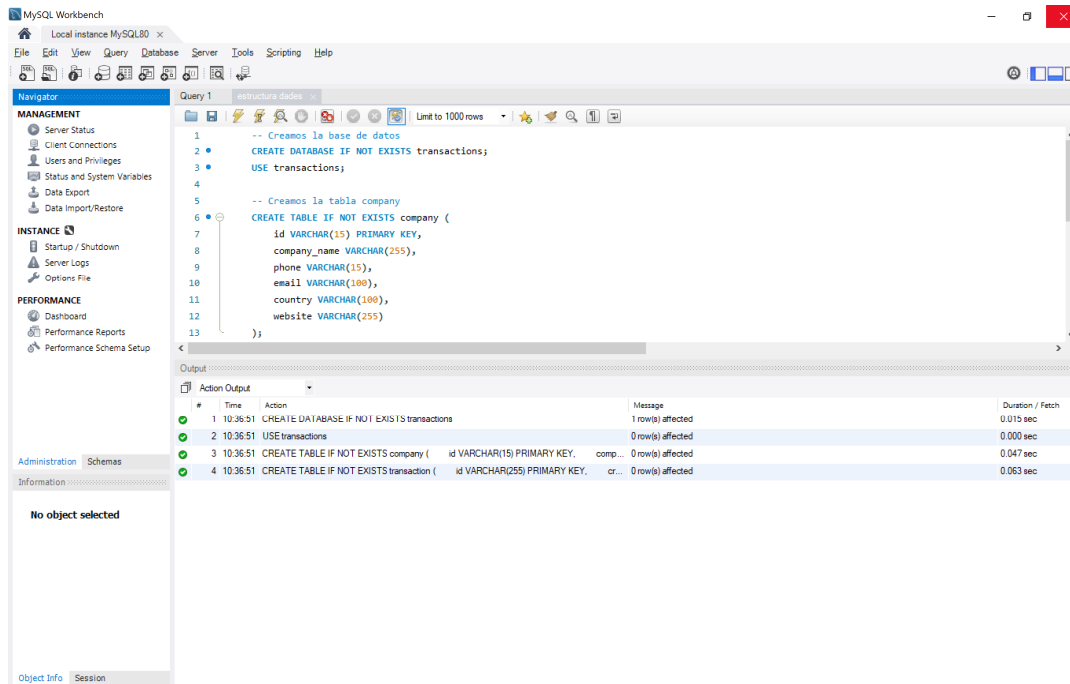


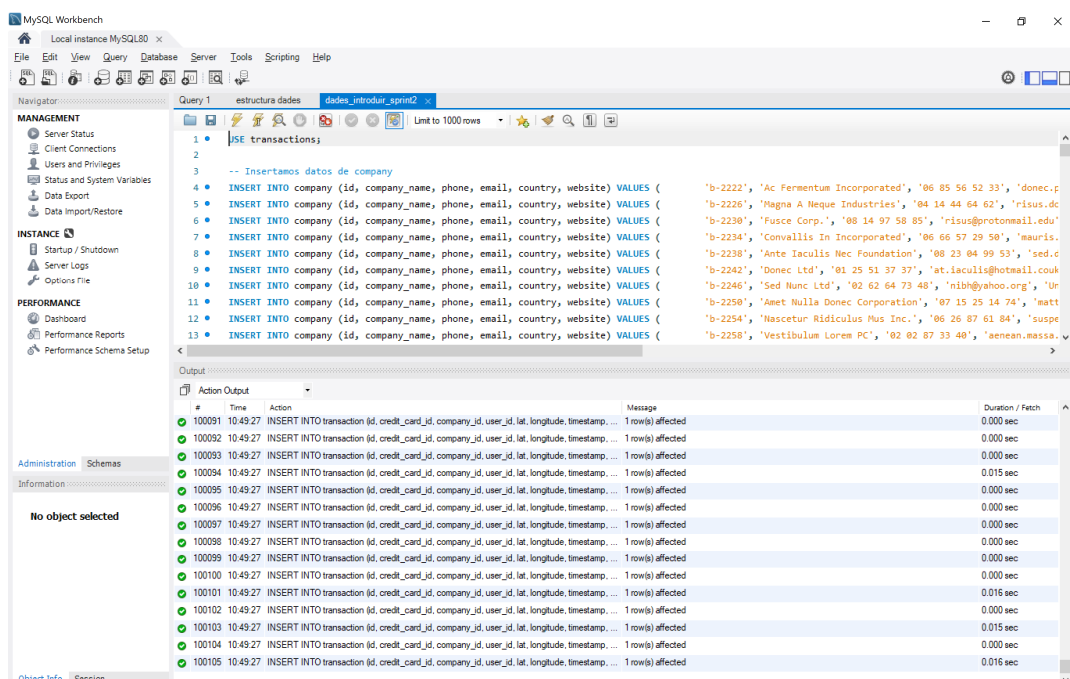
## NIVELL 01

## Exercici 1

Creació de les taules a partir del fitxer *estructura\_dades.sql* :



Creació dels registres a partir del fitxer *dades\_introduir\_sprint2.sql*:



Explicació estructura de dades:

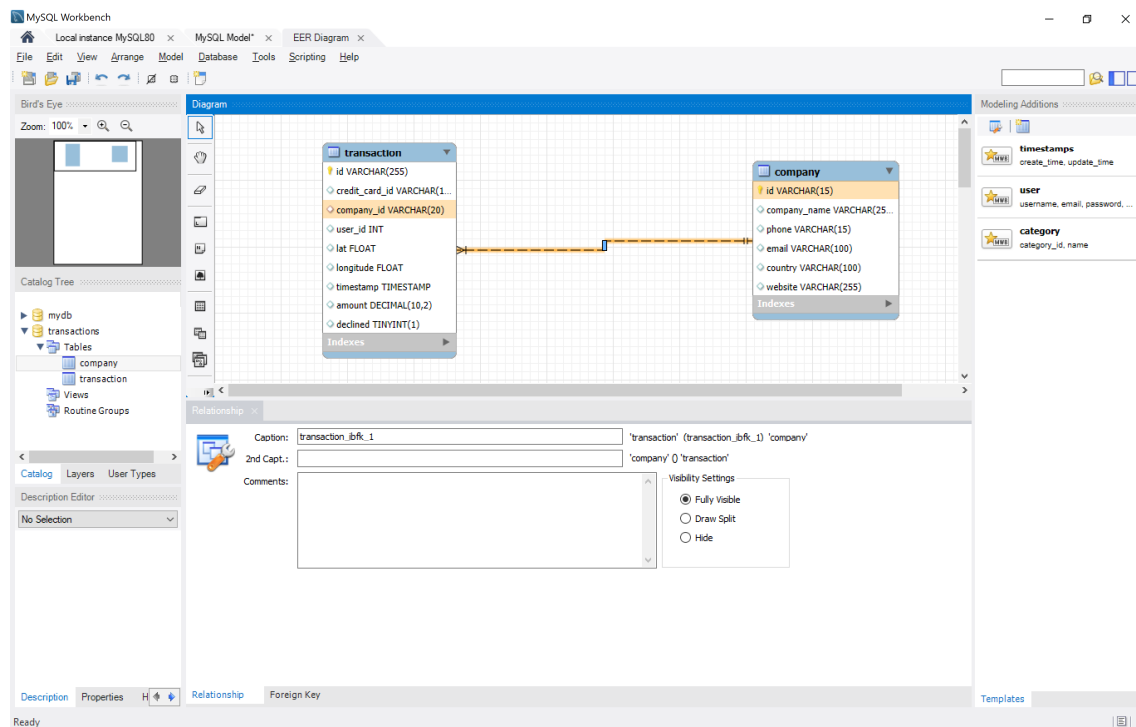
-- Taula **company**

id VARCHAR(15) PRIMARY KEY,	Clau primària única de tipus VARCHAR de mida 15
company_name VARCHAR(255),	Camp tipus VARCHAR de mida 255
phone VARCHAR(15),	Camp tipus VARCHAR de mida 15
email VARCHAR(100),	Camp tipus VARCHAR de mida 100
country VARCHAR(100),	Camp tipus VARCHAR de mida 100
website VARCHAR(255)	Camp tipus VARCHAR de mida 255

-- Taula **transaction**

id VARCHAR(255) PRIMARY KEY,	Clau primària única de tipus VARCHAR de mida 255
credit_card_id VARCHAR(15) REFERENCES credit_card(id),	Camp tipus VARCHAR de mida 15 que fa referència al camp <i>id</i> de la taula <i>credit_card</i>
company_id VARCHAR(20),	Camp tipus VARCHAR de mida 20
user_id INT REFERENCES user(id),	Camp tipus INT que fa referència al camp <i>id</i> de la taula <i>user</i>
lat FLOAT,	Camp tipus FLOAT
longitude FLOAT,	Camp tipus FLOAT
timestamp TIMESTAMP,	Camp tipus TIMESTAMP
amount DECIMAL(10, 2),	Camp tipus DECIMAL
declined BOOLEAN,	Camp tipus BOOLEAN
FOREIGN KEY (company_id) REFERENCES company(id)	Camp que fa referència al camp <i>id</i> de la taula <i>company</i> i a més és clau forana, per tan ha d'existir prèviament a la taula <i>company</i> perquè pugui se inserit en aquesta taula.

Diagrama:



Podem veure una relació de ***n a 1*** entre la taula ***transaction(n)*** i la taula ***company(1)*** creada sota la etiqueta ***transaction\_ibfk\_1***. Aquesta relació indica que pot haver una companyia/empresa en infinites transaccions dins de la taula ***transaction*** però al mateix temps ens indica que una companyia/empresa només pot estar un sol cop dins de la taula ***company***. A més a més hi ha una integritat de dades degut a que la taula ***transaction*** té una clau forana que fa referència al camp ***id*** de la taula ***company*** que ens assegura que la companyia/empresa ha d'existir prèviament a la taula ***company*** perquè pugui crear-se un registre nou a la taula ***transaction***.

**Exercici 2**

Utilitzant JOIN realitzaràs les següents consultes:

- Llistat dels països que estan generant vendes.

The screenshot shows MySQL Workbench with a query titled "consulta\_llistat dels països que...". The query is:

```
1 SELECT distinct(company_name)
2 FROM transaction
3 INNER JOIN company ON company.id = transaction.company_id;
```

The result grid displays a list of company names:

company_name
Ac Fermentum Incorporated
Magna A Neque Industries
Fusce Corp.
Convalles In Incorporated
Ante Jaculis Nec Foundation
Donec Ltd
Sed Nunc Ltd
Amet Nulla Donec Corporation
Nascetur Ridiculus Mus Inc.
Vestibulum Lorem PC
Gravida Sagittis LLP
Mus Aenean Eget Foundation

The bottom panel shows the execution log with the following output:

#	Time	Action	Message	Duration / Fetch
100154	13:01:08	SELECT count(distinct compa...	1 row(s) returned	0.110 sec / 0.000 sec
100155	13:01:18	SELECT count(distinct countr...	1 row(s) returned	0.109 sec / 0.000 sec
100156	13:02:13	SELECT count(distinct countr...	1 row(s) returned	0.188 sec / 0.000 sec
100157	13:03:12	SELECT distinct company_na...	100 row(s) returned	0.015 sec / 0.000 sec

- Des de quants països es generen les vendes.

The screenshot shows MySQL Workbench with a query titled "consulta\_llistat dels països que...". The query is:

```
1 SELECT count(distinct(country)) as països
2 FROM transaction
3 INNER JOIN company ON company.id = transaction.company_id;
```

The result grid displays the count of countries:

països
15

The bottom panel shows the execution log with the following output:

#	Time	Action	Message	Duration / Fetch
100154	13:01:08	SELECT count(distinct compa...	1 row(s) returned	0.110 sec / 0.000 sec
100155	13:01:18	SELECT count(distinct countr...	1 row(s) returned	0.109 sec / 0.000 sec
100156	13:02:13	SELECT count(distinct countr...	1 row(s) returned	0.188 sec / 0.000 sec
100157	13:03:12	SELECT distinct company_na...	100 row(s) returned	0.015 sec / 0.000 sec

**Exercici 3**

Utilitzant només subconsultes (sense utilitzar JOIN):

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

The screenshot shows the MySQL Workbench interface. The left sidebar displays the database schema, including tables like 'company' and 'transaction'. The main query editor contains the following SQL code:

```
1 SELECT count(*) as 'Transaccions empreses alemanyes'
2 FROM transaction
3 where transaction.company_id IN (select id from company where country = 'Germany')
```

The 'Result Grid' at the bottom shows the output of the query, displaying a single row with the count of transactions for German companies.

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

The screenshot shows the MySQL Workbench interface. The left sidebar displays the database schema. The main query editor contains the following SQL code:

```
1 SELECT distinct(company_name)
2 FROM company
3 where id IN (
4 select company_id from transaction where amount > (
5 select avg(amount) as ventes from transaction)
6 )
```

The 'Result Grid' at the bottom shows the output of the query, displaying a list of company names that have transactions with amounts greater than the average.

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

The screenshot shows the MySQL Workbench interface. The SQL Editor contains the following query:

```
1. SELECT distinct(company_name)
2. FROM company
3. where id NOT IN (
4. select company_id from transaction)
```

The left sidebar shows the database schema with the 'company' table selected. The 'company' table structure is displayed below the schema:

Columns:	
id	varchar(15) PK
company_name	varchar(255)
phone	varchar(15)
email	varchar(100)
country	varchar(100)
website	varchar(255)

The bottom panel shows the 'Output' tab with the following results:

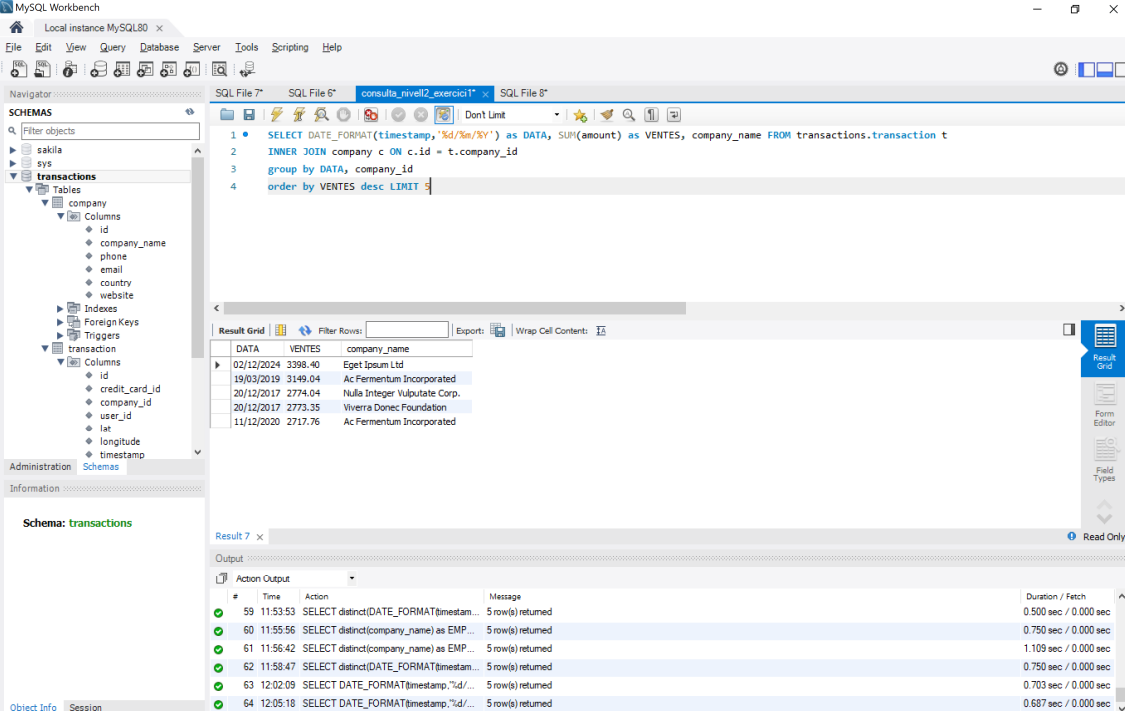
#	Time	Action	Message	Duration / Fetch
100175	13:33:58	SELECT * FROM transactions...	100000 row(s) returned	0.000 sec / 0.344 sec
100176	13:34:47	SELECT distinct(company_name...	0 row(s) returned	0.109 sec / 0.000 sec
100177	13:34:56	SELECT distinct(company_name...	100 row(s) returned	0.015 sec / 0.000 sec
100178	13:43:01	SELECT distinct(company_name...	0 row(s) returned	0.110 sec / 0.000 sec

The status bar at the bottom indicates the SQL script is saved to: "C:\LLUIS\Dropbox\IT ACADEMY\Sprint 2\consulta\_llistat empreses que no tenen transaccions registrades.sql".

## NIVELL 02

## 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.



The screenshot shows the MySQL Workbench interface. The SQL Editor contains the following query:

```
1 SELECT DATE_FORMAT(timestamp, '%d/%m/%Y') as DATA, SUM(amount) as VENTES, company_name FROM transactions.transaction t
2 INNER JOIN company c ON c.id = t.company_id
3 group by DATA, company_id
4 order by VENTES desc LIMIT 5
```

The Results window displays the following data:

DATA	VENTES	company_name
02/12/2024	3398.40	Eget Ipsum Ltd
19/03/2019	3149.04	Ac Fermentum Incorporated
20/12/2017	2774.04	Nulla Integer Vulputate Corp.
20/12/2017	2773.35	Viverra Donec Foundation
11/12/2020	2717.76	Ac Fermentum Incorporated

The bottom panel shows the Action Output with the following details:

#	Time	Action	Message	Duration / Fetch
59	11:53:53	SELECT distinct(DATE_FORMAT(timestamp, '%d/%m/%Y')) as DATA, SUM(amount) as VENTES, company_name FROM transactions.transaction t	5 row(s) returned	0.500 sec / 0.000 sec
60	11:55:56	SELECT distinct(company_name) as EMP...	5 row(s) returned	0.750 sec / 0.000 sec
61	11:56:42	SELECT distinct(company_name) as EMP...	5 row(s) returned	1.109 sec / 0.000 sec
62	11:58:47	SELECT distinct(DATE_FORMAT(timestamp, '%d/%m/%Y')) as DATA, SUM(amount) as VENTES, company_name FROM transactions.transaction t	5 row(s) returned	0.750 sec / 0.000 sec
63	12:02:09	SELECT DATE_FORMAT(timestamp, '%d/%m/%Y') as DATA, SUM(amount) as VENTES, company_name FROM transactions.transaction t	5 row(s) returned	0.703 sec / 0.000 sec
64	12:05:18	SELECT DATE_FORMAT(timestamp, '%d/%m/%Y') as DATA, SUM(amount) as VENTES, company_name FROM transactions.transaction t	5 row(s) returned	0.687 sec / 0.000 sec

A la consulta pot haver-hi varies transaccions en un mateix dia sobre una mateixa empresa, per això faig una suma de les vendes agrupades per data i per empresa ordenades pel total de la suma descendent, de major a menor, retornant el 5 primers registres que demana el exercici.

En aquest cas, encara que no ho demana l'exercici, també mostro el nom de la companyia per si es necessària aquesta informació per algun departament de la empresa.

He donat format a la data perquè només mostri el dia, el mes i l'any.

**Exercici 2**

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

The screenshot shows the MySQL Workbench interface. The SQL Editor contains the following query:

```
1 SELECT FORMAT(avg(amount),2) as MITJANA, country as PAIS FROM transactions.transaction t
2 INNER JOIN company c ON c.id = t.company_id
3 group by country
4 order by MITJANA desc
```

The Results window displays the following data:

MITJANA	PAIS
265.19	Australia
264.98	United States
261.15	Belgium
260.84	Germany
260.64	Ireland
260.47	Spain
259.98	France
259.59	New Zealand
259.38	Norway
258.44	Netherlands
258.27	Italy
257.44	Sweden

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

#	Time	Action	Message	Duration / Fetch
69	12:10:46	SELECT DATE_FORMAT(timestamp,"%d/%...)	5 row(s) returned	0.625 sec / 0.000 sec
70	12:11:06	SELECT DATE_FORMAT(timestamp,"%d/%...)	5 row(s) returned	0.968 sec / 0.000 sec
71	12:11:27	SELECT avg(amount) as MITJANA, countr...	15 row(s) returned	0.516 sec / 0.000 sec
72	12:11:47	SELECT avg(amount) as MITJANA, countr...	15 row(s) returned	0.437 sec / 0.000 sec
73	12:12:04	SELECT avg(amount) as MITJANA, countr...	15 row(s) returned	0.438 sec / 0.000 sec
74	12:13:32	SELECT FORMAT(avg(amount),2) as MITJ...	15 row(s) returned	0.421 sec / 0.000 sec

He creat una consulta perquè retorni la mitjana de les vendes amb la funció AVG(), amb un format de sortida de 2 decimals ordenat de major a menor mitjana per país.



**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.

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 SELECT company_name as EMPRESA, count(amount) as TRANSACCIONS FROM transactions.transaction t
2 INNER JOIN company c ON c.id = t.company_id
3 where country = (
4 SELECT country FROM transactions.company where company_name='Non Institute'
5 )
6 group by EMPRESA
7 order by TRANSACCIONS desc
```

The Results grid displays the following data:

EMPRESA	TRANSACCIONS
Orci Adipiscing Limited	1567
Enim Conditum Ltd	1550
Non Institute	1543
Interdum Feugiat Sed Associates	1542
Sed Nunc Ltd	1541
Viverra Donec Foundation	1537
Ac Libero Inc.	1513
Amet Faucibus Ut Foundation	1494
Non Magna LLC	1489

The bottom panel shows the 'Output' tab with a log of SQL actions and their execution times.

Conto el total de transaccions agrupats per empresa i les ordeno de major a menor. Canvio el nom del títol de les columnes perquè sigui més entenedor.

- Mostra el llistat aplicant solament subconsultes.

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 SELECT
2   c.company_name AS Empresa,
3   (
4     SELECT COUNT(*)
5     FROM transaction t
6     WHERE t.company_id = c.id
7   ) AS total_transactions
8 FROM company c
9 WHERE c.country = (
10  SELECT country
11  FROM company
12  WHERE company_name = 'Non Institute'
13 )
14 ORDER BY total_transactions desc
```

The Result Grid shows the following data:

Empresa	total_transactions
Empresa	1567
Oro Adipiscing Limited	1550
Erin Conditum Ltd	1543
Non Institute	1542
Interdum Feugiat Sed Associates	1541
Sed Nunc Ltd	1537
Viverra Donec Foundation	1513
Ac Libero Inc.	1494
Amet Faucibus Ut Foundation	1494

The Output tab shows the execution log with the following details:

#	Time	Action	Message	Duration / Fetch
52	11:36:57	SELECT	c.company_name AS Empresa, ( SELECT COUNT(*) FROM tr...	0.015 sec / 0.000 sec
53	11:37:46	SELECT	c.company_name AS Empresa, ( SELECT COUNT(*) FROM tr...	0.016 sec / 0.000 sec
54	11:37:47	SELECT	c.company_name AS Empresa, ( SELECT COUNT(*) FROM tr...	0.016 sec / 0.000 sec
55	11:37:54	SELECT	c.company_name AS Empresa, ( SELECT COUNT(*) FROM tr...	0.016 sec / 0.000 sec
56	11:37:54	SELECT	c.company_name AS Empresa, ( SELECT COUNT(*) FROM tr...	0.016 sec / 0.000 sec
57	11:38:13	SELECT	c.company_name AS Empresa, ( SELECT COUNT(*) FROM tr...	0.015 sec / 0.000 sec

En aquesta consulta aplico 2 subconsultes, la primera és la subconsulta del WHERE que filtra només les empreses del mateix país que 'Non Institute' i en la segona subconsulta, que està dins del SELECT principal, compta les transaccions per cada empresa que compleix el criteri.

## 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.

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 SELECT company_name as EMPRESA, phone as TELEFON, country as PAIS, DATE_FORMAT(timestamp, '%d/%m/%Y') as DATA, amount as TRANSACCIO
2 FROM transactions.transaction t
3 INNER JOIN company c ON c.id = t.company_id
4 WHERE (amount BETWEEN 350 AND 400) AND date(timestamp) IN (
5     '2015-04-29',
6     '2018-07-20',
7     '2024-03-13')
8 ORDER BY TRANSACCIO desc
```

The Results Grid shows the following data:

EMPRESA	TELEFON	PAIS	DATA	TRANSACCIO
Alquam PC	01 45 73 52 16	Germany	13/03/2024	399.84
Auctor Mauris Vel LLP	08 09 28 74 14	United States	20/07/2018	399.51
At Pede Corp.	06 14 48 33 15	Italy	29/04/2015	399.69
Alquam PC	01 45 73 52 16	Germany	13/03/2024	388.29
Oro Adipiscing Limited	03 18 00 77 81	United Kingdom	20/07/2018	373.71
Fringilla LLC	08 29 15 93 57	New Zealand	29/04/2015	367.62
Pede Cum Ltd	07 62 26 48 38	Norway	20/07/2018	356.87
Auctor Mauris Vel LLP	08 09 28 74 14	United States	13/03/2024	353.75

The Output tab shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
68	12:12:34	SELECT company_name as EMPRESA, phone as TELEFON, country as PAIS, timesta...	8838 row(s) returned	0.141 sec / 0.047 sec
69	12:13:59	SELECT company_name as EMPRESA, phone as TELEFON, country as PAIS, timesta...	0 row(s) returned	0.063 sec / 0.000 sec
70	12:17:18	SELECT company_name as EMPRESA, phone as TELEFON, country as PAIS, timesta...	14 row(s) returned	0.094 sec / 0.000 sec
71	12:17:43	SELECT company_name as EMPRESA, phone as TELEFON, country as PAIS, timesta...	8 row(s) returned	0.094 sec / 0.000 sec
72	12:20:24	SELECT company_name as EMPRESA, phone as TELEFON, country as PAIS, DATE...	8 row(s) returned	0.078 sec / 0.000 sec
73	12:28:54	SELECT company_name as EMPRESA, phone as TELEFON, country as PAIS, DATE...	8 row(s) returned	0.078 sec / 0.000 sec

En aquesta consulta utilitzo una DATA\_FORMAT per donar format a la data de sortida i la funció DATE() per compara només el que és la data *dia/mes/any* sense les *hores:minuts:segons* perquè no els tinguin en compte i retorni valors.

**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 especifiquis si tenen més de 400 transaccions o menys.

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 SELECT
2   c.company_name as EMPRESA,
3   CASE
4     WHEN (
5       SELECT COUNT(*)
6       FROM transactions.transaction t
7       WHERE t.company_id = c.id
8     ) > 400
9     THEN 'SI'
10    ELSE 'NO'
11  END AS mes_de_400
12 FROM company c
13 order by mes_de_400
```

The Results window displays the following data:

EMPRESA	mes_de_400
Nec Luctus LLC	NO
Fringilla LLC	NO
Lorem Eu Incorporated	NO
Alc Fermentum Incorporated	SI
Magna A Neque Industries	SI
Fusce Corp.	SI
Convallis In Incorporated	SI
Ante Taculis Nec Foundation	SI

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

#	Time	Action	Message	Duration / Fetch
91	12:57:39	SELECT company_name as EMPRESA, count(amount) as TRANSACTIONS FROM tra...	Error Code: 1064. You have an error in your SQL syntax; check the manual that corresp...	0.000 sec
92	12:57:49	SELECT company_name as EMPRESA, count(amount) as TRANSACTIONS FROM tra...	100 row(s) returned	0.687 sec / 0.000 sec
93	12:58:47	SELECT c.company_name, CASE WHEN ( SELECT COUNT(*) FR...	100 row(s) returned	0.079 sec / 0.000 sec
94	12:59:17	SELECT c.company_name, CASE WHEN ( SELECT COUNT(*) FR...	100 row(s) returned	0.047 sec / 0.000 sec
95	13:00:17	SELECT c.company_name, CASE WHEN ( SELECT COUNT(*) FR...	100 row(s) returned	0.062 sec / 0.000 sec
96	13:00:50	SELECT c.company_name as EMPRESA, CASE WHEN ( SELECT COU...	100 row(s) returned	0.110 sec / 0.000 sec

En aquesta consulta he usat la expressió *CASE* per avaluar en aquest cas dues condicions i retornar un valor dependent de cada una. Valor **SI** si hi ha més de 400 transaccions a la empresa i valor **NO** si hi ha menys de 400 transaccions a la empresa.