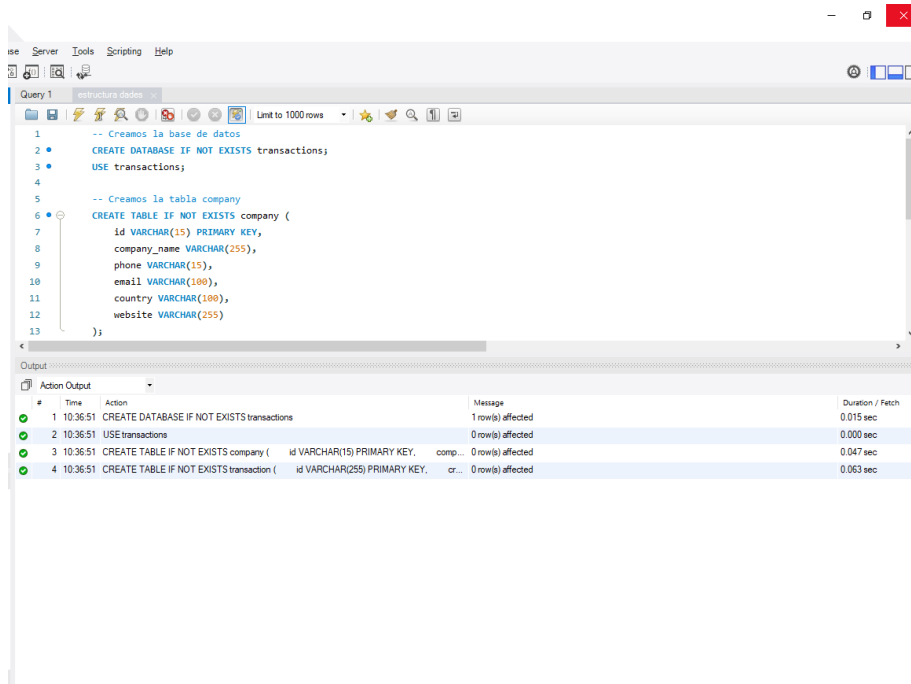


NIVELL 1: Exercici 1

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

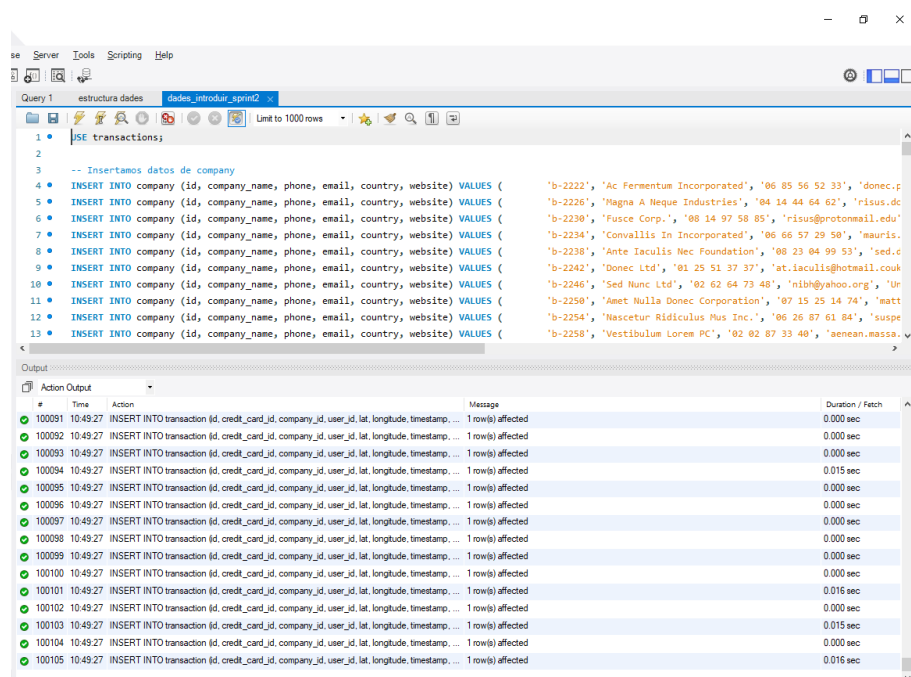


The screenshot shows a SQL IDE window with a script named 'estructura dades'. The script contains SQL commands to create a database and a table. The output window shows the execution results for four actions.

```
1 -- Creamos la base de datos
2 CREATE DATABASE IF NOT EXISTS transactions;
3 USE transactions;
4
5 -- Creamos la tabla company
6 CREATE TABLE IF NOT EXISTS company (
7     id VARCHAR(15) PRIMARY KEY,
8     company_name VARCHAR(255),
9     phone VARCHAR(15),
10    email VARCHAR(100),
11    country VARCHAR(100),
12    website VARCHAR(255)
13 );
```

#	Time	Action	Message	Duration / Fetch
1	10:36:51	CREATE DATABASE IF NOT EXISTS transactions	1 row(s) affected	0.015 sec
2	10:36:51	USE transactions	0 row(s) affected	0.000 sec
3	10:36:51	CREATE TABLE IF NOT EXISTS company (id VARCHAR(15) PRIMARY KEY, comp...	0 row(s) affected	0.047 sec
4	10:36:51	CREATE TABLE IF NOT EXISTS transaction (id VARCHAR(255) PRIMARY KEY, cr...	0 row(s) affected	0.063 sec

Creació dels registres a partir del fitxer *dades_introduir_sprint2.sql*:



The screenshot shows a SQL IDE window with a script named 'dades_introduir_sprint2'. The script contains SQL commands to insert data into a table. The output window shows the execution results for 15 actions.

```
1 USE transactions;
2
3 -- Insertamos datos de company
4 INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
5 INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
6 INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
7 INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
8 INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
9 INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
10 INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
11 INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
12 INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
13 INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
14 INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
15 INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
```

#	Time	Action	Message	Duration / Fetch
100091	10:49:27	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, ...	1 row(s) affected	0.000 sec
100092	10:49:27	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, ...	1 row(s) affected	0.000 sec
100093	10:49:27	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, ...	1 row(s) affected	0.000 sec
100094	10:49:27	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, ...	1 row(s) affected	0.015 sec
100095	10:49:27	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, ...	1 row(s) affected	0.000 sec
100096	10:49:27	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, ...	1 row(s) affected	0.000 sec
100097	10:49:27	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, ...	1 row(s) affected	0.000 sec
100098	10:49:27	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, ...	1 row(s) affected	0.000 sec
100099	10:49:27	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, ...	1 row(s) affected	0.000 sec
100100	10:49:27	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, ...	1 row(s) affected	0.000 sec
100101	10:49:27	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, ...	1 row(s) affected	0.016 sec
100102	10:49:27	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, ...	1 row(s) affected	0.000 sec
100103	10:49:27	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, ...	1 row(s) affected	0.015 sec
100104	10:49:27	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, ...	1 row(s) affected	0.000 sec
100105	10:49:27	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, ...	1 row(s) affected	0.016 sec

Explicació estructura de dades:

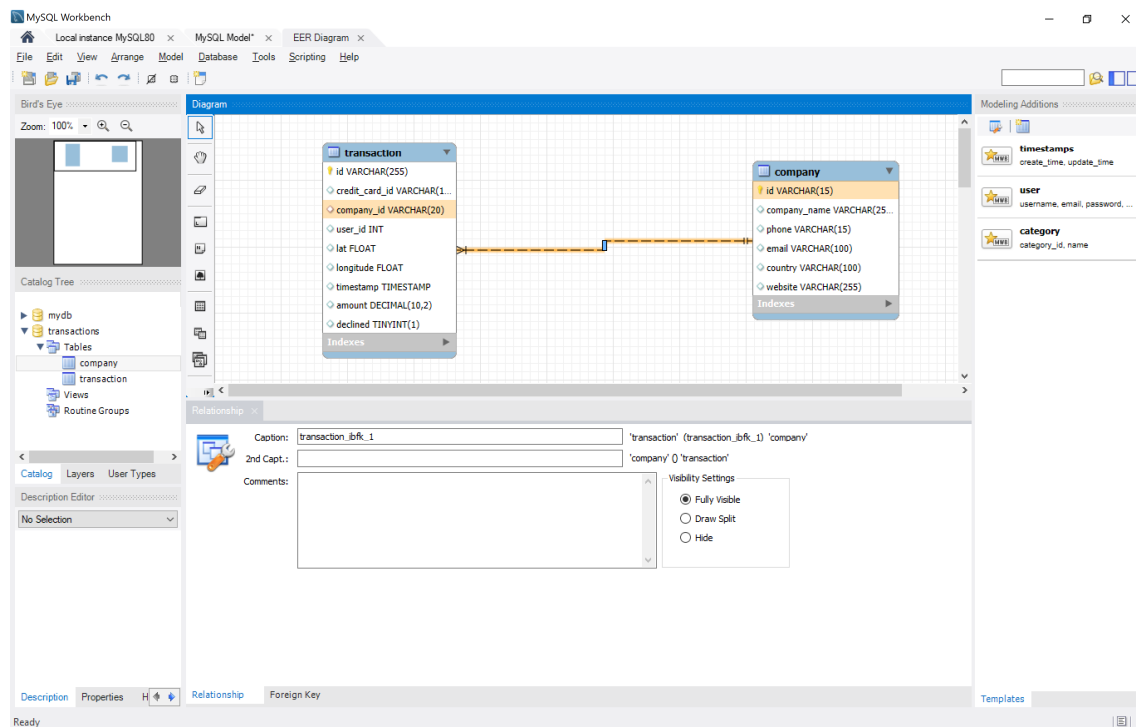
-- Taula **company**

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**.

NIVELL 1: Exercici 2

Utilitzant JOIN realitzaràs les següents consultes:

- Llistat dels països que estan generant vendes.

The screenshot shows a database IDE window titled "consulta_nivell1_exercici2". The SQL query is as follows:

```
1 /* Sprint 2: consulta nivell 1 - exercici 2*/
2 /* Llistat dels països que estan generant vendes.*/
3
4 • SELECT distinct(country) as païses
5 FROM transaction t
6 INNER JOIN company c ON c.id = t.company_id;
```

The results are displayed in a "Result Grid" showing a list of countries:

païses
Germany
Australia
United States
New Zealand
Norway
United Kingdom
Italy
Belgium
Sweden
Ireland

Below the result grid, the "Action Output" pane shows a log of database actions and their durations:

#	Time	Action	Message	Duration / Fetch
35	12:18:05	SELECT count(distinct(country)) as païses FROM transaction INNER JOIN company ON...	1 row(s) returned	0.125 sec / 0.000 sec
36	12:18:42	SELECT distinct(country) as païses FROM transaction INNER JOIN company ON comp...	15 row(s) returned	0.016 sec / 0.000 sec
37	12:18:57	SELECT distinct(country) as païses FROM transaction LEFT JOIN company ON compan...	15 row(s) returned	0.125 sec / 0.000 sec
38	12:19:54	SELECT distinct(country) as païses FROM transaction t INNER JOIN company c ON c.i...	15 row(s) returned	0.000 sec / 0.000 sec
39	12:21:57	SELECT distinct(country) as païses FROM transaction t INNER JOIN company c ON c.i...	15 row(s) returned	0.015 sec / 0.000 sec
40	12:22:24	SELECT distinct(country) as païses FROM transaction t INNER JOIN company c ON c.i...	15 row(s) returned	0.000 sec / 0.000 sec

- Des de quants països es generen les vendes.

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

```
1 /* Sprint 2: consulta nivell 1 - exercici 2 - punt 2*/
2 /* Des de quants països es generen les vendes. */
3
4 • SELECT count(distinct(country)) as païses
5 FROM transaction t
6 INNER JOIN company c ON c.id = t.company_id;
```

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

#	Time	Action	Message	Duration / Fetch
42	12:27:28	SELECT distinct(country) FROM transaction INNER JOIN company ON company.id = tra...	15 row(s) returned	0.000 sec / 0.000 sec
43	12:27:38	SELECT count(country) FROM transaction INNER JOIN company ON company.id = tran...	1 row(s) returned	0.094 sec / 0.000 sec
44	12:27:47	SELECT distinct(country) as païses FROM transaction t INNER JOIN company c ON c.i...	15 row(s) returned	0.016 sec / 0.000 sec
45	12:28:11	SELECT count(country) as païses FROM transaction t INNER JOIN company c ON c.id ...	1 row(s) returned	0.078 sec / 0.000 sec
46	12:28:58	SELECT count(distinct(country)) as païses FROM transaction t INNER JOIN company c ...	1 row(s) returned	0.109 sec / 0.000 sec
47	12:29:51	SELECT count(distinct(country)) as païses FROM transaction t INNER JOIN company c ...	1 row(s) returned	0.188 sec / 0.000 sec

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

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

```
1 /* Sprint 2: consulta nivell 1 - exercici 2 - punt 3*/
2 /* Identifica la companyia amb la mitjana més gran de vendes.*/
3
4 • SELECT round(avg(amount),2) as Mitjana, c.company_name
5 FROM transaction t
6 INNER JOIN company c ON c.id = t.company_id
7 GROUP by company_name
8 order by Mitjana desc
9 LIMIT 1
```

The results pane shows a single row with the value 284.87 for the column 'Mitjana' and 'Ac Fermentum Incorporated' for the column 'company_name'.

#	Time	Action	Message	Duration / Fetch
52	12:40:01	SELECT avg(amount) as Mitjana FROM company c INNER JOIN transaction t ON c.id ...	1 row(s) returned	0.406 sec / 0.000 sec
53	12:40:33	SELECT avg(amount) as Mitjana FROM transaction t INNER JOIN company c ON c.id ...	1 row(s) returned	0.281 sec / 0.000 sec
54	12:40:49	SELECT avg(amount) as Mitjana, id FROM transaction t INNER JOIN company c ON c.i...	Error Code: 1052, Column 'id' in field list is ambiguous	0.000 sec
55	12:40:59	SELECT avg(amount) as Mitjana, c.id FROM transaction t INNER JOIN company c ON ...	Error Code: 1140, In aggregated query without GROUP BY, expression #2 of SELECT lis...	0.000 sec
56	12:44:07	SELECT avg(amount) as Mitjana, c.company_name FROM transaction t INNER JOIN c...	1 row(s) returned	1.078 sec / 0.016 sec
57	12:45:15	SELECT round(avg(amount),2) as Mitjana, c.company_name FROM transaction t INNE...	1 row(s) returned	0.469 sec / 0.000 sec

NIVELL 1: Exercici 3

Utilitzant només subconsultes (sense utilitzar JOIN):

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

The screenshot shows a SQL IDE interface with a query editor and a results pane. The query editor contains the following SQL code:

```
1 /* Sprint 2: consulta nivell 1 - exercici 3 - punt 1*/
2 /* Mostra totes les transaccions realitzades per empreses d'Alemanya*/
3
4 • SELECT *
5 FROM transaction
6 where transaction.company_id IN (Select id from company where country = 'Germany')
```

The results pane displays a table with the following columns: id, credit_card_id, company_id, user_id, lat, longitude, timestamp, amount, and declined. The table contains 13 rows of data, all of which are transactions from German companies (company_id values: b-2222, b-2222, b-2222, b-2222, b-2222, b-2222, b-2222, b-2222, b-2222, b-2222, b-2222, b-2222, b-2222).

The output pane shows the execution log with the following messages:

```
59 12:50:23 SELECT t.* as 'Transaccions empreses alemanyes' FROM transaction t where transac... Error Code: 1064. You have an error in your SQL syntax; check the manual that correspo... 0.000 sec
60 12:50:40 SELECT t.* FROM transaction t where transaction.company_id IN (Select id from comp... Error Code: 1054. Unknown column 'transaction.company_id' in 'IN/ALL/ANY subquery' 0.000 sec
61 12:50:42 SELECT t.* FROM transaction t where transaction.company_id IN (Select id from comp... Error Code: 1054. Unknown column 'transaction.company_id' in 'IN/ALL/ANY subquery' 0.000 sec
62 12:50:51 SELECT * FROM transaction t where transaction.company_id IN (Select id from compan... Error Code: 1054. Unknown column 'transaction.company_id' in 'IN/ALL/ANY subquery' 0.000 sec
63 12:51:05 SELECT count(*) as 'Transaccions empreses alemanyes' FROM transaction where trans... 1 row(s) returned 0.000 sec / 0.000 sec
64 12:51:54 SELECT * FROM transaction where transaction.company_id IN (Select id from compan... 13291 row(s) returned 0.000 sec / 0.125 sec
```

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

```

1  /* Sprint 2: consulta nivell 1 - exercici 3 - punt 2 */
2  /* Llista les empreses que han realitzat transaccions per un amount superior a la mitjana de totes les transaccions. */
3
4  • SELECT company_name
5    FROM company
6   WHERE id IN (
7     SELECT DISTINCT company_id FROM transaction WHERE amount > (
8       SELECT AVG(amount) AS ventes FROM transaction
9     )
10  )

```

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

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

```

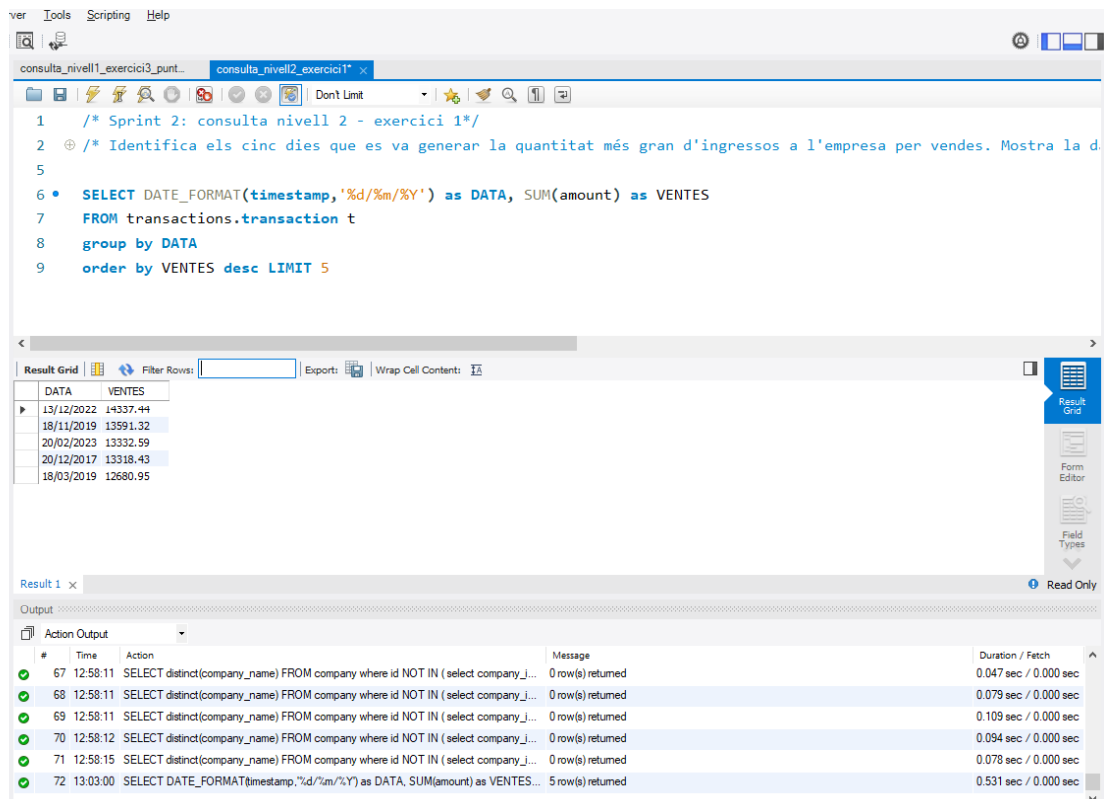
1  /* Sprint 2: consulta nivell 1 - exercici 3 - punt 3 */
2  /* Llista de les empreses empreses que no tenen transaccions registrades. */
3
4  • SELECT DISTINCT(company_name)
5    FROM company
6   WHERE id NOT IN (
7     SELECT company_id FROM transaction
8   )
9

```

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

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 a SQL IDE interface with a query editor and a results pane. The query is as follows:

```
1  /* Sprint 2: consulta nivell 2 - exercici 1 */
2  /* Identifica els cinc dies que es va generar la quantitat més gran d'ingressos a l'empresa per vendes. Mostra la d
5
6  • SELECT DATE_FORMAT(timestamp,'%d/%m/%Y') as DATA, SUM(amount) as VENTES
7  FROM transactions.transaction t
8  group by DATA
9  order by VENTES desc LIMIT 5
```

The results pane displays a table with two columns: DATA and VENTES. The data is as follows:

DATA	VENTES
13/12/2022	14337.44
18/11/2019	13991.32
20/02/2023	13332.59
20/12/2017	13318.43
18/03/2019	12680.95

The bottom pane shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
67	12:58:11	SELECT distinct(company_name) FROM company where id NOT IN (select company_j...	0 row(s) returned	0.047 sec / 0.000 sec
68	12:58:11	SELECT distinct(company_name) FROM company where id NOT IN (select company_j...	0 row(s) returned	0.079 sec / 0.000 sec
69	12:58:11	SELECT distinct(company_name) FROM company where id NOT IN (select company_j...	0 row(s) returned	0.109 sec / 0.000 sec
70	12:58:12	SELECT distinct(company_name) FROM company where id NOT IN (select company_j...	0 row(s) returned	0.094 sec / 0.000 sec
71	12:58:15	SELECT distinct(company_name) FROM company where id NOT IN (select company_j...	0 row(s) returned	0.078 sec / 0.000 sec
72	13:03:00	SELECT DATE_FORMAT(timestamp,'%d/%m/%Y') as DATA, SUM(amount) as VENTES...	5 row(s) returned	0.531 sec / 0.000 sec

Sumo totes les vendes de l'empresa, ordenades de major a menor i retornant amb la seva data corresponent. Retorno 5 registres. He donat format a la data perquè només mostri el dia, el mes i l'any.

NIVELL 02: Exercici 2

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

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

```
1 /* Sprint 2: consulta nivell 2 - exercici 2*/  
2 /* Llistat de la mitjana de vendes per país ordenats de major a menor mitjana. */  
3  
4 • SELECT FORMAT(avg(amount),2) as MITJANA, country as PAIS  
5 FROM transactions.transaction t  
6 INNER JOIN company c ON c.id = t.company_id  
7 group by country  
8 order by MITJANA desc
```

The results are displayed in a table with two columns: MITJANA and PAIS. The data is as follows:

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

The bottom of the screenshot shows the 'Action Output' section with a table of query execution details:

#	Time	Action	Message	Duration / Fetch
68	12:58:11	SELECT distinct(company_name) FROM company where id NOT IN (select company_...	0 row(s) returned	0.079 sec / 0.000 sec
69	12:58:11	SELECT distinct(company_name) FROM company where id NOT IN (select company_...	0 row(s) returned	0.109 sec / 0.000 sec
70	12:58:12	SELECT distinct(company_name) FROM company where id NOT IN (select company_...	0 row(s) returned	0.094 sec / 0.000 sec
71	12:58:15	SELECT distinct(company_name) FROM company where id NOT IN (select company_...	0 row(s) returned	0.078 sec / 0.000 sec
72	13:03:00	SELECT DATE_FORMAT(timestamp,'%d/%m/%Y') as DATA, SUM(amount) as VENTES...	5 row(s) returned	0.531 sec / 0.000 sec
73	13:09:31	SELECT FORMAT(avg(amount),2) as MITJANA, country as PAIS FROM transactions.tr...	15 row(s) returned	0.360 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.

NIVELL 02: 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 a SQL IDE interface with a query editor and a results grid. The query is as follows:

```
1
2  /* Sprint 2: consulta nivell 2 - exercici 3 - punt 1*/
3  /* Mostra totes la llista de totes les transaccions realitzades per empreses que estan situades en el mateix país que aquesta companyia.*/
4
5  SELECT t.*
6  FROM transactions.transaction t
7  INNER JOIN company c ON c.id = t.company_id
8  where country = (
9    SELECT country FROM transactions.company where company_name='Non Institute'
10 )
11
```

The results grid displays a table with the following columns: id, credit_card_id, company_id, user_id, lat, longitude, timestamp, amount, and declined. The data is sorted by company_id and then by id.

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
008629B4-C9A9-406C-A3D2-71FDA47BC546	C/S-7063	u-2246	2482	45.7666	4.83048	2015-07-30 12:12:42	486.44	0
00872BA4-54A3-4B8E-813F-2D57535AA17A	C/S-8475	b-2246	3894	55.6212	-3.7546	2017-10-26 22:08:26	414.06	0
01F075B1-07AE-4D02-AAD9-9FDD72A43F3C	C/S-8700	b-2246	4119	55.856	-3.15783	2018-01-27 13:44:36	103.73	0
023FFCE8-E618-4938-8F56-C8DF80540ADD	C/S-7816	b-2246	3235	46.3568	1.82755	2016-12-19 11:53:45	219.28	0
02683EBE-EF91-4564-957B-D6F1662AB7C5	C/S-9471	b-2246	4890	42.1332	12.396	2017-01-10 21:09:29	326.87	0
02C2F29E-CEFD-4C1E-A594-F476E8F279C0	C/S-9082	b-2246	4501	39.4662	-0.373246	2020-05-24 01:17:29	155.72	0
02F468DC-426C-47C2-880A-D8B25B7A81AF	C/S-6913	b-2246	2332	52.175	19.3508	2023-03-17 16:36:27	305.35	0
03068E3B-817B-4A49-934E-0E439291A104	C/S-5302	b-2246	721	51.9233	18.926	2021-12-02 23:06:02	339.58	0
03478FE6-8EB5-4387-8187-0E78E8F289FB	C/S-7674	b-2246	3093	45.768	4.84271	2021-12-30 08:40:24	172.93	0
03AEBD0E-DC97-4BD3-9C57-6A6D878026FD	C/S-6121	b-2246	1540	50.8113	10.3145	2018-11-11 11:28:49	114.77	0

The output section shows the execution of the query, with the following messages:

- 69 12:58:11 SELECT distinct(company_name) FROM company where id NOT IN (select company_j... 0 row(s) returned 0.109 sec / 0.000 sec
- 70 12:58:12 SELECT distinct(company_name) FROM company where id NOT IN (select company_j... 0 row(s) returned 0.094 sec / 0.000 sec
- 71 12:58:15 SELECT distinct(company_name) FROM company where id NOT IN (select company_j... 0 row(s) returned 0.078 sec / 0.000 sec
- 72 13:03:00 SELECT DATE_FORMAT(timestamp,'%d/%m/%Y') as DATA, SUM(amount) as VENTES... 5 row(s) returned 0.531 sec / 0.000 sec
- 73 13:09:31 SELECT FORMAT(avg(amount),2) as MITJANA, country as PAIS FROM transactions tr... 15 row(s) returned 0.360 sec / 0.000 sec
- 74 13:13:18 SELECT t.* FROM transactions.transaction t INNER JOIN company c ON c.id = t.compa... 13776 row(s) returned 0.000 sec / 0.062 sec

Llisto les transaccions agrupats per empresa i les ordeno de major a menor.

- Mostra el llistat aplicant solament subconsultes.

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

```

1  /* Sprint 2: consulta nivell 2 - exercici 3 - punt 2*/
2  /* Mostra totes la llista de totes les transaccions realitzades per empreses que estan situades en el mateix país
3  • SELECT t.*
4  FROM transaction t
5  where company_id IN
6  (select id from company c where country =
7  (select country from company where company_name='Non Institute'))
8
9

```

The results grid displays a table with the following columns: id, credit_card_id, company_id, user_id, lat, longitude, timestamp, amount, and declined. The table contains 13 rows of data.

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
00862984-C9A9-406C-A3D2-71FDA47BC546	CcS-7063	b-2246	2482	45.7666	4.83048	2015-07-30 12:12:42	486.44	0
00872BA4-54A3-4B8E-B13F-2D57535AA17A	CcS-847	b-2246	3894	55.6212	-3.7546	2017-10-26 22:08:26	414.06	0
01F075B1-D7AE-4D02-AAD9-5FFD72A43F3C	CcS-8700	b-2246	4119	55.856	-3.15783	2018-01-27 13:44:36	103.73	0
023FFCE8-E618-4938-BF56-C8FD80540ADD	CcS-7816	b-2246	3235	46.3568	1.82755	2016-12-19 11:53:45	219.28	0
026838EB-EF91-4564-957B-D6F1662AB7C5	CcS-9471	b-2246	4890	42.1332	12.396	2017-01-10 21:09:29	326.87	0
02C2F29E-CE2-4C1E-A594-F476E8F279C0	CcS-9082	b-2246	4501	39.4662	-0.373246	2020-05-24 01:17:29	155.72	0
02F468DC-426C-47C2-8B0A-D8B25B7A81AF	CcS-6913	b-2246	2332	52.175	19.3508	2023-03-17 16:36:27	305.35	0
03068E3B-817B-4A49-934E-0E439291A104	CcS-5302	b-2246	721	51.9233	18.926	2021-12-02 23:06:02	339.58	0
0347BFE6-8EB5-4387-B187-0E78E8F2B8FB	CcS-7674	b-2246	3093	45.768	4.84271	2021-12-30 08:40:24	172.93	0
03AEBD0E-DC97-4BD3-9C57-6A6D878026FD	CcS-6121	b-2246	1540	50.8113	10.3145	2018-11-11 11:28:49	114.77	0
03CA36D3-88FF-4DBF-8FD4-4CC7DA4EED2B	CcS-8036	b-2246	3455	52.5178	13.4131	2017-02-25 15:38:21	440.27	0

The output section shows the execution of the query, with the following messages:

```

70 12:58:12 SELECT distinct(company_name) FROM company where id NOT IN ( select company_j... 0 row(s) returned
71 12:58:15 SELECT distinct(company_name) FROM company where id NOT IN ( select company_j... 0 row(s) returned
72 13:03:00 SELECT DATE_FORMAT(timestamp,"id/%m/%Y") as DATA, SUM(amount) as VENTES... 5 row(s) returned
73 13:09:31 SELECT FORMAT(avg(amount),2) as MITJANA, country as PAIS FROM transactions.tr... 15 row(s) returned
74 13:13:18 SELECT t.* FROM transactions.transaction t INNER JOIN company c ON c.id = t.compa... 13776 row(s) returned
75 13:15:12 SELECT t.* FROM transaction t where company_id IN (select id from company c where ... 13776 row(s) returned

```

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 esta dins del SELECT principal i llista 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 a database query editor with a SQL query and its results. The query is as follows:

```

1  /* Sprint 2: consulta nivell 3 - exercici 1 */
2  /*Presenta el nom, telèfon, país, data i amount, d'aquelles empreses que van realitzar transaccions amb un valor co
3
4  * SELECT company_name as EMPRESA, phone as TELEFON, country as PAIS,
5     DATE_FORMAT(timestamp, '%d/%m/%Y') as DATA, amount as TRANSACCIO
6  FROM transactions.transaction t
7  INNER JOIN company c ON c.id = t.company_id
8  where (amount between 350 and 400) and date(timestamp) in (
9     '2015-04-29',
10    '2018-07-20',
11    '2024-03-13')
12  order by TRANSACCIO desc
  
```

The results are displayed in a table with the following columns: EMPRESA, TELEFON, PAIS, DATA, and TRANSACCIO. The results are ordered by TRANSACCIO in descending order.

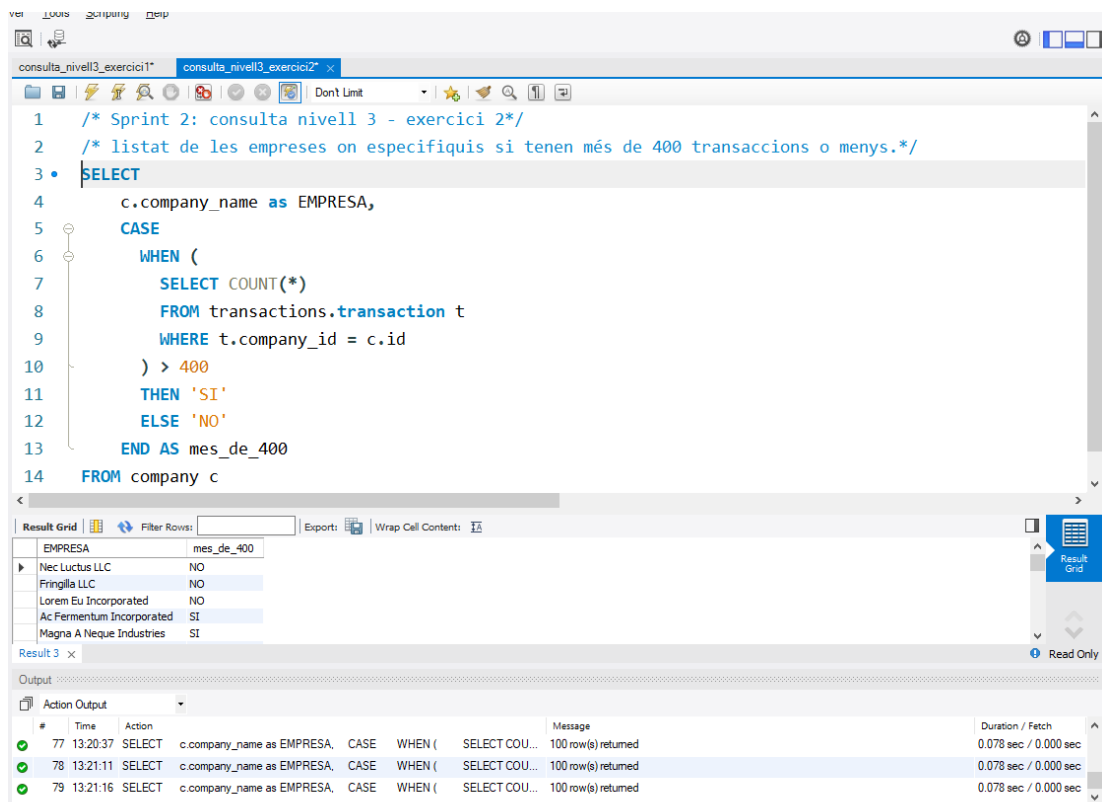
EMPRESA	TELEFON	PAIS	DATA	TRANSACCIO
Aliquam PC	01 45 73 52 16	Germany	13/03/2024	399.84
Auctor Mauris Vel LLP	08 09 28 74 14	United States	20/07/2018	399.51
At Pede Corp.	06 14 48 33 15	Italy	29/04/2015	390.69
Aliquam PC	01 45 73 52 16	Germany	13/03/2024	388.29
Orci Adipiscing Limited	03 18 00 77 81	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 bottom part of the screenshot shows the 'Action Output' section, which displays the execution details of the query, including the time taken and the number of rows returned.

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.

Nivell 3: 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 especifiqueu si tenen més de 400 transaccions o menys.



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

```
1  /* Sprint 2: consulta nivell 3 - exercici 2*/
2  /* listat de les empreses on especifiqueu si tenen més de 400 transaccions o menys.*/
3  • SELECT
4    c.company_name as EMPRESA,
5    CASE
6      WHEN (
7        SELECT COUNT(*)
8        FROM transactions.transaction t
9        WHERE t.company_id = c.id
10       ) > 400
11      THEN 'SI'
12      ELSE 'NO'
13    END AS mes_de_400
14  FROM company c
```

The results grid shows the following data:

EMPRESA	mes_de_400
Nec Luctus LLC	NO
Fringilla LLC	NO
Lorem Eu Incorporated	NO
Ac Fermentum Incorporated	SI
Magna A Neque Industries	SI

The output pane shows the execution of the query, with the following message:

```
77 13:20:37 SELECT c.company_name as EMPRESA, CASE WHEN ( SELECT COU... 100 row(s) returned 0.078 sec / 0.000 sec
78 13:21:11 SELECT c.company_name as EMPRESA, CASE WHEN ( SELECT COU... 100 row(s) returned 0.078 sec / 0.000 sec
79 13:21:16 SELECT c.company_name as EMPRESA, CASE WHEN ( SELECT COU... 100 row(s) returned 0.078 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.