SPRINT 2

NIVELL 1

✅ ***1.1.-*** *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.*

La BD está formada por dos tablas; ***company*** y ***transaction***, donde se relacionan las transacciones que efectúan un conjunto de empresas y facilitan información tanto de las empresas como de las mismas transacciones.

La relación entre ambas es de 1:n en el sentido ***company->transaction***, es decir, pueden existir varias transacciones realizadas por una única empresa.

La tabla *company* está formada por ***100 registros*** con ***6 campos (o atributos)*** de información cada uno de ellos acerca de las empresas candidatas a haber realizado transacciones.

```
3 •  describe company;
```

100% 1:3

Result Grid Filter Rows: Search Export:

Field	Type	Null	Key	Default	Extra
id	varchar(15)	NO	PRI	NULL	
company_name	varchar(255)	YES		NULL	
phone	varchar(15)	YES		NULL	
email	varchar(100)	YES		NULL	
country	varchar(100)	YES		NULL	
website	varchar(255)	YES		NULL	

- ***id***: varchar() con capacidad para 15 caracteres. Cada id lo forman dígitos y el carácter especial '-'. Actúa como PK (Primary Key). Es un código asociado a cada empresa.
- ***company_name***: varchar() con capacidad para 255 caracteres. Cada company_name se compone básicamente por varias palabras (mayúsculas y minúsculas) y espacios entre ellas. Es el nombre de la empresa.
- ***phone***: varchar() con capacidad para 15 caracteres. Están formados por distintas combinaciones numéricas separadas entre ellas por espacios. es el teléfono de cada una de las empresas.
- ***email***: varchar() con capacidad para 100 caracteres. Compuestos por distintas combinaciones de letras y los caracteres especiales '.' y '@'. Es la dirección de correo electrónico de cada una de las empresas.
- ***country***: varchar() con capacidad para 100 caracteres. Formados tan solo por letras. indica el país al que pertenece la empresa.
- ***website***: varchar() con capacidad para 255 caracteres. Compuestos por distintas combinaciones de letras y los caracteres especiales '.','/' y '/'. Indica la dirección web de cada empresa.

Por otra parte la tabla *transaction* la componen ***597 registros*** con ***9 campos (o atributos)*** que facilitan información acerca de las transacciones realizadas.

```
1 •  describe transaction;
```

100% 1:1

Result Grid Filter Rows: Search Export:

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

Result 2

Action Output

	Time	Action	Response
1	11:43:34	describe transaction	9 row(s) returned

- ***id***: varchar() con capacidad para 255 caracteres. Cada id lo forman dígitos, letras (mayúsculas) y el carácter especial '-'. Actúa como PK (Primary Key). Es un código asociado a cada transacción.
- ***credit_card_id***: varchar() con capacidad para 15 caracteres. Cada credit_card_id se compone de letras (mayúsculas y minúsculas) y un guión '-'. Es un código asociado a cada una de la tarjetas de crédito.
- ***company_id***: varchar() con capacidad para 20 caracteres. Están formados por letras (minúsculas) y un guión '-'. Actúa como FK (Foreingn Key) para relacionarse con la tabla company. Es un código asociado a cada empresa.
- ***user_id***: int. Compuestos básicamente por la combinación de 3 o 4 dígitos que indican el usuario que realizó la transacción.
- ***lat***: float. Valor numérico en coma flotante, que indica la latitud en la que se encuentra la empresa que realizó la venta.
- ***longitude***: float. Valor numérico en coma flotante, que indica la longitud en la que se encuentra la empresa que realizó la venta.
- ***timestamp***: timestamp. Variable del tipo DATA que actúa como indicador de fecha y hora y debe hacer referencia al momento en el que se realizó cada una de las transacciones.
- ***amount***: decimal(10,2). Variable del tipo decimal (mayor precisión que el float) con capacidad para 10 dígitos de los que 2 serán decimales (detrás del punto). Hace referencia la importe de la venta.
- ***declined***: tinyint(1). Variable del tipo 'integer pequeño', utilizado para rangos inferiores al de un int. En este caso e rango es de 1, e indica si la transacción se pudo realizar (1) o no (0).

Ambas se relacionan a través de de la Primary Key (***PK***) de la tabla *company* y la Foreing Key (***FK***) de la tabla *transaction*.
La ***PK*** de la tabla *company* es el campo ***id***, único en toda la tabla y que identifica inequívocamente a cada una de las empresas.
La ***PK*** de la tabla *transaction* también es el campo identificado como ***id***, pero esta vez identifica cada una de las transacciones que contiene la tabla.

2.- Utilitzant JOIN realitzaràs les següents consultes:

✅*2.1.- Llistat dels països que estan fent compres.*

El SELECT **DISTINC** evita duplicar los datos a consultar.
La JOIN permite relacionar las tablas *company* y *transaction* a través de la **PK** de company y la **FK** de transaction, de esa forma es posible mostrar el nombre de las empresas.

```
3   -- Llistat dels països que estan fent compres.
4   select distinct country as 'PAISES QUE REALIZAN COMPRAS'
5   from company
6   join transaction on company.id=transaction.company_id
7   order by country desc;
8
```

100% 4:3

Result Grid Filter Rows: Search Export:

PAISES QUE REALIZAN COMPRAS
United States
United Kingdom
Sweden
Spain
Norway
New Zealand
Netherlands
Italy
Ireland
Germany
France
China
Canada
Belgium
Australia

Result 1

Action Output

	Time	Action	Response
1	15:31:46	select distinct country as 'PAISES QUE REALIZAN COMPRAS' from company join transaction on company.id=transaction.company_id order by country desc	15 row(s) returned

✅*2.2.- Des de quants països es realitzen les compres.*

```
9   -- 2.2. Des de quants països es realitzen les compres (utilitzant JOIN):
10  select count(distinct country) AS 'Nº PAISES QUE REALIZAN COMPRAS'
11  from company
12  join transaction on company.id=transaction.company_id;
13
```

100% 4:9

Result Grid Filter Rows: Search Export:

Nº PAISES QUE REALIZAN COMPRAS
15

Result 31

Action Output

	Time	Action	Response
1	11:30:37	select count(distinct country) AS 'Nº PAISES QUE REALIZAN COMPRAS' fr...	1 row(s) returned

con la **JOIN** tan solo se realiza la consulta sobre registros comunes en ambas tablas. Con la función de agregación **COUNT** en este caso, combinada con un **DISTINCT** (para no duplicar valores) obtenemos el resultado.

✅***2.3.-** Identifica la companyia amb la mitjana més gran de vendes.*

A través de la **JOIN** entre ambas tablas, de company consulto el nombre y de la tabla transaction la media de ventas.

```
14   -- 2.3. Identifica la companyia amb la mitjana més gran de vendes (utilitzant JOIN):
15   select company_name,round(avg(amount),2) as 'MEDIA VENTAS'
16   from transaction
17   join company on transaction.company_id=company.id
18   group by company_name
19   order by avg(amount) desc
20   limit 1;
21
```

100% 4:14

Result Grid Filter Rows: Search Export:

company_name	MEDIA VENTAS
Eget Ipsum Ltd	473.08

Result 33

Action Output

	Time	Action	Response
1	11:36:53	select company_name,round(avg(amount),2) as 'MEDIA VENTAS' from transactio...	1 row(s) returned

Ordenando la consulta por las medias calculadas de mayor a menor (**DESC**) me permite seleccionar la primera (mediante la instrucción **LIMIT**) y devolver tan solo la empresa con media más alta.

3.- Utilitzant només subconsultes (sense utilitzar JOIN):

❌ ***3.1.-*** *Mostra totes les transaccions realitzades per empreses d'Alemanya.*

```
27   -- 3.1. Mostra totes les transaccions realitzades per empreses d'Alemanya (sense utilitzar JOIN).
28   select id                                    -- Sacar todos los id de empresas cuyo país es Alemania
29   from company
30   where country='Germany';
31
32   select *
33   from transaction
34   where company_id in (select id                  -- Filtrar incluyendo la query anterior.
35                        from company
36                        where country='Germany');
37
```

100% 1:26

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

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
108B1D1D-5B23-A76C-55EF-C568E49A05DD	CcU-2938	b-2222	275	83.7839	-178.86	2021-07-07 17:43:16	293.57	0
EA2C3281-C9C1-A387-44F8-729FB4B51C76	CcU-2938	b-2222	275	20.2004	-116.84	2021-05-09 10:25:08	119.36	1
0DD2E608-5C9E-D1B3-4999-B99F43AD735A	CcU-2959	b-2234	275	9.68811	130.282	2021-04-17 05:30:17	252.47	1
AB069F53-965E-A2A8-CE06-CA8C4FD92501	CcU-2959	b-2234	275	1.64819	-158.007	2021-04-15 13:37:18	60.99	0
0466A42E-47CF-8D24-FD01-C0B689713128	CcU-4219	b-2302	170	-43.9695	-117.525	2021-07-26 07:29:18	49.53	0
0A476ED9-0C13-1962-F87B-D3563924B539	CcU-4359	b-2302	221	-56.4901	114.801	2022-02-26 20:33:54	430.49	0
122DC333-E19F-D629-DCD8-9C54CF1EBB9A	CcU-4366	b-2302	221	29.6372	-166.173	2021-06-09 06:04:14	172.01	0
135267BA-2E7D-957C-C42C-6450A2B3ED54	CcU-4520	b-2302	210	20.6724	14.9732	2021-12-29 20:38:23	17.97	0
14CAE5B5-8FB1-3E4A-4C85-0EA4167534F4	CcU-4849	b-2302	189	-53.6202	93.0533	2021-12-31 00:29:42	388.04	0
158A3ACB-541C-DBCC-65BD-6373CC67BF1C	CcU-4849	b-2302	183	42.5424	-170.347	2022-03-08 05:02:19	240.29	0
162C7E78-2B6B-7971-A1E4-D2124E732451	CcU-4527	b-2302	210	-69.1381	58.0017	2021-04-11 05:59:18	231.26	0
1717FD6B-ADAD-7082-A748-9112BE892CCC	CcU-4219	b-2302	172	69.4892	-138.411	2021-12-29 16:18:54	249.91	0
1753A288-9FC1-52E6-5C39-A1FFB97B0D3A	CcU-4345	b-2302	222	57.9422	-114.729	2021-08-17 05:32:08	497.84	0
186F53DE-DE27-B1FE-B82F-15B61CEB7726	CcU-4310	b-2302	225	-72.7448	36.6211	2021-12-20 13:13:45	238.16	0
18C4E2DD-1E4C-F35E-2198-C660BD81DC25	CcU-4849	b-2302	177	68.0133	91.4839	2021-09-24 18:55:25	237.04	0
18CCBA7C-ABC1-813D-FAF3-4BAB97429368	CcU-4219	b-2302	173	51.3881	-156.371	2021-06-21 03:21:34	58.16	0

transaction 3

Action Output

	Time	Action	Response
1	16:16:53	select * from transaction where company_id in (select id -- Filtrar incluyendo la query anterior. from company where country='Germany')	118 row(s) returned

Con la ***primera query*** devuelvo todos las empresas que residen en Alemania. Esta misma consulta servirá de sub-query para la consulta definitiva.

En la ***segunda query***, la búsqueda es sobre la tabla transaction aplicando como filtro la primera query y la combinación de la sentencia ***WHERE*** y el operador ***IN***.

MENTOR: Aquí deberías mostrar todas las columnas, no solo id, cuando se dicen todas las transacciones es con su id, el valor, etc.

CORRECCIÓN: tal y como muestra la captura de pantalla, se muestran todos los campos de la tabla transaction realizando un *SELECT **.

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

En la ***primera query*** (consulta) sacamos la media de todas las transacciones. Utilizo la función ***round*** para limitar los decimales de la media devuelta.

En la ***segunda query***, devuelvo los id de las empresas residentes en Alemania, utilizando como parte del filtro la primera consulta realizada.

La tercera query (definitiva), devuelve el nombre asociado a los id obtenidos en la consulta anterior.
La instrucción ***WHERE*** combinada con el operador ***IN*** permite realizar el filtro.

```
38    -- 3.2. Llista les empreses que han realitzat transaccions per un amount superior a la mitjana de totes les transaccions (sense utilitzar JOIN)
39    select avg(amount)                          -- calculo de la media total de amount de todas las transacciones
40    from transaction;
41
42    select company_id
43    from transaction                            -- id de empresas con amount superior a la media total
44    where amount>(select avg(amount)
45                  from transaction);
46
47    select company_name
48    from company
49    where id in (select company_id
50                 from transaction
51                 where amount>(select avg(amount)
52                               from transaction));
53
```

100% 1:37

Result Grid Filter Rows: Search Export:

company_name
Ac Fermentum Incorporated
Magna A Neque Industries
Fusce Corp.
Ante Iaculis Nec Foundation
Donec Ltd
Sed Nunc Ltd
Nascetur Ridiculus Mus Inc.
Vestibulum Lorem PC
Gravida Sagittis LLP

company 13

Action Output

	Time	Action	Response
1	16:32:32	select company_name from company where id in (select company_id from transaction where amount>(select avg(amount) from transaction))	70 row(s) returned

MENTOR: En este caso que comparas, quizás sea mejor no redondear amount.

CORRECCIÓN: tal y como muestra la captura de pantalla, retiro el comando *round* que precedía a la operación de agregación *avg(amount)*.

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

```
55   -- 3.3. Eliminaran del sistema les empreses que no tenen transaccions registrades, entrega el llistat d'aquestes empreses (sense utilitzar JOIN).
56   select company_id                    -- Listado de empresas que si han realizado transacciones
57   from transaction
58   group by company_id
59   having count(id)!=0;
60
61   select company_name
62   from company
63   where id not in (select company_id      -- filtro mediante el cual obtengo las empresas que no han realizado transacciones
64                    from transaction
65                    group by company_id
66                    having count(id)!=0);
67
68
```

100% 1:54

Result Grid Filter Rows: Search Export:

company_name

company 31

Action Output

	Time	Action	Response
1	16:45:51	select company_name from company where id not in (select company_id -- filtro mediante el cual obtengo las empresas que no han realizado trans...	0 row(s) returned

~~En este caso todas las empresas han realizado mínimo una transacción.~~
~~En la primera query saco una tabla que indica el número de transacciones que ha realizado cada empresa.~~
~~El filtro having nos discrimina tan solo aquellas empresas de la nueva tabla que no tienen ningún valor en el campo de las transacciones realizadas (null).~~
~~En la segunda query con el filtro WHERE combinado con EXISTS busco coincidencias entre la primera consulta y la segunda.~~

MENTOR:
- Esta query está mal planteada.
- HAVING count(id) = 'NULL' es incorrecto. Las funciones de agregación como COUNT nunca devuelven NULL, devuelven 0 como mínimo. Además, comparar con 'NULL' (entre comillas) es un error de tipo.
- El uso de EXISTS con HAVING no es adecuado aquí. Deberías hacer una subconsulta que obtenga los IDs de las empresas que sí tienen transacciones, y luego usar NOT IN para identificar las que no tienen.

CORRECCIÓN: tal y como muestra la captura de pantalla, interpreto la consulta de forma distinta.
Primero obtengo los id de empresas que han realizado alguna transacción *(count(id)!=0)*.
Utilizo esa consulta para anidarla en la segunda dentro del filtro que me permitirá obtener el nombre de las empresas.

NIVELL 2

***2.1.-** 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.*

```
-- 2.1.- 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.
select cast(timestamp as date) as FECHA, sum(amount) as INGRESOS   -- Convierto el tipo de fecha de timestamp a data par eliminar la hora
from transaction                                                    -- y consulto la suma de ingresos por fecha
group by FECHA
order by INGRESOS desc
limit 5;
```

100% 4:73

Result Grid Filter Rows: Search Export:

FECHA	INGRESOS
2021-03-29	1564.87
2021-12-20	1532.36
2021-06-15	1469.90
2021-05-09	1463.73
2021-06-21	1443.11

Result 25

Action Output

	Time	Action	Response
1	10:23:50	select cast(timestamp as date) as FECHA, sum(amount) as INGRESOS from transaction group by FECHA order by INGRESOS desc limit 5	5 row(s) returned

Convierto el tipo de dato que indica la fecha de la tabla *transaction* de ***timestamp a data*** para eliminar la parte horaria y poder agrupar por fecha tan solo. Para ello utilizo la función **CAST**.
Una vez agrupado por fecha lo ordeno de mayor a menor y devuelvo tan solo los 5 primeros registros.

```
select date(timestamp) as FECHA, sum(amount) as INGRESOS   -- Convierto el tipo de fecha de timestamp a data par eliminar la hora
from transaction                                            -- y consulto la suma de ingresos por fecha
group by FECHA
order by INGRESOS desc
limit 5;

-- 2.2.- Quina és la mitjana de vendes per país? Presenta els resultats ordenats de major a menor mitjà.
select country,avg(amount) as 'MEDIA DE VENTAS'
from company
```

100% 1:83

Result Grid Filter Rows: Search Export:

FECHA	INGRESOS
2021-03-29	1564.87
2021-12-20	1532.36
2021-06-15	1469.90
2021-05-09	1463.73
2021-06-21	1443.11

Result 33

Action Output

	Time	Action	Response
1	16:58:26	select date(timestamp) as FECHA, sum(amount) as INGRESOS -- Convierto el tipo de fecha de timestamp a data par eliminar la hora from transaction...	5 row(s) returned

MENTOR: Podrías considerar utilizar DATE(timestamp) en lugar de CAST(... AS DATE) como una alternativa más común y legible, aunque ambas son correctas.

CORRECCIÓN: tal y como muestra la captura de pantalla, realizo de nuevo la consulta pero esta vez sin castear la fecha sino convirtiéndola al tipo *DATE*. El resultado es el mismo.

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

```
-- 2.2.- Quina és la mitjana de vendes per país? Presenta els resultats ordenats de major a menor mitjà.
select country,avg(amount) as avgAmount
from company
join transaction on company.id=transaction.company_id
group by country
order by avgAmount desc;
```

100% 1:89

Result Grid Filter Rows: Search Export:

country	avgAmount
United States	309.179412
Ireland	277.308387
United Kingdom	270.731700
Canada	269.647869
Sweden	260.615063
Norway	254.216324
Netherlands	253.017778
Germany	244.203220
Australia	232.052500
Belgium	228.147692
China	227.556667
New Zealand	222.277273
Italy	201.590000
France	179.198333
Spain	53.400000

Result 38

Action Output

	Time	Action	Response
1	17:03:10	select country,avg(amount) as avgAmount from company join transaction on company.id=transaction.company_id group by country order by avgAmou...	15 row(s) returned

Agrupo por el campo *country* de la tabla *company* y mediante la JOIN devuelvo la media de los ingresos mediante la función de agregación **avg**.

MENTOR: El ORDER BY 'MEDIA DE VENTAS' está mal escrito. Al usar comillas simples, estás ordenando por una cadena de texto fija, no por el alias.

CORRECCIÓN: tal y como muestra la captura de pantalla, cambio el alias de la función de agregación *avg(amount)* por uno más apropiado *(avgAmount)* y evito confusiones con los entrecomillados.

2.3.- *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.*

2.3.1.- *Mostra el llistat aplicant JOIN i subconsultes.*

```
100   -- 2.3.1.- Mostra el llistat aplicant JOIN i subconsultes.
101   select distinct country                                    -- Query para obtener el país al que pertenece la empresa Non Institute
102   from company
103   where company_name='Non Institute';
104
105   select transaction.id,company_name,country
106   from company
107   join transaction on company.id=transaction.company_id
108   where country=(select distinct country                     -- Filtro del país de la empresa Non Institute
109                  from company
110                  where company_name='Non Institute') and company_name!='Non Institute'; -- No devuelvo las de la empresa Non Institute
111
100%   1:99
```

Result Grid Filter Rows: Search Export:

id	company_name	country
2B928E1C-EC14-A760-0A75-871477649D6A	Sed Nunc Ltd	United Kingdom
ACD2011A-A2B1-C365-41E1-2AB00C65147A	Sed Nunc Ltd	United Kingdom
4334349E-CEB0-3D68-A4D4-FEB7718A1ACE	Non Magna LLC	United Kingdom
BC2B9A38-77B4-28CD-1FE8-14DED863E773	Non Magna LLC	United Kingdom
1479B3D2-B7BA-C7BB-4CE3-8D7C2DE85ABB	Enim Condimentum Ltd	United Kingdom
152598C2-029D-D684-4B66-91EDF393EBFF	Enim Condimentum Ltd	United Kingdom
1B636B58-A2E8-7C69-D9C9-C54535DAFD3B	Enim Condimentum Ltd	United Kingdom
20418DE5-B804-BE9B-BD7A-A95C1BFDBF5C	Enim Condimentum Ltd	United Kingdom
239B8576-6C0E-137A-C2F6-3180A188A2D3	Enim Condimentum Ltd	United Kingdom

Result 41

Action Output

	Time	Action	Response
1	17:09:45	select transaction.id,company_name,country from company join transaction on company.id=transaction.company_id where country=(select distinct co...	70 row(s) returned

En la ***primera query*** obtengo el país al que pertenece la empresa Non Institute.
En la ***segunda query***, mediante la JOIN uno ambas tablas de la base de datos y mediante la primera consulta a modo de **sub-query** filtro (***where***) la consulta por el país de la empresa Non Institute.

Decido no mostrar las transacciones realizadas por la empresa Non Institute con el ***AND*** en el filtro WHERE.

MENTOR: No necesitas hacer un JOIN en la subconsulta que obtiene el país. Es redundante porque con la tabla company basta.

CORRECCIÓN: tal y como muestra la captura de pantalla y como se indica en la corrección el uso de la *JOIN* para obtener el país no es necesaria y el resultado es el mismo.

✗ *2.3.2.- Mostra el llistat aplicant solament subconsultes.*

```
114   -- 2.3.2.- Mostra el llistat aplicant solament subconsultes.
115   select country                                  -- Devuelve el país al que pertenece la empresa Non Institute
116   from company
117   where company_name='Non Institute';
118
119   select id                                       -- Devuelve los id de empresas ubicadas enel mism país que Nopn Institute
120   from company
121   where country=(select country
122                  from company
123                  where company_name='Non Institute');
124
125   select *
126   from transaction
127   where company_id in (select id                  -- Filtro los id de empresas con el mismo id que las de la consulta anterior
128                        from company
129                        where country=(select country
130                                       from company
131                                       where company_name='Non Institute') and company_name!='Non Institute');
132
```

100% 1:113

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

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
ACD2011A-A2B1-C365-41E1-2AB00C65147A	CcU-2980	b-2246	275	-54.4792	-82.7974	2022-03-05 20:41:20	60.07	1
4334349E-CEB0-3D68-A4D4-FEB7718A1ACE	CcU-3092	b-2310	275	-20.4859	150.87	2021-05-03 22:37:23	458.74	0
BC2B9A38-77B4-28CD-1FE8-14DED863E773	CcU-3092	b-2310	275	-78.0295	18.5295	2021-10-18 07:27:35	477.95	1
1479B3D2-B7BA-C7BB-4CF3-8D7C2DE85ABB	CcU-2994	b-2326	133	66.2672	172.399	2021-08-09 00:58:07	309.45	0

transaction 44

Action Output

	Time	Action		Response
1	17:26:20	select * from transaction where company_id in (select id	-- Filtro los id de empresas con el mismo id que las de la consulta anterior...	70 row(s) returned

~~En la ***primera query*** obtengo el país al que pertenece la empresa Non Institute y que utilizo como *sub-query* de la consulta definitiva para filtrar las empresas que pertenecen al mismo país.~~
~~La segunda query está formada por la COMBINACIÓN de tablas (que no JOIN ni UNION) para devolver las transacciones realizadas desde el mismo país que Non Institute.~~

MENTOR:
- Aunque usas FROM company, transaction, esto se considera una forma implícita de JOIN, por lo que no es 100% "solo subconsultas". Lo ideal sería eliminar esa unión directa y reemplazarla con subconsultas más puras.
- Sugerencia: transforma la lógica usando WHERE company.id = (SELECT company_id ...) para hacer todo desde transaction.

CORRECCIÓN: tal y como muestra la captura de pantalla utilizo dos subconsultas dentro del a query principal.
En la primera obtengo el país al que pertenece la empresa Non Institute y en la segunda, como filtro la primera, la relación de compañías de ese mismo país.
Todo ese conjunto de código lo uso en el filtro de la consulta principal como filtro para obtener todos los datos referentes a las transacciones realizadas.

NIVELL 3

***3.1.-** Presenta el nom, telèfon, país, data i amount, d'aquelles empreses que van realitzar transaccions amb un valor comprès entre 100 i 200 euros i en alguna d'aquestes dates: 29 d'abril del 2021, 20 de juliol del 2021 i 13 de març del 2022. Ordena els resultats de major a menor quantitat.*

```
-- 3.1.- Presenta el nom, telèfon, país, data i amount, d'aquelles empreses que van realitzar transaccions amb un valor comprès entre 100 i 200 euros
-- i en alguna d'aquestes dates: 29 d'abril del 2021, 20 de juliol del 2021 i 13 de març del 2022. Ordena els resultats de major a menor quantitat.

select company_name,phone,country,date(timestamp),amount
from company
join transaction on company.id=transaction.company_id
where (amount between 100 and 200) and (date(timestamp) in ('2021-04-29','2021-07-20','2022-03-13'))
order by amount desc;
```

100% 1:134

Result Grid Filter Rows: Search Export:

company_name	phone	country	date(timestamp)	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 19

Action Output

	Time	Action	Response
1	10:42:33	select company_name,phone,country,date(timestamp),amount from company join transaction on company.id=transaction.company_i...	5 row(s) returned

~~En esta query con la JOIN accedo a datos de las dos tablas que forman la base de datos.~~
~~Mediante el filtro where selecciono los datos solicitados, por un lado con el operador BETWEEN para acotar el margen respecto a los valores de las transacciones y por otro lado con el operador LIKE combinado con la wildcard (o comodín) % para filtrar por la fechas indicadas.~~

MENTOR:
Estás usando LIKE para filtrar fechas en el campo timestamp, lo cual no es la forma más precisa ni eficiente. Estás haciendo una comparación de texto en lugar de comparar fechas. Es mejor convertir el timestamp a DATE y luego usar IN (...) o comparaciones directas.

CORRECCIÓN: realizo el cambio sugerido, cambiando el tipo de dato del campo timestamp a *date* (tanto en la *SELECT* como en el filtro *WHERE*.
De igual forma sustituyo la opción de comparación *LIKE* por *IN*.

3.2.- 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 especifiquis si tenen més de 4 transaccions o menys.

Mediante la sentencia ***CASE*** introduzco un nuevo campo a la consulta, en el que en función de la condición ***WHEN*** tiene un valor u otro.

```
-- 3.2.- 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 especifiquis si tenen més de 4 transaccions o menys.

select company_name, count(amount) as 'numTransacciones',
case
    when count(amount)>4 then 'Mas de 4 transacciones'
    when count(amount)<4 then 'Menos de 4 transacciones'
    else '4 transacciones'
end as 'notaTransacciones'
from company
join transaction on company.id=transaction.company_id
group by company_name
order by company_name asc;
```

100% 1:143

Result Grid Filter Rows: Search Export:

company_name	numTransacciones	notaTransacciones
Aliquet Sem Limited	2	Menos de 4 transacciones
Aliquet Vel Vulputate Incorporated	2	Menos de 4 transacciones
Amet Faucibus Ut Foundation	2	Menos de 4 transacciones
Amet Institute	2	Menos de 4 transacciones
Amet Lorem LLP	2	Menos de 4 transacciones
Amet Luctus Vulputate Foundation	2	Menos de 4 transacciones
Amet Nulla Donec Corporation	2	Menos de 4 transacciones
Ante Iaculis Nec Foundation	2	Menos de 4 transacciones
Arcu LLP	56	Mas de 4 transacciones
At Associates	2	Menos de 4 transacciones

Result 22

Action Output

	Time	Action	Response
1	10:47:52	select company_name, count(amount) as 'numTransacciones', case when count(amount)>4 then 'Mas de 4 transacciones' when count(amount...	100 row(s) returned