

Tasca S2.01. Nocións bàsiques SQL

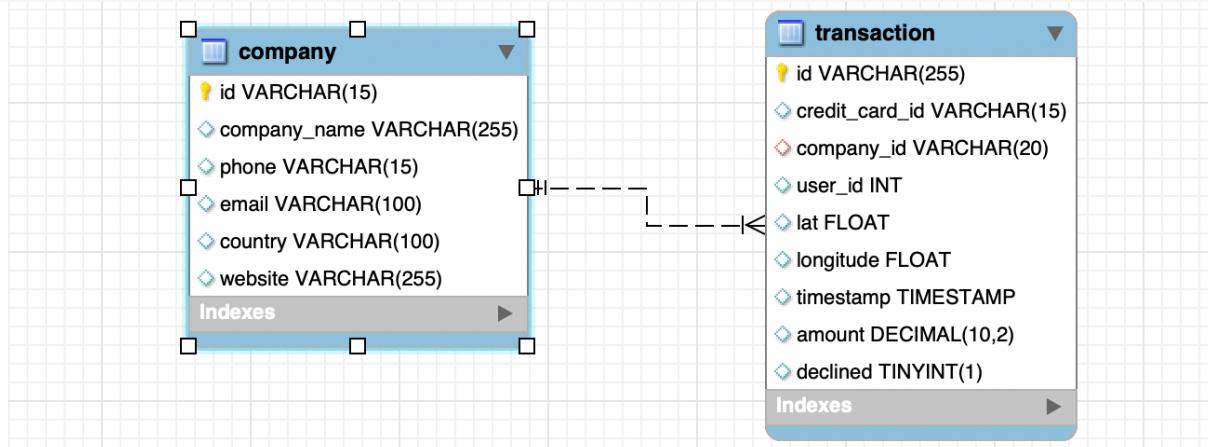
Nivell 1

Timan(Nakta) Samipour

Compañero para revisión peer-to-peer : **Alexei Levitchi**

Exercici 1

A partir dels documents adjunts (estructura_dades i dades_introduir), importa les dues taules. Mostra les característiques principals de l'esquema creat i explica les diferents taules i variables que existeixen. Assegura't d'incloure un diagrama que il·lustri la relació entre les diferents taules i variables.



Descripción:

En esta imagen se muestra el diagrama entidad-relación (ER) de la base de datos. El diagrama incluye las tablas `company` y `transaction`, las cuales están conectadas mediante una clave foránea. Este diagrama permite visualizar de forma clara la estructura de la base de datos y facilita la comprensión de las relaciones entre las tablas.

Aspectos destacados:

- La clave primaria de la tabla `company` es el campo `id`.
- La tabla `transaction` contiene la clave foránea `company_id`.
- El tipo de relación entre las tablas es uno a muchos (1:N).

```
9 • SHOW TABLES;
10
Result Grid
Tables_in_transactions
company
transaction
```

Descripción:

Esta imagen muestra el resultado del comando `SHOW TABLES`, que lista todas las tablas existentes en la base de datos `transactions`.

Resultado:

La base de datos contiene las siguientes tablas:

- `company`
- `transaction`

11 • DESCRIBE company;

100% 18:11

Result Grid Filter Rows: Search Export:

| Field | Type | Null | Key | Default | Extra |
|---------------------------|--------------|------|-----|-------------------|-------|
| <code>id</code> | varchar(15) | NO | PRI | <code>NULL</code> | |
| <code>company_name</code> | varchar(255) | YES | | <code>NULL</code> | |
| <code>phone</code> | varchar(15) | YES | | <code>NULL</code> | |
| <code>email</code> | varchar(100) | YES | | <code>NULL</code> | |
| <code>country</code> | varchar(100) | YES | | <code>NULL</code> | |
| <code>website</code> | varchar(255) | YES | | <code>NULL</code> | |

Descripción:

En esta imagen se observa la estructura de la tabla `company`.

Aspectos observables:

- El campo `id` está definido como clave primaria (PRI).
- El resto de los campos son de tipo VARCHAR.
- El único campo obligatorio es `id`.

13 • DESCRIBE transaction;

14 22:13

Result Grid Filter Rows: Search Export:

| Field | Type | Null | Key | Default | Extra |
|-----------------------------|---------------|------|-----|-------------------|-------|
| <code>id</code> | varchar(255) | NO | PRI | <code>NULL</code> | |
| <code>credit_card_id</code> | varchar(15) | YES | | <code>NULL</code> | |
| <code>company_id</code> | varchar(20) | YES | MUL | <code>NULL</code> | |
| <code>user_id</code> | int | YES | | <code>NULL</code> | |
| <code>lat</code> | float | YES | | <code>NULL</code> | |
| <code>longitude</code> | float | YES | | <code>NULL</code> | |
| <code>timestamp</code> | timestamp | YES | | <code>NULL</code> | |
| <code>amount</code> | decimal(10,2) | YES | | <code>NULL</code> | |
| <code>declined</code> | tinyint(1) | YES | | <code>NULL</code> | |

Descripción:

Esta imagen muestra la estructura de la tabla `transaction`.

Aspectos destacados:

- El campo `id` es la clave primaria.
- El campo `company_id` tiene un índice (MUL).
- Existen campos financieros, temporales y geográficos.

15 • SHOW CREATE TABLE company;

16 27:15

Form Editor Navigate: 1 / 1

Table: company

Create Table:

```
CREATE TABLE `company` (
  `id` varchar(15) NOT NULL,
  `company_name` varchar(255) DEFAULT NULL,
  `phone` varchar(15) DEFAULT NULL,
  `email` varchar(100) DEFAULT NULL,
  `country` varchar(100) DEFAULT NULL,
  `website` varchar(255) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci
```

Descripción:

En esta imagen se muestra la sentencia completa para la creación de la tabla `transaction`.

Aspectos clave:

- Definición de la clave primaria.
- Definición de la clave foránea hacia la tabla `company`.
- Creación de un índice sobre `company_id`.

```
19 • SELECT
20   TABLE_NAME,
21   COLUMN_NAME,
22   CONSTRAINT_NAME,
23   REFERENCED_TABLE_NAME,
24   REFERENCED_COLUMN_NAME
25 FROM information_schema.KEY_COLUMN_USAGE
26 WHERE TABLE_SCHEMA = 'transactions'
27   AND REFERENCED_TABLE_NAME IS NOT NULL
28 ORDER BY TABLE_NAME, COLUMN_NAME;
```

| Field Types | | | | | | | | |
|-------------|-----------------------|--------------------|-------|---------|---------------|--------------|-----------|-------|
| # | Field | Schema | Table | Type | Character Set | Display Size | Precision | Scale |
| 1 | TABLE_NAME | information_schema | | VARCHAR | utf8mb4 | 64 | 11 | 0 |
| 2 | COLUMN_NAME | information_schema | | VARCHAR | utf8mb4 | 64 | 10 | 0 |
| 3 | CONSTRAINT_NAME | information_schema | | VARCHAR | utf8mb4 | 64 | 18 | 0 |
| 4 | REFERENCED_TABLE_NAME | information_schema | | VARCHAR | utf8mb4 | 64 | 7 | 0 |

Descripción:

Esta imagen presenta una vista global de todas las columnas de las tablas, incluyendo el tipo de dato y la información sobre claves.

```
32 •   SELECT *
33      FROM company
34      LIMIT 10;
```

Descripción:

Esta imagen muestra ejemplos de datos reales almacenados en la tabla `company`.

Aspectos destacados:

- Datos coherentes y realistas.
 - Información de contacto y país claramente identificable.

```
36 •   SELECT *
37     FROM transaction
38     LIMIT 10;
```

Descripción:

En esta imagen se observan ejemplos de transacciones almacenadas en la tabla `transaction`.

Aspectos destacados:

- Cada fila representa una transacción diferente.
- La relación con la empresa se establece mediante `company_id`.
- El campo `declined` indica el estado de la transacción.

```
27 •   SELECT TABLE_NAME,COLUMN_NAME,DATA_TYPE,IS_NULLABLE,COLUMN_KEY  
28     FROM information_schema.COLUMNS  
29     WHERE TABLE_SCHEMA = 'transactions'  
30     ORDER BY TABLE_NAME, ORDINAL_POSITION;  
31  
32  
33
```

| TABLE_NAME | COLUMN_NAME | DATA_TYPE | IS_NULLABLE | COLUMN_KEY |
|-------------|----------------|-----------|-------------|------------|
| company | id | varchar | NO | PRI |
| company | company_name | varchar | YES | |
| company | phone | varchar | YES | |
| company | email | varchar | YES | |
| company | country | varchar | YES | |
| company | website | varchar | YES | |
| transaction | id | varchar | NO | PRI |
| transaction | credit_card_id | varchar | YES | |
| transaction | company_id | varchar | YES | MUL |
| transaction | user_id | int | YES | |
| transaction | lat | float | YES | |
| transaction | longitude | float | YES | |
| transaction | timestamp | timestamp | YES | |
| transaction | amount | decimal | YES | |
| transaction | declined | tinyint | YES | |

Esta imagen muestra la **estructura completa del esquema `transactions`** mediante metadatos del sistema.

Permite identificar:

- Las tablas existentes
- Las columnas de cada tabla
- El tipo de dato de cada columna
- Si acepta valores NULL
- Si la columna es clave primaria o índice

Tabla `company`

- `id` es la **clave primaria**
- Es el único campo obligatorio (NOT NULL)
- El resto de campos son opcionales

Tabla `transaction`

- `id` es la clave primaria
- `company_id` aparece como **MUL**, lo que indica:
 - Columna indexada
 - Usada como **clave foránea**
- `amount` es `DECIMAL`, adecuado para importes económicos
- `declined` indica si la transacción fue rechazada

Informe final

Introducción

En este ejercicio se ha creado una base de datos relacional llamada `transactions`, que almacena información sobre empresas y transacciones realizadas en una plataforma de venta online. Todas las tablas y los datos han sido importados exclusivamente mediante sentencias SQL, sin utilizar asistentes gráficos.

Estructura general de la base de datos

La base de datos `transactions` está formada por dos tablas principales:

- `company`
- `transaction`

Ambas tablas están relacionadas mediante una clave foránea.

Tabla `company`

La tabla `company` almacena la información corporativa básica de las empresas que participan en las transacciones. Actúa como una tabla maestra dentro del esquema.

Campos:

- `id`: identificador único de la empresa (clave primaria)
- `company_name`: nombre de la empresa
- `phone`: número de teléfono
- `email`: correo electrónico
- `country`: país
- `website`: sitio web

Esta tabla no depende de ninguna otra.

Tabla `transaction`

La tabla `transaction` contiene la información relacionada con las operaciones económicas realizadas.

Campos:

- `id`: identificador único de la transacción (clave primaria)
- `credit_card_id`: identificador de la tarjeta de crédito
- `company_id`: identificador de la empresa (clave foránea)
- `user_id`: identificador del usuario
- `lat` y `longitude`: localización geográfica
- `timestamp`: fecha y hora de la transacción
- `amount`: importe de la operación
- `declined`: indica si la transacción fue rechazada

El campo `company_id` referencia al campo `id` de la tabla `company`.

Relación entre las tablas

Existe una relación **uno a muchos (1:N)**:

- Una empresa puede tener varias transacciones.
- Cada transacción pertenece a una única empresa.

Elección de tipos de datos

Los tipos de datos se han seleccionado según la naturaleza de la información almacenada. Se ha utilizado VARCHAR para campos de texto, DECIMAL(10, 2) para importes económicos y TIMESTAMP para registrar el momento exacto de cada transacción.

Integridad de los datos

La integridad de los datos se garantiza mediante el uso de claves primarias en ambas tablas y una clave foránea en la tabla transaction, evitando referencias a empresas inexistentes.

Uso de la base de datos

Esta base de datos permite realizar distintos análisis, como el número de transacciones por empresa, el importe total de las operaciones, el análisis temporal de las transacciones y la detección de transacciones rechazadas.

Diagrama

Se ha incluido un diagrama entidad-relación (ER) generado con MySQL Workbench para representar gráficamente la estructura de la base de datos y las relaciones entre las tablas.

Exercici 2

Utilitzant JOIN realitzaràs les següents consultes:

- Llistat dels països que estan generant vendes.

```
SELECT DISTINCT country
  FROM company      AS c
  JOIN transaction AS t ON t.company_id = c.id
 ORDER BY country ASC;
```

| country |
|----------------|
| Australia |
| Belgium |
| Canada |
| China |
| France |
| Germany |
| Ireland |
| Italy |
| Netherlands |
| New Zealand |
| Norway |
| Spain |
| Sweden |
| United Kingdom |
| United States |

Time Action Response Duration / Fetch Time

1 10:44:53 SELECT DISTINCT country FROM company AS c JOIN transaction AS t ON t.company_id = c.id ORDER BY country ASC 15 row(s) returned 0.019 sec / 0.000008...

Enunciado: *Listado de los países que están generando ventas.*

La consulta identifica los **países asociados a empresas con transacciones registradas**, es decir, aquellos donde realmente se generan ventas.

Se utiliza un **JOIN** entre company y transaction para relacionar las empresas con sus transacciones y excluir aquellas que no han generado ventas.

El uso de `SELECT DISTINCT country` evita la repetición de países debido a múltiples transacciones o empresas del mismo país.

Finalmente, `ORDER BY country ASC` se aplica para ordenar el resultado alfabéticamente y mejorar la **legibilidad**.

- Des de quants països es generen les vendes.

```
50 •   SELECT COUNT(DISTINCT country) AS total_countries
51     FROM company
52       JOIN transaction
53             AS t ON t.company_id = c.id;
```

The screenshot shows a MySQL command-line interface. The query is:

```
50 •   SELECT COUNT(DISTINCT country) AS total_countries
51     FROM company
52       JOIN transaction
53             AS t ON t.company_id = c.id;
```

The results are displayed in a grid:

| total_countries |
|-----------------|
| 15 |

Below the grid, the status bar shows:

| Action | Time | Response | Duration / Fetch Time |
|------------|--|--|-------------------------|
| 1 11:19:41 | SELECT COUNT(DISTINCT country) AS total_countries FROM company | AS c JOIN transaction AS t ON t.com... 1 row(s) returned | 0.070 sec / 0.000005... |

Enunciado: ¿Desde cuántos países se generan las ventas?

La consulta tiene como objetivo **calcular el número de países desde los cuales se generan ventas**, es decir, países asociados a empresas que cuentan con **transacciones registradas**.

Se utiliza un **JOIN** entre las tablas company y transaction para considerar únicamente las empresas que han generado ventas y excluir aquellas sin transacciones.

El uso de `COUNT(DISTINCT country)` permite contar cada país una sola vez, evitando duplicados provocados por múltiples transacciones o varias empresas en un mismo país.

El alias `total_countries` facilita una **interpretación clara** del resultado obtenido.

- Identifica la compañía amb la mitjana més gran de vendes.

```
55 •   SELECT c.company_name, c.country,
56           ROUND(AVG(t.amount), 2) AS AVG_vendes
57     FROM company      AS c
58       JOIN transaction AS t ON t.company_id = c.id
59         Group BY c.company_name,c.country
60           ORDER BY AVG_vendes DESC
61           LIMIT 1;
62
```

The screenshot shows a MySQL command-line interface. The query is:

```
55 •   SELECT c.company_name, c.country,
56           ROUND(AVG(t.amount), 2) AS AVG_vendes
57     FROM company      AS c
58       JOIN transaction AS t ON t.company_id = c.id
59         Group BY c.company_name,c.country
60           ORDER BY AVG_vendes DESC
61           LIMIT 1;
62
```

The results are displayed in a grid:

| company_name | country | AVG_vendes |
|---------------------------|---------|------------|
| Ac Fermentum Incorporated | Germany | 284.87 |

Below the grid, the status bar shows:

| Action | Time | Response | Duration / Fetch Time |
|------------|---|-------------------|-------------------------|
| 1 11:47:34 | SELECT c.company_name, c.country, ROUND(AVG(t.amount), 2) AS AVG_vendes FROM company AS c JOIN transaction A... | 1 row(s) returned | 0.255 sec / 0.000009... |

Enunciado: Identifica la compañía con la media más alta de ventas.

La consulta tiene como objetivo **identificar la empresa que presenta la mayor media de ventas**, calculada a partir del importe de sus transacciones registradas.

Se utiliza un **JOIN** entre las tablas `company` y `transaction` para relacionar cada empresa con sus ventas y considerar únicamente aquellas que tienen transacciones asociadas.

La función `AVG(t.amount)` permite calcular la media de ventas por empresa y `ROUND(..., 2)` se aplica para redondear el resultado a dos decimales, ya que se trabaja con importes monetarios.

La cláusula `GROUP BY c.company_name, c.country` agrupa las transacciones por empresa, mientras que `ORDER BY AVG_vendas DESC` junto con `LIMIT 1` permite seleccionar la compañía con la media de ventas más alta.

Exercici 3

Utilitzant només subconsultes (sense utilitzar JOIN):

- Mostra totes les transaccions realitzades per

```

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

| <code>id</code> | <code>credit_card...</code> | <code>company_id</code> | <code>user_id</code> | <code>lat</code> | <code>longitude</code> | <code>timestamp</code> | <code>amount</code> | <code>declined</code> |
|---------------------------------------|-----------------------------|-------------------------|----------------------|------------------|------------------------|------------------------|---------------------|-----------------------|
| 001301B0-3B04-4D0C-0154-E2B5FEB0A0E9 | CcU-3070 | b-2222 | 409 | 41.5014 | 2.10170 | 2020-12-17 10:15:37 | 310.90 | 0 |
| 00201A11-2E62-44C4-941D-198FC8DB77F0 | CcU-3512 | b-2222 | 193 | 55.5704 | -3.65129 | 2021-01-22 23:44:27 | 453.04 | 0 |
| 00235618-0A5C-D4D9-9DCB-B3A9405D8923 | CcS-8137 | b-2222 | 3556 | 59.8421 | 18.729 | 2020-09-09 15:43:19 | 263.14 | 0 |
| 005A5A7B-1F1A-4B6C-9B15-1625A78C9C38 | CcS-8998 | b-2222 | 4417 | 41.1591 | -8.63905 | 2024-05-15 09:10:11 | 442.01 | 0 |
| 00687139-48B2-4FFA-8E73-B20376F04AB4 | CcS-4870 | b-2222 | 289 | 51.1966 | 10.4669 | 2019-03-09 19:37:49 | 524.84 | 0 |
| 0074F4DD-32F1-4827-8758-55896314623A | CcS-8081 | b-2222 | 3500 | 39.7016 | -8.50325 | 2016-12-26 23:06:57 | 491.90 | 0 |
| 00AA89CD-39D6-4DCB-8A1D-13BE73DC90A9 | CcS-6797 | b-2222 | 2216 | 55.7652 | -3.76245 | 2021-04-25 03:06:59 | 167.15 | 0 |
| 00BE09D4-6920-47D8-ABE8-325E2269829D | CcS-4983 | b-2222 | 402 | 38.708 | -9.12993 | 2019-02-27 15:25:16 | 141.66 | 0 |
| 00DA0383-E048-4577-8ED1-3C56C258FF2F | CcS-9223 | b-2222 | 4642 | 51.1742 | 10.2027 | 2019-03-21 11:47:34 | 325.62 | 0 |
| 00DD11DE-ED01-4BB0D-93A0-174D183A59DF | CcS-7681 | b-2222 | 3100 | 45.7565 | 4.83109 | 2024-01-28 18:20:49 | 242.53 | 0 |
| 01449CE0-98E9-4DE5-9810-728C6BA00E6F | CcS-5424 | b-2222 | 843 | 47.0163 | 2.26064 | 2024-02-17 19:37:14 | 451.71 | 0 |
| 0175E8C7-241E-42DA-A889-9F246DBF4D2F | CcS-7510 | b-2222 | 2929 | 52.0619 | 4.29464 | 2021-08-28 16:29:38 | 9.46 | 0 |
| 01ABDAB8-06E2-4CA0-A131-AEE6FF11B749 | CcS-5053 | b-2222 | 472 | 51.7738 | 5.17479 | 2020-01-28 01:15:07 | 388.41 | 0 |
| 01F1C7ED-0823-442D-AE0E-3134D5004866 | CcS-6776 | b-2222 | 2195 | 59.6697 | 18.6697 | 2022-12-17 09:40:14 | 168.79 | 0 |
| 01FABC61-18C8-441B-987C-459AA3F06097 | CcS-5531 | b-2222 | 950 | 55.3405 | -3.3863 | 2020-04-30 08:57:27 | 333.45 | 0 |
| 023B714C-DCC1-4B03-AD8E-29B3673F631B | CcS-5103 | b-2222 | 522 | 55.3362 | -2.99198 | 2018-04-14 18:52:13 | 155.29 | 0 |
| 024BBB812-89F5-4B21-B8E9-E0D51800A679 | CcS-9328 | b-2222 | 4747 | 42.0816 | 12.8326 | 2016-07-01 01:33:41 | 268.23 | 0 |
| 02DF083C-E622-42E0-B75E-15BA39A27594 | CcS-4859 | b-2222 | 278 | 60.6186 | 18.5278 | 2021-02-03 14:12:33 | 127.23 | 0 |
| 031C28DE-F9A2-448A-92EE-8700DBB6327C | CcU-3750 | b-2222 | 159 | 55.355 | -3.44811 | 2020-06-05 18:32:28 | 68.38 | 0 |
| 03217043-8465-46B4-9F2A-EDA02F6FFE1 | CcS-4998 | b-2222 | 417 | 39.3245 | -8.58552 | 2022-03-16 03:39:15 | 589.15 | 0 |
| 0333CE96-0EBA-40E2-8E27-506EFA14FCFB | CcS-7116 | b-2222 | 2535 | 52.3282 | 5.45831 | 2024-04-06 11:50:52 | 161.11 | 0 |

Enunciado: Utilizando solo subconsultas (sin usar JOIN): muestra todas las transacciones realizadas por empresas de Alemania.

La consulta tiene como objetivo **obtener todas las transacciones** correspondientes a empresas cuyo país es **Germany**, cumpliendo la condición del enunciado de utilizar **únicamente subconsultas** y no **JOIN**.

Se usa `WHERE company_id IN (subconsulta)` para filtrar las transacciones por las empresas que cumplen el criterio.

La subconsulta `SELECT id FROM company WHERE country = 'Germany'` devuelve los identificadores de las empresas alemanas, y el `IN` permite seleccionar todas las transacciones cuyo `company_id` pertenezca a ese conjunto.

- o Llista les empreses que han realitzat transaccions per un amount superior a la mitjana de totes les transaccions.

```

75 • SELECT (SELECT c.company_name
76   FROM company AS c
77     WHERE c.id = t.company_id) AS company_name, ROUND(t.amount, 2) AS amount
78   FROM transaction AS t
79   WHERE t.amount > (SELECT AVG(t2.amount)
80     FROM transaction AS t2)
81   ORDER BY amount DESC;
--
```

| company_name | amount |
|-----------------------------|--------|
| Nunc Ac PC | 876.66 |
| Nunc In Foundation | 858.60 |
| Nunc In Foundation | 858.60 |
| Magna Incorporated | 858.55 |
| Sed Est Corp. | 845.91 |
| Risus Associates | 825.14 |
| Dui Cras Associates | 824.65 |
| At Associates | 821.08 |
| Lorem Ipsum Dolor Corp. | 817.06 |
| Ac Industries | 811.46 |
| Rutrum Non Inc. | 800.45 |
| Integer Mollis Corp. | 794.91 |
| Mauris Incorporated | 784.46 |
| Vestibulum Lorem PC | 780.29 |
| Cras Vehicula Aliquet In... | 772.04 |
| Arcu LLP | 764.19 |
| Interdum Feugiat Sed A... | 763.37 |
| Cras Vehicula Aliquet In... | 759.81 |
| Mauris Institute | 759.35 |
| Pede Cum Ltd | 755.00 |
| Fringilla LLC | 754.39 |

| | Time | Action | Response | Duration / Fetch Time |
|---|----------|---|----------|-----------------------|
| 1 | 12:19:11 | SELECT (SELECT c.company_name FROM company AS c WHERE c.id = t.company_id) AS company_name, ROUND(t.amo... 46722 row(s) returned | | 0.087 sec / 0.041 sec |

Enunciado:

Utilizando solo subconsultas (sin utilizar JOIN), listar las empresas que han realizado transacciones con un importe superior a la media de todas las transacciones.

La consulta tiene como objetivo **identificar las empresas que han realizado transacciones cuyo importe es superior a la media global de todas las transacciones**, respetando la restricción del enunciado de no utilizar JOIN y trabajar únicamente con **subconsultas**.

La subconsulta (SELECT AVG(t2.amount) FROM transaction t2) calcula la **media total de los importes** y se utiliza en la cláusula WHERE para filtrar únicamente aquellas transacciones que superan dicho valor.

Para obtener el nombre de la empresa sin usar JOIN, se emplea una **subconsulta correlacionada en el SELECT**, que permite recuperar company_name a partir del company_id asociado a cada transacción.

Además, se aplica ROUND(t.amount, 2) para mostrar los importes con dos decimales, ya que se trata de valores monetarios.

Existía una alternativa que consistía en seleccionar los nombres de las empresas desde la tabla company y filtrar en el WHERE mediante una subconsulta con los identificadores de las empresas que tenían transacciones superiores a la media. Sin embargo, tras analizar ambas opciones, se optó por la solución actual, que aunque es **ligeramente más compleja y utiliza dos subconsultas**, permite **mostrar directamente el importe de cada transacción**, facilitando una interpretación más clara del resultado.

- o Eliminaran del sistema les empreses que no tenen transaccions registrades, entrega el llistat d'aquestes empreses.

```

85 •   SELECT *
86   FROM company AS c
87   WHERE NOT EXISTS (SELECT 1
88     FROM transaction AS t
89     WHERE t.company_id = c.id);
90
91   ##check
92 •   SELECT COUNT(*) AS companies_without_tx
93   FROM company      AS c
94   WHERE NOT EXISTS (SELECT 1
95     FROM transaction AS t
96     WHERE t.company_id = c.id);
97
    
```

Result Grid | Filter Rows: Search | Edit: | Export/Import: |

| id | company_name | phone | email | country | website |
|------|--------------|-------|-------|---------|---------|
| HULL | HULL | HULL | HULL | HULL | HULL |

Time Action Response Duration / Fetch Time

| | | | | |
|---|----------|--|-------------------|-------------------------|
| 1 | 12:35:31 | SELECT * FROM company AS c WHERE NOT EXISTS (SELECT 1 FROM transaction AS t WHERE t.company_id = c.id) | 0 row(s) returned | 0.011 sec / 0.000007... |
|---|----------|--|-------------------|-------------------------|


```

91   ##check
92 •   SELECT COUNT(*) AS companies_without_tx
93   FROM company      AS c
94   WHERE NOT EXISTS (SELECT 1
95     FROM transaction AS t
96     WHERE t.company_id = c.id);
97
    
```

Result Grid | Filter Rows: Search | Export: |

| companies_without... |
|----------------------|
| 0 |

Time Action Response Duration / Fetch Time

| | | | | |
|---|----------|---|-------------------|-------------------------|
| 1 | 12:35:31 | SELECT * FROM company AS c WHERE NOT EXISTS (SELECT 1 FROM transaction AS t WHERE t.company_id = c.id) | 0 row(s) returned | 0.011 sec / 0.000007... |
| 2 | 12:43:48 | SELECT COUNT(*) AS companies_without_tx FROM company AS c WHERE NOT EXISTS (SELECT 1 FROM transaction AS t WHERE t.company_id = c.id) | 1 row(s) returned | 0.0037 sec / 0.00001... |

Enunciado: Eliminarán del sistema las empresas que no tienen transacciones registradas; entrega el listado de estas empresas.

La consulta tiene como objetivo **identificar las empresas que no tienen ninguna transacción registrada**, utilizando subconsultas y sin emplear JOIN.

Se utiliza NOT EXISTS para comprobar que **no existe ninguna transacción asociada** a cada empresa en la tabla transaction. La subconsulta se evalúa de forma correlacionada y se detiene en cuanto encuentra una coincidencia, lo que la hace **más eficiente en términos operativos**.

La expresión SELECT 1 se emplea porque en las cláusulas EXISTS / NOT EXISTS solo importa la existencia de filas y no los valores devueltos.

Dado que el resultado de la consulta devuelve **cero filas**, se añadió una consulta de comprobación con COUNT(*) para verificar que **no existen empresas sin transacciones** y confirmar que el resultado vacío es correcto.

Como alternativa, también era posible utilizar una subconsulta con NOT IN, por ejemplo:

```

WHERE id NOT IN (
  SELECT DISTINCT company_id
  FROM transaction
);
```

Sin embargo, tras comentarlo con un compañero, se decidió utilizar NOT EXISTS, ya que en escenarios reales cada operación tiene un **coste computacional**. EXISTS finaliza la búsqueda en cuanto encuentra la primera coincidencia, mientras que NOT IN necesita evaluar **todo el conjunto de resultados**, lo que puede implicar un mayor coste en tiempo y recursos. Por este motivo, se optó por NOT EXISTS como la opción más adecuada.

Nivell 2

Exercici 1

Identifica els cinc dies que es va generar la quantitat més gran d'ingressos a l'empresa per vendes. Mostra la data de cada transacció juntament amb el total de les vendes.

```
102 •   SELECT
103      DATE(t.timestamp)          AS fecha_venta,
104      c.company_name            AS nombre_empresa,
105      ROUND(SUM(t.amount), 2)    AS total_ingeresos
106  FROM transaction           AS t
107  JOIN company                AS c ON t.company_id = c.id
108 WHERE t.declined = 0
109     AND amount IS NOT NULL
110     AND timestamp IS NOT NULL
111 GROUP BY DATE(t.timestamp), c.company_name
112 ORDER BY total_ingeresos DESC
113 LIMIT 5;
```

| | fecha_venta | nombre_empresa | total_inger... |
|--|-------------|-------------------------------|----------------|
| | 2024-12-02 | Eget Ipsum Ltd | 3398.40 |
| | 2019-03-19 | Ac Fermentum Incorporated | 3149.04 |
| | 2017-12-20 | Nulla Integer Vulputate Corp. | 2774.04 |
| | 2017-12-20 | Viverra Donec Foundation | 2773.35 |
| | 2020-12-11 | Ac Fermentum Incorporated | 2717.76 |

```
1 13:06:49 SELECT DATE(t.timestamp) AS fecha_venta, c.company_name AS nombre_empresa, ROUND(SUM(t.amount), 2)... 5 row(s) returned 0.300 sec / 0.000021...
```

Enunciado:

Identifica los cinco días en los que se generó la mayor cantidad de ingresos por ventas y muestra la fecha junto con el total de ventas.

La consulta tiene como objetivo **identificar los cinco días con mayor volumen de ingresos**, calculando el total de ventas a partir de las transacciones registradas.

Se utiliza un **JOIN** entre las tablas `transaction` y `company` para relacionar cada transacción con su empresa correspondiente.

La función `DATE(t.timestamp)` permite agrupar las transacciones por día, mientras que `SUM(t.amount)` calcula el total de ingresos diarios.

Se filtran únicamente las transacciones válidas mediante `t.declined = 0` y se excluyen valores nulos en `amount` y `timestamp` para garantizar la **consistencia y fiabilidad de los resultados**.

Finalmente, los resultados se ordenan de mayor a menor ingreso y se limitan a los **cinco días con mayor facturación**.

Durante una revisión conjunta, se planteó la posibilidad de omitir las condiciones `IS NOT NULL` asumiendo que los datos ya estarían validados previamente. Sin embargo, se decidió mantener estos filtros en la consulta principal, ya que en **entornos reales los datos pueden cambiar con el tiempo** y no siempre es posible verificar manualmente su integridad antes de cada análisis. De este modo, la consulta resulta **más robusta y**

Exercici 2

Quina és la mitjana de vendes per país? Presenta els resultats ordenats de major a menor mitjà.

```
117 •  SELECT c.country,
118      ROUND(AVG(t.amount), 2) AS media_ventas
119  FROM company AS c
120  JOIN transaction AS t ON t.company_id = c.id
121  GROUP BY c.country
122  ORDER BY media_ventas DESC;
123
```

| country | media_ventas |
|----------------|--------------|
| Australia | 265.19 |
| United States | 264.98 |
| Belgium | 261.15 |
| Germany | 260.84 |
| Ireland | 260.64 |
| Spain | 260.47 |
| France | 259.98 |
| New Zealand | 259.59 |
| Norway | 259.38 |
| Netherlands | 258.44 |
| Italy | 258.27 |
| Sweden | 257.44 |
| Canada | 257.43 |
| United King... | 256.76 |
| China | 252.42 |

```
1 13:20:00 SELECT c.country, ROUND(AVG(t.amount), 2) AS media_ventas FROM company AS c JOIN transaction AS t ON t.company_id... 15 row(s) returned 0.150 sec / 0.000017...
```

Enunciado:

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

La consulta tiene como objetivo **calcular la media de ventas para cada país**, utilizando los importes de las transacciones asociadas a las empresas.

Se emplea un **JOIN** entre las tablas `company` y `transaction` para relacionar cada transacción con el país de la empresa correspondiente.

La función `AVG(t.amount)` permite calcular la media de ventas por país y `ROUND(..., 2)` se utiliza para redondear el resultado a dos decimales, ya que se trata de valores monetarios.

Mediante `GROUP BY c.country` se agrupan las transacciones por país y `ORDER BY media_ventas DESC` ordena los resultados de mayor a menor media, tal como solicita el enunciado.

En esta consulta **no se han filtrado las transacciones rechazadas (declined) ni los valores nulos**, ya que el enunciado no especifica ninguna condición adicional sobre la validez de las ventas y únicamente solicita calcular la **media de ventas por país**.

No obstante, en un contexto real de análisis de datos, podría ser recomendable añadir filtros como `t.declined = 0 o t.amount IS NOT NULL` para asegurar que solo se tengan en cuenta transacciones válidas. En este caso, se ha optado por **no incluir estos filtros** para responder de forma estricta a lo que pide el enunciado y evitar introducir supuestos no solicitados.

Exercici 3

En la teva empresa, es planteja un nou projecte per a llançar algunes campanyes publicitàries per a fer competència a la companyia "Non Institute". Per a això, et demanen la llista de totes les transaccions realitzades per empreses que estan situades en el mateix país que aquesta companyia.

- Mostra el llistat aplicant JOIN i subconsultes.

```
130 •   SELECT company_name,company_id,t.*  
131     FROM transaction AS t  
132     JOIN company      AS c ON t.company_id=c.id  
133     WHERE c.country IN ( SELECT c2.country  
134                           FROM company AS c2  
135                             WHERE c2.company_name = 'Non Institute' );|  
136
```

| company_name | company_id | credit_card... | company_id | user_id | lat | longitude | timestamp | amount | declined | |
|--------------|------------|--------------------------------------|------------|---------|------|-----------|-----------|---------------------|----------|---|
| Sed Nunc Ltd | b-2240 | 00B72D94-39A3-4B8C-B13F-2E05733AAT7A | CcS-6913 | b-2240 | 3894 | 55.0212 | -3.7340 | 2017-10-20 22:00:20 | 414.00 | 0 |
| Sed Nunc Ltd | b-2246 | 01F075B1-D7AE-4D02-AAD9-5FFD72A43F3C | CcS-8700 | b-2246 | 4119 | 55.856 | -3.15783 | 2018-01-27 13:44:36 | 103.73 | 0 |
| Sed Nunc Ltd | b-2246 | 023FFCE8-E618-4938-BF56-CBDF80540ADD | CcS-7816 | b-2246 | 3235 | 46.3568 | 1.82755 | 2016-12-19 11:53:45 | 219.28 | 0 |
| Sed Nunc Ltd | b-2246 | 02683BEB-EF91-4564-957B-D6F1662AB7C5 | CcS-9471 | b-2246 | 4890 | 42.1332 | 12.396 | 2017-01-10 21:09:29 | 326.87 | 0 |
| Sed Nunc Ltd | b-2246 | 02C2F29E-CEF2-4C1E-A594-F476E8F279C0 | CcS-9082 | b-2246 | 4501 | 39.4662 | -0.373246 | 2020-05-24 01:17:29 | 155.72 | 0 |
| Sed Nunc Ltd | b-2246 | 02F468DC-426C-47C2-B80A-D8B25B7A8101 | CcS-6913 | b-2246 | 2332 | 52.175 | 19.3508 | 2023-03-17 16:36:27 | 305.35 | 0 |
| Sed Nunc Ltd | b-2246 | 0306BE3B-817B-4A49-934E-0E439291A104 | CcS-5302 | b-2246 | 721 | 51.9233 | 18.926 | 2021-12-02 23:06:02 | 339.58 | 0 |
| Sed Nunc Ltd | b-2246 | 0347BFE6-8EB5-4387-B187-0E78E8F2B8FB | CcS-7674 | b-2246 | 3093 | 45.768 | 4.84271 | 2021-12-30 08:40:24 | 172.93 | 0 |
| Sed Nunc Ltd | b-2246 | 03AEBD0E-DC97-4BD3-9C57-6A6DB78026FD | CcS-6121 | b-2246 | 1540 | 50.8113 | 10.3145 | 2018-11-11 22:49 | 114.77 | 0 |
| Sed Nunc Ltd | b-2246 | 03CA36D3-88FF-4DBF-BFD4-4CC7DAA4ED2B | CcS-8036 | b-2246 | 3455 | 52.5178 | 13.4131 | 2017-02-25 15:38:21 | 440.27 | 0 |
| Sed Nunc Ltd | b-2246 | 04494182-96DD-42EB-82FE-5F92C5210537 | CcS-6791 | b-2246 | 2210 | 41.9542 | 12.4607 | 2018-05-17 17:53:53 | 241.59 | 0 |
| Sed Nunc Ltd | b-2246 | 045AACF6-FF85-49FB-9DE4-E6730655366A | CcS-5363 | b-2246 | 782 | 39.2464 | -7.90454 | 2018-08-09 22:12:54 | 188.58 | 0 |
| Sed Nunc Ltd | b-2246 | 0489FDAE-86A4-4929-954E-9A2272189CAC | CcS-7296 | b-2246 | 2715 | 51.1647 | 10.7348 | 2021-03-30 04:29:22 | 205.45 | 0 |
| Sed Nunc Ltd | b-2246 | 0490C36A-4B02-4CED-B33B-80B3AC86C6C8 | CcS-7539 | b-2246 | 2958 | 46.4281 | 1.64603 | 2023-05-20 19:28:00 | 424.64 | 0 |
| Sed Nunc Ltd | b-2246 | 04A0ABDE-FB25-4E59-A901-B4B879380A33 | CcS-4871 | b-2246 | 290 | 52.0589 | 5.55327 | 2022-11-04 05:35:28 | 461.34 | 0 |
| Sed Nunc Ltd | b-2246 | 0512193E-B639-49FE-B1F2-45609B409F9D | CcS-6651 | b-2246 | 2070 | 39.684 | -8.57102 | 2024-07-08 02:51:38 | 82.15 | 0 |

| Time | Action | Response | Duration / Fetch Time |
|----------|--|-----------------------|------------------------|
| 13:28:08 | SELECT company_name,company_id,t.* FROM transaction AS t JOIN company AS c ON t.company_id=c.id WHERE c.country... 13776 row(s) returned | 13776 row(s) returned | 0.0037 sec / 0.046 sec |

Enunciado:

Listar todas las transacciones realizadas por empresas situadas en el mismo país que la compañía "Non Institute", aplicando JOIN y subconsultas.

La consulta tiene como objetivo **obtener todas las transacciones** de empresas ubicadas en el **mismo país** que "Non Institute", cumpliendo el requisito de utilizar **JOIN** y **subconsulta**.

Se utiliza un **JOIN** entre `transaction` y `company` para relacionar cada transacción con la información de su empresa (incluido el país).

La condición `WHERE c.country IN (subconsulta)` filtra las empresas cuyo país coincide con el país (o países) obtenido en la subconsulta. La subconsulta recupera el/los país(es) asociados a la empresa "Non Institute" desde la tabla `company`, y el operador `IN` permite contemplar escenarios en los que la empresa pueda estar registrada en **más de un país**, evitando limitar el resultado a un único valor.

Como alternativa, se podría haber utilizado `WHERE c.country = (...)` junto con `LIMIT 1` en la subconsulta. Sin embargo, se priorizó `IN` frente a `+ LIMIT 1` para no perder información si "Non Institute" tuviera presencia en más de un país.

En el `SELECT`, se eligió `t.*` para conservar **toda la información de la transacción**, en lugar de seleccionar únicamente `t.amount`, ya que el enunciado pide el listado de transacciones y puede ser útil disponer de todos los campos para análisis posteriores.

Durante la revisión con un compañero surgió la posibilidad de excluir a la propia empresa "Non Institute" del resultado. No obstante, el enunciado indica "empresas situadas en el mismo país que

esta compañía”, lo que podría incluir también a la propia compañía. En un entorno real, sería recomendable **confirmar este punto con el solicitante** o clarificar el criterio de exclusión antes de modificar el resultado.

- Mostra el llistat applicant solament subconsultes.

```

138 •  SELECT (SELECT c.company_name
139   FROM company c
140   WHERE c.id = t.company_id) AS company_name,
141   (SELECT c.country
142   FROM company c
143   WHERE c.id = t.company_id) AS country , t.*
144   FROM transaction AS t
145   WHERE t.company_id IN (SELECT c1.id
146     FROM company c1
147     WHERE c1.country IN (SELECT c2.country
148       FROM company AS c2
149       WHERE c2.company_name = 'Non Institute')
150   );
151

```

| Result Grid | | | | | | | | | | | |
|---------------|----------------|---------------------------------------|----------------|------------|---------|---------|-----------|---------------------|--------|----------|--|
| company_name | country | id | credit_card... | company_id | user_id | lat | longitude | timestamp | amount | declined | |
| Non Institute | United Kingdom | 00C20270-9500-4D80-A2D8-49A004C229FD | CcS-9004 | b-2010 | 4403 | 55.4021 | -0.371200 | 2010-03-24 00:00:04 | 150.00 | V | |
| Non Institute | United Kingdom | 00CA0053-4E47-4F80-AE4E-0E7AB6D65CFF | CcS-8527 | b-2618 | 3946 | 46.5307 | 2.27639 | 2016-09-19 22:11:41 | 437.09 | 0 | |
| Non Institute | United Kingdom | 00D7ECC1-F470-40CD-8DD0-8A7F709F76C7 | CcS-8926 | b-2618 | 4345 | 46.5688 | 1.93599 | 2020-01-02 13:10:34 | 345.92 | 0 | |
| Non Institute | United Kingdom | 00DC1967-4260-458B-926A-35DB317433C5 | CcS-6473 | b-2618 | 1892 | 52.3388 | 5.7368 | 2023-05-05 07:46:56 | 62.62 | 0 | |
| Non Institute | United Kingdom | 00E4DA17-35F6-4D61-99E7-B317455DB1B6 | CcS-8954 | b-2618 | 4373 | 42.2102 | 12.1252 | 2017-08-11 11:12:53 | 92.91 | 0 | |
| Non Institute | United Kingdom | 0147854E-0AEB-4A32-B26A-4F0C1B73C051 | CcS-6381 | b-2618 | 1800 | 51.7762 | 5.56833 | 2018-11-15 17:00:14 | 270.93 | 0 | |
| Non Institute | United Kingdom | 0173575D-6BE2-4450-AFED-9568E3549FBA | CcS-9255 | b-2618 | 4674 | 39.0251 | -7.97634 | 2021-11-20 11:00:18 | 30.14 | 0 | |
| Non Institute | United Kingdom | 0175A14D-921F-4D71-B3E8-AE2B4074DC2 | CcS-6339 | b-2618 | 1758 | 41.3898 | 2.16622 | 2018-05-11 07:19:30 | 493.06 | 0 | |
| Non Institute | United Kingdom | 018FC5FC-6619-4D45-BBEC-445886FCD42 | CcS-7065 | b-2618 | 2484 | 60.1232 | 18.4976 | 2023-06-30 09:37:11 | 526.48 | 0 | |
| Non Institute | United Kingdom | 01F7B8C6-F4C0-4D83-94D3-4DDA41893C0 | CcS-6388 | b-2618 | 1807 | 41.7263 | 13.0113 | 2018-03-02 05:58:33 | 226.34 | 0 | |
| Non Institute | United Kingdom | 020614D3-07AE-47FD-BD47-2B83442C894F | CcS-7922 | b-2618 | 3341 | 39.448 | -8.52514 | 2024-08-05 10:51:44 | 189.21 | 0 | |
| Non Institute | United Kingdom | 023D1462-1AEC-4B71-B65C-3BA1EF496688 | CcS-8011 | b-2618 | 3430 | 47.0052 | 1.6806 | 2016-04-14 13:29:06 | 203.06 | 0 | |
| Non Institute | United Kingdom | 02745D9B-D43B-47B8-9565-227705EE0B5C | CcS-6141 | b-2618 | 1560 | 41.3848 | 2.16505 | 2020-12-28 08:27:42 | 305.94 | 0 | |
| Non Institute | United Kingdom | 027DDE15-DE25-452F-B2AE-012399740C61 | CcS-5884 | b-2618 | 1303 | 55.5014 | -3.39182 | 2020-02-08 23:56:36 | 370.40 | 0 | |
| Non Institute | United Kingdom | 02861656-6AFB-42F0-B6E8-E497E074345 | CcS-7190 | b-2618 | 2609 | 60.3176 | 18.5818 | 2024-07-18 04:51:02 | 302.53 | 0 | |
| Non Institute | United Kingdom | 02A2FF8A-F6B2-4D88-9559-A5AACD2A2DC42 | CcS-7316 | b-2618 | 2735 | 39.9932 | -3.4031 | 2024-04-08 22:24:49 | 184.95 | 0 | |
| Non Institute | United Kingdom | 02E9E454-6A2E-4AA6-B500-FA659F618984 | CcS-9555 | b-2618 | 4974 | 39.5137 | -8.29864 | 2017-08-04 11:28:02 | 257.80 | 0 | |
| Non Institute | United Kingdom | 03144600-9658-4198-B3F3-8C27AEC44CE5 | CcS-7676 | b-2618 | 3095 | 38.9117 | -7.7944 | 2024-10-17 21:04:49 | 504.24 | 0 | |
| Non Institute | United Kingdom | 032EE997-E60A-41EA-EE9942E0A8E | CcS-6100 | b-2618 | 1489 | 55.5024 | 2.04470 | 2022-03-03 00:40:10 | 140.05 | 0 | |

1 13:41:28 SELECT (SELECT c.company_name FROM company c WHERE c.id = t.company_id) AS company_name, (SELECT... 13776 row(s) returned 0.0073 sec / 0.124 sec

Enunciado:

Listar todas las transacciones realizadas por empresas situadas en el mismo país que la compañía “Non Institute”, utilizando solamente subconsultas (sin JOIN).

La consulta tiene como objetivo **obtener el listado completo de transacciones** de empresas ubicadas en el **mismo país** que la compañía “Non Institute”, cumpliendo la restricción del enunciado de no utilizar **JOIN** y trabajar únicamente con **subconsultas**.

Para mostrar información de la empresa sin emplear **JOIN**, se utilizan **subconsultas correlacionadas en el **SELECT****, que permiten recuperar el `company_name` y el `country` a partir del `company_id` de cada transacción.

El filtrado principal se realiza mediante `WHERE t.company_id IN (...)`, donde una subconsulta anidada obtiene primero el/los país(es) de “Non Institute” y, a partir de ellos, selecciona los identificadores de todas las empresas situadas en esos mismos países.

Existía una alternativa que consistía en utilizar únicamente `t.*` en el `SELECT`, lo que habría evitado el uso de subconsultas en esa parte de la consulta. Sin embargo, esta opción no permitía mostrar el **nombre de las empresas** en el resultado final y, dado que no se podía utilizar **JOIN**, se optó por emplear subconsultas en el `SELECT` para incluir dicha información.

Nivell 3

Exercici 1

Presenta el nom, telèfon, país, data i amount, d'aquelles empreses que van realitzar transaccions amb un valor comprès entre 350 i 400 euros i en alguna d'aquestes dates: 29 d'abril del 2015, 20 de juliol del 2018 i 13 de març del 2024. Ordena els resultats de major a menor quantitat.

```
158 •  SELECT c.company_name      AS nombre_empresa,
159          c.phone            AS telefono,
160          c.country         AS pais,
161          DATE(t.timestamp) AS fecha_transaccion,
162          t.amount          AS importe
163  FROM company           AS c
164  JOIN transaction        AS t  ON t.company_id = c.id
165 WHERE t.amount >= 350
166   AND t.amount <= 400
167   AND DATE(t.timestamp) IN ('2015-04-29',
168                           '2018-07-20',
169                           '2024-03-13')
170 ORDER BY t.amount DESC;
```

| nombre_empresa | telefono | pais | fecha_transacci... | importe |
|-------------------------|----------------|----------------|--------------------|---------|
| 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 |
| Pede Cum Ltd | 07 62 26 48 38 | Norway | 2018-07-20 | 356.87 |
| Auctor Mauris Vel LLP | 08 09 28 74 14 | United States | 2024-03-13 | 353.75 |

Time Action Response Duration / Fetch Time
1 13:53:59 SELECT c.company_name AS nombre_empresa, c.phone AS telefono, c.country AS pais, DATE(t.timestamp... 8 row(s) returned 0.050 sec / 0.000011...

Enunciado:

Presentar 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 las siguientes fechas: 29/04/2015, 20/07/2018 y 13/03/2024. Ordenar los resultados de mayor a menor cantidad.

La consulta tiene como objetivo **listar las transacciones que cumplen simultáneamente un rango de importe y un conjunto concreto de fechas**, mostrando además la información básica de la empresa.

Se utiliza un **JOIN** entre las tablas `company` y `transaction` para relacionar los datos de la empresa con sus transacciones.

El filtro `t.amount >= 350 AND t.amount <= 400` limita los resultados al rango de importes solicitado. Como alternativa, se podría haber utilizado `BETWEEN 350 AND 400`, ya que esta cláusula **incluye ambos valores extremos**. Tras comentarlo con un compañero, se decidió mantener esta condición explícita para dejar claro que los valores **350 y 400 están incluidos** en el rango.

La condición `DATE(t.timestamp) IN (...)` selecciona únicamente las transacciones realizadas en las fechas indicadas en el enunciado.

Finalmente, `ORDER BY t.amount DESC` ordena los resultados de mayor a menor importe, tal como se solicita.

Exercici 2

Necessitem optimitzar l'assignació dels recursos i dependrà de la capacitat operativa que es requereixi, per la qual cosa et demanen la informació sobre la quantitat de transaccions que realitzen les empreses, però el departament de recursos humans és exigent i vol un llistat de les empreses on especificquis si tenen més de 400 transaccions o menys.

```
178 •  SELECT c.company_name AS nombre_empresa,
179      COUNT(t.id)    AS cantidad_transacciones,
180      CASE
181          WHEN COUNT(t.id) > 400 THEN 'Mas de 400'
182          ELSE '400 o menos'
183      END AS categoria
184  FROM company c
185  LEFT JOIN transaction AS t ON t.company_id = c.id
186          AND t.declined = 0
187  GROUP BY c.id, c.company_name
188  ORDER BY cantidad_transacciones DESC;
```

| Result Grid | | |
|-------------------------------------|------------------------|------------|
| nombre_empresa | cantidad_transacciones | categoria |
| Ac Fermentum Incorporated | 2400 | Mas de 400 |
| Nunc Interdum Incorporated | 1599 | Mas de 400 |
| Donec Fringilla PC | 1590 | Mas de 400 |
| Mauris Institute | 1583 | Mas de 400 |
| Aliquet Vel Vulputate Incorporated | 1581 | Mas de 400 |
| Rutrum Non Inc. | 1578 | Mas de 400 |
| Aliquam PC | 1573 | Mas de 400 |
| Orci Adipiscing Limited | 1565 | Mas de 400 |
| Etiam Bibendum Fermentum Industries | 1564 | Mas de 400 |
| Mus Aenean Eget Foundation | 1563 | Mas de 400 |
| Auctor Mauris Corp. | 1561 | Mas de 400 |
| Enim Condimentum Ltd | 1549 | Mas de 400 |
| Justo Eu Arcu Ltd | 1545 | Mas de 400 |
| Non Institute | 1541 | Mas de 400 |
| Dolor Vitae Limited | 1541 | Mas de 400 |
| Interdum Feugiat Sed Associates | 1541 | Mas de 400 |
| At Pede Corp. | 1541 | Mas de 400 |
| Netus Et Malesuada Ltd | 1540 | Mas de 400 |

| Time | Action | Response | Duration / Fetch Time |
|------------|--|----------|-------------------------|
| 1 13:59:56 | SELECT c.company_name AS nombre_empresa, COUNT(t.id) AS cantidad_transacciones, CASE WHEN COUNT(t.id) > 4... 100 row(s) returned | | 0.156 sec / 0.000067... |

Enunciado:

Se necesita optimizar la asignación de recursos según la capacidad operativa. Recursos Humanos solicita un listado de empresas indicando cuántas transacciones realizan y si tienen más de 400 o 400 o menos.

La consulta tiene como objetivo **contar las transacciones por empresa** y **clasificarlas** según si superan las 400 transacciones o no, para apoyar la planificación de recursos.

Se utiliza un `LEFT JOIN` entre `company` y `transaction` para incluir **todas las empresas**, incluso aquellas sin transacciones (que resultarían en un conteo 0).

El filtro `t.declined = 0` se aplica en la condición del `JOIN` para considerar únicamente transacciones **válidas (no rechazadas)** sin perder empresas que podrían no tener transacciones válidas.

`COUNT(t.id)` calcula la cantidad de transacciones por empresa y `GROUP BY` permite realizar la agregación a nivel de empresa.

La expresión `CASE` crea la categoría solicitada por RR.HH. (`Mas de 400 / 400 o menos`) en función del conteo agregado.

Finalmente, `ORDER BY cantidad_transacciones DESC` ordena el resultado para visualizar primero las empresas con mayor carga operativa.