

SPRINT 2

Nivel 1

Ejercicio 1: Diagrama Entidad Relación del esquema “Transacciones”

Características principales del esquema creado en el modelo.

Es un modelo de tablas relacionadas en la que tenemos dos tablas:

- Tabla “*Company*”: es una tabla en la que encontramos información de cada una de las empresas distribuidas por varios países que han hecho alguna transacción con la empresa objetivo de análisis. Tiene 6 campos (columnas), todas ellas de tipo Varchar. La PK (primary key) de la tabla es un código identificativo de la empresa que podemos encontrar en el campo “*id*”.
- Tabla “*Transacción*”: es una tabla donde tenemos la información referente a cada una de las transacciones realizadas, con un total de 9 campos, de tipo Varchar, float, timestamp, decimal y tinyint. La PK es también un campo llamado “*id*”.

Es una relación de uno a muchos (1:N). Para un registro de la tabla “*company*” existen varios registros en la tabla “*transaction*”. Se relacionan a través de la PK de la tabla “*company*” con la FK de la tabla “*transaction*” que se llama “*company_id*”.

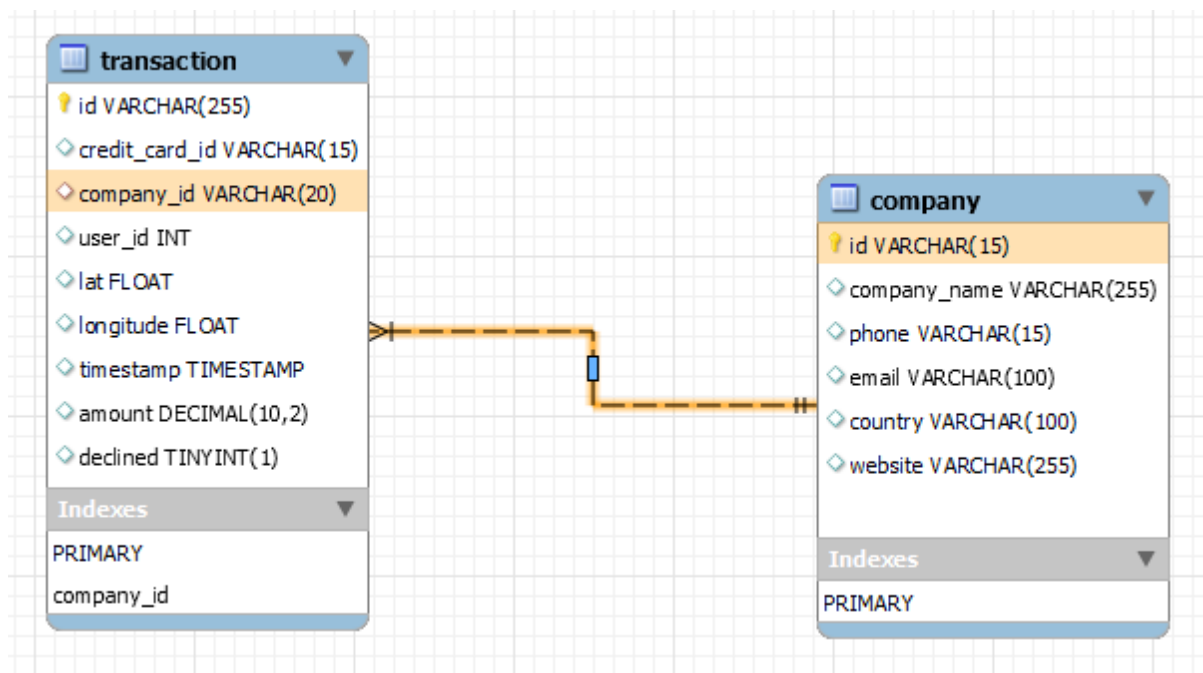


Fig. 1: Diagrama Entidad Relación

Una vez ejecutado el documento “*dades_introduir*” con éxito se pueden observar que se han ejecutado correctamente todas las filas incluidas y se han añadido a las tablas. (Fig. 2).

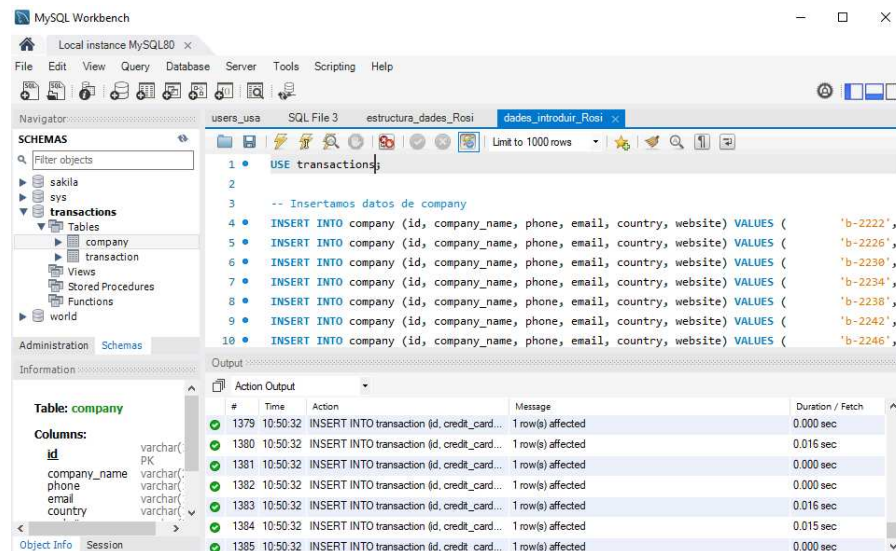


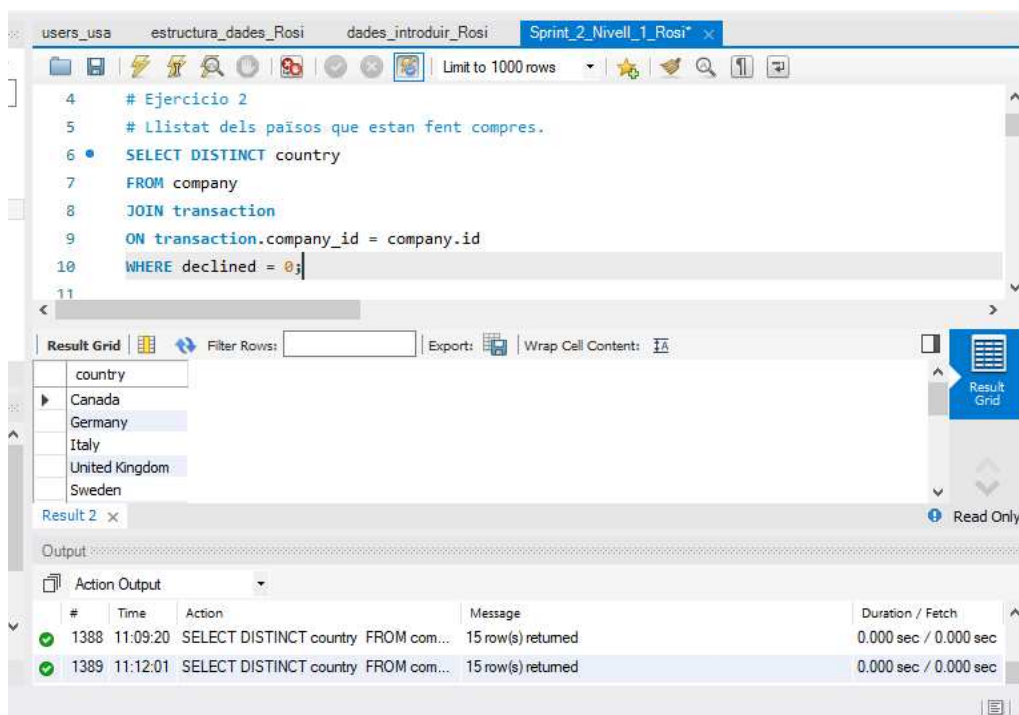
Fig. 2: Ejecución de la inserción de datos en las tablas.

Ejercicio 2: Uso de JOINS

2A. Listado de países que están haciendo compras.

De la tabla que consigo de la unión (INNER JOIN) de la tabla “company” y la tabla “transaction” hago un SELECT DISTINCT de países para conseguir la lista de los países que realizan compras en estos momentos.

Y por si había algún país que había hecho alguna transacción pero no había realizado una compra, he puesto la condición “declined = 0”. Entendiendo que el 0 = False, es decir, que no se ha declinado la transacción y que ha sido realizada. Por lo tanto, cada transacción con “declined = 0” es una compra.



2B. Cuántos países están haciendo compras.

Con la misma JOIN de las tablas “company” y “transaction”, hago un COUNT de los diferentes países que resultan. Me devuelve la cantidad de **15 países**.

```

11
12 # Des de quants països es realitzen les compres.
13 • SELECT count(distinct country)
14 FROM company
15 JOIN transaction
16 ON transaction.company_id = company.id
17 WHERE declined = 0;
18

```

Result Grid

count(distinct country)
15

Result 4 x Read Only

Output

#	Time	Action	Message	Duration / Fetch
✓ 1390	12:05:46	SELECT count(distinct country) FROM c...	1 row(s) returned	0.016 sec / 0.000 sec
✓ 1391	12:06:22	SELECT count(distinct country) FROM c...	1 row(s) returned	0.000 sec / 0.000 sec

2C. Compañía con la media más grande de ventas.

Con la misma JOIN de las tablas “company” y “transaction”, ahora hago una agrupación por compañía para hacer la media de las ventas de cada compañía, las ordeno de forma descendente y limito el resultado a un registro para ver la compañía con la media de ventas más elevada. Como nos están pidiendo la media de las ventas, elimino los registros de las transacciones que no se han llegado a realizar de nuevo con el filtro “declined = 0”.

La compañía con una media de ventas más grande es **Eget Ipsum Ltd.** con una media de **481,86 euros**.

```

19 # Identifica la compañía amb la mitjana més gran de vendes.
20 • SELECT company_name, round(AVG(amount), 2) AS promedioempresa
21 FROM company
22 JOIN transaction
23 ON transaction.company_id = company.id
24 WHERE declined = 0
25 GROUP BY company_name
26 ORDER BY promedioempresa DESC
27 LIMIT 1;
28

```

Result Grid

company_name	promedioempresa
Eget Ipsum Ltd	481.86

Result 2 x Read Only

Output

#	Time	Action	Message	Duration / Fetch
✓ 693	12:39:19	SELECT company_name, round(AVG(amount), 2) ...	1 row(s) returned	0.015 sec / 0.000 sec
✓ 694	12:43:24	SELECT company_name, round(AVG(amount), 2) ...	1 row(s) returned	0.016 sec / 0.000 sec

Ejercicio 3: Uso de SUBCONSULTAS

3A. Transacciones realizadas por empresas alemanas.

Para empezar hago una consulta en la que identifico las empresas que son alemanas con una condición *country* = "Germany", que se convertirá más tarde en la subconsulta. Después, selecciono las transacciones cuyo identificador de compañía está dentro de las empresas con sede en Alemania que me daba la subconsulta. Existen un total de **118 transacciones** realizadas en Alemania.

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

```

30 # Ejercicio 3
31 # Muestra totes les transaccions realitzades per empreses d'Alemanya.
32 SELECT * FROM transaction
33 WHERE company_id IN (SELECT id
34                       FROM company
35                       WHERE country = "Germany");

```

The results grid displays the following data:

id	credit_card_id	company_id	user_id	lat	longitude	timestamp
108B1D1D-5B23-A76C-55EF-C568E49A05DD	CcU-2938	b-2222	275	83.7839	-178.86	2021-07-07 17:...
EA2C3281-C9C1-A387-44F8-729FB4B51C76	CcU-2938	b-2222	275	20.2004	-116.84	2021-05-09 10:...
0DD2E608-5C9E-D1B3-4999-B99F43AD735A	CcU-2959	b-2234	275	9.68811	130.282	2021-04-17 05:...
AB069F53-965E-A2A8-CE06-CA8C4FD92501	CcU-2959	b-2234	275	1.64819	-158.007	2021-04-15 13:...
0466A42E-47CF-8D24-FD01-C0B689713128	CcU-4219	b-2302	170	-43.9695	-117.525	2021-07-26 07:...
0A476ED9-0C13-1962-F87B-D3563924B539	CcU-4359	b-2302	221	-56.4901	114.801	2022-02-26 20:...

The bottom section of the IDE shows the 'Action Output' table:

#	Time	Action	Message	Duration / Fetch
1394	12:09:43	SELECT company_name, round(AVG(am...	1 row(s) returned	0.015 sec / 0.000 sec
1395	12:23:24	SELECT * FROM transaction WHERE co...	118 row(s) returned	0.000 sec / 0.000 sec

3B. Lista de empresas que han realizado transacciones por una cantidad superior a la media de todas las transacciones.

Primero, realizo una consulta para conseguir el promedio de todas las transacciones y le asigno el nombre de "promediototal" (**256,74 euros de media**). Esta consulta se convertirá en mi subconsulta.

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

```

46
47 # Promedio de ventas totales
48 SELECT round(AVG(amount), 2) AS promediototal
49 FROM transaction;
50

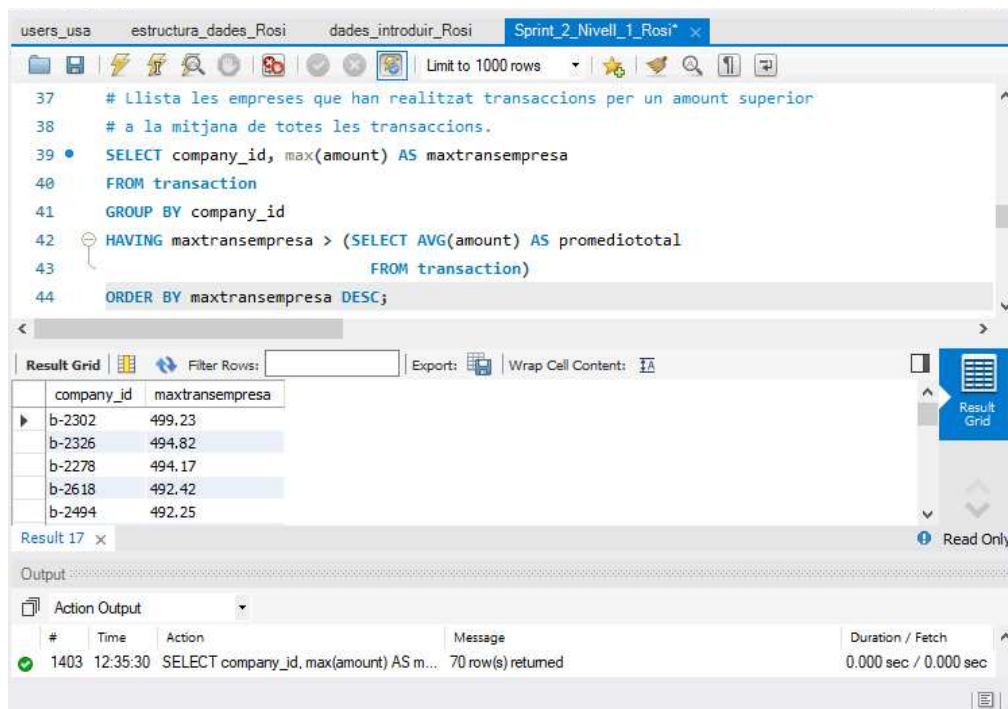
```

The results grid displays the following data:

promediototal
256.74

Después, busco la transacción más alta para cada una de las empresas, utilizando la función MAX y la renombro con "maxtransempresa". Posteriormente, comparo esta cifra al promedio total que había

calculado anteriormente con una HAVING puesto que estoy haciendo una comparación de una función de agregación. Me resultan **70 transacciones superiores** a la media de transacciones.



```

37 # Llista les empreses que han realitzat transaccions per un amount superior
38 # a la mitjana de totes les transaccions.
39 • SELECT company_id, max(amount) AS maxtransempresa
40 FROM transaction
41 GROUP BY company_id
42 HAVING maxtransempresa > (SELECT AVG(amount) AS promediototal
43 FROM transaction)
44 ORDER BY maxtransempresa DESC;

```

company_id	maxtransempresa
b-2302	499.23
b-2326	494.82
b-2278	494.17
b-2618	492.42
b-2494	492.25

Result 17 x

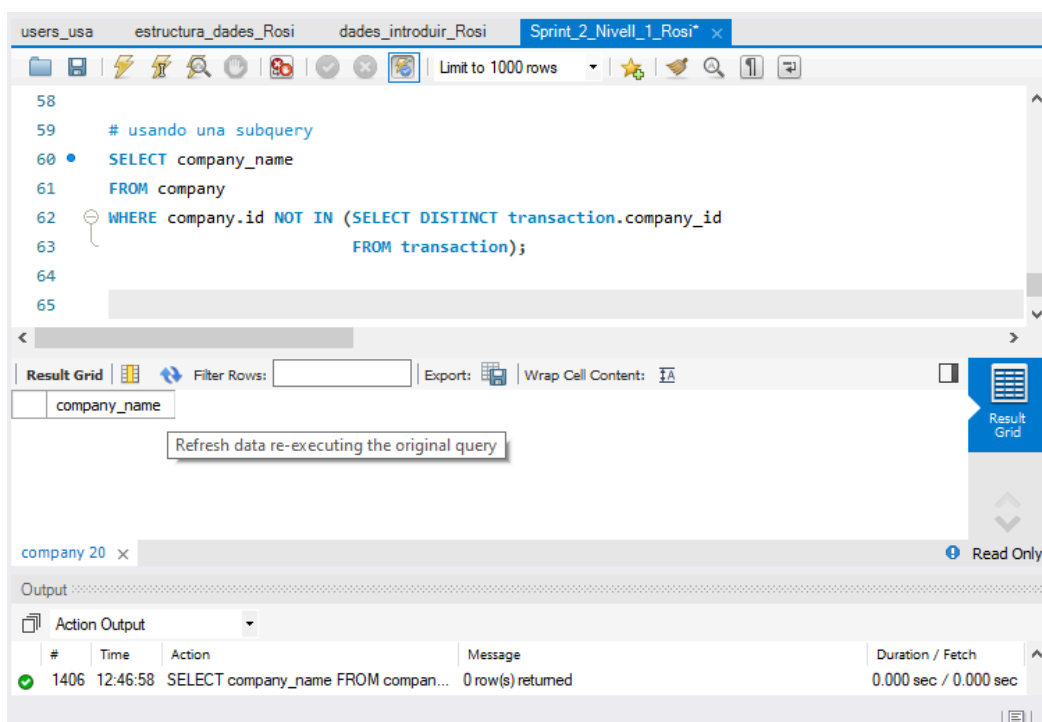
Output

Action Output

#	Time	Action	Message	Duration / Fetch
1403	12:35:30	SELECT company_id, max(amount) AS m...	70 row(s) returned	0.000 sec / 0.000 sec

3C. Listado de empresas sin transacciones registradas.

Primero, selecciono de la tabla “*transaction*” las diferentes empresas que han realizado transacciones (como subconsulta). Y de la tabla “*company*” selecciono las empresas que no están (NOT IN) en la subconsulta explicada anteriormente. Me devuelve 0 empresas, es decir, **todas las empresas de la tabla han realizado transacciones**.



```

58
59 # usando una subquery
60 • SELECT company_name
61 FROM company
62 WHERE company.id NOT IN (SELECT DISTINCT transaction.company_id
63 FROM transaction);
64
65

```

company_name

Refresh data re-executing the original query

company 20 x

Output

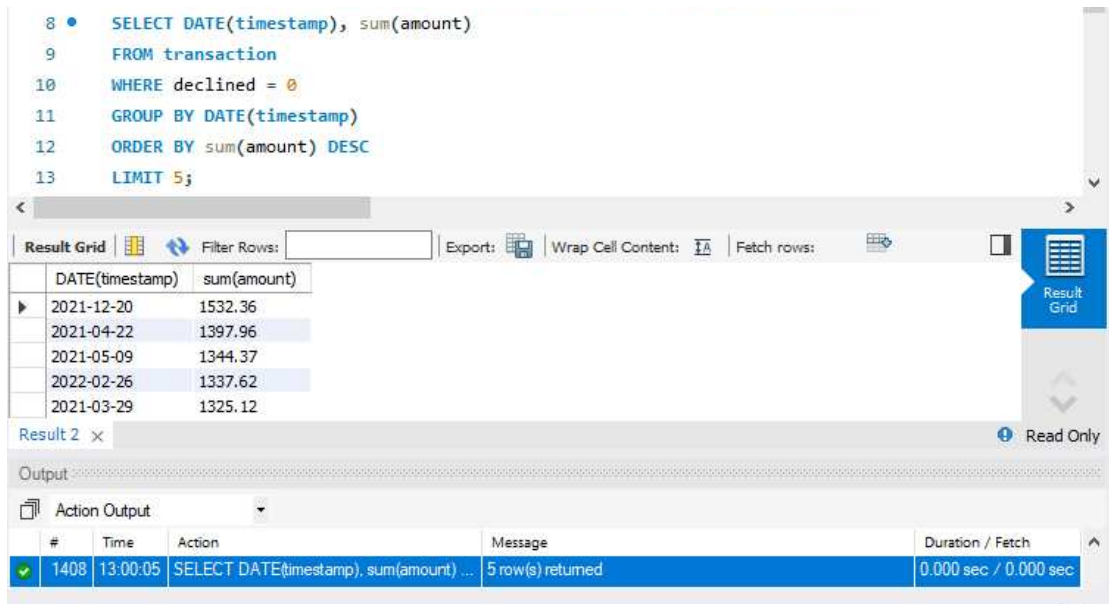
Action Output

#	Time	Action	Message	Duration / Fetch
1406	12:46:58	SELECT company_name FROM compan...	0 row(s) returned	0.000 sec / 0.000 sec

Nivel 2

Ejercicio 1: Los 5 días de más ventas en la empresa.

Para poder hacer una suma de las ventas por día, agrupo la tabla “transaction” por la fecha (*DATE(timestamp)*) y filtro por las transacciones realizadas (*declined = 0*). Una vez tengo esto, puedo realizar la suma de todas las cantidades de ventas para cada uno de los días con la función SUM. Por último ordeno de manera descendente los registros y me quedo con los 5 primeros usando la función LIMIT.



```

8 • SELECT DATE(timestamp), sum(amount)
9   FROM transaction
10  WHERE declined = 0
11  GROUP BY DATE(timestamp)
12  ORDER BY sum(amount) DESC
13  LIMIT 5;

```

DATE(timestamp)	sum(amount)
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

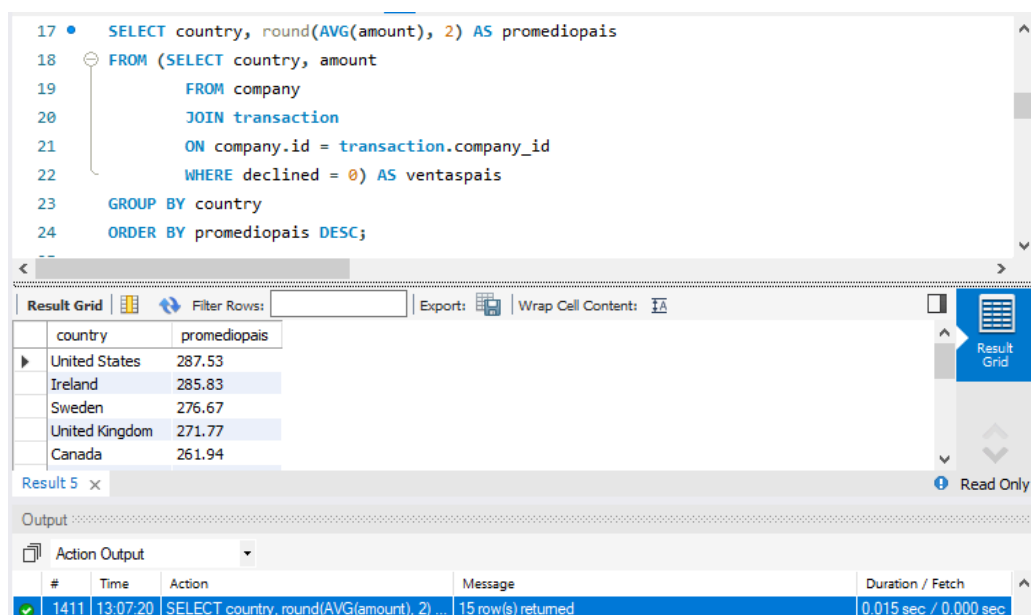
Result 2 x Read Only

Output

#	Time	Action	Message	Duration / Fetch
✓ 1408	13:00:05	SELECT DATE(timestamp), sum(amount) ...	5 row(s) returned	0.000 sec / 0.000 sec

Ejercicio 2: Media de ventas por país.

Primero de todo, creo una tabla a través de una subconsulta en la que me quedo con las transacciones realizadas (*declined = 0*) a la que llamo “ventaspais”. De esta tabla, realizo la media de ventas para cada país con la función AVG y agrupando la tabla por país (GROUP BY country).



```

17 • SELECT country, round(AVG(amount), 2) AS promediopais
18   FROM (SELECT country, amount
19         FROM company
20        JOIN transaction
21         ON company.id = transaction.company_id
22        WHERE declined = 0) AS ventaspais
23  GROUP BY country
24  ORDER BY promediopais DESC;

```

country	promediopais
United States	287.53
Ireland	285.83
Sweden	276.67
United Kingdom	271.77
Canada	261.94

Result 5 x Read Only

Output

#	Time	Action	Message	Duration / Fetch
✓ 1411	13:07:20	SELECT country, round(AVG(amount), 2) ...	15 row(s) returned	0.015 sec / 0.000 sec

Ejercicio 3: Listado de transacciones realizadas en el mismo país que “Non Institute”.

En ambos ejercicios, empiezo por hacer una consulta que me da como resultado el país de “Non Institute”, que será mi subconsulta, gracias al filtro *WHERE company_name LIKE “Non Institute”*.

3A. Aplicando JOINS y subconsultas.

En esta parte del ejercicio, completo la búsqueda con una unión de las tablas “company” y “transaction” para conseguir las transacciones cuyo país sea el mismo que el de mi subconsulta, es decir, cuyo país sea el mismo que el país de “Non Institute” (**United Kingdom**).

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

```

31 # Mostra el llistat aplicant JOIN i subconsultes.
32 • SELECT *
33 FROM transaction
34 JOIN company
35 ON company.id = transaction.company_id
36 WHERE country IN (SELECT country
37                     FROM company
38                     WHERE company_name LIKE "Non Institute");

```

Below the query, the results are displayed in a table with the following columns: id, credit_card_id, company_id, user_id, lat, longitude, timestamp.

id	credit_card_id	company_id	user_id	lat	longitude	timestamp
2B928E1C-EC14-A760-0A75-871477649D6A	CcU-2980	b-2246	275	-41.0496	161.685	2021-08-10 08:
ACD2011A-A2B1-C365-41E1-2AB00C65147A	CcU-2980	b-2246	275	-54.4792	-82.7974	2022-03-05 20:
4334349E-CEB0-3D68-A4D4-FEB7718A1ACE	CcU-3092	b-2310	275	-20.4859	150.87	2021-05-03 22:
BC2B9A38-77B4-28CD-1FE8-14DED863E773	CcU-3092	b-2310	275	-78.0295	18.5295	2021-10-18 07:
1479B3D2-B7BA-C7BB-4CE3-8D7C2DE85ABB	CcU-2994	b-2326	133	66.2672	172.399	2021-08-09 00:
152598C2-029D-D684-4B66-91EDF393EBFF	CcU-2994	b-2326	126	-67.0189	-141.672	2021-07-05 03:
1B636R5R-Δ7ER-7C6Q-J9C9-C54535NΔFN3R	CcU-2994	b-2326	131	70.2543	-13.1336	2021-07-06 08:

Below the results, the output shows the execution of the query:

#	Time	Action	Message	Duration / Fetch
1417	13:13:19	SELECT * FROM transaction WHERE co...	100 row(s) returned	0.000 sec / 0.000 sec
1418	13:13:33	SELECT * FROM transaction JOIN comp...	100 row(s) returned	0.000 sec / 0.000 sec

3B. Aplicando solamente subconsultas.

Para realizarlo solo con subconsultas, realizo unas consultas anidadas. En la segunda subconsulta, selecciono el “id” de las compañías cuyo país está dentro de mi anterior subconsulta, es decir, compañías ubicadas en United Kingdom. Obtengo como resultado al igual que en el anterior un listado de **100 transacciones realizadas en United Kingdom**.

The screenshot shows a database management tool interface with a SQL query editor and a results grid.

SQL Query:

```

48 # Selecciono las transacciones hechas por estas compañías en este país con subconsultas anidad
49 • SELECT *
50 FROM transaction
51 WHERE company_id IN (SELECT id
52                       FROM company
53                       WHERE country IN (SELECT country
54                                       FROM company
55                                       WHERE company_name LIKE "Non Institute"));

```

Result Grid:

id	credit_card_id	company_id	user_id	lat	longitude	timestamp
2B928E1C-EC14-A760-0A75-871477649D6A	CcU-2980	b-2246	275	-41.0496	161.685	2021-08-10 08:
ACD2011A-A2B1-C365-41E1-2AB00C65147A	CcU-2980	b-2246	275	-54.4792	-82.7974	2022-03-05 20:
4334349E-CEB0-3D68-A4D4-FEB7718A1ACE	CcU-3092	b-2310	275	-20.4859	150.87	2021-05-03 22:
BC2B9A38-77B4-28CD-1FE8-14DED863E773	CcU-3092	b-2310	275	-78.0295	18.5295	2021-10-18 07:
147983D2-87BA-C7BB-4CE3-8D7C2DE85ABB	CcU-2994	b-2326	133	66.2672	172.399	2021-08-09 00:
152598C2-029D-D684-4B66-91EDF393EBFF	CcU-2994	b-2326	126	-67.0189	-141.672	2021-07-05 03:
1B636B58-A2F8-7C69-D9C9-C54535DADF03B	CcU-2994	b-2326	131	70.2543	-13.1336	2021-07-06 08:

Output:

#	Time	Action	Message	Duration / Fetch
1418	13:13:33	SELECT * FROM transaction JOIN comp...	100 row(s) returned	0.000 sec / 0.000 sec
1419	13:15:33	SELECT * FROM transaction WHERE co...	100 row(s) returned	0.000 sec / 0.000 sec

Nivel 3

Ejercicio 1: Transacciones entre 100 y 200 euros en las fechas solicitadas.

1A. 29 de abril del 2021, 20 de julio del 2021 y 13 de marzo del 2022

Empiezo realizando una JOIN para conseguir toda la información de las empresas que me solicitan. Después, filtro las empresas que han realizado transacciones por una cantidad comprendida entre 100 y 200 euros. Más tarde, con un HAVING filtro las empresas que han realizado transacciones en las fechas solicitadas. Finalmente, las ordeno de mayor o menor cantidad de transacción.

Resultan un total de **5 transacciones realizadas en esas fechas por un importe entre 100 y 200 euros.**


```

121 # Ordena els resultats de major a menor quantitat.
122
123 • SELECT company_name, phone, country, DATE(timestamp) AS date, amount
124 FROM company
125 JOIN transaction
126 ON company.id = transaction.company_id
127 WHERE amount BETWEEN 100 AND 200
128 HAVING date IN ("2021-04-29", "2021-07-20", "2022-03-13")
129 ORDER BY amount DESC;
130

```

Result Grid

company_name	phone	country	date	amount
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
Enim Condimentum Ltd	09 55 51 66 25	United Kingdom	2021-04-29	149.89
Lorem Eu Incorporated	01 83 66 62 07	Canada	2021-07-20	133.39
Nunc Interdum Incorporated	05 18 15 48 13	Germany	2021-04-29	111.51

Result 3

Output

Action Output

#	Time	Action	Message	Duration / Fetch
694	12:43:24	SELECT company_name, round(AVG(amount), 2) ...	1 row(s) returned	0.016 sec / 0.000 sec
695	12:46:27	SELECT company_name, phone, country, DATE(...	5 row(s) returned	0.000 sec / 0.000 sec

Ejercicio 2: Empresas con más de 4 transacciones.

```

31 • SELECT company_name,
32 CASE WHEN numtransaccions > 4 THEN "Más de 4"
33 ELSE "4 o menos" END AS mas_cuatre_transaccions
34 FROM (SELECT company_id, count(company_id) AS numtransaccions
35 FROM transaction
36 GROUP BY company_id) AS subquery
37 JOIN company
38 ON subquery.company_id = company.id;

```

Result Grid

company_name	mas_cuatre_transaccions
Ac Fermentum Incorporated	4 o menos
Magna A Neque Industries	4 o menos
Fusce Corp.	4 o menos
Convallis In Incorporated	4 o menos
Ante Iaculis Nec Foundation	4 o menos
Donec Ltd	4 o menos

Result 5

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1424	13:26:55	SELECT company_name, CASE WHEN ...	100 row(s) returned	0.000 sec / 0.000 sec
1425	13:27:51	SELECT company_name, CASE WHEN ...	100 row(s) returned	0.000 sec / 0.000 sec

Para poder especificar cada empresa si ha realizado más de 4 transacciones, en primer lugar, necesitamos conseguir una tabla en la que se cuenten el número de transacciones realizadas por cada

compañía, gracias a un COUNT, en la tabla transacciones agrupando por *“company_id”*. Esto resultaría en la subconsulta que he realizado (*“subquery”*).

Una vez realizada esa nueva tabla con *“company_id”* y el número de transacciones por compañía. Podemos realizar una selección del nombre de la compañía (gracias a la JOIN) y crearemos una nueva columna denominada *“mas_cuatre_transacciones”* en la que se especifica si la compañía tiene más de cuatro transacciones *“Más de 4”* o si solo tiene 4 o menos transacciones *“4 o menos”*, gracias al comando CASE WHEN que hace una comparación de *“numtransacciones”* y devuelve un mensaje si se cumple u otro mensaje si no se cumple dicha condición.