

1	Descripció	2
2	Nivell 1	2
2.1	Exercici 1.....	2
2.1.1	Creació taules amb la consulta estructura_dades.sql	2
2.1.2	Omplir les taules amb la consulta dades_introduir.sql	4
2.1.3	Diagrama que relaciona taules i variables	5
2.2	Exercici 2.....	6
2.2.1	Consulta SQL en Workbench i resultats	6
2.3	Exercici 3.....	7
2.3.1	Consulta SQL en Workbench i resultats	7
2.4	Exercici 4.....	8
2.4.1	Consulta SQL en Workbench i resultats	8
2.5	Exercici 5.....	8
2.5.1	Consulta SQL en Workbench i resultats	9
2.6	Exercici 6.....	9
2.6.1	Consulta SQL en Workbench i resultats	9
3	Nivell 2	10
3.1	Exercici 1.....	10
3.1.1	Consulta SQL en Workbench i resultats	11
3.1.2	Consulta SQL en Workbench i resultats (Resolució 2).....	11
3.2	Exercici 2.....	12
3.2.1	Consulta SQL en Workbench i resultats	12
3.2.2	Detall de la solució del dia de les 5 vendes més costoses.....	13
3.3	Exercici 3.....	14
3.3.1	Consulta SQL en Workbench i resultats	14
3.3.2	Detall de les vendes del dia de les 5 vendes menys costoses.....	15
3.4	Exercici 4.....	16
3.4.1	Consulta SQL en Workbench i resultats	16
4	Nivell 3	17
4.1	Exercici 1.....	17
4.1.1	Consulta SQL en Workbench i resultats	17
4.2	Exercici 2.....	18
4.2.1	Consulta SQL en Workbench i resultats	18

1 Descripció

Repassar les nocions bàsiques per a l'ús de base de dades relacionals. En aquest sprint, iniciaràs la teva experiència pràctica amb una base de dades que conté informació d'una empresa dedicada a la venda de productes en línia. En aquesta activitat, t'enfocaràs en dades relacionades amb les transaccions efectuades i la informació corporativa de les empreses que van participar.

2 Nivell 1

2.1 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.

2.1.1 Creació taules amb la consulta *estructura_dades.sql*

Primer obre la consulta *estructura_dades.sql* amb el programa MySQL Workbench i executo la consulta per crear l'estructura de dades de la taula *company* i la taula *transaction*.

```
-- Creamos la base de datos
CREATE DATABASE IF NOT EXISTS transactions;
USE transactions;
-- Creamos la tabla company
CREATE TABLE IF NOT EXISTS company (
    id VARCHAR(15) PRIMARY KEY,
    company_name VARCHAR(255),
    phone VARCHAR(15),
    email VARCHAR(100),
    country VARCHAR(100),
    website VARCHAR(255)
);
-- Creamos la tabla transaction
CREATE TABLE IF NOT EXISTS transaction (
    id VARCHAR(255) PRIMARY KEY,
    credit_card_id VARCHAR(15) REFERENCES credit_card(id),
    company_id VARCHAR(20),
    user_id INT REFERENCES user(id),
    lat FLOAT,
    longitude FLOAT,
    timestamp TIMESTAMP,
    amount DECIMAL(10, 2),
    declined BOOLEAN,
    FOREIGN KEY (company_id) REFERENCES company(id)
);
```

La primera comanda, és USE transactions, per indicar que a partir d'ara la BBDD actual del programa MySQL Workbench

Aquesta consulta SQL em crea l'estructura d'una taula de 6 columnes si no existeix de nom **company**. La primera columna ID es de tipus caràcter i com a molt pot contenir 15 caràcters. També és una clau principal d'aquesta taula.

Els altres camps són també de tipus VARCHAR i de diferents dimensions.

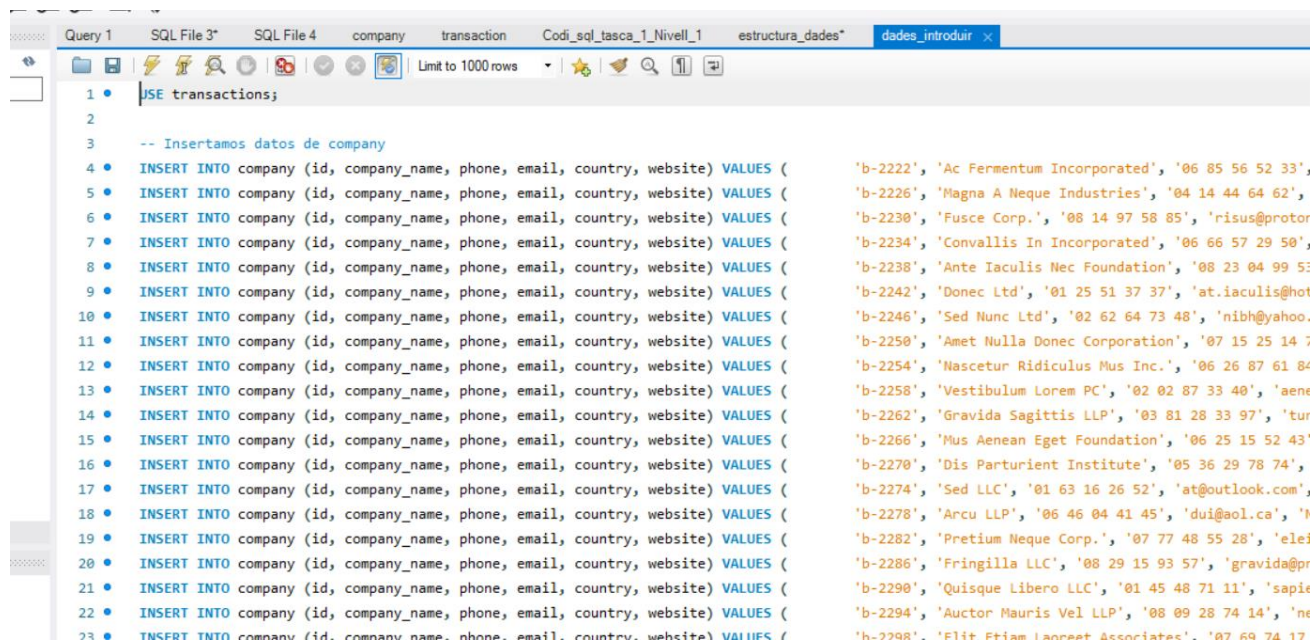
La segona taula em crea una estructura de 10 columnes que s'anomena *transaction*. Bàsicament aquesta taula contindrà totes les transaccions que fan els clients associades a diferents companyies, per això té una clau secundària (*company_id*) que és relaciona amb la taula anterior de company. Es a dir, pot haver moltes transaccions associades a una mateixa companyia, però la informació de cada companyia està només a la taula company.

Tots els camps són del tipus VARCHAR que pot contenir caràcters de longitud variable, on es defineix la longitud per cada declaració de variable.

Hi altres camps com *amount* que són del tipus float, un número real, per contenir la quantitat que un usuari ha gastat a una certa companyia, i del tipus BOOLEAN, per si la operació s'ha acceptat o no.

2.1.2 Omplir les taules amb la consulta dades_introduir.sql

Si executo aquesta consulta, ompló la taula transaccions amb diferents registres.



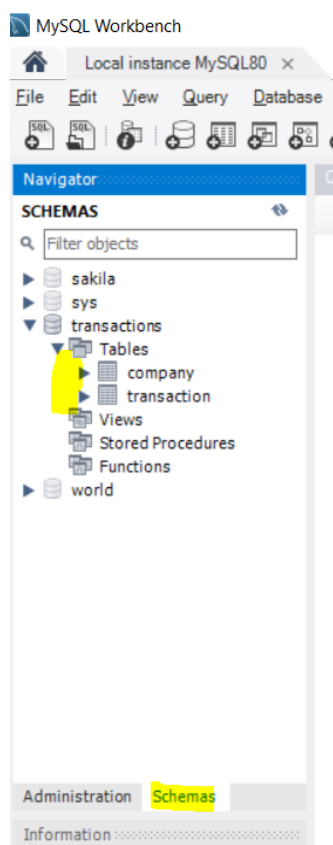
```

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 (
16 • INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
17 • INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
18 • INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
19 • INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
20 • INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
21 • INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
22 • INSERT INTO company (id, company_name, phone, email, country, website) VALUES (
23 • INSERT INTO company (id, company_name, phone, email, country, website) VALUES (

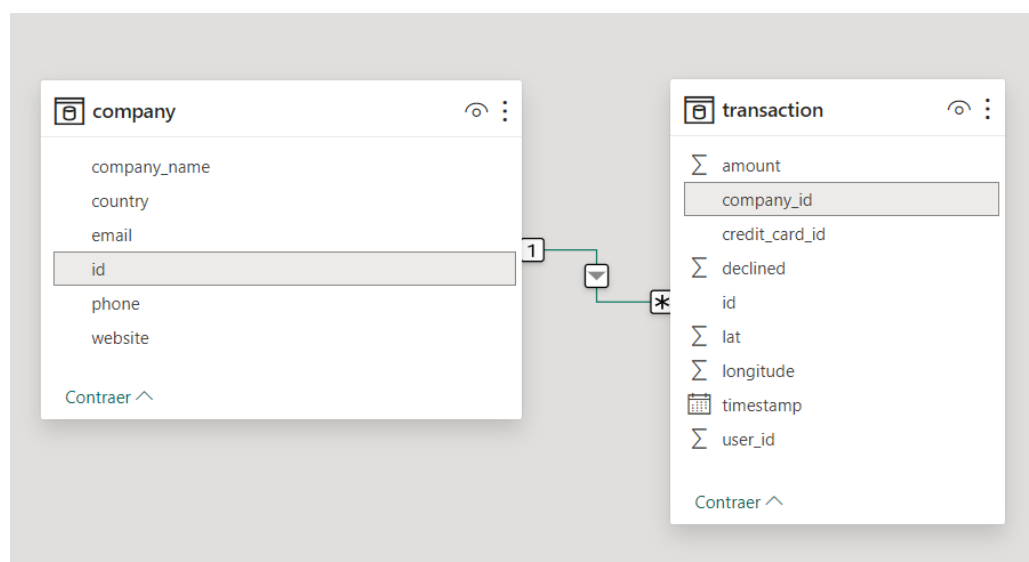
```

'b-2222', 'Ac Fermentum Incorporated', '06 85 56 52 33',
'b-2226', 'Magna A Neque Industries', '04 14 44 64 62',
'b-2230', 'Fusce Corp.', '08 14 97 58 85', 'risus@protor
'b-2234', 'Convallis In Incorporated', '06 66 57 29 50',
'b-2238', 'Ante Iaculis Nec Foundation', '08 23 04 99 5:
'b-2242', 'Donec Ltd', '01 25 51 37 37', 'at.iaculis@hot
'b-2246', 'Sed Nunc Ltd', '02 62 64 73 48', 'nibh@yahoo.
'b-2250', 'Amet Nulla Donec Corporation', '07 15 25 14 ;
'b-2254', 'Nascetur Ridiculus Mus Inc.', '06 26 87 61 84
'b-2258', 'Vestibulum Lorem PC', '02 02 87 33 40', 'aene
'b-2262', 'Gravida Sagittis LLP', '03 81 28 33 97', 'tur
'b-2266', 'Mus Aenean Eget Foundation', '06 25 15 52 43'
'b-2270', 'Dis Parturient Institute', '05 36 29 78 74',
'b-2274', 'Sed LLC', '01 63 16 26 52', 'at@outlook.com',
'b-2278', 'Arcu LLP', '06 46 04 41 45', 'dui@aol.ca', 'l
'b-2282', 'Pretium Neque Corp.', '07 77 48 55 28', 'elei
'b-2286', 'Fringilla LLC', '08 29 15 93 57', 'gravida@pr
'b-2290', 'Quisque Libero LLC', '01 45 48 71 11', 'sapie
'b-2294', 'Auctor Mauris Vel LLP', '08 09 28 74 14', 'ne
'h-2298', 'Flit Etiam Lacus Associates', '07 69 74 17

Si vaig a schema trobo la creació de les dues taules:



2.1.3 Diagrama que relaciona taules i variables



La variable **id** és una clau primària a la taula company amb integritat referencial, per identificar unívocament a cada companyia i a la taula transaccions es relaciona amb **company_id**, i és una clau secundària en aquesta segona taula. Es clar, poden haver moltes transaccions associades a la mateixa companyia.

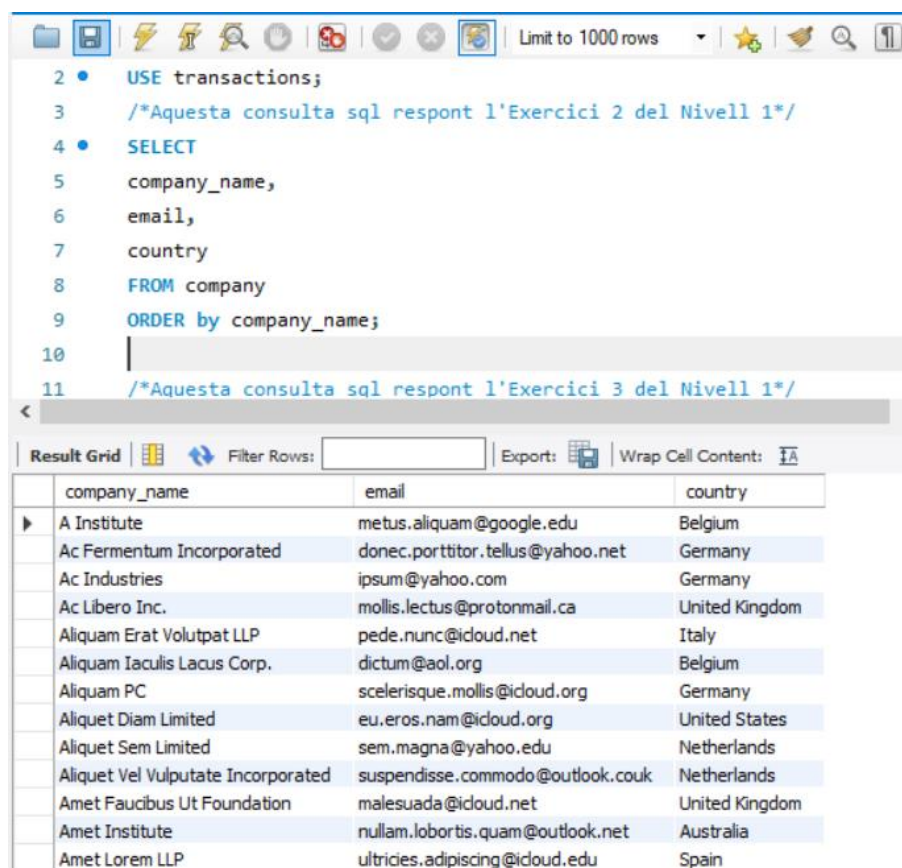
2.2 Exercici 2

Realitza la següent consulta: Has d'obtenir el nom, email i país de cada companyia, ordena les dades en funció del nom de les companyies.

2.2.1 Consulta SQL en Workbench i resultats

Com que farem servir sempre el mateix arxiu *Codi_sql_tasca_1_vf.sql* per tots els exercicis de la tasca 1, farem servir la base activa *transactions* amb `USE transactions;` a dalt de tot d'aquesta query.

Aquesta consulta fa exactament el que ens diu l'enunciat, es treuen de la taula de companyies, el nom de la companyia, el email, i el país, i s'ordena de forma ascendent per l'abecedari el camp *company_name*.



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

```

2 • USE transactions;
3 /*Aquesta consulta sql respon a l'Exercici 2 del Nivell 1*/
4 • SELECT
5   company_name,
6   email,
7   country
8 FROM company
9 ORDER by company_name;
10
11 /*Aquesta consulta sql respon a l'Exercici 3 del Nivell 1*/

```

Below the editor, the 'Result Grid' tab is active, displaying the results of the query in a table with 3 columns: *company_name*, *email*, and *country*. The results are sorted alphabetically by *company_name*.

company_name	email	country
A Institute	metus.aliquam@google.edu	Belgium
Ac Fermentum Incorporated	donec.porttitor.tellus@yahoo.net	Germany
Ac Industries	ipsum@yahoo.com	Germany
Ac Libero Inc.	mollis.lectus@protonmail.ca	United Kingdom
Aliquam Erat Volutpat LLP	pede.nunc@icloud.net	Italy
Aliquam Iaculis Lacus Corp.	dictum@aol.org	Belgium
Aliquam PC	scelerisque.mollis@icloud.org	Germany
Aliquet Diam Limited	eu.eros.nam@icloud.org	United States
Aliquet Sem Limited	sem.magna@yahoo.edu	Netherlands
Aliquet Vel Vulputate Incorporated	suspendisse.commodo@outlook.co.uk	Netherlands
Amet Faucibus Ut Foundation	malesuada@icloud.net	United Kingdom
Amet Institute	nullam.lobortis.quam@outlook.net	Australia
Amet Lorem LLP	ultrices.adipiscing@icloud.edu	Spain

2.3 Exercici 3

Des de la secció de màrqueting et sol·liciten que els passis un llistat dels països que estan fent compres.

2.3.1 Consulta SQL en Workbench i resultats

Recordem que estem utilitzant el mateix arxiu *Codi_sql_tasca_1_vf.sql* i a dalt de tot es troba actiu la instrucció `USE transactions;` per fer activa la BBDD on es troben les taules que volem aplicar la següent query.

En aquesta query volem treure el nom dels països d'on es fan transaccions. El que fem primer és fer un `INNER JOIN` per tal de relacionar la taula *company* amb la taula *transaction* amb la clau primària i secundària ja explicada anteriorment. D'aquesta manera puc veure el país per cada transacció. Com que sortien molts països repetits, és per això que utilitzo la comanda `DISTINCT` davant de `country`.

```
1  /*EXERCICI 3-----
2  •  SELECT
3  DISTINCT country
4  FROM transaction
5  INNER JOIN company ON
6  transactions.company.id=transactions.transaction.company_id;
7
8  /*EXERCICI 4-----
9  ☹ /*select
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
country			
Germany			
Australia			
United States			
New Zealand			
Norway			
United Kingdom			
Italy			
Belgium			
Sweden			
Ireland			
China			
Canada			
France			
Netherlands			
Spain			

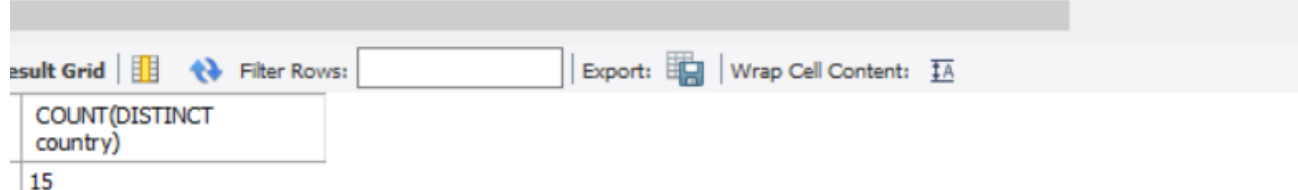
2.4 Exercici 4

Des de màrqueting també volen saber des de quants països es realitzen les compres.

2.4.1 Consulta SQL en Workbench i resultats

Es tracta de la mateixa query anterior però ara l'únic que he de fer és comptar els països i per això poso la funció COUNT davant de DISTINCT country.

```
18      /*EXERCICI 4-----*/
19  •    SELECT
20      COUNT(DISTINCT country)
21      FROM transaction
22      INNER JOIN company ON
23      transactions.company.id=transactions.transaction.company_id;
24
25      /*EXERCICI 5-----*/
```



The screenshot shows the SQL Workbench interface. The top part displays the SQL query for Exercise 4. Below the query, the 'Result Grid' is visible, showing a single row with the value 15, which represents the count of distinct countries. The interface includes a 'Filter Rows' input field, an 'Export' button, and a 'Wrap Cell Content' checkbox.

COUNT(DISTINCT country)
15

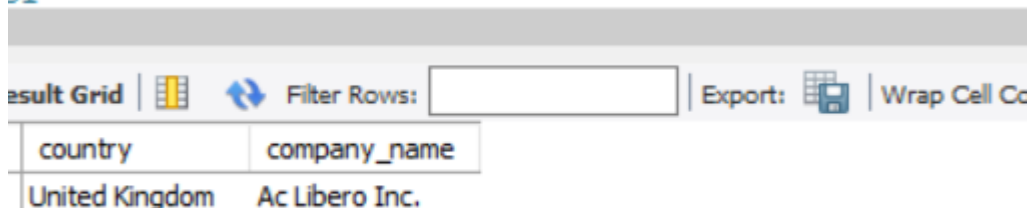
2.5 Exercici 5

El teu cap identifica un error amb la companyia que té aneu 'b-2354'. Per tant, et sol·licita que li indiquis el país i nom de companyia d'aquest aneu

2.5.1 Consulta SQL en Workbench i resultats

Només hem de filtrar de totes les companyies la que tinguin el codi de companyia que ens diu l'enunciat és a dir `id='b-2354'` i després veure els camps `country` i `company_name`

```
25      /*EXERCICI 5-----
26 •    SELECT
27      country,
28      company_name
29      FROM transactions.company
30      WHERE id='b-2354';
31
```



country	company_name
United Kingdom	Ac Libero Inc.

2.6 Exercici 6

A més, el teu cap et sol·licita que indiquis quina és la companyia amb major despesa mitjana?




2.6.1 Consulta SQL en Workbench i resultats

Com sempre faig un INNER JOIN per tal de relacionar les dues amb la clau primària i secundària respectivament. Posteriorment de la taula *transaction* el que fem és agrupar tots els registres per *company_name* i fem la mitjana de les despeses associades per cada companyia. La mitjana es fa amb la funció AVG. Per fer la agrupació segons *company_name*, es fa amb l'operador GROUP BY. Finalment fem una ordenació descendent de la despesa mitjana i així veiem quina és la companyia amb més despesa mitjana que és *Eget Ipsum Ltd*, amb una despesa mitjana de 473,075 unitats.

```

48  /*EXERCICI 6-----
49  •  SELECT
50      company_name,
51      AVG(amount)
52  FROM transaction
53  INNER JOIN company ON
54      transactions.company.id=transactions.transa
55  GROUP BY company_id
56  ORDER BY AVG(amount) DESC;

```

Result Grid   Filter Rows: <input type="text"/> Export: 	
company_name	AVG(amount)
Eget Ipsum Ltd	473.075000
Non Magna LLC	468.345000
Sed Id Limited	461.210000
Justo Eu Arcu Ltd	443.635000
Eget Tincidunt Dui Institute	442.520000
Viverra Donec Foundation	442.280000
Vestibulum Lorem PC	434.060000
Aliquet Diam Limited	425.640000
Maecenas Malesuada Fringilla Inc.	408.620000
Non Ante LLP	407.790000
Egestas Nunc Sed Limited	406.110000
Nunc Sit Incorporated	405.355000
Magna A Neque Industries	396.315000
Amet Luctus Vulputate Foundation	390.325000
Aliquam PC	385.265000

3 Nivell 2

3.1 Exercici 1

El teu cap està redactant un informe de tancament de l'any i et sol·licita que li enviïs informació rellevant per al document. Per a això et sol·licita verificar si en la base de dades existeixen companyies amb identificadors (anew) duplicats.

3.1.1 Consulta SQL en Workbench i resultats

```

56  /*-----SCRIPTS DELS EXERCICIS CORRESPONENT AL NIVELL 2-----*/
57  /*EXERCICI 1-----*/
58  •  SELECT
59      id,
60      count(id) as count_id
61  FROM transactions.company
62  GROUP BY id
63  ORDER BY count id DESC;

```

id	count_id
b-2222	1
b-2226	1
b-2230	1
b-2234	1
b-2238	1
b-2242	1
b-2246	1
b-2250	1
b-2254	1
b-2258	1

Si fem una agrupació del camp id, que és l'identificador de companyia, i fem un comptatge d'aquest identificador, veurem si hi ha repetits o no. Posteriorment ordenem el comptatge de forma descendent.

Es pot verificar que no hi ha cap companyia que tingui el identificador id duplicat, d'altra banda els valors de la variable count_id dels primers registres de solució, serien major que 1.

3.1.2 Consulta SQL en Workbench i resultats (Resolució 2)

Podem fer una variant per tal que directament ens indiqui els repetits. Clarament, no apareix cap registre fent un HAVING de la consulta anterior.

```

55  /*EXERCICI 1 (Solució Variant)-----
56  •  SELECT
57      id,
58      count(id) count_id
59  FROM transactions.company
60  GROUP BY id
61  HAVING COUNT(id)>1;
62

```

id	count_id
----	----------

3.2 Exercici 2

En quin dia es van realitzar les cinc vendes més costoses? Mostra la data de la transacció i la sumatòria de la quantitat de diners

3.2.1 Consulta SQL en Workbench i resultats

Aquest exercici es podia haver interpretat de moltes maneres, però jo ho interpretaré de la següent.

Podria ser que un dia determinat hi haguessin únicament dues transaccions, per tan dues vendes que fossin molt grans. Nosaltres no buscarem aquests dies, ja que l'enunciat busca dies que hi hagin hagut almenys 5 vendes. Per tant de la taula de transaccions buscarem el subconjunt de registres que hi hagi hagut més de 5 vendes.

També es important que el camp *timestamp*, té hora i per tant això no ens anirà bé de cara a utilitzar la funció GROUP BY. Es per això que el camp timestamp el convertirem amb tipus DATE sense hora. Això ho fem amb la funció CAST.

Un cop ja tenim la data sense hora, és a dir el camp *fecha_sin_hora*, ja que podem fer una agrupació d'aquest camp amb GROUP BY i fer un comptatge de quantes transaccions s'han fet en un mateix dia, així com quina quantitat total sumen aquestes transaccions. Finalment d'aquesta taula d'agrupacions, filtrem amb HAVING COUNT per saber quins dies s'han fet més de 5 vendes que serà el nostre subconjunt que estàvem buscant. Finalment si ordenem pel camp *suma_transaccions* de forma descendent ja podem veure quin és el dia amb les 5 vendes més altes.

```

83  /*EXERCICI 2 -----
84  •  SELECT
85      CAST(t.timestamp as DATE) fecha_sin_hora,
86      COUNT(user_id),
87      SUM(amount) suma_transaccions
88  FROM transaction t
89  GROUP BY fecha_sin_hora
90  HAVING COUNT(user_id)>=5

```

Result Grid | Filter Rows: | Export: | W

fecha_sin_hora	COUNT(user_id)	suma_transaccions
2021-05-09	5	1463.73
2021-06-21	5	1443.11
2021-12-22	5	1266.92
2021-03-23	5	1231.15
2021-04-29	6	1206.66
2021-12-29	5	1149.88
2021-12-13	5	1099.15
2021-04-15	5	1005.69
2021-07-06	5	719.71

Es pot comprovar que el dia 09-05-21 és el dia que les 5 vendes van ser més costoses respecte els altres dies que es van realitzar cinc vendes. I el cost total de les 5 vendes va ser de 1463,73 unitats.


3.2.2 Detall de la solució del dia de les 5 vendes més costoses

El detall d'aquestes 5 vendes es podria veure amb la següent consulta i ordre. No s'indica el detall de la companyia que s'ha fet cada venda ja que tampoc ho demanava l'enunciat.

```

91  /*EXERCICI 2 Detall del dia de 5 ventes més costoses---
92  •  SELECT
93      t.*
94  FROM
95  (SELECT
96      CAST(t.timestamp as DATE)  fecha_sin_hora,
97      user_id,
98      amount
99  FROM transactions.transaction t
100 WHERE CAST(t.timestamp as DATE) = '20210509') T
101 ORDER BY t.amount DESC;

```

Result Grid |  Filter Rows: | Export:  | Wrap Cell Content

fecha_sin_hora	user_id	amount
2021-05-09	272	413.50
2021-05-09	275	400.63
2021-05-09	163	371.35
2021-05-09	107	158.89
2021-05-09	275	119.36

3.3 Exercici 3

En quin dia es van realitzar les cinc ventes de menor valor? Mostra la data de la transacció i la sumatòria de la quantitat de diners.

3.3.1 Consulta SQL en Workbench i resultats

En la següent consulta es pot comprovar, que 06-07-2021 va ser el dia que les 5 ventes van ser menys costoses. Com l'exercici anterior podrien veure el detall d'aquestes 5 transaccions.

```

5  /*EXERCICI 3 -----
6  •  SELECT
7    CAST(t.timestamp as DATE) fecha_sin_hora,
8    COUNT(user_id),
9    SUM(amount) suma_transaccions
10   FROM transaction t
11   GROUP BY fecha_sin_hora
12   HAVING COUNT(user_id)>=5
13   ORDER by suma_transaccions;
14

```

fecha_sin_hora	COUNT(user_id)	suma_transaccions
2021-07-06	5	719.71
2021-04-15	5	1005.69
2021-12-13	5	1099.15
2021-12-29	5	1149.88
2021-04-29	6	1206.66
2021-03-23	5	1231.15
2021-12-22	5	1266.92
2021-06-21	5	1443.11
2021-05-09	5	1463.73

3.3.2 Detall de les vendes del dia de les 5 vendes menys costoses

```

115  /*EXERCICI 3 Detall del dia de 5 vendes menors-----
116  •  SELECT
117    t.*
118    FROM
119    (SELECT
120     CAST(t.timestamp as DATE) fecha_sin_hora,
121     user_id,
122     amount
123     FROM transaction t

```

fecha_sin_hora	user_id	amount
2021-07-06	258	26.96
2021-07-06	154	29.63
2021-07-06	272	151.32
2021-07-06	131	195.06
2021-07-06	267	316.74

3.4 Exercici 4

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

3.4.1 Consulta SQL en Workbench i resultats

Com que a la taula *transaction* no hi ha directament el camp *country* he de fer un INNER JOIN tal com hem explicat anteriorment entre els camps relacionats entre les dues taules *company* i *transaction*. Ara puc fer una agrupació de *country* amb la suma de despeses per país, i faig una ordenació descendent d'aquesta despesa mitjana per país.

```

27  /*EXERCICI 4 -----
28  •  SELECT
29      country,
30      /*count(user_id) num_transaccions,*/
31      /*SUM(amount) despesa_total_transaccions,*/
32      AVG(amount) despesa_mitjana_transaccions
33  FROM transaction
34  INNER JOIN company on
35      transaction.company_id=company.id
36  GROUP BY country
37  ORDER by AVG(amount) DESC;
38
39  /*EXERCICI 4 Per comprovar que l'ordre de les taules no i

```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
country	despesa_mitjana_transaccions			
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			

4 Nivell 3

4.1 Exercici 1

Presenta el nom, telèfon i país de les companyies, juntament amb la quantitat total gastada, d'aquelles que van realitzar transaccions amb una despesa compresa entre 100 i 200 euros. Ordena els resultats de major a menor quantitat gastada.

4.1.1 Consulta SQL en Workbench i resultats

Aquest exercici, primer he de trobar les transaccions que estan entre 100 i 200, amb els camps que s'indica, nom companyia, telèfon, país i despesa gastada.

Aquesta consulta, genera una taula que la nomenem t. Sobre aquesta taula fem una nova consulta SELECT on sumarem totes les despeses per companyia, i les ordenem de més despesa a menys tal com ens demana l'enunciat.

```

19  /*-----NIVELL 3-----*/
20  /*EXERCICI 1-----*/
21  •  SELECT
22      t.company_name,
23      t.phone,
24      t.country,
25      SUM(amount)
26  FROM (SELECT
27      company_name,
28      phone,
29      country,
30      amount
31  FROM transaction
32  INNER JOIN company on
33      transaction.company_id=company.id
34  WHERE amount BETWEEN 100 AND 200
35  ORDER BY company_name DESC) t
36  GROUP BY t.company_name,t.phone, t.country
37  ORDER BY SUM(amount) DESC;

```

company_name	phone	country	SUM(amount)
Enim Condimentum Ltd	09 55 51 66 25	United Kingdom	2747.41
Nunc Interdum Incorporated	05 18 15 48 13	Germany	2412.55
Ut Semper Foundation	01 60 36 33 06	Sweden	2285.80
Arcu LLP	06 46 04 41 45	Norway	1589.00
Lorem Eu Incorporated	01 83 66 62 07	Canada	1544.61
Malesuada PC	01 74 85 68 70	Ireland	1255.22
Non Institute	06 77 15 31 14	United Kingdom	1078.31
At Associates	09 56 61 10 65	New Zealand	368.81
Cras Vehicula Aliquet Industries	03 37 86 87 75	Netherlands	339.92

4.2 Exercici 2

Indica el nom de les companyies que van fer compres el 16 de març del 2022, 28 de febrer del 2022 i 13 de febrer del 2022.

4.2.1 Consulta SQL en Workbench i resultats

Novament hem de fer un INNER JOIN entre les dos taules dels dos camps relacionats, filtrar amb WHERE per les dates que ens diu l'enuncia de la taula transactions i de tots els noms de companyies generats, traïem els duplicats amb la comanda DISTINCT.

```
69  /*EXERCICI 3-----*/
70  •  SELECT
71  DISTINCT company_name
72  FROM transaction t
73  INNER JOIN company c ON
74  t.company_id=c.id
75  WHERE CAST(t.timestamp as DATE) = '20220316' OR CAST(t.timestamp as DATE) = '20220228' OR CAST(t.timestamp as DATE) = '20220213';
76
77
```

result Grid	Filter Rows:	Export:	Wrap Cell Content:
company_name			
Sed LLC			
Arcu LLP			
Nunc Interdum Incorporated			
Ut Semper Foundation			
Lorem Eu Incorporated			
Malesuada PC			