Vale resaltar que el resultado coincide con el numero total de empresas existentes en la base de datos.

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

The screenshot shows a SQL IDE with a query editor and an output window. The query editor contains the following SQL code:

```
114 # Eliminarán del sistema las empresas que carecen de transacciones registradas, entrega el listado de estas empresas.
115 # RESPUESTA FINAL
116 SELECT company_name AS empresas_sin_ventas
117 FROM company AS c
118 WHERE id NOT IN (SELECT company_id
119                 FROM transaction AS t
120                 WHERE declined = 0);
121
```

The output window shows the results of the query. The first two rows show the company names and the number of transactions returned. The third row shows the result of the query, which is 0 rows returned.

#	Time	Action	Message	Duration / Fetch
35	11:31:08	SELECT company_name AS empresas_sin_transacciones FROM company AS ...	0 row(s) returned	0.141 sec / 0.000 sec
36	11:31:13	SELECT company_name AS empresas_sin_transacciones FROM company AS ...	0 row(s) returned	0.125 sec / 0.000 sec
37	11:31:50	SELECT company_name AS empresas_sin_ventas FROM company AS c WH...	0 row(s) returned	0.125 sec / 0.000 sec

En esta consulta buscamos solo las empresas que no han realizado ventas, es decir que la transacción al final fue declinada (declined =1).

Aplicado diversas consultas se identifico que las empresas con transacciones declinadas eran un total de 86, sin embargo se verifico también el número de empresas con ventas culminadas, ya que una empresa puede tener tanto ventas rechazadas como ventas exitosas. El resultado fue de 100 empresas con ventas sin declinar (declined = 0).

Por tal motivo la consulta se planteo desde el enfoque de filtrar por aquellas empresas que no tuvieran ventas finalizadas, debido a que el listado que se quiere debe contener las empresas que han realizado transacciones declinadas únicamente. El resultado es 0 no hay empresas que no tengan solo transacciones declinadas.

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.

The screenshot shows a SQL query in a window titled 'SQL File 1*'. The query is as follows:

```
4
5 # Identifica los cinco días que se generó la mayor cantidad de ingresos en la empresa por ventas.
6 # Muestra la fecha de cada transacción junto con el total de las ventas.
7 # RESPUESTA
8 • SELECT DATE(timestamp) AS fecha, SUM(amount) AS max_ventas_del_dia
9 FROM transaction AS t
10 LEFT JOIN company AS c
11 ON c.id = t.company_id
12 GROUP BY fecha
13 ORDER BY SUM(amount) DESC
14 LIMIT 5;
```

Below the query, the 'Result Grid' shows the following data:

fecha	max_ventas_del_dia
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

The 'Output' window shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
87	12:35:26	SELECT company_name FROM transaction AS t INNER JOIN company AS c ...	46580 row(s) returned	0.156 sec / 0.234 sec
88	13:38:25	SELECT DATE(timestamp) AS fecha, SUM(amount) AS ventas_del_dia FROM ...	5 row(s) returned	0.797 sec / 0.000 sec
89	13:39:17	SELECT DATE(timestamp) AS fecha, SUM(amount) AS max_ventas_del_dia F...	5 row(s) returned	0.515 sec / 0.000 sec

Primero se separo fecha de hora en la variable timestamp, se agrupo por fechas y ordeno por el valor del total de ventas de cada día, considerando los valores mayores y limitando a solo 5.

Ejercicio 2

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

The screenshot shows a SQL query in a window titled 'SQL File 1*'. The query is as follows:

```
18 # ¿Cuál es la media de ventas por país? Presenta los resultados ordenados de mayor a menor medio.
19 # RESPUESTA
20 • SELECT AVG(amount) AS avg_ventas, country
21 FROM transaction AS t
22 INNER JOIN company AS c
23 ON c.id = t.company_id
24 GROUP BY country
25 ORDER BY avg_ventas DESC;
```

Below the query, the 'Result Grid' shows the following data:

avg_ventas	country
265.190742	Australia
264.977877	United States
261.153042	Belgium
260.841391	Germany
260.644761	Ireland
260.468125	Spain
259.979185	France
259.586176	New Zealand
259.375337	Norway
258.436128	Netherlands
258.272740	Italy

The 'Output' window shows the execution log with the following entries:

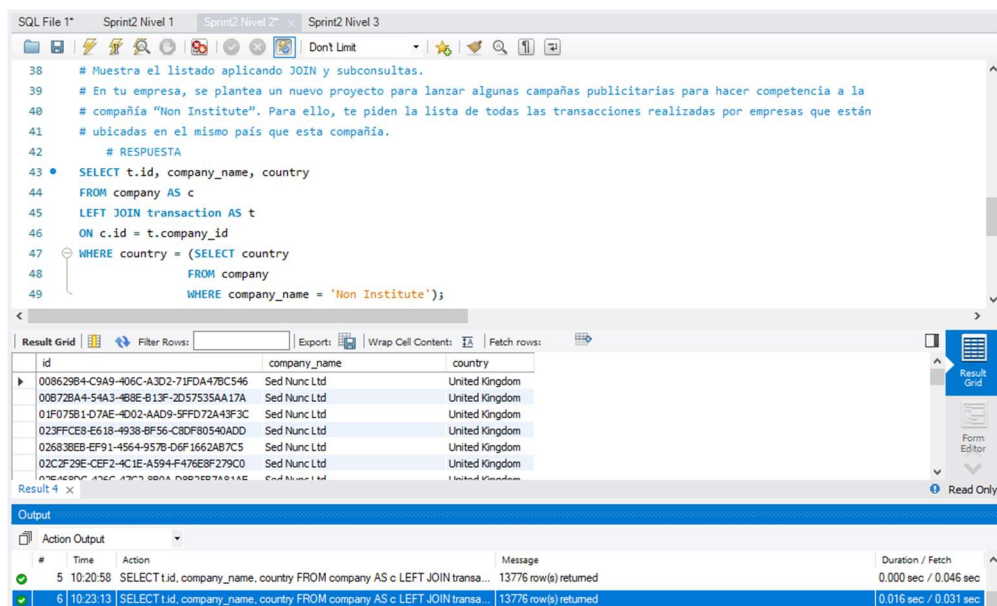
#	Time	Action	Message	Duration / Fetch
3	10:19:40	SELECT DATE(timestamp) AS fecha, MAX(amount) FROM transaction AS t LE...	5 row(s) returned	0.468 sec / 0.000 sec
4	10:20:10	SELECT AVG(amount) AS avg_ventas, country FROM transaction AS t INNER...	15 row(s) returned	0.609 sec / 0.000 sec

Similar a la consulta anterior se agrupa por países y se calcula y ordena por el promedio de ventas, considerando primeros más los países que tienen los valores mas altos.

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.

- Muestra el listado aplicando JOIN y subconsultas.



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

```
38 # Muestra el listado aplicando JOIN y subconsultas.
39 # En tu empresa, se plantea un nuevo proyecto para lanzar algunas campañas publicitarias para hacer competencia a la
40 # compañía "Non Institute". Para ello, te piden la lista de todas las transacciones realizadas por empresas que están
41 # ubicadas en el mismo país que esta compañía.
42 # RESPUESTA
43 SELECT t.id, company_name, country
44 FROM company AS c
45 LEFT JOIN transaction AS t
46 ON c.id = t.company_id
47 WHERE country = (SELECT country
48                  FROM company
49                  WHERE company_name = 'Non Institute');
```

The results grid shows the following data:

id	company_name	country
008629B4-C9A9-406C-A3D2-71FDA478C546	Sed Nunc Ltd	United Kingdom
00872BA4-54A3-4B8E-B13F-2D57535AA17A	Sed Nunc Ltd	United Kingdom
01F075B1-D7AE-4D02-AAD9-5FFD72A43F3C	Sed Nunc Ltd	United Kingdom
023FFCE8-E618-4938-8F56-C8DF80540ADD	Sed Nunc Ltd	United Kingdom
026838EB-EF91-4564-957B-D6F1662AB7C5	Sed Nunc Ltd	United Kingdom
02C2F29E-CEF2-4C1E-A594-F476E8F279C0	Sed Nunc Ltd	United Kingdom
02E4E9C1-47C1-47C1-88A1-08B1E7A1A1E	Sed Nunc Ltd	United Kingdom

The output section shows the execution of the query, indicating that 13776 rows were returned.

Para este ejercicio lo primero que se realizó fue una subconsulta para identificar el país donde se localiza la empresa “Non Institute” (resultando el Reino Unido). Luego se unieron las tablas company y transaction para listar las ventas y empresas del Reino Unido incluyendo en la búsqueda a empresa de referencia “Non Institute”.

- Muestra el listado aplicando solo subconsultas.

SQL File 1* Sprint2 Nivel 1 Sprint2 Nivel 2 Sprint2 Nivel 3

```

56 # Muestra el listado aplicando solo subconsultas.
57 # En tu empresa, se plantea un nuevo proyecto para lanzar algunas campañas publicitarias para hacer competencia a la
58 # compañía "Non Institute". Para ello, te piden la lista de todas las transacciones realizadas por empresas que están
59 # ubicadas en el mismo país que esta compañía.
60 # RESPUESTA
61 • SELECT t.id, company_name, country
62   FROM company AS c,
63   (SELECT id, company_id
64    FROM transaction) t
65  WHERE t.company_id = c.id
66  AND country = (SELECT country
67                 FROM company
68                 WHERE company_name = 'Non Institute');

```

Result Grid

id	company_name	country
008629B4-C9A9-406C-A3D2-71FDA47BC546	Sed Nunc Ltd	United Kingdom
00B72BA4-54A3-4B8E-B13F-2D5753AA17A	Sed Nunc Ltd	United Kingdom
01F075B1-D7AE-4D02-AAD9-5FFD72A43F3C	Sed Nunc Ltd	United Kingdom
023FFC8B-E618-4938-BF56-C8DF80540ADD	Sed Nunc Ltd	United Kingdom
026838EB-EF91-4564-957B-D6F1662AB7C5	Sed Nunc Ltd	United Kingdom

Result 5 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
6	10:23:13	SELECT t.id, company_name, country FROM company AS c LEFT JOIN transa...	13776 row(s) returned	0.016 sec / 0.031 sec
7	10:24:05	SELECT t.id, company_name, country FROM company AS c, (SELECT id, com...	13776 row(s) returned	0.000 sec / 0.031 sec

En este caso se mantiene la subconsulta para ubicar el país de la empresa "Non Institute". Se realiza otra subconsulta para traer los datos de las ventas y código de empresas que reemplazarían el uso del JOIN. En ambos casos se obtiene el mismo numero de registros y se incluye a la empresa de referencia "Non Institute".

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.

SQL File 1* Sprint2 Nivel 1 Sprint2 Nivel 2 Sprint2 Nivel 3

```

21 # Presenta el nombre, teléfono, país, fecha y amount, de aquellas empresas que realizaron
22 # transacciones con un valor comprendido entre 350 y 400 euros y en alguna de estas fechas:
23 # 29 de abril de 2015, 20 de julio de 2018 y 13 de marzo de 2024.
24 # Ordena los resultados de mayor a menor cantidad.
25 # RESPUESTA
26 • SELECT company_name, phone, country, DATE(timestamp) AS fecha, amount
27   FROM transaction AS t
28  LEFT JOIN company AS c
29    ON t.company_id = c.id
30  WHERE DATE(timestamp) IN ('2015-04-29', '2018-07-20', '2024-03-13')
31  AND amount BETWEEN 350 AND 400
32  ORDER BY amount DESC;

```

Result Grid

company_name	phone	country	fecha	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 Pede 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
Orci Adipiscing Limited	03 18 00 77 81	United Kingdom	2018-07-20	373.71
Fringilla LLC	08 29 15 93 57	New Zealand	2015-04-29	367.62

Result 1 x

Output

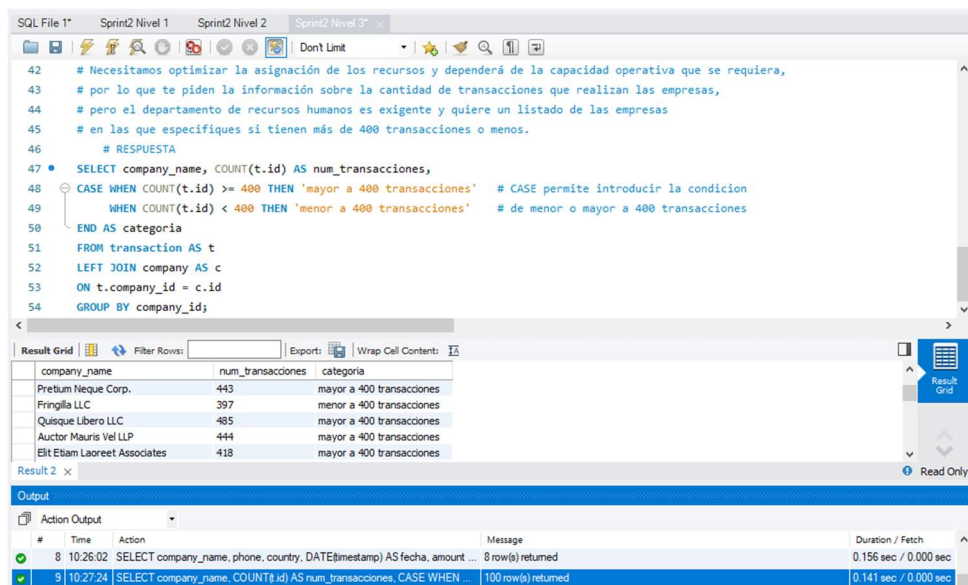
Action Output

#	Time	Action	Message	Duration / Fetch
7	10:24:05	SELECT t.id, company_name, country FROM company AS c, (SELECT id, com...	13776 row(s) returned	0.000 sec / 0.031 sec
8	10:26:02	SELECT company_name, phone, country, DATE(timestamp) AS fecha, amount ...	8 row(s) returned	0.156 sec / 0.000 sec

En esta consulta se aplicaron 2 filtros: BETWEEN para definir el intervalo de ventas de 350 a 400, y DATE (timestamp) IN para separar fecha de hora en timestamp y reducir la búsqueda al conjunto de fechas que solicitaba el ejercicio.

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.



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

```
42 # Necesitamos optimizar la asignación de los recursos y dependerá de la capacidad operativa que se requiera,  
43 # por lo que te piden la información sobre la cantidad de transacciones que realizan las empresas,  
44 # pero el departamento de recursos humanos es exigente y quiere un listado de las empresas  
45 # en las que especifiques si tienen más de 400 transacciones o menos.  
46 # RESPUESTA  
47 SELECT company_name, COUNT(t.id) AS num_transacciones,  
48 CASE WHEN COUNT(t.id) >= 400 THEN 'mayor a 400 transacciones' # CASE permite introducir la condicion  
49 WHEN COUNT(t.id) < 400 THEN 'menor a 400 transacciones' # de menor o mayor a 400 transacciones  
50 END AS categoria  
51 FROM transaction AS t  
52 LEFT JOIN company AS c  
53 ON t.company_id = c.id  
54 GROUP BY company_id;
```

The results grid shows the following data:

company_name	num_transacciones	categoria
Pretium Neque Corp.	443	mayor a 400 transacciones
Fringilla LLC	397	menor a 400 transacciones
Quisque Libero LLC	485	mayor a 400 transacciones
Auctor Mauris Vel LLP	444	mayor a 400 transacciones
Elit Etiam Laoreet Associates	418	mayor a 400 transacciones

The output panel shows the execution of the query, indicating that 100 rows were returned.

En esta consulta se requería calcular la cantidad de transacciones por empresa, se realizó un COUNT y un GROUP BY que permite contar el numero de ventas y agruparlo por empresas. Para la solicitud de recursos humanos se usó el comando CASE que funciona como un condicional para SQL. En este caso se especificó el límite de 400 transacciones, se determina la condición con WHEN y después THEN define la salida que se imprime en la columna en este ejercicio categoría.